



SDP

Version 3.26STD5

Complete API Reference

| | |
|--------------------|---|
| Filename | SDP_API_CompleteAPIReference_3.26STD5_pub.pdf |
| Document Nr | SDP-API |
| Date | 2019-10-17 |
| Author(s) | KUDELSKI SA |
| Information domain | |
| Data Owner | Nagra Media UK Limited |

KUDELSKI SA

22-24, Route de Geneve, Case Postale 134, 1033 Cheseaux , Switzerland

tel: [+41 21 732 01 01](tel:+41217320101), +41 21 732 01 00

<http://www.nagra.com/>

All trademarks and registered trademarks are the property of their respective owners.

This document is supplied with an understanding that the notice(s) herein or any other contractual agreement(s) made that instigated the delivery of a hard copy, electronic copy, facsimile or file transfer of this document are strictly observed and maintained.

The information contained in this document is subject to change without notice.

Security Policy of Nagra Media UK Limited Kudelski Group

Any recipient of this document, without exception, is subject to a Non-Disclosure Agreement (NDA) and access authorization.

Tracking data

Contributors

| Event | Actor | Date | Notes |
|----------|--------------|------------|-------|
| reviewer | David Linton | 2014-01-30 | |
| approver | David Linton | 2014-01-30 | |

Reviewers

| Reviewer | Date | Notes |
|----------|------|-------|
| | | |

Approvers

| Approver | Date | Notes |
|--------------|------------|-------|
| David Linton | 2014-01-30 | |

Contents

| | |
|---|-------------------|
| 1 Introduction | 53 |
| 1.1 References | 53 |
| 2 accessPointService | 54 |
| 2.1 Access Point | 54 |
| 2.1.1 AccessPoint | 54 |
| 2.2 accessPointService | 56 |
| 2.2.1 accessPointService.addCatalogue | 57 |
| 2.2.2 accessPointService.addDeviceGroup | 57 |
| 2.2.3 accessPointService.addPolicyGroup | 58 |
| 2.2.4 accessPointService.addServiceProvider | 58 |
| 2.2.5 accessPointService.create | 59 |
| 2.2.6 accessPointService.delete | 59 |
| 2.2.7 accessPointService.getAccessPointByCat | 60 |
| 2.2.8 accessPointService.getAccessPointsByPolicyGroup | 60 |
| 2.2.9 accessPointService.getAccessPointsBySPUID | 61 |
| 2.2.10 accessPointService.getAllAccessPoints | 61 |
| 2.2.11 accessPointService.getByOrigin | 62 |
| 2.2.12 accessPointService.getByUID | 62 |
| 2.2.13 accessPointService.getDeviceGroups | 63 |
| 2.2.14 accessPointService.getServiceProvidersByAPUID | 63 |
| 2.2.15 accessPointService.removeCatalogue | 64 |
| 2.2.16 accessPointService.removeDeviceGroup | 65 |
| 2.2.17 accessPointService.removePolicyGroup | 65 |
| 2.2.18 accessPointService.removeServiceProvider | 66 |
| 2.2.19 accessPointService.update | Updated 66 |
| 3 accountAggregateService | 68 |
| 3.1 accountAggregateService | 68 |
| 3.1.1 accountAggregateService.addAccountProfile | 68 |

| | |
|--|-------------------|
| 3.1.2 accountAggregateService.createAccount | 69 |
| 3.1.3 accountAggregateService.deleteAccount | 69 |
| 3.1.4 accountAggregateService.getAccountProfileFeatures. | New 70 |
| 3.1.5 accountAggregateService.removeAccountProfile | 71 |
| 3.1.6 accountAggregateService.updateAccount | 72 |
| 4 accountPaymentCredentialsService | 73 |
| 4.1 AccountPaymentCredentials | 73 |
| 4.1.1 AccountPaymentCredentials | 73 |
| 4.2 accountPaymentCredentialsService | 74 |
| 4.2.1 accountPaymentCredentialsService.create | 74 |
| 4.2.2 accountPaymentCredentialsService.createAccountPaymentCredentials | 75 |
| 4.2.3 accountPaymentCredentialsService.delete | 75 |
| 4.2.4 accountPaymentCredentialsService.deleteAccountPaymentCredentials | 76 |
| 4.2.5 accountPaymentCredentialsService.getAccountPaymentCredentials | 76 |
| 4.2.6 accountPaymentCredentialsService.getByOrigin | 77 |
| 4.2.7 accountPaymentCredentialsService.getByUID | 77 |
| 4.2.8 accountPaymentCredentialsService.update | 78 |
| 4.2.9 accountPaymentCredentialsService.updateAccountPaymentCredentials | 78 |
| 5 accountService | 79 |
| 5.1 tv.quative.service.account.model.ApplicabilityPeriod | 79 |
| 5.2 accountService | 80 |
| 5.2.1 accountService.addQumRole | 80 |
| 5.2.2 accountService.create | 81 |
| 5.2.3 accountService.delete | Updated 81 |
| 5.2.4 accountService.getAccByUserUID | 81 |
| 5.2.5 accountService.getAccNoByRolloutProfileUid | 82 |
| 5.2.6 accountService.getApplicabilityPeriod | 82 |
| 5.2.7 accountService.getByAccNoAccPassWithoutServiceProvider | 83 |
| 5.2.8 accountService.getByAccountNumber | 83 |

| | | |
|----------|--|-------------------|
| 5.2.9 | accountService.getByAccountNumberAndServiceProviderID | 84 |
| 5.2.10 | accountService.getByAccountNumberAnyServiceProvider | 84 |
| 5.2.11 | accountService.getByAccountUIDWithoutSpid | 85 |
| 5.2.12 | accountService.getByCreationDate | 86 |
| 5.2.13 | accountService.getByOrigin | 86 |
| 5.2.14 | accountService.getByStatus | 87 |
| 5.2.15 | accountService.getByUID | 87 |
| 5.2.16 | accountService.getCountByRolloutProfileUid | 88 |
| 5.2.17 | accountService.getCreditUsageResetDate | 89 |
| 5.2.18 | accountService.deleteMpRemainingCountByUID | New 89 |
| 5.2.19 | accountService.getRecltemsCreditUsageByAccountUID | 90 |
| 5.2.20 | accountService.getRoleUIDListByAccountUID | 90 |
| 5.2.21 | accountService.purgeAccount. | New 91 |
| 5.2.22 | accountService.removeAllQumRoles | 92 |
| 5.2.23 | accountService.removeQumRole | 93 |
| 5.2.24 | accountService.update | 94 |
| 6 | acquiredContentListService | 95 |
| 6.1 | acquiredContentListService | 95 |
| 6.1.1 | acquiredContentListService.create | 95 |
| 6.1.2 | acquiredContentListService.create | 95 |
| 6.1.3 | acquiredContentListService.delete | 96 |
| 6.1.4 | acquiredContentListService.delete | 96 |
| 6.1.5 | acquiredContentListService.getALCTypeACLByAccountUidAndSmartCardId | 96 |
| 6.1.6 | acquiredContentListService.getByAccountUID | 97 |
| 6.1.7 | acquiredContentListService.getByAccountUIDAndItemType | Updated 98 |
| 6.1.8 | acquiredContentListService.getByAccountUIDAndItemTypeWithDate | 99 |
| 6.1.9 | acquiredContentListService.getByAccountUIDWithDate | 100 |
| 6.1.10 | acquiredContentListService.getByAccountUIDWithDateAndStatus | 100 |
| 6.1.11 | acquiredContentListService.getByItemAndAccountUID | 101 |

| | | |
|----------|--|--------------------|
| 6.1.12 | acquiredContentListService.getByItemAndAccountUIDSmartCard | 102 |
| 6.1.13 | acquiredContentListService.getByItemAndAccountUIDWithDate | 103 |
| 6.1.14 | acquiredContentListService.getByItemAndAccountUIDWithDateSmartCard | 103 |
| 6.1.15 | acquiredContentListService.getByItemAndSmartcard | 104 |
| 6.1.16 | acquiredContentListService.getByItemAndSmartcardWithDate | 105 |
| 6.1.17 | acquiredContentListService.getByItemTypeAndSmartcardValidForDates | 105 |
| 6.1.18 | acquiredContentListService.getByOrigin | 106 |
| 6.1.19 | acquiredContentListService.getByPurchasedItemType | 107 |
| 6.1.20 | acquiredContentListService.getBySmartcard | 107 |
| 6.1.21 | acquiredContentListService.getByUID | 108 |
| 6.1.22 | acquiredContentListService.getByUserUID | 108 |
| 6.1.23 | acquiredContentListService.getByUserUIDAndItemType | 109 |
| 6.1.24 | acquiredContentListService.getByUserUIDAndItemTypeWithDate | 110 |
| 6.1.25 | acquiredContentListService.getByUserUIDWithDate | 111 |
| 6.1.26 | acquiredContentListService.getLiveSubscriptions | 111 |
| 6.1.27 | acquiredContentListService.getValidACIByItemAndAccountUID | 112 |
| 6.1.28 | acquiredContentListService.update | Updated 113 |
| 6.1.29 | acquiredContentListService.update | 113 |
| 7 | Ad Management Service | Updated 114 |
| 7.1 | tv.quative.service.advertisement.adm.model.AdPlayType | 114 |
| 7.2 | tv.quative.service.cod.model.CodItem | 114 |
| 7.3 | tv.quative.service.cod.model.ExpiryDuration | 120 |
| 7.4 | adManagementService | 121 |
| 7.4.1 | adManagementService.addAnAdvertisement | 121 |
| 7.4.2 | adManagementService.addBulkAdvertisements | 121 |
| 7.4.3 | adManagementService.addPostAdvertisement | 122 |
| 7.4.4 | adManagementService.addPreAdvertisement | 122 |
| 7.4.5 | adManagementService.addPreAndPostAdvertisements | 123 |
| 7.4.6 | adManagementService.deleteAdvertisements | 124 |

| | |
|---|------------|
| 7.4.7 adManagementService.deleteAnAdvertisement | 124 |
| 7.4.8 adManagementService.deleteBulkAdvertisements | 125 |
| 7.4.9 adManagementService.deletePostAdvertisement | 125 |
| 7.4.10 adManagementService.deletePreAdvertisement | 126 |
| 7.4.11 adManagementService.deletePreAndPostAdvertisements | 127 |
| 7.4.12 adManagementService.getAdvertisementsByDate | 127 |
| 7.4.13 adManagementService.getCurrentAdvertisement | 128 |
| 7.4.14 adManagementService.getCurrentPostAdvertisement | 128 |
| 7.4.15 adManagementService.getCurrentPreAdvertisement | 129 |
| 7.4.16 adManagementService.getPostAdvertisements | 130 |
| 7.4.17 adManagementService.getPostAdvertisementsByDate | 130 |
| 7.4.18 adManagementService.getPreAdvertisements | 131 |
| 7.4.19 adManagementService.getPreAdvertisementsByDate | 131 |
| 8 aLaCarteService | 133 |
| 8.1 tv.quative.service.alacarte.model.ALaCarteProducts | 133 |
| 8.2 Channel | 133 |
| 8.2.1 Channel | 133 |
| 8.3 Package | 140 |
| 8.3.1 Package | 140 |
| 8.4 Profile | 144 |
| 8.4.1 Profile | 144 |
| 8.5 aLaCarteService | 146 |
| 8.5.1 aLaCarteService.deferredUnsubscribe | 146 |
| 8.5.2 aLaCarteService.getAlaCarteProductList | 146 |
| 8.5.3 aLaCarteService.getChannelListByProduct | 147 |
| 8.5.4 aLaCarteService.getPagedAlaCarteProductList | 148 |
| 8.5.5 aLaCarteService.getPagedChannelListByProduct | 149 |
| 8.5.6 aLaCarteService.getPagedProductListByChannel | 150 |
| 8.5.7 aLaCarteService.getProductListByChannel | 151 |

| | |
|---|--------------------|
| 8.5.8 aLaCarteService.subscribeToAlaCarteProduct | 152 |
| 8.5.9 aLaCarteService.unSubscribeAlaCarteProduct | 153 |
| 9 assetService | 156 |
| 9.1 Asset | 156 |
| 9.1.1 Asset | <i>Updated</i> 156 |
| 9.2 tv.quative.service.asset.model.AssetAudioPid | 171 |
| 9.3 assetService | 172 |
| 9.3.1 assetService.addAudioPid | 172 |
| 9.3.2 assetService.addAudioPid | 173 |
| 9.3.3 assetService.addMultiLanguage | 173 |
| 9.3.4 assetService.create | 174 |
| 9.3.5 assetService.delete | 174 |
| 9.3.6 assetService.delete | 175 |
| 9.3.7 assetService.getAssetAudioPids | 175 |
| 9.3.8 assetService.getAssetListByTrailerUID | 176 |
| 9.3.9 assetService.getAssetsByAssetGroupUID | 176 |
| 9.3.10 assetService.getAssetsByCODCategoryUID | 177 |
| 9.3.11 assetService.getAssetsByCatalogueUID | 177 |
| 9.3.12 assetService.getAssetsByScheduleUID | 178 |
| 9.3.13 assetService.getAvailableAssetListCount | 178 |
| 9.3.14 assetService.getAvailableList | 179 |
| 9.3.15 assetService.getAvailablePagedAssetList | 179 |
| 9.3.16 assetService.getByOrigin | 180 |
| 9.3.17 assetService.getByOriginID | 181 |
| 9.3.18 assetService.getByUID | 181 |
| 9.3.19 assetService.getList | 182 |
| 9.3.20 assetService.getMultiLanguageList | 182 |
| 9.3.21 assetService.getPagedAssetsByAssetGroupUID | 183 |
| 9.3.22 assetService.getPagedAssetsByOriginID | 183 |

| | | |
|-----------|--|------------|
| 9.3.23 | assetService.getPagedAssetsByScheduleUID | 184 |
| 9.3.24 | assetService.getPagedList | 185 |
| 9.3.25 | assetService.getTrailersForAssetUID | 185 |
| 9.3.26 | assetService.increaseTotalRequests | 186 |
| 9.3.27 | assetService.modifyAudioPid | 186 |
| 9.3.28 | assetService.modifyMultiLanguage | 186 |
| 9.3.29 | assetService.removeAudioPid | 187 |
| 9.3.30 | assetService.removeMultiLanguage | 187 |
| 9.3.31 | assetService.update | 188 |
| 9.3.32 | assetService.updateModelAndML | 188 |
| 10 | assetGroupService | 189 |
| 10.1 | Asset Group | 189 |
| 10.1.1 | AssetGroup | 189 |
| 10.2 | tv.quative.service.asset.model.TrickPlay | 195 |
| 10.3 | assetGroupService | 195 |
| 10.3.1 | assetGroupService.getByOrigin | 196 |
| 10.3.2 | assetGroupService.getByUID | 196 |
| 11 | billingService | 198 |
| 11.1 | Billing | 198 |
| 11.1.1 | Billing | 198 |
| 11.2 | billingService | 201 |
| 11.2.1 | billingService.create | 201 |
| 11.2.2 | billingService.delete | 202 |
| 11.2.3 | billingService.getByAccountUID | 202 |
| 11.2.4 | billingService.getByAccountUIDAndRange | 203 |
| 11.2.5 | billingService.getByAccountUIDAndStatus | 203 |
| 11.2.6 | billingService.getByAccountUIDAndUser | 204 |
| 11.2.7 | billingService.getByAccountUIDStatusAndRange | 205 |
| 11.2.8 | billingService.getByAclUID | 205 |

| | |
|---|--------------------|
| 11.2.9 billingService.getByOrigin | 206 |
| 11.2.10 billingService.getByOriginID | 206 |
| 11.2.11 billingService.getByRange | 207 |
| 11.2.12 billingService.getByStatus | 207 |
| 11.2.13 billingService.getByUID | 208 |
| 11.2.14 billingService.getList | 208 |
| 11.2.15 billingService.getMoneySpentByAccountAndDateAndBillType | 209 |
| 11.2.16 billingService.getMoneySpentInLastNDaysByAccountAndBillType | 209 |
| 11.2.17 billingService.update | 210 |
| 12 blockingService | 211 |
| 12.1 Blocking | 211 |
| 12.1.1 Blocking | 211 |
| 12.2 tv.quative.service.blocking.model.ItemType | 213 |
| 12.3 blockingService | <i>Updated</i> 214 |
| 12.3.1 blockingService.create | 214 |
| 12.3.2 blockingService.create | 214 |
| 12.3.3 blockingService.create | 215 |
| 12.3.4 blockingService.delete | 216 |
| 12.3.5 blockingService.delete | 216 |
| 12.3.6 blockingService.getBlockedItemList | 216 |
| 12.3.7 blockingService.getBlockedStatusForItem | 217 |
| 12.3.8 blockingService.getByOrigin | 218 |
| 12.3.9 blockingService.getByOriginUID | 219 |
| 12.3.10 blockingService.getByUID | 219 |
| 12.3.11 blockingService.getList | 220 |
| 12.3.12 blockingService.update | 221 |
| 12.3.13 blockingService.update | 221 |
| 12.3.14 blockingService.update | 222 |
| 13 bocPurchaseService | 223 |

| | |
|---|--------------------|
| 13.1 tv.quative.service.boc.purchase.api.PurchaseFor | 223 |
| 13.2 tv.quative.service.policygroup.api.OfferSpecification | 223 |
| 13.3 tv.quative.service.boc.purchase.api.BocExternalPaymentSignatureRequest | 224 |
| 13.4 tv.quative.service.boc.purchase.api.PaymentDetails | 225 |
| 13.5 tv.quative.service.paygateway.api.ExternalPaymentSignature | 225 |
| 13.6 tv.quative.service.boc.purchase.api.BocPurchaseWithProofOfPaymentRequest | 226 |
| 13.7 tv.quative.service.boc.purchase.api.ProofOfPayment | 226 |
| 13.8 bocPurchaseService | 227 |
| 13.8.1 bocPurchaseService.acquireZeroPricedPolicy | 227 |
| 13.8.2 bocPurchaseService.channelSelectionResume | 228 |
| 13.8.3 bocPurchaseService.channelSelectionStart | 229 |
| 13.8.4 bocPurchaseService.getExternalPaymentSignature | 230 |
| 13.8.5 bocPurchaseService.getPaymentOptionsForAccount | 230 |
| 13.8.6 bocPurchaseService.ngodPavCheck | 231 |
| 13.8.7 bocPurchaseService.otvPavCheck | 231 |
| 13.8.8 bocPurchaseService.purchasePolicy | 232 |
| 13.8.9 bocPurchaseService.purchasePolicyAndGetLicense | 233 |
| 13.8.10 bocPurchaseService.purchasePolicyAndPay | 233 |
| 13.8.11 bocPurchaseService.purchasePolicyPayAndGetLicense | 234 |
| 13.8.12 bocPurchaseService.purchasePolicyWithProofOfPayment | 235 |
| 13.8.13 bocPurchaseService.selectionRestart | 235 |
| 13.8.14 bocPurchaseService.selectionResume | 237 |
| 13.8.15 bocPurchaseService.selectionStart | 238 |
| 14 bookmarkService | 239 |
| 14.1 Bookmark | 239 |
| 14.1.1 Bookmark | <i>Updated</i> 239 |
| 14.2 bookmarkService | <i>Updated</i> 240 |
| 14.2.1 bookmarkService.create | 240 |
| 14.2.2 bookmarkService.delete | 241 |

| | |
|---|------------|
| 14.2.3 bookmarkService.deleteBookmarkByUid | 241 |
| 14.2.4 bookmarkService.getBookmarkForContent | 242 |
| 14.2.5 bookmarkService.getBookmarkForOrigin. | 242 |
| 14.2.6 bookmarkService.getByOrigin | 243 |
| 14.2.7 bookmarkService.getByUID | 244 |
| 14.2.8 bookmarkService.setBookmarkForContent | 244 |
| 14.2.9 bookmarkService.setBookmarkForOrigin. | 245 |
| 14.2.10 bookmarkService.update | 246 |
| 15 btvBrowsingService | 247 |
| 15.1 tv.quative.service.btvbrowsing.api.ChannelRequest | 247 |
| 15.2 tv.quative.service.btvbrowsing.api.Filter | 248 |
| 15.3 tv.quative.service.btvbrowsing.api.OriginSpecification | 249 |
| 15.4 tv.quative.service.btvbrowsing.api.Sort | 249 |
| 15.5 btvBrowsingService | 250 |
| 15.5.1 btvBrowsingService.getChannels | 250 |
| 15.5.2 btvBrowsingService.getChannelsCount | 250 |
| 16 BTVService | 252 |
| 16.1 SPChannelMap | 252 |
| 16.1.1 SPChannelMap | 252 |
| 16.2 BTVService | 253 |
| 16.2.1 BTVService.createSPChannelMap | 253 |
| 16.2.2 BTVService.deleteSPChannelMap | 254 |
| 16.2.3 BTVService.getChannelsByCatAndPkgUID | 254 |
| 16.2.4 BTVService.getChannelsByPkgPattern | 255 |
| 16.2.5 BTVService.getChannelsByPkgPatternAndCategory | 256 |
| 16.2.6 BTVService.getDetEventsByChannelUIDTimeAndType | 257 |
| 16.2.7 BTVService.getDetEvtByChannelListAndKeyword | 257 |
| 16.2.8 BTVService.getDetEvtByKeyword | 258 |
| 16.2.9 BTVService.getDetEvtsByCatAndPkgUIDIndexedByChannel | 259 |

| | |
|--|------------|
| 16.2.10 BTVService.getDetEvtsByChannelList | 260 |
| 16.2.11 BTVService.getDetEvtsByPkgPatternIndexedByChannel | 260 |
| 16.2.12 BTVService.getDetEvtsByPkgUIDIndexedByChannel | 261 |
| 16.2.13 BTVService.getPackages | 262 |
| 16.2.14 BTVService.getPackagesByAccountUID | 262 |
| 16.2.15 BTVService.getPagedChannelsByCatAndPkgUID | 263 |
| 16.2.16 BTVService.getPagedChannelsByPkgPattern | 264 |
| 16.2.17 BTVService.getPagedChannelsByPkgPatternAndCategory | 265 |
| 16.2.18 BTVService.getPagedDetEventsByChannelUIDTimeAndType | 265 |
| 16.2.19 BTVService.getPagedDetEvtsByCatAndPkgUIDIndexedByChannel | 266 |
| 16.2.20 BTVService.getPagedDetEvtsByChannelList | 267 |
| 16.2.21 BTVService.getPagedDetEvtsByPkgUIDIndexedByChannel | 268 |
| 16.2.22 BTVService.getPagedSOCUEventsByIdentifierTimeAndType | 269 |
| 16.2.23 BTVService.getPagedSPChannelsByPkgPattern | 270 |
| 16.2.24 BTVService.getPagedSPChannelsByPkgPatternAndCategory | 271 |
| 16.2.25 BTVService.getSOCUEventsByIdentifierTimeAndType | 272 |
| 16.2.26 BTVService.getSPChannelList | 272 |
| 16.2.27 BTVService.getSPChannelsByChannelUID | 273 |
| 16.2.28 BTVService.getSPChannelsByPkgPattern | 273 |
| 16.2.29 BTVService.getSPChannelsByPkgPatternAndCategory | 274 |
| 16.2.30 BTVService.getSPChannelsByPkgUID | 275 |
| 16.2.31 BTVService.subscribeToChannel | 275 |
| 16.2.32 BTVService.subscribeToPolicyGroup | 276 |
| 16.2.33 BTVService.unsubscribeChannel | 277 |
| 16.2.34 BTVService.unsubscribePackage | 278 |
| 16.2.35 BTVService.updateSPChannelMap | 279 |
| 17 catalogueService | 280 |
| 17.1 Catalogue | 280 |
| 17.1.1 Catalogue | 281 |

| | |
|--|------------|
| 17.2 CatalogueTreeNode | 285 |
| 17.2.1 CatalogueTreeNode | 285 |
| 17.3 catalogueService | 287 |
| 17.3.1 catalogueService.addMultiLanguage | 287 |
| 17.3.2 catalogueService.addNodeToCatalogue | 288 |
| 17.3.3 catalogueService.create | 289 |
| 17.3.4 catalogueService.delete | 289 |
| 17.3.5 catalogueService.getAllCatalogues | 289 |
| 17.3.6 catalogueService.getBaseCataloguesByAcptUid | 290 |
| 17.3.7 catalogueService.getBaseChildCataloguesByAcptUidAndName | 290 |
| 17.3.8 catalogueService.getByOrigin | 291 |
| 17.3.9 catalogueService.getByUID | 292 |
| 17.3.10 catalogueService.getCatBySbscrProfUID | 293 |
| 17.3.11 catalogueService.getCatalogueTreeLevel | 293 |
| 17.3.12 catalogueService.getCataloguesByCatalogueUid | 294 |
| 17.3.13 catalogueService.getCataloguesByNodeUidAndType | 294 |
| 17.3.14 catalogueService.getList | 295 |
| 17.3.15 catalogueService.getMultiLanguageList | 295 |
| 17.3.16 catalogueService.getParentCatalogue | 296 |
| 17.3.17 catalogueService.getSubscribedBaseCatalogues | 296 |
| 17.3.18 catalogueService.modifyMultiLanguage | 297 |
| 17.3.19 catalogueService.removeMultiLanguage | 297 |
| 17.3.20 catalogueService.removeNodeFromCatalogue | 298 |
| 17.3.21 catalogueService.update | 298 |
| 17.3.22 catalogueService.updateCatalogueTreeLevel | 299 |
| 17.3.23 catalogueService.updateModelAndML | 299 |
| 17.3.24 catalogueService.updateOrder | 300 |
| 18 categoryService | 301 |
| 18.1 Category | 301 |

| | |
|---|------------|
| 18.1.1 Category | 301 |
| 18.2 categoryService | 302 |
| 18.2.1 categoryService.addChannel | 302 |
| 18.2.2 categoryService.addMultiLanguage | 303 |
| 18.2.3 categoryService.addMultiLanguage | 304 |
| 18.2.4 categoryService.create | 304 |
| 18.2.5 categoryService.delete | 304 |
| 18.2.6 categoryService.getAllCategories | 305 |
| 18.2.7 categoryService.getByOriginUID | 305 |
| 18.2.8 categoryService.getByUID | 306 |
| 18.2.9 categoryService.getMultiLanguageList | 307 |
| 18.2.10 categoryService.getPagedCategories | 307 |
| 18.2.11 categoryService.getPagedCategoriesByOriginUID | 308 |
| 18.2.12 categoryService.modifyMultiLanguage | 309 |
| 18.2.13 categoryService.removeChannel | 309 |
| 18.2.14 categoryService.removeMultiLanguage | 310 |
| 18.2.15 categoryService.removeMultiLanguage | 310 |
| 18.2.16 categoryService.update | 311 |
| 18.2.17 categoryService.updateModelAndML | 311 |
| 19 cdnService | 312 |
| 19.1 CDNService | 312 |
| 19.1.1 CDNService.getCdnUrl | 312 |
| 19.1.2 CDNService.validateToken | 313 |
| 20 cdnTokenService | 314 |
| 20.1 CdnToken | 314 |
| 20.1.1 CdnToken | 314 |
| 20.2 cdnTokenService | 315 |
| 20.2.1 cdnTokenService.getByOrigin | 315 |
| 20.2.2 cdnTokenService.getByUID | 316 |

| | |
|---|------------|
| 21 challengeAggregateService | 317 |
| 21.1 challengeAggregateService | 317 |
| 21.1.1 challengeAggregateService.acquireSecureChallenge | 317 |
| 21.1.2 challengeAggregateService.acquireSecureChallengeWithAdditionalInfoAndContext | 317 |
| 21.1.3 challengeAggregateService.acquireSecureChallengeWithContext | 318 |
| 21.1.4 challengeAggregateService.verifySecureChallengeWithContext | 318 |
| 22 challengeService | 320 |
| 22.1 Challenge | 320 |
| 22.1.1 Challenge | 320 |
| 22.2 challengeService | 321 |
| 22.2.1 challengeService.create | 321 |
| 22.2.2 challengeService.delete | 322 |
| 22.2.3 challengeService.getByChallengeID | 322 |
| 22.2.4 challengeService.getByDeviceUID | 323 |
| 22.2.5 challengeService.getByOrigin | 323 |
| 22.2.6 challengeService.getByUID | 324 |
| 22.2.7 challengeService.update | 324 |
| 23 channelService | 326 |
| 23.1 tv.quative.service.channel.model.ChannelUIDSet | 326 |
| 23.2 channelService | 326 |
| 23.2.1 channelService.addMultiLanguage | 326 |
| 23.2.2 channelService.create | 327 |
| 23.2.3 channelService.delete | 327 |
| 23.2.4 channelService.getAllAuthorizedChannelsForAccount | 328 |
| 23.2.5 channelService.getAllAuthorizedChannelsForDevice | 328 |
| 23.2.6 channelService.getAllChannels | 328 |
| 23.2.7 channelService.getAllChannelsAllLocales | 329 |
| 23.2.8 channelService.getAllChannelsByLocale | 329 |
| 23.2.9 channelService.getAuthorizedChannelsForAccountByChannelUIDSet | 330 |

| | | |
|-----------|--|------------|
| 23.2.10 | channelService.getAuthorizedChannelsForDeviceByChannelUIDSet | 330 |
| 23.2.11 | channelService.getByCasID | 331 |
| 23.2.12 | channelService.getByCatUID | 331 |
| 23.2.13 | channelService.getByDrmID | 332 |
| 23.2.14 | channelService.getByDrmIdAnyServiceProvider | 333 |
| 23.2.15 | channelService.getByNumber | 333 |
| 23.2.16 | channelService.getByOrigin | 334 |
| 23.2.17 | channelService.getByOriginUID | 334 |
| 23.2.18 | channelService.getByPkgPattern | 335 |
| 23.2.19 | channelService.getByPkgUID | 336 |
| 23.2.20 | channelService.getByUID | 336 |
| 23.2.21 | channelService.getChannelType | 337 |
| 23.2.22 | nvodChannelService.getMultiLanguageList | 338 |
| 23.2.23 | channelService.getPagedChannelsByCatUID | 338 |
| 23.2.24 | channelService.getPagedChannelsByOriginUID | 339 |
| 23.2.25 | channelService.getPagedChannelsByPkgPattern | 339 |
| 23.2.26 | channelService.getPagedChannelsByPkgUID | 340 |
| 23.2.27 | channelService.modifyMultiLanguage | 340 |
| 23.2.28 | channelService.removeMultiLanguage | 341 |
| 23.2.29 | channelService.update | 341 |
| 23.2.30 | channelService.updateModelAndML | 342 |
| 24 | codBrowsingService | 343 |
| 24.1 | tv.quative.service.codbrowsing.api.Filter | 343 |
| 24.2 | tv.quative.service.codbrowsing.api.Filter.AssetPlayType | 350 |
| 24.3 | tv.quative.service.codbrowsing.api.Filter.OfferFrequencyType | 350 |
| 24.4 | tv.quative.service.codbrowsing.api.Filter.ScheduleStatus | 351 |
| 24.5 | tv.quative.service.codbrowsing.api.ScheduledItemRequest | 351 |
| 24.6 | tv.quative.service.codbrowsing.api.Sort | 352 |
| 24.7 | tv.quative.service.codbrowsing.api.TrailerFilter | 354 |

| | |
|---|------------|
| 24.8 tv.quative.service.codbrowsing.api.TrailerItemRequest | 355 |
| 24.9 codBrowsingService | 355 |
| 24.9.1 codBrowsingService.getCurrentContext | 355 |
| 24.9.2 codBrowsingService.getScheduledAssetGroups | 356 |
| 24.9.3 codBrowsingService.getScheduledAssetGroupsCount | 356 |
| 24.9.4 codBrowsingService.getScheduledAssetGroupsForCurrentContext | 357 |
| 24.9.5 codBrowsingService.getScheduledAssetGroupsForCurrentContextCount | 357 |
| 24.9.6 codBrowsingService.getScheduledItems | 358 |
| 24.9.7 codBrowsingService.getScheduledItemsCount | 358 |
| 24.9.8 codBrowsingService.getScheduledItemsForCurrentContext | 359 |
| 24.9.9 codBrowsingService.getScheduledItemsForCurrentContextCount | 359 |
| 24.9.10 codBrowsingService.getTrailerItems | 360 |
| 24.9.11 codBrowsingService.getTrailerItemsForCurrentContext | 360 |
| 24.9.12 codBrowsingService.testContext | 361 |
| 25 codCategoryService | 362 |
| 25.1 COD Category | 362 |
| 25.1.1 CODCategory | 362 |
| 25.2 codCategoryService | 363 |
| 25.2.1 codCategoryService.addAsset | 364 |
| 25.2.2 codCategoryService.addAssetGroup | 364 |
| 25.2.3 codCategoryService.addMultiLanguage | 365 |
| 25.2.4 codCategoryService.create | 365 |
| 25.2.5 codCategoryService.delete | 366 |
| 25.2.6 codCategoryService.getAssetGroupUIDList | 366 |
| 25.2.7 codCategoryService.getAssetUIDList | 367 |
| 25.2.8 codCategoryService.getByName | 367 |
| 25.2.9 codCategoryService.getByOrigin | 368 |
| 25.2.10 codCategoryService.getByUID | 369 |
| 25.2.11 codCategoryService.getCategoryList | 369 |

| | | |
|-----------|---|----------------|
| 25.2.12 | codCategoryService.getCategoryListByItemUidAndType | 370 |
| 25.2.13 | codCategoryService.getMultiLanguageList | 370 |
| 25.2.14 | codCategoryService.getPagedCategoryList | 371 |
| 25.2.15 | codCategoryService.modifyMultiLanguage | 372 |
| 25.2.16 | codCategoryService.modifyMultiLanguage | 372 |
| 25.2.17 | codCategoryService.removeAsset | 373 |
| 25.2.18 | codCategoryService.removeAssetGroup | 373 |
| 25.2.19 | codCategoryService.removeMultiLanguage | 374 |
| 25.2.20 | codCategoryService.removeMultiLanguage | 375 |
| 25.2.21 | codCategoryService.update | 375 |
| 25.2.22 | codCategoryService.updateModelAndML | 375 |
| 26 | codHelperService | 377 |
| 26.1 | codHelperService | 377 |
| 26.1.1 | codHelperService.hasPurchaseAllAvailableForCatAndTvodAssetGroup | 377 |
| 27 | codScheduleService | New 379 |
| 27.1 | codScheduleService | New 379 |
| 27.1.1 | codScheduleService.getSchedulesByPolicyGroupUid. | New 379 |
| 28 | CODService | 381 |
| 28.1 | tv.quative.service.cod.model.PurchasedContent | 381 |
| 28.2 | CODService | 381 |
| 28.2.1 | CODService.getAssetGroupAssetCountByCat | 381 |
| 28.2.2 | CODService.getCatalogueChildCatalogueNodeWrappers | 382 |
| 28.2.3 | CODService.getCatalogueChildCatalogueNodes | 383 |
| 28.2.4 | CODService.getCataloguesByAccountUID | 383 |
| 28.2.5 | CODService.getPagedAssetGroupsByCatAndGenre | 384 |
| 28.2.6 | CODService.getPagedCatalogueChildCatalogueNodes | 385 |
| 28.2.7 | CODService.getPagedPolicyForCatAndAssetGroupAndAsset | 385 |
| 28.2.8 | CODService.getSubscribedBaseCatalogues | 386 |
| 28.2.9 | CODService.getSubscribedPolicyGroups | 387 |

| | | |
|-----------|--|--------------------|
| 28.2.10 | CODService.isSubscribed | 388 |
| 28.2.11 | CODService.isSubscribedSpidKnown | 389 |
| 28.2.12 | CODService.subscribeToAssetGroup | 390 |
| 28.2.13 | CODService.subscribeToPolicyGroup | 390 |
| 28.2.14 | CODService.unsubscribePolicyGroup | 391 |
| 29 | contentConsumptionService | 393 |
| 29.1 | ContentConsumption | 393 |
| 29.1.1 | ContentConsumption ContentConsumption | 393 |
| 29.2 | contentConsumptionService | 395 |
| 29.2.1 | contentConsumptionService.create | 395 |
| 29.2.2 | contentConsumptionService.create | 395 |
| 29.2.3 | contentConsumptionService.create | 395 |
| 29.2.4 | contentConsumptionService.delete | 397 |
| 29.2.5 | contentConsumptionService.getByOrigin | 397 |
| 29.2.6 | contentConsumptionService.getByUID | 397 |
| 29.2.7 | contentConsumptionService.update | 398 |
| 30 | contentRatingService | Updated 399 |
| 30.1 | tv.quative.service.contentrating.model.AverageRating | 399 |
| 30.2 | contentRatingService | 399 |
| 30.2.1 | contentRatingService.getAverageAssetRating | 399 |
| 30.2.2 | contentRatingService.getAverageAssetRatings | 400 |
| 30.2.3 | contentRatingService.getAverageProgramRating | 400 |
| 30.2.4 | contentRatingService.getAverageProgramRatings | 401 |
| 30.2.5 | contentRatingService.getUserAssetRatings | 401 |
| 30.2.6 | contentRatingService.rateAsset | 402 |
| 30.2.7 | contentRatingService.rateProgram | 402 |
| 31 | contextService | 404 |
| 31.1 | tv.quative.service.profile.model.ProfileType | 404 |
| 31.2 | contextService | 404 |

| | |
|---|--------------------|
| 31.2.1 contextService.getCurrentContext | <i>Updated</i> 405 |
| 31.2.2 contextService.getProfilesForFeatures | 405 |
| 32 cpvrOperationService | 406 |
| 32.1 tv.quative.service.cpvrOperation.model.CpvrOperation | 406 |
| 32.2 tv.quative.service.cpvrOperation.model.Source | 408 |
| 32.3 tv.quative.service.cpvrOperation.model.Status | 408 |
| 32.4 tv.quative.service.cpvrOperation.model.Type | 408 |
| 32.5 cpvrOperationService | 409 |
| 32.5.1 cpvrOperationService.delete | 409 |
| 32.5.2 cpvrOperationService.getByAccount | 409 |
| 32.5.3 cpvrOperationService.getByAccountVerbose | 410 |
| 32.5.4 cpvrOperationService.getByUID | 411 |
| 32.5.5 cpvrOperationService.update | 411 |
| 33 deviceAggregateService | 413 |
| 33.1 Media Player | 413 |
| 33.1.1 MediaPlayer | <i>Updated</i> 413 |
| 33.2 Set Top Box | 415 |
| 33.2.1 SetTopBox | <i>Updated</i> 416 |
| 33.3 tv.quative.service.device.model.SetTopBox.Status | 420 |
| 33.4 tv.quative.service.device.model.SmartCardType | 421 |
| 33.5 deviceAggregateService | 421 |
| 33.5.1 deviceAggregateService.assignSmartcardToCas | 421 |
| 33.5.2 deviceAggregateService.createMp | 422 |
| 33.5.3 deviceAggregateService.createSTB | 422 |
| 33.5.4 deviceAggregateService.deleteMp | 423 |
| 33.5.5 deviceAggregateService.deleteSTB | 423 |
| 33.5.6 deviceAggregateService.getvUA | <i>New</i> 424 |
| 33.5.7 deviceAggregateService.swapHybridSTB | 424 |
| 33.5.8 deviceAggregateService.swapSTBOrSmrtCrds | 425 |

| | |
|---|-----------------------|
| 33.5.9 deviceAggregateService.updateDeviceWithNewAccount | 426 |
| 33.5.10 deviceAggregateService.updateMp | 426 |
| 33.5.11 deviceAggregateService.updateMpState | 427 |
| 33.5.12 deviceAggregateService.updateSTB | 427 |
| 33.5.13 deviceAggregateService.updateSmartCard | 428 |
| 33.5.14 deviceAggregateService.updateSmartCardWithCasInstanceId | 428 |
| 33.5.15 deviceAggregateService.setNetworkBySetTopBox | New 429 |
| 34 deviceClassifierService | New 430 |
| 34.1 DeviceClassifier | New 430 |
| 34.2 deviceClassifierService.create | New 433 |
| 34.3 deviceClassifierService.delete. | New 433 |
| 35 deviceGroupService | 434 |
| 35.1 Device Group | 434 |
| 35.1.1 DeviceGroup | 434 |
| 35.2 devicegroupService | 435 |
| 35.2.1 devicegroupService.addDevice | 436 |
| 35.2.2 devicegroupService.create | 436 |
| 35.2.3 devicegroupService.delete | 437 |
| 35.2.4 devicegroupService.getAllDeviceGroups | 437 |
| 35.2.5 devicegroupService.getByAccessPointUID | 437 |
| 35.2.6 devicegroupService.getByOrigin | 438 |
| 35.2.7 devicegroupService.getByUID | 439 |
| 35.2.8 devicegroupService.getDeviceGroupByDeviceUID | 439 |
| 35.2.9 devicegroupService.getDevices | 440 |
| 35.2.10 devicegroupService.getDevicesByDeviceUID | 440 |
| 35.2.11 devicegroupService.getStreamingServerDeviceGroups | 441 |
| 35.2.12 devicegroupService.removeDevice | 441 |
| 35.2.13 devicegroupService.update | 442 |
| 36 deviceService | 443 |

| | |
|---|-----|
| 36.1 Application Server | 443 |
| 36.1.1 ApplicationServer | 443 |
| 36.2 ERM | 444 |
| 36.2.1 Erm | 445 |
| 36.3 QAM | 447 |
| 36.3.1 Qam | 447 |
| 36.4 Simulcrypt Synchronizer | 449 |
| 36.4.1 Scs | 449 |
| 36.5 Streaming Server | 451 |
| 36.5.1 StreamingServer | 451 |
| 36.6 tv.quative.service.device.model.DeviceStatus | 453 |
| 36.7 deviceService | 453 |
| 36.7.1 deviceService.createAppServer | 453 |
| 36.7.2 deviceService.createErm | 453 |
| 36.7.3 deviceService.createQam | 454 |
| 36.7.4 deviceService.createSTB | 454 |
| 36.7.5 deviceService.createSTBWithServiceProvider | 455 |
| 36.7.6 deviceService.createScs | 455 |
| 36.7.7 deviceService.createStreamingServer | 455 |
| 36.7.8 deviceService.deleteDeviceByUID | 456 |
| 36.7.9 deviceService.getAvailableSetTopBox | 456 |
| 36.7.10 deviceService.getBySerialNumber | 457 |
| 36.7.11 deviceService.getBySmartcardIDAnyServiceProvider | 457 |
| 36.7.12 deviceService.getDeviceByCASN | 458 |
| 36.7.13 deviceService.getDeviceByCASNAnyServiceProvider | 459 |
| 36.7.14 deviceService.getDeviceByDevGrpAndType | 459 |
| 36.7.15 deviceService.getDeviceByMACAddress | 460 |
| 36.7.16 deviceService.getDeviceByMACAddressAnyServiceProvider | 460 |
| 36.7.17 deviceService.getDeviceByNUID | 461 |

| | |
|---|------------|
| 36.7.18 deviceService.getDeviceByPhysicalAddress | 462 |
| 36.7.19 deviceService.getDeviceBySmartcardID | 462 |
| 36.7.20 deviceService.getDeviceByType | 463 |
| 36.7.21 deviceService.getDeviceByUID | 463 |
| 36.7.22 deviceService.getErmByName | 464 |
| 36.7.23 deviceService.getErmByUID | 464 |
| 36.7.24 deviceService.getMediaPlayerByAccountUID | 465 |
| 36.7.25 deviceService.getMediaPlayerByAccountUIDAndStatus | 465 |
| 36.7.26 deviceService.getMpById | 466 |
| 36.7.27 deviceService.getNextVirtualSmartcardId | 466 |
| 36.7.28 deviceService.getPrivateData | 467 |
| 36.7.29 deviceService.getQams | 467 |
| 36.7.30 deviceService.getSetTopBoxByAccountUID | 468 |
| 36.7.31 deviceService.getSetTopBoxByAccountUIDAndSpid | 468 |
| 36.7.32 deviceService.getSetTopBoxByExternalGroup | 469 |
| 36.7.33 deviceService.getSetTopBoxByOrigin | 469 |
| 36.7.34 deviceService.getSwapSTBsForAccountUID | 470 |
| 36.7.35 deviceService.setPrivateData | 470 |
| 36.7.36 deviceService.updateAppServer | 471 |
| 36.7.37 deviceService.updateErm | 471 |
| 36.7.38 deviceService.updateMpName | 472 |
| 36.7.39 deviceService.updateMpState | 472 |
| 36.7.40 deviceService.updateQam | 473 |
| 36.7.41 deviceService.updateSTB | 473 |
| 36.7.42 deviceService.updateSTBWithServiceProvider | 474 |
| 36.7.43 deviceService.updateScs | 474 |
| 36.7.44 deviceService.updateStreamingServer | 474 |
| 37 drmworkflowservice | 476 |
| 37.1 drmworkflowservice | 476 |

| | |
|---|------------|
| 37.1.1 drmworkflowservice.getMarlinMs3Token | 476 |
| 38 dynamicAdvertsService | 477 |
| 38.1 tv.quative.service.dynamicads.api.DynamicAdvertsResponse | 477 |
| 38.2 dynamicAdvertsService | 477 |
| 38.2.1 dynamicAdvertsService.getAdverts | 477 |
| 39 EdaConfigurationService | 478 |
| 39.1 FoxtelEdaCas | 478 |
| 39.1.1 FoxtelEdaCas | 478 |
| 39.2 EdaConfigurationService | 478 |
| 39.2.1 EdaConfigurationService.create | 479 |
| 39.2.2 EdaConfigurationService.get | 479 |
| 39.2.3 EdaConfigurationService.remove | 479 |
| 39.2.4 EdaConfigurationService.update | 480 |
| 40 eventService | 481 |
| 40.1 Event | 481 |
| 40.1.1 Event | 481 |
| 40.2 tv.quative.service.event.model.DetailedEvent | 484 |
| 40.3 eventService | 486 |
| 40.3.1 eventService.addPolicyGroup | 486 |
| 40.3.2 eventService.create | 487 |
| 40.3.3 eventService.create | 487 |
| 40.3.4 eventService.delete | 488 |
| 40.3.5 eventService.getByMediaFileName | 488 |
| 40.3.6 eventService.getByOrigin | 489 |
| 40.3.7 eventService.getByPolicyGroupUID | 489 |
| 40.3.8 eventService.getByUID | 490 |
| 40.3.9 eventService.getDetEvtByUID | 490 |
| 40.3.10 eventService.getDetEvtsByAllChnls | 491 |
| 40.3.11 eventService.getDetEvtsByCatUID | 491 |

| | |
|--|-----|
| 40.3.12 eventService.getDetEvtsByChnUID | 492 |
| 40.3.13 eventService.getDetEvtsByChnlAcct | 493 |
| 40.3.14 eventService.getDetEvtsByPkgAcct | 493 |
| 40.3.15 eventService.getDetEvtsByPkgPattern | 494 |
| 40.3.16 eventService.getDetEvtsByPkgUID | 495 |
| 40.3.17 eventService.getDetEvtsMultiLanguageList | 495 |
| 40.3.18 eventService.getEvtsByAllChnls | 496 |
| 40.3.19 eventService.getEvtsByCatUID | 496 |
| 40.3.20 eventService.getEvtsByChnUID | 497 |
| 40.3.21 eventService.getEvtsByChnlAcct | 497 |
| 40.3.22 eventService.getEvtsByOriginUID | 498 |
| 40.3.23 eventService.getEvtsByPgmUID | 498 |
| 40.3.24 eventService.getEvtsByPkgAcct | 499 |
| 40.3.25 eventService.getEvtsByPkgUID | 499 |
| 40.3.26 eventService.getPagedDetailEventsByAllChnls | 500 |
| 40.3.27 eventService.getPagedDetailEventsByCatUID | 500 |
| 40.3.28 eventService.getPagedDetailEventsByChnUID | 501 |
| 40.3.29 eventService.getPagedDetailEventsByChnlAcct | 502 |
| 40.3.30 eventService.getPagedDetailEventsByPkgAcct | 503 |
| 40.3.31 eventService.getPagedDetailEventsByPkgPattern | 503 |
| 40.3.32 eventService.getPagedDetailEventsByPkgUID | 504 |
| 40.3.33 eventService.getPagedDetailEventsMultiLanguageList | 505 |
| 40.3.34 eventService.getPagedEventsByAllChnls | 505 |
| 40.3.35 eventService.getPagedEventsByCatUID | 506 |
| 40.3.36 eventService.getPagedEventsByChnUID | 507 |
| 40.3.37 eventService.getPagedEventsByChnlAcct | 507 |
| 40.3.38 eventService.getPagedEventsByPgmUID | 508 |
| 40.3.39 eventService.getPagedEventsByPkgAcct | 509 |
| 40.3.40 eventService.getPagedEventsByPkgUID | 509 |

| | |
|--|--------------------|
| 40.3.41 eventService.getPagedSOCUEventsByChnIdentifier | 510 |
| 40.3.42 eventService.getPpvEventPolicyGroupByEvent | 510 |
| 40.3.43 eventService.removePolicyGroup | 511 |
| 40.3.44 eventService.update | 511 |
| 40.3.45 eventService.update | 512 |
| 41 favouriteService | 513 |
| 41.1 Favourite | 513 |
| 41.1.1 Favourite | 513 |
| 41.2 favouriteService | 514 |
| 41.2.1 favouriteService.create | <i>Updated</i> 514 |
| 41.2.2 favouriteService.create | 515 |
| 41.2.3 favouriteService.delete | 515 |
| 41.2.4 favouriteService.delete | 516 |
| 41.2.5 favouriteService.getByOrigin | 516 |
| 41.2.6 favouriteService.getByOriginUID | 517 |
| 41.2.7 favouriteService.getByUID | 518 |
| 41.2.8 favouriteService.getFavouriteItemlist | 518 |
| 41.2.9 favouriteService.getFavouriteStatusForItem | 519 |
| 41.2.10 favouriteService.getList | 519 |
| 41.2.11 favouriteService.update | 520 |
| 41.2.12 favouriteService.update | 521 |
| 42 featureService | 522 |
| 42.1 Feature | 522 |
| 42.1.1 Feature | 522 |
| 42.2 tv.quative.service.feature.model.FeatureTypeEnum | 523 |
| 42.3 featureService | 524 |
| 42.3.1 featureService.create | 524 |
| 42.3.2 featureService.delete | 524 |
| 42.3.3 featureService.getByName | 525 |

| | |
|--|----------------|
| 42.3.4 featureService.getByOrigin | 525 |
| 42.3.5 featureService.getByType | 526 |
| 42.3.6 featureService.getByTypeAndNames | 526 |
| 42.3.7 featureService.getByUID | 526 |
| 42.3.8 featureService.getByUids | 527 |
| 42.3.9 featureService.getFeaturesByProfUID | 527 |
| 42.3.10 featureService.update | 528 |
| 42.3.11 featureService.update | 528 |
| 43 geoBlockingService | New 530 |
| 43.1 geoBlockingService.createRegionIPMapping | New 530 |
| 43.2 geoBlockingService.deleteRegionIPMappingByRegionId | New 531 |
| 43.3 geoBlockingService.deleteRegionIPMappingByRegionIdAndCidr | New 531 |
| 43.4 geoBlockingService.getAllRegionIPMappings | New 531 |
| 43.5 geoBlockingService.getAllWhitLists | New 532 |
| 43.6 geoBlockingService.getIPMappingsByRegionId | New 533 |
| 43.7 geoBlockingService.validateClientIpWithRegion | New 533 |
| 44 initialisationService | 535 |
| 44.1 tv.quative.service.favourite.model.ItemType | 535 |
| 44.2 tv.quative.service.initialisation.model.InitData | 535 |
| 44.3 initialisationService | 536 |
| 44.3.1 initialisationService.getInitDataByCASN | 537 |
| 44.3.2 initialisationService.getInitDataByMacAddress | 537 |
| 44.3.3 initialisationService.getInitDataBySmartCardId | 538 |
| 45 intervalService | 539 |
| 45.1 Interval | 539 |
| 45.1.1 Interval | 539 |
| 45.2 intervalService.addMultiLanguage | 541 |
| 45.2.1 intervalService.addMultiLanguage | 541 |
| 45.2.2 intervalService.create | 542 |

| | |
|---|------------|
| 45.2.3 intervalService.create | 542 |
| 45.2.4 intervalService.delete | 543 |
| 45.2.5 intervalService.delete | 543 |
| 45.2.6 intervalService.getByName | 543 |
| 45.2.7 intervalService.getByOrigin | 544 |
| 45.2.8 intervalService.getByType | 545 |
| 45.2.9 intervalService.getByUID | 545 |
| 45.2.10 intervalService.getList | 546 |
| 45.2.11 intervalService.getListByTypeValueFrequency | 546 |
| 45.2.12 intervalService.getMultiLanguageList | 547 |
| 45.2.13 intervalService.modifyMultiLanguage | 548 |
| 45.2.14 intervalService.removeMultiLanguage | 548 |
| 45.2.15 intervalService.removeMultiLanguage | 549 |
| 45.2.16 intervalService.update | 549 |
| 45.2.17 intervalService.update | 550 |
| 45.2.18 intervalService.updateModelAndML | 550 |
| 46 License Services | 551 |
| 46.1 licenseService | 551 |
| 46.1.1 licenseService.getLicense | 551 |
| 46.1.2 licenseService.getLicenseWithAdditionalInfo | 552 |
| 46.1.3 licenseService.getLicenseWithChallengeAuthentication | 553 |
| 46.1.4 licenseService.getLicenseWithContext | 553 |
| 46.1.5 licenseService.linkAuxiliarySTB | 554 |
| 47 linkedSTBService | 555 |
| 47.1 tv.quative.service.linkedstb.api.PairNocsRequest | 555 |
| 47.2 linkedStbService | 555 |
| 47.2.1 linkedStbService.joinHomeDomain | 555 |
| 47.2.2 linkedStbService.pairNocsDevices | 556 |
| 47.2.3 linkedStbService.refurbish | 556 |

| | |
|---|--------------------|
| 48 mdrmService | New 558 |
| 48.1 mdrmService. | New 558 |
| 48.1.1 mdrmService.isContentAuthorised | New 558 |
| 48.1.2 mdrmService.isContentAuthorised | New 562 |
| 48.1.3 mdrmService.isDeviceAuthorised | New 567 |
| 48.1.4 mdrmService.isDeviceAuthorised | New 570 |
| 48.1.5 mdrmService.isHomeDomainAuthorised | New 573 |
| 49 NmpExtendedService | 577 |
| 49.1 NmpExtendedService | 577 |
| 49.1.1 NmpExtendedService.getEntitlements | Updated 577 |
| 49.1.2 NmpExtendedService.initializeDevice | Updated 580 |
| 49.1.3 NmpExtendedService.initializeDevice | New 583 |
| 49.1.4 NmpExtendedService.transformEntitlements | Updated 586 |
| 50 NmpOperatorService | 589 |
| 50.1 NmpOperatorService | 589 |
| 50.1.1 NmpOperatorService.getEntitlements | 589 |
| 50.1.2 NmpOperatorService.initializeDevice | 590 |
| 50.1.3 NmpOperatorService.transformEntitlements | 590 |
| 51 nvodChannelService | 592 |
| 51.1 nvodChannelService | 592 |
| 51.1.1 nvodChannelService.addMultiLanguage | 592 |
| 51.1.2 nvodChannelService.create | 592 |
| 51.1.3 nvodChannelService.delete | 593 |
| 51.1.4 nvodChannelService.getByOrigin | 593 |
| 51.1.5 nvodChannelService.getByOriginUID | 594 |
| 51.1.6 nvodChannelService.getByUID | 595 |
| 51.1.7 nvodChannelService.getList | 595 |
| 51.1.8 nvodChannelService.getMultiLanguageList | 596 |
| 51.1.9 nvodChannelService.isNvodChannel | 596 |

| | |
|--|------------|
| 51.1.10 nvodChannelService.modifyMultiLanguage | 597 |
| 51.1.11 nvodChannelService.removeMultiLanguage | 597 |
| 51.1.12 nvodChannelService.update | 598 |
| 52 nvodProgramService | 599 |
| 52.1 nvodProgramService | 599 |
| 52.1.1 nvodProgramService.addAltAudioPID | 599 |
| 52.1.2 nvodProgramService.addAltAudioPID | 599 |
| 52.1.3 nvodProgramService.addMultiLanguage | 600 |
| 52.1.4 nvodProgramService.addMultiLanguage | 600 |
| 52.1.5 nvodProgramService.create | 601 |
| 52.1.6 nvodProgramService.create | 602 |
| 52.1.7 nvodProgramService.delete | 603 |
| 52.1.8 nvodProgramService.delete | 603 |
| 52.1.9 nvodProgramService.getAltAudioPIDList | 604 |
| 52.1.10 nvodProgramService.getByOrigin | 604 |
| 52.1.11 nvodProgramService.getByOriginUID | 605 |
| 52.1.12 nvodProgramService.getByUID | 605 |
| 52.1.13 nvodProgramService.getMultiLanguageList | 606 |
| 52.1.14 nvodProgramService.getSimpleProgramByUID | 607 |
| 52.1.15 nvodProgramService.modifyAltAudioPID | 607 |
| 52.1.16 nvodProgramService.modifyAltAudioPID | 608 |
| 52.1.17 nvodProgramService.modifyMultiLanguage | 608 |
| 52.1.18 nvodProgramService.modifyMultiLanguage | 609 |
| 52.1.19 nvodProgramService.removeAltAudioPID | 609 |
| 52.1.20 nvodProgramService.removeAltAudioPID | 610 |
| 52.1.21 nvodProgramService.removeMultiLanguage | 610 |
| 52.1.22 nvodProgramService.removeMultiLanguage | 611 |
| 52.1.23 nvodProgramService.update | 611 |
| 52.1.24 nvodProgramService.update | 612 |

| | |
|---|--------------------|
| 53 ottSessionService | 614 |
| 53.1 OTT Session | 614 |
| 53.1.1 OTTSession | 614 |
| 53.2 ottSessionService | 615 |
| 53.2.1 ottSessionService.getMPDevicesWithOpenSessions | 615 |
| 53.2.2 ottSessionService.getSessionBySessionId | 616 |
| 53.2.3 ottSessionService.getSessions | 616 |
| 53.2.4 ottSessionService.setUp | Updated 617 |
| 53.2.5 ottSessionService.tearDown | 618 |
| 53.2.6 ottSessionService.tearDownNoDeviceChecks | 618 |
| 54 ottUpgradeService | New 619 |
| 54.1 ottUpgradeService | New 619 |
| 54.1.1 ottUpgradeService.upgrade | 619 |
| 55 packageService | 620 |
| 55.1 packageService | 620 |
| 55.1.1 packageService.addChannel | 620 |
| 55.1.2 packageService.addMultiLanguage | 620 |
| 55.1.3 packageService.create | 621 |
| 55.1.4 packageService.create | 621 |
| 55.1.5 packageService.delete | 622 |
| 55.1.6 packageService.getAllPackages | 622 |
| 55.1.7 packageService.getByOrigin | 622 |
| 55.1.8 packageService.getByOriginUID | 623 |
| 55.1.9 packageService.getByPackageOriginKeys | 624 |
| 55.1.10 packageService.getByPackageUIDs | 624 |
| 55.1.11 packageService.getByUID | 624 |
| 55.1.12 packageService.getChannels | 625 |
| 55.1.13 packageService.getMultiLanguageList | 626 |
| 55.1.14 packageService.getPagedPackagesByChannelUID | 626 |

| | |
|---|------------|
| 55.1.15 packageService.getPagedPkgMapsByPkgID | 627 |
| 55.1.16 packageService.getPagedPkgMapsByPkgPatternAndCatUID | 627 |
| 55.1.17 packageService.getPagedPkgsByChanUID | 628 |
| 55.1.18 packageService.getPagedPkgsByOriginUID | 629 |
| 55.1.19 packageService.getPagedPkgsBySbscrProfUID | 629 |
| 55.1.20 packageService.getPkgBySbscrProfUID | 630 |
| 55.1.21 packageService.getPkgMapsByPkgID | 630 |
| 55.1.22 packageService.getPkgMapsByPkgPattern | 631 |
| 55.1.23 packageService.getPkgMapsByPkgPatternAndCatUID | 632 |
| 55.1.24 packageService.getPkgsByChanUID | 632 |
| 55.1.25 packageService.getQspDefaultPackage | 633 |
| 55.1.26 packageService.modifyMultiLanguage | 633 |
| 55.1.27 packageService.removeChannel | 634 |
| 55.1.28 packageService.removeMultiLanguage | 634 |
| 55.1.29 packageService.update | 635 |
| 55.1.30 packageService.update | 635 |
| 55.1.31 packageService.updateModelAndML | 636 |
| 56 PAVService | 637 |
| 56.1 tv.quative.service.boc.asset.model.BocAsset | 637 |
| 56.2 tv.quative.service.boc.asset.model.BocAssetPlayType | 638 |
| 56.3 tv.quative.service.boc.asset.model.PlayList | 638 |
| 56.4 PAVService | 638 |
| 56.4.1 PAVService.getPlayList | 638 |
| 56.4.2 PAVService.getPlayListByMacAddress | 639 |
| 56.4.3 PAVService.getPlayListBySmartCard | 639 |
| 57 PaymentOptionService | 641 |
| 57.1 tv.quative.service.payment.model.PaymentProviderDetails | 641 |
| 57.2 tv.quative.service.payment.model.PaymentServiceProviderSession | 643 |
| 57.3 PaymentOptionService | 644 |

| | |
|---|------------|
| 57.3.1 PaymentOptionService.getPayMethodProvidersKeyData | 644 |
| 57.3.2 PaymentOptionService.getPaymentOption | 644 |
| 57.3.3 PaymentOptionService.getPaymentOptionByPaymentCredentialsUid | 645 |
| 57.3.4 PaymentOptionService.getPaymentOptionsForAccount | 645 |
| 57.3.5 PaymentOptionService.updatePaymentProviderSession | 646 |
| 58 PaymentTransactionService | 647 |
| 58.1 PaymentTransaction | 647 |
| 58.1.1 PaymentTransaction | 647 |
| 58.2 PaymentTransactionService | 648 |
| 58.2.1 PaymentTransactionService.create | 648 |
| 58.2.2 PaymentTransactionService.createPaymentTransaction | 648 |
| 58.2.3 PaymentTransactionService.delete | 649 |
| 58.2.4 PaymentTransactionService.deletePaymentTransaction | 649 |
| 58.2.5 PaymentTransactionService.getByOrigin | 649 |
| 58.2.6 PaymentTransactionService.getByUID | 650 |
| 58.2.7 PaymentTransactionService.getPaymentTransaction | 650 |
| 58.2.8 PaymentTransactionService.update | 651 |
| 59 Ping Service | 652 |
| 59.1 pingService | 652 |
| 59.1.1 pingService.ping | 652 |
| 60 policyGroupService | 653 |
| 60.1 policyGroupService | 653 |
| 60.1.1 policyGroupService.addAccessPoint | 653 |
| 60.1.2 policyGroupService.addMultiLanguage | 654 |
| 60.1.3 policyGroupService.addPolicy | 654 |
| 60.1.4 policyGroupService.create | 655 |
| 60.1.5 policyGroupService.delete | 655 |
| 60.1.6 policyGroupService.getByBasePriceUID | 655 |
| 60.1.7 policyGroupService.getByEvent | 656 |

| | | |
|-----------|---|--------------------|
| 60.1.8 | policyGroupService.getByName | 657 |
| 60.1.9 | policyGroupService.getByOrigin | 657 |
| 60.1.10 | policyGroupService.getByPolicyUID | 658 |
| 60.1.11 | policyGroupService.getByType | 659 |
| 60.1.12 | policyGroupService.getByTypeLocaleAndValidity | Updated 659 |
| 60.1.13 | policyGroupService.getByUID | 660 |
| 60.1.14 | policyGroupService.getDefaultQspZeroPrice | 661 |
| 60.1.15 | policyGroupService.getList | 661 |
| 60.1.16 | policyGroupService.getMultiLanguageList | 662 |
| 60.1.17 | policyGroupService.getPagedPolicyForCatAndAssetGroupAndAsset | 662 |
| 60.1.18 | policyGroupService.getPolicyGroupsByAccpt | 663 |
| 60.1.19 | policyGroupService.getPolicyGroupsBySchedule | 664 |
| 60.1.20 | policyGroupService.getPolicyGroupsByScheduleAndAccessPoint | 665 |
| 60.1.21 | policyGroupService.getPolicyGroupsByScheduleAndAccessPointAndLocale | 665 |
| 60.1.22 | policyGroupService.getPolicyGroupsByScheduleAndLocale | 666 |
| 60.1.23 | policyGroupService.getScheduledValidPolicyGroups | 667 |
| 60.1.24 | policyGroupService.getTechsByBusKey | 667 |
| 60.1.25 | policyGroupService.modifyMultiLanguage | 668 |
| 60.1.26 | policyGroupService.removeAccessPoint | 668 |
| 60.1.27 | policyGroupService.removeMultiLanguage | 669 |
| 60.1.28 | policyGroupService.removePolicy | 669 |
| 60.1.29 | policyGroupService.update | 670 |
| 60.1.30 | policyGroupService.updateModelAndML | 670 |
| 61 | policyGrpSubscriptionAggreService | 671 |
| 61.1 | policyGrpSubscriptionAggreService | 671 |
| 61.1.1 | policyGrpSubscriptionAggreService.subscribeToPolicyGroup | 671 |
| 61.1.2 | policyGrpSubscriptionAggreService.unsubscribePolicyGroup | 671 |
| 62 | policyService | 673 |
| 62.1 | Policy | 673 |

| | |
|--|-----|
| 62.1.1 Policy | 673 |
| 62.2 Base Price | 675 |
| 62.2.1 BasePrice | 676 |
| 62.3 policyService.addMultiLanguage | 679 |
| 62.3.1 policyService.addMultiLanguage | 680 |
| 62.3.2 policyService.addMultiLanguage | 680 |
| 62.3.3 policyService.create | 681 |
| 62.3.4 policyService.create | 681 |
| 62.3.5 policyService.createBasePrice | 681 |
| 62.3.6 policyService.delete | 682 |
| 62.3.7 policyService.delete | 682 |
| 62.3.8 policyService.deleteBasePrice | 683 |
| 62.3.9 policyService.getByByName | 683 |
| 62.3.10 policyService.getByOrigin | 684 |
| 62.3.11 policyService.getByPolicyGroupUID | 684 |
| 62.3.12 policyService.getByType | 685 |
| 62.3.13 policyService.getByUID | 686 |
| 62.3.14 policyService.getByUIDAndType | 686 |
| 62.3.15 policyService.getMultiLanguageList | 687 |
| 62.3.16 policyService.getMultiLanguageListByType | 688 |
| 62.3.17 policyService.isPolicyApplied | 688 |
| 62.3.18 policyService.modifyMultiLanguage | 689 |
| 62.3.19 policyService.modifyMultiLanguage | 689 |
| 62.3.20 policyService.removeMultiLanguage | 689 |
| 62.3.21 policyService.removeMultiLanguage | 690 |
| 62.3.22 policyService.update | 690 |
| 62.3.23 policyService.update | 691 |
| 62.3.24 policyService.updateBasePrice | 691 |
| 62.3.25 policyService.updateModelAndML | 691 |

| | |
|--|------------|
| 62.3.26 policyService.updateModelAndML | 692 |
| 63 PPVService | 693 |
| 63.1 PPVService | 693 |
| 63.1.1 PPVService.getChannelListWithPPVEvents | 693 |
| 63.1.2 PPVService.getPPVEventList | 693 |
| 63.1.3 PPVService.getPPVEventListByChannel | 694 |
| 63.1.4 PPVService.getPagedChannelListWithPPVEvents | 695 |
| 63.1.5 PPVService.getPagedPPVEventList | 696 |
| 63.1.6 PPVService.getPagedPPVEventListByChannel | 697 |
| 63.1.7 PPVService.getPagedPurchasedPPVEventList | 698 |
| 63.1.8 PPVService.getPurchasedPPVEventList | 699 |
| 63.1.9 PPVService.purchasePPVEventByPolicyGroup | 700 |
| 63.1.10 PPVService.purchasePPVEventByProduct | 701 |
| 64 preferenceService | 703 |
| 64.1 Preference | 703 |
| 64.1.1 Preference | 703 |
| 64.2 preferenceService | 704 |
| 64.2.1 preferenceService.create | 704 |
| 64.2.2 preferenceService.delete | 705 |
| 64.2.3 preferenceService.getByOrigin | 705 |
| 64.2.4 preferenceService.getByOriginUID | 706 |
| 64.2.5 preferenceService.getByUID | 706 |
| 64.2.6 preferenceService.getList | 707 |
| 64.2.7 preferenceService.update | 708 |
| 65 presenceService | 709 |
| 65.1 tv.quative.service.user.model.PurchaseAbility | 709 |
| 65.2 User | 709 |
| 65.2.1 User | 710 |
| 65.3 presenceService | 713 |

| | |
|--|----------------|
| 65.3.1 presenceService.addUserToAccount | 713 |
| 65.3.2 presenceService.addUserToAccount | 714 |
| 65.3.3 presenceService.modifyUserOfAccount | 714 |
| 65.3.4 presenceService.removeUserByNameAndAccountNumber | 715 |
| 65.3.5 presenceService.removeUserFromAccount | 715 |
| 66 profileService | 717 |
| 66.1 profileService | 717 |
| 66.1.1 profileService.addCatToSbscrProf | 717 |
| 66.1.2 profileService.addFeatureToDProf | 717 |
| 66.1.3 profileService.addFeatureToProf | 718 |
| 66.1.4 profileService.addPkgToSbscrProf | 718 |
| 66.1.5 profileService.addProfile | 719 |
| 66.1.6 profileService.create | 720 |
| 66.1.7 profileService.create | 720 |
| 66.1.8 profileService.delete | 720 |
| 66.1.9 profileService.getALaCarteProfiles | 721 |
| 66.1.10 profileService.getByFeatureUIDAndProfileType | 721 |
| 66.1.11 profileService.getByName | 722 |
| 66.1.12 profileService.getByOrigin | 722 |
| 66.1.13 profileService.getByType | 723 |
| 66.1.14 profileService.getByUID | 723 |
| 66.1.15 profileService.getEmptyProfiles | 724 |
| 66.1.16 profileService.getEmptyProfileUids | 725 |
| 66.1.17 profileService.getFeatureValueForAccount. | New 725 |
| 66.1.18 profileService.getPagedAlacarteProfilesByChannelNetCarry | 727 |
| 66.1.19 profileService.getProfiles | 728 |
| 66.1.20 profileService.removeCatFromSbscrProf | 728 |
| 66.1.21 profileService.removeDeviceTypeFromDvcProf. | New 729 |
| 66.1.22 profileService.removeFeatureFromDProf | 729 |

| | |
|---|------------|
| 66.1.23 profileService.removeFeatureFromSrcProf | 730 |
| 66.1.24 profileService.removePkgFromSbscrProf | 730 |
| 66.1.25 profileService.removeProfile | 731 |
| 66.1.26 profileService.update | 732 |
| 66.1.27 profileService.update | 732 |
| 66.1.28 profileService.updateFeatureProfileValue | 732 |
| 67 programService | 734 |
| 67.1 Verbose Program | 734 |
| 67.1.1 VerboseProgram | 734 |
| 67.2 tv.quative.service.program.model.ProgramAltAudioPID | 738 |
| 67.3 Program | 739 |
| 67.3.1 Program | 739 |
| 67.4 tv.quative.service.contentrating.model.primitive.AggregateRating | 741 |
| 67.5 programService.addMultiLanguage | 741 |
| 67.5.1 programService.addMultiLanguage | 742 |
| 67.5.2 programService.addProgramAltAudioPID | 743 |
| 67.5.3 programService.create | 743 |
| 67.5.4 programService.delete | 743 |
| 67.5.5 programService.getAudioPIDsByPgmUID | 744 |
| 67.5.6 programService.getByOrigin | 744 |
| 67.5.7 programService.getByUID | 745 |
| 67.5.8 programService.getByUIDs | 746 |
| 67.5.9 programService.getMultiLanguageList | 746 |
| 67.5.10 programService.getPagedPgmAltAudioPIDsByPgmUID | 747 |
| 67.5.11 programService.getPagedVerboseProgramsByOriginUID | 747 |
| 67.5.12 programService.getSimpleProgramByUID | 748 |
| 67.5.13 programService.getVerboseProgramByEventUID | 748 |
| 67.5.14 programService.getVerboseProgramByOriginUID | 749 |
| 67.5.15 programService.getVerboseProgramByUID | 749 |

| | |
|---|------------|
| 67.5.16 programService.modifyMultiLanguage | 750 |
| 67.5.17 programService.modifyProgramAltAudioPID | 750 |
| 67.5.18 programService.removeMultiLanguage | 751 |
| 67.5.19 programService.removeProgramAltAudioPID | 751 |
| 67.5.20 programService.update | 752 |
| 67.5.21 programService.updateModelAndML | 752 |
| 68 provisionService | 754 |
| 68.1 provisionService | 754 |
| 68.1.1 provisionService.pcClientRegistration | 754 |
| 68.1.2 provisionService.provisionMediaPlayer | 754 |
| 69 QUMActivity Service | 756 |
| 69.1 QUM Activity | 756 |
| 69.1.1 QUMActivity | 756 |
| 69.2 tv.quative.service.qum.activity.model.ActivityType | 757 |
| 69.3 qumActivityService | 757 |
| 69.3.1 qumActivityService.create | 757 |
| 69.3.2 qumActivityService.create | 758 |
| 69.3.3 qumActivityService.delete | 758 |
| 69.3.4 qumActivityService.delete | 759 |
| 69.3.5 qumActivityService.getActivityListByRoleName | 759 |
| 69.3.6 qumActivityService.getActivityListByRoleUID | 759 |
| 69.3.7 qumActivityService.getActivityTreeNodeesByRoleName | 760 |
| 69.3.8 qumActivityService.getByActivityName | 760 |
| 69.3.9 qumActivityService.getByOrigin | 761 |
| 69.3.10 qumActivityService.getByUID | 762 |
| 69.3.11 qumActivityService.update | 762 |
| 69.3.12 qumActivityService.update | 762 |
| 70 QUM User Service | 764 |
| 70.1 QUM User | 764 |

| | |
|--|------------|
| 70.1.1 QUMUser | 764 |
| 70.2 qumUserService | 765 |
| 70.2.1 qumUserService.addRole | 765 |
| 70.2.2 qumUserService.create | 766 |
| 70.2.3 qumUserService.create | 766 |
| 70.2.4 qumUserService.delete | 766 |
| 70.2.5 qumUserService.getByOrigin | 767 |
| 70.2.6 qumUserService.getByUID | 767 |
| 70.2.7 qumUserService.getQUMUsersForRole | 768 |
| 70.2.8 qumUserService.getRoles | 768 |
| 70.2.9 qumUserService.removeRole | 769 |
| 70.2.10 qumUserService.update | 769 |
| 70.2.11 qumUserService.update | 770 |
| 71 ratingService | 771 |
| 71.1 Rating | 771 |
| 71.1.1 Rating | 771 |
| 71.2 ratingService | 772 |
| 71.2.1 ratingService.addMultiLanguage | 773 |
| 71.2.2 ratingService.create | 773 |
| 71.2.3 ratingService.delete | 773 |
| 71.2.4 ratingService.getByOrigin | 774 |
| 71.2.5 ratingService.getByUID | 775 |
| 71.2.6 ratingService.getDefaultRating | 775 |
| 71.2.7 ratingService.getList | 776 |
| 71.2.8 ratingService.getMultiLanguageList | 776 |
| 71.2.9 ratingService.getRatingByRatingCode | 777 |
| 71.2.10 ratingService.modifyMultiLanguage | 777 |
| 71.2.11 ratingService.removeMultiLanguage | 778 |
| 71.2.12 ratingService.update | 778 |

| | |
|--|------------|
| 71.2.13 ratingService.updateModelAndML | 779 |
| 72 recommendationService | 780 |
| 72.1 tv.quative.service.recommendation.api.ExcludeContentRequest | 780 |
| 72.2 tv.quative.service.recommendation.api.RecommendationFilter | 781 |
| 72.3 tv.quative.service.recommendation.api.RecommendationRequest | 781 |
| 72.4 tv.quative.service.recommendation.api.ContentRatingRequest | 782 |
| 72.5 tv.quative.service.recengintegration.enums.RecommendationEngineUserAction | 783 |
| 72.6 tv.quative.service.recommendation.api.ReportActionRequest | 784 |
| 72.7 recommendationService | 784 |
| 72.7.1 recommendationService.clearCache | 785 |
| 72.7.2 recommendationService.excludeRecommendation | 785 |
| 72.7.3 recommendationService.getRawRecommendations | 785 |
| 72.7.4 recommendationService.getRecommendations | 786 |
| 72.7.5 recommendationService.rate | 786 |
| 72.7.6 recommendationService.reportAction | 787 |
| 73 Report Engine Service | 788 |
| 73.1 reportEngineService | 788 |
| 73.1.1 reportEngineService.cancelRunningReport | 788 |
| 73.1.2 reportEngineService.processAdHocReport | 789 |
| 74 Report Scheduler Service | 790 |
| 74.1 reportSchedulerService | 790 |
| 74.1.1 reportSchedulerService.scheduleAllActiveReports | 790 |
| 74.1.2 reportSchedulerService.scheduleAllActiveSchedulesOfReport | 791 |
| 74.1.3 reportSchedulerService.scheduleReportSchedule | 791 |
| 74.1.4 reportSchedulerService.unscheduleReport | 792 |
| 74.1.5 reportSchedulerService.unscheduleReportSchedule | 793 |
| 75 Report Service | 794 |
| 75.1 Report | 794 |
| 75.1.1 Report | 794 |

| | |
|--|-----|
| 75.2 tv.quative.service.report.model.Protocol | 795 |
| 75.3 tv.quative.service.report.model.ReportDestination | 795 |
| 75.4 tv.quative.service.report.model.ReportDestination.CompressionScheme | 797 |
| 75.5 tv.quative.service.report.model.ReportRunLog | 797 |
| 75.6 tv.quative.service.report.model.Status | 798 |
| 75.7 tv.quative.service.report.model.ReportSchedule | 799 |
| 75.8 reportService | 799 |
| 75.8.1 reportService.addReportDestination | 800 |
| 75.8.2 reportService.addReportRunLog | 800 |
| 75.8.3 reportService.addReportSchedule | 801 |
| 75.8.4 reportService.create | 801 |
| 75.8.5 reportService.delete | 801 |
| 75.8.6 reportService.getByOrigin | 802 |
| 75.8.7 reportService.getByOriginID | 802 |
| 75.8.8 reportService.getByUID | 803 |
| 75.8.9 reportService.getLastReportRunLogByStatus | 803 |
| 75.8.10 reportService.getLatestReportRunLog | 804 |
| 75.8.11 reportService.getList | 805 |
| 75.8.12 reportService.getReportDestination | 805 |
| 75.8.13 reportService.getReportDestinationList | 805 |
| 75.8.14 reportService.getReportRunLog | 806 |
| 75.8.15 reportService.getReportRunLogList | 807 |
| 75.8.16 reportService.getReportSchedule | 807 |
| 75.8.17 reportService.getReportScheduleList | 808 |
| 75.8.18 reportService.modifyReportDestination | 808 |
| 75.8.19 reportService.modifyReportRunLog | 809 |
| 75.8.20 reportService.modifyReportSchedule | 809 |
| 75.8.21 reportService.removeReportDestination | 809 |
| 75.8.22 reportService.removeReportRunLog | 810 |

| | |
|---|------------|
| 75.8.23 reportService.removeReportSchedule | 810 |
| 75.8.24 reportService.update | 811 |
| 76 Role Service | 812 |
| 76.1 Role | 812 |
| 76.1.1 Role | 812 |
| 76.2 roleService | 813 |
| 76.2.1 roleService.addActivity | 814 |
| 76.2.2 roleService.addMultiLanguage | 814 |
| 76.2.3 roleService.addMultiLanguage | 815 |
| 76.2.4 roleService.create | 815 |
| 76.2.5 roleService.create | 815 |
| 76.2.6 roleService.delete | 816 |
| 76.2.7 roleService.delete | 816 |
| 76.2.8 roleService.getAllRoles | 816 |
| 76.2.9 roleService.getByOrigin | 817 |
| 76.2.10 roleService.getByUID | 818 |
| 76.2.11 roleService.getMultiLanguageList | 818 |
| 76.2.12 roleService.getRoleListByAccountUID | 819 |
| 76.2.13 roleService.modifyActivity | 819 |
| 76.2.14 roleService.modifyMultiLanguage | 820 |
| 76.2.15 roleService.modifyMultiLanguage | 820 |
| 76.2.16 roleService.removeActivity | 821 |
| 76.2.17 roleService.removeMultiLanguage | 821 |
| 76.2.18 roleService.removeMultiLanguage | 822 |
| 76.2.19 roleService.update | 822 |
| 76.2.20 roleService.update | 823 |
| 76.2.21 roleService.updateModelAndML | 823 |
| 77 ServiceProvider Service | 824 |
| 77.1 Service Provider | 824 |

| | |
|--|--------------------|
| 77.1.1 ServiceProvider | 824 |
| 77.2 serviceproviderService | 825 |
| 77.2.1 serviceproviderService.create | 825 |
| 77.2.2 serviceproviderService.delete | 826 |
| 77.2.3 serviceproviderService.getByOrigin | 826 |
| 77.2.4 serviceproviderService.getByUID | 827 |
| 77.2.5 serviceproviderService.getByGlobalSPID | 827 |
| 77.2.6 serviceproviderService.update | 827 |
| 78 ServiceProviderPaymentOptionService | 829 |
| 78.1 ServiceProviderPaymentOption | 829 |
| 78.1.1 ServiceProviderPaymentOption | 829 |
| 78.2 ServiceProviderPaymentOptionService | 830 |
| 78.2.1 ServiceProviderPaymentOptionService.create | 831 |
| 78.2.2 ServiceProviderPaymentOptionService.createServiceProviderPaymentOption | 831 |
| 78.2.3 ServiceProviderPaymentOptionService.delete | 831 |
| 78.2.4 ServiceProviderPaymentOptionService.deleteServiceProviderOption | 832 |
| 78.2.5 ServiceProviderPaymentOptionService.getByOrigin | 832 |
| 78.2.6 ServiceProviderPaymentOptionService.getByUID | 833 |
| 78.2.7 ServiceProviderPaymentOptionService.getPaymentOptionsForServiceProvider | 833 |
| 78.2.8 ServiceProviderPaymentOptionService.getServiceProviderPaymentOption | 834 |
| 78.2.9 ServiceProviderPaymentOptionService.update | 834 |
| 78.2.10 ServiceProviderPaymentOptionService.updateServiceProviderPaymentOption | 835 |
| 79 signonService | 836 |
| 79.1 signonService | 836 |
| 79.1.1 signonService.doSomeMongoLogging | 836 |
| 79.1.2 signonService.signonByCASN | Updated 837 |
| 79.1.3 signonService.signonByMACAddress | 837 |
| 79.1.4 signonService.signonByMpDeviceIdAndUser | 838 |
| 79.1.5 signonService.signonByPublicDeviceID | New 838 |

| | |
|--|----------------|
| 79.1.6 signonService.signonBySmartcardID | 838 |
| 79.1.7 signonService.signonByUser | 839 |
| 79.1.8 signonService.signonByUserMpldAndSpid | 839 |
| 79.1.9 signonService.signonInitialByUserAndSpid | 840 |
| 79.1.10 signonService.updateTokenWithMediaPlayer | 841 |
| 79.1.11 signonService.registerDeviceAndGetToken | New 841 |
| 79.1.12 signonService.getCastToken | New 843 |
| 79.1.13 signonService.signonByDASDeviceandUser | New 845 |
| 80 smartCardService | 848 |
| 80.1 Smart Card | 848 |
| 80.1.1 SmartCard | 848 |
| 80.2 smartCardService | 851 |
| 80.2.1 smartCardService.create | 851 |
| 80.2.2 smartCardService.delete | 851 |
| 80.2.3 smartCardService.getAvailableSmartcardList | 852 |
| 80.2.4 smartCardService.getAvailableSmartcardListCount | 853 |
| 80.2.5 smartCardService.getByOrigin | 853 |
| 80.2.6 smartCardService.getBySmartCardId | 853 |
| 80.2.7 smartCardService.getByUID | 854 |
| 80.2.8 smartCardService.update | 854 |
| 81 spChannelMapService | 856 |
| 81.1 sPChannelMapService | 856 |
| 81.1.1 sPChannelMapService.create | 856 |
| 81.1.2 sPChannelMapService.delete | 856 |
| 81.1.3 sPChannelMapService.getByOrigin | 857 |
| 81.1.4 sPChannelMapService.getByUID | 857 |
| 81.1.5 sPChannelMapService.update | 858 |
| 82 statisticsReportingService | 859 |
| 82.1 tv.quative.service.statisticsreporting.model.AggregationLevel | 859 |

| | |
|--|------------|
| 82.2 tv.quative.service.statisticsreporting.model.DomainAggregationLevel | 859 |
| 82.3 statisticsReportingService | 859 |
| 82.3.1 statisticsReportingService.getActivity | 860 |
| 82.3.2 statisticsReportingService.getChannelPopularity | 860 |
| 82.3.3 statisticsReportingService.getLiveActivity | 861 |
| 82.3.4 statisticsReportingService.getLiveChannelPopularity | 861 |
| 82.3.5 statisticsReportingService.getLiveServiceGroupPopularity | 862 |
| 82.3.6 statisticsReportingService.getLiveServiceGroupUtilisation | 862 |
| 82.3.7 statisticsReportingService.getServiceGroupPopularity | 862 |
| 82.3.8 statisticsReportingService.getServiceGroupUtilisation | 863 |
| 83 subscriptionProfileService | 864 |
| 83.1 tv.quative.service.subscriptionprofile.impl.ProfileSmartcardSubscribableModel | 864 |
| 83.2 subscriptionProfileService | 864 |
| 83.2.1 subscriptionProfileService.getProfilesByAccountUID | 865 |
| 83.2.2 subscriptionProfileService.subscribeCatalogue | 865 |
| 83.2.3 subscriptionProfileService.subscribePackage | 866 |
| 83.2.4 subscriptionProfileService.subscribeProfile | 866 |
| 83.2.5 subscriptionProfileService.unsubscribeCatalogue | 867 |
| 83.2.6 subscriptionProfileService.unsubscribePackage | 867 |
| 83.2.7 subscriptionProfileService.unsubscribeProfile | 868 |
| 84 subscriptionService | 869 |
| 84.1 tv.quative.service.subscription.model.PriceInfo | 869 |
| 84.2 tv.quative.service.subscription.model.SubscribableModel | 870 |
| 84.3 tv.quative.service.subscription.model.LocaleSubscribableModel | 871 |
| 84.4 subscriptionService | 872 |
| 84.4.1 subscriptionService.getCalculatedPrice | 873 |
| 84.4.2 subscriptionService.getCalculatedPriceWithOffer | 873 |
| 84.4.3 subscriptionService.subscribe | 874 |
| 84.4.4 subscriptionService.subscribeLocaleBased | 874 |

| | |
|--|------------|
| 84.4.5 subscriptionService.unsubscribe | 875 |
| 84.4.6 subscriptionService.unsubscribeByAclUID | 875 |
| 84.4.7 subscriptionService.unsubscribeLocaleBased | 876 |
| 85 targetAreaService | 877 |
| 85.1 Target Area | 877 |
| 85.1.1 TargetArea | 877 |
| 85.2 targetAreaService | 878 |
| 85.2.1 targetAreaService.addMultiLanguage | 878 |
| 85.2.2 targetAreaService.associateAsset | 879 |
| 85.2.3 targetAreaService.associateAssetList | 879 |
| 85.2.4 targetAreaService.associateAssetList | 880 |
| 85.2.5 targetAreaService.associatePostCode | 880 |
| 85.2.6 targetAreaService.associatePostCodeList | 881 |
| 85.2.7 targetAreaService.associatePostCodeList | 881 |
| 85.2.8 targetAreaService.create | 882 |
| 85.2.9 targetAreaService.delete | 882 |
| 85.2.10 targetAreaService.disassociateAsset | 882 |
| 85.2.11 targetAreaService.disassociateAssetList | 883 |
| 85.2.12 targetAreaService.disassociatePostCode | 883 |
| 85.2.13 targetAreaService.disassociatePostCodeList | 884 |
| 85.2.14 targetAreaService.disassociatePostCodeList | 884 |
| 85.2.15 targetAreaService.getAssetUids | 885 |
| 85.2.16 targetAreaService.getByAssetUid | 885 |
| 85.2.17 targetAreaService.getByOrigin | 886 |
| 85.2.18 targetAreaService.getByPostCode | 887 |
| 85.2.19 targetAreaService.getByUID | 887 |
| 85.2.20 targetAreaService.getMultiLanguageList | 888 |
| 85.2.21 targetAreaService.getPostCodes | 888 |
| 85.2.22 targetAreaService.modifyMultiLanguage | 889 |

| | |
|---|------------|
| 85.2.23 targetAreaService.removeMultiLanguage | 889 |
| 85.2.24 targetAreaService.update | 890 |
| 85.2.25 targetAreaService.updateModelAndML | 890 |
| 86 tunerService | 892 |
| 86.1 tv.quative.service.tuner.model.Tuner | 892 |
| 86.2 tunerService | 893 |
| 86.2.1 tunerService.countByRtspSessionUID | 893 |
| 86.2.2 tunerService.create | 893 |
| 86.2.3 tunerService.create | 894 |
| 86.2.4 tunerService.delete | 894 |
| 86.2.5 tunerService.delete | 895 |
| 86.2.6 tunerService.getActiveBySetTopBoxUID | 895 |
| 86.2.7 tunerService.getActiveOlderThan | 895 |
| 86.2.8 tunerService.getByOrigin | 896 |
| 86.2.9 tunerService.getByRtspSessionUID | 896 |
| 86.2.10 tunerService.getBySetTopBoxSessionID | 897 |
| 86.2.11 tunerService.getBySetTopBoxUID | 898 |
| 86.2.12 tunerService.getByUID | 898 |
| 86.2.13 tunerService.update | 899 |
| 86.2.14 tunerService.update | 899 |
| 87 userManagementService | 900 |
| 87.1 userManagementService | 900 |
| 87.1.1 userManagementService.changePinByAccNumAndUserName | 900 |
| 87.1.2 userManagementService.changePinByUserUid | 901 |
| 87.1.3 userManagementService.createUser | 901 |
| 87.1.4 userManagementService.removeUser | 902 |
| 87.1.5 userManagementService.updateDateOfBirth | 902 |
| 87.1.6 userManagementService.updateLogin | 902 |
| 87.1.7 userManagementService.updatePassword | 903 |

| | |
|--|---------------------------|
| 87.1.8 userManagementService.updatePurchaseAbility | 903 |
| 87.1.9 userManagementService.updateRatingId | 904 |
| 87.1.10 userManagementService.updateType | 904 |
| 88 userService | 906 |
| 88.1 userService | 906 |
| 88.1.1 userService.create | 906 |
| 88.1.2 userService.delete | 906 |
| 88.1.3 userService.getByAcctUID | 907 |
| 88.1.4 userService.getByOrigin | 907 |
| 88.1.5 userService.getByOriginUID | 908 |
| 88.1.6 userService.getByUID | 908 |
| 88.1.7 userService.getDefaultUserByAcctUID | 909 |
| 88.1.8 userService.getUserPIN | <i>New</i> 909 |
| 88.1.9 userService.update | 910 |
| 89 visitorService. | <i>Updated</i> 911 |
| 89.1 visitorService | 915 |
| 89.1.1 visitorService.create | 915 |
| 89.1.2 visitorService.update | 916 |
| 89.1.3 visitorService.getGroup | 916 |
| 89.1.4 visitorService.getValueAsInt | 918 |
| 90 workflowSubscriptionService | 919 |
| 90.1 workflowSubscriptionService | 919 |
| 90.1.1 workflowSubscriptionService.createAccountUserSubscription | 919 |
| 90.1.2 workflowSubscriptionService.createNewSubscription | 919 |
| 90.1.3 workflowSubscriptionService.createSmartCardSubscription | 920 |
| 90.1.4 workflowSubscriptionService.createSubscription | 921 |
| 90.1.5 workflowSubscriptionService.updateExistingSubscription | 921 |
| 90.1.6 workflowSubscriptionService.updateExsistingSubscription | 922 |
| 90.1.7 workflowSubscriptionService.updateSmartCardSubscription | 923 |

| | |
|---|-----|
| 90.1.8 workflowSubscriptionService.updateSubscription | 924 |
| 90.1.9 workflowSubscriptionService.updateSubscription | 925 |

1 Introduction

This document lists all the operations made available to Web Services or HUE clients by a standard installation of SDP. It describes each of the **Services** in turn, providing full details of the methods it exposes to manipulate QSP **Entities**. It is intended for use as a complete reference and also for application developers acting on recommendations from solution architects.

Caution!

This document contains full details of all exposed APIs provided by all **Services**, including APIs intended for use only in specific circumstances, and APIs which are only meaningful in specific deployment environments.

For recommendations on how to use the SDP API set for specific use cases, please consult the Integration Guides supplied with SDP, or request guidance from a solution architect.

Use of any API in this document which is not covered in an Integration Guide is not supported by NAGRA without explicit agreement.

1.1 References

| Reference | Title | |
|---------------|--|---|
| SDP-SVC-BTV-A | Broadcast Services API Reference | This document describes the full API to the NAGRA Media SDP Broadcast Services. |
| SDP-SVC-C-A | Core Services API Reference | This document describes the full API to the NAGRA Media SDP Core Services. |
| SDP-SVC-S-A | Specialized Services API Reference | This document describes the full API to the public Services provided by NAGRA Media SDP in addition to the Core, On-Demand, Broadcast and SRM Services. |
| SDP-SVC-SRM-A | Session Resource Management Services API Reference | This document describes the full API to the NAGRA Media SDP Session Resource Management Services. |

2 accessPointService

This is a primitive *Service* which provides the *Access Point* Entity.

It manages relationships between *Access Points* and *Device Groups*, *Service Providers*, *Catalogues* and *Policy Groups*.

2.1 Access Point

An *Access Point* represents a logical grouping of *Accounts* which access the operator's service in the same manner.

Figure 1 AccessPoint and associated Entities.

Access Points can be used to group *Accounts* by access method (for example, cable and IP clients), by geographical region, or by some other desirable sub-group of the subscriber base.

An *Access Point* can be associated with *Device Groups* (or *Zones*, where relevant) to represent the head-end devices which service that set of *Accounts*. It may also be associated with *Policy Groups* and *Catalogues* to provide content and special offers specific to a region or sub-group.

The CMS ingest uses the **Regions** metadata key of the CMS *Catalogue* and *Product* objects to assign *Catalogues* and *Policy Groups* to *Access Points*.

Caution!

You must create *Access Points* with names corresponding to all the **Regions** metadata keys in use before the CMS ingest will work.

Example

A service provider has a single *Device Group*, **A**, containing *Application Servers*, and two *Device Groups*, **S1** and **S2**, containing *Streaming Servers* in different regions.

An *Access Point* is created for *Accounts* in each region. One is associated with **A** and **S1**; the other with **A** and **S2**.

When a *Set Top Box* in the first region wants to access content, it requests information from SDP to find out first the *Access Point* associated with the *Set Top Box*'s controlling *Account*, and then the *Device Group* containing *Streaming Servers* for that *Access Point* (that is, **S1**.) Once it has this information, it can set up a streaming session by sending RTSP requests to one of the servers in **S1**.

2.1.1 AccessPoint

| Name | Type | Description |
|--------------|--|---|
| changeLog | Deprecated. Not visible to HUE. List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |

| Name | Type | Description |
|-----------------------------|---|---|
| <code>description</code> | <code>String</code> | A short description of this <i>Entity</i> . |
| <code>enabled</code> | <code>Boolean</code> | Is this <i>Access Point</i> currently enabled? |
| <code>exportID</code> | <code>Integer</code> | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> <code>Quick Search criterion.</code> <code>String</code> | The name of this <i>Entity</i> . |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>publicAddrAPS</code> | <code>String</code> | The public IP address of the application server used by this <i>Access Point</i> . <i>For example:</i> <code>172.31.10.40</code> <i>For example:</i> <code>fffe:f030:4971:ecdd:39b2:19a8</code> Client applications may use this field to connect to a more local application server, improving performance. |

Note

This field is only used if your deployment does not use *Application Server* Entities to represent application servers.

If it does, you should place *Application Servers* in a *Device Group* and associate this *Device Group* with the *Access Point*.

| Name | Type | Description |
|--------------------------------|----------------------|---|
| <code>publicAddrSTS</code> | <code>String</code> | <p>The public IP address of the streaming server (video server) used by this <i>Access Point</i>.</p> <p>IPTV client applications may use this field to work out where to obtain video streams.</p> <p>Note This field is only used if your deployment does not use <i>Streaming Server</i> Entities to represent application servers. If it does, you should place <i>Streaming Servers</i> in a <i>Device Group</i> and associate this <i>Device Group</i> with the <i>Access Point</i>.</p> |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |
| <code>zoneID</code> | <code>Integer</code> | <p>The IRD zone which covers clients served by this <i>Access Point</i>.</p> <p>IRD is a Nagra CAS feature which allows operators to send messages to specific clients. If you are using Nagra CAS, you should assign IRD zones to regions in the CAS and fill in this field appropriately.</p> <p>Note All <i>Accounts</i> in a region (<i>Access Point</i>) must have the same Zone ID, but several <i>Access Points</i> may share the same Zone ID.</p> |

2.2 accessPointService

AccessPoint Service Implementation. Object which implements the business logic of all the methods of the interface. The Access Point is just a reference to a different collection of Device Groups, depending on the location of the customer (mainly represented by the country, locale and location of the customer). So it makes you easy to distinguish between different groups of streaming servers which are language dependent. The services are made visible to Hue or to Web Services by writing the annotations in the Service Implementation class. Ex: The services are made visible to Hue or to WebServices by writing the annotations in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="accesspointService", hue = true, ws = true) For the Methods: @QubeMethod(hue=false, ws = true) The methods used to create, delete or update model data only could be accesible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the Web Services and the Hue.

jraya @author

Naeem Bilal @author

2.2.1 accessPointService.addCatalogue

Adds an association of Catalogue with Access Point..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pCatalogueUid | Long | as catalogueUid |
| pAccessPointUid | Long | as accessPointUid |

Return value

Type:

Integer as affected rows

Example

```
<addCatalogue>
  <pCatalogueUid>72</pCatalogueUid>
  <pAccessPointUid>59</pAccessPointUid>
</addCatalogue>
```

2.2.2 accessPointService.addDeviceGroup

Contains the business logic to add an DeviceGroup to an AccessPoint.

The relationship between DeviceGroup and AccessPoint is many to many.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|-----------------------|
| pdevgrpuid | Long | Long devicegroup uid |
| paccptuid | Long | Long access point uid |

Return value

Type:

Integer num rows added

Example

```
<addDeviceGroup>
  <pdevgrpuid>27</pdevgrpuid>
  <pacctuid>84</pacctuid>
</addDeviceGroup>
```

2.2.3 accessPointService.addPolicyGroup

Adds an association of Policy Group with Access Point..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pAccessPointUid | Long | as accessPointUid |
| pPolicyGroup | Long | as policyGroup |

Return value

Type:

Integer as affected rows

Example

```
<addPolicyGroup>
  <pAccessPointUid>84</pAccessPointUid>
  <pPolicyGroup>2</pPolicyGroup>
</addPolicyGroup>
```

2.2.4 accessPointService.addServiceProvider

Contains the business logic to add an ServiceProvider (OperatorRegion) to an AccessPoint.

The relationship between ServiceProvider and AccessPoint is many to many.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---------------------------|
| psrvprouid | Long | Long service provider uid |
| pacptuid | Long | Long access point uid |

Return value

Type:

Integer num rows added

Example

```
<addServiceProvider>
  <psrvprouid>64</psrvprouid>
  <pacptuid>73</pacptuid>
</addServiceProvider>
```

2.2.5 accessPointService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

2.2.6 accessPointService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

2.2.7 accessPointService.getAccessPointByCat

Gets List of AccessPoint associated to Catalogue..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|------|-----------------|
| pCatalogueUid | Long | as catalogueUid |

Return value

Type: `List<AccessPoint>`

List of AccessPoint

Example

```
<getAccessPointByCat>
  <pCatalogueUid>19</pCatalogueUid>
</getAccessPointByCat>
```

2.2.8 accessPointService.getAccessPointsByPolicyGroup

Gets List of AccessPoint associated to Policy Group..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|------|-------------|
| pPolicyGroup | Long | |

Return value

Type: `List<AccessPoint>`

List of AccessPoint

Example

```
<getAccessPointsByPolicyGroup>
  <pPolicyGroup>64</pPolicyGroup>
</getAccessPointsByPolicyGroup>
```

2.2.9 accessPointService.getAccessPointsBySPUID

Retrieve, from storage, a list of available `AccessPoint` objects..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|------|-------------|
| spUId | Long | |

Return value

Type: `List<AccessPoint>`

List If found, a list of the matching objects, null otherwise.

Example

```
<getAccessPointsBySPUID>
  <spUId>1</spUId>
</getAccessPointsBySPUID>
```

2.2.10 accessPointService.getAllAccessPoints

Gets all the *Access Point* Entities in the database.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/accessPointService/getAllAccessPoints`

Available through Web Services.

Parameters

None.

Return value

Type: `List<AccessPoint>`

Example

```
<getAllAccessPoints/>
```

2.2.11 accessPointService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>15</originID>
</getByOrigin>
```

2.2.12 accessPointService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/accessPointService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>66</uid>
</getByUID>
```

2.2.13 accessPointService.getDeviceGroups

Gets all the device groups associated to an accesspoint..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/accessPointService/getDeviceGroups`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|------|------------------------------|
| pacptuid | Long | unique id of the AccessPoint |

Return value

Type: `List<DeviceGroup>`

List of DeviceGroup objects with the information of the device groups associated to the accesspoint.

Example

```
<getDeviceGroups>
  <pacptuid>61</pacptuid>
</getDeviceGroups>
```

2.2.14 accessPointService.getServiceProvidersByAPUID

Retrieve, from storage, a list of available **ServiceProvider** objects..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|------|-------------|
| apUid | Long | |

Return value

Type: **List<ServiceProvider>**

List If found, a list of the matching objects, null otherwise.

Example

```
<getServiceProvidersByAPUID>
  <apUid>49</apUid>
</getServiceProvidersByAPUID>
```

2.2.15 accessPointService.removeCatalogue

Remove an association of Catalogue with Access Point..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pCatalogueUid | Long | as catalogueUid |
| pAccessPointUid | Long | as accessPointUid |

Return value

Type:

Integer as affected rows

Example

```
<removeCatalogue>
  <pCatalogueUid>26</pCatalogueUid>
  <pAccessPointUid>40</pAccessPointUid>
</removeCatalogue>
```


2.2.16 accessPointService.removeDeviceGroup

Contains the business logic to remove an DeviceGroup to an AccessPoint.

The relationship between DeviceGroup and AccessPoint is many to many.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-----------------------|
| pdvgrpuid | Long | Long devicegroup uid |
| paccptuid | Long | Long access point uid |

Return value

Type:

Integer num rows deleted

Example

```
<removeDeviceGroup>
  <pdvgrpuid>13</pdvgrpuid>
  <paccptuid>93</paccptuid>
</removeDeviceGroup>
```

2.2.17 accessPointService.removePolicyGroup

Remove an association of Policy Group with Access Point..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pAccessPointUid | Long | as accessPointUid |
| pPolicyGroup | Long | as policyGroup |

Return value

Type:

Integer as affected rows

Example

```
<removePolicyGroup>
  <pAccessPointUid>44</pAccessPointUid>
  <pPolicyGroup>19</pPolicyGroup>
</removePolicyGroup>
```

2.2.18 accessPointService.removeServiceProvider

Contains the business logic to remove an ServiceProvider (OperatorRegion) to an AccessPoint.

The relationship between ServiceProvider and AccessPoint is many to many.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---------------------------|
| psrvprouid | Long | Long service provider uid |
| paccptuid | Long | Long access point uid |

Return value

Type:

Integer num rows deleted

Example

```
<removeServiceProvider>
  <psrvprouid>78</psrvprouid>
  <paccptuid>12</paccptuid>
</removeServiceProvider>
```

2.2.19 accessPointService.update *Updated*

Update, to storage, an existing BaseModel..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|----------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to testbuild |

Return value

Type:

the number of rows updated.

3 accountAggregateService

This *Service* provides *Account* update and delete methods which integrate with subscriber management systems (SMS) or conditional access systems (CAS, where necessary).

Calling these methods uses the standard *Account* Service to alter the *Entity* in SDP, but also notifies the CAS where necessary through the *SMSgw* Service.

Note

You cannot create an *Account* using this *Service* because an *Account* must always be created together with its first *User*. Use the *Workflow Subscription* Service instead.

Updating Accounts

The `updateAccount` method can be called either with a list of parameters, or with a pre-populated model.

If CAS integration is enabled, calling `updateAccount` will automatically notify the CAS about changes that affect it:

- ▶ If `status` is changed from `Active` or `Restricted` to `Suspended`, `Inactive` or `Cancelled`, all the smartcards for the account will be suspended, and all the set-top boxes will be disabled.
- ▶ If `status` is changed from `Suspended`, `Inactive` or `Cancelled` to `Active` or `Restricted`, all the smartcards for the account will be reactivated, and all the set-top boxes will be enabled.
- ▶ If `postcode` is changed, the CAS will be instructed to update the zip code for all the smartcards for the account.

Deleting Accounts

This method updates `status` to `Cancelled` rather than performing a physical delete on the database.

If CAS integration is enabled, calling `deleteAccount` will instruct the CAS to suspend all the smartcards for the account and disable all the set-top boxes.

3.1 accountAggregateService

AggregateService for the account service.

zlaposa @author

Miguel Gil (BNET Conax Cas) @author

Ashish Gupta @author

Will Dickson @author

Miguel Gil (Credit Limit) @author

3.1.1 accountAggregateService.addAccountProfile

Adds a profile to an `Account` identified by the account number (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|--|
| accountNumber | String | The account number of the Account to update |
| profileName | String | |

Return value

Type: void

Example

```
<addAccountProfile>
  <accountNumber>ABC-32765</accountNumber>
  <profileName>stringValue</profileName>
</addAccountProfile>
```

3.1.2 accountAggregateService.createAccount

Create, to storage, a new **Account** and return back the generated identifier of the account (UID) ..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|----------------|-------------------|
| account | <u>Account</u> | Account to update |

Return value

Type:

The number of rows updated

3.1.3 accountAggregateService.deleteAccount

Cancels an **Account**

The *Account* to update is identified by the **uid** supplied in the model object.

This method sets the **status** of the *Account* to **CANCELLED**, rather than physically deleting the *Entity* from the database. This enables any outstanding billing actions to complete, and ensures the integrity of the operator's records.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|----------------|---|
| account | <u>Account</u> | A model object representing the <i>Account</i> to delete. |

Return value

Type:

The number of *Accounts* affected: either 0 or 1.

3.1.4 accountAggregateService.getAccountProfileFeatures *New*

Overview

AccountAggregateService for get the accountProfileFeatures based on the accountUid.

This operation will work based on the accountUid and returns the list of features.

Description

This operation requires as mandatory inputs accountUid and returns list of features as out put in the response.

This operation can be called in the following scenarios:

1. if provide valid account uid in the request - operation validates the account uid and returns the list of account features in the response section.
2. For the subsequent if not provide valid account uid then returns the error in the response.

Definition

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/ws/accountAggregateService/getAccountFeatureProfiles>

Request input parameters

| Name | Type | Description | Cardinality |
|------|------|-------------|-------------|
| arg0 | Long | AccountUid | mandatory |

Requests examples

1.Request with only mandatory fields:

```
<soapenv:Body>
<acc:getDeleteMpRemainingCountByUID>
<uid>689</uid>
</acc:getDeleteMpRemainingCountByUID>
```

</soapenv:Body>

Response example

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">1989300327</rqid:request
ID>
  </env:Header>
  <env:Body>
    <ns2:getAccountProfileFeaturesResponse xmlns:ns2="http://quative.tv/Account
AggregateServiceNamespace">
      <return>
        <creationDate>2019-09-09T10:34:14Z</creationDate>
        <originID>1</originID>
        <originKey>FEATURE1</originKey>
        <serviceProviderID>1</serviceProviderID>
        <UID>1</UID>
        <enabled>true</enabled>
        <name>FEATURE1</name>
        <type>S</type>
      </return>
    </ns2:getAccountProfileFeaturesResponse>
  </env:Body>
</env:Envelope>
```

3.1.5 accountAggregateService.removeAccountProfile

Removes a profile from an `Account` identified by the account number (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|--|
| <code>accountNumber</code> | String | The account number of the <code>Account</code> to update |
| <code>profileName</code> | String | |

Return value

Type: void

Example

```
<removeAccountProfile>
```

```
<accountNumber>ABC-54340</accountNumber>
<profileName>stringValue</profileName>
</removeAccountProfile>
```

3.1.6 accountAggregateService.updateAccount

Update the fields of an *Account*.

The *Account* to update is identified by the `uid` supplied in the model object.

SDP uses optimistic locking to ensure that updates do not conflict. You should always get a fresh copy of an *Account* before making changes.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|-----------------------|--|
| <code>account</code> | <u><i>Account</i></u> | A model object containing all the fields of the <i>Account</i> , updated as necessary. |

Return value

Type:

The number of *Accounts* affected: either 0 or 1.

4 accountPaymentCredentialsService

This is a primitive *Service* which provides the *AccountPaymentCredentials* Entity.

4.1 AccountPaymentCredentials

An *AccountPaymentCredentials* Entity represents the credentials of a subscriber with a specific *PaymentMethodProvider*.

Not all *PaymentMethodProviders* require *Account* credentials to be stored in SDP: some (like the default "post-paid" payment option) do not require credentials at all, whilst others (like the PayWizard E-wallet payment option) expect credentials to be entered by the user at purchase time.

Where a *ServiceProvider* has chosen to work with a *PaymentMethodProvider*, its customers will not be able to use that *PaymentMethodProvider* for purchases until an *AccountPaymentCredentials* Entity has been created for them. This must be done by the SMS.

When browsing, end users will see all the available payment options, regardless of whether they have registered appropriate credentials. The client is responsible for letting users know which options are available immediately and which require further action to activate.

4.1.1 AccountPaymentCredentials

| Name | Type | Description |
|-------------------------|---|--|
| <code>accountUid</code> | Long | The <i>Account</i> that these credentials belong to. |
| <code>cardExpiry</code> | String | For card-based payment providers, the expiry date as printed on the card (09/12...) Used for display only, so that users can verify which card will be charged. |
| <code>cardPan</code> | String | For card-based payment providers, the card number, with the first 12 digits hidden. Used for display only, so that users can verify which card will be charged. <i>For example: **** * 3244</i> |
| <code>cardToken</code> | String | A token generated by the payment provider, which authorizes SDP to take payments from the card. |
| <code>cardType</code> | String | For card-based payment providers, the type of card. (Visa, Mastercard...) Used for display only, so that users can verify which card will be charged. |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |

| Name | Type | Description |
|--|-------------------|---|
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created. |
| <code>customerRef</code> | String | The identifier used for the customer (that is, <i>Account</i> holder) in the payment provider's system. |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>paymentMethod ProviderUid</code> | Long | The <code>uid</code> of the <i>Payment Method Provider</i> representing the payment provider. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |

4.2 accountPaymentCredentialsService

Implementation of `AccountPaymentCredentialsService` to provide CRUD operations for `AccountPaymentCredentials`.

mharvey @author

4.2.1 accountPaymentCredentialsService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

4.2.2

accountPaymentCredentialsService.createAccountPaymentCredentials

Create an *AccountPaymentCredentials* Entity.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|---|--|
| pAccPayCred | AccountPaymentCredentials | ^{p.7A} An object specifying the fields of the <i>AccountPayment Credentials</i> . |

Return value

Type:

The `uid` of the created *AccountPaymentCredentials*.

4.2.3 accountPaymentCredentialsService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

4.2.4 accountPaymentCredentialsService.deleteAccountPaymentCredentials

Delete the *AccountPaymentCredentials* with the specified UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|--|
| pAccPayCredUid | Long | The uid of the <i>AccountPaymentCredentials</i> to delete. |

Return value

Type:

Example

```
<deleteAccountPaymentCredentials>
  <pAccPayCredUid>87</pAccPayCredUid>
</deleteAccountPaymentCredentials>
```

4.2.5 accountPaymentCredentialsService.getAccountPaymentCredentials

Get the *AccountPaymentCredentials* for the supplied UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|--|
| pAccPayCredUid | Long | the ID of the <i>AccountPaymentCredentials</i> . |

Return value

Type:

the *AccountPaymentCredentials*.

Example

```
<getAccountPaymentCredentials>
  <pAccPayCredUid>54</pAccPayCredUid>
```

```
</getAccountPaymentCredentials>
```

4.2.6 accountPaymentCredentialsService.getByOrigin

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|-------------|
| originKey | String | |
| originID | Integer | |

Return value

Type:

Example

```
<getByOrigin>  
  <originKey>stringValue</originKey>  
  <originID>60</originID>  
</getByOrigin>
```

4.2.7 accountPaymentCredentialsService.getByUID

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | |

Return value

Type:

Example

```
<getByUID>  
  <uid>67</uid>  
</getByUID>
```

4.2.8 accountPaymentCredentialsService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

4.2.9

accountPaymentCredentialsService.updateAccountPaymentCredentials

Update an AccountPaymentCreentials..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|---|--------------------------|
| pAccPayCred | AccountPaymentCredentials | the updated credentials, |

Return value

Type:

the number of rows updated.

5 accountService

This is a primitive **Service** which provides the **Account** Entity.

It also manages relationships between **Accounts** and **Roles**.

Relationships between **Accounts**, **Users**, and **Access Points** are handled by the **Presence** Service.

Relationships between **Accounts** and **Set Top Boxes** are handled by the **Provision** Service.

Subscriptions to broadcast and on-demand **Entities** are handled by the **BTV** and **COD** Services, respectively.

Note

You cannot create an **Account** using this **Service** because an **Account** must always be created together with its first **User**. Use the **Workflow Subscription** Service instead.

Caution!

If you update an **Account** using this **Service**, external systems will not be notified. The **Account Aggregate** Service provides methods which integrate with these systems.

Avoiding Stale Objects

UpdateAccount Method

The updateAccount method of the AccountAggregateService is the recommended API for updating an account. The following example shows how the npvrProfile can be updated using this method.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:acc="http://quative.tv/AccountAggregateServiceNamespace">
  <soapenv:Header/>
  <soapenv:Body>
    <acc:updateAccount>
      <!--Optional:-->
      <account>
        <modifiedDate>2014-08-12T07:35:13+01:00</modifiedDate>
        <originKey>910-5754355</originKey>
        <serviceProviderID>3</serviceProviderID>
        <UID>552</UID>
        <accessPointUID>?</accessPointUID>
        <accountNumber>910-5754355</accountNumber>
        <npvrProfile>NPVR_NONE</npvrProfile>
      </account>
    </acc:updateAccount>
  </soapenv:Body>
</soapenv:Envelope>
```

Example SOAP envelope.

5.1 tv.quative.service.account.model.ApplicabilityPeriod

| Name | Type | Description |
|---------|------|---------------------------------------|
| DAILY | | DAILY and number of days. |
| MONTHLY | | WEEKLY and number of days in a month. |
| WEEKLY | | WEEKLY and number of days. |

5.2 accountService

Provides all of the required functionality to Create, Retrieve, Update and Delete 'Account' entities on the system.

Steve @author

Naeem Bilal @author

5.2.1 accountService.addQumRole

Add a qum role to the account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------------|
| accountUID | Long | the uid of the account |
| roleUID | Long | the uid of the role |

Return value

Type:

Integer number of qum roles added to account.

Example

```
<addQumRole>
  <accountUID>74</accountUID>
  <roleUID>88</roleUID>
</addQumRole>
```


5.2.2 accountService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

5.2.3 accountService.delete *Updated*

Delete, from storage, a **BaseModel**..

Available through Web Services.

This API remove the account from the Database. It mandatory to delete all users & devices associated with the account before issue the delete account.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

5.2.4 accountService.getAccByUserUID

Gets the *Account* which is associated to the specified *User*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/accountService/getAccByUserUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|------|------------------------------|
| usruid | Long | The uid of the <i>User</i> . |

Return value

Type:

The matching *Account*, or null if none is found.

Example

```
<getAccByUserUID>
  <usruid>11</usruid>
</getAccByUserUID>
```

5.2.5 accountService.getAccNoByRolloutProfileUid

This method retrieves the list of account numbers associated to a rollout profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|------|--|
| rolloutProfileUid | Long | the uid of the rollout profile assigned to the account(s). |

Return value

Type: `List<String>`

List of account numbers associated to that rollout profile.

Example

```
<getAccNoByRolloutProfileUid>
  <rolloutProfileUid>8</rolloutProfileUid>
</getAccNoByRolloutProfileUid>
```

5.2.6 accountService.getApplicabilityPeriod

This method gets the applicability period..

Available through Web Services.

Parameters

None.

Return value

Type:

ApplicabilityPeriod as enum object.

Example

```
<getApplicabilityPeriod/>
```

5.2.7 accountService.getByAccNoAccPassWithoutServiceProvider

Retrieve, from storage, a parent **Account** identified by account Number and provided spid..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--------|-----------------------|
| accountNumber | String | The Account Number. |
| accountPassword | String | The Account Password. |

Return value

Type:

Account If found, the matching object, null otherwise.

Example

```
<getByAccNoAccPassWithoutServiceProvider>
<accountNumber>ABC-95210</accountNumber>
<accountPassword>stringValue</accountPassword>
</getByAccNoAccPassWithoutServiceProvider>
```

5.2.8 accountService.getByAccountNumber

Gets the **Account** with the specified **accountNumber**.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|---------------------|
| accountNumber | String | The account number. |

Return value

Type:

The matching **Account**, or null if none is found.

Example

```
<getByAccountNumber>
  <accountNumber>ABC-46867</accountNumber>
</getByAccountNumber>
```

5.2.9 accountService.getByAccountNumberAndServiceProviderID

Retrieve, from storage, a parent **Account** identified by account Number and provided spid..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|-------------------------------|
| accountNumber | String | The Account Number. |
| spid | Long | Long The Service Provider ID. |

Return value

Type:

Account If found, the matching object, null otherwise.

Example

```
<getByAccountNumberAndServiceProviderID>
  <accountNumber>ABC-37304</accountNumber>
  <spid>39</spid>
</getByAccountNumberAndServiceProviderID>
```

5.2.10 accountService.getByAccountNumberAnyServiceProvider

Retrieve, from storage, a parent **Account** identified by account Number.

WARNING: Service Provider ID is not used in this method.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|---------------------|
| accountNumber | String | The Account Number. |

Return value

Type: List<Account>

List If found, the matching object, null otherwise.

Example

```
<getByAccountNumberAnyServiceProvider>
  <accountNumber>ABC-22689</accountNumber>
</getByAccountNumberAnyServiceProvider>
```

5.2.11 accountService.getByAccountUIDWithoutSpid

Retrieve, from storage, a List of **Account** entities By their UID and not with SPID..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--|
| uid | Long | The UID of the Account to retrieve. |

Return value

Type:

Account If found, the matching objects, null otherwise.

Example

```
<getByAccountUIDWithoutSpid>
  <uid>63</uid>
</getByAccountUIDWithoutSpid>
```

5.2.12 accountService.getByCreationDate

Retrieve, from storage, a List of **Account** entities identified by their creation date..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>starttime</code> | String | A string representation of the start of the date range to retrieve by |
| <code>endtime</code> | String | A string representation of the end of the date range to retrieve by |
| <code>dateformat</code> | String | The date format e.g. 'dd/mm/yyyy:HH:MI:SS' |

Return value

Type: `List<Account>`

List If found, the matching objects, null otherwise.

Example

```
<getByCreationDate>
<starttime>stringValue</starttime>
<endtime>stringValue</endtime>
<dateformat>stringValue</dateformat>
</getByCreationDate>
```

5.2.13 accountService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |

| Name | Type | Description |
|-----------------------|---------|--|
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>81</originID>
</getByOrigin>
```

5.2.14 accountService.getByStatus

Gets all the *Account* Entities with the specified **status**.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>status</code> | String | The status of the <i>Accounts</i> to retrieve. A verbose <i>Account Status</i> value: one of INACTIVE , ACTIVE , RESTRICTED , SUSPENDED or CANCELLED . |

Return value

Type: `List<Account>`

A list of all matching *Accounts* (may be empty.)

Example

```
<getByStatus>
  <status>stringValue</status>
</getByStatus>
```

5.2.15 accountService.getByUID

Gets the *Account* with the specified **uid**.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/accountService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|---------------------------------|
| uid | Long | The uid of the <i>Account</i> . |

Return value

Type:

The matching *Account*, or null if none is found.

Example

```
<getByUID>
  <uid>89</uid>
</getByUID>
```

5.2.16 accountService.getCountByRolloutProfileUid

This method gets the count of accounts associated to a rollout profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|------|--|
| rolloutProfileUid | Long | the uid of the rollout profile assigned to the account(s). |

Return value

Type:

Long count of how many accounts are associated to that rollout profile.

Example

```
<getCountByRolloutProfileUid>
  <rolloutProfileUid>1</rolloutProfileUid>
</getCountByRolloutProfileUid>
```


5.2.17 accountService.getCreditUsageResetDate

This method gets the get credit usage reset date..

Available through Web Services.

Parameters

None.

Return value

Type:

Date as next credit usage reset date.

Example

```
<getCreditUsageResetDate/>
```

5.2.18 accountService.getDeleteMpRemainingCountByUID *New*

Overview

AccountService for the getDeleteMpRemainingCountByUID API.

This operation will get the count of undeleted Media player.

Description

This operation requires Uid in request section and returns the unlimited in the response.

This operation can be called in the following scenarios:1) For depend on the uid will get the count of

Definition

Available through HUE at <http://<server>:<port>/ws-gateway/gateway/ws/accountservice/getDeleteMpRemainingCountByUID>

Request input parameters

| Name | Type | Description | Cardinality |
|------|------|------------------|-------------|
| arg0 | Long | uid (Device uid) | mandatory |

Requests examples

Request with uid field:

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:acc="http://quative.tv/AccountServiceNamespace"><soapenv:Header/><soapenv:Body><acc:getDeleteMpRemainingCountByUID><uid>689</uid></acc:getDeleteMpRemainingCountByUID></soapenv:Body></soapenv:Envelope>
```

Response example

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">1715742873</rqid:requestID>
  </env:Header>
  <env:Body>
    <ns2:getDeleteMpRemainingCountByUIDResponse xmlns:ns2="http://quative.tv/AccountServiceNamespace">
      <return>unlimited</return>
    </ns2:getDeleteMpRemainingCountByUIDResponse>
  </env:Body>
</env:Envelope>
```

5.2.19 accountService.getRecItemsCreditUsageByAccountUID

This method get called every after applicability period, where we use call this method only after reading value 'F' of COR_ACCOUNT.CREDIT_SPENT_RST.

Normally this value remain 'T' but there is one Oracle job run after every applicability period and make all accounts 'T' to 'F'. This approach is used to avoid more call on this method.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------------|
| accountUID | Long | the uid of the account |

Return value

Type:

recursive items credit usage by an account.

Example

```
<getRecItemsCreditUsageByAccountUID>
  <accountUID>39</accountUID>
</getRecItemsCreditUsageByAccountUID>
```

5.2.20 accountService.getRoleUIDListByAccountUID

Retrieves the list of role uids for an account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------------|
| accountUID | Long | the uid of the account |

Return value

Type: List<Long>

List<Long> the list of role uids associated to the account.

Example

```
<getRoleUIDListByAccountUID>
  <accountUID>73</accountUID>
</getRoleUIDListByAccountUID>
```

5.2.21 accountService.purgeAccount *New*

Overview

Purges for Account Service.

This operation will delete the account related data.

Description

This operation requires any one of the inputs need to provide AccountNumber or AccountUid or OrginUid/OrginKey pair and returns the 1 ['1' means success]in the response.

This operation can be called in the following scenarios:1) If provide an accountNumber only in the request operation validate and purge the account data and return 1 in the response section.2) If provide an accountUID only in the request operation validate and purge the account data and return 1 in the response section.3) If provide an orginUid/Orgin Key pair only in the request operation validate and purge the account data and return 1 in the response section. 4) If provide all input parameter are valid data in the request operation validate and purge the account data and return 1 in the response section. 5) If not provide any of the input parameter in request then it returns error in the response section.6) If provide all input parameters in request but a/c number and a/c UID not matched as per DB then returns error in response section.

Definition

Available through HUE at <http://<server>:<port>/ws-gateway/gateway/ws/accountservice/purgeAccount>

Request input parameters

| Name | Type | Description | Cardinality |
|------|--------|---------------|-------------|
| arg0 | String | accountNumber | optional |

| Name | Type | Description | Cardinality |
|------|--------|-------------|---|
| arg1 | Long | accountUid | optional |
| arg2 | Long | originUid | optional(if we provide origin uid then origin key is mandatory) |
| arg3 | String | originKey | optional(if we provide origin key then origin uid is mandatory) |

Requests examples Request with only account number Example

```
<soapenv:Body>
<acc:purgeAccount>
<!--Optional:-->
<accountSpecification>
<!--Optional:-->
<accountNumber>acc12345</accountNumber><originSpecification>
<!--Optional:-->
<!--Optional:-->
</originSpecification>
</accountSpecification>
</acc:purgeAccount>
</soapenv:Body>
```

Success Response example:
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"><env:Header><rqid:requestID xmlns:rqid="http://quative.tv/requestId">1310028996</rqid:requestID></env:Header><env:Body><ns2:purgeAccountResponse xmlns:ns2="http://quative.tv/AccountServiceNamespace"><return>1</return></ns2:purgeAccountResponse></env:Body></env:Envelope>

Error Response Example (If not provide any of the input just hit empty request it returns below error response):
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"><env:Header><rqid:requestID xmlns:rqid="http://quative.tv/requestId">1862582849</rqid:requestID></env:Header><env:Body><env:Fault><faultcode>env:Server</faultcode><faultstring>ErrorCode: [code=11823, severity=ERROR] Invalid AccountSpecification, need to provide accountUid or accountNumber or an originId/originKey pair.</faultstring><detail><ns2:ServiceException xmlns:ns2="http://quative.tv/AccountServiceNamespace"><chainedCode>11823</chainedCode><code/><message>ErrorCode: [code=11823, severity=ERROR] Invalid Account Specification, need to provide accountUid or accountNumber or an originId/originKey pair.</message><usageParams/></ns2:ServiceException></detail></env:Fault></env:Body></env:Envelope>

5.2.22 accountService.removeAllQumRoles

Remove all the qum role for an account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------------|
| accountUID | Long | the uid of the account |

Return value

Type:

Integer number of qum roles removed from the account.

Example

```
<removeAllQumRoles>
  <accountUID>68</accountUID>
</removeAllQumRoles>
```

5.2.23 accountService.removeQumRole

Remove the qum role from the account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------------|
| accountUID | Long | the uid of the account |
| roleUID | Long | the uid of the role |

Return value

Type:

Integer number of qum roles removed from the account.

Example

```
<removeQumRole>
  <accountUID>87</accountUID>
  <roleUID>57</roleUID>
</removeQumRole>
```

5.2.24 accountService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

6 acquiredContentListService

This is a primitive *Service* which provides the *Acquired Content List* Entity.

6.1 acquiredContentListService

AcquiredContentListService Implementation, implements the compulsory methods defined in the AcquiredContentListService interface. The acquired content list contains the subscriptions of accounts. The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="acquiredContentListService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The methods provide the functionality to create, retrieve, update and delete acquired content list entries.

zlaposa @author

Naeem Bilal @author

bsingh @author

Miguel Gil (NagraDVS) @author

6.1.1 acquiredContentListService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|-------------------------------------|-------------|
| model | AcquiredContentList | |

Return value

Type:

6.1.2 acquiredContentListService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

6.1.3 `acquiredContentListService.delete`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|----------------------------------|-------------|
| <code>model</code> | <code>AcquiredContentList</code> | |

Return value

Type:

6.1.4 `acquiredContentListService.delete`

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

6.1.5 `acquiredContentListService.getALCTypeACLByAccountUidAndSmartCardId`

Returns a Single ACL.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|-------------|
| accountUid | Long | |
| smartcardID | String | |
| productUid | Long | TODO |
| productType | String | TODO |

Return value

Type:

Example

```
<getALCTypeACLByAccountUidAndSmartCardId>
<accountUid>10</accountUid>
<smartcardID>ABC-83290</smartcardID>
<productUid>16</productUid>
<productType>stringValue</productType>
</getALCTypeACLByAccountUidAndSmartCardId>
```

6.1.6 acquiredContentListService.getByAccountUID

Returns a list of expired ACL entries belonging to the specified account.

Calls `getByAccountUIDWithDate(Date, Integer, String)` with the current date.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByAccountUID>

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--------|--|
| accountUID | Long | AccountUID |
| statusByDate | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `<code>AcquiredContentList</code>`s

Example

```
<getByAccountUID>
<accountUID>48</accountUID>
<statusByDate>stringValue</statusByDate>
</getByAccountUID>
```

6.1.7 `acquiredContentListService.getByAccountUIDAndItemType` *Updated*

Get the **Acquired Content List** entries of the specified type and expiry status which were acquired by the specified **Account**.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByAccountUIDAndItemType`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|---|
| <code>accountUID</code> | Long | The uid of the Account . |
| <code>statusByDate</code> | String | <p>The expiry status of the Acquired Content List entries to return. An Acquired Content List <code>statusByDate</code> value: one of ALL, ALL_INCL_UNSUB, CURRENT, EXPIRED or FUTURE.</p> <p>ALL</p> <p>(ALL) Include all subscribed Acquired Content List Entities, whether current, expired, or future.</p> <p>Note</p> <p>Expired Acquired Content List Entities are automatically unsubscribed after a short time. When this happens, they will no longer be included in ALL. To search for historical purchases, use ALL_INCL_UNSUB.</p> <p>ALL_INCL_UNSUB</p> <p>(ALL_INCL_UNSUB) Include all Acquired Content List Entities, whether current, expired, or future, and regardless of whether they are currently subscribed. status.</p> |

| Name | Type | Description |
|-----------------|--------|--|
| | | <p>Note</p> <p>Unsubscribed Acquired Content List Entities may be removed to an archive after some period of time, depending on the deployment's database policy. Once archived, they will no longer be available to any client.</p> <p>CURRENT</p> <p>(CURRENT) Include only Acquired Content List Entities which are currently valid: their validFrom date is in the past, and their expiryDate is in the future.</p> <p>EXPIRED</p> <p>(EXPIRED) Include only Acquired Content List Entities whose expiryDate has already passed.</p> <p>FUTURE</p> <p>(FUTURE) Include only Acquired Content List Entities whose validFrom date has not yet passed.</p> |
| itemType | String | <p>The type of the Acquired Content List entries to return. An Acquired Content List AcquiredContentListItemType value: one of AST, AGR, CAT, EVT, PKG, PLG or SUB.</p> |

Return value

Type: `List<AcquiredContentList>`

A list of the matching **Acquired Content List** Entities.

Example

```
<getByAccountUIDAndItemType>
  <accountUID>64</accountUID>
  <statusByDate>stringValue</statusByDate>
  <itemType>stringValue</itemType>
</getByAccountUIDAndItemType>
```

6.1.8

acquiredContentListService.getByAccountUIDAndItemTypeWithDate

Returns a list of ACL entries belonging to the specified account..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>baseDate</code> | Date | The expiration is calculated by this date. |
| <code>accountUID</code> | Long | AccountUID |
| <code>statusByDate</code> | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |
| <code>itemType</code> | String | Purchased item type |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `AcquiredContentList`s

6.1.9 `acquiredContentListService.getByAccountUIDWithDate`

Returns a list of expired ACL entries belonging to the specified account..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>baseDate</code> | Date | The expiration is calculated by this date. |
| <code>accountUID</code> | Long | AccountUID |
| <code>statusByDate</code> | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `AcquiredContentList`s

6.1.10

`acquiredContentListService.getByAccountUIDWithDateAndStatus`

Returns a list of ACL entries belonging to the specified account.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|---|
| <code>baseDate</code> | Date | The expiration is calculated by this date. |
| <code>accountUID</code> | Long | AccountUID |
| <code>statusByDate</code> | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |
| <code>status</code> | String | currently one of "S" or "U" or null for Subscribed or Unsubscribed or ignored respectively. |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `AcquiredContentList`s

6.1.11 `acquiredContentListService.getByItemAndAccountUID`

Returns the entry belonging to the specified account and item.

NOTE: This is the original version that does not discriminate on smartcardID (for WS backwards compatability if required).

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByItemAndAccountUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|--------|---------------------|
| <code>accountUID</code> | Long | Account UID |
| <code>purchasedItemUID</code> | Long | Purchased item UID |
| <code>purchasedItemType</code> | String | Purchased item type |

| Name | Type | Description |
|------------|------|-------------|
| profileUID | Long | profileUID |

Return value

Type: `List<AcquiredContentList>`

List `<AcquiredContentList>`; - list of ACLs.

Example

```
<getByItemAndAccountUID>
<accountUID>50</accountUID>
<purchasedItemUID>31</purchasedItemUID>
<purchasedItemType>stringValue</purchasedItemType>
<profileUID>18</profileUID>
</getByItemAndAccountUID>
```

6.1.12 acquiredContentListService.getByItemAndAccountUIDSmartCard

Returns the entry belonging to the specified account and item.

This API may return multiple record for Asset,AssetGroup and catalogue.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|---|
| accountUID | Long | Account UID |
| purchasedItemUID | Long | Purchased item UID |
| purchasedItemType | String | Purchased item type |
| profileUID | Long | profileUID |
| smartcardID | String | the associated smartcardID (or null if not required). |

Return value

Type: `List<AcquiredContentList>`

List `<AcquiredContentList>`; - list of ACL records.

Example

```
<getByItemAndAccountUIDSmartCard>
<accountUID>57</accountUID>
<purchasedItemUID>94</purchasedItemUID>
<purchasedItemType>stringValue</purchasedItemType>
<profileUID>36</profileUID>
<smartcardID>ABC-20029</smartcardID>
</getByItemAndAccountUIDSmartCard>
```

6.1.13 acquiredContentListService.getByItemAndAccountUIDWithDate

Returns the entry belonging to the specified account and item that is valid during the specified date.

NOTE: This is the original version that does not discriminate on smartcardID (for WS backwards compatibility if required).

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|---|
| accountUID | Long | Account UID |
| purchasedItemUID | Long | Purchased item UID |
| purchasedItemType | String | Purchased item type |
| profileUID | Long | profileUID |
| date | Date | Content date must be valid for this date. |

Return value

Type: List<AcquiredContentList>

<AcquiredContentList> - List of ACLs.

6.1.14 acquiredContentListService.getByItemAndAccountUIDWithDateSmartCard

Returns the entry belonging to the specified account and item that is valid during the specified date..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|---|
| accountUID | Long | Account UID |
| purchasedItemUID | Long | Purchased item UID |
| purchasedItemType | String | Purchased item type |
| profileUID | Long | profileUID |
| date | Date | Content date must be valid for this date. |
| smartcardID | String | the associated smartcardID (or null if not required). |

Return value

Type: `List<AcquiredContentList>`

<AcquiredContentList> - List of ACLs.

6.1.15 acquiredContentListService.getByItemAndSmartcard

Get an ACL using the item type for a smartcard..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByItemAndSmartcard>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|---|
| smartcard | String | The smartcard to get the ACL entry for. |
| itemId | Long | The item identifier. |
| itemType | String | The item type. |
| profileId | Long | The profile Id that the ACL entry has. |

Return value

Type: `List<AcquiredContentList>`

list The ACL entry list. This maybe null.

Example

```
<getByItemAndSmartcard>
  <smartcard>stringValue</smartcard>
  <itemId>90</itemId>
  <itemType>stringValue</itemType>
  <profileId>5</profileId>
</getByItemAndSmartcard>
```

6.1.16 `acquiredContentListService.getByItemAndSmartcardWithDate`

Returns the entry belonging to the specified smartcard and item that is valid during the specified date..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|--------|--|
| <code>smartcard</code> | String | The smartcard (ID) to get the ACL entries for. |
| <code>purchasedItemUID</code> | Long | Purchased item UID |
| <code>purchasedItemType</code> | String | Purchased item type |
| <code>profileUID</code> | Long | profileUID |
| <code>date</code> | Date | Content date must be valid for this date. |

Return value

Type: `List<AcquiredContentList>`

<AcquiredContentList> - List of ACLs.

6.1.17 `acquiredContentListService.getByItemTypeAndSmartcardValidForDates`

Get the ACL record for a smartcard with the item Id, item type and profile Id which is valid for the supplied dates.

Will look for current subscriptions only

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByItemTypeAndSmartcardValidForDates`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---|
| <code>smartcard</code> | String | The smartcard to get the ACL record for. |
| <code>itemId</code> | Long | The item Id to get the ACL record for. |
| <code>itemType</code> | String | The item type for the item. |
| <code>profileId</code> | Long | The profile Id to get the ACL record for. |
| <code>start</code> | Date | The start range to get the ACL record in. |
| <code>end</code> | Date | The end range to get the ACL record in. |

Return value

Type: `Collection<AcquiredContentList>`

The found ACL records.

6.1.18 `acquiredContentListService.getByOrigin`

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>78</originID>
</getByOrigin>
```

6.1.19 acquiredContentListService.getByPurchasedItemType

Returns ACL entries with the specified purchased item type..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|-------------|
| type | String | Type code. |

Return value

Type: List<AcquiredContentList>

List - List of <code>AcquiredContentList</code>s with the specified type

Example

```
<getByPurchasedItemType>
  <type>stringValue</type>
</getByPurchasedItemType>
```

6.1.20 acquiredContentListService.getBySmartcard

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getBySmartcard>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-------------|
| smartcard | String | |

| Name | Type | Description |
|--------------|--------|-------------|
| statusByDate | String | |

Return value

Type: `Collection<AcquiredContentList>`

Example

```
<getBySmartcard>
<smartcard>stringValue</smartcard>
<statusByDate>stringValue</statusByDate>
</getBySmartcard>
```

6.1.21 acquiredContentListService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

`BaseModel` If found, the matching object, null otherwise.

Example

```
<getByUID>
<uid>80</uid>
</getByUID>
```

6.1.22 acquiredContentListService.getByUserID

Returns a list of expired ACL entries belonging to the specified user.

Calls `getByUserIDWithDate(Date, Integer, String)` with the current date.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByUserID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>userID</code> | Long | UserUID |
| <code>statusByDate</code> | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `AcquiredContentList`s

Example

```
<getByUserID>
  <userID>76</userID>
  <statusByDate>stringValue</statusByDate>
</getByUserID>
```

6.1.23 `acquiredContentListService.getByUserIDAndItemType`

Get the *Acquired Content List* entries of the specified type and expiry status which were acquired for the specified *User*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByUserIDAndItemType`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>userID</code> | Long | The uid of the <i>User</i> . |
| <code>statusByDate</code> | String | The expiry status of the <i>Acquired Content List</i> entries to return. An <i>Acquired Content List statusByDate</i> value: one of ALL, CURRENT, EXPIRED or FUTURE. |

| Name | Type | Description |
|-----------------------|--------|---|
| <code>itemType</code> | String | The type of the <i>Acquired Content List</i> entries to return. An <i>Acquired Content List</i> <code>AcquiredContentListItem</code> Type value: one of AST, AGR, CAT, EVT, PKG, PLG or SUB. |

Return value

Type: `List<AcquiredContentList>`

A list of the matching *Acquired Content List* Entities.

Example

```
<getByUserUIDAndItemType>
  <userUID>66</userUID>
  <statusByDate>stringValue</statusByDate>
  <itemType>stringValue</itemType>
</getByUserUIDAndItemType>
```

6.1.24 `acquiredContentListService.getByUserUIDAndItemTypeWithDate`

Returns a list of ACL entries belonging to the specified user..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getByUserUIDAndItemTypeWithDate`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>baseDate</code> | Date | The expiration is calculated by this date. |
| <code>userUID</code> | Long | UserUID |
| <code>statusByDate</code> | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |
| <code>itemType</code> | String | Purchased item type |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `AcquiredContentList`s

6.1.25 `acquiredContentListService.getByUserIDWithDate`

Returns a list of expired ACL entries belonging to the specified user..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>baseDate</code> | Date | The expiration is calculated by this date. |
| <code>userID</code> | Long | UserUID |
| <code>statusByDate</code> | String | The status of the ACL entry according to the date (Values: EXPIRED, CURRENT, FUTURE) |

Return value

Type: `List<AcquiredContentList>`

List - List of expired `AcquiredContentList`s

6.1.26 `acquiredContentListService.getLiveSubscriptions`

Returns a list of expired ACL entries belonging to the specified account that have the specified policyUid..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|-------------|
| <code>channelUid</code> | Long | |
| <code>accountUID</code> | Long | |
| <code>spid</code> | Long | |

Return value

Type: `List<AcquiredContentList>`

Example

```
<getLiveSubscriptions>
<channelUid>23</channelUid>
<accountUID>51</accountUID>
```

```
<spid>87</spid>
</getLiveSubscriptions>
```

6.1.27 acquiredContentListService.getValidACIByItemAndAccountUID

Returns the valid entry belonging to the specified account and item.

(returns item which are not expired).

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/acquiredContentListService/getValidACIByItemAndAccountUID>

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|---|
| accountUID | Long | Account UID |
| purchasedItemUID | Long | Purchased item UID |
| purchasedItemType | String | Purchased item type |
| profileUID | Long | profileUID |
| smartcardID | String | the associated smartcardID (not optional - null will be included in the query). |

Return value

Type: `List<AcquiredContentList>`

<AcquiredContentList> - List of ACLs

Example

```
<getValidACIByItemAndAccountUID>
<accountUID>80</accountUID>
<purchasedItemUID>40</purchasedItemUID>
<purchasedItemType>stringValue</purchasedItemType>
<profileUID>33</profileUID>
<smartcardID>ABC-34241</smartcardID>
</getValidACIByItemAndAccountUID>
```


6.1.28 `acquiredContentListService.update` *Updated*

Available through Web Services.

The efficient way to update an ACL is to retrieve the ACL Model with a get API and send this model after adding the required modifications.

Parameters

| Name | Type | Description |
|--------------------|----------------------------------|-------------|
| <code>model</code> | <code>AcquiredContentList</code> | |

Return value

Type:

6.1.29 `acquiredContentListService.update`

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

7 Ad Management Service *Updated*

This aggregate **Service** manages the relationship between advertisement **Assets** and entertainment **Assets**.

Caution! ■

7.1 tv.quative.service.advertisement.adm.model.AdPlayType

| Name | Type | Description |
|------|------|----------------------------------|
| POST | | Advertisement play type as Post. |
| PRE | | Advertisement play type as Pre. |

7.2 tv.quative.service.cod.model.CodItem

| Name | Type | Description |
|---------------|--------|---|
| actors | String | The <code>actors</code> of the <i>Asset</i> . > |
| aspectRatio | String | The <code>aspectRatio</code> of the <i>Asset</i> . > |
| assetGroupUid | Long | The <code>uid</code> of the <i>Asset Group</i> for which this <i>Asset</i> was scheduled in the current <i>Catalogue</i> . It's possible for <i>Assets</i> to belong to more than one <i>Asset Group</i> in SDP. When scheduling an <i>Asset</i> into a <i>Catalogue</i> , data providers may optionally specify an <i>Asset Group</i> , which must be one of the <i>Asset Groups</i> associated with the <i>Asset</i> . Clients may use this information to group <i>Assets</i> visually. The NAGRA Media CMS never creates multiple <i>Asset Groups</i> for the same per <i>Asset</i> . If an <i>Asset</i> is in an <i>Asset Group</i> , the <i>Asset Group</i> is always returned in this field. |
| assetList | String | The <code>assetList</code> of the <i>Asset</i> . |

| Name | Type | Description |
|--|---|---|
| <code>assetOrder</code> | Integer | The <code>assetOrder</code> of the Asset. |
| <code>assetPlayType</code> | String | The <code>assetPlayType</code> of the Asset. |
| <code>assetTitle</code> | String | The name of the Asset. |
| <code>assetType</code> | <i>Search criterion.</i> String | The <code>assetType</code> of the Asset. |
| <code>audioMode</code> | String | The <code>audioMode</code> of the Asset. |
| <code>audioPid</code> | List<AssetAudioPid> | The <code>audioPid</code> of the Asset. |
| <code>audioSubtracks</code> | <i>Deprecated.</i> String | The <code>audioSubtracks</code> of the Asset. |
| <code>availabilityIntervalValue</code> | Long | This field should not be used. |
| <code>availabilityIntervalValueType</code> | String | This field should not be used. |
| <code>bestSalesIndex</code> | Long | The <code>bestSalesIndex</code> of the Asset. |
| <code>bitRate</code> | Long | |
| <code>casDescriptor</code> | <i>Deprecated.</i> String | The <code>casDescriptor</code> of the Asset. |
| <code>casId</code> | <i>Deprecated.</i> String | The <code>casId</code> of the Asset. |
| <code>catalogueUid</code> | Long | The <code>uid</code> of the <i>Catalogue</i> in which the Asset has been scheduled. |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | The <code>changeLog</code> of the Asset. |
| <code>contentRating</code> | AggregateRating | |
| <code>contentType</code> | String | The <code>contentType</code> of the Asset. |

| Name | Type | Description |
|----------------------------|---------------------------------|-----------------------------------|
| copyProtections | String | The copyProtections of the Asset. |
| copyright | String | The copyright of the Asset. |
| countries | String | The countries of the Asset. |
| creationDate | Date | The creationDate of the Asset. |
| cuBroadcast ChannelName | String | |
| cuBroadcastEnd Date | Date | |
| cuBroadcastStart Date | Date | |
| currencySymbol | String | This field should not be used. |
| definition | String | The definition of the Asset. |
| description | String | The description of the Asset. |
| directors | String | The directors of the Asset. |
| drmDescriptor | String | |
| drmId | String | |
| drmInstanceName | <i>Search criterion.</i> String | |
| dualMono | <i>Deprecated.</i> Boolean | The dualMono field of the Asset. |
| duration | <i>Deprecated.</i> Long | The duration of the Asset. |
| durationMillis | Long | |

| Name | Type | Description |
|-----------------|-------------------------------|--|
| encoding | String | The encoding of the Asset. |
| episode | String | |
| expiryDurations | List<ExpiryDuration> | A list of the expiry durations represented in the expiry <i>Intervals</i> of the <i>Base Prices</i> associated with the <i>Schedule</i> . This list is in the same order as the policyGroups and policyList fields. Clients should examine all three lists to discover information about the purchase methods available for the item. |
| exportID | Integer | The exportID of the Asset. |
| fileName | String | The fileName of the Asset. |
| fileSize | Long | The fileSize of the Asset. |
| genre | String | The genre of the Asset. |
| hearingImpaired | <i>Deprecated.</i> String | The hearingImpaired field of the Asset. |
| internalOrder | Integer | The internalOrder of the Asset. |
| language | String | The language field of the Asset. |
| locale | String | The locale for which fields of this model object have been returned. |
| longDescription | String | The longDescription of the Asset. |
| mainContentUID | <i>Search criterion.</i> Long | The mainContentUID of the Asset. |
| modifiedDate | Date | The modifiedDate of the Asset. |
| modifiedDateML | Date | The modifiedDateML of the Asset. |
| nPVR | <i>Deprecated.</i> Boolean | |

| Name | Type | Description |
|------------------------------------|--|--|
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> <code>String</code> | The <code>name</code> of the Asset. |
| <code>originID</code> | <code>Integer</code> | The <code>originID</code> of the Asset. |
| <code>originIdAndKey</code> | <code>String</code> | |
| <code>originKey</code> | <code>String</code> | The <code>originKey</code> of the Asset. |
| <code>ownerID</code> | <code>String</code> | The <code>ownerID</code> of the Asset. |
| <code>pVPurchase</code> | <code>boolean</code> | |
| <code>parentAclUID</code> | <code>Long</code> | This field is not used. |
| <code>parentPolicyGroup UID</code> | <code>Long</code> | This field is not used. |
| <code>parentalRating</code> | <code>Long</code> | The <code>parentalRating</code> of the Asset. |
| <code>pmtString</code> | <code>String</code> | The <code>pmtString</code> of the Asset. |
| <code>policyGroupUID</code> | <code>Long</code> | This field is not used. |
| <code>policyGroups</code> | <code>List<PolicyGroup></code> | A list of the <i>Policy Groups</i> associated with the <i>Schedule</i> . This list is in the same order as the <code>expiryDurations</code> and <code>policyList</code> fields. Clients should examine all three lists to discover information about the purchase methods available for the item. |
| <code>policyList</code> | <code>List<Policy></code> | A list of the <i>Base Prices</i> associated with the <i>Schedule</i> . This list is in the same order as the <code>policyGroups</code> and <code>expiryDurations</code> fields. Clients should examine all three lists to discover information about the purchase methods available for the item. |
| <code>price</code> | <code>Double</code> | This field is not used. |

| Name | Type | Description |
|----------------------------------|---------------------------|--|
| <code>priceExpirationDate</code> | Date | This field is not used. |
| <code>privateMetadata</code> | String | |
| <code>producers</code> | String | The <code>producers</code> of the Asset. |
| <code>profileUID</code> | Long | This field is not used. |
| <code>programId</code> | String | The <code>programId</code> of the Asset. |
| <code>promoUrl</code> | String | The <code>promoUrl</code> of the Asset. |
| <code>rating</code> | Rating | The <code>rating</code> of the Asset (a <i>Rating</i> UID.) |
| <code>recordable</code> | Boolean | The <code>recordable</code> field of the Asset. |
| <code>scheduleEndTime</code> | Date | The <code>endTime</code> of the <i>Schedule</i> : when this item will stop being available in the <i>Catalogue</i> . |
| <code>scheduleStartTime</code> | Date | The <code>startTime</code> of the <i>Schedule</i> : when this item started being available in the <i>Catalogue</i> . |
| <code>scheduleStatus</code> | String | The <code>status</code> field of the <i>Schedule</i> : always P (PUBLISHED) when returned in a <i>CodItem</i> . |
| <code>scheduleuid</code> | Long | The <code>uid</code> of the <i>Schedule</i> . |
| <code>scoring</code> | Integer | The <code>scoring</code> field of the Asset. |
| <code>seriesId</code> | <i>Deprecated.</i> String | The <code>seriesId</code> field of the Asset. |
| <code>serviceProviderID</code> | Long | The <code>serviceProviderID</code> of the Asset. |
| <code>shortTitle</code> | String | The <code>shortTitle</code> of the Asset. |
| <code>status</code> | String | The <code>status</code> field of the Asset: always A (ACTIVE) when returned in a <i>CodItem</i> . |

| Name | Type | Description |
|-----------------------------------|--|---|
| <code>studio</code> | <code>String</code> | The <code>studio</code> of the <i>Asset</i> . |
| <code>subTitles</code> | <code>Boolean</code> | The <code>subTitles</code> field of the <i>Asset</i> . |
| <code>subscribableItemType</code> | <code>String</code> | The <code>subscribableItemType</code> of the <i>Asset</i> : always <code>AST</code> . |
| <code>subscribed</code> | <code>boolean</code> | Is this <i>Account</i> already subscribed to this <i>Asset</i> ? |
| <code>subtitleLanguages</code> | <i>Deprecated.</i> <code>String</code> | The <code>subtitleLanguages</code> field of the <i>Asset</i> . |
| <code>totalDownloads</code> | <code>Integer</code> | The <code>totalDownloads</code> field of the <i>Asset</i> . |
| <code>totalRequests</code> | <code>Long</code> | |
| <code>trickPlay</code> | <code>TrickPlay</code> | The <code>trickPlay</code> field of the <i>Asset</i> . |
| <code>uID</code> | <code>Long</code> | The <code>uID</code> field of the <i>Asset</i> . |
| <code>url</code> | <code>String</code> | The <code>url</code> of the <i>Asset</i> . |
| <code>viewingNumber</code> | <code>Integer</code> | The <code>viewingNumber</code> of the <i>Asset</i> . |
| <code>year</code> | <i>Search criterion.</i> <code>String</code> | The <code>year</code> field of the <i>Asset</i> . |

7.3 tv.quative.service.cod.model.ExpiryDuration

| Name | Type | Description |
|------------------------|---------------------|--|
| <code>value</code> | <code>Long</code> | The value of the expiry <i>Interval</i> , in the unit specified by the <code>valueType</code> . |
| <code>valueType</code> | <code>String</code> | The unit the expiry <i>Interval</i> is measured in. This is an <code>IntervalValueType</code> value: one of <code>YEAR</code> , <code>MONTH</code> , <code>WEEK</code> , <code>DAY</code> , <code>HOURL</code> , <code>MIN</code> , <code>SEC</code> , <code>DOW</code> , <code>FDOM</code> , <code>LDOM</code> , <code>MOY</code> or <code>FIXED</code> . |

7.4 adManagementService

Advertisement Management Service implementation. Implements the compulsory methods to deals with Advertisement Management. The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="adManagementService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The getters methods used to retrieve information using different data model could be accessible both the WebServices and the Hue.

Naeem Bilal @author

7.4.1 adManagementService.addAnAdvertisement

A Service Method will add advertisement of adPlayType type.

We don't have any UID to be generated in this operations hence this method is void.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/addAnAdvertisement`

Available through Web Services.

Parameters

| Name | Type | Description |
|--|---|--------------------------------|
| <code>entertainmentAssetUID</code> | Long | as Long |
| <code>advertisementAssetScheduleUID</code> | Long | as Long |
| <code>adPlayType</code> | tv.quative.service.advertisement.adm.model.AdPlayType | as AdPlayType ^{p.114} |

Return value

Type: void

7.4.2 adManagementService.addBulkAdvertisements

A Service Method will add List of advertisements..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/addBulkAdvertisements`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--------------------|-------------|
| adManagementList | List<AdManagement> | as List |

Return value

Type: void

7.4.3 adManagementService.addPostAdvertisement

A Service Method will add Post Advertisement.

We don't have any UID to be generated in this operations hence this method is void.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/addPostAdvertisement>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |
| postAdAssetScheduleUID | Long | as Long |

Return value

Type: void

Example

```
<addPostAdvertisement>
  <entertainmentAssetUID>22</entertainmentAssetUID>
  <postAdAssetScheduleUID>10</postAdAssetScheduleUID>
</addPostAdvertisement>
```

7.4.4 adManagementService.addPreAdvertisement

A Service Method will add Pre Advertisement.

We don't have any UID to be generated in this operations hence this method is void.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/addPreAdvertisement`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------------|------|-------------|
| <code>entertainmentAssetUID</code> | Long | as Long |
| <code>preAdAssetScheduleUID</code> | Long | as Long |

Return value

Type: void

Example

```
<addPreAdvertisement>
  <entertainmentAssetUID>35</entertainmentAssetUID>
  <preAdAssetScheduleUID>21</preAdAssetScheduleUID>
</addPreAdvertisement>
```

7.4.5 adManagementService.addPreAndPostAdvertisements

A Service Method will add Pre and Post Advertisement.

If any one of the Pre or Post is null, then the relevant null advertisement will not be added. We don't have any UID to be generated in this operations hence this method is void.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/addPreAndPostAdvertisements`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------------|------|-------------|
| <code>entertainmentAssetUID</code> | Long | as Long |
| <code>preAdAssetScheduleUID</code> | Long | as Long |

| Name | Type | Description |
|------------------------|------|-------------|
| postAdAssetScheduleUID | Long | as Long |

Return value

Type: void

Example

```
<addPreAndPostAdvertisements>
  <entertainmentAssetUID>28</entertainmentAssetUID>
  <preAdAssetScheduleUID>72</preAdAssetScheduleUID>
  <postAdAssetScheduleUID>30</postAdAssetScheduleUID>
</addPreAndPostAdvertisements>
```

7.4.6 adManagementService.deleteAdvertisements

A Service Method will delete all advertisements with the passed entertainmentAssetUID..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/deleteAdvertisements>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |

Return value

Type: void

Example

```
<deleteAdvertisements>
  <entertainmentAssetUID>69</entertainmentAssetUID>
</deleteAdvertisements>
```

7.4.7 adManagementService.deleteAnAdvertisement

A Service Method will delete an Advertisement with the passed type adPlayType..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/deleteAnAdvertisement`

Available through Web Services.

Parameters

| Name | Type | Description |
|--|---|------------------------------------|
| <code>entertainmentAssetUID</code> | Long | as Long |
| <code>advertisementAssetScheduleUID</code> | Long | as Long |
| <code>adPlayType</code> | tv.quative.service.advertisement.adf.model.AdPlayType | as AdPlayType <small>p.114</small> |

Return value

Type: void

7.4.8 adManagementService.deleteBulkAdvertisements

A Service Method will delete List of advertisements..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/deleteBulkAdvertisements`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|--------------------|-------------|
| <code>adManagementList</code> | List<AdManagement> | as List |

Return value

Type: void

7.4.9 adManagementService.deletePostAdvertisement

A Service Method will delete Post advertisement..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/deletePostAdvertisement`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |
| postAdAssetScheduleUID | Long | as Long |

Return value

Type: void

Example

```
<deletePostAdvertisement>
  <entertainmentAssetUID>86</entertainmentAssetUID>
  <postAdAssetScheduleUID>63</postAdAssetScheduleUID>
</deletePostAdvertisement>
```

7.4.10 adManagementService.deletePreAdvertisement

A Service Method will delete Pre advertisement..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/deletePreAdvertisement>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |
| preAdAssetScheduleUID | Long | as Long |

Return value

Type: void

Example

```
<deletePreAdvertisement>
  <entertainmentAssetUID>2</entertainmentAssetUID>
```

```
<preAdAssetScheduleUID>45</preAdAssetScheduleUID>
</deletePreAdvertisement>
```

7.4.11 adManagementService.deletePreAndPostAdvertisements

A Service Method will delete Pre and Post Advertisement.

If any one of the Pre or Post is null, then the relevant null advertisement will not be deleted.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/deletePreAndPostAdvertisements>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |
| preAdAssetScheduleUID | Long | as Long |
| postAdAssetScheduleUID | Long | as Long |

Return value

Type: void

Example

```
<deletePreAndPostAdvertisements>
<entertainmentAssetUID>13</entertainmentAssetUID>
<preAdAssetScheduleUID>46</preAdAssetScheduleUID>
<postAdAssetScheduleUID>80</postAdAssetScheduleUID>
</deletePreAndPostAdvertisements>
```

7.4.12 adManagementService.getAdvertisementsByDate

Gets the list of Advertisement by start and end date/time, with passed adPlayType..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|---|--------------------------------|
| entertainment AssetUID | Long | as Long |
| adPlayType | tv.quative.service.advertisements.adm.models.adPlayType | as AdPlayType ^{p.114} |
| advertisement StartDate | Date | as Date |
| advertisementEnd Date | Date | as Date |

Return value

Type: List<CodItem>

List of CodItem contains Asset and schedule related information.

7.4.13 adManagementService.getCurrentAdvertisement

Gets the current Advertisement valid for current date/time, with passed adPlayType we might find many current assets but only pick one..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|---|--------------------------------|
| entertainment AssetUID | Long | as Long |
| adPlayType | tv.quative.service.advertisements.adm.models.adPlayType | as AdPlayType ^{p.114} |

Return value

Type:

CodItem contains Asset and schedule related information.

7.4.14 adManagementService.getCurrentPostAdvertisement

Gets the Post Advertisement valid for current date/time.

We might find many current assets but only pick one.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/getCurrentPostAdvertisement`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------------|------|-------------|
| <code>entertainmentAssetUID</code> | Long | as Long |

Return value

Type:

CodItem contains Asset and schedule related information.

Example

```
<getCurrentPostAdvertisement>
<entertainmentAssetUID>3</entertainmentAssetUID>
</getCurrentPostAdvertisement>
```

7.4.15 adManagementService.getCurrentPreAdvertisement

Gets the Pre Advertisement valid for current date/time.

We might find many current assets but only pick one.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/getCurrentPreAdvertisement`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------------|------|-------------|
| <code>entertainmentAssetUID</code> | Long | as Long |

Return value

Type:

CodItem contains Asset and schedule related information.

Example

```
<getCurrentPreAdvertisement>
```

```
<entertainmentAssetUID>15</entertainmentAssetUID>
</getCurrentPreAdvertisement>
```

7.4.16 adManagementService.getPostAdvertisements

Gets all post Advertisements associated to entertainmentAssetUID..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/getPostAdvertisements`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |

Return value

Type: `List<CodItem>`

List of CodItem contains Asset and schedule related information.

Example

```
<getPostAdvertisements>
<entertainmentAssetUID>20</entertainmentAssetUID>
</getPostAdvertisements>
```

7.4.17 adManagementService.getPostAdvertisementsByDate

Gets the list of post Advertisement by start and end date/time..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/adManagementService/getPostAdvertisementsByDate`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |

| Name | Type | Description |
|------------------------|------|-------------|
| advertisementStartDate | Date | as Date |
| advertisementEndDate | Date | as Date |

Return value

Type: List<CodItem>

List of CodItem contains Asset and schedule related information.

7.4.18 adManagementService.getPreAdvertisements

Gets all pre Advertisements associated to entertainmentAssetUID.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| entertainmentAssetUID | Long | as Long |

Return value

Type: List<CodItem>

List of CodItem contains Asset and schedule related information.

Example

```
<getPreAdvertisements>
  <entertainmentAssetUID>88</entertainmentAssetUID>
</getPreAdvertisements>
```

7.4.19 adManagementService.getPreAdvertisementsByDate

Gets the list of Pre Advertisement by start and end date/time..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|------|-------------|
| entertainment AssetUID | Long | as Long |
| advertisement StartDate | Date | as Date |
| advertisementEnd Date | Date | as Date |

Return value

Type: List<CodItem>

List of CodItem contains Asset and schedule related information.

8 aLaCarteService

This *Service* provides methods which allow set-top boxes to browse and subscribe to broadcast content. Only content marked for *à la carte* subscription may be accessed through this *Service*.

8.1 tv.quative.service.alacarte.model.ALaCarteProducts

| Name | Type | Description |
|----------|---------------|-------------------|
| packages | List<Package> | List of packages. |
| profiles | List<Profile> | List of profiles. |

8.2 Channel

A *Channel* represents a broadcast television channel.

Figure 2 Channel and associated Entities.

A *Channel* may be associated with a *Policy Group*. This *Policy Group* is used to define an additional price for the *Channel* in cases where it is marked as a premium *Channel* within a *Package*.

Tip

In SDP, a *Channel* represents a TV channel as understood in colloquial English: a complete broadcast signal which a device may tune to and receive. This is different from the definition used in DVB, which uses the term "service" for this; in DVB, "channels" represent the individual data streams which are multiplexed into a "service".

Channels can be grouped into *Categories* for navigation purposes, and *Packages* for subscription purposes. End users cannot subscribe to *Channels* individually, but only as part of a *Package*.

An *Event* links a *Channel* with a *Program*.

Channels may be ingested in bulk from Spring bean files.

Caution!

In systems using a CMS, *Channels* are created automatically by the ingest process. Broadcast content and pricing should be defined in the CMS.

The CMS ingest creates *Channels* from the CMS *Service* objects.

8.2.1 Channel

| Name | Type | Description |
|------------------------------|---------------------------------|--|
| <code>bandwidth</code> | Long | <p>The bit rate of this <i>Channel</i>.</p> <p>This field is always null for <i>Channels</i> which are not used in switched digital broadcast (SDV).</p> |
| <code>callLetters</code> | <i>Search criterion.</i> String | <p>The call letters of the <i>Channel</i>.</p> <p><i>For example:</i> BBC2, ESPN</p> <p>The CMS ingest places the <i>Service's</i> <code>shortName</code> attribute in this field, or its <code>id</code> if a <code>shortName</code> is not provided.</p> |
| <code>carryingNetwork</code> | String | <p>The unique identifier used by the eventual broadcaster for the <i>Channel</i>. Where possible, operators should use an identifier known to the set-top box, such as the channel's DVB triplet.</p> <p>The CMS ingest places the <i>Service's</i> <code>carryingNetwork</code> EPG attribute in this field.</p> <p>Note</p> <p>Although this field is not mandatory, SDP's pay-per-view (PPV) and start-over and catch-up (SOCU) features use it to identify <i>Channels</i>. Operators wishing to use these features must ensure that it is populated.</p> |
| <code>casID</code> | <i>Deprecated.</i> String | <p>For <i>Channels</i> protected by DRM, the identity of this <i>Channel</i> for the purposes of the DRM system which protects it.</p> <p>This field is deprecated and should not be used. It is not populated by the CMS ingest.</p> |
| <code>catchUpEnabled</code> | Boolean | <p>Is this <i>Channel</i> available for catch-up viewing?</p> <p>The implications of this field depend on the video streaming architecture. Some video servers, such as Harmonic, can be configured to record broadcast channels continuously into a buffer, which clients can draw from to obtain time-shifted streams. To implement catch-up, the client instructs the video server to begin streaming from the scheduled program start time and to end at the scheduled program end time.</p> <p>Catch-up features can be configured for individual technical <i>Channels</i>. Clients should check this field to determine whether it will be possible to offer the catch-up feature on a particular <i>Channel</i>.</p> |

| Name | Type | Description |
|----------------------------|--|---|
| | | <p>In some deployments, catch-up events are implemented as a kind of VOD Asset. This field may be set in these situations to let clients know when they will need to use SDP's VOD API to search for catch-up events on a channel.</p> <p>Tip The <code>cuBroadcastChannelName</code>, <code>cuBroadcastStartDate</code> and <code>cuBroadcastEndDate</code> fields of <code>Asset</code> are intended for this purpose.</p> |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | A log of the changes to this <i>Entity</i> history. |
| <code>cls</code> | <code>String</code> | <p>The class of the <i>Channel</i>, either <code>VT</code> for a technical <i>Channel</i> or <code>VE</code> for an editorial <i>Channel</i>.</p> <p>The CMS ingest sets this field to <code>VT</code> if the <i>Service</i> has a <code>mainChannelId</code> attribute, otherwise <code>VE</code>.</p> |
| <code>creationDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was created. |
| <code>definition</code> | <code>String</code> | <p>The definition of this <i>Channel</i>.</p> <p>For SDB channels, this must be a <code>tv.quative.service.channel.model.ChannelDefinition</code> value: one of <code>sd</code> or <code>hd</code>.</p> <p>This field is not ingested from the CMS.</p> |
| <code>description</code> | <code>String</code> | A short description of this <i>Entity</i> . |
| <code>drmDescriptor</code> | <code>String</code> | <p>For technical <i>Channels</i> protected by DRM, a descriptor used by the DRM system which protects this <i>Channel</i>.</p> <p>This field is not ingested from the CMS.</p> <p>This field is empty for editorial <i>Channels</i> and for technical <i>Channels</i> protected by a traditional CAS.</p> <p>See BZ#14136</p> |
| <code>drmID</code> | <code>String</code> | For technical <i>Channels</i> protected by DRM, the identity of this <i>Channel</i> for the purposes of the DRM system which protects it. |

| Name | Type | Description |
|------------------------|--|---|
| | | <p>The CMS ingest places the <i>Service</i> object's drmId in this field.</p> <p>This field is empty for editorial <i>Channels</i> and for technical <i>Channels</i> protected by a traditional CAS.</p> <p>See BZ#14136</p> |
| drmInstanceName | <i>Search criterion.</i> String | <p>An identifier for the DRM system which protects this technical <i>Channel</i>, if any.</p> <p>The CMS ingest places the <i>Service</i> object's drmInstanceName in this field.</p> <p>The client can use this name to determine how to process licenses returned by SDP. To do this, DRM instance names need to be defined by the operator beforehand and synchronized between the application, CMS and SMS.</p> <p>This field is empty for editorial <i>Channels</i> and for technical <i>Channels</i> protected by a traditional CAS.</p> <p>See BZ#14136</p> |
| encoding | String | <p>The codec to use for this <i>Channel</i>.</p> <p>For SDB channels, this must be a tv.quative.service.channel.model.ChannelEncoding value: one of mpeg2 or mpeg4.</p> <p>This field is not ingested from the CMS.</p> |
| exportID | Integer | <p>The identifier to use when exporting this <i>Entity</i> to third-party systems.</p> |
| locale | String | <p>When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned.</p> |
| mainContentUID | <i>Search criterion.</i> Long | <p>If this is a technical <i>Channel</i> the UID of the associated editorial <i>Channel</i>.</p> <p>If this field is empty, this is an editorial <i>Channel</i>.</p> <p>The CMS ingest creates both technical and editorial <i>Channels</i> as appropriate from the ingested <i>Service</i> objects. A <i>Service</i> object representing a technical <i>Channel</i> is linked to the object representing its editorial <i>Channel</i> through its mainChannelId attribute.</p> |

| Name | Type | Description |
|------------------------------|---|--|
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The name of the <i>Channel</i> . The CMS ingest places the <i>Service's Title</i> EPG attribute in this field for those locales for which it is defined, and uses the <code>longName</code> attribute for any other locales. |
| <code>networkLocation</code> | String | The location of the stream for this <i>Channel</i> on the network, using an addressing scheme known to the client. <i>For example:</i> <ul style="list-style-type: none"> ▶ For streams exposed as plain sockets, this may be the IP address and port number of the stream: <code>10.110.111.112:80</code> ▶ For MPTS streams, one multicast stream may carry multiple transport streams. In this case, the <i>Program ID</i> could be separated from the IP address and port by a slash character <code>/</code>: <code>10.110.111.112:8099/14</code> The CMS ingest places the <i>Service</i> object's <code>Network Location</code> EPG attribute in this field. |
| <code>number</code> | Integer | The number of the <i>Channel</i> , as seen by end users. The CMS ingest places the <i>Service's tvChannel</i> EPG attribute in this field. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character <code> </code> to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |

| Name | Type | Description |
|-----------------------------------|---------------------------|---|
| <code>originalNetwork</code> | String | The network which originated this <i>Channel</i> . The CMS ingest places the <i>Service</i> object's <code>originatingNetwork</code> EPG attribute in this field. |
| <code>ownerId</code> | String | A code representing the owner (content provider) for this <i>Channel</i> . The CMS ingest sets this field to the <code>ownerId</code> attribute of the <i>Service</i> . In deployments using NAGRA Media CMS, it must be no longer than 3 characters. |
| <code>pVVPurchase</code> | <i>Read only.</i> boolean | This field is always <code>false</code> for <i>Channels</i> . |
| <code>parentAclUID</code> | Long | This field should not be used. |
| <code>parentPolicyGroupUID</code> | Long | This field should not be used. |
| <code>policyGroupUID</code> | Long | The UID of an additional <i>Policy Group</i> to apply to this <i>Channel</i> when it is marked as premium within any <i>Package</i> . |
| <code>policyID</code> | <i>Deprecated.</i> Long | This field should not be used. |
| <code>privateMetadata</code> | String | This field is provided for operators to use for their own purposes. SDP passes it to the client unchanged. Its format (string, JSON object...) and meaning should be decided by the operator. The CMS ingest places the <i>Service</i> object's <code>Private Metadata</code> EPG attribute in this field. |
| <code>profileUID</code> | Long | This field should not be used. The UID of the <i>Channel Profile</i> describing this channel's features. The CMS ingest uses the <i>Service</i> object's <code>profileRef</code> attribute to link the appropriate <i>Channel Profile</i> . |
| <code>promoImage</code> | String | The location of a logo image for this <i>Channel</i> . The CMS ingest places the <i>Service</i> object's <code>Promo Images</code> EPG attribute in this field. |

| Name | Type | Description |
|-----------------------------------|---------------------------------|--|
| <code>ratingID</code> | Long | The <i>Rating</i> appropriate for this <i>Channel</i> . The CMS ingest uses the <i>Service</i> object's Rating EPG attribute to link the appropriate <i>Rating</i> . |
| <code>sdbCompatible</code> | <i>Read only.</i> boolean | This field should not be used. |
| <code>selfSubscribe</code> | Boolean | This field is deprecated and should no longer be used. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>sourceAddress</code> | String | The IP address and port number of the head end host which transmits this <i>Channel</i> , in dotted-quad notation. <i>For example:</i> 10.1.1.2:80 Typically this field specifies the unicast address and port which transmits the media stream to the multicast address given in <code>networkLocation</code> . This field is always null for <i>Channels</i> which are not used in switched digital broadcast (SDV). |
| <code>startOverEnabled</code> | Boolean | Is this <i>Channel</i> available for start-over viewing? The implications of this field depend on the video streaming architecture. Some video servers, such as Harmonic, can be configured to record broadcast channels continuously into a buffer, which clients can draw from to obtain time-shifted streams. To implement start-over, the client instructs the video server to begin streaming from the scheduled program start time. Start-over features can be configured for individual technical <i>Channels</i> . Clients should check this field to determine whether it will be possible to offer the start-over feature on a particular <i>Channel</i> . |
| <code>status</code> | <i>Search criterion.</i> String | The status of the <i>Channel</i> . A <code>tv.quative.service.channel.model.ChannelStatus</code> value: one of A or I . The CMS ingest sets this field to A if the <i>Service</i> object is marked active; otherwise I . |
| <code>subscribableItemType</code> | String | This field is always CHN for <i>Channels</i> . |

| Name | Type | Description |
|-------------------------|--|--|
| <code>subscribed</code> | <i>Read only.</i> <code>boolean</code> | When <i>Channels</i> are returned to clients, this field reports whether the viewer has purchased the <i>Channel</i> . This field is not part of the <i>Channel</i> Entity in the database. |
| <code>type</code> | <code>String</code> | The type of the <i>Channel</i> . The CMS ingest sets this field to BTV . |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |

8.3 Package

A *Package* represents a group of *Channels* which are subscribed to as a unit.

Figure 3 Package and associated Entities.

Channels in a *Package* may be standard or premium. Premium *Channels* attract an additional charge and must be subscribed to separately.

Every *Package* should be associated with a *Policy Group* to permit subscription. The *Policy Group* determines the price of the *Package* (which may be zero.)

A *Channel* must be part of a *Package* for an end-user to subscribe to it. Since a subscription is required to view any scrambled channel (even if this subscription is offered for free), only free-to-air *Channels* do not need to be in a *Package*. Even then, it is often convenient to create a *Package* for the free-to-air *Channels*, and subscribe every *Account* to this *Package*, so that applications do not have to implement special logic to browse these *Channels*.

Note

Packages have a similar role to on-demand *Catalogues*, but *Packages* may not be nested. *Subscription Profiles* can be used to provide a tree structure for *Packages* if required.

If a *Package* does not have a *Policy Group*, it cannot have a CAS product ID or attract a charge. This will only be the case for true free-to-air services. *Packages* of channels which are free of charge but whose access is managed by a CAS must have a zero price *Policy Group* applied to them.

Caution!

In systems using a CMS, *Packages* are created automatically by the ingest process. On-demand content and pricing should be defined in the CMS.

The CMS ingest creates *Packages* from the CMS *Products*.

Premium *Channels* are not currently supported by the CMS.

8.3.1 Package

| Name | Type | Description |
|--------------------------------------|--|--|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| channels | List<Channel> | List of channels associated with a Package. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| deferredExpiryDate | Date | The date this <i>Package</i> is due for unsubscription. SDP uses this field to manage minimum subscription periods. If a user unsubscribes from a <i>Package</i> before the minimum period has finished, unsubscription does not happen until this date has passed. Clients should not modify this field if it is set. |
| description | String | A short description of this <i>Entity</i> . |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| locale | String | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| minimumSubscriptionLengthIntervalUID | Long | An <i>Interval</i> representing the minimum time for which a subscription to this <i>Package</i> must be active. |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |
| modifiedDateML | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| name | <i>Search criterion.</i> Quick <i>Search criterion.</i> string | The name of this <i>Package</i> . The CMS ingest places the <i>Product</i> object's Title in this field. |
| originID | Integer | An identifier for the system which owns this <i>Entity</i> . The CMS ingest uses a fixed value of 1. |

| Name | Type | Description |
|-----------------------------------|--|--|
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. The CMS ingest sets this field to the <code>ID</code> of the <i>Product</i> . |
| <code>ppvPurchase</code> | <i>Read only.</i> <code>boolean</code> | Should this product be treated as a PPV product by the CAS? This field is managed by SDP and should not be changed. It should normally be <code>false</code> for <i>Packages</i> . |
| <code>parentAclUID</code> | <code>Long</code> | This field should not be used. |
| <code>parentPolicyGroupUID</code> | <code>Long</code> | This field should not be used. |
| <code>policyGroupUID</code> | <code>Long</code> | The <i>Policy Group</i> for this <i>Package</i> . A <i>Package</i> must have a <i>Policy Group</i> so that viewers will be able to subscribe to it. The CMS ingest creates <i>Policy Groups</i> from the information in the referenced <i>Products</i> and uses it in this field. |
| <code>policyID</code> | <i>Deprecated.</i> <code>Long</code> | This field should not be used. |
| <code>privateMetadata</code> | <code>String</code> | This field is provided for operators to use for their own purposes. SDP passes it to the client unchanged. Its format (string, JSON object...) and meaning should be decided by the operator. The CMS ingest does not populate this field. The <code>PrivateMetadata</code> EPG key of a <i>Product</i> object is ingested as the <code>privateMetadata</code> of the created <i>Policy Group</i> . |
| <code>profileUID</code> | <code>Long</code> | This field should not be used. |

| Name | Type | Description |
|-----------------------------------|--------------------------|--|
| <code>ratingID</code> | Long | The <i>Rating</i> for this <i>Package</i> ; normally the most inclusive <i>Rating</i> which is still appropriate for everything in the <i>Package</i> |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>status</code> | Search criterion. String | The activity status of this <i>Package</i> : one of A (Active) , I (Inactive) or S(Suspended) . The CMS ingest sets this field to A . |
| <code>subscribableItemType</code> | String | This field is always PKG for <i>Packages</i> . |
| <code>subscribed</code> | Read only. boolean | Is the user already subscribed to this item? This field is returned as part of the <i>Package</i> model object when clients query <i>Packages</i> . It is not part of the <i>Package</i> Entity itself. Note Attempting to alter this field will have no effect. To subscribe to a <i>Package</i> , use the dedicated subscription APIs or the <i>Modify Subscription</i> workflow in QSP Manager. |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>userSubscribable</code> | Read only. boolean | Is this <i>Package</i> available for <i>à la carte</i> subscription? If this box is checked, viewers will be able to see this <i>Package</i> when browsing through <i>à la carte</i> content. They will also be able to subscribe to this <i>Package</i> directly from their set-top box. Note SDP can limit the number of concurrent <i>à la carte</i> subscriptions a customer can have, whilst subscriptions to non- <i>à la carte Packages</i> are unlimited. If you want to offer a set of channels through both traditional and <i>à la carte</i> channels, you may prefer to create separate <i>Packages</i> for the two |

| Name | Type | Description |
|------|------|---|
| | | cases to avoid inadvertently using up the customer's allowance. |

8.4 Profile

A *Profile* is an abstract *Entity* representing a group of features or subscribable items.

Device Profiles, *Asset Profiles*, *Product Profiles* and *Subscription Profiles* are kinds of *Profile*.

8.4.1 Profile

| Name | Type | Description |
|---|--|---|
| <code>allProfiles</code> | <i>Read only.</i> <code>Hashtable<Long, Profile></code> | |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | A log of the changes to this <i>Entity</i> history. |
| <code>creationDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was created. |
| <code>deferredExpiryDate</code> | <code>Date</code> | what is the date this product should be unsubscribed on. |
| <code>description</code> | <code>String</code> | A short description of this <i>Entity</i> . |
| <code>enabled</code> | <code>Boolean</code> | Status of the profile. |
| <code>exportID</code> | <code>Integer</code> | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>minimumSubscriptionLengthIntervalUID</code> | <code>Long</code> | Identifier of the Interval that represents the minimum Subscription length that this package should be subscribe for. |

| Name | Type | Description |
|------------------------------------|--|---|
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>name</code> | <code>String</code> | The name of this <i>Entity</i> . |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>pVPurchase</code> | <i>Read only.</i> <code>boolean</code> | |
| <code>parentAclUID</code> | <code>Long</code> | UID of the parent ACL, this is used for Multiple, Season pass and Subscription ACLs. It refers to the ACL with the policy group, used when we need to unsubscribe (event or asset). |
| <code>parentPolicyGroup UID</code> | <code>Long</code> | UID of the parent policy group, if will be used if policy GroupUID is null andd profile UID is not null. |
| <code>policyGroupUID</code> | <code>Long</code> | Policy Group UID. |
| <code>profileUID</code> | <code>Long</code> | Profile UID. |
| <code>profiles</code> | <code>List<Profile></code> | Holder for the child profiles. |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>subscribableItem Type</code> | <code>String</code> | Indicates the type of the subscribable item. |

| Name | Type | Description |
|-------------------------------|--|---|
| <code>subscribed</code> | <i>Read only.</i> <code>boolean</code> | Flag indicating whether the user is already subscribed to this item. |
| <code>type</code> | <code>String</code> | Type of the profile. tv.quative.service.profile.model.ProfileType @see |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |
| <code>userSubscribable</code> | <i>Read only.</i> <code>boolean</code> | can the user self subscribe to this item . |

8.5 aLaCarteService

AlaCarteService Interface. A La Carte Service is an aggregate service which will facilitates all sort of A La Carte requests by encapsulating A La Carte business and calling underlying primitive services. The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass:

hue=true, ws=true) For the Methods: `@QubeService(name="AlaCarteService",`

ws=true) The getters methods used to retrieve information using different datamodel could be accessible both the Web Services and the Hue. `@QubeMethod(hue=false,`

Naeem Bilal @author

8.5.1 aLaCarteService.deferredUnsubscribe

This method is invoked by a Quartz trigger job Its purpose is to unsubscribe ALaCarte ACLs that have been marked for delete and have now expired..

Available through Web Services.

Parameters

None.

Return value

Type: `void`

Example

```
<deferredUnsubscribe/>
```

8.5.2 aLaCarteService.getAlaCarteProductList

Gets all the *Packages* and *Subscription Profiles* which are available for subscription by the signed-on *Account* and optional smart card.

This method returns an *AlacarteProduct* model object containing two lists, one containing the *Packages* which are available for *à la carte* subscription, and one containing the *Subscription Profiles* which are available for *à la carte* subscription.

This call also sets the *isSubscribed* field of each returned *Package* or *Subscription Profile*, to reflect whether that item is currently subscribed by the *Account* (or *Smart Card* if specified.)

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/getAlaCarteProductList`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID which the client intends to subscribe, if appropriate. If this field is null, the objects returned can be subscribed across the whole <i>Account</i> . |
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> , when this API is called through Web Services. If called through HUE, the signed-on <i>Account</i> is always used, so this parameter should be left null. |
| <code>locale</code> | String | The locale to use for the localizable fields of returned objects. |

Return value

Type:

Example

```
<getAlaCarteProductList>
  <smartcardId>stringValue</smartcardId>
  <accountId>stringValue</accountId>
  <locale>en_us</locale>
</getAlaCarteProductList>
```

8.5.3 aLaCarteService.getChannelListByProduct

get channel by product..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/getChannelListByProduct`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--------------------|
| <code>productUID</code> | Long | unique Identifier |
| <code>productType</code> | String | Profile or package |
| <code>locale</code> | String | as String |

Return value

Type: `List<Channel>`

ACL list of UIDs

Example

```
<getChannelListByProduct>
  <productUID>11</productUID>
  <productType>stringValue</productType>
  <locale>en_us</locale>
</getChannelListByProduct>
```

8.5.4 aLaCarteService.getPagedAlaCarteProductList

This method will return list of Subscribable items where Product List is specific to packages and the subscription profiles..

This method returns an *AlacarteProduct* model object containing two lists, one containing the *Packages* which are available for *à la carte* subscription, and one containing the *Subscription Profiles* which are available for *à la carte* subscription.

This call also sets the `isSubscribed` field of each returned *Package* or *Subscription Profile*, to reflect whether that item is currently subscribed by the *Account* (or *Smart Card* if specified.)

The `startRowNum` and `endRowNum` parameters treat the result set as a list which starts with the *Packages* and moves on to the *Subscription Profiles*. For example, if there are 10 *Packages* and 5 *Subscription Profiles*, asking for indexes 1..8 will return a list of 8 *Packages* and an empty list for the *Subscription Profiles*, whilst asking for indexes 8..15 will return a list of 3 *Packages* and a list of 5 *Subscription Profiles*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/getPagedAlaCarteProductList`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID which the client intends to subscribe, if appropriate. If this field is null, the objects returned can be subscribed across the whole <i>Account</i> . |
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> , when this API is called through Web Services. If called through HUE, the signed-on <i>Account</i> is always used, so this parameter should be left null. |
| <code>locale</code> | String | The locale to use for the localizable fields of returned objects. |
| <code>startRowNum</code> | Long | The index of the first record to return. |
| <code>endRowNum</code> | Long | The index of the last record to return. |

Return value

Type:

Example

```
<getPagedAlaCarteProductList>
  <smartcardId>stringValue</smartcardId>
  <accountId>stringValue</accountId>
  <locale>en_us</locale>
  <startRowNum>94</startRowNum>
  <endRowNum>71</endRowNum>
</getPagedAlaCarteProductList>
```

8.5.5 aLaCarteService.getPagedChannelListByProduct

Gets a page from the list of *Channels* which are included in a subscription to the specified *Package* or *Subscription Profile*.

If a *Subscription Profile* is specified, this call will recursively look for *Packages* contained in any contained *Subscription Profiles*, as well as *Packages* which are directly included in the *Subscription Profile*

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/getPagedChannelListByProduct`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>productUID</code> | Long | The <code>uid</code> of the <i>Package</i> or <i>Subscription Profile</i> to investigate. |
| <code>productType</code> | String | What type of product the <code>productUID</code> refers to: <code>PKG</code> if it is a <i>Package</i> , or <code>SUB</code> if it is a <i>Subscription Profile</i> . |
| <code>locale</code> | String | The locale to use for the localizable fields of returned objects. |
| <code>startRowNum</code> | Long | The index in the list of the first <i>Channel</i> to return. |
| <code>endRowNum</code> | Long | The index in the list of the last <i>Channel</i> to return. |

Return value

Type: `List<Channel>`

Example

```
<getPagedChannelListByProduct>
  <productUID>95</productUID>
  <productType>stringValue</productType>
  <locale>en_us</locale>
  <startRowNum>41</startRowNum>
  <endRowNum>48</endRowNum>
</getPagedChannelListByProduct>
```

8.5.6 aLaCarteService.getPagedProductListByChannel

Get a page from the list of à la carte subscription options which will enable the client to view a particular broadcast *Channel*.

This method returns an *AlacarteProduct* model object containing two lists, one containing the *Packages* which are available for à la carte subscription, and one containing the *Subscription Profiles* which are available for à la carte subscription.

This call also sets the `isSubscribed` field of each returned *Package* or *Subscription Profile*, to reflect whether that item is currently subscribed by the *Account* (or *Smart Card* if specified.)

The `startRow` and `endRow` parameters treat the result set as a list which starts with the *Packages* and moves on to the *Subscription Profiles*. For example, if there are 10 *Packages* and 5 *Subscription Profiles*, asking for indexes 1..8 will return a list of 8 *Packages* and an empty list for the *Subscription Profiles*, whilst asking for indexes 8..15 will return a list of 3 *Packages* and a list of 5 *Subscription Profiles*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/getPagedProductListByChannel`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--------|---|
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID which the client intends to subscribe, if appropriate. If this field is null, the objects returned can be subscribed across the whole <i>Account</i> . |
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> , when this API is called through Web Services. If called through HUE, the signed-on <i>Account</i> is always used, so this parameter should be left null. |
| <code>channelNetCarry</code> | String | The <code>carryingNetwork</code> of the <i>Channel</i> . |
| <code>locale</code> | String | The locale to use for the localizable fields of returned objects. |
| <code>startRow</code> | Long | The index of the first record to return. |
| <code>endRow</code> | Long | The index of the last record to return. |

Return value

Type:

Example

```
<getPagedProductListByChannel>
<smartcardId>stringValue</smartcardId>
<accountId>stringValue</accountId>
<channelNetCarry>stringValue</channelNetCarry>
<locale>en_us</locale>
<startRow>61</startRow>
<endRow>36</endRow>
</getPagedProductListByChannel>
```

8.5.7 `aLaCarteService.getProductListByChannel`

Get the à la carte subscription options which will enable the client to view a particular broadcast *Channel*.

This method returns an *AlacarteProduct* model object containing two lists, one containing the *Packages* which are available for à la carte subscription, and one containing the *Subscription Profiles* which are available for à la carte subscription.

This call also sets the `isSubscribed` field of each returned *Package* or *Subscription Profile*, to reflect whether that item is currently subscribed by the *Account* (or *Smart Card* if specified.)

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/getProductListByChannel`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--------|---|
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID which the client intends to subscribe, if appropriate. If this field is null, the objects returned can be subscribed across the whole <i>Account</i> . |
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> , when this API is called through Web Services. If called through HUE, the signed-on <i>Account</i> is always used, so this parameter should be left null. |
| <code>channelNetCarry</code> | String | The <code>carryingNetwork</code> of the <i>Channel</i> . |
| <code>locale</code> | String | The locale to use for the localizable fields of returned objects. |

Return value

Type:

Example

```
<getProductListByChannel>
<smartcardId>stringValue</smartcardId>
<accountId>stringValue</accountId>
<channelNetCarry>stringValue</channelNetCarry>
<locale>en_us</locale>
</getProductListByChannel>
```

8.5.8 aLaCarteService.subscribeToAlaCarteProduct

Subscribes an *Account* and optional *Smart Card* to an *à la carte Package* or *Subscription Profile*.

If the *Package* or *Subscription Profile* has not been enabled for *à la carte* subscription, this method will fail. The operator should use the `workflowSubscriptionService` Web Services API to subscribe *Accounts* to these products.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/subscribeToAlaCarteProduct`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>productUID</code> | Long | The <code>uid</code> of the <i>Package</i> or <i>Subscription Profile</i> to subscribe. |
| <code>productType</code> | String | What type of product the <code>productUID</code> refers to: <code>PKG</code> if it is a <i>Package</i> , or <code>SUB</code> if it is a <i>Subscription Profile</i> . |
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> to subscribe, when this API is called through <i>Web Services</i> . If called through <i>HUE</i> , the signed-on <i>Account</i> is always used, so this parameter should not be supplied. |
| <code>userID</code> | Long | The <code>uid</code> of the <i>User</i> making the subscription, or null to leave this unspecified. This is used to identify the subscribing user in subscription and billing records: it does not affect who is given access to the content. If a <i>User</i> is specified, it must be linked to the <i>Account</i> supplied, or to the <i>Account</i> associated with the <i>Smart Card</i> supplied (in the case where <code>accountId</code> is null). |
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID of the device to give access to the content, or null to leave this unspecified. If specified, the content will be made available to this device only; otherwise, the content will be made available across the whole <i>Account</i> . If this and the <i>Account</i> are both specified, the device must be provisioned to the <i>Account</i> supplied. |

Return value

Type: `List<Long>`

A list containing the `uid` of each *Acquired Content List* Entity created to record this subscription.

Example

```
<subscribeToAlaCarteProduct>
  <productUID>60</productUID>
  <productType>stringValue</productType>
  <accountId>stringValue</accountId>
  <userID>41</userID>
  <smartcardId>stringValue</smartcardId>
</subscribeToAlaCarteProduct>
```

8.5.9 aLaCarteService.unSubscribeAlaCarteProduct

Cancels a current subscription to an *à la carte Package* or *Subscription Profile*.

You can use this API to unsubscribe a single smartcard from the *Policy Group* by passing a *Smart Card* ID, or to unsubscribe the whole *Account* from the *Policy Group* by passing only the *Account*. If you pass both, the smartcard will be unsubscribed, but SDP will also verify that the smartcard belongs to the requested *Account*.

Note

Unsubscribing a *Smart Card* will only affect subscriptions which were made explicitly for that *Smart Card*: if the subscription was made for the whole *Account*, it must be unsubscribed from the whole *Account*

Unsubscribing a whole *Account* will remove the subscription from all associated devices, even if the original subscription was made to the individual *Smart Cards*.

Some subscription products enforce a minimum subscription period. If you unsubscribe a *Policy Group* with a minimum subscription period before that period has finished, SDP will mark the affected subscriptions as pending and unsubscribe them at that point.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/aLaCarteService/unSubscribeAlaCarteProduct`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>productUID</code> | Long | The <code>uid</code> of the <i>Package</i> or <i>Subscription Profile</i> to subscribe. |
| <code>productType</code> | String | What type of product the <code>productUID</code> refers to: <code>PKG</code> if it is a <i>Package</i> , or <code>SUB</code> if it is a <i>Subscription Profile</i> . |
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> to unsubscribe, when this API is called through Web Services. If called through HUE, the signed-on <i>Account</i> is always used, so this parameter should not be supplied. |
| <code>userID</code> | Long | This field should not be supplied. |
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID to unsubscribe, or null to unsubscribe all devices provisioned to the <i>Account</i> . |

Return value

Type:

A list containing the `uid` of each *Acquired Content List* Entity which was modified by the unsubscribe action.

Example

```
<unSubscribeAlaCarteProduct>
<productUID>21</productUID>
<productType>stringValue</productType>
```

```
<accountId>stringValue</accountId>  
<userID>12</userID>  
<smartcardId>stringValue</smartcardId>  
</unsubscribeAlaCarteProduct>
```

9 assetService

This is a primitive *Service* which provides the *Asset* Entity.

The *Asset Group* Service manages the relationships between *Asset Groups* and *Assets*.

Relationships with *Catalogues* and *COD Categories* are handled in their respective *Services*.

9.1 Asset

An *Asset* represents a film, episode, or another media object, that a subscriber can view.

Figure 4 Asset and associated Entities.

Asset Entities are used for both editorial and technical information. Editorial *Assets* represent the "idea" of a content item, such as "Quantum Of Solace", whilst technical *Assets* represent the realization of that item in the form of a data stream or object. The same *Entity* is used for both types, because they share many metadata fields; they can be distinguished using the **assetType** field, which is **VT** for technical *Assets* and **VE** for editorial *Assets*.

The **Status** field determines whether client applications should show this *Asset* to users. It is either **P** for "published", or **U** for "unpublished". Clients should not normally display unpublished *Assets*.

Trailers and advertisements are represented as *Assets* where these are in use in the deployment. They can be distinguished using the **assetPlayType** field:

- ▶ If the **Play Type** is **E** ("entertainment"), the *Asset* is a regular piece of content such as a film or TV program, which may be associated with other *Assets* representing trailers. In deployments using SDP's targeted advertising features, an entertainment *Asset* may also be associated with *Schedules* representing the choice of advertising to be supplied with the *Asset*.
- ▶ If it is **A** ("advertisement"), the *Asset* is an advertisement.
- ▶ If it is **T** ("trailer"), the *Asset* is a trailer for one or more entertainment *Assets*.

Caution!

Technical and editorial *Assets* are created automatically by the CMS ingest process. On-demand content and pricing should be defined in the CMS.

The CMS exports *Content* objects as *Assets*. Every *Content* object yields one editorial *Asset* and at least one technical *Asset*. *Content* objects with multiple content versions (for example, SD and HD versions of the same film) yield one editorial *Asset* and multiple technical *Assets*, one for each content version. These *Assets* are automatically linked by the ingest process.

9.1.1 Asset *Updated*

| Name | Type | Description |
|--------|--------|---|
| actors | String | The names of actors appearing in this Asset , separated by semicolons. |

| Name | Type | Description | | | | |
|-------------------------|---|---|-------|-------------|-------------------------|---|
| | | The CMS ingest places the Content object's actors EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty. | | | | |
| aspectRatio | String | <p>The aspect ratio of this Asset.</p> <p>For example:</p> <ul style="list-style-type: none"> ▶ 16:9 ▶ 4:3 ▶ Widescreen <p>The CMS ingest places the Content object's Aspect EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> | | | | |
| assetList | String | <p>This field is not managed by SDP; operators may decide how to use it.</p> <p>This field is not ingested from the CMS.</p> | | | | |
| assetOrder | Integer | <p>The episode number of the Asset in its series.</p> <p>Tip This field does not have to be a numeric value; episode numbers in unusual formats, such as "7-4", "S7E4", or "3F02", are supported.</p> <p>The CMS ingest places the Content object's episode Number in this field. When an Asset is created from a Promotion object, the field is left empty.</p> | | | | |
| assetPlayType | String | <p>The role performed by this Asset. One of:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>E(Entertainment)</td> <td> <p>This Asset is a main content item, which may be added to Catalogues and purchased.</p> <p>Entertainment Assets can have attached advertisement Assets.</p> </td> </tr> </tbody> </table> | Value | Description | E(Entertainment) | <p>This Asset is a main content item, which may be added to Catalogues and purchased.</p> <p>Entertainment Assets can have attached advertisement Assets.</p> |
| Value | Description | | | | | |
| E(Entertainment) | <p>This Asset is a main content item, which may be added to Catalogues and purchased.</p> <p>Entertainment Assets can have attached advertisement Assets.</p> | | | | | |

| Name | Type | Description | | | | | | |
|-----------------------------|--|--|-------|-------------|-----------------------------|--|-----------------------------|--|
| | | <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>This Asset is an advertisement, (Advertisement) which can be attached to entertainment Assets.</td> </tr> <tr> <td>T (Trailer)</td> <td>This Asset is a trailer for another Asset, or for a content item which is not yet known to SDP. Trailer Assets cannot have attached advertisement Assets.</td> </tr> </tbody> </table> | Value | Description | A | This Asset is an advertisement, (Advertisement) which can be attached to entertainment Assets . | T (Trailer) | This Asset is a trailer for another Asset , or for a content item which is not yet known to SDP. Trailer Assets cannot have attached advertisement Assets . |
| Value | Description | | | | | | | |
| A | This Asset is an advertisement, (Advertisement) which can be attached to entertainment Assets . | | | | | | | |
| T (Trailer) | This Asset is a trailer for another Asset , or for a content item which is not yet known to SDP. Trailer Assets cannot have attached advertisement Assets . | | | | | | | |
| | | <p>The CMS ingest process creates Assets of play type E for Content objects, and Assets of play type T for Promotion objects.</p> | | | | | | |
| assetType | Search criterion. String | <p>The type of asset this is. One of:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>vT (Video-Technical)</td> <td>This is a technical VOD Asset representing a single video stream.</td> </tr> <tr> <td>vE (Video-Editorial)</td> <td>This is an editorial VOD Asset representing a content item, which may be realized in several different forms (which will also be present in SDP as Assets of type vT).</td> </tr> </tbody> </table> <p>The CMS ingest process creates one Asset of type vE for each editorial Content, and one Asset of type vT for each content version. Therefore, there will always be at least two Assets ingested, even if a content item has only one associated media stream.</p> | Value | Description | vT (Video-Technical) | This is a technical VOD Asset representing a single video stream. | vE (Video-Editorial) | This is an editorial VOD Asset representing a content item, which may be realized in several different forms (which will also be present in SDP as Assets of type vT). |
| Value | Description | | | | | | | |
| vT (Video-Technical) | This is a technical VOD Asset representing a single video stream. | | | | | | | |
| vE (Video-Editorial) | This is an editorial VOD Asset representing a content item, which may be realized in several different forms (which will also be present in SDP as Assets of type vT). | | | | | | | |
| audioMode | String | <p>A string defined by the operator to represent the audio properties of this Asset.</p> <p>For example:</p> <ul style="list-style-type: none"> ▶ Mono ▶ Dual Mono | | | | | | |

| Name | Type | Description |
|-----------------------|---|--|
| | | <p>▶ 5.1</p> <p>The CMS ingest places the Content object's AudioMode EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> |
| audioPid | List<AssetAudioPid> | <p>A list of AssetAudioPid objects representing the choice of audio streams available for the Asset.</p> <p>The CMS ingest creates AssetAudioPid objects as necessary based on the information in the Audio EPG key of the Content object. When an Asset is created from a Promotion object, the field is left empty.</p> |
| audioSubtracks | Deprecated. String | <p>A description of any additional audio tracks or audio-replacement tracks provided by the Asset.</p> <p>Note This field is no longer provided by the CMS ingest. Audio variants are used instead.</p> |
| bestSalesIndex | Long | <p>The best sales index of this Asset.</p> <p>This field can be used to order Assets by popularity or by other criteria, allowing clients to display a top 40 style interface.</p> <p>A value of 1 denotes the most popular Asset.</p> <p>A value of -1 means that the Asset should not appear in any list of the most popular Assets.</p> <p>The CMS ingest places the Content object's Best sales Index EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> |
| bitRate | Long | |
| casDescriptor | Deprecated. String | |
| casId | Deprecated. String | |
| changeLog | Deprecated. Not visible to HUE. List<String> | <p>A log of the changes to this Entity history.</p> <p>This field is managed by SDP and is not ingested from the CMS.</p> |

| Name | Type | Description |
|------------------------------|------------------------------|---|
| <code>contentRating</code> | <code>AggregateRating</code> | |
| <code>contentType</code> | <code>String</code> | <p>This field is provided for the use of operators which need to distinguish different types of Asset. When it is in use, the client is expected to know how to use it; any value stored here is opaque to SDP.</p> <p>The CMS ingest sets this field from the Content object's contentType EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>copyProtections</code> | <code>String</code> | <p>A semicolon-separated list of opaque identifiers representing usage rules for preventing unauthorized content copying. The format of this field is understood by the CAK inside the set-top box.</p> <p>SDP includes this information when requesting a license from a NAGRA Media ACCESS PRM license server. It is then incorporated in encrypted form in the returned license string. The CAK inside the set-top box is responsible for decoding the license and applying the restriction.</p> <p>The CMS ingest sets this field from the Content object's CopyProtections EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>copyright</code> | <code>String</code> | <p>The copyright status and/or owner of this Asset.</p> <p>This field is not managed by SDP; operators may decide how to use it.</p> <p>The CMS ingest places the Content object's Copyright EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>countries</code> | <code>String</code> | <p>A list of 2-letter ISO codes of countries for the Asset.</p> <p>For example: <code>us, fr, cn</code></p> <p>The CMS ingest places the Content object's Countries EPG attribute in this field.</p> |
| <code>creationDate</code> | <code>Date</code> | <p>The date and time this Entity was created in SDP.</p> <p>This field is managed by SDP and is not ingested from the CMS.</p> |

| Name | Type | Description |
|-------------------------------------|--------|---|
| <code>cuBroadcastChannelName</code> | String | <p>In deployments implementing catch-up services through the VOD catalogue, the name of the channel this Asset was broadcast on.</p> <p>The CMS ingest sets this field from the Content object's <code>ServiceLongName</code> EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> <p>Note</p> <div style="border-left: 2px solid black; padding-left: 10px;"> <p>There is no enforced link in SDP between this field and any field of the Channel object; it is not mandatory for the referenced channel to be known to SDP. Deployments can choose which way of identifying a channel will be easiest for clients to support.</p> </div> |
| <code>cuBroadcastEndDate</code> | Date | <p>In deployments implementing catch-up services through the VOD catalogue, the date and time this Asset's broadcast finished.</p> <p>The CMS ingest sets this field from the Content object's <code>ProgrammeEndDate</code> EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>cuBroadcastStartDate</code> | Date | <p>In deployments implementing catch-up services through the VOD catalogue, the date and time this Asset was originally broadcast.</p> <p>The CMS ingest sets this field from the Content object's <code>ProgrammeStartDate</code> EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>definition</code> | String | <p>The picture definition of this Asset.</p> <p>The CMS ingest places the Content object's <code>Definition</code> EPG attribute in this field. If no <code>Definition</code> is specified, a default value of <code>SD</code> is used.</p> <p>In deployments using SRM, this field must have a value, because SRM uses it to work out how much bandwidth to allocate for the stream. Allowed values are:</p> <ul style="list-style-type: none"> ▶ HD ▶ SD ▶ 3D |
| <code>description</code> | String | A short description of this Asset . |

| Name | Type | Description |
|-------------------------|---|--|
| | | The CMS ingest places the Content object's Description EPG attribute in this field. |
| directors | String | <p>The names of the directors of this Asset, separated by semicolons.</p> <p>The CMS ingest places the Content object's Directors EPG attribute in this field.</p> |
| downloadToGoAuth | Integer | Flag representing if the asset is allowed to download or not. |
| drmDescriptor | String | <p>A descriptor used by the DRM system which protects this Asset.</p> <p>The CMS ingest sets this field to the drmDescriptor of the Media object which represents the stream, if it has one.</p> <p>See BZ#14136</p> |
| drmId | String | <p>The identity of this Asset for the purposes of the DRM system which protects it.</p> <p>The CMS ingest sets this field to the drmId of the Media object which represents the stream, if it has one.</p> <p>See BZ#14136</p> |
| drmInstanceName | Search criterion. String | <p>An identifier for the DRM system which protects this Asset.</p> <p>The CMS ingest sets this field to the drmInstanceName of the Media object which represents the stream, if it has one.</p> <p>The client can use this name to determine how to process licenses returned by SDP. To do this, DRM instance names need to be defined by the operator beforehand and synchronized between the application, CMS and SMS.</p> <p>See BZ#14136</p> |
| dualMono | Deprecated. Boolean | <p>Does this Asset have dual-mono sound?</p> <p>The CMS ingest sets this field to true if the AudioMode EPG attribute of the Content is set to Dual Mono.</p> |
| duration | Deprecated. Long | The duration of this Asset , in minutes. |

| Name | Type | Description |
|-----------------------|---------|--|
| | | The CMS ingest calculates this field from the Content object's duration (which is expressed in seconds.) |
| durationMillis | Long | The duration of this Asset , in milliseconds. The CMS ingest calculates this field from the Content object's Duration (which is expressed in seconds.) |
| encoding | String | The encoding or format of the video stream for the Asset . The CMS ingest sets this field to the format of the Media object which represents the stream. The value of this field also determines which Media object represents the stream for this Asset , if more than one is supplied. By default, SDP will choose the one with the format which comes first in this list: <ul style="list-style-type: none"> ▶ AV_HarmonicOSPlaylistName ▶ AV_PlaylistName ▶ AV_EncryptedTS ▶ AV_ClearTS |
| episode | String | The episode number or designation for this Asset within its series, if it has one. This is a localized string field which clients should use when they need to display the episode number. Episodes of a series also have an order within the Asset Group which represents the series. To display episodes in the correct order, you should use the assetOrderInAsset Group sort criterion, not this field. The CMS ingest places the Content object's Episode EPG attribute in this field. (The order of the episode within its Asset Group is determined by its episodeNumber attribute, which is numeric.) |
| exportID | Integer | The identifier to use when exporting this Entity to third-party systems. This field is managed by SDP and is not ingested from the CMS. |
| fileName | String | The filename of this Asset . The CMS ingest sets this field to the filename of the Media object which represents the stream. |

| Name | Type | Description |
|------------------------|------------------------|---|
| | | If the encoding is AV_HarmonicOSPlaylistName , this field contains the full URL to the Asset . |
| fileSize | Long | The size of the data file for this Asset . The CMS ingest sets this field to the fileSize of the Media object which represents the stream. |
| genre | String | A list of genres to which this Asset belongs, separated by semicolons. The CMS ingest sets this field from the Categories EPG attribute of the Content object. It also adds the Asset to any COD Categories whose originKey match a value in the list, and whose originID is 1. |
| hearingImpaired | Deprecated. String | A representation of whether the Asset provides accessibility features for hearing-impaired people. This field is not managed by SDP; operators may decide how to use it. The CMS ingest sets this field from the Content object's HearingImpaired EPG attribute. The CMS ingest does not set this field. |
| language | String | The original language of this Asset . The CMS ingest places the Content object's Language EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty. |
| locale | String | When Entities are returned as model objects, the locale for which localizable fields of the Entity have been returned. |
| longDescription | String | A longer description of this Asset . The CMS ingest places the Content object's synopsis EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty. |
| mainContentUID | Search criterion. Long | If this is a technical Asset the UID of the associated editorial Asset . If this field is empty, this is an editorial Asset . |

| Name | Type | Description |
|-----------------------------|--|---|
| | | The CMS ingest creates both technical and editorial Assets as appropriate from the ingested Content , Promotion and Media objects. |
| <code>modifiedDate</code> | Date | The date and time this Entity was last modified in the database. This field is managed by SDP and is not ingested from the CMS. |
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this Entity were last modified in the database. This field is managed by SDP and is not ingested from the CMS. |
| <code>nPVR</code> | Deprecated. Read only. Boolean | Does this Asset represent an NPVR recording of a broadcast programme? This field is no longer required and should not be used. The CMS ingest does not set this field. |
| <code>name</code> | Search criterion. Quick Search criterion. String | The full name of this Asset . The CMS ingest places the Content object's Title EPG attribute in this field, if it has one. If no Title EPG attribute is given for the default locale, or the Asset is being created from a Promotion object, the title attribute is used instead. |
| <code>originID</code> | Integer | An identifier for the system which owns this Entity . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. This field is set to 1 for all Entities ingested from NAGRA Media CMS. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this Entity uses to identify it. For all Entities ingested from NAGRA Media CMS, this field is set to the <code>id</code> of the object. |

| Name | Type | Description |
|-----------------------------------|--|--|
| <code>ownerID</code> | <code>String</code> | <p>A code representing the owner (content provider) for this Asset.</p> <p>The CMS ingest places the Content object's <code>companyId</code> in this field. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>pVPurchase</code> | Read only. <code>boolean</code> | This field is always <code>false</code> for Assets . |
| <code>parentAclUID</code> | <code>Long</code> | <p>This field is used internally by SDP during the purchase process. Clients should not attempt to use it.</p> <p>The CMS ingest does not set this field.</p> |
| <code>parentPolicyGroupUID</code> | <code>Long</code> | <p>This field is used internally by SDP during the purchase process. Clients should not attempt to use it.</p> <p>The CMS ingest does not set this field.</p> |
| <code>parentalRating</code> | <code>Long</code> | <p>The Rating appropriate to this Asset.</p> <p>The CMS ingest uses the Content object's <code>Rating</code> EPG attribute to determine a suitable value for this field. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>pmtString</code> | <code>String</code> | <p>An identifier for specialized uses in certain deployments, set by the CMS. This field is not managed by SDP; operators may decide how to use it.</p> <p>The CMS ingest sets this field from the content of the Media object of type <code>Data_PMT</code> associated with the ingested Content object, if there is one.</p> |
| <code>policyGroupUID</code> | <code>Long</code> | <p>This field is used internally by SDP during the purchase process. Clients should not attempt to use it.</p> <p>The CMS ingest does not set this field.</p> |
| <code>privateMetadata</code> | <code>String</code> | <p>This field is provided for operators to use for their own purposes. SDP passes it to the client unchanged. Its format (string, JSON object...) and meaning should be decided by the operator.</p> <p>The CMS ingest places the Content object's <code>PrivateMetadata</code> EPG attribute in this field.</p> |

| Name | Type | Description |
|-------------------------|---------|---|
| <code>producers</code> | String | <p>The names of the producers of this Asset, separated by semicolons.</p> <p>The CMS ingest places the Content object's Producers EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>profileUID</code> | Long | <p>The <code>uid</code> of the Asset Profile for this Asset.</p> <p>The CMS ingest creates Asset Profiles as necessary using the Profiles defined in the ingest file.</p> |
| <code>programId</code> | String | <p>An identifier representing the Asset for editorial purposes. This field is not managed by SDP; operators may decide how to use it.</p> <ul style="list-style-type: none"> ▶ In deployments which manage series content on the client side (rather than using Asset Groups to represent series) this field could contain the index of an episode in its series. ▶ In deployments which want to use the on-demand catalogue for catch-up content, this field could contain an ID which links the Asset with the corresponding broadcast Program. <p>The CMS ingest sets this field from the Content object's ProgramId EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> |
| <code>promoUrl</code> | String | <p>The URL of a promotional image for this Asset.</p> <p>Tip Either editorial or technical Assets may have promotional images.</p> <p>Caution! This field does not refer to trailers (which are called Promotions in CMS) but to static images. In SDP, trailers are represented by Asset Entities whose <code>assetPlayType</code> is T.</p> <p>The CMS ingest places the Content object's Promo Images EPG attribute in this field. When an Asset is created from a Promotion object (representing a trailer), the field is left empty.</p> |
| <code>recordable</code> | Boolean | <p>May this Asset be recorded by users?</p> |

| Name | Type | Description |
|--------------------------|---------------------------|---|
| | | The CMS ingest sets this field from the Content object's IsRecordable EPG attribute. When an Asset is created from a Promotion object, the field is left empty. |
| scoring | Integer | <p>A score or popularity value for this Asset.</p> <p>The CMS ingest sets this field from the Content object's scoring EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> <p>Note This field is not updated by SDP. To use it, a deployment must use a separate process to collect scoring information and inject it into the CMS.</p> |
| seriesId | Deprecated. String | <p>An identifier representing the series the Asset belongs to, in a form defined by the CMS. This field is not managed by SDP; operators may decide how to use it.</p> <p>The CMS ingest sets this field from the Content object's seriesId EPG attribute.</p> <p>The CMS ingest does not set this field.</p> <p>Note This field is only used in specific circumstances where the deployment does not use Asset Groups to represent series, but instead manages series content on the client side.</p> |
| serviceProviderID | Long | <p>The SDP internal UID for the Service Provider which created this Entity.</p> <p>For Entities ingested from NAGRA Media CMS, this is the uid of the Service Provider whose originKey matches the id found on the ingest file's root. CMSvod Data element.</p> |
| shortTitle | String | <p>A shorter name for this Asset.</p> <p>The CMS ingest places the Content object's shortTitle EPG attribute in this field.</p> |
| status | String | The publication status of this Asset . One of: |

| Name | Type | Description | | | | | | | | |
|-----------------------------|---|---|-------|-------------|----------------------|---|------------------------|---|--------------------|--|
| | | <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>A (Published)</td> <td>This Asset is available for viewing now.</td> </tr> <tr> <td>I (Unpublished)</td> <td>This Asset has been added to the database, but should not be made available to subscribers. check</td> </tr> <tr> <td>R (Deleted)</td> <td>This Asset is no longer available. check</td> </tr> </tbody> </table> | Value | Description | A (Published) | This Asset is available for viewing now. | I (Unpublished) | This Asset has been added to the database, but should not be made available to subscribers. check | R (Deleted) | This Asset is no longer available. check |
| Value | Description | | | | | | | | | |
| A (Published) | This Asset is available for viewing now. | | | | | | | | | |
| I (Unpublished) | This Asset has been added to the database, but should not be made available to subscribers. check | | | | | | | | | |
| R (Deleted) | This Asset is no longer available. check | | | | | | | | | |
| | | <p>The CMS ingest assigns the fixed status A (Published) to every Asset it creates.</p> <p>Note This field does not appear in the Create Asset tab. If you create an Asset in SDP Manager, it is given the status Unpublished. When you want to publish it, you must go into the Edit Asset tab and change this field to Published.</p> | | | | | | | | |
| studio | String | <p>The name of the studio which produced this Asset: the content owner.</p> <p>The CMS ingest places the Content object's studio EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> | | | | | | | | |
| subTitles | Boolean | <p>Does this Asset include subtitles (in any language)?</p> <p>The CMS ingest sets this field to true if the subtitles EPG attribute contains a value, false if it is null. When an Asset is created from a Promotion object, the field is always set to false.</p> | | | | | | | | |
| subscribableItemType | String | <p>This field is always AST for Assets.</p> | | | | | | | | |
| subscribed | Read only. boolean | <p>Is the user already subscribed to this Asset?</p> | | | | | | | | |

| Name | Type | Description |
|--------------------------------|--|--|
| | | <p>This field is returned as part of the Asset model object when clients query on-demand offerings. It is not part of the Asset Entity itself.</p> <p>Note Attempting to alter this field will have no effect. Clients should always subscribe to on-demand content through Policy Groups.</p> |
| <code>subtitleLanguages</code> | Deprecated. <code>String</code> | <p>A semicolon-separated list of subtitle specifiers representing the subtitle languages and types available for the Asset.</p> <p>A subtitle specifier consists of a 3-character ISO 639-3 language code and one of the type designations Standard and HearingImpaired, separated by a hyphen -.</p> <p>For example: <code>eng-Standard, fra-Hearing Impaired, zho-Standard</code></p> |
| <code>totalDownloads</code> | <code>Integer</code> | <p>The number of times this Asset has been downloaded.</p> <p>The CMS ingest sets this field from the Content object's TotalDownloads EPG attribute. When an Asset is created from a Promotion object, the field is left empty.</p> <p>Note This field is not updated by SDP. To use it, a deployment must use a separate process to monitor the number of downloads and inject this information into the CMS.</p> |
| <code>totalRequests</code> | <code>Long</code> | |
| <code>trickPlay</code> | <code>TrickPlay</code> | This field is not persisted and is always empty. |
| <code>uID</code> | <code>Long</code> | <p>The SDP internal UID for this Entity.</p> <p>This field is managed by SDP and is not ingested from the CMS.</p> |
| <code>url</code> | <code>String</code> | <p>The URI of this Asset on the network.</p> <p>The CMS ingest sets this field from the <code>filename</code> of the Media object. SDP can be configured at deployment</p> |

| Name | Type | Description |
|----------------------|---------------------------------|---|
| | | <p>time to prefix this with a string specified by the operator to produce a full URI; contact Nagra for more details.</p> <p>This field should be ignored if the encoding is AV_HarmonicOSPlaylistName, because the full URL is given in the fileName field.</p> |
| viewingNumber | Integer | <p>How many times a user may watch this Asset, by default, once they have purchased it.</p> <p>This value may be overridden by the viewingNumber of a Policy Group which provides this Asset.</p> <p>A value of 0 means there is no limitation on viewings.</p> <p>The CMS ingest sets this field from the Content object's viewingNumber EPG attribute. When an Asset is created from a Promotion object, the field is set to 0.</p> <p>SDP includes this information when requesting a license from a NAGRA Media ACCESS PRM license server. It is then incorporated in encrypted form in the returned license string. The CAK inside the set-top box is responsible for decoding the license and applying the restriction.</p> |
| year | Search criterion. String | <p>The year of production of this Asset, or another notable date defined by the operator.</p> <p>The CMS ingest places the Content object's production year EPG attribute in this field. When an Asset is created from a Promotion object, the field is left empty.</p> |

9.2 tv.quative.service.asset.model.AssetAudioPid

| Name | Type | Description |
|--------------------|---------------|--|
| assetUID | Long | The uID of the Asset . |
| audioType | String | The type of this PID. One of AUDIO or SUBTITLES . |
| businessKey | String | A unique business key for this PID, made from the language and audioType joined with a colon : |

| Name | Type | Description |
|---------------------------|--|--|
| <code>defaultPid</code> | <code>Boolean</code> | Should this PID be active by default when the stream is played? |
| <code>language</code> | <code>String</code> | The language this PID is presented in. |
| <code>modifiedDate</code> | <code>Date</code> | The date of the last changes to this object. |
| <code>oldAudioType</code> | <i>Deprecated.</i> <code>String</code> | This field is for internal use and should not be changed. |
| <code>oldLanguage</code> | <i>Deprecated.</i> <code>String</code> | This field is for internal use and should not be changed. |
| <code>pidId</code> | <code>String</code> | The identifier of this PID within the transport stream. |
| <code>primaryUID</code> | <code>Long</code> | This field is always the same as the <code>assetUID</code> . It is used internally to SDP. |

9.3 assetService

AssetService Implementation. Manages the business logic of all the operations with Assets.Object which implements the business logic of all the methods of the interface. The services are made visible to Hue or to WebServices using annotations in the Service Implementation class. Ex: For the ServiceClass:

= "assetService", hue = true, ws = true) For the Methods: `@QubeService(name`

`ws=true)` The methods used to create, delete or update items of the COD System, only could be accesible to Web Services (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel would be accessible on both the WebServices and the Hue interfaces. `@QubeMethod(hue=false,`

Zafar Ali `@author`

1.0 `@version`

Ashish Gupta - rewritten/refactored for MSP `@author`

1.3 `@version`

Aled Rhys Jones `@author`

Naeem Bilal `@author`

9.3.1 assetService.addAudioPid

A Service Method to add audio pids against the asset..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|---|---------------------------------|
| <code>assetAudioPid</code> | <code>tv.quative.service.asset.model.AssetAudioPid</code> | Asset audio pid. ¹⁷¹ |

Return value

Type:

number of rows affected

9.3.2 `assetService.addAudioPid`

A Service Method to add audio pids against the asset..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/addAudioPid`

Parameters

| Name | Type | Description |
|--------------------------|--------|-------------|
| <code>assetUID</code> | Long | Asset UID |
| <code>pidUID</code> | String | PId id |
| <code>pidLanguage</code> | String | Language |
| <code>paudioType</code> | String | Audio type |

Return value

Type:

number of rows affected

9.3.3 `assetService.addMultiLanguage`

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

9.3.4 assetService.create

Create (persist) a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------------|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

9.3.5 assetService.delete

A Service method to expose the deletion of Asset from the system.

Calls the corresponding overloaded method with Asset model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/delete`

Parameters

| Name | Type | Description |
|------------------|------|-------------------|
| <code>uid</code> | Long | Integer Asset uid |

Return value

Type:

Integer number of records deleted

9.3.6 assetService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows deleted.

9.3.7 assetService.getAssetAudioPids

Gets the list of asset audio pids.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| <code>assetUID</code> | Long | Asset UID |

Return value

Type: `List<AssetAudioPid>`

List of Asset Audio pids

Example

```
<getAssetAudioPids>
  <assetUID>92</assetUID>
</getAssetAudioPids>
```

9.3.8 assetService.getAssetListByTrailerUID

Gets the list of assets associated with the specific trailer UID..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getAssetListByTrailerUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|-------------|
| <code>trailerUid</code> | Long | |
| <code>locale</code> | String | Locale code |

Return value

Type: `List<Asset>`

List of Asset object

Example

```
<getAssetListByTrailerUID>
<trailerUid>88</trailerUid>
<locale>en_us</locale>
</getAssetListByTrailerUID>
```

9.3.9 assetService.getAssetsByAssetGroupUID

Gets the list of assets belonging to the given asset group..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getAssetsByAssetGroupUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|--------------------------------------|
| <code>assetGroupUID</code> | Long | Identifies the catalogue to look for |
| <code>locale</code> | String | Locale |

Return value

Type: `List<Asset>`

List of assets belonging to the given asset group

Example

```
<getAssetsByAssetGroupUID>
  <assetGroupUID>14</assetGroupUID>
  <locale>en_us</locale>
</getAssetsByAssetGroupUID>
```

9.3.10 assetService.getAssetsByCODCategoryUID

Gets the list of assets belonging to the given cod category UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|--------|------------------|
| codcategoryUID | Long | COD cateogry UID |
| locale | String | Locale code |

Return value

Type: `List<Asset>`

List of Asset object

Example

```
<getAssetsByCODCategoryUID>
  <codcategoryUID>9</codcategoryUID>
  <locale>en_us</locale>
</getAssetsByCODCategoryUID>
```

9.3.11 assetService.getAssetsByCatalogueUID

Gets the list of assets belonging to the given catalogue..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--------|--------------------------------------|
| catalogueUID | Long | Identifies the catalogue to look for |
| locale | String | Locale |

Return value

Type: `List<Asset>`

List of assets belonging to the given target area

Example

```
<getAssetsByCatalogueUID>
<catalogueUID>49</catalogueUID>
<locale>en_us</locale>
</getAssetsByCatalogueUID>
```

9.3.12 assetService.getAssetsByScheduleUID

Gets the list of assets belonging to the given schedule UID..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getAssetsByScheduleUID`

Parameters

| Name | Type | Description |
|-------------|--------|--------------|
| scheduleUID | Long | Schedule UID |
| locale | String | Locale |

Return value

Type: `List<Asset>`

List of Asset object

9.3.13 assetService.getAvailableAssetListCount

A Service method to get the count of total number of assets for a particular locale.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getAvailableAssetListCount`

Parameters

| Name | Type | Description |
|-----------|--------|-------------|
| locale | String | Locale code |
| assetType | String | as String |

Return value

Type:

Long-Count of active asset objects

9.3.14 assetService.getAvailableList

A Service method to get the all Asset details for a particular locale, which are Active(published)..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getAvailableList>

Parameters

| Name | Type | Description |
|-----------|---------|---|
| locale | String | Locale code |
| startRow | Integer | Row number of the first entry to return |
| endRow | Integer | Row number of the last entry to return |
| assetType | String | as String |

Return value

Type: `List<Asset>`

List of active asset objects

9.3.15 assetService.getAvailablePagedAssetList

A Service method to get the all Asset details for a particular locale, which are Active(published)..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getAvailablePagedAssetList>

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>locale</code> | String | Locale code |
| <code>startRowNum</code> | Long | Row number of the first entry to return |
| <code>endRowNum</code> | Long | Row number of the last entry to return |
| <code>assetType</code> | String | as String |

Return value

Type: `List<Asset>`

List of active asset objects

9.3.16 `assetService.getByOrigin`

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
```

```
<originKey>stringValue</originKey>
<originID>88</originID>
<locale>en_us</locale>
</getByOrigin>
```

9.3.17 assetService.getByOriginID

Retrieve all the models related to the external origin id..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---------|--|
| originID | Integer | Integer unique origin id for customer representation |
| locale | String | String |

Return value

Type: List<Asset>

AssetList objects for this external origin uid and origin key.

Example

```
<getByOriginID>
<originID>64</originID>
<locale>en_us</locale>
</getByOriginID>
```

9.3.18 assetService.getByUID

Gets the Asset with the specified uid.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getByUID>

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-----------------------|
| uid | Long | The uid of the Asset. |

| Name | Type | Description |
|---------------------|--------|--|
| <code>locale</code> | String | The locale to use for the localizable fields of the Asset. |

Return value

Type:

The matching Asset, or null if none is found.

Example

```
<getByUID>
  <uid>98</uid>
  <locale>en_us</locale>
</getByUID>
```

9.3.19 assetService.getList

A Service method to get the Asset details for a particular locale..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getList`

Parameters

| Name | Type | Description |
|---------------------|--------|-------------|
| <code>locale</code> | String | String |

Return value

Type: `List<Asset>`

List of Asset object

9.3.20 assetService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: `List<M>`

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>74</uid>
</getMultiLanguageList>
```

9.3.21 assetService.getPagedAssetsByAssetGroupUID

Gets the list of assets belonging to the given asset group..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getPagedAssetsByAssetGroupUID`

Parameters

| Name | Type | Description |
|---------------|--------|---|
| assetGroupUID | Long | Identifies the catalogue to look for |
| locale | String | Locale code |
| startRowNum | Long | Row number of the first entry to return |
| endRowNum | Long | Row number of the last entry to return |

Return value

Type: `List<Asset>`

List of assets belonging to the given asset group

9.3.22 assetService.getPagedAssetsByOriginID

Retrieve all the models related to the external origin id..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getPagedAssetsByOriginID>

Parameters

| Name | Type | Description |
|-------------|---------|--|
| originID | Integer | Integer unique origin id for customer representation |
| locale | String | Locale code |
| startRowNum | Long | Row number of the first entry to return |
| endRowNum | Long | Row number of the last entry to return |

Return value

Type: `List<Asset>`

AssetList objects for this external origin uid and origin key.

9.3.23 assetService.getPagedAssetsByScheduleUID

Gets the list of assets belonging to the given schedule UID..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getPagedAssetsByScheduleUID>

Parameters

| Name | Type | Description |
|-------------|--------|---|
| scheduleUID | Long | Long |
| locale | String | Locale code |
| startRowNum | Long | Row number of the first entry to return |
| endRowNum | Long | Row number of the last entry to return |

Return value

Type: `List<Asset>`

List of Asset object

9.3.24 assetService.getPagedList

A Service method to get the Asset details for a particular locale..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getPagedList`

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>locale</code> | String | Locale code |
| <code>startRowNum</code> | Long | Row number of the first entry to return |
| <code>endRowNum</code> | Long | Row number of the last entry to return |

Return value

Type: `List<Asset>`

List of Asset object

9.3.25 assetService.getTrailersForAssetUID

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetService/getTrailersForAssetUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|-------------|
| <code>primaryUid</code> | Long | |

Return value

Type: `List<Asset>`

Example

```
<getTrailersForAssetUID>
  <primaryUid>96</primaryUid>
</getTrailersForAssetUID>
```

9.3.26 assetService.increaseTotalRequests

increase total requests..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|------|-------------|
| assetUID | Long | Asset UID |

Return value

Type:

Example

```
<increaseTotalRequests>
  <assetUID>36</assetUID>
</increaseTotalRequests>
```

9.3.27 assetService.modifyAudioPid

A Service Method to modify the existing audio pids against the asset..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--|---------------------------------|
| assetAudioPid | tv.quative.service.asset.model.AssetAudioPid | Asset audio pid. ¹⁷¹ |

Return value

Type:

number of rows affected

9.3.28 assetService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

9.3.29 assetService.removeAudioPid

A Service Method to remove the existing audio pids against the asset..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--|---------------------------------|
| assetAudioPid | tv.quative.service.asset.model.AssetAudioPid | Asset audio pid. ¹⁷¹ |

Return value

Type:

number of rows affected

9.3.30 assetService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

9.3.31 assetService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

9.3.32 assetService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

10 assetGroupService

This is a primitive *Service* which provides the *Asset Group* Entity.

It manages relationships between *Asset Groups* and their constituent *Assets*.

Relationships with *Catalogues* and *COD Categories* are handled in their respective *Services*.

10.1 Asset Group

An *Asset Group* represents a natural grouping of editorial *Assets*.

Figure 5 AssetGroup and associated Entities.

A TV series, containing several episodes, would be represented in SDP as an *Asset Group*.

Asset Groups exist for information purposes. They do not affect how content is offered for purchase.

It's possible to schedule a technical *Asset* into a *Catalogue* in conjunction with an *Asset Group*. If an *Asset Group* is specified, this tells SDP that the *Asset* is in the *Catalogue* because its editorial *Asset* is a member of the *Asset Group*, rather than for some other reason (because it is in this week's "Top Ten"; because there is a special offer available for it. ..)

Asset Groups are created automatically by the CMS ingest process. On-demand content and pricing should be defined in the CMS.

The ingest assigns *Assets* to *Asset Groups* on the basis of the **seriesRef** metadata key of the **Content** object.

Caution!

seriesRef is distinct from **seriesId**. SDP will ingest the **seriesId** EPG key, but does not use it to assign *Assets* to *Asset Groups*.

10.1.1 AssetGroup

| Name | Type | Description |
|--------------------|---------------|--|
| actors | String | The names of actors appearing in this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Actors EPG attribute. |
| aspectRatio | String | The aspect ratio of <i>Assets</i> in this <i>Asset Group</i> . <i>For example:</i> <ul style="list-style-type: none"> ▶ 16:9 ▶ 4:3 ▶ Widescreen |

| Name | Type | Description |
|--------------------|--|--|
| | | The CMS ingest sets this field from the <i>Series</i> object's Aspect EPG attribute. |
| assets | List<Asset> | A list of the <i>Assets</i> in this <i>Asset Group</i> . |
| audioMode | String | <p>The audio type for <i>Assets</i> in this <i>Asset Group</i>.</p> <p><i>For example:</i></p> <ul style="list-style-type: none"> ▶ Mono ▶ SRS ▶ THX ▶ 5.1 ▶ Nicam Stereo ▶ Dual Mono <p>The CMS ingest sets this field from the <i>Series</i> object's Audio Mode EPG attribute.</p> |
| audioPid | String | The value of an alternative audio PID. This is used when playing an asset that contains multiple audio streams, to correctly decode the correct stream. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | <p>A log of the changes to this <i>Entity</i> history.</p> <p>This field is managed by SDP and is not ingested from the CMS.</p> |
| contentType | String | <p>This field is provided for the use of operators which need to distinguish <i>Asset Groups</i> containing different types of <i>Asset</i>. When it is in use, the client is expected to know how to use it; any value stored here is opaque to SDP.</p> <p>The CMS ingest sets this field from the <i>Series</i> object's ContentType EPG attribute.</p> |
| copyright | String | <p>The copyright status or owner of this <i>Asset Group</i>.</p> <p>The CMS ingest sets this field from the <i>Series</i> object's Copyright EPG attribute.</p> |
| country | String | <p>A list of 2-letter ISO codes of countries for the <i>Asset Group</i>.</p> <p><i>For example:</i> us, fr, cn</p> |

| Name | Type | Description |
|---------------------------|--------------------|--|
| | | The CMS ingest sets this field from the <i>Series</i> object's Country of origin EPG attribute. |
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created in SDP. This field is managed by SDP and is not ingested from the CMS. |
| <code>definition</code> | String | The picture definition of <i>Assets</i> in this <i>Asset Group</i> . <i>For example:</i> <ul style="list-style-type: none"> ▶ HD ▶ SD |
| | | The CMS ingest sets this field from the <i>Series</i> object's Definition EPG attribute. |
| <code>description</code> | String | A short description of this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Description EPG attribute. |
| <code>directors</code> | String | The names of the directors of this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Directors EPG attribute. |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. This field is managed by SDP and is not ingested from the CMS. |
| <code>genre</code> | String | A semicolon-separated list of genres to which this <i>Asset Group</i> belongs, separated by semicolons. The CMS ingest sets this field from the categories EPG attribute of the <i>Series</i> object. It also adds the <i>Asset Group</i> to any <i>COD Categories</i> whose originKey match a value in the list, and whose originID is 1. |
| <code>groupType</code> | Deprecated. String | The type of <i>Asset Group</i> this is. An organisation running SDP may use its own criteria to decide how to organize <i>Asset Groups</i> into types. The CMS ingest assigns the type G to every <i>Asset Group</i> it creates. |

| Name | Type | Description |
|-----------------------------|--|--|
| <code>language</code> | <code>string</code> | The original language of this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Original language EPG attribute. |
| <code>locale</code> | <code>string</code> | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. This field is managed by SDP and is not ingested from the CMS. |
| <code>modifiedDateML</code> | <code>Date</code> | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. This field is managed by SDP and is not ingested from the CMS. |
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> <code>string</code> | The name of this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Display Title EPG attribute. |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. This field is set to 1 for all <i>Entities</i> ingested from NAGRA Media CMS. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>string</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>string</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. For all <i>Entities</i> ingested from NAGRA Media CMS, this field is set to the <code>id</code> of the object. |
| <code>ppVPurchase</code> | <i>Read only.</i> <code>boolean</code> | This field is always <code>false</code> for <i>Asset Groups</i> . |

| Name | Type | Description |
|-----------------------------------|---------|---|
| <code>parentAclUID</code> | Long | This field is used internally by SDP during the purchase process. Clients should not attempt to use it. The CMS ingest does not set this field. |
| <code>parentPolicyGroupUID</code> | Long | This field is used internally by SDP during the purchase process. Clients should not attempt to use it. The CMS ingest does not set this field. |
| <code>parentalRating</code> | Long | The UID of the <i>Rating</i> for this <i>Asset Group</i> ; normally the most inclusive <i>Rating</i> which is still appropriate for all <i>Assets</i> in the group. The CMS ingest uses the <i>Series</i> object's Parental Rating , Mpaa Rating and TV Rating EPG attributes to determine a suitable value for this field. |
| <code>policyGroupUID</code> | Long | This field is used internally by SDP during the purchase process. Clients should not attempt to use it. The CMS ingest does not set this field. |
| <code>privateMetadata</code> | String | This field is provided for operators to use for their own purposes. SDP passes it to the client unchanged. Its format (string, JSON object...) and meaning should be decided by the operator. The CMS ingest places the <i>Series</i> object's Private Metadata EPG attribute in this field. |
| <code>producers</code> | String | The names of the producers of this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Producers EPG attribute. |
| <code>profileUID</code> | Long | This field should not be used. |
| <code>promoUrl</code> | String | The URL of a promotional image for this <i>Asset Group</i> . The CMS ingest sets this field from the <i>Series</i> object's Promoting picture URL EPG attribute. |
| <code>recordable</code> | Boolean | May this <i>Asset Group</i> be recorded by users? The CMS ingest sets this field from the <i>Series</i> object's Is Recordable EPG attribute. |

| Name | Type | Description |
|-----------------------------------|--------------------|--|
| <code>serviceProviderID</code> | Long | <p>The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i>.</p> <p>For <i>Entities</i> ingested from NAGRA Media CMS, this is the <code>uid</code> of the <i>Service Provider</i> whose <code>originKey</code> matches the <code>id</code> found on the ingest file's root. CMsvod Data element.</p> |
| <code>shortTitle</code> | String | <p>A shorter name for this <i>Asset Group</i>.</p> <p>The CMS ingest places the <i>Series</i> object's Short Title EPG attribute in this field.</p> |
| <code>studio</code> | String | <p>The name of the studio which produced this <i>Asset Group</i>.</p> <p>The CMS ingest sets this field from the <i>Series</i> object's Studio EPG attribute.</p> |
| <code>subscribableItemType</code> | String | <p>This field is always AGR for <i>Asset Groups</i>.</p> |
| <code>subscribed</code> | Read only. boolean | <p>Is the user already subscribed to this <i>Asset Group</i>?</p> <p>This field is returned as part of the <i>Asset Group</i> model object when clients query on-demand offerings. It is not part of the <i>Asset Group</i> Entity itself.</p> <p>Note Attempting to alter this field will have no effect. Clients should always subscribe to on-demand content through <i>Policy Groups</i>.</p> |
| <code>subtitles</code> | String | <p>A list of the languages in which the <i>Asset Group</i> includes subtitles.</p> <p>For example: <code>eng, fra, zho</code></p> <p>The CMS ingest sets this field from the <i>Series</i> object's Subtitles EPG attribute.</p> |
| <code>synopsis</code> | String | <p>A longer description of this <i>Asset Group</i>.</p> <p>The CMS ingest sets this field from the <i>Series</i> object's Short Synopsis EPG attribute.</p> |
| <code>uID</code> | Long | <p>The SDP internal UID for this <i>Entity</i>.</p> |

| Name | Type | Description |
|-------------|----------------|--|
| | | This field is managed by SDP and is not ingested from the CMS. |
| year | Integer | The year of production of this <i>Asset Group</i> , or another notable date defined by the operator. The CMS ingest sets this field from the <i>Series</i> object's Production year EPG attribute. |

10.2 tv.quative.service.asset.model.TrickPlay

| Name | Type | Description |
|--------------------|----------------|--|
| fastforward | Boolean | The RTSP Fast Forward mode Trick play. |
| pause | Boolean | The RTSP Pause mode Trick play. |
| play | Boolean | The RTSP play mode Trick play. |
| rewind | Boolean | The RTSP Rewind mode Trick play. |
| stop | Boolean | The RTSP Stop mode Trick play. |

10.3 assetGroupService

AssetGroupServiceImpl . Implements the AssetGroupService and used as services to interface to db to create, update and manipulate assetgroup records within the COD system The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass:

hue=true, ws=true) For the Methods: `@QubeService(name="assetGroupService",`

`ws=true)` The methods used to create, delete or update items of the COD System, only could be accesible to Web Services (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accesible both the WebServices and the Hue. `@QubeMethod(hue=false,`

Zafar Ali `@author`

Naeem Bilal `@author`

1.0 `@version`

10.3.1 assetGroupService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetGroupService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
<originKey>stringValue</originKey>
<originID>94</originID>
<locale>en_us</locale>
</getByOrigin>
```

10.3.2 assetGroupService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/assetGroupService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|---|
| <code>uid</code> | Long | Integer unique id for the <code>LocaleBaseModel</code> object |

| Name | Type | Description |
|--------|--------|-----------------------------------|
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel if found, the matching object, null otherwise.

Example

```
<getByUID>  
<uid>83</uid>  
<locale>en_us</locale>  
</getByUID>
```

11 billingService

This is a primitive *Service* which provides the *Billing Event* Entity.

11.1 Billing

A *Billing* Entity represents an action of an end user that can be charged for.

A *Billing* Entity is linked to a single *Account*, *Policy Group* and content item. It may also identify the *User* who caused the charge to be made (although the charge is incurred by the *Account* as a whole.)

When a user purchases a subscription which involves recurring payments, SDP produces a new *Billing* Entity whenever a payment is due. This is managed by a scheduled job running on the SDP.

11.1.1 Billing

| Name | Type | Description |
|--------------|---|--|
| accountUID | Long | The UID of the <i>Account</i> associated with this <i>Billing</i> Entity. |
| aclUID | Long | The UID of the <i>Acquired Content List</i> which records the purchase for which the <i>Account</i> is being billed. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| creditStatus | <i>Search criterion.</i> String | The direction of the transaction. A <i>Billing CreditStatus</i> value: CR or DB. |
| currency | String | The currency used for the purchase. |
| description | String | A short description of this <i>Entity</i> . |
| deviceUID | Long | The uid of the <i>Set Top Box</i> or <i>Media Player</i> with which the purchase was made, if known. |

Note

This field is populated using the device identified from the sign-on token, if available. If the user has not signed on by device, this field will be left empty. SDP does not use the device identifier supplied

| Name | Type | Description |
|----------------------------|--------------------------|---|
| | | <p>in the <i>Context</i>, even if this is supplied, as devices are permitted to use the <i>Context</i> to identify themselves as another device if they need to.</p> <p>Note In SDP, purchases are normally made across the whole <i>Account</i>, but it is sometimes useful to know which devices viewers are using to perform the purchase action.</p> |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>extRef1</code> | String | <p>An optional identifier for this billing event on an external system such as an SMS.</p> <p>SDP does not apply any logic to this field. Operators are free to use it how they want.</p> |
| <code>extRef2</code> | String | <p>A second optional identifier for this billing event on an external system.</p> <p>SDP does not apply any logic to this field. Operators are free to use it how they want.</p> |
| <code>frequencyType</code> | String | <p>The frequency type of the purchase: one of IMP (impulse), MUL (multiple), REC (recurring), or SEP (season pass).</p> <p>These values have the same meaning as the <code>frequency</code> field of the <i>Base Price</i> Entity. When SDP creates a <i>Billing</i> Entity, it sets this field to match the <code>frequency</code> of the <i>Base Price</i> associated with the purchased <i>Policy Group</i>.</p> |
| <code>incurredDate</code> | Date | The date and time the billable activity took place. |
| <code>itemName</code> | String | The name of the item involved in the billable activity. |
| <code>itemType</code> | Search criterion. String | <p>The type of the purchased item. A <i>Billing ItemType</i> or <i>Acquired Content List AcquiredContentListItemType</i> value (PLG, PKG, SUB, CAT...)</p> <p>Note The Java layer enforces <code>Billing.ItemType</code> when setting this field, but when</p> |

| Name | Type | Description |
|-----------------------------|-------------------|--|
| | | created by the DB billing batch job, the item type is taken directly from the ACL item type. |
| <code>itemUID</code> | Long | The UID of the purchased item. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>offerPrice</code> | Double | |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>paymentMethod</code> | String | The payment method used. One of <ul style="list-style-type: none"> ▶ SDP-Post Paid, for post-paid <i>Payment Method Providers</i>; ▶ Pay Wizard eWallet, for <i>Payment Method Providers</i> using the PayWizard E-wallet payment method; or ▶ Card, for <i>Payment Method Providers</i> using the Pay Wizard card payment method. |
| <code>paymentType</code> | String | The type of payment method used. One of <ul style="list-style-type: none"> ▶ POST, for post-paid <i>Payment Method Providers</i>; ▶ PRE, for pre-paid <i>Payment Method Providers</i> (Pay Wizard E-wallet and card payment methods); or ▶ FREE, if no payment was needed because the content was available at zero price. |

| Name | Type | Description |
|------------------------------------|---------------------------------|--|
| <code>serviceProviderID</code> | Long | The <code>uid</code> of the <i>Service Provider</i> associated with the bill. This will be the <i>Service Provider</i> associated with the <i>Account</i> if the subscriber purchased the item themselves, or the <i>Service Provider</i> associated with the logged-in <i>User</i> if the purchase was initiated through QSP Manager. |
| <code>status</code> | <i>Search criterion.</i> String | The payment status of the bill. A <i>Billing status</i> value: P, U, A or D. SDP does not manage this field. It is provided to allow client applications to display feedback on billing status. To provide this feature, the operator must implement a process which updates this field. |
| <code>transaction Reference</code> | String | An identifier generated by SDP and sent to the payment provider to represent this transaction, if applicable. |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>userName</code> | String | The username of the <i>User</i> who initiated the purchase. |
| <code>value</code> | Float | The value of the bill incurred, in the specified currency. |

11.2 billingService

Billing Service Implementation. Implementation which performs the compulsory methods defined in the BillingService interface. BillingService implemented object is used as WebServices to communicate with the Qube Billing system. The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The methods used to create, delete or update items of the Billing System, only could be accesible to Web Services (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Narasimha Kamaraju @author

1.1 @version

11.2.1 billingService.create

Create (persist) a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

11.2.2 `billingService.delete`

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

11.2.3 `billingService.getByAccountUID`

Gets all the *Billing* Entities which record payments billed to the specified *Account*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getByAccountUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|---------------------------------|
| <code>accountUID</code> | Long | The uid of the <i>Account</i> . |

Return value

Type: `List<Billing>`

Example

```
<getByAccountUID>
<accountUID>15</accountUID>
</getByAccountUID>
```

11.2.4 billingService.getByAccountUIDAndRange

Gets all the *Billing* Entities which record payments billed to the specified *Account* in the specified date range.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getByAccountUIDAndRange`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|--|
| <code>accountUID</code> | Long | The <code>uid</code> of the <i>Account</i> . |
| <code>startDate</code> | Date | Include payments billed after this date. |
| <code>endDate</code> | Date | Include payments billed before this date. |

Return value

Type: `List<Billing>`

11.2.5 billingService.getByAccountUIDAndStatus

Gets all the *Billing* Entities which record payments billed to the specified *Account* which have the specified `status`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getByAccountUIDAndStatus`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|--|
| <code>accountUID</code> | Long | The <code>uid</code> of the <i>Account</i> . |

| Name | Type | Description |
|---------------------|--------|--|
| <code>status</code> | String | The <code>status</code> of the <i>Billing</i> Entities to return. A <i>Billing Status</i> value: one of A , D , P or U . |

Return value

Type: `List<Billing>`

Example

```
<getByAccountUIDAndStatus>
  <accountUID>72</accountUID>
  <status>stringValue</status>
</getByAccountUIDAndStatus>
```

11.2.6 billingService.getByAccountUIDAndUser

Gets all the *Billing* Entities which record payments billed to the specified *Account* and *User*.

Some *Billing* Entities do not specify a *User*, for example those arising from purchases made by an SMS. These will not be included in the returned list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getByAccountUIDAndUser`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>accountUID</code> | Long | The <code>uid</code> of the <i>Account</i> . |
| <code>userName</code> | String | The <code>name</code> of the <i>User</i> . |

Return value

Type: `List<Billing>`

Example

```
<getByAccountUIDAndUser>
  <accountUID>79</accountUID>
  <userName>stringValue</userName>
</getByAccountUIDAndUser>
```

11.2.7 billingService.getByAccountUIDStatusAndRange

Gets all the *Billing* Entities which record payments billed to the specified *Account* in the specified date range which have the specified *status*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getByAccountUIDStatusAndRange`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>accountUID</code> | Long | The <code>uid</code> of the <i>Account</i> . |
| <code>status</code> | String | The <code>status</code> of the <i>Billing</i> Entities to return. A <i>Billing status</i> value: one of <code>A</code> , <code>D</code> , <code>P</code> or <code>U</code> . |
| <code>startDate</code> | Date | Include payments billed after this date. |
| <code>endDate</code> | Date | Include payments billed before this date. |

Return value

Type: `List<Billing>`

11.2.8 billingService.getByAcID

A Service method to retrieve billing details by `acID`.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|------|-------------|
| <code>acID</code> | Long | Long |

Return value

Type: `List<Billing>`

List of Billing objects

Example

```
<getByAcID>
  <acID>28</acID>
</getByAcID>
```

11.2.9 billingService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>25</originID>
</getByOrigin>
```

11.2.10 billingService.getByOriginID

A Service method to get the billing details list for origin uid.

Contains the business logic required to get billing details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|-------------|
| originUID | Integer | Integer |

Return value

Type: List<Billing>

List of Billing object

Example

```
<getByOriginID>
  <originUID>87</originUID>
</getByOriginID>
```

11.2.11 billingService.getByRange

Gets all the *Billing* Entities which record payments billed in the specified date range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|---|
| startDate | Date | Include payments billed after this date. |
| endDate | Date | Include payments billed before this date. |

Return value

Type: List<Billing>

11.2.12 billingService.getByStatus

Gets all the *Billing* Entities which record payments billed to any *Account* which have the specified *status*.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| status | String | The <i>status</i> of the <i>Billing</i> Entities to return. A <i>Billing Status</i> value: one of A , D , P or U . |

Return value

Type: List<Billing>

Example

```
<getByStatus>
  <status>stringValue</status>
</getByStatus>
```

11.2.13 billingService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

`BaseModel` If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>40</uid>
</getByUID>
```

11.2.14 billingService.getList

A Service method to get the billing details list for locale.

Contains the business logic required to get billing details list.

Available through Web Services.

Parameters

None.

Return value

Type: `List<Billing>`

List of Billing object

Example

```
<getList/>
```


11.2.15 billingService.getMoneySpentByAccountAndDateAndBillType

A Service method to get the total amount of money spent by an account in a specific type of item for a period of time..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getMoneySpentByAccountAndDateAndBillType`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>accountUID</code> | Long | the accountUID |
| <code>startDate</code> | Date | the start date of the period of time we want to consider. If you specified null value and a proper value in <code>endDate</code> , we will only consider the <code>endDate</code> in the query. If you provide null values for both <code>startDate</code> and <code>endDate</code> , we will consider all the records. |
| <code>endDate</code> | Date | the end date of the period of time we want to consider. If you specified null value and a proper value in <code>startDate</code> , we will only consider the <code>startDate</code> in the query. If you provide null values for both <code>startDate</code> and <code>endDate</code> , we will consider all the records. |
| <code>billType</code> | String | the type of bill we want to include in the sum |

Return value

Type:

the amount of money spent by 'accountUID' from 'startDate' to 'endDate' in items of type 'billType'.

11.2.16

billingService.getMoneySpentInLastNDaysByAccountAndBillType

A Service method to get the total amount of money spent by an account in a specific type of item for a number of days back in the past..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/billingService/getMoneySpentInLastNDaysByAccountAndBillType`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|---|
| accountUID | Long | the accountUID |
| daysBack | Long | the number of days back to the past to consider in the calculation. |
| billType | String | the type of bill we want to include in the sum |

Return value

Type:

the amount of money spent by 'accountUID' from 'startDate' to 'endDate' in items of type 'billType'.

Example

```
<getMoneySpentInLastNDaysByAccountAndBillType>
<accountUID>85</accountUID>
<daysBack>51</daysBack>
<billType>stringValue</billType>
</getMoneySpentInLastNDaysByAccountAndBillType>
```

11.2.17 billingService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows updated.

12 blockingService

This is a primitive *Service* which provides the *Blocking* Entity.

12.1 Blocking

The *Blocking* Entity.

No content

12.1.1 Blocking

| Name | Type | Description | | | | | | | | |
|--------------|--|---|-------|-------------|-------------|--------------------------------------|-----------|--|-------------|--|
| accountUID | Long | The UID of the <i>Account</i> which is blocked. | | | | | | | | |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. | | | | | | | | |
| creationDate | Date | The date created. | | | | | | | | |
| endTime | Date | The date and time the blocking will stop being effective. | | | | | | | | |
| exportID | Integer | The id to export object to third part systems. | | | | | | | | |
| intervalUID | Long | When the blocking type is TIME_ON_CHANNEL , an <i>Interval</i> describing the time the <i>Channel</i> should be blocked. | | | | | | | | |
| itemType | String | The type of blocking to carry out. One of: <table border="1" data-bbox="794 1458 1401 1783"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>C (CHANNEL)</td> <td>Block the specified <i>Channel</i>.</td> </tr> <tr> <td>E (EVENT)</td> <td>Block the specified <i>Event</i> (but not other showings of the same program.)</td> </tr> <tr> <td>P (PROGRAM)</td> <td></td> </tr> </tbody> </table> | Value | Description | C (CHANNEL) | Block the specified <i>Channel</i> . | E (EVENT) | Block the specified <i>Event</i> (but not other showings of the same program.) | P (PROGRAM) | |
| Value | Description | | | | | | | | | |
| C (CHANNEL) | Block the specified <i>Channel</i> . | | | | | | | | | |
| E (EVENT) | Block the specified <i>Event</i> (but not other showings of the same program.) | | | | | | | | | |
| P (PROGRAM) | | | | | | | | | | |

| Name | Type | Description | | | | | | | | | | | | |
|--------------------------------|---|---|-------|-------------|--|---|-----------|--|----------------|--|------------|--|----------------------|---|
| | | <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td></td> <td>Block the specified <i>Program</i> whenever it is shown on any channel.</td> </tr> <tr> <td>A (ASSET)</td> <td>Block the specified VOD <i>Asset</i>.</td> </tr> <tr> <td>CA (CATALOGUE)</td> <td>Block the specified VOD <i>Catalogue</i>.</td> </tr> <tr> <td>R (RATING)</td> <td>Block all content with the specified <i>Rating</i>.</td> </tr> <tr> <td>TC (TIME_ON_CHANNEL)</td> <td>Block the specified <i>Channel</i> every day between the specified times.</td> </tr> </tbody> </table> | Value | Description | | Block the specified <i>Program</i> whenever it is shown on any channel. | A (ASSET) | Block the specified VOD <i>Asset</i> . | CA (CATALOGUE) | Block the specified VOD <i>Catalogue</i> . | R (RATING) | Block all content with the specified <i>Rating</i> . | TC (TIME_ON_CHANNEL) | Block the specified <i>Channel</i> every day between the specified times. |
| Value | Description | | | | | | | | | | | | | |
| | Block the specified <i>Program</i> whenever it is shown on any channel. | | | | | | | | | | | | | |
| A (ASSET) | Block the specified VOD <i>Asset</i> . | | | | | | | | | | | | | |
| CA (CATALOGUE) | Block the specified VOD <i>Catalogue</i> . | | | | | | | | | | | | | |
| R (RATING) | Block all content with the specified <i>Rating</i> . | | | | | | | | | | | | | |
| TC (TIME_ON_CHANNEL) | Block the specified <i>Channel</i> every day between the specified times. | | | | | | | | | | | | | |
| <code>itemTypeCode</code> | <i>Deprecated.</i> String | | | | | | | | | | | | | |
| <code>itemUID</code> | Long | The UID of the <i>Entity</i> to be blocked. | | | | | | | | | | | | |
| <code>modifiedDate</code> | Date | The date last modified and persisted. | | | | | | | | | | | | |
| <code>originID</code> | Integer | The uid of the origin system (3rd part system). | | | | | | | | | | | | |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object derived by concatenating the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. | | | | | | | | | | | | |
| <code>originKey</code> | String | The key of the model object in the origin system (3rd part system). | | | | | | | | | | | | |
| <code>serviceProviderID</code> | Long | The Service provider UID. | | | | | | | | | | | | |
| <code>startTime</code> | Date | The date and time the blocking will start being effective. | | | | | | | | | | | | |

| Name | Type | Description |
|--------|------|---|
| uID | Long | The unique identifier. |
| userID | Long | The UID of the <i>User</i> who should be blocked. |

12.2 tv.quative.service.blocking.model.ItemType

| Name | Type | Description |
|-----------|------|------------------------------|
| A | | ItemType for 'A' Code. |
| ASSET | | Code for ASSET ItemType. |
| C | | ItemType for 'C' Code. |
| CA | | ItemType for 'CA' Code. |
| CATALOGUE | | Code for CATALOGUE ItemType. |
| CHANNEL | | Code for CHANNEL ItemType. |
| E | | ItemType for 'E' Code. |
| EVENT | | Code for EVENT ItemType. |
| P | | ItemType for 'P' Code. |
| PROGRAM | | Code for PROGRAM ItemType. |
| R | | ItemType for 'R' Code. |
| RATING | | Code for RATING Status. |
| TC | | ItemType for 'TC' Code. |

| Name | Type | Description |
|-----------------|------|------------------------------------|
| TIME_ON_CHANNEL | | Code for TIME_ON_CHANNEL ItemType. |

12.3 blockingService *Updated*

Blocking Service Implementation. Implementation which performs the compulsory methods defined in the Blocking Service interface. BlockingService implemented object is used as WebServices to communicate with the Account Management (AM) system. The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true, instrument = false, roles = "<allowed-roles>") The methods used to create, delete or update items of the AM System, only could be accessible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Ashish Gupta @author

1.0 @version

Caution! ■

12.3.1 blockingService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|-------------|
| blocking | Blocking ^{p.211} | |

Return value

Type:

12.3.2 blockingService.create

A Service method to expose the creation of blocking object into the system.

Calls the corresponding overloaded method with Blocking model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/create`

Parameters

| Name | Type | Description |
|-------------|---------|--|
| accountUID | Long | Long account uid |
| userUID | Long | Long user uid |
| itemType | String | String item type |
| itemUID | Long | Long item uid |
| intervalUID | Long | Long interval uid |
| startTime | Date | Date |
| endTime | Date | Date |
| originKey | String | String origin key i.e. client identifier |
| originUID | Integer | Integer origin uid |

Return value

Type:

Long blocking uid

12.3.3 blockingService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

12.3.4 blockingService.delete

A Service method to expose the deletion of blocking from the system.

Calls the corresponding overloaded method with Blocking model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/delete`

Parameters

| Name | Type | Description |
|------|------|-------------------|
| uid | Long | Long blocking uid |

Return value

Type:

Integer number of records deleted

12.3.5 blockingService.delete

Delete, from storage, a `BaseModel`.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

12.3.6 blockingService.getBlockedList

A Service method to get the blocking item list for user uid.

Contains the business logic required to get blocking details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/getBlockedItemList`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|------------------|
| accountUID | Long | Long account uid |
| userID | Long | Long user uid |
| itemType | String | String item type |

Return value

Type: `List<Blocking>`

List of Blocking object

Example

```
<getBlockedItemList>
<accountUID>67</accountUID>
<userID>73</userID>
<itemType>stringValue</itemType>
</getBlockedItemList>
```

12.3.7 blockingService.getBlockedStatusForItem

A Service method to get the blocked status for item.

Contains the business logic required to get blocked status.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/getBlockedStatusForItem`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------|
| accountUID | Long | Long account uid |
| userID | Long | Long user uid |

| Name | Type | Description |
|-----------------------|--------|------------------|
| <code>itemUID</code> | Long | Long item uid |
| <code>itemType</code> | String | String item type |

Return value

Type:

Boolean value true if item blocked

Example

```
<getBlockedStatusForItem>
<accountUID>79</accountUID>
<userID>49</userID>
<itemUID>72</itemUID>
<itemType>stringValue</itemType>
</getBlockedStatusForItem>
```

12.3.8 blockingService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
<originKey>stringValue</originKey>
```

```
<originID>15</originID>
</getByOrigin>
```

12.3.9 blockingService.getByOriginUID

A Service method to get the blocking details list for origin uid.

Contains the business logic required to get blocking details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/getByOriginUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|---------|------------------|
| originUID | Integer | Long origin uid |
| accountUID | Long | Long account uid |
| userUID | Long | Long user uid |

Return value

Type: `List<Blocking>`

List of Blocking object

Example

```
<getByOriginUID>
<originUID>29</originUID>
<accountUID>65</accountUID>
<userUID>57</userUID>
</getByOriginUID>
```

12.3.10 blockingService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>87</uid>
</getByUID>
```

12.3.11 blockingService.getList

A Service method to get the blocking details list for user uid.

Contains the business logic required to get blocking details list.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/getList>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------|
| accountUID | Long | Long account uid |
| userUID | Long | Long user uid |

Return value

Type: List<Blocking>

List of Blocking object

Example

```
<getList>
  <accountUID>89</accountUID>
  <userUID>56</userUID>
</getList>
```

12.3.12 blockingService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|-------------|
| blocking | Blocking ^{p.211} | |

Return value

Type:

12.3.13 blockingService.update

A Service method to expose the updation of blocking into the system.

Calls the corresponding overloaded method with Blocking model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/blockingService/update`

Parameters

| Name | Type | Description |
|-------------|--------|-------------------|
| uid | Long | Long blocking uid |
| accountUID | Long | Long account uid |
| userUID | Long | Long user uid |
| itemType | String | String item type |
| itemUID | Long | Long item uid |
| intervalUID | Long | Long interval uid |
| startTime | Date | Date |
| endTime | Date | Date |

| Name | Type | Description |
|---------------------------|------|-------------|
| <code>modifiedDate</code> | Date | Date |

Return value

Type:

Integer number of records updated

12.3.14 blockingService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

13 bocPurchaseService

This *Service* provides general APIs for clients to purchase content.

It provides two main methods:

- ▶ *bocPurchaseService.purchasePolicy* can be used to purchase a *Policy Group*.
- ▶ *bocPurchaseService.purchasePolicyAndGetLicense* can be used to purchase a *Policy Group* and also contact a license server to issue a license to watch content provided by that *Policy Group*.

The *bocPurchaseService* supports *Policy Groups* which provide on-demand content (*Assets* via *Schedules*); broadcast content (*Channels* via *Packages*); and pay-per-view content (*Events*).

Note | The *Policy Group* may be linked to *Schedules*, or *Packages* or *Events*, but not to a mixture of content types.

13.1 tv.quative.service.boc.purchase.api.PurchaseFor

Defines the access that should be provided by a purchase. Normally only one of these values will be in use in a given deployment.

| Name | Type | Description |
|-----------|------|---|
| ACCOUNT | | The item should be purchased across the whole <i>Account</i> . Any device belonging to the <i>Account</i> will be entitled to view the content, including devices which are provisioned to the <i>Account</i> after purchase. |
| | | Note This value must be passed in to the <i>bocPurchaseService</i> in quotes: "ACCOUNT". |
| SMARTCARD | | The item should be purchased for use by the device with the specified <i>Smart Card</i> only. |
| | | Note This value must be passed in to the <i>bocPurchaseService</i> in quotes: "SMARTCARD". |

13.2 tv.quative.service.policygroup.api.OfferSpecification

A specification object which can be used to identify a *Policy Group* using the identifier which is most convenient for the client. Clients only need to populate enough fields of this object that will ensure the *Policy Group* is uniquely identified.

| Name | Type | Description |
|----------------------------------|---------|--|
| <code>offerId</code> | String | |
| <code>polgrpCasId</code> | String | The <code>casId</code> of the <i>Policy Group</i> . No other fields are required if this field is set. |
| <code>polgrpCasInstanceId</code> | String | |
| <code>polgrpOriginId</code> | Integer | The <code>originId</code> of the <i>Policy Group</i> . If this field is set, the <code>polgrpOriginKey</code> should also be set. |
| <code>polgrpOriginKey</code> | String | The <code>originKey</code> of the <i>Policy Group</i> . If this field is set, the <code>polgrpOriginId</code> should also be set. |
| <code>polgrpUid</code> | Long | The <code>uid</code> of the <i>Policy Group</i> . No other fields are required if this field is set. |

13.3

tv.quative.service.boc.purchase.api.BocExternalPaymentSignatureReq

| Name | Type | Description |
|-----------------------------|--------------------|---|
| <code>context</code> | Context | The context of the request. |
| <code>offer</code> | OfferSpecification | The content being purchased. |
| <code>paymentDetails</code> | PaymentDetails | The choice of payment method. |
| <code>purchaseFor</code> | PurchaseFor | What to purchase for, ACCOUNT or SMARTCARD etc. |

13.4 tv.quative.service.boc.purchase.api.PaymentDetails

A transfer object allowing clients to specify the *Service Provider Payment Option* they want to use for a purchase, together with any other information required by the payment provider to process the purchase.

| Name | Type | Description |
|---|---------------------|---|
| <code>cvc</code> | <code>String</code> | For card-based payments, the card security code, otherwise known as CVV, CVVC, CSC, CVD, CCV, etc. Note Other card details (card number, expiry date...) are held by the payment provider and are not handled by SDP. |
| <code>password</code> | <code>String</code> | For payment methods which require a username and password (PayWizard E-wallet...), the password to use to access the payment account. |
| <code>payment CredentialsUid</code> | <code>Long</code> | For payment methods which require <i>Accounts</i> to register credentials (for example, credit cards), the <code>uid</code> of the <i>Account Payment Credentials</i> to use. |
| <code>pin</code> | <code>String</code> | The account holder's PIN, if required by the payment method. Note This field is not currently used by any payment methods. |
| <code>serviceProvider PaymentOptionUid</code> | <code>Long</code> | The <code>uid</code> of the <i>Service Provider Payment Option</i> which specifies the payment method to use. |
| <code>username</code> | <code>String</code> | For payment methods which require a username and password (PayWizard E-wallet...), the username to use to access the payment account. |

13.5

tv.quative.service.paygateway.api.ExternalPaymentSignature

| Name | Type | Description |
|--------------------------------|--------|--|
| <code>digest</code> | String | The digest we are using to verify the parameters passed to the payment provider. This is a combination of several parameters, concatenated together and hashed/encrypted in some form. |
| <code>sdpTransactionRef</code> | Long | SDP Transaction Ref is the unique SDP reference for the transaction we are making. |

13.6

tv.quative.service.boc.purchase.api.BocPurchaseWithProofOfPayment

| Name | Type | Description |
|-----------------------------|----------------|---|
| <code>context</code> | Context | The context of the request. |
| <code>proofOfPayment</code> | ProofOfPayment | Proof of external payment. |
| <code>purchaseFor</code> | PurchaseFor | What to purchase for, ACCOUNT or SMARTCARD etc. |

13.7 tv.quative.service.boc.purchase.api.ProofOfPayment

| Name | Type | Description |
|-------------------------|--------|---|
| <code>cardExpiry</code> | String | The card expiry date in 'MM/YY' format. |
| <code>cardPan</code> | String | The card number with mask. |
| <code>cardToken</code> | String | The token that identifies the card with the payment provider. |
| <code>cardType</code> | String | The type of card. |

| Name | Type | Description |
|--|--------|---|
| <code>paymentProviderTransactionRef</code> | String | The payment transaction reference assigned by the payment provider. |
| <code>receiptDigest</code> | String | The digest to validate the response. |
| <code>sdpTransactionRef</code> | Long | The transaction reference assigned by SDP. |

13.8 bocPurchaseService

Implementation of the BocPurchaseService interface.

Chris Francis @author

13.8.1 bocPurchaseService.acquireZeroPricedPolicy

Subscribe to a *Policy Group* that is available at no cost.

This method bypasses the normal payment processing, but verifies that the specified *Policy Group* has zero price. If the *Policy Group* must be paid for, an error is returned.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/acquireZeroPricedPolicy`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--|---|
| <code>pContext</code> | <code>tv.quative.service.context.api.Context</code> | Context object with information about the client making the purchase. |
| <code>pOffer</code> | <code>tv.quative.service.policygroup.api.OfferSpecification</code> | An Offer Specification object containing information which identifies the <i>Policy Group</i> to be purchased. |
| <code>pPurchaseFor</code> | <code>tv.quative.service.boc.purchase.api.PurchaseFor</code> | Whether to provide access across the whole <i>Account</i> or just to a single smart card. One of ACCOUNT or SMARTCARD . |

Return value

Type: `List<String>`

A list of the `uid` of each *Acquired Content List* created as a result of the purchase.

13.8.2 bocPurchaseService.channelSelectionResume

Channel selection resume..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/channelSelectionResume`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--------|----------------------|
| <code>purchaseToken</code> | String | the token |
| <code>deviceId</code> | String | device ID |
| <code>channelID</code> | String | channel ID |
| <code>eventID</code> | String | event ID |
| <code>fromStart</code> | String | from start |
| <code>privateString</code> | String | the private string |
| <code>microStartTime</code> | Long | the micro start time |
| <code>microEndTime</code> | Long | the micro end time |

Return value

Type:

the token

Example

```
<channelSelectionResume>
  <purchaseToken>stringValue</purchaseToken>
  <deviceId>stringValue</deviceId>
  <channelID>ABC-32729</channelID>
  <eventID>ABC-31995</eventID>
  <fromStart>stringValue</fromStart>
  <privateString>stringValue</privateString>
  <microStartTime>36</microStartTime>
  <microEndTime>17</microEndTime>
</channelSelectionResume>
```

13.8.3 bocPurchaseService.channelSelectionStart

channel Selection Start..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/channelSelectionStart`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------------|--------|----------------------|
| <code>deviceId</code> | String | device ID |
| <code>channelID</code> | String | channel ID |
| <code>eventID</code> | String | event ID |
| <code>startDateTimeString</code> | String | start time |
| <code>privateString</code> | String | the private string |
| <code>microStartTime</code> | Long | the micro start time |
| <code>microEndTime</code> | Long | the micro end time |

Return value

Type:

the token

Example

```
<channelSelectionStart>
  <deviceId>stringValue</deviceId>
  <channelID>ABC-61243</channelID>
  <eventID>ABC-91613</eventID>
  <startDateTimeString>stringValue</startDateTimeString>
  <privateString>stringValue</privateString>
  <microStartTime>2</microStartTime>
  <microEndTime>40</microEndTime>
</channelSelectionStart>
```

13.8.4 bocPurchaseService.getExternalPaymentSignature

Get the necessary payment signature to authenticate a payment request to a third party payment provider via client redirection.

The purchase will also be validated as part of this method. This method will create a payment transaction record which is necessary for #purchasePolicyWithProofOfPayment (BocPurchaseWithProofOfPaymentRequest) to work.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/getExternalPaymentSignature`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|--|--|
| pRequest | tv.quative.service.boc.purchase.api.BocExternalPaymentSignatureRequest | details of the purchase and payment method to be used. |

Return value

Type:

signature details for the external payment request.

13.8.5 bocPurchaseService.getPaymentOptionsForAccount

Fetch the payment options that are available to the specified *Account*.

All payment options supported by the *Account's Service Provider* are returned. The return objects include details of any credentials SDP holds for the *Account*.

Note

If a payment option requires credentials, but these are missing, the payment method cannot currently be used. The *Account* holder will need to contact the operator or payment provider to set up their credentials before they will be able to use this payment option.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/getPaymentOptionsForAccount`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--|--|
| pContext | tv.quative.service.context.api.Context | Context of the request. |
| pAccountUid | Long | The uid of the <i>Account</i> that is making the purchase. |

Return value

Type: `List<PaymentProviderDetails>`

Details of the payment options available.

13.8.6 bocPurchaseService.ngodPavCheck

Unified NGOD-compatible PAV (Playout access verification) request from SRM.

Verifies the purchase token, determines the correct playlist, and returns the lot.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|--------|--|
| <code>clientId</code> | String | The client's identifier. Usually a smartcard ID, although it could be a MAC address. |
| <code>purchaseTokenUuid</code> | String | The UUID for the purchase token. This is the STB's proof-of-purchase. |

Return value

Type: `Ngods1PavData`

A DTO containing all the SDP-owned information required to do to playout, and create the user-behavior record.

Example

```
<ngodPavCheck>
  <clientId>stringValue</clientId>
  <purchaseTokenUuid>stringValue</purchaseTokenUuid>
</ngodPavCheck>
```

13.8.7 bocPurchaseService.otvPavCheck

Unified OTV-compatible PAV (Playout access verification) request from SRM.

Verifies the asset uri, determines the correct playlist, and returns the lot.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|--------|--|
| <code>clientId</code> | String | The client's identifier. Usually a smartcard ID, although it could be a MAC address. |
| <code>assetUri</code> | String | The uri for the asset. |

Return value

Type: `OtvPavData`

A DTO containing all the SDP-owned information required to do to payout, and create the user-behavior record.

Example

```
<otvPavCheck>
  <clientId>stringValue</clientId>
  <assetUri>stringValue</assetUri>
</otvPavCheck>
```

13.8.8 bocPurchaseService.purchasePolicy

Subscribe to a policy group..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/purchasePolicy`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--|---|
| <code>context</code> | <code>tv.quative.service.context.api.Context</code> | Context object from which the <i>Account</i> (and, if necessary, <i>Smartcard</i>) making the purchase can be derived. |
| <code>offer</code> | <code>tv.quative.service.policygroup.api.OfferSpecification</code> | An <i>OfferSpecification</i> object which identifies the <i>Policy Group</i> to be purchased. |
| <code>purchaseFor</code> | <code>tv.quative.service.boc.purchase.api.PurchaseFor</code> | Whether the purchase should apply to the entire <i>Account</i> or only to the identified <i>Smart Card</i> . The value of this parameter must be supplied in quotes (JSON style): " ACCOUNT " or " SMARTCARD ". |

Return value

Type: `List<String>`

A list of the `uid` of each *Acquired Content List* created as a result of the purchase.

13.8.9 `bocPurchaseService.purchasePolicyAndGetLicense`

Shortcut method to subscribe to a policy group and get a license in one step.

A policy group may consist of more than one item so a separate license spec is still required as a parameter to say which individual asset we want the license for.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/purchasePolicyAndGetLicense`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--|--|
| <code>context</code> | <code>tv.quative.service.context.api.Context</code> | Current context |
| <code>offer</code> | <code>tv.quative.service.policygroup.api.OfferSpecification</code> | the offer specification ^{p.223} |
| <code>purchaseFor</code> | <code>tv.quative.service.boc.purchase.api.PurchaseFor</code> | what to purchase for, account or smartcard ^{p.223} |
| <code>licenseSpec</code> | <code>tv.quative.service.license.api.LicenseSpecification</code> | the license specification. content id for the underlying content we need a license for. Also would include any challenge parameters. |

Return value

Type: `LicenseResponse`

an encoded license for the content.

13.8.10 `bocPurchaseService.purchasePolicyAndPay`

Subscribe to a *Policy Group* and process payment using the details supplied.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/purchasePolicyAndPay`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|---|---|
| pContext | tv.quative.service.context.api.Context | A <i>Context</i> object from which the <i>Account</i> (and, if necessary, <i>Smartcard</i>) making the purchase can be derived. |
| pOffer | tv.quative.service.policygroup.api.OfferSpecification | An <i>OfferSpecification</i> object which identifies the <i>Policy Group</i> to be purchased. |
| pPurchaseFor | tv.quative.service.boc.purchase.api.PurchaseFor | Whether the purchase should apply to the entire <i>Account</i> or only to the identified <i>Smart Card</i> . The value of this parameter must be supplied in quotes (JSON style): "ACCOUNT" or "SMARTCARD". |
| pPayDetails | tv.quative.service.boc.purchase.api.PaymentDetails | A <i>PaymentDetails</i> object specifying how this purchase will be paid for. The required fields of this object depend on the payment type which has been chosen. |

Return value

Type: `List<String>`

A list of the `uid` of each *Acquired Content List* created as a result of the purchase.

13.8.11 bocPurchaseService.purchasePolicyPayAndGetLicense

Subscribe to a *Policy Group*, process payment using the details supplied, and obtain a license to watch content immediately.

Because a *Policy Group* may provide more than one item, the client must supply both an *OfferSpecification* (to specify the *Policy Group*) and a *LicenseSpecification* (to specify the content item which will be viewed first.)

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/purchasePolicyPayAndGetLicense`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|--|
| pContext | tv.quative.service.context.api.Context | A <i>Context</i> object from which the <i>Account</i> (and, if necessary, <i>Smartcard</i>) making the purchase can be derived. |
| pOffer | tv.quative.service.policygroup.api.OfferSpecification | An <i>OfferSpecification</i> object which identifies the <i>Policy Group</i> to be purchased. |

| Name | Type | Description |
|---------------------------|--|---|
| <code>pPurchaseFor</code> | <code>tv.quative.service.boc.purchase.api.PurchaseFor</code> | Whether the purchase should apply to the entire <i>Account</i> or only to the identified <i>Smart Card</i> . The value of this parameter must be supplied in quotes (JSON style): "ACCOUNT" or "SMARTCARD". |
| <code>pLicenseSpec</code> | <code>tv.quative.service.license.api.LicenseSpecification</code> | <i>License Specification</i> containing information which identifies the item to be licensed. |
| <code>pPayDetails</code> | <code>tv.quative.service.boc.purchase.api.PaymentDetails</code> | A <i>PaymentDetails</i> object specifying how this purchase will be paid for. The required fields of this object depend on the payment type which has been chosen. |

Return value

Type: `LicenseResponse`

An encoded license for the content.

13.8.12 `bocPurchaseService.purchasePolicyWithProofOfPayment`

Subscribe to a policy group using a proof of payment obtained via client redirection to a third party payment provider.

This method can only be called if `#getExternalPaymentSignature(BocExternalPaymentSignatureRequest)` has previously been executed to create a payment transaction.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/purchasePolicyWithProofOfPayment`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|---|---|
| <code>pRequest</code> | <code>tv.quative.service.boc.purchase.api.BocPurchaseWithProofOfPaymentRequest</code> | details of the purchase and proof of payment ^{p.226} |

Return value

Type: `List<String>`

list of uid of the created ACLs

13.8.13 `bocPurchaseService.selectionRestart`

NGOD E2 Interface.

restart means start again at the beginning. Allow a re-selection of format, screen shape, and languages.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/selectionRestart`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--------|--|
| <code>purchaseToken</code> | String | a UUID string for purchase token |
| <code>deviceId</code> | String | the MAC address of the STB |
| <code>titleProviderID</code> | String | provider ID of title asset |
| <code>titleAssetID</code> | String | asset ID of the title asset |
| <code>format</code> | String | format "HD" for high definition and "SD" for standard definition |
| <code>privateString</code> | String | the private string |
| <code>microStartTime</code> | Long | the micro start time |
| <code>microEndTime</code> | Long | the micro end time |

Return value

Type:

a UUID string for purchase token

Example

```
<selectionRestart>
  <purchaseToken>stringValue</purchaseToken>
  <deviceId>stringValue</deviceId>
  <titleProviderID>ABC-10348</titleProviderID>
  <titleAssetID>ABC-59970</titleAssetID>
  <format>stringValue</format>
  <privateString>stringValue</privateString>
  <microStartTime>8</microStartTime>
  <microEndTime>39</microEndTime>
</selectionRestart>
```

13.8.14 bocPurchaseService.selectionResume

NGOD E2 Interface.

Resume means start from the beginning or from the bookmark point. Allow a re-selection of format, screen shape, and languages.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/selectionResume`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--------|---|
| <code>purchaseToken</code> | String | a UUID string for purchase token |
| <code>deviceId</code> | String | the MAC address of the STB |
| <code>titleProviderID</code> | String | provider ID of title asset |
| <code>titleAssetID</code> | String | asset ID of the title asset |
| <code>format</code> | String | format "HD" for high definition and "SD" for standard definition |
| <code>fromStart</code> | String | Y for resuming from start of the VOD, N for getting the start point from the bookmark |
| <code>privateString</code> | String | the private string |
| <code>microStartTime</code> | Long | the micro start time |
| <code>microEndTime</code> | Long | the micro end time |

Return value

Type:

a UUID string for purchase token

Example

```
<selectionResume>
<purchaseToken>stringValue</purchaseToken>
<deviceId>stringValue</deviceId>
<titleProviderID>ABC-36812</titleProviderID>
<titleAssetID>ABC-42448</titleAssetID>
```

```
<format>stringValue</format>
<fromStart>stringValue</fromStart>
<privateString>stringValue</privateString>
<microStartTime>9</microStartTime>
<microEndTime>80</microEndTime>
</selectionResume>
```

13.8.15 bocPurchaseService.selectionStart

NGOD E2 Interface. When a subscriber selects a VOD show to be viewed, the client call this interface to get a purchase token..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/bocPurchaseService/selectionStart>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|---------|---|
| deviceId | String | the MAC address of the STB |
| titleProviderID | String | provider ID of title asset |
| titleAssetID | String | asset ID of the title asset |
| format | String | "HD" for high definition and "SD" for standard definition |
| privateString | String | the private string |
| microStartTime | Long | the micro start time |
| microEndTime | Long | the micro end time |
| isPreview | boolean | if requests a trailer |

Return value

Type:

a UUID string for purchase token

14 bookmarkService

This is an aggregate *Service* which allows a specified position in a content item to be saved as a bookmark.

14.1 Bookmark

The *Bookmark* Entity.

No content

14.1.1 Bookmark *Updated*

| Name | Type | Description | | | | | | |
|---------------|---|--|-------|-------------|-------------|--------------|---------------|----------------|
| accountUid | Long | The UID of the Account which owns the Bookmark . | | | | | | |
| changeLog | Deprecated. Not visible to HUE. List<String> | A log of the changes to this Entity history. | | | | | | |
| contentType | String | The type of content containing the Bookmark . One of: <table border="1" data-bbox="794 1137 1401 1395"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>AST (ASSET)</td> <td>Asset</td> </tr> <tr> <td>PRG (PROGRAM)</td> <td>Program</td> </tr> </tbody> </table> | Value | Description | AST (ASSET) | Asset | PRG (PROGRAM) | Program |
| Value | Description | | | | | | | |
| AST (ASSET) | Asset | | | | | | | |
| PRG (PROGRAM) | Program | | | | | | | |
| contentUid | Long | The UID of the content item containing the Bookmark . | | | | | | |
| creationDate | Date | The date and time this Entity was created. | | | | | | |
| exportID | Integer | The identifier to use when exporting this Entity to third-party systems. | | | | | | |
| modifiedDate | Date | The date and time this Entity was last modified in the database. | | | | | | |
| originID | Integer | An identifier for the system which owns this Entity . | | | | | | |

| Name | Type | Description |
|--------------------------------|-------------------|--|
| | | Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this Entity uses to identify it. |
| <code>position</code> | Long | The position, in seconds, of the Bookmark within the content item's stream. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the Service Provider which created this Entity . |
| <code>uID</code> | Long | The SDP internal UID for this Entity . |

14.2 bookmarkService *Updated*

BookmarkService Implementation. Manages the business logic of all the operations with Bookmarks. Object which implements the business logic of all the methods of the interface. The services are made visible to Hue or to Web Services using annotations in the Service Implementation class. Ex:

1.3 @version

Chris Morgan, Antony Bosco - Created for Mostel @author

Caution! ■

14.2.1 bookmarkService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

14.2.2 bookmarkService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

14.2.3 bookmarkService.deleteBookmarkByUid

Deletes the bookmark by uid..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/bookmarkService/deleteBookmarkByUid>

Parameters

| Name | Type | Description |
|--------------------------|------|-------------|
| <code>bookmarkUid</code> | Long | |

Return value

Type:

no of rows deleted.

14.2.4 bookmarkService.getBookmarkForContent

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/bookmarkService/getBookmarkForContent`

Parameters

| Name | Type | Description |
|--------------------------|--------|-------------|
| <code>contentUid</code> | Long | |
| <code>contentType</code> | String | |

Return value

Type:

14.2.5 bookmarkService.getBookmarkForOrigin

Description

`getBookmarkForOrigin` is used to get a Bookmark for a given content type (either Asset/Channel/Event).

This method has been added to support Locker and reuses the existing fields in the underlying database where possible.

The content may be identified in many subsystems, as shown in the examples below.

| Subsystem | Identifier | Example |
|-----------|------------------------------------|--------------------------|
| SDP | Asset is identified by ASS_UID | 2378 |
| CMS | Asset is identified by ORIGIN_KEY | YSGS10HDContHD |
| LOCKER | Asset is identified by a Locker Id | 53722221ff5edf02fc4650c5 |

URL

`http://<APIAdaptorHost>/adaptor/hue-gateway/gateway/http/js/bookmarkService/setBookmarkForOrigin`

Parameters

| Positional Parameter | Type | Named Parameter | Description | Mandatory | Example |
|----------------------|--------|---------------------|--|-----------|-----------------------------|
| arg0 | int | originUid | The origin uid of the subsystem used to identify the content (CMS/Locker/SDP) | Y | 1 |
| arg1 | string | originKey | The identifier of the content in the aforementioned subsystem. | Y | 53722221ff5edf02fc4650c5 |
| arg2 | string | origin Content Type | A free-form field that the client may use to further classify/organise the content | Y | Recording |
| arg3 | string | contentType Name | Describes the content type. | Y | ASSET/ PROGRAM/ EVENT |

14.2.6 bookmarkService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>12</originID>
</getByOrigin>
```

14.2.7 bookmarkService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>26</uid>
</getByUID>
```

14.2.8 bookmarkService.setBookmarkForContent

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/bookmarkService/setBookmarkForContent>

Parameters

| Name | Type | Description |
|-------------|--------|-------------|
| contentUid | Long | |
| contentType | String | |
| position | Long | |

Return value

Type:

14.2.9 bookmarkService.setBookmarkForOrigin

Description

setBookmarkForOrigin is used to set a Bookmark for a given content type (either Asset/Channel/Event).

This method has been added to support Locker and reuses the existing fields in the underlying database where possible.

The content may be identified in many subsystems, as shown in the examples below.

| Subsystem | Identifier | Example |
|-----------|------------------------------------|--------------------------|
| SDP | Asset is identified by ASS_UID | 2378 |
| CMS | Asset is identified by ORIGIN_KEY | YSGS10HDContHD |
| LOCKER | Asset is identified by a Locker Id | 53722221ff5edf02fc4650c5 |

URL

`http://<APIAdaptorHost>/adaptor/hue-gateway/gateway/http/js/bookmarkService/setBookmarkForOrigin`

Parameters

| Positional Parameter | Type | Named Parameter | Description | Mandatory | Example |
|----------------------|--------|---------------------|--|-----------|-----------------------------|
| arg0 | int | originUid | The origin uid of the subsystem used to identify the content (CMS/Locker/SDP) | Y | 1 |
| arg1 | string | originKey | The identifier of the content in the aforementioned subsystem. | Y | 53722221ff5edf02fc4650c5 |
| arg2 | string | origin Content Type | A free-form field that the client may use to further classify/organise the content | Y | Recording |
| arg3 | string | contentType Name | Describes the content type. | Y | ASSET/ PROGRAM/ EVENT |
| arg4 | long | position | The bookmark position. | Y | |

14.2.10 bookmarkService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

15 btvBrowsingService

This *Service* provides APIs for browsing through available broadcast *Channels*.

It provides the API *btvBrowsingService.getChannels*, which returns technical and editorial *Channels* in the form of *ChannelResponseItem* objects.

A *ChannelResponseItem* contains all the information a client will need to display a *Channel*,

The `context` and `filter` supplied to the API determine which items to return. For example, clients can retrieve the *Channels* in currently-purchased *Packages*, or those in a *Category*. Multiple filters can be applied at the same time; clients can also specify the sort order of returned items.

btvBrowsingService.getChannelsCount reflects this API but instead of returning the items, it returns the number of items in the result set. Clients can then request full details from pages out of that set as desired.

15.1 tv.quative.service.btvbrowsing.api.ChannelRequest

| Name | Type | Description |
|--------------------------------------|-------------------------------|--|
| <code>context</code> | <code>Context</code> | A <i>Context</i> describing the client's situation. |
| <code>endRow</code> | <code>Long</code> | The index of the last record to return, if paging is required, otherwise null. |
| <code>filter</code> | <code>Filter</code> | A <i>Filter</i> object specifying the criteria <i>Channels</i> should meet. |
| <code>sortOrder</code> | <code>List<Sort></code> | A list of <i>Sort</i> criteria specifying the order to return items in, most significant first. |
| <code>startRow</code> | <code>Long</code> | The index of the first record to return, if paging is required, otherwise null. |
| <code>useSPChannel Number</code> | <code>boolean</code> | <p><code>true</code> to remap channel numbers to suit each <i>Service Provider</i>; <code>false</code> to use the channel numbers of the <i>Channel Entities</i>.</p> <p>Channel number remapping is only available for <i>Channels</i> belonging to the GLOBAL <i>Service Provider</i>.</p> |
| | | <p>Note</p> <p>To be able to remap channel numbers, you must have previously populated the <i>Service Provider</i> channel numbers table. Contact Nagra for details of how to do this.</p> |

15.2 tv.quative.service.btvbrowsing.api.Filter

| Name | Type | Description |
|-------------------------------------|---------------------------|---|
| <code>categoryUid</code> | Long | Only return results where the technical <i>Channel</i> belongs to the <i>Category</i> with the specified <code>uid</code> . |
| <code>channelProfileUidList</code> | List<Long> | Only return results where the technical <i>Channel</i> belongs to a <i>Profile</i> whose <code>uid</code> is in the list. |
| <code>channelType</code> | String | Only return results where the technical <i>Channel</i> has the specified <code>type</code> . |
| <code>disableDeviceFiltering</code> | boolean | <p>If this field is set to <code>false</code>, or it is not set, SDP will use the <i>Context</i> to set an <code>channelProfileUidList</code> filter so that only items suitable for the client device are returned.</p> <p>If this field is set to <code>true</code>, SDP will not filter on the basis of device features.</p> |
| <code>editorialChannelOrigin</code> | OriginSpecification | <p>Only return results where the technical <i>Channel</i> is associated with the editorial <i>Channel</i> with the specified <code>originId</code> and <code>originKey</code>.</p> <p>Tip Because SDP groups technical <i>Channels</i> with the same editorial <i>Channel</i>, using this filter will limit the length of the response to 1 (or 0 if there is no match.)</p> |
| <code>editorialChannelUid</code> | Long | <p>Only return results where the technical <i>Channel</i> is associated with the editorial <i>Channel</i> with the specified <code>uid</code> (through its <code>mainContentUID</code>.)</p> <p>Tip Because SDP groups technical <i>Channels</i> with the same editorial <i>Channel</i>, using this filter will limit the length of the response to 1 (or 0 if there is no match.)</p> |
| <code>packageOriginList</code> | List<OriginSpecification> | Only return results where the technical <i>Channel</i> belongs to the <i>Package</i> with the specified <code>originId</code> and <code>originKey</code> . |

| Name | Type | Description |
|-------------------------------------|----------------------------------|---|
| <code>packageUidList</code> | <code>List<Long></code> | Only return results where the technical <i>Channel</i> belongs to the <i>Package</i> with the specified <code>uid</code> . |
| <code>technicalChannelOrigin</code> | <code>OriginSpecification</code> | Only return results where the technical <i>Channel</i> has the specified <code>originId</code> and <code>originKey</code> . Tip Using this filter will limit the length of the response to 1 (or 0 if there is no match.) |
| <code>technicalChannelUid</code> | <code>Long</code> | Only return results where the technical <i>Channel</i> has the specified <code>uid</code> . Tip Using this filter will limit the length of the response to 1 (or 0 if there is no match.) |

15.3 tv.quative.service.btvbrowsing.api.OriginSpecification

| Name | Type | Description |
|------------------------|---------------------|---|
| <code>originId</code> | <code>int</code> | The ID representing the originating system (1 for the CMS.) |
| <code>originKey</code> | <code>String</code> | The unique ID of the object within the originating system |

15.4 tv.quative.service.btvbrowsing.api.Sort

| Name | Type | Description |
|------------------------------|------|---|
| <code>callLetters</code> | | Sort by the <code>callLetters</code> field of the editorial <i>Channel</i> , in alphabetical order. |
| <code>callLettersDesc</code> | | Sort by the <code>callLetters</code> field of the editorial <i>Channel</i> , in reverse alphabetical order. |

| Name | Type | Description |
|-----------------------------|------|--|
| <code>channelNum</code> | | Sort by the <code>channelNumber</code> field of the editorial <i>Channel</i> , smallest first. |
| <code>channelNumDesc</code> | | Sort by the <code>channelNumber</code> field of the editorial <i>Channel</i> , largest first. |

15.5 btvBrowsingService

BtvBrowsing Service Interface.

Liu, Shu @author

15.5.1 btvBrowsingService.getChannels

Get a list of `ChannelResponseItems` for a given set of context and filters..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/btvBrowsingService/getChannels`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--|---|
| <code>request</code> | <code>tv.quative.service.btvbrowsing.api.ChannelRequest</code> | context and filters. <small>p.247</small> |

Return value

Type: `List<ChannelResponseItem>`

a list of `ChannelResponseItems`

15.5.2 btvBrowsingService.getChannelsCount

Get the number of `ChannelResponseItems` that would be returned by a call to `btvBrowsingService.getChannels` using the same request object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/btvBrowsingService/getChannelsCount`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--|--------------------------------------|
| <code>request</code> | <code>tv.quative.service.btvbrowsing.api.ChannelRequest</code> | The request object. ^{p.247} |

Return value

Type:

a list of ChannelResponseItems

16 BTVService

This *Service* provides aggregate methods for broadcast *Entities*.

It provides

- ▶ `get` methods which retrieve broadcast *Entities* based on an associated channel, package or time range; and
- ▶ methods which work with the mapping between *Service Providers* and *Channels*.

This *Service* does not provide methods for subscribing to broadcast *Entities*. Broadcast subscriptions are handled by the Web Services API provided by *workflowSubscriptionService*.

For a HUE client application to subscribe to broadcast *Channels*, they must be marked *a la carte* and subscribed through the *aLaCarteService*.

16.1 SPChannelMap

SPChannelMaps represent mappings between *Channels* and *Service Providers*.

They are used to allow different *Service Providers* to apply different properties to a single *Channel* provided by the *Global Service Provider*.

Example

Service Providers can use this feature to give a different channel number to a shared channel than is used by the parent *Service Provider*, to fit in with local service expectations.

16.1.1 SPChannelMap

| Name | Type | Description |
|----------------------------|--|---|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>channelNumber</code> | <code>Integer</code> | Channel number that overrides the channel number of the specific channel. |
| <code>channelUID</code> | <code>Long</code> | Channel UID. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |

| Name | Type | Description |
|--------------------------------|-------------------|---|
| <code>originID</code> | Integer | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | Read only. String | |
| <code>originKey</code> | String | The key of the model object in the origin system (3rd part system). |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>uID</code> | Long | The unique identifier. |

16.2 BTVService

Broadcast TV Service Interface. This BTVService is the most important layer in the Broadcast TV Module. It manages all the operations between 3rd part Systems, Set Top Boxes and the Broadcast TV Layer. All the business logic may be implemented in this layer, which deals with ProductService, ChannelService, ScheduleService, PackageService and AccountService. Every method defined in this interface will be used by third part systems to manage or obtain the information or views of the BTV .

jraya @author

Miguel Gil @author

Naeem Bilal @author

16.2.1 BTVService.createSPChannelMap

Creates a `SPChannelMap` entity..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|---------|----------------|
| <code>channelUID</code> | Long | Channel UID |
| <code>channelNumber</code> | Integer | Channel number |
| <code>originUID</code> | Integer | The Origin UID |

| Name | Type | Description |
|------------------------|--------|----------------|
| <code>originKey</code> | String | The Origin Key |

Return value

Type:

UID of the created record

Example

```
<createSPChannelMap>
  <channelUID>87</channelUID>
  <channelNumber>81</channelNumber>
  <originUID>98</originUID>
  <originKey>stringValue</originKey>
</createSPChannelMap>
```

16.2.2 BTVService.deleteSPChannelMap

Deletes an existing `SPChannelMap` entity..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|-----------------------------|
| <code>uid</code> | Long | UID of the record to delete |

Return value

Type:

Number of deleted rows

Example

```
<deleteSPChannelMap>
  <uid>35</uid>
</deleteSPChannelMap>
```

16.2.3 BTVService.getChannelsByCatAndPkgUID

Gets the channels stored in the db for a specific package and category.

This method interacts with the ChannelService to get all the channels for a package and all the channels for a category. It also uses logic business code to merge both subgroups and return the final list.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| pkguid | Long | Integer unique id of the package. |
| catuid | Long | Integer unique id of the category. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |

Return value

Type: List<Channel>

List of Channel objects retrieved

Example

```
<getChannelsByCatAndPkgUID>
  <pkguid>15</pkguid>
  <catuid>9</catuid>
  <locale>en_us</locale>
  <type>stringValue</type>
</getChannelsByCatAndPkgUID>
```

16.2.4 BTVService.getChannelByPkgPattern

Gets the channels for a package pattern (hyphen separated list of package UIDs) and locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getChannelsByPkgPattern`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example

```
<getChannelsByPkgPattern>
<pkgPattern>stringValue</pkgPattern>
<locale>en_us</locale>
</getChannelsByPkgPattern>
```

16.2.5 BTVService.getChannelsByPkgPatternAndCategory

Gets the channels for a package pattern (hyphen separated list of package UIDs) a category and a locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>categoryUID</code> | Long | Category UID |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example


```
<getChannelsByPkgPatternAndCategory>
<pkgPattern>stringValue</pkgPattern>
<categoryUID>95</categoryUID>
<locale>en_us</locale>
</getChannelsByPkgPatternAndCategory>
```

16.2.6 BTVService.getDetEventsByChannelUIDTimeAndType

Gets all the events for a list of channel for a period of time of a specific event type in a particular locale.

There are some special behaviour in specific circumstances: -if type = "SO", we will retrieve "SO" and "SOCU" events because "SOCU" means "SO" and "CU" at the same time. -if type = "CU", we will retrieve "CU" and "SOCU" events because "SOCU" means "SO" and "CU" at the same time. -if type = "SOCU", we will retrieve "CU" and "SO" and "SOCU" events because "SOCU" means "SO", "CU" and "SOCU" at the same time. -otherwise we will retrieve those elements whose event_type = type passed as parameter. This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|--|
| channelUID | String | Channel Identifier. |
| fromDate | Date | |
| toDate | Date | |
| type | String | "SO", "CU", "PPV", "SOCU", "SOCU" means "SO" or "CU" or both at the same time. |
| locale | String | |

Return value

Type:

HashMap of ChannelUID to a List of DetailedEvent. It includes one entry for each channel specified in channelUIDList, the value of that entry could be null that means no events for that channel with the current parameters.

16.2.7 BTVService.getDetEvtByChannelListAndKeyword

This method only for GDC Phase 1 only, it will gets all the events for a list of channel for a period of time of a specific event type in a particular locale.

There are some special behaviour in specific circumstances: -if type = "SO", we will retrieve "SO" and "SOCU" events because "SOCU" means "SO" and "CU" at the same time. -if type = "CU", we will retrieve "CU" and "SOCU" events because "SOCU" means "SO" and "CU" at the same time. -if type = "SOCU", we will retrieve "CU" and "SO" and "SOCU" events because "SOCU" means "SO", "CU" and "SOCU" at the same time. -otherwise we will retrieve those elements whose event_type = type passed as parameter. This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getDetEvtByChannelListAndKeyword`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>chnUidList</code> | String | channel UID list string, split by comma |
| <code>keyword</code> | String | the specific word will be search |
| <code>startDate</code> | Date | |
| <code>endDate</code> | Date | |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String event type |

Return value

Type:

HashMap with channeluid as key and list of events as a value.

16.2.8 BTVService.getDetEvtByKeyword

Gets all the events for a specific keyword, and a time range.

This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getDetEvtByKeyword`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|--|
| <code>keyword</code> | String | the specific word will be search |
| <code>startDate</code> | Date | |
| <code>endDate</code> | Date | |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String event type |

Return value

Type:

HashMap with channeluid as key and list of events as a value.

16.2.9 BTVService.getDetEvtsByCatAndPkgUIDIndexedByChannel

Gets all the events for a specific category, and a time range.

This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|--|
| <code>pkguid</code> | Long | Integer unique id of the package. |
| <code>catuid</code> | Long | unique category uid. |
| <code>startdate</code> | Date | start date of the range of events we want to get. |
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |

| Name | Type | Description |
|-------------------|--------|---------------------|
| <code>type</code> | String | String program type |

Return value

Type:

HashMap with channeluid as key and list of events as a value. Format useful for create associative array in the client app.

16.2.10 BTVService.getDetEvtsByChannelList

Gets all the events for all the channels set in a list of channelsUIDs, separated by comma.

The time range to retrieve events is also considered. This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getDetEvtsByChannelList>

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>chnuidlist</code> | String | List of channel UIDs separated by comma. |
| <code>startdate</code> | Date | start date of the range of events we want to get. |
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type:

List list of Channel objects retrieved for this time range.

16.2.11 BTVService.getDetEvtsByPkgPatternIndexedByChannel

Gets all the events stored in the db for specific time range.

This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getDetEvsByPkgPatternIndexedByChannel`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>startdate</code> | Date | start date of the range of events we want to get. |
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type:

HashMap with channeluid as key and list of events as a value Format useful for create associative array in the client app.

16.2.12 BTVService.getDetEvsByPkgUIDIndexedByChannel

Gets all the events stored in the db for specific time range.

This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|---|
| <code>pkguid</code> | Long | Integer unique id of the package. |
| <code>startdate</code> | Date | start date of the range of events we want to get. |

| Name | Type | Description |
|----------------------|--------|--|
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type:

HashMap with channeluid as key and list of events as a value Format useful for create associative array in the client app.

16.2.13 BTVService.getPackages

Gets the packages the user is subscribed to, without needing to supply the accountUID; it will be retrieved from the Auth Token..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getPackages`

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>recursive</code> | boolean | Should SDP include <i>Packages</i> subscribed as part of a <i>Subscription Profile</i> ? If <code>false</code> , only <i>Packages</i> which have been explicitly subscribed will be returned. |

Return value

Type: `List<Package>`

Unique list of packages

16.2.14 BTVService.getPackagesByAccountUID

Gets a list of distinct broadcast *Packages* which an *Account* has subscribed to.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getPackagesByAccountUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|---------|---|
| accountUID | Long | The uid of the <i>Account</i> . |
| recursive | boolean | Should SDP include <i>Packages</i> subscribed as part of a <i>Subscription Profile</i> ? If false , only <i>Packages</i> which have been explicitly subscribed will be returned. |

Return value

Type: `List<Package>`

16.2.15 BTVService.getPagedChannelsByCatAndPkgUID

Gets the channels stored in the db for a specific package and category.

This method interacts with the ChannelService to get all the channels for a package and all the channels for a category. It also uses logic business code to merge both subgroups and return the final list. If no elements found it returns 'null'.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pkguid | Long | Integer unique id of the package. |
| catuid | Long | Integer unique id of the category. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<Channel>`

List of Channel objects retrieved

Example

```
<getPagedChannelsByCatAndPkgUID>
<pkguid>62</pkguid>
<catuid>70</catuid>
<locale>en_us</locale>
<type>stringValue</type>
<startRowNum>78</startRowNum>
<endRowNum>42</endRowNum>
</getPagedChannelsByCatAndPkgUID>
```

16.2.16 BTVService.getPagedChannelsByPkgPattern

Gets the channels for a package pattern (hyphen separated list of package UIDs) and locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--------|--|
| pkgPattern | String | "-" separated list of package uid-s |
| locale | String | String with the language of the channel we want to retrieve. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example

```
<getPagedChannelsByPkgPattern>
<pkgPattern>stringValue</pkgPattern>
<locale>en_us</locale>
<startRowNum>63</startRowNum>
<endRowNum>2</endRowNum>
</getPagedChannelsByPkgPattern>
```


16.2.17 BTVService.getPagedChannelsByPkgPatternAndCategory

Gets the channels for a package pattern (hyphen separated list of package UIDs) a category and a locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>categoryUID</code> | Long | Category UID |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example

```
<getPagedChannelsByPkgPatternAndCategory>
<pkgPattern>stringValue</pkgPattern>
<categoryUID>43</categoryUID>
<locale>en_us</locale>
<startRowNum>91</startRowNum>
<endRowNum>17</endRowNum>
</getPagedChannelsByPkgPatternAndCategory>
```

16.2.18 BTVService.getPagedDetEventsByChannelUIDTimeAndType

Gets all the events for a list of channel for a period of time of a specific event type in a particular locale.

There are some special behaviour in specific circumstances: -if type = "SO", we will retrieve "SO" and "SOCU" events because "SOCU" means "SO" and "CU" at the same time. -if type = "CU", we will retrieve "CU" and "SOCU" events because "SOCU" means "SO" and "CU" at the same time. -if type = "SOCU", we will retrieve "CU" and "SO" and "SOCU" events because "SOCU" means "SO", "CU" and "SOCU" at the same time. -otherwise we will retrieve those elements whose event_type = type passed as parameter. This method interacts with the EventService to get all the event data,

and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getPagedDetEventsByChannelUIDTimeAndType`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--------|--|
| <code>channelUIDList</code> | String | Channel Identifier. |
| <code>fromDate</code> | Date | |
| <code>toDate</code> | Date | |
| <code>type</code> | String | "SO", "CU", "PPV", "SOCU", "SOCU" means "SO" or "CU" or both at the same time. |
| <code>locale</code> | String | |
| <code>startRowNum</code> | Long | |
| <code>endRowNum</code> | Long | |

Return value

Type:

HashMap of ChannelUID to a List of DetailedEvent. It includes one entry for each channel specified in channelUIDList, the value of that entry could be null that means no events for that channel with the current parameters.

16.2.19

BTVService.getPagedDetEvtsByCatAndPkgUIDIndexedByChannel

Gets all the events for a specific category, and a time range.

This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pkguid</code> | Long | Integer unique id of the package. |
| <code>catuid</code> | Long | unique category uid. |
| <code>startdate</code> | Date | start date of the range of events we want to get. |
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type:

List list of Channel objects retrieved for this time range.

16.2.20 BTVService.getPagedDetEvtsByChannelList

Gets all the events for all the channels set in a list of channelsUIDs, separated by comma.

The time range to retrieve events is also considered. This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>chnuidlist</code> | String | List of channel UIDs separated by comma. |
| <code>startdate</code> | Date | start date of the range of events we want to get. |

| Name | Type | Description |
|--------------------------|--------|--|
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type:

List list of Channel objects retrieved for this time range.

16.2.21 BTVService.getPagedDetEvtsByPkgUIDIndexedByChannel

Gets all the events stored in the db for specific time range.

This method interacts with the EventService to get all the event data, and re-format it in a list of Channels. Each one includes a list of all the events for this channel and times. If no elements found it returns 'null'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|--|
| <code>pkguid</code> | Long | Integer unique id of the package. |
| <code>startdate</code> | Date | start date of the range of events we want to get. |
| <code>enddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

| Name | Type | Description |
|--------------------------|------|-------------|
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type:

List list of Channel objects retrieved for this time range.

16.2.22 BTVService.getPagedSOCUEventsByIdentifierTimeAndType

Gets a page from the list of *Events* of a particular type being shown on the specified *Channels* during the specified time period.

`soctv` means "both start-over and catch-up".

If you search for `so` or `ctv`, SDP will return *Events* with a `type` of `soctv` as well as those with the type you specified.

If you search for `soctv`, SDP will return *Events* with a `type` of `so`, `ctv` or `soctv`.

Events are returned in a map whose keys represent each requested *Channel*. The value for each key is a list of matching *Events*, or null if no *Events* on that *Channel* matched the criteria.

>

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getPagedSOCUEventsByIdentifierTimeAndType`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|--|
| <code>idCsvList</code> | String | The <code>carryingNetwork</code> values of the <i>Channels</i> to investigate, separated by commas , |
| <code>fromDate</code> | Date | The start date and time of the time period to investigate. |
| <code>toDate</code> | Date | The end date and time of the time period to investigate. |
| <code>type</code> | String | The <code>type</code> of <i>Events</i> to look for. One of <code>so</code> (start-over events only), <code>ctv</code> (catch-up events only), or <code>soctv</code> (both start-over and catch-up events.) |

| Name | Type | Description |
|--------------------------|--------|--|
| <code>locale</code> | String | The locale to use for localizable fields of the returned <i>Events</i> . |
| <code>startRowNum</code> | Long | The index of the first <i>Event</i> to return. |
| <code>endRowNum</code> | Long | The index of the last <i>Event</i> to return. |

Return value

Type:

parameters.

16.2.23 BTVService.getPagedSPChannelsByPkgPattern

Gets the channels for a package pattern (hyphen separated list of package UIDs) and locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium. It takes care of the overriding channel numbers, if there is an overriding channel channel number for a given service provider and channel uid then it will be used instead of using the channel number of the channel itself (this is to allow different channel numbers for different service providers).

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getPagedSPChannelsByPkgPattern`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example

```
<getPagedSPChannelsByPkgPattern>
<pkgPattern>stringValue</pkgPattern>
<locale>en_us</locale>
<startRowNum>35</startRowNum>
<endRowNum>58</endRowNum>
</getPagedSPChannelsByPkgPattern>
```

16.2.24 BTVService.getPagedSPChannelsByPkgPatternAndCategory

Gets the channels for a package pattern (hyphen separated list of package UIDs) a category and a locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium. The method takes care of the overriding channel numbers, if there is an overriding channel number for a given service provider and channel uid then it will be used instead of using the channel number of the channel itself (this is to allow different channel numbers for different service providers).

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>categoryUID</code> | Long | Category UID |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example

```
<getPagedSPChannelsByPkgPatternAndCategory>
<pkgPattern>stringValue</pkgPattern>
<categoryUID>20</categoryUID>
<locale>en_us</locale>
<startRowNum>20</startRowNum>
<endRowNum>28</endRowNum>
```

</getPagedSPChannelsByPkgPatternAndCategory>

16.2.25 BTVService.getSOCUEventsByIdentifierTimeAndType

Gets the *Events* of a particular type being shown on the specified *Channels* during the specified time period.

soctū means "both start-over and catch-up".

If you search for so or ctū, SDP will return *Events* with a *type* of soctū as well as those with the type you specified.

If you search for soctū, SDP will return *Events* with a *type* of so, ctū or soctū.

Events are returned in a map whose keys represent each requested *Channel*. The value for each key is a list of matching *Events*, or null if no *Events* on that *Channel* matched the criteria.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getSOCUEventsByIdentifierTimeAndType>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|---|
| idCsvList | String | The carryingNetwork values of the <i>Channels</i> to investigate, separated by commas , |
| fromDate | Date | The start date and time of the time period to investigate. |
| toDate | Date | The end date and time of the time period to investigate. |
| type | String | The <i>type</i> of <i>Events</i> to look for. One of so (start-over events only), ctū (catch-up events only), or soctū (both start-over and catch-up events.) |
| locale | String | The locale to use for localizable fields of the returned <i>Events</i> . |

Return value

Type:

16.2.26 BTVService.getSPChannelList

Gets the list of channel maps for the given service provider found in the request context..

Available through Web Services.

Parameters

None.

Return value

Type: `List<SPChannelMap>`

List of service provider channel maps

Example

```
<getSPChannelList/>
```

16.2.27 BTVService.getSPChannelsByChannelUID

Gets the channel map for the passed channel UID and for the service provider found in the request context..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|-------------------------|
| <code>channelUID</code> | Long | Channel UID to look for |

Return value

Type:

ServiceProvider Channel Map if found, `<code>null</code>`

 otherwise

Example

```
<getSPChannelsByChannelUID>
<channelUID>8</channelUID>
</getSPChannelsByChannelUID>
```

16.2.28 BTVService.getSPChannelsByPkgPattern

Gets the distinct set of *Channels* which are provided by the specified set of *Packages*.

A *MappedChannel* object has the same fields as a *Channel* object, but for two differences:

- ▶ The **number** field represents the channel number in use by the current *Service Provider*.
- ▶ The **premium** field represents whether the *Channel* has been subscribed as a premium *Channel* within the subscribed *Package*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/BTVService/getSPChannelsByPkgPattern`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pkgPattern</code> | String | The <code>uid</code> values of the <i>Packages</i> to investigate, separated by hyphen - |
| <code>locale</code> | String | The locale to use when returning localizable fields of the <i>Channels</i> . |

Return value

Type: `List<MappedChannel>`

A list of distinct *Channels*, formatted as *MappedChannel* objects.

Example

```
<getSPChannelsByPkgPattern>
  <pkgPattern>stringValue</pkgPattern>
  <locale>en_us</locale>
</getSPChannelsByPkgPattern>
```

16.2.29 BTVService.getSPChannelsByPkgPatternAndCategory

Gets the channels for a package pattern (hyphen separated list of package UIDs) a category and a locale.

Duplicate channels are filtered, if a channel found as premium as well as not premium the list will contain the one that is not premium. It takes care of the overriding channel numbers, if there is an overriding channel channel number for a given service provider and channel uid then it will be used instead of using the channel number of the channel itself (this is to allow different channel numbers for different service providers).

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pkgPattern</code> | String | "-" separated list of package uid-s |
| <code>categoryUID</code> | Long | Category UID |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects retrieved

Example

```
<getSPChannelsByPkgPatternAndCategory>
  <pkgPattern>stringValue</pkgPattern>
  <categoryUID>29</categoryUID>
  <locale>en_us</locale>
</getSPChannelsByPkgPatternAndCategory>
```

16.2.30 BTVService.getSPChannelsByPkgUID

Gets the Channel list using the uid of the package as the key.

It takes care of the overriding channel numbers, if there is an overriding channel channel number for a given service provider and channel uid then it will be used instead of using the channel number of the channel itself (this is to allow different channel numbers for different service providers).

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>pkguid</code> | Long | Long unique id of the package |
| <code>locale</code> | String | String with the language and locale specification. |
| <code>type</code> | String | String type of the channel |

Return value

Type: `List<Channel>`

List of Channel objects with the information of the channels found

16.2.31 BTVService.subscribeToChannel

Subscribe to a channel..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|--------|---|
| accountUID | Long | Account UID of the end user |
| userID | Long | User UID of the end user (if available) |
| channelUID | Long | UID of the channel to subscribe to |
| policyGroupUID | Long | PolicyGroup UID |
| smartcardID | String | Smartcard ID assigned to the end user |
| locale | String | Locale |

Return value

Type:

UID of the created ACL entry

Example

```
<subscribeToChannel>
<accountUID>69</accountUID>
<userID>36</userID>
<channelUID>33</channelUID>
<policyGroupUID>62</policyGroupUID>
<smartcardID>ABC-14661</smartcardID>
<locale>en_us</locale>
</subscribeToChannel>
```

16.2.32 BTVService.subscribeToPolicyGroup

Subscribe to an policy group.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|-------------------------|
| accountuid | Long | The Account primary key |

| Name | Type | Description |
|---------------------------|--------|--|
| <code>policygrpuid</code> | Long | The Policy Group primary key |
| <code>userid</code> | Long | The User primary key |
| <code>smartcardID</code> | String | ID of the users smartcard, if there is any |
| <code>locale</code> | String | Locale |

Return value

Type: `List<Long>`

UID of the created ACL

Example

```
<subscribeToPolicyGroup>
<accountuid>11</accountuid>
<policygrpuid>75</policygrpuid>
<userid>4</userid>
<smartcardID>ABC-63959</smartcardID>
<locale>en_us</locale>
</subscribeToPolicyGroup>
```

16.2.33 BTVService.unsubscribeChannel

Unsubscribes from a channel..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---------------------------------------|
| <code>accountUID</code> | Long | Account UID of the end user |
| <code>channelUID</code> | Long | UID of the channel to subscribe to |
| <code>smartcardID</code> | String | Smartcard ID assigned to the end user |
| <code>locale</code> | String | Locale |

Return value

Type:

Number of updated rows

Example

```
<unsubscribeChannel>
<accountUID>92</accountUID>
<channelUID>48</channelUID>
<smartcardID>ABC-42964</smartcardID>
<locale>en_us</locale>
</unsubscribeChannel>
```

16.2.34 BTVService.unsubscribePackage

Unsubscribes from a package.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---------------------------------------|
| accountUID | Long | Account UID of the end user |
| packageUID | Long | UID of the channel to subscribe to |
| smartcardID | String | Smartcard ID assigned to the end user |
| locale | String | Locale |

Return value

Type:

Number of updated rows

Example

```
<unsubscribePackage>
<accountUID>41</accountUID>
<packageUID>33</packageUID>
<smartcardID>ABC-76999</smartcardID>
<locale>en_us</locale>
</unsubscribePackage>
```

16.2.35 BTVService.updateSPChannelMap

Updates an existing `SPChannelMap` entity..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|---------|----------------------------|
| <code>uid</code> | Long | UID of the existing recorx |
| <code>channelUID</code> | Long | Channel UID |
| <code>channelNumber</code> | Integer | Channel number |
| <code>modifiedDate</code> | Date | Data of modification |

Return value

Type:

Number of updated rows

17 catalogueService

This is a primitive *Service* which provides the COD *Catalogue* Entity.

It manages relationships between *Catalogues* and their constituent *Catalogue Nodes*—*Assets*, *Asset Groups* and other *Catalogues*.

17.1 Catalogue

A *Catalogue* represents a collection of on-demand items to be presented in a single menu within a catalogue tree.

Figure 6 Catalogue and associated Entities.

Catalogues may contain sub-*Catalogues* to form the tree structure. More than one independent tree may exist. *Catalogues* at the top level are known as root *Catalogues*.

Client applications are expected to display each *Catalogue* in the form of a menu of choices, with child *Catalogues* representing submenus. If you want to be able to display different *Catalogue* structures to different subscribers, you have two choices:

- ▶ To differentiate between subscribers in different geographical regions, you can assign *Access Points* to root *Catalogues*. Clients can then discover the root catalogue by calling `catalogueService.getRootCataloguesByAccept` *Uid*.

You can provide multiple root catalogues for the same *Access Point*, but in this case the client must be able to decide which one it should use.

- ▶ To differentiate between subscribers on an individual basis, you can add *Catalogues* to their subscription, just like *Packages* or *Policy Groups*. Clients can discover which root catalogues have been subscribed by calling `catalogueService.getSubscribedBaseCatalogues`.

You can add multiple root catalogues to a subscription, but again the client must be able to decide which one it should use.

A subscription to a *Catalogue* only means the subscriber can browse that *Catalogue*. To get access to the *Assets* inside that *Catalogue*, they need to purchase *Policy Groups* in the usual way.

Tip

This is in addition to your ability to provide targeted content and offers inside the *Catalogues*, where you can differentiate subscribers either geographically (by assigning *Access Points* to *Policy Groups*) or by the device they are using (by assigning *Product Profiles* to *Policy Groups* and *Asset Profiles* to *Assets*.)

On-demand content items are added to a *Catalogue* as *Catalogue* entries. A *Catalogue* entry combines three *Entities*:

- ▶ an *Asset*, which provides information about the content;
- ▶ an optional *Asset Group*, if the *Asset* is part of a series; and
- ▶ a *Schedule*, which determines when the item will be offered and identifies ways to purchase it (through its *Policy Groups*.)

Caution!

In systems using a CMS, *Catalogues* are created automatically by the ingest process. Broadcast content and pricing should be defined in the CMS.

The CMS ingest creates *Catalogues* from the CMS *Catalogue* and *Catalogue Node* objects, and assigns them to *Access Points* on the basis of the **Regions** EPG attribute.

Catalogues may be associated with *Assets* and *Asset Groups*, but these relationships are deprecated.

17.1.1 Catalogue

| Name | Type | Description |
|-------------------------------|--|--|
| <code>barkerChannelRef</code> | <code>String</code> | The URL of a stream which displays promotional content for the <i>Catalogue</i> . The CMS ingest sets this field from a <i>Node</i> object's Barker channel URI EPG attribute. |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | A log of the changes to this <i>Entity</i> history. This field is managed by SDP and is not ingested from the CMS. |
| <code>creationDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was created in SDP. This field is managed by SDP and is not ingested from the CMS. |
| <code>description</code> | <code>String</code> | A short description of this <i>Entity</i> . |
| <code>entries</code> | <code>List<CatalogueTree Node></code> | This field is deprecated and should not be used. |
| <code>exportID</code> | <code>Integer</code> | The identifier to use when exporting this <i>Entity</i> to third-party systems. This field is managed by SDP and is not ingested from the CMS. |
| <code>locale</code> | <code>String</code> | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>logoImage</code> | <code>String</code> | The URL of a logo for the <i>Catalogue</i> . |

Caution!

This field has no equivalent in the CMS and is not ingested. Deployments wishing to

| Name | Type | Description |
|-----------------------------|--|--|
| | | use it must populate it through some other means. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. This field is managed by SDP and is not ingested from the CMS. |
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. This field is managed by SDP and is not ingested from the CMS. |
| <code>name</code> | Search criterion. Quick Search criterion. String | The full name of this <i>Catalogue</i> . The CMS ingest places the <i>Catalogue</i> object's Title or its Title EPG attribute in this field. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. This field is set to 1 for all <i>Entities</i> ingested from NAGRA Media CMS. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. For all <i>Entities</i> ingested from NAGRA Media CMS, this field is set to the <code>id</code> of the object. |
| <code>ownerID</code> | String | The owner of the <i>Catalogue</i> . The CMS ingest sets this field from a <i>Node</i> object's companyId attribute. |
| <code>pVPPurchase</code> | Read only. boolean | This field is always false for <i>Catalogues</i> . |

| Name | Type | Description |
|----------------------|-------------------------|---|
| parentAclUID | Long | This field should not be used. |
| parentPolicyGroupUID | Long | This field should not be used. |
| parentalRating | Long | <p>The UID of the <i>Rating</i> for this <i>Catalogue</i>; normally the most inclusive <i>Rating</i> which is still appropriate for everything in the <i>Catalogue</i> (though this is up to the operator.)</p> <p>The CMS ingest uses the <i>Node</i> object's Rating EPG attribute to determine a suitable value for this field.</p> |
| policy | <i>Deprecated.</i> Long | This field is deprecated and should not be used. |
| policyGroupUID | Long | <p>The <i>Policy Group</i> for this <i>Catalogue</i>.</p> <p>Caution! The CMS does not support purchasable <i>Catalogues</i> so this field is not populated by the CMS ingest. Deployments wishing to use it must populate it through some other means.</p> |
| privateMetadata | String | <p>This field is provided for operators to use for their own purposes. SDP passes it to the client unchanged. Its format (string, JSON object...) and meaning should be decided by the operator.</p> <p>The CMS ingest places the <i>Node</i> object's Private Metadata EPG attribute in this field.</p> |
| profileUID | Long | This field should not be used. |
| promoUrl | String | <p>The URL or filename of a promotional image for the <i>Catalogue</i>, or a semicolon-separated list of URLs or filenames. The operator is responsible for ensuring that an image or other supporting media is available at this location.</p> <p>The CMS ingest sets this field from the <i>Catalogue</i> or <i>Node</i> object's PromoImages EPG attribute.</p> |
| serviceProviderID | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |

| Name | Type | Description | | | | | | | | | | | | |
|-----------------------|--|--|-------|-------------|-----------|--|-----------|---------------|-----------|-------------------------|-----------|---------------|-----------|--------------------|
| | | For <i>Entities</i> ingested from NAGRA Media CMS, this is the <code>uid</code> of the <i>Service Provider</i> whose <code>originKey</code> matches the <code>id</code> found on the ingest file's root. CMSvod Data element. | | | | | | | | | | | | |
| <code>skinName</code> | String | <p>The skin or interface type to use when presenting the <i>Catalogue</i>.</p> <p>The CMS ingest sets this field from the <i>Node</i> object's Background color EPG attribute.</p> <p>Caution! This field has no equivalent in the CMS and is not ingested. Deployments wishing to use it must populate it through some other means.</p> | | | | | | | | | | | | |
| <code>sortMask</code> | Integer | <p>A bit field specifying the sorting types clients may offer for items in this <i>Catalogue</i>.</p> <p>The available sorting types are:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Bit 0 set</td> <td>No sorting (retain the supplied order)</td> </tr> <tr> <td>Bit 1 set</td> <td>Sort by genre</td> </tr> <tr> <td>Bit 2 set</td> <td>Sort by parental rating</td> </tr> <tr> <td>Bit 3 set</td> <td>Sort by title</td> </tr> <tr> <td>Bit 4 set</td> <td>Sort by popularity</td> </tr> </tbody> </table> <p>This field is only used for client-side sorting. Clients using this field are expected to implement their own sorting to the operator's requirements.</p> | Value | Description | Bit 0 set | No sorting (retain the supplied order) | Bit 1 set | Sort by genre | Bit 2 set | Sort by parental rating | Bit 3 set | Sort by title | Bit 4 set | Sort by popularity |
| Value | Description | | | | | | | | | | | | | |
| Bit 0 set | No sorting (retain the supplied order) | | | | | | | | | | | | | |
| Bit 1 set | Sort by genre | | | | | | | | | | | | | |
| Bit 2 set | Sort by parental rating | | | | | | | | | | | | | |
| Bit 3 set | Sort by title | | | | | | | | | | | | | |
| Bit 4 set | Sort by popularity | | | | | | | | | | | | | |

| Name | Type | Description |
|---------------------------------------|--------------------|--|
| | | <p>Note Operators developing their own clients may define their own meanings for the bits in this field.</p> <p>Caution! This field has no equivalent in the CMS and is not ingested. Deployments wishing to use it must populate it through some other means.</p> |
| <code>subscribableItem</code> Type | String | This field is always CAT for <i>Catalogues</i> . |
| <code>subscribed</code> | Read only. boolean | <p>Is the user already subscribed to this item?</p> <p>This field is returned as part of the <i>Catalogue</i> model object when clients query <i>Catalogues</i>. It is not part of the <i>Catalogue</i> Entity itself.</p> <p>Note Attempting to alter this field will have no effect. To subscribe to a <i>Catalogue</i>, use the dedicated subscription APIs or the <i>Modify Subscription</i> workflow in QSP Manager.</p> |
| <code>uID</code> | Long | <p>The SDP internal UID for this <i>Entity</i>.</p> <p>This field is managed by SDP and is not ingested from the CMS.</p> |

17.2 CatalogueTreeNode

A *CatalogueTreeNode* represents a position in the catalogue tree.

This object is not intended for use by external systems. Clients should use the *catalogueService* to work with *Catalogues* and their structure.

17.2.1 CatalogueTreeNode

| Name | Type | Description | | | | | | | | |
|----------------------------|--|--|-------|-------------|------------------|-----------------------|------------------|-----------------------------|------------------|------------------|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. | | | | | | | | |
| <code>creationDate</code> | <code>Date</code> | The date created. | | | | | | | | |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. | | | | | | | | |
| <code>internalOrder</code> | <code>Integer</code> | Order the node should appear in relation to its siblings. Nodes with a lower value in this field should appear before nodes with a higher value. Note This ordering may be overridden by the client, either by sorting items according to its own rules or by requesting a different sort order from SDP. | | | | | | | | |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. | | | | | | | | |
| <code>nodeModel</code> | <code>LocaleBaseModel</code> | Locale information belonging to the node. | | | | | | | | |
| <code>nodeSpid</code> | <code>Long</code> | SPID of the item which is present at this position in the tree, based on the type. | | | | | | | | |
| <code>nodeType</code> | <code>String</code> | The type of item which is present at this position in the tree. One of: <table border="1" data-bbox="794 1417 1401 1720"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><code>AST</code></td> <td><i>Asset (legacy)</i></td> </tr> <tr> <td><code>AGR</code></td> <td><i>Asset Group (legacy)</i></td> </tr> <tr> <td><code>CAT</code></td> <td><i>Catalogue</i></td> </tr> </tbody> </table> | Value | Description | <code>AST</code> | <i>Asset (legacy)</i> | <code>AGR</code> | <i>Asset Group (legacy)</i> | <code>CAT</code> | <i>Catalogue</i> |
| Value | Description | | | | | | | | | |
| <code>AST</code> | <i>Asset (legacy)</i> | | | | | | | | | |
| <code>AGR</code> | <i>Asset Group (legacy)</i> | | | | | | | | | |
| <code>CAT</code> | <i>Catalogue</i> | | | | | | | | | |

| Name | Type | Description |
|--------------------------------|---------------------------------------|---|
| <code>nodeUId</code> | <code>Long</code> | The UID of the item which is present at this position in the tree. |
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object derived by concatenating the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |
| <code>serviceProviderID</code> | <code>Long</code> | The Service provider UID. |
| <code>treeLevel</code> | <code>Integer</code> | The level of the node in the tree. 0 represents a root node with no parent. |
| <code>uID</code> | <code>Long</code> | The unique identifier of the node. |

17.3 catalogueService

The services are made visible to Hue or to WebServices using annotations in the Service Implementation class. Ex: For the ServiceClass:

= "assetService", hue = true, ws = true) For the Methods: `@QubeService(name`

`ws=true)` The methods used to create, delete or update items of the COD System, only could be accesible to Web Services (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel would be accessible on both the WebServices and the Hue interfaces. `@QubeMethod(hue=false,`

Zafar Ali `@author`

Naeem Bilal `@author`

Matthew Willard `@author`

1.0 `@version`

17.3.1 catalogueService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

17.3.2 catalogueService.addNodeToCatalogue

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|---------|-------------|
| <code>catalogueUid</code> | Long | |
| <code>internalOrder</code> | Integer | |
| <code>treeLevel</code> | Integer | |
| <code>nodeUid</code> | Long | |
| <code>nodeType</code> | String | |

Return value

Type:

Integer

Example

```
<addNodeToCatalogue>
  <catalogueUid>66</catalogueUid>
  <internalOrder>27</internalOrder>
  <treeLevel>46</treeLevel>
  <nodeUid>35</nodeUid>
  <nodeType>stringValue</nodeType>
</addNodeToCatalogue>
```


17.3.3 catalogueService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to persist. |

Return value

Type:

UID of the created object.

17.3.4 catalogueService.delete

Delete, from storage, a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows deleted.

17.3.5 catalogueService.getAllCatalogues

Parameters

None.

Return value

Type: `List<Catalogue>`

List of all catalogues

17.3.6 catalogueService.getBaseCataloguesByAcptUid

Gets all *Catalogues* with no parent *Catalogue* which are associated with the specified *Access Point*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/catalogueService/getBaseCataloguesByAcptUid`

Parameters

| Name | Type | Description |
|-----------------------------|--------|--|
| <code>accessPointUid</code> | Long | The <code>uID</code> of the <i>Access Point</i> |
| <code>locale</code> | String | The locale to use for the localizable fields of the returned <i>Catalogues</i> . |

Return value

Type: `List<Catalogue>`

A list of matching *Catalogues* (may be empty.)

17.3.7 catalogueService.getBaseChildCataloguesByAcptUidAndName

Returns a List of *Catalogue* objects which represent the immediate child nodes of the node which matches the specified catalogue name.

For example, if the specified name matches a node at the first level in the catalogue tree then the returned list will contain *Catalogue* objects for all of its immediate child nodes. The catalogue name is intended to be that of a device type, eg: "PC" or "STB" for which the immediate child catalogue nodes are of interest.

CatalogueServiceErrorCodes @see

SearchableService#MIN_ROW_VALUE @see

SearchableService#MAX_ROW_VALUE @see

LocaleUtil#getDefaultLocale() @see

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/catalogueService/getBaseChildCataloguesByAcptUidAndName`

Parameters

| Name | Type | Description |
|-----------------------------|--------|---|
| <code>accessPointUid</code> | Long | The access point UID. This argument must not be null. |
| <code>catalogueName</code> | String | The name of the catalogue to look for in the catalogue tree. This name must be unique in the catalogue tree. This argument must not be null. |
| <code>locale</code> | String | Lower-case two-letter codes as defined by ISO-639 underscore (_) lower-case two-letter codes as defined by ISO-3166. If this argument is null then the system default of <code>LocaleUtil.getDefaultLocale()</code> will be used. |
| <code>startRowNumber</code> | Long | If this argument is null then <code>SearchableService.MIN_ROW_VALUE</code> will be used. |
| <code>endRowNumber</code> | Long | If this argument is null then <code>SearchableService.MIN_ROW_VALUE</code> will be used. |

Return value

Type: `List<Catalogue>`

Catalogue list. These objects represent the nodes which are direct children of the node which matches the specified catalogue name. If no such node exists in the catalogue tree then an empty list will be returned.

17.3.8 catalogueService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/catalogueService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

| Name | Type | Description |
|--------|--------|-----------------------------------|
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>87</originID>
  <locale>en_us</locale>
</getByOrigin>
```

17.3.9 catalogueService.getByUID

Gets the *Catalogue* with the specified *uid*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/catalogueService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| uid | Long | The <i>uid</i> of the <i>Catalogue</i> . |
| locale | String | The locale to use for the localizable fields of the <i>Catalogue</i> . |

Return value

Type:

The matching *Catalogue*, or null if none is found.

Example

```
<getByUID>
  <uid>6</uid>
  <locale>en_us</locale>
</getByUID>
```

17.3.10 catalogueService.getCatBySbscrProfUID

Returns a list of catalogues belonging to the given profile..

zlaposa @author

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|--------------|
| profileUID | Long | Profile UID. |
| locale | String | The locale. |

Return value

Type: List<Catalogue>

List List of features belonging to the given profile.

Example

```
<getCatBySbscrProfUID>
  <profileUID>71</profileUID>
  <locale>en_us</locale>
</getCatBySbscrProfUID>
```

17.3.11 catalogueService.getCatalogueTreeLevel

Gets the Tree Level of the Catalogue.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|------|--------------------------------|
| catalogueUid | Long | The uid to find the catalogue. |

Return value

Type:

treeLevel The tree level.

Example

```
<getCatalogueTreeLevel>
  <catalogueUid>67</catalogueUid>
```

</getCatalogueTreeLevel>

17.3.12 catalogueService.getCataloguesByCatalogueUid

Gets the list of catalogues belonging to the specified catalog..

zlaposa @author

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--------|---------------------------|
| catalogueUID | Long | The catalogue to look for |
| locale | String | Locale |

Return value

Type: List<Catalogue>

List of catalogues belonging to the specified catalogue

Example

```
<getCataloguesByCatalogueUid>
  <catalogueUID>62</catalogueUID>
  <locale>en_us</locale>
</getCataloguesByCatalogueUid>
```

17.3.13 catalogueService.getCataloguesByNodeUidAndType

Retrieves the list of catalogues that contain the given node..

zlaposa @author

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|--------|---|
| nodeUID | Long | The UID of the node to look for |
| nodeType | String | Type of the node (asset/ asset group/catalogue) |

| Name | Type | Description |
|--------|--------|-------------|
| locale | String | Locale |

Return value

Type: `List<Catalogue>`

List List of catalogues that contain the node

Example

```
<getCataloguesByNodeUidAndType>
  <nodeUID>50</nodeUID>
  <nodeType>stringValue</nodeType>
  <locale>en_us</locale>
</getCataloguesByNodeUidAndType>
```

17.3.14 catalogueService.getList

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/catalogueService/getList`

Parameters

| Name | Type | Description |
|--------|--------|-------------|
| locale | String | |

Return value

Type: `List<Catalogue>`

List of catalogues which match the locale.

17.3.15 catalogueService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: `List<M>`

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>34</uid>
</getMultiLanguageList>
```

17.3.16 catalogueService.getParentCatalogue

Gets the parent catalogue for given catalogue.

Parameters

| Name | Type | Description |
|--------------|------|---------------------------------|
| catalogueUid | Long | The uid to find the parent for. |
| accessPUid | Long | The uid for the access point. |

Return value

Type:

The parent catalogue.

17.3.17 catalogueService.getSubscribedBaseCatalogues

Gets all *Catalogues* with no parent *Catalogue* which are subscribed by the currently signed-on *Account*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/catalogueService/getSubscribedBaseCatalogues`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|--|
| catLocale | String | The locale to use for the localizable fields of the returned <i>Catalogues</i> . |

Return value

Type: `List<Catalogue>`

A list of matching *Catalogues* (may be empty.)

Example

```
<getSubscribedBaseCatalogues>
  <catLocale>stringValue</catLocale>
</getSubscribedBaseCatalogues>
```

17.3.18 catalogueService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

17.3.19 catalogueService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

17.3.20 catalogueService.removeNodeFromCatalogue

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|---------|-------------|
| catalogueUid | Long | |
| internalOrder | Integer | |
| nodeUid | Long | |
| nodeType | String | |

Return value

Type:

Integer

Example

```
<removeNodeFromCatalogue>
  <catalogueUid>5</catalogueUid>
  <internalOrder>40</internalOrder>
  <nodeUid>94</nodeUid>
  <nodeType>stringValue</nodeType>
</removeNodeFromCatalogue>
```

17.3.21 catalogueService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

17.3.22 catalogueService.updateCatalogueTreeLevel

Updates the Tree Level of Parent Catalogue.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|------|---------------------------------|
| <code>catalogueUid</code> | Long | The uid to find the parent for. |
| <code>treeLevel</code> | Long | The tree level. |

Return value

Type:

`treeLevel` The tree level.

Example

```
<updateCatalogueTreeLevel>
  <catalogueUid>96</catalogueUid>
  <treeLevel>88</treeLevel>
</updateCatalogueTreeLevel>
```

17.3.23 catalogueService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

17.3.24 catalogueService.updateOrder

Updates the internal order of a catalogue..

zlaposa @author

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|---------|---------------------------|
| <code>catalogueUID</code> | Long | The catalogue to look for |
| <code>oldOrder</code> | Integer | Old internal order |
| <code>newOrder</code> | Integer | New internal order |

Return value

Type:

List Number of updated rows

Example

```
<updateOrder>
  <catalogueUID>57</catalogueUID>
  <oldOrder>95</oldOrder>
  <newOrder>73</newOrder>
</updateOrder>
```

18 categoryService

This is a primitive *Service* which provides the BTV *Category* Entity.

It manages relationships between *Categories* and their constituent *Channels*.

18.1 Category

A (broadcast) *Category* represents a group of *Channels* related by theme or genre.

Figure 7 Category and associated Entities.

Categories should not be confused with *COD Categories*, which perform a similar function for Content On Demand (COD) services.

A *Channel* can be a member of several *Categories*.

Channels are grouped into *Categories* for ease of navigation only. Users subscribe to *Channels* as part of a *Package*.

18.1.1 Category

| Name | Type | Description |
|-----------------------------|---|--|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE. List<String></i> | A log of the changes to this <i>Entity</i> history. |
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created. |
| <code>description</code> | String | A short description of this <i>Entity</i> . |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>locale</code> | String | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |

| Name | Type | Description |
|--------------------------------|--|---|
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> String | The name of the <i>Category</i> . |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>promoImage</code> | String | The URL of a promotional image for the <i>Category</i> . Caution! The CMS does not support metadata for <i>Categories</i> so this field is not ingested. Deployments wishing to use it must populate it through some other means. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |

18.2 categoryService

Category Service Implementation. Manages the business logic of all the operations for Categories A Category is a group of Channels with a common theme or genre. It's a logic group basically oriented to show a channel classification in the client site.

Steve Roberts & Juanjo Raya @author

18.2.1 categoryService.addChannel

Adds a channel to a Category group..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/addChannel`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|-----------------------------|
| channeluid | Long | The Channel uid to be added |
| categoryuid | Long | The parent Category uid |

Return value

Type:

The number of rows added.

Example

```
<addChannel>
  <channeluid>56</channeluid>
  <categoryuid>20</categoryuid>
</addChannel>
```

18.2.2 categoryService.addMultiLanguage

Contains the business logic to create a new category description.

Receives all the information as a parameter values. This methods internally calls `toaddMultiLanguage(LocaleBase Model)`

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/addMultiLanguage`

Parameters

| Name | Type | Description |
|-------------|--------|--|
| uid | Long | unique id of the category to create multilanguage |
| name | String | String Name of the category |
| description | String | String description of the category |
| locale | String | String with the language and locale specification. |

Return value

Type:

Integer id for the new channel

18.2.3 categoryService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

18.2.4 categoryService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

18.2.5 categoryService.delete

Delete, from storage, a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

18.2.6 categoryService.getAllCategories

Gets all the categories we have got for an specific locale..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/getAllCategories`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>locale</code> | String | Optional language identifier, null assumes the default language |

Return value

Type: `List<Category>`

List If found, return all Category entities, null otherwise.

Example

```
<getAllCategories>
  <locale>en_us</locale>
</getAllCategories>
```

18.2.7 categoryService.getByOriginUID

Gets the list of Category objects using the external system's id..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/getByOriginUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|--|
| <code>originuid</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type: `List<Category>`

Category object with the verbose information retrieved in the dao layer.

Example

```
<getByOriginUID>
  <originuid>25</originuid>
  <locale>en_us</locale>
</getByOriginUID>
```

18.2.8 categoryService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>uid</code> | Long | Integer unique id for the <code>LocaleBaseModel</code> object |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

`LocaleBaseModel` if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>57</uid>
  <locale>en_us</locale>
</getByUID>
```

18.2.9 categoryService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: List<M>

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>12</uid>
</getMultiLanguageList>
```

18.2.10 categoryService.getPagedCategories

Gets all the categories we have got for an specific locale..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---|
| locale | String | Optional language identifier, null assumes the default language |
| startRowNum | Long | Long |

| Name | Type | Description |
|-----------|------|-------------|
| endRowNum | Long | Long |

Return value

Type: `List<Category>`

List If found, return all Category entities, null otherwise.

Example

```
<getPagedCategories>
  <locale>en_us</locale>
  <startRowNum>59</startRowNum>
  <endRowNum>28</endRowNum>
</getPagedCategories>
```

18.2.11 categoryService.getPagedCategoriesByOriginUID

Gets the list of Category objects using the external system's id..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|---------|--|
| originuid | Integer | Integer unique origin id for customer representation |
| locale | String | String reference for the language |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<Category>`

Category object with the verbose information retrieved in the dao layer.

Example

```
<getPagedCategoriesByOriginUID>
  <originuid>13</originuid>
  <locale>en_us</locale>
  <startRowNum>29</startRowNum>
```

```
<endRowNum>30</endRowNum>
</getPagedCategoriesByOriginUID>
```

18.2.12 categoryService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

18.2.13 categoryService.removeChannel

Remove a database mapping between a **Channel** and a **Category** entity.

Ex. Remove 'Sky Sports 1' from the 'Sports' package.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/removeChannel>

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|-------------------------------|
| channeluid | Long | The Channel uid to be removed |
| categoryuid | Long | The parent Category uid |

Return value

Type:

Integer The num of rows removed.

Example

```
<removeChannel>
<channeluid>74</channeluid>
<categoryuid>52</categoryuid>
</removeChannel>
```

18.2.14 categoryService.removeMultiLanguage

Contains the business logic to delete a category multilanguage.

Receives all the information as a parameter values.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/categoryService/removeMultiLanguage>

Parameters

| Name | Type | Description |
|--------|--------|--|
| uid | Long | Integer with the id of the channel to remove. |
| locale | String | String with the language and locale specification. |

Return value

Type:

Integer num of rows deleted

18.2.15 categoryService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

18.2.16 categoryService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

18.2.17 categoryService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

19 cdnService

This *Service* provides an interface to external Content Delivery Networks (CDNs)

Set-top boxes can obtain the URL of content stored on the CDN by calling `cdnService.getCdnUrl`. This *Service* determines which CDN is in use based on SDP's configuration, and uses this information to construct the URL.

Note | Currently only the Akamai CDN is supported.

19.1 CDNService

Cdn Service Implementation. This service is exposed to the gateway to provide services described in the Content DeliveryNetworkService interface. Methods may be exposed via HUE or WebService, depending on the operation required. The methods here perform business operations, utilising methods in an injected, specific implementation where required. A single underlying implementation of CDN can be in use at any one time. This is referenced via a QubeRef injection from the CDN spring configuration file.

John Kramer @author

1.0 @version

19.1.1 CDNService.getCdnUrl

Obtain a suitable URL for a content item on the configured content delivery network.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/CDNService/getCdnUrl`

Parameters

| Name | Type | Description |
|------------------------|--------|---|
| <code>inUrlPath</code> | String | The original filename or URL path of the item. |
| <code>param</code> | String | The name of the URL query parameter which is used to hold the CDN's authentication token, if different from the default for the CDN type in use. (For example, Akamai normally uses <code>__gda__</code> .) |
| <code>window</code> | Long | A string representing the validity period of the CDN's authentication token, in a format understood by the CDN, if the CDN type in use requires it. |
| <code>salt</code> | String | The random salt to use to hash the encryption token, if the CDN type in use requires it. |

| Name | Type | Description |
|---------------------------|---------|---|
| <code>extract</code> | String | Any additional information required by the CDN type in use to generate the URL or token. |
| <code>time</code> | Long | A string representing the time the CDN's authentication token should become valid, in a format understood by the CDN. |
| <code>extensionFix</code> | boolean | Should SDP apply the IE extension fix? |

Return value

Type:

The URL of the content item, including a query string with the appropriate authentication token where relevant.

19.1.2 CDNService.validateToken

A method that validates a token originating from the CDN network.

The Token is validated against other information know to SDP about the original generation process.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|-------------|
| <code>generatedURL</code> | String | |

Return value

Type: `int`

`int` to the response status.

Example

```
<validateToken>
  <generatedURL>stringValue</generatedURL>
</validateToken>
```

20 cdnTokenService

This is a primitive *Service* which provides the *CdnToken* Entity.

20.1 CdnToken

A *CdnToken* Entity contains the access credentials required by a Content Delivery Network (CDN) to access content stored on the network.

CdnToken Entities are automatically managed by the CDN *Services* in SDP. Clients should not create, modify or delete them.

20.1.1 CdnToken

| Name | Type | Description |
|----------------|---|---|
| accountUID | Long | The Account UID used in the original URL generation. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| creationDate | Date | The date created. |
| deviceUID | Long | The Device UID used in the original URL generation. |
| exportID | Integer | The id to export object to third part systems. |
| failureCount | Long | How many re-verification accesses were unsuccessful. |
| generatedURL | String | The validated SDP-generated URL sent back to CDN on request. |
| modifiedDate | Date | The date last modified and persisted. |
| originID | Integer | The uid of the origin system (3rd part system). |
| originIdAndKey | <i>Read only.</i> String | |
| originKey | String | The key of the model object in the origin system (3rd part system). |

| Name | Type | Description |
|--------------------------------|--------|--|
| <code>originalURL</code> | String | The original URL supplied to create the CDN Token. |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>successCount</code> | Long | How many re-verification accesses were successful. |
| <code>uID</code> | Long | The unique identifier. |

20.2 cdnTokenService

CdnToken Service Implementation. This service provides a persistence mechanism to store information about the generation of Tokens for Content Delivery Networks eg. Akamai

John Kramer @author

20.2.1 cdnTokenService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>31</originID>
</getByOrigin>
```

20.2.2 cdnTokenService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>  
<uid>94</uid>  
</getByUID>
```

21 challengeAggregateService

This *Service* provides encrypted challenges for set-top boxes to solve in order to obtain licenses.

To get a challenge, the set-top box calls *ChallengeAggregateService.acquireSecureChallenge*.

In the Nagra PRM system, the set-top box uses its DVL to decrypt the challenge, and sends the result to SDP with the next license request. SDP only acts on the license request if the challenge was correctly solved.

SDP uses *Challenge* Entities to store the challenges it has supplied to set-top boxes.

21.1 challengeAggregateService

The Challenge Aggregate Service is a SDP aggregate service that is responsible for issuing and validating authentication challenges. The Challenge Aggregate Service is made visible to HUE or to WebServices using annotation in the Service Implementation class e.g.: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) This service has methods which are exposed to HUE but not to web-services. Hence the API exposed by this service has the pre-requisite that the set top box must be signed on and as a result the account UID and the device UID have been set in the Auth Token of the request context.

mdouthwaite @author

21.1.1 challengeAggregateService.acquireSecureChallenge

Obtain a new encrypted challenge.

The challenge received can only be decrypted by the device's encryption hardware. The license server identifies *Set Top Boxes* and *Media Players* using their **caSN**, derived from the information provided during signon.

Caution!

Clients which do not provide device information when it signed on cannot use this API. This also applies to clients which sign on anonymously.

Media Players which initially signed on without providing device information can update their status using *signonService.updateTokenWithMediaPlayer*, supplying their **mediaPlayerId**. SDP responds with a new token which should be used in subsequent requests.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/challengeAggregateService/acquireSecureChallenge>

Parameters

None.

Return value

Type:

An encrypted / signed challenge, suitably encoded.

21.1.2

challengeAggregateService.acquireSecureChallengeWithAdditionalInfoAndContent

WS specific version of acquireSecureChallenge(String nuid, String cts).

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|--|--|
| context | tv.quative.service.context.api.Context | Context context, must not be null |
| nuid | String | The NUID is the Nagra Unique ID for a CAK. |
| cts | String | The Chipset Type String, a human-transcribable encoding of a vector of capability flags. |

Return value

Type:

An encrypted / signed challenge, suitably encoded.

21.1.3 challengeAggregateService.acquireSecureChallengeWithContext

WS specific version of acquireSecureChallenge().

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|--|-----------------------------------|
| context | tv.quative.service.context.api.Context | Context context, must not be null |

Return value

Type:

An encrypted / signed challenge, suitably encoded.

21.1.4 challengeAggregateService.verifySecureChallengeWithContext

Validate and authenticate a challenge..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|--|---|
| context | tv.quative.service.context.api.Context | Current context (containing a valid challenge response) |

Return value

Type:

Whether the challenge was successfully authorised.

22 challengeService

This is a primitive *Service* which provides the *Challenge* Entity.

Caution!

This *Service* is intended for other *Services* to use. Clients should not attempt to call any API of this *Service*.

22.1 Challenge

A *Challenge* represents a challenge to a set-top box to prove its credentials.

In the Nagra DVS license system, a set-top box wanting to obtain a *License* to view content must first obtain a *Challenge* from the SDP server and return the correct response to SDP in the *License* request.

22.1.1 Challenge

| Name | Type | Description |
|------------------|---|---|
| challengeID | String | The unique ID for the challenge |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| deviceUID | Long | UID of the device STB table. |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| issueDate | Date | The date time at which the challenge was issued |
| lastVerifiedDate | Date | The time at which the challenge was last verified |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |
| originID | Integer | An identifier for the system which owns this <i>Entity</i> . |

| Name | Type | Description |
|--------------------------------|---------------------------------------|--|
| | | Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |
| <code>validity</code> | <code>Integer</code> | Number of seconds from the issue date the license is valid for |

22.2 challengeService

The Challenge Service is a primitive service that encapsulates all the functionality necessary to retrieve, create, update and delete challenge details stored on the database. The Challenge Service implementation is an extension of the non-locale based AbstractCRUDService business layer (part of our standard non-locale based service implementation using the QSP Platform). The AbstractCRUDService is wrapped around the underlying AbstractDAO interface which acts as the as the DAO layer. The Challenge Service is made visible to HUE or to WebServices using annotation in the Service Implementation class e.g.: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true)

mdouthwaite @author

22.2.1 challengeService.create

Create (persist) a `BaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

22.2.2 challengeService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

22.2.3 challengeService.getByChallengeID

Retrieve a challenge from the database using a specified challenge ID.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/challengeService/getByChallengeID>

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|-------------|
| challengeID | String | - String |

Return value

Type:

Challenge

Example

```
<getByChallengeID>  
  <challengeID>ABC-74618</challengeID>  
</getByChallengeID>
```

22.2.4 challengeService.getByDeviceUID

Retrieve a challenge from the database using a specified device UID.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/challengeService/getByDeviceUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-------------|
| deviceUID | Long | - Long |

Return value

Type:

Challenge

Example

```
<getByDeviceUID>  
  <deviceUID>16</deviceUID>  
</getByDeviceUID>
```

22.2.5 challengeService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>28</originID>
</getByOrigin>
```

22.2.6 challengeService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>3</uid>
</getByUID>
```

22.2.7 challengeService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

23 channelService

This is a primitive *Service* which provides the BTV *Channel* Entity.

Relationships with *Packages* and *Categories* are handled in their respective *Services*.

23.1 tv.quative.service.channel.model.ChannelUIDSet

| Name | Type | Description |
|---------------|-----------|----------------------------------|
| channelUIDSet | Set<Long> | The channel UIDs to investigate. |

Tip

When calling this API over HUE, supply the *Channel* UIDs as a JSON list, in any order.
For example:
[1, 2, 4, 6]

23.2 channelService

Channel Service Implementation. Manages the business logic of all the operations with Channels. Object which implements the business logic of all the methods of the interface. The services are made visible to Hue or to Web Services by writing the annotations in the Service Implementation class. Ex: 0 For the ServiceClass: @QubeService(name="packageService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The methods used to create, delete or update items of the BTV System, only could be accesible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

jraya @author

23.2.1 channelService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

23.2.2 channelService.create

Create (persist) a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

23.2.3 channelService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

23.2.4 channelService.getAllAuthorizedChannelsForAccount

Retrieve ALL channels that the account is authorized to view.

The account is looked up using the auth token.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAllAuthorizedChannelsForAccount`**Parameters**

None.

Return valueType: `Set<Channel>`

All channels that the account is authorized to view.

23.2.5 channelService.getAllAuthorizedChannelsForDevice

Retrieve ALL channels that the device is authorized to view.

The device is looked up using the auth token.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAllAuthorizedChannelsForDevice`**Parameters**

None.

Return valueType: `Set<Channel>`All channels that the device is authorized to view, including those purchased across the whole *Account* (the standard behaviour) and those where the subscription has been explicitly restricted to a single *Smart Card*.

23.2.6 channelService.getAllChannels

Gets a list of all the existing Channels..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAllChannels`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| locale | String | String with the language and locale specification. |
| type | String | String type of the channel |

Return value

Type: `List<Channel>`

List with Channel objects for every one of this channels

Example

```
<getAllChannels>
  <locale>en_us</locale>
  <type>stringValue</type>
</getAllChannels>
```

23.2.7 channelService.getAllChannelsAllLocales

Lists all (really) due to the current service provider..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAllChannelsAllLocales`

Available through Web Services.

Parameters

None.

Return value

Type: `List<Channel>`

The list.

Example

```
<getAllChannelsAllLocales/>
```

23.2.8 channelService.getAllChannelsByLocale

Gets a list of all the existing Channels, without any paging..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAllChannelsByLocale`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| locale | String | String with the language and locale specification. |

Return value

Type: `List<Channel>`

List with Channel objects for every one of this channels

Example

```
<getAllChannelsByLocale>
  <locale>en_us</locale>
</getAllChannelsByLocale>
```

23.2.9

channelService.getAuthorizedChannelsForAccountByChannelUIDSet

Given a Set of Channel UIDs, return those that the account is authorized to view.

The account is looked up using the auth token.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAuthorizedChannelsForAccountByChannelUIDSet`

Parameters

| Name | Type | Description |
|---------------------|--|--|
| passedInChannelUIDs | tv.quative.service.channel.model.ChannelUIDSet | The UIDs of the Channels to investigate. |

Return value

Type:

Those channel UIDs that the account is authorized to view.

23.2.10

channelService.getAuthorizedChannelsForDeviceByChannelUIDSet

Given a Set of Channel UIDs, return those that the device is authorized to view.

The device is looked up using the auth token.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getAuthorizedChannelsForDeviceByChannelUIDSet`

Parameters

| Name | Type | Description |
|----------------------------------|---|---|
| <code>passedInChannelUIDs</code> | <code>tv.quative.service.channel.model.ChannelUIDSet</code> | The UIDs of the <i>Channels</i> to investigate. |

Return value

Type:

The UIDs of the *Channels* from the passed-in set that the device is authorized to view, including those purchased across the whole *Account* (the standard behaviour) and those where the subscription has been explicitly restricted to a single *Smart Card*.

23.2.11 channelService.getByCasID

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|-------------|
| <code>casID</code> | String | |
| <code>locale</code> | String | |

Return value

Type:

Example

```
<getByCasID>
  <casID>ABC-28036</casID>
  <locale>en_us</locale>
</getByCasID>
```

23.2.12 channelService.getByCatUID

Gets all the Channels for the category passed as a parameter..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getByCatUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|--|
| <code>pcatuid</code> | Long | Long unique id of the category. |
| <code>locale</code> | String | String with the language and locale specification. |
| <code>type</code> | String | String type of the channel |

Return value

Type: `List<Channel>`

List with Channel objects for every one of the channels retrieved

Example

```
<getByCatUID>
  <pcatuid>73</pcatuid>
  <locale>en_us</locale>
  <type>stringValue</type>
</getByCatUID>
```

23.2.13 channelService.getByDrmID

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|-------------|
| <code>drmID</code> | String | |
| <code>locale</code> | String | |

Return value

Type:

Example

```
<getByDrmID>
  <drmID>ABC-91390</drmID>
  <locale>en_us</locale>
```

</getByDrmID>

23.2.14 channelService.getByDrmIdAnyServiceProvider

Retrieve a channel based on drmID on any service provider..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| drmID | String | the business unique identifier of a Channel . |
| locale | String | reference for the language |

Return value

Type:

Channel If found, the matching object, null otherwise.

Example

```
<getByDrmIdAnyServiceProvider>
<drmID>ABC-97290</drmID>
<locale>en_us</locale>
</getByDrmIdAnyServiceProvider>
```

23.2.15 channelService.getByNumber

The Channels related to the given number.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getByNumber>

Parameters

| Name | Type | Description |
|--------|---------|-----------------------------------|
| number | Integer | String the Channel Number |
| locale | String | String reference for the language |

Return value

Type: `List<Channel>`

List of channels with the given number.

23.2.16 channelService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | The <code>originKey</code> of the <i>Channel</i> . |
| <code>originID</code> | Integer | The <code>originID</code> of the <i>Channel</i> . |
| <code>locale</code> | String | The locale to use in localizable fields of the returned object. |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>64</originID>
  <locale>en_us</locale>
</getByOrigin>
```

23.2.17 channelService.getByOriginUID

Gets the Channel object information using the external system's id.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|--|
| <code>originuid</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |
| <code>type</code> | String | String type of the channel |

Return value

Type: `List<Channel>`

List of Channel objects with the multilanguage information

Example

```
<getByOriginUID>
  <originuid>79</originuid>
  <locale>en_us</locale>
  <type>stringValue</type>
</getByOriginUID>
```

23.2.18 channelService.getByPkgPattern

Gets all the channels belonging to list of packages specified in the pattern parameter..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getByPkgPattern`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>pkgPattern</code> | String | List of hyphen separated package UIDs e.g. pkgUID1-pkgUID2-pkgUID3... |
| <code>locale</code> | String | object that contains the id of the current language. |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects with the information of the channel and the premium information

Example

```
<getByPkgPattern>
  <pkgPattern>stringValue</pkgPattern>
  <locale>en_us</locale>
</getByPkgPattern>
```

23.2.19 channelService.getByPkgUID

Gets the Channel object information using the uid of the package as the key..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getByPkgUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|--|
| <code>ppkguid</code> | Long | Long unique id of the package |
| <code>locale</code> | String | String with the language and locale specification. |
| <code>type</code> | String | String type of the channel |

Return value

Type: `List<Channel>`

List of Channel objects with the information of the channels found.

Example

```
<getByPkgUID>
  <ppkguid>50</ppkguid>
  <locale>en_us</locale>
  <type>stringValue</type>
</getByPkgUID>
```

23.2.20 channelService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|---|
| uid | Long | Integer unique id for the LocaleBaseModel object |
| locale | String | The locale to use in localizable fields of the returned object. |

Return value

Type:

LocaleBaseModel if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>3</uid>
  <locale>en_us</locale>
</getByUID>
```

23.2.21 channelService.getChannelType

Gets the channel type information using the unique id..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getChannelType>

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------------------------|
| cuid | Long | Long unique id of the channel |

Return value

Type:

String type of the channel retrieved in the dao layer

Example

```
<getChannelType>
  <cuid>59</cuid>
</getChannelType>
```

23.2.22 nvodChannelService.getMultiLanguageList

A Service method to get the nvodchannel multi language details list for uid.

Contains the business logic required to get nvodchannel multi language details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | Long |

Return value

Type: `List<Channel>`

List of Channel objects

Example

```
<getMultiLanguageList>
  <uid>27</uid>
</getMultiLanguageList>
```

23.2.23 channelService.getPagedChannelsByCatUID

Gets all the Channels for the category passed as a parameter..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getPagedChannelsByCatUID>

Parameters

| Name | Type | Description |
|-------------|--------|--|
| catid | Long | Long unique id of the category. |
| locale | String | String with the language and locale specification. |
| type | String | String type of the channel |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<Channel>`

List with Channel objects for every one of the channels retrieved

23.2.24 `channelService.getPagedChannelsByOriginUID`

Gets the Channel object information using the external system's id..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getPagedChannelsByOriginUID`

Parameters

| Name | Type | Description |
|--------------------------|---------|--|
| <code>originuid</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |
| <code>type</code> | String | String type of the channel |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<Channel>`

List of Channel objects with the multilanguage information

23.2.25 `channelService.getPagedChannelsByPkgPattern`

Gets all the channels belonging to list of packages specified in the pattern parameter..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getPagedChannelsByPkgPattern`

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>pkgPattern</code> | String | List of hyphen separated package UIDs e.g. pkgUID1-pkgUID2-pkgUID3... |

| Name | Type | Description |
|-------------|--------|--|
| locale | String | object that contains the id of the current language. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<MappedChannel>`

List of MappedChannel objects with the information of the channel and the premium information

23.2.26 channelService.getPagedChannelsByPkgUID

Gets the Channel object information using the uid of the package as the key..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/channelService/getPagedChannelsByPkgUID>

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pkguid | Long | Long unique id of the package |
| locale | String | String with the language and locale specification. |
| type | String | String type of the channel |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<Channel>`

List of Channel objects with the information of the channels found.

23.2.27 channelService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

23.2.28 channelService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

23.2.29 channelService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

23.2.30 channelService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

24 codBrowsingService

This *Service* provides APIs for browsing through the content in on-demand *Catalogues* and for discovering purchase offers.

It provides three sets of APIs:

- ▶ *codBrowsingService.getScheduledItems* returns items in the *Catalogue* structure in the form of *ScheduledItem* objects.

A *ScheduledItem* contains all the information a client will need to display an *Asset* and the purchase offers available for it.

The **context** and **filter** supplied to the API determine which items to return. For example, a selection can be made from a single *Catalogue*, from a *Catalogue* subtree, or across all *Catalogues*. Multiple filters can be applied at the same time; clients can also specify the sort order of returned items.

codBrowsingService.getScheduledItemsForCurrentContext works in the same way, but the client's context is derived from the information provided at signon, so there is no need to supply one.

codBrowsingService.getScheduledItemsCount and *codBrowsingService.getScheduledItemsForCurrentContextCount* reflect these two APIs but instead of returning the items, they return the number of items in the result set. Clients can then request full details from pages out of that set as desired.

- ▶ *codBrowsingService.getScheduledAssetGroups* returns *Asset Groups* represented in the *Catalogue*.

It takes the same parameters as *codBrowsingService.getScheduledItems*. An *Asset Group* will be returned if the corresponding call to *codBrowsingService.getScheduledItems* would have returned at least one *Asset* belonging to that *Asset Group*.

- ▶ *codBrowsingService.getTrailerItems* returns *Assets* representing the trailers for other *Assets*.

It takes similar parameters to *codBrowsingService.getScheduledItems*, but the **filter** parameter provides different filtering options.

24.1 tv.quative.service.codbrowsing.api.Filter

| Name | Type | Description |
|-------------------------------------|--------------|---|
| assetActorSubstringList | List<String> | Only return results where the actor of the <i>Asset</i> contains any string from the list. |
| assetCUBroadcastStartsAfter | Date | Only return results where the cuBroadcastStartDate of the <i>Asset</i> is after the specified date and time. In combination with assetCUBroadcastStartsBefore , this can be used to discover catch-up events from a desired time period that are being offered in the VOD <i>Catalogue</i> . |
| assetCUBroadcastStartsBefore | Date | Only return results where the cuBroadcastStartDate of the <i>Asset</i> is before the specified date and time. |

| Name | Type | Description |
|--|---------------------------------|--|
| | | In combination with <code>assetCUBroadcastStartsAfter</code> , this can be used to discover catch-up events from a desired time period that are being offered in the VOD <i>Catalogue</i> . |
| <code>assetCategoryNameList</code> | <code>List<String></code> | <p>Only return results where the <i>Asset</i> belongs to a <i>COD Category</i> whose <code>name</code> is in the list.</p> <p><i>For example:</i> Clients can use this filter to discover content in a named genre.</p> <p>Compare with <code>assetGroupCategoryNameList</code>, where only items which belong to an <i>Asset Group</i> in the named <i>COD Category</i> are returned.</p> |
| <code>assetCategoryUIdList</code> | <code>List<Long></code> | Only return results where the <i>Asset</i> belongs to a <i>COD Category</i> whose <code>uID</code> is in the list. |
| <code>assetDescriptionSubstringList</code> | <code>List<String></code> | Only return results where the <code>description</code> of the <i>Asset</i> contains any string from the list. |
| <code>assetDirectorSubstringList</code> | <code>List<String></code> | Only return results where the <code>director</code> of the <i>Asset</i> contains any string from the list. |
| <code>assetGroupCategoryNameList</code> | <code>List<String></code> | <p>Only return results where the <i>Asset</i> is scheduled as part of an <i>Asset Group</i>, and the <i>Asset Group</i> belongs to a <i>COD Category</i> whose <code>name</code> is in the list.</p> <p><i>For example:</i> Clients can use this filter to discover episodes of series in a named genre.</p> <p>Compare with <code>assetCategoryNameList</code>, where the <i>Asset</i> itself must belong to the <i>COD Category</i>.</p> |
| <code>assetGroupCategoryUIdList</code> | <code>List<Long></code> | Only return results where the <i>Asset</i> is scheduled as part of an <i>Asset Group</i> , and the <i>Asset Group</i> belongs to a <i>COD Category</i> whose <code>uID</code> is in the list. |
| <code>assetGroupIsNull</code> | <code>boolean</code> | Only return results for <i>Assets</i> which are not scheduled as part of an <i>Asset Group</i> . |

Note

Specifying `false` for a boolean filter criterion has no effect.

| Name | Type | Description |
|--|---------------------------------|--|
| <code>assetGroupUId</code> | <code>Long</code> | Only return results for <i>Assets</i> scheduled as part of the <i>Asset Group</i> with the supplied <code>uID</code> . |
| <code>assetKeywordSubstringList</code> | <code>List<String></code> | Only return results where any of the <code>name</code> , <code>description</code> , <code>actor</code> or <code>director</code> of the <i>Asset</i> contains any string from the list. |
| <code>assetLongDescriptionSubstringList</code> | <code>List<String></code> | Only return results where the <code>longDescription</code> of the <i>Asset</i> contains any string from the list. |
| <code>assetPlayType</code> | <code>AssetPlayType</code> | <p>Only return results where the <code>assetPlayType</code> of the <i>Asset</i> has the supplied value.</p> <p>This is most often used to ensure that only entertainment <i>Assets</i> (with <code>assetPlayType = E</code>) are returned during catalogue browsing.</p> <p>This filter is only useful in deployments where <i>Assets</i> of other types are scheduled, such as advertisements (<code>assetPlayType = A</code>) or trailers (<code>assetPlayType = T</code>).</p> |
| <code>assetProfileUIdList</code> | <code>List<Long></code> | <p>Only return results where the <i>Asset</i> is associated with an <i>Asset Profile</i> whose <code>uID</code> is in the list.</p> <p>Caution! SDP uses this filter to ensure that returned items are suitable for the client device. Its value is automatically derived from the <i>Context</i>. Clients should not attempt to supply their own value for this filter; if you want to disable filtering by <i>Asset Profile</i> and <i>Product Profile</i>, use the filter <code>disableDeviceFiltering</code>.</p> |
| <code>assetRatingUIdList</code> | <code>List<Long></code> | <p>Only return results where the <i>Asset</i> has a <i>Rating</i>, whose <code>uID</code> is in the list.</p> <p>Compare with <code>excludeAssetRatingUIdList</code>.</p> |
| <code>assetShortTitleSubstringList</code> | <code>List<String></code> | |
| <code>assetTitleSubstringList</code> | <code>List<String></code> | Only return results where the <code>title</code> of the <i>Asset</i> contains any string from the list. |

| Name | Type | Description |
|-------------------------------|---------|---|
| <code>assetUId</code> | Long | <p>Only return results for the <i>Asset</i> with the specified <code>uID</code>.</p> <p>Note You may still get multiple items, if there are several versions of the <i>Asset</i> suitable for the device.</p> |
| <code>catalogueName</code> | String | <p>Only return results where the <i>Asset</i> is in the <i>Catalogue</i> with the supplied <code>name</code>.</p> <p>Tip Clients wanting to implement a standard catalogue browsing interface should include either this filter or <code>catalogueUId</code> to specify the <i>Catalogue</i>.</p> <p>Note Unless you also supply <code>true</code> in the <code>cataloguesubtree</code> field, the <i>Asset</i> must be in this specific <i>Catalogue</i>, not in any sub-<i>Catalogue</i>.</p> |
| <code>cataloguesubtree</code> | boolean | <p>If this field is set to <code>true</code>, references to <i>Catalogues</i> in other filters include all child <i>Catalogues</i> and descendants of these <i>Catalogues</i>.</p> <p>If it is <code>false</code>, or it is not set, references to a <i>Catalogue</i> refer only to that <i>Catalogue</i>.</p> |
| <code>catalogueUId</code> | Long | <p>Only return results where the <i>Asset</i> is in the <i>Catalogue</i> with the supplied <code>uID</code>.</p> <p>Tip Clients wanting to implement a standard catalogue browsing interface should include either this filter or <code>catalogueName</code> to specify the <i>Catalogue</i>.</p> <p>Note Unless you also supply <code>true</code> in the <code>cataloguesubtree</code> field, the <i>Asset</i> must be in this specific <i>Catalogue</i>, not in any sub-<i>Catalogue</i>.</p> |

| Name | Type | Description |
|--|---|---|
| <code>disableDeviceFiltering</code> | <code>boolean</code> | <p>If this field is set to <code>false</code>, or it is not set, SDP will use the <code>Context</code> to set an <code>assetProfileUidList</code> filter so that only items suitable for the client device are returned.</p> <p>If this field is set to <code>true</code>, SDP will not filter on the basis of device features.</p> |
| <code>excludeAssetRatingUidList</code> | <code>List<Long></code> | <p>Only return results where the <i>Asset</i> does not have a <i>Rating</i> whose <code>uid</code> is in the list.</p> <p>Compare with <code>assetRatingUidList</code>.</p> |
| <code>offerFrequencyTypeList</code> | <code>List<OfferFrequencyType></code> | <p>Only return <i>Offers</i> for products where the <i>Policy Group</i> has the supplied <code>frequencyType</code>.</p> <p>Note If all <i>Offers</i> for an item are filtered out, SDP will not return the item at all.</p> |
| <code>offerIsCurrent</code> | <code>boolean</code> | <p>Only return <i>Offers</i> for products where the <i>Base Price</i> is currently valid (that is, the current date is between the <code>validFrom</code> and <code>validTo</code> dates of the <i>Base Price</i>.)</p> <p>Note Specifying <code>false</code> for a boolean filter criterion has no effect. You can only use this criterion to discover current <i>Offers</i>.</p> <p>Note If all <i>Offers</i> for an item are filtered out, SDP will not return the item at all.</p> <p>Tip It's recommended to use <code>scheduleIsCurrent</code> instead of this filter. In a correctly-configured system, current <i>Schedules</i> will never be associated with non-current <i>Base Prices</i>, so you never need to specify both filters.</p> |
| <code>offerUid</code> | <code>Long</code> | <p>Only return items which are provided by the specified <i>Offer</i>, identified as the <i>Policy Group</i> with the specified <code>uid</code>.</p> |

| Name | Type | Description |
|---|-------------------------------|---|
| <code>productProfileUid</code> <code>List</code> | <code>List<Long></code> | <p>Only return results where the <i>Policy Group</i> is associated with a <i>Product Profile</i> whose <code>uID</code> is in the list.</p> <p>Caution! SDP uses this filter to ensure that returned offers are targeted at the client device. Its value is automatically derived from the <i>Context</i>. Clients should not attempt to supply their own value for this filter; if you want to disable filtering by <i>Asset Profile</i> and <i>Product Profile</i>, use the filter <code>disableDeviceFiltering</code>.</p> |
| <code>scheduleEndsAfter</code> | <code>Date</code> | <p>Only return items where the <code>endTime</code> of the <i>Schedule</i> is after the specified date.</p> <p>Note This filter will include <i>Schedules</i> which have not yet started or which have already ended, as long as they fit the criterion.</p> |
| <code>scheduleEndsBefore</code> | <code>Date</code> | <p>Only return items where the <code>endTime</code> of the <i>Schedule</i> is before the specified date.</p> <p>Note This filter will include <i>Schedules</i> which have not yet started or which have already ended, as long as they fit the criterion. Compare <code>scheduleWillEndBefore</code>.</p> |
| <code>scheduleIsCurrent</code> | <code>boolean</code> | <p>Only return items where the <i>Schedule</i> is currently available (that is, the <code>startTime</code> is before the current date and the <code>endTime</code> is after the current date.)</p> <p>Note Specifying <code>false</code> for a boolean filter criterion has no effect. You can only use this criterion to discover current <i>Schedules</i>.</p> <p>Caution! In normal browsing you should always include this filter and give it a value of <code>true</code>, because users can only purchase content which is associated with a current <i>Schedule</i>. If you are deliberately querying for historical <i>Schedules</i>, or for <i>Schedules</i> which are not</p> |

| Name | Type | Description |
|---|---------------------------------|--|
| | | <p>yet available (for example, to implement a "coming soon" page), you should use <code>scheduleStartsAfter</code>, <code>scheduleStartsBefore</code>, <code>scheduleEndsAfter</code>, <code>scheduleEndsBefore</code>, <code>scheduleStartedAfter</code>, and/or <code>scheduleWillEndBefore</code> instead of this filter.</p> |
| <code>scheduleOriginKey</code> <code>List</code> | <code>List<String></code> | |
| <code>scheduleStarted</code> <code>After</code> | <code>Date</code> | <p>Only return items where the <code>startTime</code> of the <i>Schedule</i> is between the specified past date and now. This is useful to highlight <i>Schedules</i> which have recently become available ("new this week")</p> <p>Note Compare <code>scheduleEndsBefore</code>.</p> |
| <code>scheduleStarts</code> <code>After</code> | <code>Date</code> | <p>Only return items where the <code>startTime</code> of the <i>Schedule</i> is after the specified date.</p> <p>Note This filter will include <i>Schedules</i> which have not yet started or which have already ended, as long as they fit the criterion.</p> |
| <code>scheduleStarts</code> <code>Before</code> | <code>Date</code> | <p>Only return items where the <code>startTime</code> of the <i>Schedule</i> is before the specified date.</p> <p>Note This filter will include <i>Schedules</i> which have not yet started or which have already ended, as long as they fit the criterion.</p> |
| <code>scheduleStatus</code> | <code>ScheduleStatus</code> | <p>Only return items where the <i>Schedule</i> has the specified <code>status</code>.</p> <p>Caution! You should always include this filter and give it a value of <code>P</code> (published), since users should only be able to view published <i>Schedules</i>.</p> |

| Name | Type | Description |
|-------------------------------------|------|---|
| <code>scheduleUid</code> | Long | Only return items associated with the <i>Schedule</i> with the specified <code>uID</code> . |
| <code>scheduleWillEnd Before</code> | Date | Only return items where the <code>endTime</code> of the <i>Schedule</i> is between now and the specified future date. This is useful to highlight <i>Schedules</i> which will soon become unavailable ("last chance to see...") |
| | | Note Compare <code>scheduleEndsBefore</code> . |
| <code>technicalAssetUid</code> | Long | Only return items where the <code>uid</code> of the technical <i>Asset</i> matches the specified value. This is useful to find a specific <i>Asset</i> wherever it occurs in the <i>Catalogue</i> tree: perhaps one which has been previously purchased, or one which is bookmarked for later purchase. |

24.2 tv.quative.service.codbrowsing.api.Filter.AssetPlayType

| Name | Type | Description |
|----------------|------|--|
| <code>A</code> | | The editorial <i>Asset</i> is an advertisement. |
| <code>E</code> | | The editorial <i>Asset</i> is an entertainment (normal) <i>Asset</i> . |

24.3

tv.quative.service.codbrowsing.api.Filter.OfferFrequencyType

| Name | Type | Description |
|------------------|------|---|
| <code>IMP</code> | | Offers of type IMPULSE (<code>IMP</code> in <i>Base Price</i>) |
| <code>MUL</code> | | Offers of type MULTIPLE (<code>MUL</code> in <i>Base Price</i>) |

| Name | Type | Description |
|------|------|--|
| REC | | Offers of type RECURRING (REC in <i>Base Price</i>) |
| SEP | | Offers of type SEASON PASS (SEP in <i>Base Price</i>) |

24.4 tv.quative.service.codbrowsing.api.Filter.ScheduleStatus

| Name | Type | Description |
|------|------|-------------------------------------|
| P | | The <i>Schedule</i> is published. |
| U | | The <i>Schedule</i> is unpublished. |

24.5

tv.quative.service.codbrowsing.api.ScheduledItemRequest

| Name | Type | Description |
|-----------|------------|--|
| context | Context | The <i>Context</i> to use for supporting information about the request. |
| endRow | Long | The index of the last row of the result set to return. |
| filter | Filter | A <i>Filter</i> object (<code>tv.quative.service.codbrowsing.api.Filter</code>) specifying which items to return. |
| sortOrder | List<Sort> | The <i>Sort</i> criteria from <code>tv.quative.service.codbrowsing.api.Sort</code> , in the order they are to be applied.. |
| startRow | Long | The index of the first row of the result set to return. |

24.6 tv.quative.service.codbrowsing.api.Sort

| Name | Type | Description |
|--|------|--|
| <code>assetCUBroadcastStartDate</code> | | Sort by the <code>cuBroadcastStartDate</code> of the <i>Asset</i> , soonest first. |
| <code>assetCUBroadcastStartDateDesc</code> | | Sort by the <code>cuBroadcastStartDate</code> of the <i>Asset</i> , latest first. |
| <code>assetDownload</code> | | Sort by the <code>downloads</code> field of the editorial <i>Asset</i> , least first. |
| <code>assetDownloadDesc</code> | | Sort by the <code>downloads</code> field of the editorial <i>Asset</i> , most first. |
| <code>assetGroupName</code> | | Sort by the <code>name</code> of the <i>Asset Group</i> associated with the schedule (if any), in alphabetical order. |
| <code>assetGroupNameDesc</code> | | Sort by the <code>name</code> of the <i>Asset Group</i> associated with the schedule (if any), in reverse alphabetical order. |
| <code>assetGroupUid</code> | | Sort by the <code>uid</code> of the <i>Asset Group</i> associated with the schedule (if any), smallest first. |
| <code>assetGroupUidDesc</code> | | Sort by the <code>uid</code> of the <i>Asset Group</i> associated with the schedule (if any), largest first. |
| <code>assetLongDescription</code> | | Sort by the <code>longDescription</code> of the editorial <i>Asset</i> , in alphabetical order. |
| <code>assetLongDescriptionDesc</code> | | Sort by the <code>longDescription</code> of the editorial <i>Asset</i> , in reverse alphabetical order. |
| <code>assetOrderInAssetGroup</code> | | Sort by the position of the editorial <i>Asset</i> within the <i>Asset Group</i> associated with the schedule, smallest first. |
| <code>assetOrderInAssetGroupDesc</code> | | Sort by the position of the editorial <i>Asset</i> within the <i>Asset Group</i> associated with the schedule, largest first. |
| <code>assetScoring</code> | | Sort by the <code>scoring</code> field of the editorial <i>Asset</i> , largest first. |

| Name | Type | Description |
|-----------------------------------|------|---|
| <code>assetScoringDesc</code> | | Sort by the <code>scoring</code> field of the editorial <i>Asset</i> , smallest first. |
| <code>assetTitle</code> | | Sort by the <code>name</code> of the editorial <i>Asset</i> , in alphabetical order. |
| <code>assetTitleDesc</code> | | Sort by the <code>name</code> of the editorial <i>Asset</i> , in reverse alphabetical order. |
| <code>assetUid</code> | | Sort by the <code>uid</code> of the editorial <i>Asset</i> . |
| <code>offerPrice</code> | | Sort by the <code>value</code> of the <i>Offer</i> (and <i>Base Price</i>), lowest first. |
| | | <p>Caution! If there is more than one <i>Offer</i> associated with a scheduled <i>Asset</i>, a complete <i>ScheduledItem</i> object needs to be returned for each <i>Offer</i>, in order to sort them correctly. This will significantly increase the amount of data returned.</p> |
| <code>offerPriceDesc</code> | | Sort by the <code>value</code> of the <i>Offer</i> (and <i>Base Price</i>), highest first. |
| | | <p>Caution! If there is more than one <i>Offer</i> associated with a scheduled <i>Asset</i>, a complete <i>ScheduledItem</i> object needs to be returned for each <i>Offer</i>, in order to sort them correctly. This will significantly increase the amount of data returned.</p> |
| <code>playCount</code> | | |
| <code>playCountDesc</code> | | |
| <code>scheduleEndDate</code> | | Sort by the end date of the <i>Schedule</i> , soonest first. |
| <code>scheduleEndDate Desc</code> | | Sort by the end date of the <i>Schedule</i> , latest first. |

| Name | Type | Description |
|--|------|---|
| <code>scheduleOrderIn Catalogue</code> | | Sort by the position (<code>displayPriority</code>) of the scheduled item within its <i>Catalogue</i> , lowest first. Tip This sort order only makes sense if you also filter to a specific <i>Catalogue</i> . If multiple <i>Catalogues</i> are returned, you will get all the item 1's, then the 2's, and so on. |
| <code>scheduleOrderIn CatalogueDesc</code> | | Sort by the position (<code>displayPriority</code>) of the scheduled item within its <i>Catalogue</i> , highest first. Tip This sort order only makes sense if you also filter to a specific <i>Catalogue</i> . If multiple <i>Catalogues</i> are returned, you will get all the item 1's, then the 2's, and so on. |
| <code>scheduleStartDate</code> | | Sort by the start date of the <i>Schedule</i> , soonest first. |
| <code>scheduleStartDate Desc</code> | | Sort by the start date of the <i>Schedule</i> , latest first. |

24.7 tv.quative.service.codbrowsing.api.TrailerFilter

| Name | Type | Description |
|---------------------------------------|-------------------------------|---|
| <code>assetProfileUid List</code> | <code>List<Long></code> | Only return results where the <i>Asset</i> is associated with an <i>Asset Profile</i> whose <code>uID</code> is in the list. Caution! SDP uses this filter to ensure that returned items are suitable for the client device. Its value is automatically derived from the <i>Context</i> . Clients should not attempt to supply their own value for this filter; if you want to disable filtering by <i>Asset Profile</i> and <i>Product Profile</i> , use the filter <code>disable DeviceFiltering</code> . |

| Name | Type | Description |
|-------------------------------------|-------------------------------|--|
| <code>disableDeviceFiltering</code> | <code>boolean</code> | If this field is set to <code>false</code> , or it is not set, SDP will use the <code>Context</code> to set an <code>assetProfileUidList</code> filter so that only items suitable for the client device are returned. |
| <code>mainContentAssetUid</code> | <code>Long</code> | Only return results where the <code>Asset</code> is a trailer for the <code>Asset</code> with the specified <code>uid</code> . |
| <code>productProfileUidList</code> | <code>List<Long></code> | This field is not currently used, because purchase offers are not considered when building <code>TrailerItem</code> objects. |

24.8 tv.quative.service.codbrowsing.api.TrailerItemRequest

| Name | Type | Description |
|----------------------|----------------------------|---|
| <code>context</code> | <code>Context</code> | The <code>Context</code> to use for supporting information about the request. |
| <code>filter</code> | <code>TrailerFilter</code> | A <code>TrailerFilter</code> object (<code>tv.quative.service.codbrowsing.api.TrailerFilter</code>) specifying which items to return. |

24.9 codBrowsingService

24.9.1 codBrowsingService.getCurrentContext

Get default context for this device..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getCurrentContext`

Available through Web Services.

Parameters

None.

Return value

Type:

the `Context` for this device

Example

```
<getCurrentContext/>
```

24.9.2 codBrowsingService.getScheduledAssetGroups

Get a paged list of the *Asset Groups* which are associated with the items scheduled into the on-demand *Catalogue* tree, taking into account the supplied *Context*, *Filter* and *sortOrder*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledAssetGroups`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--|--|
| <code>theRequest</code> | tv.quative.service.codbrowsing.ap.ScheduledItemRequest | A request object specifying the <i>Context</i> and <i>Filter</i> and other request parameters. |

Return value

Type: `List<AssetGroup>`

The *Asset Groups* which match the query.

24.9.3 codBrowsingService.getScheduledAssetGroupsCount

Count how many *Asset Groups* would be returned by a call to *getScheduledAssetGroups* with the supplied *Context* and *Filter*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledAssetGroupsCount`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--|--|
| <code>theRequest</code> | tv.quative.service.codbrowsing.ap.ScheduledItemRequest | A request object specifying the <i>Context</i> and <i>Filter</i> , and if desired the <i>startRowNum</i> or <i>endRowNum</i> . (You can also specify a <i>sortOrder</i> , but it will make no difference to the result.) |

Return value

Type:

The number of *Asset Groups* which match the query.

24.9.4 codBrowsingService.getScheduledAssetGroupsForCurrentContext

Get a paged list of the *Asset Groups* which are associated with the items scheduled into the on-demand *Catalogue* tree, taking into account the supplied *Filter* and *sortOrder* and using a *Context* derived from the information provided at sign-on.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledAssetGroupsForCurrentContext`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--|--|
| <code>theRequest</code> | tv.quative.service.codbrowsing.ap.scheduledItemRequest | A request object specifying the <i>Filter</i> and other request parameters. (If a <i>Context</i> is included, SDP will ignore it.) |

Return value

Type: `List<AssetGroup>`

The *Asset Groups* which match the query.

24.9.5 codBrowsingService.getScheduledAssetGroupsForCurrentContextCount

Count how many *Asset Groups* would be returned by a call to *getScheduledAssetGroups* with the supplied *Filter* and a *Context* derived from the information provided at sign-on.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledAssetGroupsForCurrentContextCount`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--|--|
| <code>theRequest</code> | tv.quative.service.codbrowsing.ap.scheduledItemRequest | A request object specifying the <i>Filter</i> and other request parameters. (If a <i>Context</i> or <i>sortOrder</i> is included, SDP will ignore them.) |

Return value

Type:

The number of *Asset Groups* which match the query.

24.9.6 `codBrowsingService.getScheduledItems`

Get a paged list of *ScheduledItems* representing content available in the on-demand *Catalogue* tree, taking into account the supplied *Context*, *Filter* and `sortOrder`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledItems`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---|--|
| <code>theRequest</code> | <code>tv.quative.service.codbrowsing.apis.ScheduledItemRequest</code> | A request object specifying the <i>Context</i> and <i>Filter</i> and other request parameters. |

Return value

Type: `List<ScheduledItem>`

The *ScheduledItems* which match the query.

24.9.7 `codBrowsingService.getScheduledItemsCount`

Count how many *ScheduledItem* objects would be returned by a call to *getScheduledItems* with the supplied *Context* and *Filter*.

Note

It is possible for the sort order to affect the count, because some sort orders will cause *Scheduled TechnicalItem* objects which would otherwise have been collected into a single *ScheduledItem* to be split into several *ScheduledItem* objects.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledItemsCount`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---|--|
| <code>theRequest</code> | <code>tv.quative.service.codbrowsing.apis.ScheduledItemRequest</code> | A request object specifying the <i>Context</i> and <i>Filter</i> and other request parameters. |

Return value

Type:

24.9.8 `codBrowsingService.getScheduledItemsForCurrentContext`

Get a paged list of *ScheduledItems* representing content available in the on-demand *Catalogue* tree, taking into account the supplied *Filter* and `sortOrder`, and a *Context* derived from the information provided at sign-on.

package.class#member label @see

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledItemsForCurrentContext`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---|--|
| <code>theRequest</code> | <code>tv.quative.service.codbrowsing.ap.ScheduledItemRequest</code> | A request object specifying the <i>Filter</i> and other request parameters. (If a <i>Context</i> is included, SDP will ignore it.) |

Return value

Type: `List<ScheduledItem>`

The *ScheduledItems* which match the query.

24.9.9 `codBrowsingService.getScheduledItemsForCurrentContextCount`

Count how many *ScheduledItem* objects would be returned by a call to *getScheduledItems* with the supplied *Filter* and a *Context* derived from the information provided at sign-on.

Note

It is possible for the sort order to affect the count, because some sort orders will cause *Scheduled TechnicalItem* objects which would otherwise have been collected into a single *ScheduledItem* to be split into several *ScheduledItem* objects.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getScheduledItemsForCurrentContextCount`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---|--|
| <code>theRequest</code> | <code>tv.quative.service.codbrowsing.ap.ScheduledItemRequest</code> | A request object specifying the <i>Filter</i> and other request parameters. (If a <i>Context</i> is included, SDP will ignore it.) |

Return value

Type:

24.9.10 `codBrowsingService.getTrailerItems`

Get a paged list of *ScheduledItems* representing trailers for on-demand content, taking into account the supplied *Context* and *Filter*.

Note

This API takes a **TrailerFilter** object in the **filter** field, which provides a different selection of filters from the **Filter** object used for `codBrowsingService.getScheduledItems`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getTrailerItems`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---|---|
| <code>theRequest</code> | tv.quative.service.codbrowsing.api.TrailerItemRequest | A TrailerItemRequest complex object specifying the context of the request; the filter to apply; the desired sortOrder ; and (optionally) startRowNum and endRowNum fields for paging. |

Return value

Type: `List<TrailerItem>`

The *TrailerItems* matching the query.

24.9.11 `codBrowsingService.getTrailerItemsForCurrentContext`

Get a paged list of *ScheduledItems* representing trailers for on-demand content, taking into account the supplied *Filter* and a *Context* derived from the information provided at sign-on.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codBrowsingService/getTrailerItemsForCurrentContext`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---|--|
| <code>theRequest</code> | tv.quative.service.codbrowsing.api.TrailerItemRequest | context and filters <small>p.355</small> |

Return value

Type: `List<TrailerItem>`

a list of TrailerItems

24.9.12 `codBrowsingService.testContext`

Test helper method for testing complex Context interaction..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|---|-------------------------------|
| <code>theRequest</code> | <code>tv.quative.service.codbrowsing.ap.ScheduledItemRequest</code> | Test request ^{p.351} |
| <code>defaultContext</code> | <code>boolean</code> | Context |

Return value

Type: `void`

25 codCategoryService

This is a primitive *Service* which provides the *COD Category* Entity.

It manages relationships between *Catalogues* and their constituent *Assets* and *Asset Groups*.

25.1 COD Category

A *COD Category* represents a group of *Assets* or *Asset Groups* related by theme or genre.

Figure 8 CODCategory and associated Entities.

COD Categories should not be confused with *Categories*, which perform a similar function for broadcast TV services. The **genre** EPG attribute of each *VOD Item* in the CMS is used to determine the *COD Categories* the corresponding *Asset* should belong to. These *COD Categories* should be created beforehand by an administrator.

25.1.1 CODCategory

| Name | Type | Description |
|-----------------------------|---|--|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | A log of the changes to this <i>Entity</i> history. |
| <code>creationDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was created. |
| <code>description</code> | <code>String</code> | A short description of this <i>Entity</i> . |
| <code>exportID</code> | <code>Integer</code> | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>locale</code> | <code>String</code> | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | <code>Date</code> | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> <code>Quick Search criterion.</code> <code>String</code> | The name of the <i>COD Category</i> . |

| Name | Type | Description |
|--------------------------------|--|---|
| | | The CMS ingest sets this field to the part of the Genre EPG attribute shared by all <i>Vod Items</i> in the <i>COD Category</i> . |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>promoImage</code> | <code>String</code> | The URL of a promotional image for the <i>Category</i> . Caution! The CMS does not support metadata for <i>COD Categories</i> so this field is not ingested. Deployments wishing to use it must populate it through some other means. |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>status</code> | <code>String</code> | The availability status of the <i>COD Category</i> , either AVAILABLE or UNAVAILABLE . The CMS ingest sets this field to AVAILABLE . |
| <code>statusCode</code> | <i>Deprecated.</i> <code>String</code> | The availability status of the <i>COD Category</i> , expressed as a short code; either A (AVAILABLE) or U (UNAVAILABLE) The CMS ingest sets this field to A . |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |

25.2 codCategoryService

Category Service Implementation. Implementation which performs the compulsory methods defined in the Category Service interface. The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="codCategoryService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true, instrument = false, roles = "<allowed-roles>") The methods used to create, delete or update items of the NVOD System, only could be accessible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different data model could be accessible both the WebServices and the Hue.

Naeem Bilal @author

1.0 @version

25.2.1 codCategoryService.addAsset

A Service method to expose the addition of Asset association with Category..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|-------------|
| categoryUID | Long | Long |
| assetUID | Long | Long |

Return value

Type:

Integer number of records added

Example

```
<addAsset>
  <categoryUID>33</categoryUID>
  <assetUID>93</assetUID>
</addAsset>
```

25.2.2 codCategoryService.addAssetGroup

A Service method to expose the addition of Asset Group association with Category.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|------|-------------|
| categoryUID | Long | Long |
| assetGroupUID | Long | Long |

Return value

Type:

Integer number of records added

Example

```
<addAssetGroup>
  <categoryUID>74</categoryUID>
  <assetGroupUID>46</assetGroupUID>
</addAssetGroup>
```

25.2.3 codCategoryService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|--|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

25.2.4 codCategoryService.create

Create (persist) a **LocaleBaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

25.2.5 codCategoryService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

25.2.6 codCategoryService.getAssetGroupUIDList

A Service method to expose to get the List of Asset Group UIDs relevant to provided Category UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|------|-------------|
| <code>categoryUID</code> | Long | Long |

Return value

Type: `List<Long>`

List of Integer object

Example

```
<getAssetGroupUIDList>
  <categoryUID>12</categoryUID>
</getAssetGroupUIDList>
```

25.2.7 codCategoryService.getAssetUIDList

A Service method to expose to get the List of Asset UIDs relevant to provided Category UID.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|-------------|
| categoryUID | Long | Long |

Return value

Type: List<Long>

List of Integer object

Example

```
<getAssetUIDList>
  <categoryUID>28</categoryUID>
</getAssetUIDList>
```

25.2.8 codCategoryService.getByname

A Service method to get the Category object based on name and locale.

Contains the business logic required to get category multi language details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|---------------------------|
| name | String | - name of the CODCategory |

| Name | Type | Description |
|--------|--------|-------------|
| locale | String | |

Return value

Type: `List<CODCategory>`

CODCategory - List of CODCategory objects

Example

```
<getByName>
  <name>stringValue</name>
  <locale>en_us</locale>
</getByName>
```

25.2.9 codCategoryService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>61</originID>
  <locale>en_us</locale>
</getByOrigin>
```


25.2.10 codCategoryService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>uid</code> | Long | Integer unique id for the <code>LocaleBaseModel</code> object |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

`LocaleBaseModel` if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>50</uid>
  <locale>en_us</locale>
</getByUID>
```

25.2.11 codCategoryService.getCategoryList

A Service method to get the Category details list for locale.

Contains the business logic required to get category details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/getCategoryList`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|-------------|
| <code>locale</code> | String | String |

Return value

Type: `List<CODCategory>`

List of Category object

Example

```
<getCategoryList>
  <locale>en_us</locale>
</getCategoryList>
```

25.2.12 codCategoryService.getCategoryListByItemUidAndType

A Service method to get the List of Category object based on uid, type and locale..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/getCategoryListByItemUidAndType>

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-----------------------------------|
| uid | Long | - uid of the Asset or Asset Group |
| type | String | - Asset or Asset Group (A / G) |
| locale | String | - locale |

Return value

Type: List<Long>

CODCategory - List of CODCategory uid

Example

```
<getCategoryListByItemUidAndType>
  <uid>6</uid>
  <type>stringValue</type>
  <locale>en_us</locale>
</getCategoryListByItemUidAndType>
```

25.2.13 codCategoryService.getMultiLanguageList

A Service method to get the Category multi language details list for uid.

Contains the business logic required to get category multi language details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | Long |

Return value

Type: `List<M>`

List of Category object

Example

```
<getMultiLanguageList>
  <uid>76</uid>
</getMultiLanguageList>
```

25.2.14 codCategoryService.getPagedCategoryList

A Service method to get the Category details list for locale.

Contains the business logic required to get category details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/getPagedCategoryList`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|-------------|
| locale | String | String |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<CODCategory>`

List of Category object

Example

```
<getPagedCategoryList>
  <locale>en_us</locale>
  <startRowNum>70</startRowNum>
```

```
<endRowNum>83</endRowNum>
</getPagedCategoryList>
```

25.2.15 codCategoryService.modifyMultiLanguage

A Service method to expose the modification of category multi language into the system.

Calls the corresponding overloaded method with Category model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/modifyMultiLanguage`

Parameters

| Name | Type | Description |
|-----------------------------|--------|-------------|
| <code>uid</code> | Long | Long |
| <code>locale</code> | String | String |
| <code>name</code> | String | String |
| <code>description</code> | String | String |
| <code>modifiedDateML</code> | Date | Date |

Return value

Type:

Integer number of records modified

25.2.16 codCategoryService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

25.2.17 codCategoryService.removeAsset

A Service method to expose the deletion of Asset association with Category..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/removeAsset`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|-------------|
| categoryUID | Long | Long |
| assetUID | Long | Long |

Return value

Type:

Integer number of records removed

Example

```
<removeAsset>
  <categoryUID>79</categoryUID>
  <assetUID>52</assetUID>
</removeAsset>
```

25.2.18 codCategoryService.removeAssetGroup

A Service method to expose the deletion of Asset Group association with Category..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/removeAssetGroup`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|------|-------------|
| <code>categoryUID</code> | Long | Long |
| <code>assetGroupUID</code> | Long | Long |

Return value

Type:

Integer number of records removed

Example

```
<removeAssetGroup>
  <categoryUID>64</categoryUID>
  <assetGroupUID>39</assetGroupUID>
</removeAssetGroup>
```

25.2.19 codCategoryService.removeMultiLanguage

A Service method to expose the removal of category multi language from the system.

Calls the corresponding overloaded method with Category model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/codCategoryService/removeMultiLanguage`

Parameters

| Name | Type | Description |
|---------------------|--------|-------------|
| <code>uid</code> | Long | Long |
| <code>locale</code> | String | String |

Return value

Type:

Integer number of records removed

25.2.20 codCategoryService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

25.2.21 codCategoryService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

25.2.22 codCategoryService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

26 codHelperService

This *Service* provides specialized APIs which provide VOD series support in certain non-standard deployments.

Deployments using the recommended data model should not need to use this *Service*. The APIs of the *COD Service* provide complete VOD series functionality.

26.1 codHelperService

CodHelperService Interface. Cod Service uses the primitive service to provide user convenience operations across various cod sub system primitive services The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="CodService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Hafeez Ur Rehman @author

Naeem Bilal @author

1.0 @version

26.1.1

codHelperService.hasPurchaseAllAvailableForCatAndTvodAssetGroup

This API returns list of applicable policy groups where there are one or more policy groups (type MUL) associated with all the members of the specified asset group in the specified catalogue node that allows purchase of all episodes belonging to that asset group in that catalogue node.

This API returns a null or empty list if Multiple type Policy group is not available or at least one Episode is not associated with the respective MUL policy group. This API deals only with PolicyGroup type MUL as this is needed to indicate if the purchase of the respective MUL policy Group would buy all the published episodes of the series at the time of purchase. This API will be written in new aggregate CodHelper service. This API is special CNS requirement which has concrete implementation rather than calling generic APIs.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/codHelperService/hasPurchaseAllAvailableForCatAndTvodAssetGroup>

Parameters

| Name | Type | Description |
|---------------|--------|-------------|
| catalogUID | Long | |
| assetGroupUID | Long | |
| locale | String | |

Return value

Type: List<PolicyGroup>

List<PolicyGroup>

27 codScheduleService *New*

This **Service** provides specialized APIs which provide VOD series support in certain non-standard deployments.

Deployments using the recommended data model should not need to use this **Service**. The APIs of the **COD** Service provide complete VOD series functionality.

27.1 codScheduleService *New*

CodHelperService Interface. Cod Service uses the primitive service to provide user convenience operations across various cod sub system primitive services. The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="CodService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

27.1.1 codScheduleService.getSchedulesByPolicyGroupId *New*

Overview

codScheduleService for get the list COD schedule details based on the policyGroupId.

Description

This operation requires as mandatory inputs policyGroupId and returns list of COD schedule details in the response section.

This operation can be called in the following scenarios:

1. If provide valid policyGroupId in the request - operation validates the policy group uid and returns the list of COD schedule details in the response section.
2. For the subsequent if not provide valid policy group uid in the request then operation validates and throws the service exception.
3. If not provided mandatory fields in the request and operation validates and returns the service exception.

Definition

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/codScheduleService/getSchedulesByPolicyGroupId>

Request input parameters

| Name | Type | Description | Cardinality |
|------|------|---------------|-------------|
| arg0 | Long | PolicyGroupId | mandatory |

Requests examples

1. Request with only mandatory field:

GET Method: <http://10.15.2.60/hue-gateway/gateway/http/js/codScheduleService/getSchedulesByPolicyGroupUid?arg0=26&token=Dx611FUeCGWUAxRqqZxerJgvmgddel7kLHN1YnNjcmliZXItMSwXLEtCUk8sMSw2NzQ1Mjg0OSwsLDEsLCxvYW0tZ>

Response example

```
<Response xmlns="http://10.15.2.60/hue-gateway/gateway/http/js/codScheduleService/getSchedulesByPolicyGroupUid">
  <requestId>1173660296</requestId>
  <result>
    <e>
      <creationDate null="true"/>
      <endTime>2082383999000</endTime>
      <exportID null="true"/>
      <itemScheduledId null="true"/>
      <itemType null="true"/>
      <mainScheduleUID null="true"/>
      <modifiedDate null="true"/>
      <originID null="true"/>
      <originKey null="true"/>
      <ownerID null="true"/>
      <privateMetadata null="true"/>
      <serviceProviderID null="true"/>
      <startTime>1348783200000</startTime>
      <status>P</status>
      <uid>1</uid>
    </e>
    <e>
      <creationDate null="true"/>
      <endTime>2082383999000</endTime>
      <exportID null="true"/>
      <itemScheduledId null="true"/>
      <itemType null="true"/>
      <mainScheduleUID null="true"/>
      <modifiedDate null="true"/>
      <originID null="true"/>
      <originKey null="true"/>
      <ownerID null="true"/>
      <privateMetadata null="true"/>
      <serviceProviderID null="true"/>
      <startTime>1348783200000</startTime>
      <status>P</status>
      <uid>1</uid>
    </e>
  </result>
  <resultCode>0</resultCode>
  <token null="true"/>
</Response>
```

28 CODService

This *Service* provides aggregate methods for on-demand *Entities*.

It provides `get` methods which retrieve on-demand *Entities* based on an associated catalogue or schedule.

This *Service* also provides internal methods for subscribing to broadcast *Entities*, but these methods only affect the SDP system. To subscribe to any subscribable *Entity*, clients should use the *Subscription Service* so that the SMS is also notified.

28.1 tv.quative.service.cod.model.PurchasedContent

| Name | Type | Description |
|----------------------------------|---------------------------------|----------------------|
| <code>aclUIDs</code> | <code>List<String></code> | ACL references. |
| <code>actualLicenseString</code> | <code>String</code> | The encoded license. |

28.2 CODService

CodService Interface. Cod Service uses the primitive service to provide user convenience operations across various cod sub system primitive services The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: `@QubeService(name="CodService", hue=true, ws=true)` For the Methods: `@QubeMethod(hue=false, ws=true)` The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Zafar Ali `@author`

Naeem Bilal `@author`

Miguel Gil (NAGRA DVS) `@author`

1.0 `@version`

28.2.1 CODService.getAssetGroupAssetCountByCat

This method returns the number of Assets associated to a Asset Group and Catalogue, Filtered by the Policy Group Types.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--------|-------------|
| catalogueUid | Long | as Long. |
| assetGroupUid | Long | as Long. |
| policyGroupType | String | as String. |
| accountUID | Long | as Long. |

Return value

Type:

Asset Count as Long.

Example

```
<getAssetGroupAssetCountByCat>
<catalogueUid>97</catalogueUid>
<assetGroupUid>12</assetGroupUid>
<policyGroupType>stringValue</policyGroupType>
<accountUID>20</accountUID>
</getAssetGroupAssetCountByCat>
```

28.2.2 CODService.getCatalogueChildCatalogueNodeWrappers

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/CODService/getCatalogueChildCatalogueNodeWrappers>

Parameters

| Name | Type | Description |
|--------------|--------|---------------|
| catalogueUid | Long | the catalogue |
| locale | String | the local |

Return value

Type: `List<CatalogueWrapper>`

a list of catalogue wrapper.

28.2.3 CODService.getCatalogueChildCatalogueNodes

Gets a list of all *Catalogues* which are children of the specified *Catalogue*

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/CODService/getCatalogueChildCatalogueNodes`

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>catalogueUid</code> | Long | The <code>uid</code> of the <i>Catalogue</i> to look in. |
| <code>locale</code> | String | The locale to use for the localizable fields of the returned <i>Catalogues</i> . |

Return value

Type: `List<Catalogue>`

A list of matching *Catalogues* (may be empty.)

28.2.4 CODService.getCataloguesByAccountUID

Gets the *Catalogues* the user is subscribed to..

This API can optionally distinguish *Catalogues* subscribed as individual items from those which are subscribed as part of a *Subscription Profile*.

Setting `recursive` to `false` will restrict the response to *Catalogues* which were subscribed individually. To return all *Catalogues* available to the subscriber, regardless of their status, set `recursive` to `true`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/CODService/getCataloguesByAccountUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---------|--|
| <code>accountUID</code> | Long | The <code>uid</code> of the <i>Account</i> . |
| <code>recursive</code> | boolean | Should <i>Catalogues</i> be included which are subscribed as part of a <i>Subscription Profile</i> ? |

Return value

Type: `List<Catalogue>`

The list of *Catalogues* matching the query.

28.2.5 CODService.getPagedAssetGroupsByCatAndGenre

Gets a page from the list of **Asset Groups** which have at least one **Asset** scheduled into the **Catalogue** and which also match the specified **genre**.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>catalogUID</code> | Long | The <code>uID</code> of the Catalogue to look in. |
| <code>accountUID</code> | Long | The <code>uID</code> of the Account performing the query, if called through Web Services. This is used to determine whether the items returned are already subscribed. HUE clients should not supply this parameter, as the API will use the signed-on Account . |
| <code>genre</code> | String | A comma separated list of genres to include. These will be matched against the <code>genre</code> field of each Asset Group . |
| <code>locale</code> | String | The locale to use for the localizable fields of the returned Asset Groups . |
| <code>startRowNum</code> | Long | The index of the first Asset Group to return from the result list, or null to start at the beginning of the list. |
| <code>endRowNum</code> | Long | The index of the last Asset Group to return from the result list, or null to continue until the end of the list. |

Return value

Type: `List<AssetGroup>`

.

Example

```
<getPagedAssetGroupsByCatAndGenre>
<catalogUID>86</catalogUID>
<accountUID>69</accountUID>
<genre>stringValue</genre>
<locale>en_us</locale>
<startRowNum>93</startRowNum>
<endRowNum>47</endRowNum>
</getPagedAssetGroupsByCatAndGenre>
```


28.2.6 CODService.getPagedCatalogueChildCatalogueNodes

Gets a page from the list of all *Catalogues* with no parent *Catalogue* which are associated with the specified *Access Point*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/CODService/getPagedCatalogueChildCatalogueNodes`

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>catalogueUid</code> | Long | The uID of the <i>Catalogue</i> to look in. |
| <code>locale</code> | String | The locale to use for the localizable fields of the returned <i>Catalogues</i> . |
| <code>startRowNum</code> | Long | The index of the first <i>Catalogue</i> to return from the list. |
| <code>endRowNum</code> | Long | The index of the last <i>Catalogue</i> to return from the list. |

Return value

Type: `List<Catalogue>`

A list of matching *Catalogues* (may be empty.)

28.2.7 CODService.getPagedPolicyForCatAndAssetGroupAndAsset

Returns the list of policies depending upon many way the parameters passed.

This method throw exception on sending Catalogue Uid as null. If Catalogue Uid passed as not null and Asset Group and Asset passed as null then we return all the policy groups of non episodes assets filtered by policy group type. When Asset Group passed as not null and asset as null then policy groups return of all assets of that Asset Group within the Catalogue and filtered by the Policy Group Type. If Asset Group passed as null and Asset not null then policy groups of non episodes assets returns associated with the Catalogue filtered by the policy group type. Policy types can be passed as IMP, MUL, SEP or REC where combination of these can be passed as comma delimited values like "IMP, MUL, SEP". If Policy types passed as null then method returns all the policy groups of all policy types.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--------|--|
| catalogueUid | Long | as Long Catalogue Uid and should not passed as null. |
| assetGroupUid | Long | as Long Asset Group Uid and can be passed as null. |
| assetUid | Long | as Long Asset Uid and can be passed as null. |
| policyGroupType | String | as String and can be passed as null. |
| locale | String | as String and can be passed as null, default locale can be used if pass as null. |
| startRowNum | Long | as Long. |
| endRowNum | Long | as Long. |
| accountUID | Long | as Long. |

Return value

Type: `List<PolicyGroup>`

List of policy groups.

Example

```
<getPagedPolicyForCatAndAssetGroupAndAsset>
<catalogueUid>39</catalogueUid>
<assetGroupUid>15</assetGroupUid>
<assetUid>51</assetUid>
<policyGroupType>stringValue</policyGroupType>
<locale>en_us</locale>
<startRowNum>47</startRowNum>
<endRowNum>1</endRowNum>
<accountUID>52</accountUID>
</getPagedPolicyForCatAndAssetGroupAndAsset>
```

28.2.8 CODService.getSubscribedBaseCatalogues

Gets the list of Base level(with Tree level = 0) catalogues subscribed to the account.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-----------------------|
| locale | String | The catalogue locale. |

Return value

Type: `List<Catalogue>`

List<Catalogue>; list of catalogue subscribed to the account

Example

```
<getSubscribedBaseCatalogues>
  <locale>en_us</locale>
</getSubscribedBaseCatalogues>
```

28.2.9 CODService.getSubscribedPolicyGroups

This method been used to collect all the Policy Groups an Account and Asset has associated with.

1- For purchase type IMP and MUL associated to an account can straight forward as Purchased Item knows it Policy Group. 2- Where REC and SEP types don't know what items has attached to it, so we find Policy Groups from ACL table first for specific account then get the Schedules of all the Policy Groups and then fetch List of CodItem and apply comparison with the provided assetUID. On true comparison we keep the Policy Group.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|-------------|
| assetUid | Long | as Long |
| accountUid | Long | as Long |
| userUid | Long | as Long |
| smartcardID | String | as Long |
| profileUid | Long | as Long |

Return value

Type: `Set<Long>`

plgList as Set

Example

```
<getSubscribedPolicyGroups>
  <assetUId>77</assetUId>
  <accountUId>47</accountUId>
  <userUId>70</userUId>
  <smartcardID>ABC-64094</smartcardID>
  <profileUId>94</profileUId>
</getSubscribedPolicyGroups>
```

28.2.10 CODService.isSubscribed

Is the supplied on-demand content item subscribed to the current *Account*, *User* and/or *Smart Card*?

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/CODService/isSubscribed`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>itemUId</code> | Long | The <code>uID</code> of the content item. |
| <code>itemType</code> | String | The type of the content item. An <i>Acquired Content List AcquiredContentListItemType</i> value: one of <code>AST</code> , <code>AGR</code> , <code>CAT</code> , <code>SUB</code> or <code>PLG</code> . (The <i>CODService</i> does not support other content types.) |
| <code>accountUId</code> | Long | The <code>uID</code> of the <i>Account</i> , or null if not known. (In this case, the <i>Smart Card</i> must be specified.) |
| <code>userUId</code> | Long | The <code>uID</code> of the current <i>User</i> , or null to consider purchases made by any <i>User</i> . |
| <code>smartcardID</code> | String | The <code>uID</code> of the <i>Smart Card</i> , or null to consider purchases made for any <i>Smart Card</i> . (In this case, the <i>Account</i> must be specified.) |
| <code>profileUId</code> | Long | The <code>uID</code> of the <i>Subscription Profile</i> as part of which this item was subscribed, or null to consider purchases made in any manner. |

Return value

Type: `boolean`

Example

```
<isSubscribed>
  <itemUid>35</itemUid>
  <itemType>stringValue</itemType>
  <accountUid>56</accountUid>
  <userId>3</userId>
  <smartcardID>ABC-67256</smartcardID>
  <profileUid>81</profileUid>
</isSubscribed>
```

28.2.11 CODService.isSubscribedSpidKnown

Checks if an item is Subscribed for the account and/or smartcard.

The User id is optional information which should be supplied if there is more than one user for an account and data is expected to be different for those users. Allows for cases where the SPID is not known. Either the AccountUid or SmartcardId must be supplied.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>itemUid</code> | Long | the unique id for the item to be checked for subscription. |
| <code>itemType</code> | String | The type of item. To view the supported types <code>AcquiredContentListItemType</code> . The type "PKG" is not supported in this API |
| <code>accountUid</code> | Long | Account UID. |
| <code>userId</code> | Long | User UID if exists, null otherwise. |
| <code>smartcardID</code> | String | Smartcard ID used for subscription, if exists |
| <code>profileUid</code> | Long | The profile Uid, or null. This can be supplied if the result needs a filter on the basis of subscription profile. |
| <code>spid</code> | long | a predefined SPID for usages where the spid can't be found ie. no session |

Return value

Type: `boolean`

boolean true if subscribed false otherwise.

28.2.12 CODService.subscribeToAssetGroup

Parameters

| Name | Type | Description |
|----------------|--------|--------------|
| accountUid | Long | |
| userId | Long | |
| policyGroupUid | Long | |
| assetGroupUid | Long | |
| itemLocale | String | |
| smartcardID | String | Smartcard ID |

Return value

Type: `List<String>`

List of String UID of the created Acquired Content List entry

28.2.13 CODService.subscribeToPolicyGroup

Subscribes an *Account* and optional *Smart Card* to a *Policy Group* which provides on-demand content.

This method should be used to purchase all types of on-demand *Policy Group*, including transactional and subscription purchases.

You can use this API to subscribe a single smartcard to the *Policy Group* by passing a *Smart Card* ID, or to subscribe the whole *Account* to the *Policy Group* by passing only the *Account*. If you pass both, the smartcard will be subscribed, but SDP will also verify that the smartcard belongs to the requested *Account*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/CODService/subscribeToPolicyGroup`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--------|---|
| <code>accountUid</code> | Long | The <code>uid</code> of the <i>Account</i> . This is required if the <code>smartcardID</code> is not supplied. |
| <code>userId</code> | Long | The <code>uid</code> of the <i>User</i> making the subscription, or null to leave this unspecified. This is used to identify the subscribing user in subscription and billing records: it does not affect who is given access to the content. If a <i>User</i> is specified, it must be linked to the <i>Account</i> supplied, or to the <i>Account</i> associated with the <i>Smart Card</i> supplied (in the case where <code>accountUid</code> is null.) |
| <code>policyGroupUid</code> | Long | The <code>uid</code> of the <i>Policy Group</i> representing the product to subscribe. |
| <code>itemLocale</code> | String | An optional locale code. This determines the locale used for localizable fields in <i>Acquired Content List</i> and <i>Billing Entities</i> created by this subscription. If this field is not supplied, SDP will use its configured default locale. |
| <code>smartcardID</code> | String | The <i>Smart Card</i> ID of the device to give access to the content, or null to leave this unspecified. If specified, the content will be made available to this device only; otherwise, the content will be made available across the whole <i>Account</i> . If this and the <i>Account</i> are both specified, the device must be provisioned to the <i>Account</i> supplied. |

Return value

Type: `List<String>`

A list containing the `uid` of each *Acquired Content List* Entity created to record this subscription.

Example

```
<subscribeToPolicyGroup>
  <accountUid>50</accountUid>
  <userId>59</userId>
  <policyGroupUid>87</policyGroupUid>
  <itemLocale>en_us</itemLocale>
  <smartcardID>ABC-99752</smartcardID>
</subscribeToPolicyGroup>
```

28.2.14 CODService.unsubscribePolicyGroup

Cancels a current subscription to a **Policy Group** which provides on-demand content including transactional (where supported) and subscription purchases.

You can use this API to unsubscribe a single smartcard from the **Policy Group** by passing a **Smart Card ID**, or to unsubscribe the whole **Account** from the **Policy Group** by passing only the **Account**. If you pass both, the smartcard will be unsubscribed, but SDP will also verify that the smartcard belongs to the requested **Account**.

Note

Unsubscribing a **Smart Card** will only affect subscriptions which were made explicitly for that **Smart Card**: if the subscription was made for the whole **Account**, it must be unsubscribed from the whole **Account**

Unsubscribing a whole **Account** will remove the subscription from all associated devices, even if the original subscription was made to the individual **Smart Cards**.

Some subscription products enforce a minimum subscription period. If you unsubscribe a **Policy Group** with a minimum subscription period before that period has finished, SDP will mark the affected subscriptions as pending and unsubscribe them at that point.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--------|---|
| accountUid | Long | The uid of the Account . This is required if the smartcardID is null. |
| policyGrpUid | Long | |
| smartcardID | String | The Smart Card ID to unsubscribe, or null to unsubscribe all devices provisioned to the Account . |

Return value

Type:

The number of updated **Acquired Content List Entities**.

Example

```
<unsubscribePolicyGroup>
<accountUid>85</accountUid>
<policyGrpUid>62</policyGrpUid>
<smartcardID>ABC-58065</smartcardID>
</unsubscribePolicyGroup>
```


29 contentConsumptionService

This *Service* provides the *ContentConsumption* report type, and allows client devices to create records of user actions.

The operator is responsible for defining the set of user actions which should be reported, and for ensuring that client applications correctly notify SDP when an action is performed.

29.1 ContentConsumption

A *ContentConsumption* Entity represents a logged occasion when a content item was viewed by the end user.

29.1.1 ContentConsumption ContentConsumption

| Name | Type | Description |
|----------------------|---|---|
| accountNumber | String | The Account Number linked to the device that consumed the content. |
| action | <i>Report criterion.</i> String | An action linked to the consumed content. E.g. Purchase/Play/Stop/Pause/FFW/REW/Play_Error. |
| casn | String | The CAS serial number of the device that consumed the content. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | |
| consumptionTimestamp | <i>Report criterion.</i> Date | The time at which the device consumed the content. |
| contentTitle | String | The title of the consumed content. |
| creationDate | <i>Report criterion.</i> Date | |
| currency | String | The currency of the consumed content. |
| deviceOriginKey | String | The key of the device that consumed the content. |
| exportID | Integer | |

| Name | Type | Description |
|----------------------------------|-------------------------------|--|
| externalContentId | String | This is an external id that identifies the consumed content. |
| externalProductId | String | This is an external id that identifies the product linked to the consumed content. |
| ipAddress | String | The IP Address of the device that consumed the content. |
| macAddress | String | The MAC Address of the device that consumed the content. |
| mediaPlayerId | String | The Media Player Id of the device that consumed the content. |
| modifiedDate | Date | |
| originID | Integer | |
| originIdAndKey | <i>Read only.</i> String | |
| originKey | String | |
| price | BigDecimal | The price of the consumed content. |
| serviceProviderID | <i>Report criterion.</i> Long | |
| serviceProvider Name | String | The service provider name of the device that consumed the content. |
| serviceProvider OriginKey | String | The service provider key of the device that consumed the content. |
| smartcardId | String | The Smart Card Id of the device that consumed the content. |
| uID | Long | |
| userName | String | The User Name linked to the account of the device that consumed the content. |

29.2 contentConsumptionService

A CRUD Service enabling users to log content consumed by their devices.

lewis @author

29.2.1 contentConsumptionService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | ContentConsumption ContentConsumption <small>p.393</small> | |

Return value

Type:

29.2.2 contentConsumptionService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence. model.BaseModel | |

Return value

Type:

29.2.3 contentConsumptionService.create

Creates a *contentConsumption* record for some action (purchase, playout, trick-play...) which can later be output in a *contentConsumption* report.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/contentConsumptionService/create`

Parameters

| Name | Type | Description |
|-----------------------------------|--------|--|
| <code>userName</code> | String | The username of the viewer on their <i>Account</i> . |
| <code>deviceOriginKey</code> | String | The <code>originKey</code> of the <i>Set Top Box</i> or <i>Media Player</i> , if known. |
| <code>casn</code> | String | The <code>casn</code> of the <i>Set Top Box</i> , if appropriate. |
| <code>macAddress</code> | String | The <code>macAddress</code> of the <i>Set Top Box</i> , if appropriate. |
| <code>smartcardId</code> | String | The <code>smartcardId</code> of the <i>Smart Card</i> , if appropriate. |
| <code>mediaPlayerId</code> | String | The <code>mediaPlayerId</code> of the <i>Media Player</i> , if appropriate. |
| <code>contentTitle</code> | String | The title of the content item which has been purchased or viewed. |
| <code>consumptionTimestamp</code> | Date | The date and time of the activity which is being reported. |
| <code>externalContentId</code> | String | The ID of the content item which has been purchased or viewed on an external system. This normally corresponds to the <code>originKey</code> of the <i>Asset</i> , but operators may define their own IDs if required. |
| <code>externalProductId</code> | String | The ID of the product which has been purchased or viewed on an external system. This normally corresponds to the <code>originKey</code> of the <i>Policy Group</i> , but operators may define their own IDs if required. |
| <code>price</code> | | For a purchase action, the price of the product which was purchased. |
| <code>currency</code> | String | For a purchase action, the currency of the <code>price</code> field. |
| <code>action</code> | String | A name defined by the operator for the action which has been performed (such as <code>PURCHASE</code> , <code>PLAY</code> , <code>PAUSE</code> , <code>REW</code> , <code>FF...</code>) |

Return value

Type:

29.2.4 contentConsumptionService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

29.2.5 contentConsumptionService.getByOrigin

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|-------------|
| originKey | String | |
| originID | Integer | |

Return value

Type:

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>6</originID>
</getByOrigin>
```

29.2.6 contentConsumptionService.getByUID

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | |

Return value

Type:

Example

```
<getByUID>
  <uid>37</uid>
</getByUID>
```

29.2.7 contentConsumptionService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

30 contentRatingService *Updated*

This **Service** allows clients to update the user ratings stored in SDP for content.

Caution! This **Service** should not be called directly, as it only affects ratings stored in SDP. It does not pass this information to an external recommendation engine. Instead, clients should use the **recommendation Service** to rate content.

Caution! ■

30.1 tv.quative.service.contentrating.model.AverageRating

| Name | Type | Description |
|-----------------|------------|--|
| averageRating | BigDecimal | |
| numberOfRatings | Long | |
| originKey | String | Origin key of content this average concerns. |

30.2 contentRatingService

ContentRatingAggregate front end allowing callers to rate content, and return basic statistics based on that content. It is worth noting (for utility and security reasons) that any user/account/spid information is taken from the current context, and not passed in via query.

mlucas @author

30.2.1 contentRatingService.getAverageAssetRating

Retrieves average rating for an asset..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/contentRatingService/getAverageAssetRating>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---------------------|
| <code>originKey</code> | String | - asset identifier. |

Return value

Type:

average rating for asset. - If SDP is configured for like/dislike this will represent a percentage of likes. - If SDP is configured for real valued ratings, it will be the true average. - If the content has no ratings, then a NULL OBJECT with zero average and zero rating count will be returned.

Example

```
<getAverageAssetRating>
  <originKey>stringValue</originKey>
</getAverageAssetRating>
```

30.2.2 contentRatingService.getAverageAssetRatings

Retrieves average ratings for a list of asset originKeys..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------------|---------------------|
| <code>originKeys</code> | List<String> | - asset identifiers |

Return value

Type: List<AverageRating>

a set of average ratings for an asset. - If SDP is configured for like/dislike this will represent a percentage of likes. - If SDP is configured for real valued ratings, it will be the true average. - If the content has no ratings, then a NULL OBJECT with zero average and zero rating count will be returned.

30.2.3 contentRatingService.getAverageProgramRating

Retrieves average rating for a list of program originKeys..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/contentRatingService/getAverageProgramRating`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-----------------------|
| originKey | String | - program identifier. |

Return value

Type:

average rating for program. - If SDP is configured for like/dislike this will represent a percentage of likes. - If SDP is configured for real valued ratings, it will be the true average. - If the content has no ratings, then a NULL OBJECT with zero average and zero rating count will be returned.

Example

```
<getAverageProgramRating>
  <originKey>stringValue</originKey>
</getAverageProgramRating>
```

30.2.4 contentRatingService.getAverageProgramRatings

Retrieves average ratings for a list of originKeys..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------------|---------------------|
| originKeys | List<String> | - asset identifiers |

Return value

Type: List<AverageRating>

a set of average ratings for an asset. - If SDP is configured for like/dislike this will represent a percentage of likes. - If SDP is configured for real valued ratings, it will be the true average. - If the content has no ratings, then a NULL OBJECT with zero average and zero rating count will be returned.

30.2.5 contentRatingService.getUserAssetRatings

Retrieves list of ratings that the currently logged in user has made.

NOTE** USER IS DERIVED FROM THE CONTEXT

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/contentRatingService/getUserAssetRatings>

Parameters

None.

Return value

Type: `List<ContentRating>`

list of ratings.

30.2.6 contentRatingService.rateAsset

Create a new rating for an asset.

User is taken from the context, so no need to explicitly provide. If the user has previously rated this content, then the existing rating will be updated rather than a new separate rating created. NOTE** USER IS DERIVED FROM THE CONTEXT

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/contentRatingService/rateAsset`

Parameters

| Name | Type | Description |
|------------------------|--------|--------------------------|
| <code>originKey</code> | String | - asset identifier |
| <code>score</code> | | - the rating value given |

Return value

Type: `void`

30.2.7 contentRatingService.rateProgram

Create a new rating for a program.

User is taken from the context, so no need to explicitly provide. If the user has previously rated this content, then the existing rating will be updated rather than a new separate rating created. NOTE** USER IS DERIVED FROM THE CONTEXT

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/contentRatingService/rateProgram`

Parameters

| Name | Type | Description |
|------------------------|--------|----------------------|
| <code>originKey</code> | String | - program identifier |

| Name | Type | Description |
|--------------------|------|--------------------------|
| <code>score</code> | | - the rating value given |

Return value

Type: `void`

31 contextService

This *Service* manages the contextual information provided by clients to help SDP return relevant information, and provides APIs for retrieving the contextual information already held by SDP.

Context-aware *Services* take a *Context* object as a parameter (or as a member of a data object passed in as a parameter.) Clients can use this object to pass in any information they deem relevant to the query they want to make. The *contextService* performs two functions:

- ▶ Some fields of the *Context* model object can be used to derive values of other fields. For example, if a **smartcardId** is supplied, the **deviceId** is implied, because a smartcard can only be provisioned to one *Set Top Box*. This logic is performed by the *contextService* on behalf of context-aware *Services*.
- ▶ Depending on the authentication system in use, some contextual information may already be available as a result of the sign-on process. This is called the Request Context. The *contextService* can read the information in the Request Context and convert it into a full *Context* object ready to be used by clients. This functionality can be accessed directly using the API *contextService.getCurrentContext*. Context-aware *Services* may also wrap it into the API they provide, as in *codBrowsingService.getScheduledItemsForCurrentContext* and related methods.

Note

Where a client passes in a *Context* to a context-aware API, the *contextService* uses only this information to derive implicit context. It does not use any information from the request context, except to perform security checks.

31.1 tv.quative.service.profile.model.ProfileType

| Name | Type | Description |
|----------------------|------|---|
| ASSET_PROFILE | | Asset Profile. |
| CHANNEL_PROFILE | | Channel Profile. |
| DEVICE_PROFILE | | Device Profile (applies to Set Top Boxes, Media Players...) |
| PRODUCT_PROFILE | | Product Profile (applies to Policy Groups) |
| SERVICE_PROFILE | | Service Profile, currently not used. |
| SUBSCRIPTION_PROFILE | | Subscription Profile. |

31.2 contextService

31.2.1 contextService.getCurrentContext *Updated*

Get information about the client's context, based on the sign-on information

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/contextService/getCurrentContext`

Available through Web Services.

Parameters

None.

Request:

[http://10.15.0.212/hue-gateway/gateway/http/js/contextService/getCurrentContext?](http://10.15.0.212/hue-gateway/gateway/http/js/contextService/getCurrentContext?token=r9Si1lI2SShHCAaHECK4gbgDShNVQbUILHN1YnNjcmliZXItMTUxLDEsROxPQkFMLDE1MSx6eXhlc0xLCwsMTMxLDEsLG9h)

[token=r9Si1lI2SShHCAaHECK4gbgDShNVQbUILHN1YnNjcmliZXItMTUxLDEsROxPQkFMLDE1MSx6eXhlc0xLCwsMTMxLDEsLG9h](http://10.15.0.212/hue-gateway/gateway/http/js/contextService/getCurrentContext?token=r9Si1lI2SShHCAaHECK4gbgDShNVQbUILHN1YnNjcmliZXItMTUxLDEsROxPQkFMLDE1MSx6eXhlc0xLCwsMTMxLDEsLG9h)

Response:

Type:

A **Context** object containing all the information SDP can derive about the client from the data held in the sign-on token.

Example

```
{ "token": null, "resultCode": "0", "result": { "deviceProfileUid": 5, "deviceUid": 131, "deviceOriginKey": "1", "deviceOriginId": null, "casn": null, "mediaPlayerId": null, "deviceType": "STB", "accountUid": 151, "accountNumber": "zyxel-1", "accountOriginKey": null, "accountOriginId": null, "userId": null, "accessPointUid": 1, "featureUidList": null, "featureNameList": null, "globalSpid": 1, "spid": 1, "locale": "zh_tw", "nlsSort": null, "smartcardId": null, "challengeId": null, "drmInstanceId": null, "deviceTypeList": "{ \"VOD\": [ \"OTT\", \"BTV\" ], \"BTV\": [ \"OTT\", \"DVBNOTT\" ], \"nPVR\": [ \"OTT\", \"DVBNOTT\" ] }", "requestId": 425631100 }
```

31.2.2 contextService.getProfilesForFeatures

Get the profiles for appropriate features..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|---|-------------------------------|
| <code>theContext</code> | <code>tv.quative.service.context.api.Context</code> | Context |
| <code>profileType</code> | <code>tv.quative.service.profile.model.ProfileType</code> | profile type ^{p.404} |

Return value

Type: `List<Long>`

a list of Profile Uids

32 cpvrOperationService

This is a primitive *Service* which provides the *cpvrOperation* Entity.

32.1 tv.quative.service.cpvrOperation.model.CpvrOperation

| Name | Type | Description |
|---------------------|---|--|
| accountUID | Long | Defines the account uid associated to the cpvrOperation. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| channelCall Letters | String | Defines the channel Call Letters associated to the cpvr Operation. |
| channelName | String | Defines the channel Name associated to the cpvr Operation. |
| channelNumber | Integer | Defines the channel Number associated to the cpvr Operation. |
| channelUID | Long | Defines the channel uid associated to the cpvrOperation. |
| creationDate | Date | The date created. |
| eventTitle | String | Defines the event title associated to the cpvrOperation. |
| eventUID | Long | Defines the event uid associated to the cpvrOperation. |
| exportID | Integer | The id to export object to third part systems. |
| locale | String | Defines the user locale to the cpvrOperation. |
| modifiedDate | Date | The date last modified and persisted. |
| originID | Integer | The uid of the origin system (3rd part system). |
| originIdAndKey | String | |

| Name | Type | Description |
|---------------------------|----------------------------------|---|
| originKey | String | The key of the model object in the origin system (3rd part system). |
| recordingEndTime | Date | Defines the recording end time associated to the cpvr Operation. |
| recordingStartTime | Date | Defines the recording start time associated to the cpvr Operation. |
| scheduledEndTime | Date | Defines the recording scheduled end time associated to the cpvrOperation. |
| scheduledStartTime | Date | Defines the recording scheduled start time associated to the cpvrOperation. |
| serviceProviderID | Long | The Service provider UID. |
| source | String | Defines the type of the source media associated to the recording. For eg. 'LS' for local STB, 'RS' for remote STB, 'MB' for Mobile etc. |
| sourceCode | <i>Deprecated.</i> String | |
| status | String | Defines the recording status associated to the cpvrOperation. For eg. C for COMPLETED ,I for INCOMPLETE ,R for RECORDING etc etc |
| statusCode | <i>Deprecated.</i> String | |
| targetDeviceUID | Long | Defines the recording target Device uid associated to the cpvrOperation. |
| type | String | Defines the recording type associated to the cpvr Operation. For eg. SE for SINGLEEVENT ,TB for TIMEBASED etc etc |
| typeCode | <i>Deprecated.</i> String | |
| uID | Long | The unique identifier. |

| Name | Type | Description |
|---------|------|---|
| userUID | Long | Defines the user uid associated to the cpvrOperation. |

32.2 tv.quative.service.cpvrOperation.model.Source

| Name | Type | Description |
|-----------|------|-----------------------------------|
| LOCALSTB | | Code for Local SetTopBox Source. |
| MOBILE | | Code for Mobile Source. |
| REMOTESTB | | Code for Remote SetTopBox Source. |

32.3 tv.quative.service.cpvrOperation.model.Status

| Name | Type | Description |
|------------|------|-----------------------------|
| COMPLETED | | Code for Complete Status. |
| FAILED | | Code for FAILED Status. |
| INCOMPLETE | | Code for INCOMPLETE Status. |
| RECORDING | | Code for RECORDING Status. |
| SCHEDULE | | Code for Schedule Status. |

32.4 tv.quative.service.cpvrOperation.model.Type

| Name | Type | Description |
|-------------|------|-----------------------------------|
| SINGLEEVENT | | Code for Local single event type. |
| TIMEBASED | | Code for Remote time based type. |

32.5 cpvrOperationService

CpvrOperation Service Implementation. Implementation which performs the compulsory methods defined in the CpvrOperationService interface. CpvrOperationService implemented object is used as WebServices to communicate with the Client PVR (CPVR) system The service is made visible to Hue or to WebServices using annotation in the ServiceImplementation class. Ex: For the ServiceClass: @QubeService(name="CPvrOperationService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true, instrument = false, roles = ">allowed-roles<") The methods used to create, delete or update items of the CPVR System, only could be accesible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the Web Services and the Hue.

Binod Singh @author

1.0 @version

32.5.1 cpvrOperationService.delete

A Service method to expose the deletion of CpvrOperation from the system.

Calls the corresponding overloaded method with CpvrOperation model object.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/cpvrOperationService/delete>

Parameters

| Name | Type | Description |
|------|------|------------------------|
| uid | Long | Long CpvrOperation uid |

Return value

Type:

Integer number of records deleted

32.5.2 cpvrOperationService.getByAccount

A Service method to get the CpvrOperation details list for Account uid..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/cpvrOperationService/getByAccount`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------|
| accountUID | Long | Long account uid |

Return value

Type: `List<CpvrOperation>`

List of CpvrOperation object

Example

```
<getByAccount>
  <accountUID>38</accountUID>
</getByAccount>
```

32.5.3 cpvrOperationService.getByAccountVerbose

A Service method to get the CpvrOperation verbose details list for Account uid..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/cpvrOperationService/getByAccountVerbose`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|------------------|
| accountUID | Long | Long account uid |

Return value

Type: `List<CpvrOperation>`

List of CpvrOperation object

Example

```
<getByAccountVerbose>
  <accountUID>53</accountUID>
</getByAccountVerbose>
```

32.5.4 cpvrOperationService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/cpvrOperationService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|------------------------------------|
| <code>uid</code> | Long | Unique id for the BaseModel object |

Return value

Type:

`BaseModel` If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>43</uid>
</getByUID>
```

32.5.5 cpvrOperationService.update

A Service method to expose the updation of `CpvrOperation` into the system.

Calls the corresponding overloaded method with `CpvrOperation` model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/cpvrOperationService/update`

Parameters

| Name | Type | Description |
|-------------------------|------|-------------------------------------|
| <code>uid</code> | Long | Long <code>CpvrOperation</code> uid |
| <code>accountUID</code> | Long | Long account uid |
| <code>userUID</code> | Long | Long user uid |
| <code>eventUID</code> | Long | Long event uid |

| Name | Type | Description |
|---------------------------------|--------|------------------------------------|
| <code>channelUID</code> | Long | Long channel uid |
| <code>targetDeviceUID</code> | Long | Long recording Device uid |
| <code>status</code> | String | String recording status |
| <code>type</code> | String | String recording type |
| <code>source</code> | String | String recording request source |
| <code>scheduledStartTime</code> | Date | Date recording schedule start time |
| <code>scheduledEndTime</code> | Date | Date recording schedule end time |
| <code>recordingStartTime</code> | Date | Date recording event starting time |
| <code>recordingEndTime</code> | Date | Date recording event end time |
| <code>modifiedDate</code> | Date | Date Modified date. |

Return value

Type:

Integer number of records updated

33 deviceAggregateService

This *Service* provides *Set Top Boxes* update and delete methods which integrate with a subscriber management system (SMS).

Calling these methods uses the standard *Device Service* to alter the *Entity* in SDP, but also notifies the SMS where necessary through the *SMSgw Service*.

The `updateSTB` method can be called either with a list of parameters, or with a pre-populated `Model`.

Updating Set Top Boxes

The `updateSTB` method can be called either with a list of parameters, or with a pre-populated model.

If CAS integration is enabled, calling `updateSTB` will automatically notify the CAS about changes that affect it:

- ▶ If `status` is changed from `Enabled` to `Disabled`, the smartcard will be suspended.
- ▶ If `status` is changed from `Disabled` to `Enabled`, the smartcard will be reactivated.
- ▶ If `smartcardID` is changed, the old smartcard is cancelled and the new smartcard is initialized and paired with the set-top box.
- ▶ If `macAddress` or `casN` is changed, the new identity is paired with the smartcard.

Deleting Set Top Boxes

This method deletes the *Set Top Box* entry from the database.

If CAS integration is enabled, calling `deleteSTB` will instruct the CAS to disable the set-top box and cancel the smartcard currently paired with it.

33.1 Media Player

A *Media Player* represents a Nagra Media Player application running on a viewer's PC, set-top box or other device.

It is a kind of *Device*, but may not belong to a *Device Group*. The *Media Player* is created when a subscriber installs and runs the Nagra Media Player application, so it is always associated with an *Account*.

33.1.1 MediaPlayer *Updated*

| Name | Type | Description |
|--------|-----------------|-------------|
| accUID | Read only. Long | |

| Name | Type | Description |
|----------------------|--|--|
| accountUID | Not visible to Web Services. Not visible to HUE. Long | |
| accountUid | Long | The unique identifier of the account to which the media player is associated, a mandatory value. |
| changeLog | Deprecated. Not visible to HUE. List<String> | A log of the changes to this Entity history. |
| creationDate | Date | The date and time this Entity was created. |
| description | String | A short description of this Entity . |
| deviceClassifier UID | Long | Reference to a device classifier. |
| deviceEnabled | Boolean | Is the Device enabled - true/false. |
| statusDate | Boolean | The date that the status changed (from enabled to disabled, or vica-versa) |
| deviceType | String | Device Type. Valid values are: 'STB' - SetTopBox 'APS' - ApplicationServer 'STS' - StreamingServer 'QAM' - Qadrature Amplitude Modulation 'ERM' - Edge Resource Manager (NGOD) 'SCS' - SCS (SBS) 'MP' - Media Player |
| drmClientId | String | |
| drmInstanceName | String | The name of the DRM Instance which represents the DRM system used by this Set Top Box . |
| exportID | Integer | The identifier to use when exporting this Entity to third-party systems. |
| mediaPlayerId | Search criterion. Quick Search criterion. String | The unique identifier for the media player, as generated by SDP. |

| Name | Type | Description |
|--------------------------------|-------------------|---|
| <code>modifiedDate</code> | Date | The date and time this Entity was last modified in the database. |
| <code>name</code> | String | The name of this Entity . |
| <code>originID</code> | Integer | An identifier for the system which owns this Entity . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this Entity uses to identify it. |
| <code>physicalAddress</code> | String | Physical Address of this Device. |
| <code>profileID</code> | Long | Profile ID of this Device. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the Service Provider which created this Entity . |
| <code>uID</code> | Long | The SDP internal UID for this Entity . |
| <code>deviceInfo</code> | String | The device information <pre>{ "device": { "screen": { "height": "1920", "density": "480", "width": "1080" }, "hardware": { "model": "SM-G9006V", "manufacturer": "samsung", "type": "Handheld" }, "OS": { "type": "android", "version": "4.4.2" }, "CPU": { "neon": true, "frequency": "2.4576", "cores": "4" }, "GPU": { "frequency": "0.45", "cores": "32" }, "secureplayer": { "DRMs": ["PRM"], "codecs": ["H264"], "streamings": ["HLS"] } } }</pre> |

33.2 Set Top Box

A *Set Top Box* represents the device that a user interacts with in their home.

Figure 9 SetTopBox and associated Entities.

It is a kind of *Device*, but may not belong to a *Device Group*.

The *Set Top Box* Entity can represent any device which uses the NAGRA Media system to purchase and view content, whether a traditional set-top box, a component of a TV or media system, or a mobile device.

Note

Set Top Boxes are physical devices. Software applications, such as NAGRA Media Player, do not count as *Set Top Boxes*.

Some devices support multiple smart cards, or simultaneous card-based and cardless CAS. Multiple *Set Top Box* Entities may be needed to represent one such device. Contact Nagra for guidance on how to manage this kind of system.

The **banned** flag of *Set Top Boxes* can be managed by bulk ingest using the *banlist* bean.

Set Top Boxes can be ingested and updated from *External Group* assignment files, but only the **serviceProviderId**, **externalGroupId** and **originKey** fields can be specified.

33.2.1 SetTopBox *Updated*

| Name | Type | Description |
|------------|--------------------------|--|
| accUID | Read only. Long | |
| accountUID | Long | The UID of the Account provisioned with this Set Top Box . Note In QSP Manager, this field is set by the Account Number control, which uses the Account number, rather than the UID. |
| banned | Read only. Boolean | Is the device currently banned from accessing media services? Note In QSP Manager, this field is read only. To change it, use the SDP Services , a bulk ingest method, or the separate Ban List Qubelet. |
| caSN | Search criterion. String | The unique serial number, used by the CAS to identify the device. |

| Name | Type | Description |
|----------------------------------|--|---|
| | | <p>In card-based environments, the CA serial number is associated with the smartcard ID during the provisioning process.</p> <p>In NAGRA CAS and PRM environments, the CASN includes a two-digit checksum, which must be included in this field. You should verify with Nagra that the CASN values in your deployment include this checksum.</p> <p>Normally, the <code>casn</code> must be unique across all Set Top Boxes. Some deployments require a physical device to be modelled as multiple Set Top Box Entities, and therefore need the ability for multiple Set Top Boxes to share the same <code>casn</code>. In these deployments, this constraint will have been removed at deployment time.</p> |
| <code>changeLog</code> | Deprecated. Not visible to HUE. <code>List<String></code> | A log of the changes to this Entity history. |
| <code>creationDate</code> | <code>Date</code> | The date and time this Entity was created. |
| <code>description</code> | <code>String</code> | A short description of this Entity . |
| <code>deviceClassifierUID</code> | <code>Long</code> | The Device Classifier identifying the device manufacturer and model. |
| <code>deviceEnabled</code> | <code>Boolean</code> | <p>Is this Set Top Box enabled?</p> <p>Set Top Boxes can be enabled and disabled independently from Accounts. To fully access services, a viewer must have both an active Account and an enabled Set Top Box.</p> |
| <code>deviceType</code> | <code>String</code> | This field is always <code>STB</code> for Set Top Boxes . |
| <code>drmDeviceID</code> | <code>String</code> | This field the <code>mediaPlayerID</code> passed from the MDRM |
| <code>exportID</code> | <code>Integer</code> | The identifier to use when exporting this Entity to third-party systems. |
| <code>externalGroupUID</code> | <code>Long</code> | <p>The <code>uid</code> of the ExternalGroup this Set Top Box belongs to.</p> <p>The ExternalGroup of a Set Top Box corresponds to the PlayerProfile on the Rollout Manager. All devices in</p> |

| Name | Type | Description |
|-----------------------------|---|---|
| | | an ExternalGroup share the same firmware images and upgrade lifecycle. |
| <code>ipAddress</code> | Search criterion. Quick Search criterion. <code>String</code> | The IP address for this Set Top Box , if known. |
| <code>macAddress</code> | Search criterion. Quick Search criterion. <code>String</code> | The hardware address of the device, in standard form: six pairs of upper-case hexadecimal digits, separated by colons. For example: <code>00:1C:26:18:A8:D7</code> |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this Entity was last modified in the database. |
| <code>network</code> | <code>String</code> | A character string identifying the network to which the device belongs, in a format meaningful to the operator. |
| <code>nuID</code> | Search criterion. <code>String</code> | The NUID (Nagra universal identifier) assigned to the device, where it is known by SDP. In deployments using NAGRA Media ACCESS PRM, the NUID (Nagra universal identifier) assigned to the device. Not set in deployments using other types of license server. This field is only used in certain circumstances. In particular, it is not used during the normal license workflow. If the DVS has been configured to require a NUID, the client must supply it when requesting a license via SDP. This field is not used. |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this Entity . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this Entity uses to identify it. |

| Name | Type | Description |
|--------------------------------|--|--|
| <code>physicalAddress</code> | String | The physical address of this Set Top Box . The meaning and format of this field should be decided by the operator. |
| <code>port</code> | String | The UDP port to which RTSP messages for this Set Top Box should be sent. In deployments using RTSP over UDP, SDP uses this field to store the port from which it last received an RTSP message, so that it can reply, if necessary, to the same port. Currently, only switched digital broadcast uses RTSP over UDP. |
| <code>profileID</code> | Long | The Device Profile describing the device's capabilities. |
| <code>provisionID</code> | Long | An optional ID which may be used to identify this Set Top Box during provisioning. |
| <code>provisionPwd</code> | String | An optional password which may be used to authenticate this Set Top Box during provisioning. |
| <code>serialNumber</code> | Search criterion. String | The manufacturer's serial number for the device. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the Service Provider which created this Entity . |
| <code>smartcardID</code> | Search criterion. String | The ID of the Smartcard associated with the device. In NAGRA cardless CAS environments, the smartcard ID includes a two-digit checksum, which must be included in this field. You should verify with Nagra that the smartcard IDs in your deployment include this checksum. Note If the device uses a cardless CAS, depending on the CAS system in use, SDP may store a virtual Smartcard ID in this field. |
| <code>smartcardType</code> | String | The type of smart card system which the device uses. One of CARDBASED or CARDLESS . |
| <code>smartcardTypeEnum</code> | Deprecated. Read only. SmartCardType | The type of smart card system which the device uses, expressed as an enumeration of strings. A tv.quative . |

| Name | Type | Description |
|--------------------------|--------------------------------|---|
| | | <code>service.device.model.SmartCardType</code> value: one of <code>CARDBASED</code> or <code>CARDLESS</code> . |
| <code>status</code> | String | The <code>status</code> of the Account provisioned with the Set Top Box , if known. Caution! This field is used by SDP for temporary storage. It does not represent the status of the Set Top Box . To see if a Set Top Box is enabled, use the <code>deviceEnabled</code> field. |
| <code>statusCode</code> | Deprecated. String | The <code>status</code> of the Account provisioned with the Set Top Box , if known, expressed as a short code. |
| <code>uID</code> | Long | The SDP internal UID for this Entity . |
| <code>validForCAS</code> | Deprecated. Read only. boolean | Does this Set Top Box have values for all the fields required by the Nagra CAS to deliver entitlements? These fields are <code>smartcardID</code> , <code>casN</code> and <code>smartcardType</code> . Caution! This field is no longer recommended for use because its logic relates to the Nagra CAS system only. |
| <code>version</code> | String | |

33.3 tv.quative.service.device.model.SetTopBox.Status

| Name | Type | Description |
|------------------------|------|----------------------------|
| <code>ACTIVE</code> | | Code for ACTIVE Status. |
| <code>CANCELLED</code> | | Code for CANCELLED Status. |
| <code>INACTIVE</code> | | Code for INACTIVE Status. |

| Name | Type | Description |
|------------|------|-----------------------------|
| RESTRICTED | | Code for RESTRICTED Status. |
| SUSPENDED | | Code for SUSPENDED Status. |

33.4 tv.quative.service.device.model.SmartCardType

| Name | Type | Description |
|-------------|------|---|
| CARDBASED | | (CARDBASED) The device uses a traditional CAS with a separate smartcard, identified by the Smartcard ID . |
| CARDLESS | | (CARDLESS) The device uses a cardless CAS. If the device has been provisioned, the Smartcard ID contains the VUA (virtual user-agent) which SDP has assigned to the device. |
| UNSPECIFIED | | (UNSPECIFIED) The SDP does not know whether the device uses a cardless or a cardbased CAS. This is normally because the device has not yet been fully provisioned. The Smartcard ID field is undefined. |

33.5 deviceAggregateService

Aggregate service for the device service.

zlajosa @author

Miguel Gil (BNET Conax Cas) @author

Ashish Gupta @author

Miguel Gil (NMPS) @author

33.5.1 deviceAggregateService.assignSmartcardToCas

Multicas specific provisioning API.

Associates a smartcard with a cas instance. Will also enable the stb.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--------|-------------|
| smartcardID | String | |
| casInstanceName | String | |

Return value

Type: void

Example

```
<assignSmartcardToCas>
<smartcardID>ABC-80861</smartcardID>
<casInstanceName>stringValue</casInstanceName>
</assignSmartcardToCas>
```

33.5.2 deviceAggregateService.createMp

Create, to storage, a MP type **Device** with aggregated functionality..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|------------------------------|
| mp | MediaPlayer ^{p.413} | The Mp data model to persist |

Return value

Type:

The UID of the created Mp

33.5.3 deviceAggregateService.createSTB

Create, to storage, a SetTopBox (STB) type **Device**..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|--|
| stb | SetTopBox ^{p.416} | A model object containing all the fields of the <i>Set Top Box</i> , |

Return value

Type:

The `uid` of the created *Set Top Box*.

33.5.4 deviceAggregateService.deleteMp

Create, to storage, a MP type `Device` with aggregated functionality.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|-------------|
| mpId | String | |

Return value

Type:

The number of rows deleted

Example

```
<deleteMp>
  <mpId>stringValue</mpId>
</deleteMp>
```

33.5.5 deviceAggregateService.deleteSTB

Deletes an STB.

If the CAS integration is configured it sends a suspend/reactivate message to the smartcard if the status has been changed.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| uid | Long | The uid of the Set Top Box to delete. |
| smartcardId | String | The smartcardId of the associated Smart Card . |

Return value

Type:

The number of **Set Top Boxes** affected: either 0 or 1.

Example

```
<deleteSTB>
  <uid>60</uid>
  <smartcardId>stringValue</smartcardId>
</deleteSTB>
```

33.5.6 deviceAggregateService.getvUA *New*

Call from STB to Get the vUA generated and stored by SDP

Available through Web Services.

Parameters

None

Return value

Type: String

String Value of the vUA

Example

```
1000000023
```

33.5.7 deviceAggregateService.swapHybridSTB

Swap the hybrid stb.

It will swap the hybrid stb (will not swap the smartcard)

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|-------------------------|
| accountNumber | String | The account number |
| oldCASN | String | The CASN of the old STB |
| newCASN | String | The CASN of the new STB |

Return value

Type:

The api success status true or false

Example

```
<swapHybridSTB>
  <accountNumber>ABC-39117</accountNumber>
  <oldCASN>stringValue</oldCASN>
  <newCASN>stringValue</newCASN>
</swapHybridSTB>
```

33.5.8 deviceAggregateService.swapSTBOrSmrtCrds

Swaps an STB or a smart card.

The *Set Top Boxes* to swap are identified by the `uid` supplied in the model objects.

SDP uses optimistic locking to ensure that updates do not conflict. You should always get fresh copies of the affected *Set Top Boxes* before swapping them.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--|---|
| stb | SetTopBox ^{p.416} | A model object containing all the fields of the <i>Set Top Box</i> to swap in, replacing the old <i>Set Top Box</i> . |
| oldSTB | SetTopBox ^{p.416} | A model object containing all the fields of the <i>Set Top Box</i> to swap out. |

Return value

Type:

33.5.9 deviceAggregateService.updateDeviceWithNewAccount

Update an MP or STB device with a new account.

For an STB purchases on the old account are revoked from the cas and any existing purchases on the new account are added. For an MP device a straight swap is done after validation. The device to update is inferred from the sign in token.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceAggregateService/updateDeviceWithNewAccount`

Parameters

| Name | Type | Description |
|---------------------------------|--------|--|
| <code>newAccountNumber</code> | String | the new account account-number. |
| <code>oldAccountUsername</code> | String | for validation, current account login details - must be 'SUP' user |
| <code>oldAccountPassword</code> | String | for validation, current account login details |
| <code>newAccountUsername</code> | String | for validation, new account login details - must be 'SUP' user |
| <code>newAccountPassword</code> | String | for validation, new account login details |

Return value

Type:

number of rows affected

33.5.10 deviceAggregateService.updateMp

Update an mp device state (enabled/disabled) in the database.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--|-------------|
| <code>mp</code> | MediaPlayer ^{p.413} | |

Return value

Type:
the number of rows affected

33.5.11 deviceAggregateService.updateMpState

Update an mp device state (enabled/disabled) in the database.
Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|---------|-------------|
| mpId | String | |
| deviceEnabled | Boolean | |

Return value

Type:
the number of rows affected

33.5.12 deviceAggregateService.updateSTB

Update the fields of a *Set Top Box*.

The *Set Top Box* to update is identified by the `uid` supplied in the model object.

SDP uses optimistic locking to ensure that updates do not conflict. You should always get a fresh copy of a *Set Top Box* before making changes.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|--|
| stb | SetTopBox ^{p.416} | A model object containing all the fields of the <i>Set Top Box</i> , updated as necessary. |

Return value

Type:
The number of *Set Top Boxes* affected: either 0 or 1.

33.5.13 deviceAggregateService.updateSmartCard

Update the status of a *Smart Card*.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>uid</code> | Long | The <code>uid</code> of the <i>Smart Card</i> . |
| <code>smartcardID</code> | String | The <code>smartcardId</code> of the <i>Smart Card</i> . |
| <code>status</code> | String | The <code>status</code> of the <i>Smart Card</i> . |

Return value

Type:

The number of *Smart Cards* affected: either 0 or 1.

Example

```
<updateSmartCard>
  <uid>49</uid>
  <smartcardID>ABC-23149</smartcardID>
  <status>stringValue</status>
</updateSmartCard>
```

33.5.14 deviceAggregateService.updateSmartCardWithCasInstanceid

Update, to storage, a SmartCard identified by its primary key (UID).

This is used in this service to ensure the set top box is stripped of its smart card id if the smart card is cancelled

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|----------------------------|
| <code>uid</code> | Long | - The UID of the SmartCard |
| <code>smartcardID</code> | String | - The id of the smart Card |

| Name | Type | Description |
|----------------------------|--------|---|
| <code>status</code> | String | - The status of the smart Card |
| <code>casInstanceId</code> | String | the cas instance id to set, may be null |

Return value

Type:

The Number of rows updated.

Example

```
<updateSmartCardWithCasInstanceId>
  <uid>56</uid>
  <smartcardID>ABC-74873</smartcardID>
  <status>stringValue</status>
  <casInstanceId>stringValue</casInstanceId>
</updateSmartCardWithCasInstanceId>
```

33.5.15 deviceAggregateService.setNetworkBySetTopBox *New*

This API is used to update the network-Id of the SetTopBox. The network-Id is received as a string parameter and updated on the STB which is fetched using the token details. CAS Command: 122 is sent after the update.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--|---|
| <code>network</code> | SetTopBox ^{p.416} tring | Network Id as a String that needs to be updated for the device. |

Return value

Type: void

34 deviceClassifierService *New*

This is a primitive Service which provides the Device Classifier Entity.

34.1 DeviceClassifier *New*

| Name | Type | Description |
|-----------------|---------|---|
| name | String | Platform name |
| deviceKey | String | Regular expression used to identify the user agent |
| manufacturer | String | Manufacturer of the device |
| model | String | Device model |
| os | String | Operating system. |
| modifiedDate | Date | The date and time this Entity was last modified in the database. |
| hardwareVersion | String | Hardware version of the device |
| originID | Integer | An identifier for the system which owns this Entity . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| originIdAndKey | String | A unique reference for the object, derived by joining the originID and originKey , together, with a pipe character to separate them. |
| originKey | String | The key or identifier which the system which owns this Entity uses to identify it. |
| softwareVersion | String | User agent version |
| matchingOrder | Integer | Specified the order in what the matching executed |

| Name | Type | Description |
|-----------|---------|--|
| profileID | Long | Profile ID of this Device |
| uID | Long | The SDP internal UID for this Entity . |
| rules | Integer | List of rules as string for this Device classifier <pre>[{"operator": "\$eq", "key": "OS.type", "value": "Android"}, {"operator": "\$gt", "key": "OS.version", "value": "4.4"}]</pre> |

How to define Rules for a given DeviceInformation :

| Operation | Operator | Rule | DeviceInformation | Value |
|------------------------|------------|---|---|-------|
| EQUALS | \$eq | [{"operator": "\$eq", "key": "OS.type", "value": "Android"}] | {"device": {"OS": {"type": "android", "version": "4.4.2"}} | true |
| LESS_THAN | \$lt | [{"operator": "\$lt", "key": "OS.version", "value": "5.0"}] | {"device": {"OS": {"type": "android", "version": "4.4.2"}} | false |
| LESS_THAN_OR_EQUALS | \$le | [{"operator": "\$le", "key": "OS.version", "value": "4.4"}] | {"device": {"OS": {"type": "android", "version": "4.4.2"}} | true |
| GREATER_THAN | \$gt | [{"operator": "\$gt", "key": "OS.version", "value": "4.4"}] | {"device": {"OS": {"type": "android", "version": "5.0"}} | true |
| GREATER_THAN_OR_EQUALS | \$ge | [{"operator": "\$ge", "key": "OS.version", "value": "5.0"}] | {"device": {"OS": {"type": "android", "version": "4.4.2"}} | false |
| NOT_EQUALS | \$ne | [{"operator": "\$ne", "key": "OS.type", "value": "Windows"}] | {"device": {"OS": {"type": "android", "version": "4.4.2"}} | true |
| CONTAINS | \$contains | [{"operator": "\$contains", "key": "securePlayer.streaming", "value": "HLS"}] | {"secureplayer": {"DRMs": ["PRM"], "codecs": ["H264"], "streamings": ["hls", "ott"]}} | true |

| Operation | Operator | Rule | DeviceInformation | Value |
|--------------|----------------|--|---|-------|
| NOT_CONTAINS | \$not contains | [{"operator": "\$not contains", "key": "securePlayer.streamings", "value": "XYZ"}] | {"secureplayer": {"DRMs": ["PRM"], "codecs": ["H264"], "streamings": ["hls", "ott"]}} | true |

Default data for DeviceClassifier :

SDP will provide default data for DeviceClassifier with the following details :

| COLUMN | VALUE |
|----------------|-------|
| DEVCLASS_UID | 0 |
| DEVCLASS_SPID | 1 |
| CREATION_DATE | |
| MATCHING_ORDER | 1 |
| DEVCLASS_NAME | iOS |
| DEVCLASS_KEY | iOS |
| DEV_MAKE | null |
| DEV_MODEL | null |
| DEV_OS | null |
| DEV_HW_VER | null |
| DEV_SW_VER | null |
| MODIFIED_DATE | null |
| ORIGIN_UID | 0 |

| | |
|------------|--------------------|
| ORIGIN_KEY | defaultDeviceClass |
| EXPORT_KEY | null |
| PROF_UID | 0 |
| RULES | null |

34.2 deviceClassifierService.create *New*

Create, to storage, a new device classifier and return back the generated identifier of the deviceClassifier (UID). Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|---|----------------------------|
| deviceClassifier | DeviceClassifier ^{p.430} | DeviceClassifier to update |

Return value

Type:

UID of the created object.

34.3 deviceClassifierService.delete *New*

Remove a DeviceClassifier from the database.

Parameters

| Name | Type | Description |
|------------------|---|----------------------------|
| deviceClassifier | DeviceClassifier ^{p.430} | DeviceClassifier to delete |

Return value

Type:

Integer as affected rows

35 deviceGroupService

This is a primitive *Service* which provides the *Device Group* Entity.

It manages relationships between *Device Groups* and their constituent *Devices*.

35.1 Device Group

A *Device Group* represents a logical grouping of *Devices*.

Figure 10 DeviceGroup and associated Entities.

A *Device Group* contains *Devices* of the same type (*Application Server*, *Streaming Server*...)

A *Device Group* does not need to represent a physical location. *Device Groups* may be associated with *Access Points* to represent which *Application Servers* and *Streaming Servers* should be used in each region.

Note

The *Streaming Server* holds connection information which set-top boxes may use to set up video streams. Many video server designs use a central "library server" which accepts connections from all set-top boxes and forwards the session on to a local video server. In deployments of this type, only the library server needs to be known to SDP.

35.1.1 DeviceGroup

| Name | Type | Description |
|--------------|---|---|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| deviceuid | Long | device uid of the device. |
| enabled | Boolean | Enabled Boolean value to indicate the status of the Access Point. |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |

| Name | Type | Description |
|--------------------------------|--|---|
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The name of this <i>Entity</i> . |
| <code>networkID</code> | Integer | network ID of the zone (only for QAM zones) |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>sequence</code> | Long | Enabled Boolean value to indicate the status of the Access Point. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |

35.2 devicegroupService

DeviceGroup Service Implementation. Object which implements the business logic of all the methods of the interface. The Device Group is just a reference to a different collection of devices: Application Servers, Gaming Servers, ... The services are made visible to Hue or to WebServices by writing the annotations in the Service Implementation class. Ex: The services are made visible to Hue or to WebServices by writing the annotations in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="devicegroupService", hue = true, ws = true) For the Methods: @QubeMethod(hue=false, ws = true) The methods used to create, delete or update model data only could be accesible to Web Services (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

jraya @author

35.2.1 devicegroupService.addDevice

Contains the business logic to add an Device to an DeviceGroup.

The relationship between Device and DeviceGroup is many to many.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|------|--------------------------|
| deviceuid | Long | Integer device uid |
| devicgroupuid | Long | Integer access point uid |
| sequence | Long | Long sequence |

Return value

Type:

Integer num rows added

Example

```
<addDevice>
  <deviceuid>69</deviceuid>
  <devicgroupuid>76</devicgroupuid>
  <sequence>81</sequence>
</addDevice>
```

35.2.2 devicegroupService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

35.2.3 devicegroupService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

35.2.4 devicegroupService.getAllDeviceGroups

Gets all the access points that are stored in database..

Available through Web Services.

Parameters

None.

Return value

Type: `List<DeviceGroup>`

List of DeviceGroup objects with the information of the access points

Example

```
<getAllDeviceGroups/>
```

35.2.5 devicegroupService.getByAccessPointUID

A Service method to get the device group details list for access point uid.

Contains the business logic required to get device group details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|-------------|
| accessPointUID | Long | Integer |

Return value

Type: List<DeviceGroup>

List of DeviceGroup objects

Example

```
<getByAccessPointUID>
  <accessPointUID>42</accessPointUID>
</getByAccessPointUID>
```

35.2.6 devicegroupService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>36</originID>
</getByOrigin>
```

35.2.7 devicegroupService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>  
<uid>59</uid>  
</getByUID>
```

35.2.8 devicegroupService.getDeviceGroupByDeviceUID

Gets the list of the device groups that contain the specified device.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|------------------------------|
| deviceUID | Long | The deviceUID to search for. |

Return value

Type: **List<DeviceGroup>**

List of DeviceGroup objects with the given device

Example

```
<getDeviceGroupByDeviceUID>  
<deviceUID>4</deviceUID>  
</getDeviceGroupByDeviceUID>
```

35.2.9 devicegroupService.getDevices

Gets all the devices associated to a device.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|------------------------------|
| <code>paccptuid</code> | Long | unique id of the DeviceGroup |

Return value

Type: `List<Device>`

List of Device objects with the information of the devices associated to the device group.

Example

```
<getDevices>
  <paccptuid>86</paccptuid>
</getDevices>
```

35.2.10 devicegroupService.getDevicesByDeviceUID

Gets all the devices associated to a device by device uid.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|------------------------------|
| <code>pdevgrpuid</code> | Long | unique id of the DeviceGroup |
| <code>deviceuid</code> | Long | unique id of the device |

Return value

Type:

List of Device objects with the information of the devices associated to the device group.

Example

```
<getDevicesByDeviceUID>
  <pdevgrpuid>30</pdevgrpuid>
  <deviceuid>34</deviceuid>
</getDevicesByDeviceUID>
```


35.2.11 devicegroupService.getStreamingServerDeviceGroups

Gets all the streaming server device groups that are stored in database.

Available through Web Services.

Parameters

None.

Return value

Type: `List<DeviceGroup>`

List of DeviceGroup objects

Example

```
<getStreamingServerDeviceGroups/>
```

35.2.12 devicegroupService.removeDevice

Contains the business logic to remove an Device to an DeviceGroup.

The relationship between Device and DeviceGroup is many to many.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|--------------------------|
| pdvgrpuid | Long | Integer device uid |
| paccptuid | Long | Integer access point uid |

Return value

Type:

Integer num rows deleted

Example

```
<removeDevice>
  <pdvgrpuid>11</pdvgrpuid>
  <paccptuid>35</paccptuid>
</removeDevice>
```

35.2.13 devicegroupService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

36 deviceService

This is a primitive *Service* which provides the *Device* Entities, including *Application Server*, *Streaming Server*, *Set Top Box* and *QAM*.

Relations between *Devices* and *Accounts* are handled by the *Presence Service*, though in most cases this will not be done directly: instead, the *Provision Service* prepares a *Set Top Box* for use with an *Account*.

Caution!

If you create or update a *Device* using this *Service*, external systems such as an SMS will not be notified. The *DeviceAggregate Service* provides methods which integrate with the SMS.

36.1 Application Server

An *Application Server* represents the device running the server-side application, that a user interacts with through client applications.

Figure 11 ApplicationServer and associated Entities.

It is a kind of *Device*, and normally belongs to one or more *Device Groups*.

36.1.1 ApplicationServer

| Name | Type | Description |
|----------------------|--|--|
| accUID | Read only. Long | |
| changeLog | Deprecated. Not visible to HUE. List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| deviceClassifier UID | Long | The <i>Device Classifier</i> for the server. |
| deviceEnabled | Boolean | Is the <i>Application Server</i> enabled? |
| deviceType | String | The type of device: always APS for <i>Application Servers</i> . |

| Name | Type | Description |
|--------------------------------|---|---|
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>hostname</code> | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The hostname of the host running the server. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>password</code> | String | The password required to access the server. |
| <code>physicalAddress</code> | String | The IP address of the server. |
| <code>port</code> | <i>Search criterion.</i> String | The port number running the server. |
| <code>profileID</code> | Long | The <i>Device Profile</i> describing the server. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>username</code> | <i>Search criterion.</i> String | The user name required to access the server. |

36.2 ERM

An *ERM* (Edge Resource Manager) represents a separate device in a cable environment that allocates transmission resources to clients.

Figure 12 ERM and associated Entities.

It is a kind of *Device*, but cannot belong to a *Device Group*.

An *ERM* should be associated with all the *Service Groups* for which it manages transmission resources.

When a client device requests resources for switched digital broadcast, it specifies the *Service Group* and SDP contacts the corresponding *ERM* to allocate resources. SDP creates *Service Group Resource* Entities as required to represent the resources that the ERM allocates.

Note

Edge Resource Managers are only necessary in deployments where separate systems are in use to allocate resources.

In these deployments, *QAM* Entities are not used, because the management of QAMs is handled by the ERM.

ERM compatibility

Different ERM systems use different variants of the RTSP protocol to communicate with SDP.

Because of this, SDP needs to know the specific type of ERM system you are using. This is done by creating a suitable *Device Classifier* and associating it with the *Edge Resource Manager* Entity.

36.2.1 Erm

| Name | Type | Description |
|-----------------|--|---|
| accUID | Read only. Long | |
| accessPointName | String | Name of the access point. Used during bulk-ingest only. The name will be resolved to a UID by the ingest service. |
| accessPointUID | Long | Foreign Key attribute associating an Access Point with this Qam. |
| changeLog | Deprecated. Not visible to HUE. List<String> | A log of the changes to this Entity history. |
| creationDate | Date | The date and time this Entity was created. |
| description | String | A short description of this Entity. |

| Name | Type | Description |
|-----------------------------|--|---|
| deviceClassifierName | String | Name of the device classifier. Used during bulk-ingest only. The name will be resolved to a UID by the ingest service. |
| deviceClassifierUID | Long | Reference to a device classifier. |
| deviceEnabled | Boolean | Is the Device enabled - true/false. |
| deviceType | String | Device Type. Valid values are: 'STB' - SetTopBox 'APS' - ApplicationServer 'STS' - StreamingServer 'QAM' - Qadrature Amplitude Modulation 'ERM' - Edge Resource Manager (NGOD) 'SCS' - SCS (SBS) 'MP' - Media Player |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |
| name | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The name of this <i>Entity</i> . |
| originID | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| originIdAndKey | <i>Read only.</i> String | A unique reference for the object, derived by joining the originID and originKey , together, with a pipe character to separate them. |
| originKey | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| physicalAddress | String | Physical Address of this Device. |
| profileID | Long | Profile ID of this Device. |
| serviceProviderID | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |

| Name | Type | Description |
|------|------|---|
| uID | Long | The SDP internal UID for this <i>Entity</i> . |

36.3 QAM

A *QAM* (Quadrature Amplitude Modulator) represents the device in a cable environment that transmits media content to clients.

Figure 13 QAM and associated Entities.

It is a kind of *Device*, but cannot belong to a *Device Group*. Instead, *QAMs* are grouped into *Zones*.

A physical *QAM* contains a number of cards, each of which transmits a modulated signal to a group of clients. These cards are represented in SDP as *Service Groups*.

QAM ingestion

SDP can use RTSP to get information about *QAMs* from their MIB (Management Information Block). However, *QAMs* do not all behave the same way. To be able to query the MIB of a *QAM*, you must tell SDP what kind of *QAM* it is. This is done by creating a suitable **Device Classifier** and associating it with the *QAM*.

36.3.1 Qam

| Name | Type | Description |
|----------------|---|--|
| accUID | <i>Read only.</i> Long | |
| accessPointUID | Long | Foreign Key attribute associating an Access Point with this Qam. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |

| Name | Type | Description |
|----------------------------|---|---|
| deviceClassifierUID | Long | Reference to a device classifier. |
| deviceEnabled | Boolean | Is the Device enabled - true/false. |
| deviceType | String | Device Type. Valid values are: 'STB' - SetTopBox 'APS' - ApplicationServer 'STS' - StreamingServer 'QAM' - Qadrature Amplitude Modulation 'ERM' - Edge Resource Manager (NGOD) 'SCS' - SCS (SBS) 'MP' - Media Player |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |
| name | <i>Search criterion.</i> String | The name of this <i>Entity</i> . |
| originID | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| originIdAndKey | <i>Read only.</i> String | A unique reference for the object, derived by joining the originID and originKey , together, with a pipe character to separate them. |
| originKey | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| physicalAddress | String | Physical Address of this Device. |
| profileID | Long | Profile ID of this Device. |
| qamAddress | <i>Search criterion. Quick Search criterion.</i> String | The physical address of the EdgeQam. |
| serviceProviderID | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |

| Name | Type | Description |
|------|------|---|
| uID | Long | The SDP internal UID for this <i>Entity</i> . |

36.4 Simulcrypt Synchronizer

An *SCS* (Simulcrypt Synchronizer) represents a device in a cable environment that manages on-the-fly scrambling of video sessions according to the Simulcrypt standard.

Figure 14 SCS and associated Entities.

Some manufacturers produce *SCS* devices as stand-alone units; others build one or more *SCS* devices into a *QAM*. Both types are treated as a separate device in SDP, so a physical *QAM* unit may be represented in SDP as a collection of *QAM* and *SCS* Entities. There is no direct connection between these *Entities*; they are related through the *Service Groups* that they collectively manage.

An *SCS* Entity is a kind of *Device*, but cannot belong to a *Device Group*.

An *SCS* should be associated with all the *Service Groups* for which it scrambles video sessions.

36.4.1 Scs

| Name | Type | Description |
|-------------------------|---|--|
| accUID | <i>Read only.</i> Long | |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| deviceClassifier UID | Long | Reference to a device classifier. |
| deviceEnabled | Boolean | Is the Device enabled - true/false. |
| deviceType | String | Device Type. Valid values are: 'STB' - SetTopBox 'APS' - ApplicationServer 'STS' - StreamingServer 'QAM' - |

| Name | Type | Description |
|--------------------------------|--|---|
| | | Qadrature Amplitude Modulation 'ERM' - Edge Resource Manager (NGOD) 'SCS' - SCS (SBS) 'MP' - Media Player |
| <code>exportID</code> | <code>Integer</code> | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>physicalAddress</code> | <code>String</code> | Physical Address of this Device. |
| <code>profileID</code> | <code>Long</code> | Profile ID of this Device. |
| <code>scsAddress</code> | <i>Search criterion.</i> <code>String</code> | Ip address. |
| <code>scsName</code> | <i>Search criterion. Quick Search criterion.</i> <code>String</code> | |
| <code>scsPort</code> | <i>Search criterion.</i> <code>Integer</code> | |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |

36.5 Streaming Server

A *Streaming Server* represents the device that streams media content to client-side applications.

Figure 15 StreamingServer and associated Entities.

It is a kind of *Device*, and normally belongs to one or more *Device Groups*.

36.5.1 StreamingServer

| Name | Type | Description |
|----------------------------------|---|---|
| <code>accUID</code> | <i>Read only.</i> Long | |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created. |
| <code>description</code> | String | A short description of this <i>Entity</i> . |
| <code>deviceClassifierUID</code> | Long | Reference to a device classifier. |
| <code>deviceEnabled</code> | Boolean | Is the Device enabled - true/false. |
| <code>deviceType</code> | String | Device Type. Valid values are: 'STB' - SetTopBox 'APS' - ApplicationServer 'STS' - StreamingServer 'QAM' - Qadrature Amplitude Modulation 'ERM' - Edge Resource Manager (NGOD) 'SCS' - SCS (SBS) 'MP' - Media Player |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |

| Name | Type | Description |
|--------------------------------|---|--|
| <code>originIdAndKey</code> | <i>Read only.</i> <code>string</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>string</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>physicalAddress</code> | <code>string</code> | Physical Address of this Device. |
| <code>platformType</code> | <code>string</code> | Platform type. |
| <code>platformVersion</code> | <code>string</code> | Platform version. |
| <code>profileID</code> | <code>Long</code> | Profile ID of this Device. |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>streamingAddress</code> | <i>Search criterion.</i> <i>Quick Search criterion.</i> <code>string</code> | Address of Server. |
| <code>streamingPort</code> | <i>Search criterion.</i> <code>string</code> | Port that the Server runs on. |
| <code>streamingProtocol</code> | <i>Search criterion.</i> <code>string</code> | Protocol used for the video stream. |
| <code>streamingSecure</code> | <code>Boolean</code> | Is the stream secure. |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |
| <code>uploadAddress</code> | <code>string</code> | Address used for upload. |
| <code>uploadPort</code> | <code>string</code> | Port used for upload. |
| <code>uploadPwd</code> | <code>string</code> | Password for upload. |
| <code>uploadSecure</code> | <code>Boolean</code> | Is the upload secure. |

| Name | Type | Description |
|-------------------------|---------------------|----------------------|
| <code>uploadUser</code> | <code>String</code> | Username for upload. |

36.6 tv.quative.service.device.model.DeviceStatus

| Name | Type | Description |
|-----------------------|------|------------------|
| <code>DISABLED</code> | | Disabled devices |
| <code>ENABLED</code> | | Enabled only |

36.7 deviceService

Provides all of the required functionality to Create, Retrieve, Update and Delete 'Device' entities on the system.

Steve @author

Miguel (Scs, Nmpps) @author

36.7.1 deviceService.createAppServer

Create, to storage, an ApplicationServer type `Device`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|--|---|
| <code>applicationServer</code> | ApplicationServer ^{p.443} | The ApplicationServer data model to persist |

Return value

Type:

The UID of the created ApplicationServer

36.7.2 deviceService.createErm

Create, to storage, a Erm type `Device`..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|-------------------------------|
| erm | Erm <small>p.445</small> | The Erm data model to persist |

Return value

Type:

The UID of the created Erm

36.7.3 deviceService.createQam

Create, to storage, a Qam type **Device**..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|-------------------------------|
| qam | Qam <small>p.447</small> | The Qam data model to persist |

Return value

Type:

The UID of the created Qam

36.7.4 deviceService.createSTB

Create, to storage, a SetTopBox (STB) type **Device**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--|-------------------------------------|
| setTopBox | SetTopBox <small>p.416</small> | The SetTopBox data model to persist |

Return value

Type:

The UID of the created STB

36.7.5 deviceService.createSTBWithServiceProvider

Create, to storage, a SetTopBox (STB) type **Device**.

with the provided service provider ID

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--|-------------------------------------|
| <code>setTopBox</code> | SetTopBox ^{p.416} | The SetTopBox data model to persist |
| <code>serviceProvider</code> | Long | Service providerID. |

Return value

Type:

The UID of the created STB

36.7.6 deviceService.createScs

Create, to storage, a Scs type **Device**..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--------------------------------------|-------------------------------|
| <code>scs</code> | Scs ^{p.449} | The Scs data model to persist |

Return value

Type:

The UID of the created Scs

36.7.7 deviceService.createStreamingServer

Create, to storage, an ApplicationServer type **Device**..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--|---|
| <code>streamingServer</code> | StreamingServer ^{p.451} | The ApplicationServer data model to persist |

Return value

Type:

The UID of the created ApplicationServer

36.7.8 deviceService.deleteDeviceByUID

Delete, from storage, a `Device` identified by its primary key (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|--|
| <code>type</code> | String | The type of <code>Device</code> to delete |
| <code>uid</code> | Long | The primary key (UID) of the <code>Device</code> to delete |

Return value

Type:

The number of rows deleted

Example

```
<deleteDeviceByUID>
  <type>stringValue</type>
  <uid>12</uid>
</deleteDeviceByUID>
```

36.7.9 deviceService.getAvailableSetTopBox

Gets a page from the list of all *Set Top Boxes* which are ready to be provisioned to an *Account*.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|---------|-------------------------------------|
| <code>startRow</code> | Integer | The index of the start of the page. |
| <code>endRow</code> | Integer | The index of the end of the page. |

Return value

Type: `List<SetTopBox>`

A list of *Set Top Boxes* within the page (may be empty.)

Example

```
<getAvailableSetTopBox>
  <startRow>19</startRow>
  <endRow>69</endRow>
</getAvailableSetTopBox>
```

36.7.10 deviceService.getBySerialNumber

Retrieve, from storage, a `SetTopBox` identified by its Serial Number..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|---|
| <code>serialNumber</code> | String | The serialNumber of the <code>SetTopBox</code> to retrieve. |

Return value

Type: `List<SetTopBox>`

`List<SetTopBox>` If found, the matching list object, empty list otherwise.

Example

```
<getBySerialNumber>
  <serialNumber>ABC-32653</serialNumber>
</getBySerialNumber>
```

36.7.11 deviceService.getBySmartcardIDAnyServiceProvider

Retrieve, from storage, a `SetTopBox` identified by its SmartcardID. **Warning:** This API is for internal use only.

The current service provider UID is not taken into account which is a potential security risk in an MSP environment if this method is exposed to a service gateway.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>smartcardID</code> | String | The Smartcard ID of the <code>SetTopBox</code> to retrieve |

Return value

Type:

`SetTopBox` If found, the matching object, null otherwise.

Example

```
<getBySmartcardIDAnyServiceProvider>
  <smartcardID>ABC-38443</smartcardID>
</getBySmartcardIDAnyServiceProvider>
```

36.7.12 deviceService.getDeviceByCASN

Gets the *Set Top Box* with the specified `casn`.

As installed, SDP requires the `casn` field to be unique across *Set Top Boxes*, so there will never be more than one object in the returned list. However, some deployments require this constraint to be removed. Clients of an SDP which allows duplicate `casn` may see multiple objects in the list, and will need to choose between them using their own criteria.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/getDeviceByCASN`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|--------------------------------------|
| <code>casn</code> | String | The CASN of the <i>Set Top Box</i> . |

Return value

Type: `List<SetTopBox>`

A list containing the matching *Set Top Box*, or an empty one if none is found.

Example

```
<getDeviceByCASN>
  <casn>stringValue</casn>
</getDeviceByCASN>
```

36.7.13 deviceService.getDeviceByCASNAnyServiceProvider

Retrieve, from storage, **SetTopBox** list identified by its CASN. *Warning:* This API is for internal use only.

The current service provider UID is not taken into account which is a potential security risk in an MSP environment if this method is exposed to a service gateway.

As installed, SDP requires the **casn** field to be unique across *Set Top Boxes*, so there will never be more than one object in the returned list. However, some deployments require this constraint to be removed. Clients of an SDP which allows duplicate **casn** may see multiple objects in the list, and will need to choose between them using their own criteria.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| casn | String | The CASN of the SetTopBox to retrieve |

Return value

Type: **List<SetTopBox>**

SetTopBox If found, the matching object, null otherwise.

Example

```
<getDeviceByCASNAnyServiceProvider>
  <casn>stringValue</casn>
</getDeviceByCASNAnyServiceProvider>
```

36.7.14 deviceService.getDeviceByDevGrpAndType

Retrieve, from storage, a **Device** identified by the device group id and device type..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|----------------------------|
| devgrpId | Long | The id of the device group |

| Name | Type | Description |
|-------------------|--------|--------------------------------|
| <code>type</code> | String | The type of Device to retrieve |

Return value

Type: `List<Device>`

Device If found, the matching object, null otherwise.

Example

```
<getDeviceByDevGrpAndType>
  <devgrpId>15</devgrpId>
  <type>stringValue</type>
</getDeviceByDevGrpAndType>
```

36.7.15 deviceService.getDeviceByMACAddress

Gets the *Set Top Box* with the specified `macAddress`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/getDeviceByMACAddress`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>macAddress</code> | String | The MAC address of the <i>Set Top Box</i> . |

Return value

Type:

The matching *Set Top Box*, or null if none is found.

Example

```
<getDeviceByMACAddress>
  <macAddress>stringValue</macAddress>
</getDeviceByMACAddress>
```

36.7.16 deviceService.getDeviceByMACAddressAnyServiceProvider

Retrieve, from storage, a `SetTopBox` identified by its MAC address. *Warning:* This API is for internal use only.

The current service provider UID is not taken into account which is a potential security risk in an MSP environment if this method is exposed to a service gateway.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>macAddress</code> | String | The MAC address of the <code>SetTopBox</code> to retrieve |

Return value

Type:

`SetTopBox` If found, the matching object, null otherwise.

Example

```
<getDeviceByMACAddressAnyServiceProvider>
  <macAddress>stringValue</macAddress>
</getDeviceByMACAddressAnyServiceProvider>
```

36.7.17 deviceService.getDeviceByNUID

Retrieve, from storage, `SetTopBox` list identified by its NUID..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/getDeviceByNUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--------|--|
| <code>pNuid</code> | String | The NUID of the <code>SetTopBox</code> to retrieve |

Return value

Type:

`SetTopBox` If found, the matching object, null otherwise.

Example

```
<getDeviceByNUID>
  <pNuid>stringValue</pNuid>
</getDeviceByNUID>
```

36.7.18 deviceService.getByPhysicalAddress

Gets the Device *Entity* with the specified `physicalAddress`.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|--|
| <code>type</code> | String | The type of device to retrieve. A <code>DeviceType</code> value: one of <code>APS</code> , <code>ERM</code> , <code>MP</code> , <code>QAM</code> , <code>SCS</code> , <code>STB</code> or <code>STS</code> . |
| <code>address</code> | String | The <code>physicalAddress</code> of the Device <i>Entity</i> . |

Return value

Type:

The matching device, as a model object of the type specified; null if none is found or if the Device *Entity* with the specified `uid` was of a different type.

Example

```
<getDeviceByPhysicalAddress>
  <type>stringValue</type>
  <address>stringValue</address>
</getDeviceByPhysicalAddress>
```

36.7.19 deviceService.getBySmartcardID

Retrieve, from storage, a `SetTopBox` identified by its SmartcardID..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/getDeviceBySmartcardID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>smartcardID</code> | String | The Smartcard ID of the <code>SetTopBox</code> to retrieve |

Return value

Type:

SetTopBox If found, the matching object, null otherwise.

Example

```
<getDeviceBySmartcardID>
  <smartcardID>ABC-42822</smartcardID>
</getDeviceBySmartcardID>
```

36.7.20 deviceService.getDeviceByType

Retrieve, from storage, all the **Devices** of a given type..

DeviceType @see

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|--------------------------------|
| type | String | The type of Device to retrieve |

Return value

Type: List<Device>

Device If found, the matching object, null otherwise.

Example

```
<getDeviceByType>
  <type>stringValue</type>
</getDeviceByType>
```

36.7.21 deviceService.getDeviceByUID

Gets the Device *Entity* with the specified uid.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|--|
| type | String | The type of device to retrieve. A DeviceType value: one of APS , ERM , MP , QAM , SCS , STB or STS . |

| Name | Type | Description |
|------------------|------|--|
| <code>uid</code> | Long | The <code>uid</code> of the Device <i>Entity</i> . |

Return value

Type:

The matching device, as a model object of the type specified; null if none is found or if the Device *Entity* with the specified `uid` was of a different type.

Example

```
<getDeviceByUID>
  <type>stringValue</type>
  <uid>1</uid>
</getDeviceByUID>
```

36.7.22 deviceService.getErmByName

Obtain ERMs by name.

Names should usually be unique, although this is not enforced in the DB.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|------------------------|
| <code>name</code> | String | The name to search on. |

Return value

Type: `List<Erm>`

The ERMs for that name, if any.

Example

```
<getErmByName>
  <name>stringValue</name>
</getErmByName>
```

36.7.23 deviceService.getErmByUID

Obtain ERM from its UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-----------------|
| UID | Long | UID of the ERM. |

Return value

Type:

Erm.

Example

```
<getErmByUID>
  <UID>20</UID>
</getErmByUID>
```

36.7.24 deviceService.getMediaPlayerByAccountUID

Retrieves the list of Media Players belonging to the given account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------------|
| accountUID | Long | UID of the account |

Return value

Type: List<MediaPlayer>

List<MediaPlayer> If found, the matching object, null otherwise.

Example

```
<getMediaPlayerByAccountUID>
  <accountUID>42</accountUID>
</getMediaPlayerByAccountUID>
```

36.7.25 deviceService.getMediaPlayerByAccountUIDAndStatus

Retrieves the list of Media Players belonging to the given account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--|--|
| accountUID | Long | UID of the account |
| status | tv.quative.service.device.model.DeviceStatus | The device status (enabled/disabled usually) |

Return value

Type: `List<MediaPlayer>`

List<MediaPlayer>; If found, the matching object, null otherwise.

36.7.26 deviceService.getMpById

Obtain Mp by its id.

Id is unique, and enforced in the DB.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/getMpById`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|----------------------------------|
| id | String | The id to identify media player. |

Return value

Type:

The MediaPlayer model object, if exist.

Example

```
<getMpById>
  <id>stringValue</id>
</getMpById>
```

36.7.27 deviceService.getNextVirtualSmartcardId

Gets the next virtual smartcard ID..

Available through Web Services.

Parameters

None.

Return value

Type:

Next virtual smartcard ID.

Example

```
<getNextVirtualSmartcardId/>
```

36.7.28 deviceService.getPrivateData

Retrieve the private data associated with a device..

#setPrivateData() @see

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/getPrivateData`

Parameters

| Name | Type | Description |
|----------|------|--|
| deviceId | Long | The ID of the device against which to retrieve data. |

Return value

Type:

36.7.29 deviceService.getQams

Retrieve, from storage all **Qam** entities..

Available through Web Services.

Parameters

None.

Return value

Type: `List<Qam>`

List If found, the matching objects, null otherwise.

Example

```
<getQams/>
```

36.7.30 deviceService.getSetTopBoxByAccountUID

Gets all the *Set Top Boxes* provisioned to the specified *Account*

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---------------------------------|
| accountUID | Long | The uid of the <i>Account</i> . |

Return value

Type: List<SetTopBox>

A list of matching *Set Top Boxes* (may be empty.)

Example

```
<getSetTopBoxByAccountUID>
  <accountUID>88</accountUID>
</getSetTopBoxByAccountUID>
```

36.7.31 deviceService.getSetTopBoxByAccountUIDAndSpid

Retrieves the list of STBs belonging to the given account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------------|
| accountUID | Long | UID of the account |
| spid | Long | spid to use |

Return value

Type: List<SetTopBox>

List<SetTopBox>; If found, the matching object, null otherwise.

Example

```
<getSetTopBoxByAccountUIDAndSpid>
  <accountUID>91</accountUID>
  <spid>96</spid>
</getSetTopBoxByAccountUIDAndSpid>
```

36.7.32 deviceService.getSetTopBoxByExternalGroup

Retrieve a list of **SetTopBox** objects identified by their external group UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|------|---------------------------------------|
| <code>externalGroupUid</code> | Long | the external group we are looking for |

Return value

Type: `List<SetTopBox>`

List of SetTopBox objects that match this UID.

Example

```
<getSetTopBoxByExternalGroup>
  <externalGroupUid>56</externalGroupUid>
</getSetTopBoxByExternalGroup>
```

36.7.33 deviceService.getSetTopBoxByOrigin

Gets a set top box given an Origin Id and key..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|-----------------------------|
| <code>originId</code> | Integer | Origin identifier |
| <code>originKey</code> | String | Unique external identifier. |

Return value

Type:

set top box

Example

```
<getSetTopBoxByOrigin>
```

```
<originId>24</originId>
<originKey>stringValue</originKey>
</getSetTopBoxByOrigin>
```

36.7.34 deviceService.getSwapSTBsForAccountUID

Retrieve, from storage, list of `SetTopBox`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|---------|-------------|
| <code>accountUID</code> | Long | |
| <code>startRow</code> | Integer | |
| <code>endRow</code> | Integer | |

Return value

Type: `List<SetTopBox>`

Device If found, the matching object, null otherwise.

Example

```
<getSwapSTBsForAccountUID>
<accountUID>69</accountUID>
<startRow>7</startRow>
<endRow>72</endRow>
</getSwapSTBsForAccountUID>
```

36.7.35 deviceService.setPrivateData

Sets the value of a private data string associated with a device, aimed mainly at MediaPlayer variants.

The SDP will not place any structure or validation on the data stored, nor transform it in any way. It will only validate that the device ID matches the signed-on device.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/setPrivateData`

Parameters

| Name | Type | Description |
|----------|--------|---|
| deviceId | Long | The ID of the device against which to store data. |
| data | String | The data to store. |

Return value

Type:

An empty string, indicating a successful storage of the data.

36.7.36 deviceService.updateAppServer

Update, to storage, an ApplicationServer type **Device** identified by its primary key (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--|---|
| appserver | ApplicationServer ^{p.443} | The ApplicationServer data model to persist |

Return value

Type:

The UID of the created ApplicationServer

36.7.37 deviceService.updateErm

Update an ERM **Device**..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------------------------------------|--------------------|
| erm | Erm ^{p.445} | The ERM to update. |

Return value

Type:

The number of rows modified.

36.7.38 deviceService.updateMpName

Update an Mp Device name..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/updateMpName`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|----------------------------------|
| <code>mpId</code> | String | The media player id to identify. |
| <code>name</code> | String | The media player name to update. |

Return value

Type:

The number of rows modified.

Example

```
<updateMpName>
  <mpId>stringValue</mpId>
  <name>stringValue</name>
</updateMpName>
```

36.7.39 deviceService.updateMpState

Update an Mp Device state (enabled / disabled)..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/deviceService/updateMpState`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|----------------------------------|
| <code>mpId</code> | String | The media player id to identify. |

| Name | Type | Description |
|----------------------------|---------|--|
| <code>deviceEnabled</code> | Boolean | The media player device enabled state to update. |

Return value

Type:

The number of rows modified

36.7.40 `deviceService.updateQam`

Update, to storage, a Qam type `Device` identified by its primary key (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--------------------------------------|-------------------------------|
| <code>qam</code> | Qam ^{p.447} | The Qam data model to persist |

Return value

Type:

The UID of the created Qam

36.7.41 `deviceService.updateSTB`

Update, to storage, a STB type `Device` identified by its primary key (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--|-------------------------------------|
| <code>stb</code> | SetTopBox ^{p.416} | The SetTopBox data model to persist |

Return value

Type:

The UID of the created STB

36.7.42 deviceService.updateSTBWithServiceProvider

Update, to storage, a SetTopBox (STB) type **Device**.

with the provided service provider ID

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|--|-------------------------------------|
| <code>stb</code> | SetTopBox ^{p.416} | The SetTopBox data model to persist |
| <code>serviceProviderID</code> | Long | Service providerID. |

Return value

Type:

The row count effected.

36.7.43 deviceService.updateScs

Update, to storage, a Scs type **Device** identified by its primary key (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--------------------------------------|-------------------------------|
| <code>scs</code> | Scs ^{p.449} | The Scs data model to persist |

Return value

Type:

The UID of the created Scs

36.7.44 deviceService.updateStreamingServer

Update, to storage, a StreamingServer type **Device** identified by its primary key (UID)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|--|---|
| <code>streamingserver</code> | StreamingServer ^{p.451} | The ApplicationServer data model to persist |

Return value

Type:

The UID of the created ApplicationServer

37 drmworkflowservice

This is an aggregate *Service* which allows clients to obtain authentication tokens from the Marlin DRM system.

It provides one method, which checks the client is entitled to view the requested content, and returns a Marlin MS3 token.

SDP does not obtain the token itself, but requests it from a Multi-DRM proxy. This is a separate component of the deployment which must be installed and integrated with SDP and with Marlin. For details of this component, see Multi-DRM Proxy Installation Guide, [MDRMPROXY-INS](#) and Multi-DRM Proxy API Reference, [MDRMPROXY-API](#)

Note

The Marlin DRM has multiple workflows, some of which use different types of token. The current Nagra solution supports only the MS3 token workflow.

37.1 drmworkflowservice

The `DrmWorkflowService` allows Marlin clients to retrieve Marlin MS3 tokens.

rrajendram @author

37.1.1 drmworkflowservice.getMarlinMs3Token

Get a Marlin MS3 token.

The Marlin client must have previously signed-on using `signonByUserMpIdAndSpid(String userLogin, String password, String spOriginKey, String mpId)`, and must be subscribed to view the specified content.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/drmworkflowservice/getMarlinMs3Token`

Parameters

| Name | Type | Description |
|---------------------------|--------|--|
| <code>drmContentId</code> | String | The unique content identifier used by Marlin. This must be supplied. |
| <code>keyId</code> | String | Optionally, the unique identifier for the content key. |

Return value

Type:

The Marlin MS3 token.

38 dynamicAdvertsService

This *Service* manages the advert insertion points associated with *Assets* and allows clients to retrieve their details.

It provides a single API, *dynamicAdvertsService.getAdverts*, which returns a list of the *AdInsertionTags* associated with a specified *Asset*.

Clients are expected to pass this information on to an advertising platform to determine which advertisement(s) should be placed at each insertion point.

38.1

tv.quative.service.dynamicads.api.DynamicAdvertsResponse

| Name | Type | Description |
|---------|---------------------------|--|
| adverts | List<AdInsertionTag Type> | A list of <i>AdInsertionTag</i> objects describing the points in the <i>Asset</i> where advertising may be inserted. |

38.2 dynamicAdvertsService

mphoenix @author

38.2.1 dynamicAdvertsService.getAdverts

Retrieve Dynamic Adverts for an *Asset*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/dynamicAdvertsService/getAdverts`

Parameters

| Name | Type | Description |
|-----------|---------|-------------------------|
| originId | Integer | Asset Origin Identifier |
| originKey | String | Asset Origin Key |

Return value

Type:

DynamicAdvertsResponse Dynamic Adverts

39 EdaConfigurationService

This *Service* is used to interact with the EDA CAS in use at Foxtel. It should not be used in other situations.

39.1 FoxtelEdaCas

This *Entity* represents the EDA CAS used by Foxtel.

39.1.1 FoxtelEdaCas

| Name | Type | Description |
|-----------------------------|---------------------------------|--|
| <code>casAddress</code> | String | |
| <code>casDescription</code> | <i>Search criterion.</i> String | a description |
| <code>casName</code> | <i>Search criterion.</i> String | User friendly identifier for the cas. Used for nothing but simple user identification in UI searches and such. |
| <code>casType</code> | CasType | discriminator |
| <code>casUid</code> | Long | PK |
| <code>createDate</code> | Date | Create date |
| <code>modifiedDate</code> | Date | last modified date (updated automatically by an entity Listener). |
| <code>port</code> | int | |
| <code>requestTimeout</code> | int | |
| <code>secretKey</code> | String | |

39.2 EdaConfigurationService

39.2.1 EdaConfigurationService.create

Create a Foxtel EDA instance..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|-------------|
| casModel | FoxtelEdaCas <small>p.478</small> | |

Return value

Type:

39.2.2 EdaConfigurationService.get

Get a Foxtel EDA instance..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | |

Return value

Type:

Example

```
<get>
  <uid>48</uid>
</get>
```

39.2.3 EdaConfigurationService.remove

Remove a Foxtel EDA instance..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|-------------|
| casModel | FoxtelEdaCas ^{p.478} | |

Return value

Type: void

39.2.4 EdaConfigurationService.update

Update a Foxtel EDA instance..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|-------------|
| casModel | FoxtelEdaCas ^{p.478} | |

Return value

Type: void

40 eventService

This is a primitive *Service* which provides the BTV *Event* Entity.

40.1 Event

An *Event* represents a broadcast transmission, a *Program* being shown on a *Channel* at a particular time.

Figure 16 Event and associated Entities.

Events have no metadata apart from the start and end time. Details of the content being broadcast are stored in the *Program* Entity.

Events can have a *Policy Group*, which marks them as available for pay-per-view. If an *Event* does not have a *Policy Group*, it is treated as freely available to all users who have access to the *Channel* through a *Package*. Clients can query SDP to find out details of the *Events* scheduled to be broadcast on a particular *Channel*, or on the *Channels* within a *Package*. This is normally done to display an EPG (Electronic Program Guide) to the end user.

Events can also be marked as available for time-shifting: either start-over or catch-up, or both. Start-over and catch-up are features which can be added to a subscription. If a user has the start-over or catch-up features enabled, they can watch the content at some time after the *Event*: before it has finished, in the case of "start-over", or afterwards, in the case of "catch-up". The content is delivered by a video server, like a VOD *Asset*. Users without time-shifting enabled can still watch the *Event* on its *Channel* as it is being broadcast.

Caution!

In systems using a CMS, *Events* are created automatically by the ingest process. On-demand content and pricing should be defined in the CMS.

The CMS ingest creates *Events* from *EPG Units*.

40.1.1 Event

| Name | Type | Description |
|----------------------------|--|--|
| <code>assetFileName</code> | <code>String</code> | The file name on the video server of the program being broadcast . This field is required for <i>Events</i> which need to be stored on the video server, such as start-over or catch-up <i>Events</i> . The CMS ingest places the <i>Programme</i> object's <code>soctv_filename</code> EPG attribute in this field. |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | A log of the changes to this <i>Entity</i> history. |

| Name | Type | Description | | | | | | | | | | |
|---------------------|--|--|-------|-------------|--------|--|----|--|----|--|------|--|
| channelUID | Long | The uid of the <i>Channel</i> on which the program is being broadcast. The CMS ingest uses the <i>DownloadPeriod</i> object's serviceRef attribute to link the appropriate <i>Channel</i> . | | | | | | | | | | |
| creationDate | Date | The date and time this <i>Entity</i> was created. | | | | | | | | | | |
| definition | String | | | | | | | | | | | |
| endTime | Search criterion. Date | The end time of the broadcast event. The CMS ingest places the end attribute of the <i>Programme</i> 's child <i>Period</i> in this field, or works it out from the start and duration attributes. | | | | | | | | | | |
| eventType | String | The event type. One of: <table border="1" data-bbox="794 1010 1401 1496"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>(null)</td> <td>This <i>Event</i> may only be viewed as a regular broadcast event.</td> </tr> <tr> <td>so</td> <td>This <i>Event</i> is available for start-over viewing.</td> </tr> <tr> <td>cu</td> <td>This <i>Event</i> is available for catch-up viewing.</td> </tr> <tr> <td>soCU</td> <td>This <i>Event</i> is available for both start-over and catch-up viewing.</td> </tr> </tbody> </table> <p>The CMS ingest uses the <i>Programme</i> object's isStartOver and isCatchUp attributes to set this field.</p> | Value | Description | (null) | This <i>Event</i> may only be viewed as a regular broadcast event. | so | This <i>Event</i> is available for start-over viewing. | cu | This <i>Event</i> is available for catch-up viewing. | soCU | This <i>Event</i> is available for both start-over and catch-up viewing. |
| Value | Description | | | | | | | | | | | |
| (null) | This <i>Event</i> may only be viewed as a regular broadcast event. | | | | | | | | | | | |
| so | This <i>Event</i> is available for start-over viewing. | | | | | | | | | | | |
| cu | This <i>Event</i> is available for catch-up viewing. | | | | | | | | | | | |
| soCU | This <i>Event</i> is available for both start-over and catch-up viewing. | | | | | | | | | | | |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. | | | | | | | | | | |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. | | | | | | | | | | |

| Name | Type | Description |
|-----------------------------------|----------------------------------|---|
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>ownerId</code> | String | |
| <code>pVPPurchase</code> | <i>Read only.</i> boolean | This field is deprecated and should not be used. An <i>Event</i> represents a pay-per-view (PPV) event if (and only if) it has a <i>Policy Group</i> . Clients should check the <code>policyGroupUID</code> field to discover PPV <i>Events</i> . |
| <code>parentAclUID</code> | Long | This field should not be used. |
| <code>parentPolicyGroupUID</code> | Long | This field should not be used. |
| <code>policyGroupUID</code> | Long | The <i>Policy Group</i> for this <i>Event</i> , if it represents a pay-per-view event. A pay-per-view <i>Event</i> must always have a <i>Policy Group</i> , so that viewers will be able to subscribe to it. The CMS ingest creates <i>Policy Groups</i> from the information in the referenced <i>Products</i> and uses it in this field. |
| <code>policyID</code> | Long | This field should not be used. |
| <code>profileUID</code> | Long | This field should not be used. |
| <code>programUID</code> | Long | The <code>uid</code> of the <i>Program</i> being broadcast. The CMS ingest links the <i>Program</i> created from the associated <i>Programme</i> object. |

| Name | Type | Description |
|-----------------------------------|-------------------------------|--|
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>startTime</code> | <i>Search criterion.</i> Date | The start time of the broadcast event. |
| <code>subscribableItemType</code> | String | This field is always EVT for <i>Events</i> . |
| <code>subscribed</code> | <i>Read only.</i> boolean | Is the user already subscribed to this item? This field is returned as part of the <i>Event</i> model object when clients query <i>Events</i> . It is not part of the <i>Event</i> Entity itself. |
| | | Note Attempting to set this field will have no effect. To subscribe to an <i>Event</i> , use the dedicated subscription APIs. |
| <code>totalRequests</code> | Long | |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |

40.2 tv.quative.service.event.model.DetailedEvent

| Name | Type | Description |
|----------------------------|---|-------------|
| <code>assetFileName</code> | String | |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | |
| <code>channelUID</code> | Long | |
| <code>creationDate</code> | Date | |
| <code>definition</code> | String | |

| Name | Type | Description |
|--------------------------|---------|-------------------|
| endTime | Date | Search criterion. |
| eventName | String | |
| eventRating | Long | |
| eventType | String | |
| exportID | Integer | |
| modifiedDate | Date | |
| originID | Integer | |
| originIdAndKey | String | |
| originKey | String | |
| ownerId | String | |
| pVPPurchase | boolean | |
| parentAclUID | Long | |
| parentPolicyGroup UID | Long | |
| policyGroupUID | Long | |
| policyID | Long | |
| profileUID | Long | |
| programUID | Long | |
| serviceProviderID | Long | |

| Name | Type | Description |
|------------------------------------|------------------------|-------------|
| <code>shortDescription</code> | String | |
| <code>startTime</code> | Search criterion. Date | |
| <code>subscribableItem Type</code> | String | |
| <code>subscribed</code> | boolean | |
| <code>totalRequests</code> | Long | |
| <code>uID</code> | Long | |

40.3 eventService

Schedule Service Implementation. Manages the business logic of all the operations with Schedules and Events. Object which implements the business logic of all the methods of the interface. A Schedule is a group of Events, and every event has a relationship between one program, one channel, and the time to start and end. It is composed of two different model elements: Event with simple information, and the DetailedEvent which includes some information retrieved from the program model (name, rating, ...). The services are made visible to Hue or to WebServices by writing the annotations in the Service Implementation class. Ex: For the ServiceClass:

hue=true, ws=true) For the Methods: `@QubeService(name="eventService",`

`ws=true)` The methods used to create, delete or update items of the BTV System, only could be accessible to Web Services (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue. `@QubeMethod(hue=false,`

`jraya @author`

40.3.1 eventService.addPolicyGroup

Add Policy group to an event.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|-------------|
| <code>eventUId</code> | Long | Event UID |

| Name | Type | Description |
|----------------|------|------------------|
| policyGroupUid | Long | Policy Group UID |

Return value

Type:

Number of added rows

Example

```
<addPolicyGroup>
  <eventUid>95</eventUid>
  <policyGroupUid>18</policyGroupUid>
</addPolicyGroup>
```

40.3.2 eventService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|-------------|
| model | Event ^{p-481} | |

Return value

Type:

40.3.3 eventService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

40.3.4 eventService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

40.3.5 eventService.getByMediaFileName

Gets the list of events for a given Asset File name.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|----------------------|
| <code>mediaFileName</code> | String | The Asset File name. |

Return value

Type: `List<Event>`

List of events for the asset file name.

Example

```
<getByMediaFileName>
  <mediaFileName>stringValue</mediaFileName>
</getByMediaFileName>
```


40.3.6 eventService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/eventService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>93</originID>
</getByOrigin>
```

40.3.7 eventService.getByPolicyGroupUID

Gets the list of events assigned to the given policy group.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|------|--------------|
| <code>policyGroupUID</code> | Long | Policy group |

Return value

Type: `List<Event>`

List of events assigned to the policy group

Example

```
<getByPolicyGroupUID>
  <policyGroupUID>42</policyGroupUID>
</getByPolicyGroupUID>
```

40.3.8 eventService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/eventService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>91</uid>
</getByUID>
```

40.3.9 eventService.getDetEvtByUID

Gets the Detailed Event object by the DetailedEvent unique id.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------|---------------------------------|
| pevtuid | Long | Integer unique id of the event. |

| Name | Type | Description |
|---------------------|--------|--|
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |

Return value

Type:

DetailedEvent with the information retrieved by the query.

Example

```
<getDetEvtByUID>
  <pevtuid>68</pevtuid>
  <locale>en_us</locale>
</getDetEvtByUID>
```

40.3.10 eventService.getDetEvtsByAllChnls

Search the Schedule information(all Events objects)for all the existing channels in an specific time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for this time range.

40.3.11 eventService.getDetEvtsByCatUID

Search the Schedule information(all Events objects)for the Category.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|--|
| catuid | Long | unique category uid. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |

Return value

Type: List<DetailedEvent>

List list of Event objects retrieved for all the channels.

40.3.12 eventService.getDetEvtsByChnUID

Gets all the detailed events for a specific channel.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|--|
| pchnuid | Long | uid of the Event. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the channel uid specified.

40.3.13 `eventService.getDetEvtsByChnlAcct`

Gets all the detailed events for all the channels subscribed by an Account.

The detailed events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pacctuid</code> | Long | Integer unique id of the user account. |
| <code>pstartdate</code> | Date | start date of the range of detailed events we want to get. |
| <code>penddate</code> | Date | end date of the range of detailed events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type: `List<DetailedEvent>`

List of detailed events with the information retrieved by the query.

40.3.14 `eventService.getDetEvtsByPkgAcct`

Gets all the detailed events for all the packages subscribed by an Account.

The detailed events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|------|--|
| <code>pacctuid</code> | Long | Integer unique id of the user account. |

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pstartdate</code> | Date | start date of the range of detailed events we want to get. |
| <code>penddate</code> | Date | end date of the range of detailed events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type: `List<DetailedEvent>`

List of detailed events with the information retrieved by the query.

40.3.15 `eventService.getDetEvtsByPkgPattern`

Gets all the detailed events for a package pattern (hyphen separated list of package UIDs).

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>ppkgPattern</code> | String | "-" separated list of package uid-s. |
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the package uid specified.

40.3.16 eventService.getDetEvtsByPkgUID

Gets all the detailed events for a specific package.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>ppkguid</code> | Long | uid of the package. |
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the package uid specified.

40.3.17 eventService.getDetEvtsMultiLanguageList

Gets a list of DetailedEvent objects with the multilanguage information for a program unique id.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|--|
| <code>uid</code> | Long | Integer unique origin id for customer representation |

Return value

Type: `List<DetailedEvent>`

List of DetailedEvent objects with the verbose information

Example

```
<getDetEvtsMultiLanguageList>
<uid>52</uid>
```

</getDetEvtsMultiLanguageList>

40.3.18 eventService.getEvtsByAllChnls

Search the Schedule information(all Events objects)for all the existing channels in an specific time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---|
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |

Return value

Type: List<Event>

List list of Event objects retrieved for this time range.

40.3.19 eventService.getEvtsByCatUID

Search the Schedule information(all Events objects)for the Category.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---|
| catuid | Long | unique category uid. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |

Return value

Type: List<Event>

List list of Event objects retrieved for all the channels.

40.3.20 eventService.getEvtsByChnUID

Gets all the events for a specific channel.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---|
| pchnuid | Long | uid of the Event. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |

Return value

Type: `List<Event>`

List list of Event objects retrieved for the channel uid specified.

40.3.21 eventService.getEvtsByChnlAcct

Gets all the events for all the channels suscribed by an Account.

The events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|---|
| pacctuid | Long | Long unique id of the user account. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |

Return value

Type: `List<Event>`

List of events with the information retrieved by the query.

40.3.22 eventService.getEvtsByOriginUID

Gets the list of Event objects using the external system's id.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|--|
| originuid | Integer | Integer unique origin id for customer representation |

Return value

Type: List<Event>

List of Event objects

Example

```
<getEvtsByOriginUID>
  <originuid>36</originuid>
</getEvtsByOriginUID>
```

40.3.23 eventService.getEvtsByPgmUID

Search all the Events for a given Program.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| pgmuid | Long | - unique program uid. |
| locale | String | Locale to which the program belongs to |
| type | String | event type whether NVOD, COD or BTV |

Return value

Type: List<Event>

List of Event objects retrieved for all the program id

Example

```
<getEvtsByPgmUID>
```

```
<pgmuid>88</pgmuid>
<locale>en_us</locale>
<type>stringValue</type>
</getEvtsByPgmUID>
```

40.3.24 eventService.getEvtsByPkgAcct

Gets all the events for all the packages suscribed by an Account.

The events retrieved depends on a time range

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|------|---|
| pacctuid | Long | Integer unique id of the user account. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |

Return value

Type: `List<Event>`

List of events with the information retrieved by the query.

40.3.25 eventService.getEvtsByPkgUID

Gets all the events for a specific package.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|------|---|
| ppkguid | Long | uid of the package. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |

Return value

Type: `List<Event>`

List list of Event objects retrieved for the package uid specified.

40.3.26 `eventService.getPagedDetailEventsByAllChnls`

Search the Schedule information(all Events objects)for all the existing channels in an specific time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for this time range.

40.3.27 `eventService.getPagedDetailEventsByCatUID`

Search the Schedule information(all Events objects)for the Category.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|------|----------------------|
| <code>catuid</code> | Long | unique category uid. |

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for all the channels.

40.3.28 `eventService.getPagedDetailEventsByChnUID`

Gets all the detailed events for a specific channel.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>pchnuid</code> | Long | uid of the Event. |
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

| Name | Type | Description |
|-------------|------|-------------|
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the channel identifier specified.

40.3.29 eventService.getPagedDetailEventsByChnlAcct

Gets all the detailed events for all the channels subscribed by an Account.

The detailed events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pacctuid | Long | Integer unique id of the user account. |
| pstartdate | Date | start date of the range of detailed events we want to get. |
| penddate | Date | end date of the range of detailed events we want to get. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<DetailedEvent>`

List of detailed events with the information retrieved by the query.

40.3.30 eventService.getPagedDetailEventsByPkgAcct

Gets all the detailed events for all the packages subscribed by an Account.

The detailed events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pacctuid | Long | Integer unique id of the user account. |
| pstartdate | Date | start date of the range of detailed events we want to get. |
| penddate | Date | end date of the range of detailed events we want to get. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: List<DetailedEvent>

List of detailed events with the information retrieved by the query.

40.3.31 eventService.getPagedDetailEventsByPkgPattern

Gets all the detailed events for a package pattern (hyphen separated list of package UIDs).

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--------------------------------------|
| ppkgPattern | String | "-" separated list of package uid-s. |

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the package uid specified.

40.3.32 `eventService.getPagedDetailEventsByPkgUID`

Gets all the detailed events for a specific package.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--|
| <code>ppkguid</code> | Long | uid of the package. |
| <code>pstartdate</code> | Date | start date of the range of events we want to get. |
| <code>penddate</code> | Date | end date of the range of events we want to get. |
| <code>locale</code> | String | String with the language of the channel we want to retrieve. |
| <code>type</code> | String | String program type |

| Name | Type | Description |
|-------------|------|-------------|
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the package uid specified.

40.3.33 eventService.getPagedDetailEventsMultiLanguageList

Gets a list of DetailedEvent objects with the multilanguage information for a program unique id.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|--|
| uid | Long | Integer unique origin id for customer representation |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<DetailedEvent>`

List of DetailedEvent objects with the verbose information

Example

```
<getPagedDetailEventsMultiLanguageList>
  <uid>29</uid>
  <startRowNum>62</startRowNum>
  <endRowNum>94</endRowNum>
</getPagedDetailEventsMultiLanguageList>
```

40.3.34 eventService.getPagedEventsByAllChnls

Search the Schedule information(all Events objects)for all the existing channels in an specific time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: List<Event>

List list of Event objects retrieved for this time range.

40.3.35 eventService.getPagedEventsByCatUID

Search the Schedule information(all Events objects)for the Category.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| catuid | Long | unique category uid. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: List<Event>

List list of Event objects retrieved for all the channels.

40.3.36 eventService.getPagedEventsByChnUID

Gets all the events for a specific channel.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| pchnuid | Long | uid of the Event. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: List<Event>

List list of Event objects retrieved for the channel uid specified.

40.3.37 eventService.getPagedEventsByChnAcct

Gets all the events for all the channels subscribed by an Account.

The events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| pacctuid | Long | Long unique id of the user account. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| startRowNum | Long | Long |

| Name | Type | Description |
|-----------|------|-------------|
| endRowNum | Long | Long |

Return value

Type: `List<Event>`

List of events with the information retrieved by the query.

40.3.38 eventService.getPagedEventsByPgmUID

Search all the Events for a given Program.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pgmuid | Long | - unique program uid. |
| locale | String | Locale to which the program belongs to |
| pgmtype | String | Program type whether NVOD, COD or BTv |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<Event>`

List of Event objects retrieved for all the program id

Example

```
<getPagedEventsByPgmUID>
  <pgmuid>5</pgmuid>
  <locale>en_us</locale>
  <pgmtype>stringValue</pgmtype>
  <startRowNum>59</startRowNum>
  <endRowNum>91</endRowNum>
</getPagedEventsByPgmUID>
```

40.3.39 eventService.getPagedEventsByPkgAcct

Gets all the events for all the packages subscribed by an Account.

The events retrieved depends on a time range.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| pacctuid | Long | Integer unique id of the user account. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<Event>`

List of events with the information retrieved by the query.

40.3.40 eventService.getPagedEventsByPkgUID

Gets all the events for a specific package.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| ppkguid | Long | uid of the package. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| startRowNum | Long | Long |

| Name | Type | Description |
|-----------|------|-------------|
| endRowNum | Long | Long |

Return value

Type: `List<Event>`

List list of Event objects retrieved for the package uid specified.

40.3.41 `eventService.getPagedSOCUEventsByChnIdentifier`

Gets all the detailed events for a specific channel.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pidentifier | String | chanIdentifier of the Event. |
| pstartdate | Date | start date of the range of events we want to get. |
| penddate | Date | end date of the range of events we want to get. |
| locale | String | String with the language of the channel we want to retrieve. |
| type | String | String program type |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<DetailedEvent>`

List list of Event objects retrieved for the channel uid specified.

40.3.42 `eventService.getPpvEventPolicyGroupByEvent`

Gets the policy group uid for an event.

An event will have policy group iff it is a ppv event.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|-------------|
| event | Event ^{p.481} | |

Return value

Type:

Long the policygroupUid if it has one otherwise null

40.3.43 eventService.removePolicyGroup

Remove Policy Group from an event.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|------------------|
| eventUid | Long | Event UID |
| policyGroupUid | Long | Policy Group UID |

Return value

Type:

Number of deleted rows

Example

```
<removePolicyGroup>
  <eventUid>69</eventUid>
  <policyGroupUid>16</policyGroupUid>
</removePolicyGroup>
```

40.3.44 eventService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|-------------|
| model | Event ^{p.481} | |

Return value

Type:

40.3.45 eventService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows updated.

41 favouriteService

This is a primitive *Service* which provides the *Favourite* Entity.

41.1 Favourite

The *Favourite* Entity.

No content

41.1.1 Favourite

| Name | Type | Description |
|-----------------------------|--|---|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>itemType</code> | <code>String</code> | Defines the item type associated to the favourite. For eg. 'C' for Channel, 'P' for Program etc. |
| <code>itemTypeCode</code> | <i>Deprecated.</i> <code>String</code> | |
| <code>itemUID</code> | <code>Long</code> | Defines the item uid associated to the favourite. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object derived by concatenating the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |

| Name | Type | Description |
|--------------------------------|---------|---|
| <code>prioritySeqNo</code> | Integer | Defines the priority sequence number associated to the favourite. |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>uID</code> | Long | The unique identifier. |
| <code>userUID</code> | Long | Defines the user uid associated to the favourite. |

41.2 favouriteService

Favourite Service Implementation. Implementation which performs the compulsory methods defined in the Favourite Service interface. FavouriteService implemented object is used as WebServices to communicate with the Account Management (AM) system The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: `@QubeService(name="assetService", hue=true, ws=true)` For the Methods: `@QubeMethod(hue=false, ws=true, instrument = false, roles = "<allowed-roles>")` The methods used to create, delete or update items of the AM System, only could be accesible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Ashish Gupta `@author`

1.0 `@version`

41.2.1 favouriteService.create *Updated*

A Service method to expose the creation of favourite object into the system.

Calls the corresponding overloaded method with Favourite model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/create`

Parameters

| Name | Type | Description |
|-----------------------|--------|---|
| <code>userUID</code> | Long | Long user uid |
| <code>itemType</code> | String | String item type. A - Asset, C - Channel, CA - Catalogue E - Event, P - Program, R - Rating |

| Name | Type | Description |
|---------------|---------|--|
| itemUID | Long | Long item uid |
| prioritySeqNo | Integer | Integer priority number |
| originKey | String | String origin key i.e. client identifier |
| originUID | Integer | Integer origin uid |

Return value

Type:

Long favourite uid

41.2.2 favouriteService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

41.2.3 favouriteService.delete

A Service method to expose the deletion of favourite from the system.

Calls the corresponding overloaded method with Favourite model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/delete`

Parameters

| Name | Type | Description |
|------|------|--------------------|
| uid | Long | Long favourite uid |

Return value

Type:

Integer number of records deleted

41.2.4 favouriteService.delete

Delete, from storage, a `BaseModel`.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

41.2.5 favouriteService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|---|
| originKey | String | String the primary key of the record in a customer system |

| Name | Type | Description |
|----------|---------|--|
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>3</originID>
</getByOrigin>
```

41.2.6 favouriteService.getByOriginUID

A Service method to get the favourite details list for origin uid.

Contains the business logic required to get favourite details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/getByOriginUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|--------------------|
| originUID | Integer | Integer origin uid |
| userID | Long | Long user uid |

Return value

Type: `List<Favourite>`

List of Favourite object

Example

```
<getByOriginUID>
  <originUID>85</originUID>
  <userID>58</userID>
</getByOriginUID>
```

41.2.7 favouriteService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|------------------------------------|
| <code>uid</code> | Long | Unique id for the BaseModel object |

Return value

Type:

`BaseModel` If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>24</uid>
</getByUID>
```

41.2.8 favouriteService.getFavouriteItemList

A Service method to get the favourite item list.

Contains the business logic required to get favourite details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/getFavouriteItemList`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|--------|------------------|
| <code>userUID</code> | Long | Long user uid |
| <code>itemType</code> | String | String item type |

Return value

Type: `List<Favourite>`

List of Favourite object

Example

```
<getFavouriteItemList>
<userUID>45</userUID>
<itemType>stringValue</itemType>
</getFavouriteItemList>
```

41.2.9 favouriteService.getFavouriteStatusForItem

A Service method to get the favourite status for item.

Contains the business logic required to get favourite status.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/getFavouriteStatusForItem`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|--------|------------------|
| userUID | Long | Long user uid |
| itemUID | Long | Long item uid |
| itemType | String | String item type |

Return value

Type:

Boolean value true if item is a favourite

Example

```
<getFavouriteStatusForItem>
<userUID>93</userUID>
<itemUID>84</itemUID>
<itemType>stringValue</itemType>
</getFavouriteStatusForItem>
```

41.2.10 favouriteService.getList

A Service method to get the favourite details list for user uid.

Contains the business logic required to get favourite details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|------|---------------|
| userID | Long | Long user uid |

Return value

Type: List<Favourite>

List of Favourite object

Example

```
<getList>
  <userID>45</userID>
</getList>
```

41.2.11 favouriteService.update

A Service method to expose the updation of favourite into the system.

Calls the corresponding overloaded method with Favourite model object.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/favouriteService/update>

Parameters

| Name | Type | Description |
|---------------|---------|-------------------------|
| uid | Long | Long favourite uid |
| userID | Long | Long user uid |
| itemType | String | String item type |
| itemUID | Long | |
| prioritySeqNo | Integer | Integer priority number |

| Name | Type | Description |
|---------------------------|------|-------------|
| <code>modifiedDate</code> | Date | Date |

Return value

Type:

Integer number of records updated

41.2.12 favouriteService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

42 featureService

This is a primitive *Service* which provides the *Feature* Entity.

42.1 Feature

A *Feature* represents a specific capability of a device.

Figure 17 Feature and associated Entities.

Device Profiles are made up of *Features* of type **Device**. The same *Feature* may be included in any number of *Device Profiles*.

Asset Profiles and *Product Profiles* also contain *Features* of type **Device**. These *Features* are used when browsing the on-demand catalogue, to ensure that clients only display content which can be viewed on the device, and purchase offers which are targeted at the device.

42.1.1 Feature

| Name | Type | Description |
|--------------|---|---|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| enabled | Boolean | Is the Feature enabled. Default is true |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |
| name | <i>Search criterion. Quick Search criterion.</i> String | The name of this <i>Entity</i> . |
| originID | Integer | An identifier for the system which owns this <i>Entity</i> . |

| Name | Type | Description |
|--------------------------------|--------------------------|---|
| | | Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>policyGroupUID</code> | Long | Reference to a <i>PolicyGroup</i> . |
| <code>ratingID</code> | Long | Rating of this <i>Feature</i> . This is a Foreign Key reference to the <i>Rating</i> table |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>type</code> | String | <p>Feature type. One of <code>D</code> for a <i>Feature</i> which can be part of a <i>Device Profile</i>, or <code>S</code> for a <i>Feature</i> which can be part of a <i>Service Profile</i>.</p> <p><i>Profiles</i> of content and products (<i>Asset Profile</i>, <i>Product Profile</i>...) may be of either type.</p> <p>Note <i>Service Profiles</i> are not currently used in SDP.</p> |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |

42.2 tv.quative.service.feature.model.FeatureTypeEnum

| Name | Type | Description |
|--------|------|-----------------|
| DEVICE | | Device feature. |

| Name | Type | Description |
|----------------------|------|------------------|
| <code>SERVICE</code> | | Service feature. |

42.3 featureService

Provides all of the required functionality to Create, Retrieve, Update and Delete 'Feature' objects on the system.

Steve @author

42.3.1 featureService.create

Create (persist) a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

42.3.2 featureService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

42.3.3 featureService.getByName

Retrieve, from storage, a **Feature** identified by its 'name' attribute.

Parameters

| Name | Type | Description |
|-------------------|--------|--|
| <code>name</code> | String | The 'name' of the Feature to retrieve |

Return value

Type:

Feature If found, the matching object, null otherwise.

42.3.4 featureService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>5</originID>
</getByOrigin>
```

42.3.5 featureService.getByType

Retrieve, from storage, a list of **Feature** objects identified by type.

Parameters

| Name | Type | Description |
|------|--------|--|
| type | String | The type of the Feature objects to retrieve |

Return value

Type: **List<Feature>**

List If found, a list of the matching objects, null otherwise.

42.3.6 featureService.getByTypeAndNames

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---------------------------|-------------|
| nameList | List<String> | |
| type | String | |

Return value

Type: **List<Feature>**

42.3.7 featureService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|---|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>51</uid>
</getByUID>
```

42.3.8 featureService.getByUids

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------------|-------------|
| uidList | List<Long> | |

Return value

Type: List<Feature>

42.3.9 featureService.getFeaturesByProfUID

Returns a list of features belonging to the given profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------|
| profileUID | Long | Profile UID. |

Return value

Type: List<Feature>

List List of features belonging to the given profile.

Example

```
<getFeaturesByProfUID>
  <profileUID>62</profileUID>
</getFeaturesByProfUID>
```

42.3.10 featureService.update

Update, to storage, an existing **Feature** identified by its primary key (UID).

Parameters

| Name | Type | Description |
|-----------------------------|---------|--|
| <code>uid</code> | Long | The UID of the Feature to be updated |
| <code>name</code> | String | The name of the Feature |
| <code>description</code> | String | Textual description of the Feature |
| <code>ratingid</code> | Long | The Feature Rating FK reference ID |
| <code>enabled</code> | Boolean | Is the Feature enabled |
| <code>type</code> | String | The Type of the Feature [D]evice or [S]ervice |
| <code>policyGroupUID</code> | Long | Reference to a Policy Group. |
| <code>modifiedDate</code> | Date | The date of the modification |

Return value

Type:

Integer - The number of rows updated

42.3.11 featureService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------|
| <code>model</code> | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows updated.

43 geoBlockingService *New*

This service enables the operator to provision **region ID** and **IP address range** mappings to be used for Geo Blocking

The service provides the following APIs

- ▶ **createRegionIPMapping** - API to create mapping between region and IP range, regionId, cidr and type are mandatory
- ▶ **deleteRegionIPMappingByRegionId** - API to delete the mapping by region ID. All IP ranges for the specified region ID are deleted
- ▶ **deleteRegionIPMappingByRegionIdAndCidr** - API to delete specific mapping between region ID and IP range
- ▶ **getAllRegionIPMappings** - Retrieve a list of all region ID, IP range mappings
- ▶ **getIPMappingsByRegionId** - Retrieve a list of mapping of the specified region ID, IP range mappings
- ▶ **getAllWhitLists** - Retrieve list of all white listed IP addresses i.e., all IPv4 addresses with netmask 32 and all IPv6 addresses with netmask 128

43.1 geoBlockingService.createRegionIPMapping *New*

API to create mapping between region ID of the device (network field in database device table) and allowed IP address range (using CIDR format)

This API is available through Webservice at <http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/createRegionIPMapping>

| Name | Type | Description |
|----------|--------|---|
| regionId | string | ID of the region to which the device belongs to. Identified by "network" field in COR_DEVICE_STB table. Will be set during STB creation |
| cidr | string | Allowed IP address range in CIRD format (Example: 10.0.0.0/8 or 2001:db8:85a3:8d3:1319:8a2e:370:7348/128) |
| type | enum | Type of the IP address. Should be one of IPv4 or IPv6 |

Return Value

Integer - 1 when successfully inserted

Error - In case of validation failures

43.2 geoBlockingService.deleteRegionIPMappingByRegionId

New

API to delete the all the mappings between specified region ID of the device (network field in database device table) and allowed IP address range (using CIDR format)

This API is available through WebService at <http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/deleteRegionIPMappingByRegionId>

| Name | Type | Description |
|----------|--------|---|
| regionId | string | Region ID for which the allowed IP range mappings are to be deleted |

Return Value

Integer - Total number of entries deleted

43.3

geoBlockingService.deleteRegionIPMappingByRegionIdAndCidr

New

API to delete the specific mappings between region ID of the device (network field in database device table) and allowed IP address range (using CIDR format)

This API is available through WebService at <http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/deleteRegionIPMappingByRegionIdAndCidr>

| Name | Type | Description |
|----------|--------|---|
| regionId | string | Region ID for which the allowed IP range mapping is to be deleted |
| cidr | string | CIDR range to be deleted for the specified regionId |

Return Value

Integer - Total number of entries deleted

43.4 geoBlockingService.getAllRegionIPMappings **New**

API to retrieve all the mappings between region ID of the device (network field in database device table) and allowed IP address range (using CIDR format)

This API is available through WebService at `http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/getAllRegionIPMappings`

Return Value

List of all region ID and IP range mappings

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">767431530</rqid:requestID>
  </env:Header>
  <env:Body>
    <ns2:getAllRegionIPMappingsResponse xmlns:ns2="http://quative.tv/GeoblockingService
Namespace">
      <return type="IPv4" cidr="172.0.0.0/8" regionId="ABCD"/>
      <return type="IPv6" cidr="2001:db8:85a3:8d3:1319:8a2e:370:7348/120" region
Id="ABCD"/>
    </ns2:getAllRegionIPMappingsResponse>
  </env:Body>
</env:Envelope>
```

43.5 geoBlockingService.getAllWhitLists *New*

API to retrieve all whitelisted IPs for all the regions. Whitelisted IPs are specialised IPs that have only one IP address in the range (identified by netmask /32 or /128 for IPv4 and IPv6 addresses respectively)

This API is available through WebService at `http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/getAllWhitLists`

Return Value

List of all whitelisted region ID and IP range mappings

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">767431530</rqid:requestID>
  </env:Header>
  <env:Body>
    <ns2:getAllWhitListsResponse xmlns:ns2="http://quative.tv/GeoblockingService
Namespace">
      <return type="IPv4" cidr="172.16.25.34/32" regionId="ABCD"/>
      <return type="IPv6" cidr="2001:db8:85a3:8d3:1319:8a2e:370:7348/128" region
Id="ABCD"/>
    </ns2:getAllWhitListsResponse>
  </env:Body>
</env:Envelope>
```

43.6 geoBlockingService.getIPMappingsByRegionId New

API to retrieve the mappings between the specified region ID of the device (network field in database device table) and allowed IP address range (using CIDR format)

This API is available through Webservice at `http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/getIPMappingsByRegionId`

| Name | Type | Description |
|----------|--------|--|
| regionId | string | Region ID for which the allowed IP ranges is to be retrieved |

Return Value

Mappings between Region ID and allowed IP ranges

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">1609499476</rqid:request
ID>
  </env:Header>
  <env:Body>
    <ns2:getIPMappingsByRegionIdResponse xmlns:ns2="http://quative.tv/Geoblocking
ServiceNamespace">
      <return type="IPv4" cidr="172.0.0.0/8" regionId="ABCD"/>
      <return type="IPv6" cidr="2001:db8:85a3:8d3:1319:8a2e:370:7348/120" region
Id="ABCD"/>
      <return type="IPv6" cidr="2001:db8:85a3:8d3:1319:8a2e:370:7348/128" region
Id="ABCD"/>
    </ns2:getIPMappingsByRegionIdResponse>
  </env:Body>
</env:Envelope>
```

43.7 geoBlockingService.validateClientIpWithRegion New

API to retrieve the mappings between the specified region ID of the device (network field in database device table) and allowed IP address range (using CIDR format)

This API is available through Webservice at `http://<server>:<port>/ws-gateway/gateway/ws/geoblockingservice/validateClientIpWithRegion`

| Name | Type | Description |
|-------------|-------------|---|
| region | string | Region ID against which the provided IP address has to be validated |
| ipAddress | string | IP address of the client that needs to be validated against the specified region ID |

Return Value

TRUE : If the specified IP address falls under the range of allowed IP addresses for the region specified

FALSE : If the specified IP address does not falls under the range of allowed IP addresses for the region specified

44 initialisationService

This is an aggregate *Service* which retrieves data relevant to the currently signed-on subscriber account and the currently signed-on device.

The data can be retrieved either in one package with the `getInitDataBy<xxx>` methods, which return an `InitData` object, or as separate pieces using the remaining methods. The `InitData` object is not persisted to the SDP database.

44.1 tv.quative.service.favourite.model.ItemType

| Name | Type | Description |
|---------|------|----------------------------|
| C | | ItemType for 'C' Code. |
| CHANNEL | | Code for CHANNEL ItemType. |
| R | | ItemType for 'R' Code. |
| RATING | | Code for RATING Status. |

44.2 tv.quative.service.initialisation.model.InitData

| Name | Type | Description |
|--------------------|---------------------------|--|
| accountBlockings | List<Blocking> | List of <i>Blocking</i> objects associated to the <i>Account</i> , in deployments where these are used. |
| accountCatalogues | List<Catalogue> | List of <i>Catalogues</i> currently subscribed by the <i>Account</i> . |
| accountPackages | List<Package> | List of <i>Package</i> currently subscribed by the <i>Account</i> . |
| accountPreferences | List<Preference> | List of <i>Preference</i> objects associated to the <i>Account</i> . |
| accountUID | Long | The unique ID of the <i>Account</i> . |
| activePurchases | List<AcquiredContentList> | List of <i>Acquired Content List</i> entries representing the currently-active purchases of the <i>Account</i> . |

| Name | Type | Description |
|------------------------------------|----------------------------|--|
| <code>defaultLocale</code> | String | The default SDP locale for this deployment. Clients can use this to decide which language to use in the absence of any other information about the customer. |
| <code>deviceEnabled</code> | Boolean | Is the <i>Set Top Box</i> enabled? |
| <code>deviceFeatures</code> | List<Feature> | List of <i>Features</i> of the <i>Set Top Box</i> , in deployments where these are used. |
| <code>devicePhysicalAddress</code> | String | The <code>physicalAddress</code> of the <i>Set Top Box</i> . |
| <code>deviceSmartcardID</code> | String | The <i>Smartcard</i> ID of the <i>Set Top Box</i> . |
| <code>deviceUID</code> | Long | The UID of the <i>Set Top Box</i> . |
| <code>publicAddressAPS</code> | String | The <code>publicAddressAPS</code> of the <i>Account's Access Point</i> . Where it is set, this field stores the IP address of the local SDP application server. Clients should direct further requests to this address. |
| <code>publicAddressSTS</code> | String | The <code>publicAddressSTS</code> of the <i>Account's Access Point</i> . Where it is set, this field stores the IP address of the video server. Clients may use it when building RTSP requests for content in deployments not using session resource management. |
| <code>userFavourites</code> | Map<Long, List<Favourite>> | A map where the keys are the UIDs of <i>Users</i> associated with the <i>Account</i> , and the values are lists of <i>Favourite</i> Entities representing the content favoured by each <i>User</i> . |
| <code>users</code> | List<User> | List of <i>Users</i> associated to the <i>Account</i> . |

44.3 initialisationService

The interface for the Initialisation Aggregate Service. The Initialisation service uses various primitive services to retrieve data relevant to the currently signed on subscriber account and currently signed on device. The data is aggregated into a data model for use by the client application. Various data model attributes have separate methods to access the primitives. By exposing these methods we allow the client to get all data or refresh a subset of data by calling the specific method.

Steve @author

Naeem Bilal @author

44.3.1 initialisationService.getInitDataByCASN

Caution!

As installed, SDP requires the `casn` field to be unique across *Set Top Boxes*. However, some deployments require this constraint to be removed. Clients of an SDP which allows duplicate `casn` should avoid this API, because if a client uses it to get initial data, information will be supplied about one of the *Set Top Boxes* with the supplied `casn`, but which one is undefined.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/initialisationService/getInitDataByCASN`

Parameters

| Name | Type | Description |
|-------------------|--------|---|
| <code>casn</code> | String | The <code>casn</code> of the currently signed on device |

Return value

Type:

InitData object representing an aggregation of data specific the currently signed on device and subscriber account

44.3.2 initialisationService.getInitDataByMacAddress

Gets an *InitData* object containing information useful to a set-top box during initialization, identifying it by its `macAddress`.

Note

Ensure you use the correct capitalization for this API: `getInitDataByMacAddress` (compare `signonService.signonByMACAddress`)

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/initialisationService/getInitDataByMacAddress`

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>macAddress</code> | String | The <code>macAddress</code> of the currently signed on <i>Set Top Box</i> . |

Return value

Type:

44.3.3 initialisationService.getInitDataBySmartCardId

Gets an *InitData* object containing information useful to a set-top box during initialization, identifying it by its *Smart Card* ID.

Note

Ensure you use the correct capitalization for this API: `getInitDataBySmartCardId` (compare `signonService.signonBySmartcardID`)

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/initialisationService/getInitDataBySmartCardId`

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>smartCardId</code> | String | The <i>Smart Card</i> ID of the currently signed on device. |

Return value

Type:

45 intervalService

This is a primitive *Service* which provides the *Interval* Entity.

45.1 Interval

An *Interval* represents the idea of recurrence.

Figure 18 Interval and associated Entities.

Policies of type Base Price have an associated *Interval*, which is used to handle regular subscriptions: the *Interval* describes concepts such as every two weeks, once a month for twelve months, and so on.

The actual date on which the recurring event occurs depends on the start date of the *Policy*.

Most operators will use only a few *Intervals*, to represent particular types of subscription offer.

Caution!

In systems using a CMS, *Intervals* are created automatically by the ingest process. Pricing should be defined in the CMS.

The CMS ingest creates *Intervals* as required to represent the rental durations, subscription durations or billing time units required for the *Products* and *Prices* ingested.

45.1.1 Interval

| Name | Type | Description |
|--------------|---|--|
| changeLog | <i>Deprecated. Not visible to HUE. List<String></i> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| frequency | Integer | For an <i>Interval</i> with a Type of Recurring , the number of times the <i>Interval</i> should recur, or 0 to denote forever. For other <i>Intervals</i> , this field is ignored. |

| Name | Type | Description |
|--------------------------------|---|---|
| <code>locale</code> | <code>String</code> | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | <code>Date</code> | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> <code>Quick Search criterion.</code> <code>String</code> | The name of this <i>Entity</i> . |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>type</code> | <code>String</code> | The type of the <i>Interval</i> . This is an <code>IntervalType</code> value: one of <code>EXP</code> , <code>REC</code> , <code>MSP</code> , or <code>CNSMW</code> . |
| <code>uID</code> | <code>Long</code> | The SDP internal UID for this <i>Entity</i> . |
| <code>value</code> | <code>Long</code> | The value of the <i>Interval</i> , in the unit specified by the <code>valueType</code> . <i>For example:</i> To represent every two months, you would set <code>valueType</code> to <code>Month</code> and <code>value</code> to <code>2</code> . |
| <code>valueInSeconds</code> | <i>Read only.</i> <code>Long</code> | The length of the <i>Interval</i> , calculated in seconds from the <code>value</code> and <code>valueType</code> fields, where this can be reliably calculated (that is, <code>valueType</code> is a fixed time period like <code>MIN</code> as opposed to a variable one like <code>MONTH</code>) |

| Name | Type | Description |
|------------------------|--------|---|
| <code>valueType</code> | String | The unit the <i>Interval</i> is measured in. This is an Interval ValueType value: one of YEAR, MONTH, WEEK, DAY, HOUR, MIN, SEC, DOW, FDOM, LDOM, MOY or FIXED . |

45.2 intervalService.addMultiLanguage

A Service method to expose the addition of interval multi language data into the system.

Calls the corresponding overloaded method with a Interval model object.

Parameters

| Name | Type | Description |
|--------------------------|--------|----------------------|
| <code>intervalUID</code> | Long | Interval UID |
| <code>name</code> | String | Interval name |
| <code>description</code> | String | Interval description |
| <code>locale</code> | String | Locale code |

Return value

Type:

Integer - Number of records added

45.2.1 intervalService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|--|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

45.2.2 intervalService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | Interval ^{p.539} | |

Return value

Type:

45.2.3 intervalService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------------|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

45.2.4 intervalService.delete

A Service method to expose the deletion of interval from the system.

Calls the corresponding overloaded method with a Interval model object.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/intervalService/delete`

Parameters

| Name | Type | Description |
|--------------------------|------|-------------------|
| <code>intervalUID</code> | Long | Long interval UID |

Return value

Type:

Integer - Number of records deleted

45.2.5 intervalService.delete

Delete, from storage, a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------------|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

45.2.6 intervalService.getByName

Returns an interval with the given name..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---------------------------|
| <code>name</code> | String | The name of the interval. |
| <code>locale</code> | String | Locale code |

Return value

Type: `List<Interval>`

List - List of intervals with the given name.

Example

```
<getByName>
  <name>stringValue</name>
  <locale>en_us</locale>
</getByName>
```

45.2.7 intervalService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>35</originID>
```



```
<locale>en_us</locale>
</getByOrigin>
```

45.2.8 intervalService.getByType

Returns a list of policies with the given type..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/intervalService/getByType`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---------------------------|
| <code>type</code> | String | The type of the interval. |
| <code>locale</code> | String | Locale code |

Return value

Type: `List<Interval>`

List - Interval list returned by the query.

Example

```
<getByType>
<type>stringValue</type>
<locale>en_us</locale>
</getByType>
```

45.2.9 intervalService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/intervalService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| uid | Long | Integer unique id for the LocaleBaseModel object |
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>39</uid>
  <locale>en_us</locale>
</getByUID>
```

45.2.10 intervalService.getList

Returns a list of intervals..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|------------------------------------|
| locale | String | String representation of a Locale. |

Return value

Type: `List<Interval>`

List - List of intervals with the locale.

Example

```
<getList>
  <locale>en_us</locale>
</getList>
```

45.2.11 intervalService.getListByTypeValueFrequency

Gets a list of interval by type, value type, value and frequency..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/intervalService/getListByTypeValueFrequency`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---------------------|
| <code>type</code> | String | Interval type |
| <code>valueType</code> | String | Interval value type |
| <code>value</code> | Long | Value |
| <code>frequency</code> | Integer | Frequency |
| <code>locale</code> | String | Locale |

Return value

Type: `List<Interval>`

Interval list

Example

```
<getListByTypeValueFrequency>
  <type>stringValue</type>
  <valueType>stringValue</valueType>
  <value>70</value>
  <frequency>53</frequency>
  <locale>en_us</locale>
</getListByTypeValueFrequency>
```

45.2.12 intervalService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: `List<M>`

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>49</uid>
</getMultiLanguageList>
```

45.2.13 intervalService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

45.2.14 intervalService.removeMultiLanguage

A Service method to expose the removal of interval multi language data from the system.

Calls the corresponding overloaded method with a Interval model object.

Parameters

| Name | Type | Description |
|--------------------------|--------|--------------|
| <code>intervalUID</code> | Long | Interval UID |
| <code>locale</code> | String | Locale code |

Return value

Type:

Integer - Number of records removed

45.2.15 intervalService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

45.2.16 intervalService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------|
| <code>model</code> | Interval ^{p.539} | |

Return value

Type:

45.2.17 intervalService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

45.2.18 intervalService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

46 License Services

License *Services* are responsible for obtaining licenses for clients to view content from a license server.

Set-top boxes can obtain a license by calling `licenseService.getLicenseWithContext`. This *Service* uses SDP's configuration to request a license in the manner expected by the license server in use.

Different license servers handle licenses in different ways. The `licenseService` works out what type of license server is in use and passes the command on to the appropriate license server specific *Service*.

Tip

If something goes wrong, an error will be raised by the underlying license server specific *Service*, not by the `licenseService`, because different license servers have different error conditions.

SDP also provides the convenience API `bocPurchaseService.purchasePolicyAndGetLicense`, which allows a set-top box to purchase on-demand content and obtain a license in a single call. See the `bocPurchaseService` documentation for details of this API.

Note

The recommended API for this *Service* has changed in SDP 3.1 to allow more flexibility in the parameters. The previous APIs `licenseService.getLicense`, `licenseService.getLicenseWithAdditionalInfo`, `licenseService.getLicenseWithChallengeAuthentication`, `CODService.purchaseContentAndGetLicense` and `CODService.purchaseContentAndGetLicenseWithAdditionalInfo` all remain available for compatibility.

Challenge authentication

In the Nagra PRM system, the set-top box must decrypt a challenge to authenticate themselves before they can obtain a license. A challenge should be obtained in a separate call to the *Challenge* Service. The set-top box must solve this challenge within a time limit, and send the solution in the *Context* object supplied to `licenseService.getLicenseWithContext`.

46.1 licenseService

The `LicenseServiceImpl` is a class that exposed the operations available in `LicenseService` interface but it uses internally `LicenseServiceProxyacts` so that the user does not need to know to actual License Service Implementation. This class is implementing `LicenseService` to follow the standard of Generic `LicenseService`.

Miguel Gil @author

46.1.1 licenseService.getLicense

Obtains a licence for the signed-on device to access the specified Asset.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/licenseService/getLicense`

Parameters

| Name | Type | Description |
|-----------|--------|--|
| contentID | String | The unique identifier for the <i>Asset</i> as used by the license server. In Nagra PRM deployments, this is the <i>casId</i> of the <i>Asset</i> . |

Return value

Type:

The license returned by the license server, encoded as a character string.

46.1.2 licenseService.getLicenseWithAdditionalInfo

Retrieves a license for the currently signed-on *Set Top Box* to view the specified *Asset*, with optional challenge authentication, in a NAGRA Media ACCESS PRM environment which uses the Nagra UID (NUID) to identify devices.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/licenseService/getLicenseWithAdditionalInfo>

Parameters

| Name | Type | Description |
|-------------|--------|---|
| contentId | String | the unique identifier for the content (VOD etc) as used by the licensing authority |
| challengeId | String | the ID of the decrypted challenge to be verified by the SDP |
| nuid | String | In NAGRA Media ACCESS PRM environments where devices are identified by their NUID (Nagra Unique ID), the NUID identifying the device. Not used in other environments. |
| cts | String | The Chipset Type String (a human-transcribable encoding of a set of capability flags, known by the client.) In NAGRA Media ACCESS PRM environments, the Chipset Type String (a human-transcribable encoding of a set of capability flags, known by the client.) Not used in other environments. This field is not used. |

Return value

Type:

46.1.3 licenseService.getLicenseWithChallengeAuthentication

Retrieves a license for the currently signed-on *Set Top Box* to view the specified *Asset*, with challenge authentication.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/licenseService/getLicenseWithChallengeAuthentication`

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>contentID</code> | String | The unique identifier for the <i>Asset</i> as used by the license server. In Nagra PRM deployments, this is the <code>casID</code> of the <i>Asset</i> . |
| <code>challengeID</code> | String | The decrypted challenge. |

Return value

Type:

The license returned by the license server, encoded as a character string.

46.1.4 licenseService.getLicenseWithContext

Request a license from a configured license server for a previously-purchased content item.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/licenseService/getLicenseWithContext`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--|--|
| <code>context</code> | <code>tv.quative.service.context.api.Context</code> | <code>Context</code> object with information about the client requesting the license (including a decrypted <code>challengeId</code> if required.) |
| <code>licenseSpec</code> | <code>tv.quative.service.license.api.LicenseSpecification</code> | <code>LicenseSpecification</code> containing information which identifies the item to be licensed. |

Return value

Type: `LicenseResponse`

A license string suitable for passing to the client's cryptography subsystem.

46.1.5 licenseService.linkAuxiliarySTB

Retrieves an LCI to allow a STB to access the encrypted content of another STB..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/licenseService/linkAuxiliarySTB`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|---|
| <code>casn</code> | String | the CASN of the STB requesting the LCI. |
| <code>nuid</code> | String | the NUID of the STB requesting the LCI. |
| <code>lciOwningCasn</code> | String | the CASN of the STB that encoded the content. |
| <code>lciOwningNuid</code> | String | the NUID of the STB that encoded the content. |

Return value

Type:

String The LCI that will allow access to the encrypted content.

Example

```
<linkAuxiliarySTB>
  <casn>stringValue</casn>
  <nuid>stringValue</nuid>
  <lciOwningCasn>stringValue</lciOwningCasn>
  <lciOwningNuid>stringValue</lciOwningNuid>
</linkAuxiliarySTB>
```

47 linkedSTBService

This is an aggregate *Service* for use in deployments with Nagra PRM license servers which allows a device to request access to content encrypted by a different device.

47.1 tv.quative.service.linkedstb.api.PairNocsRequest

| Name | Type | Description |
|--------------------------------|---------------------|--|
| <code>addresseeCasn</code> | <code>String</code> | The casn of the device that wants to access the content. |
| <code>addresseeCsadList</code> | <code>String</code> | The casn of the device that wants to access the content. |
| <code>addresseeNuid</code> | <code>String</code> | The casn of the device that wants to access the content. |
| <code>operation</code> | <code>String</code> | Which LCI operation is to be performed (e.g. "add"). |
| <code>ownerCasn</code> | <code>String</code> | The CASN of the device that "owns" the content. |
| <code>ownerCsadList</code> | <code>String</code> | The casn of the device that "owns" the content. |
| <code>ownerNuid</code> | <code>String</code> | The NUID of the device that "owns" the content. |

47.2 linkedStbService

This class provides HUE and WS exposure for the STB linking operations. There are 2 types of link available:
Refurbish This is when storage media is removed from one box and reinstalled in another. It is necessary to allow the new box to access content encrypted by the old box.
Join Home Domain This is when there are several boxes in one account, it allows each box to access content encrypted and stored on the other boxes in the account.

mharvey @author

47.2.1 linkedStbService.joinHomeDomain

Allows a device to obtain the LCIs for other devices in the same home domain (those associated with the same *Account*)

The application should install all the LCIs returned by this API on the DVL to enable it to decrypt content stored on the other devices in the home domain. See reference [CAK-PRM-API-DVL](#) for details of this operation.

Note

If the new device is also going to share content, the existing devices in the home domain will also need to call this API to obtain the LCI for the new device.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/linkedStbService/joinHomeDomain`

Parameters

| Name | Type | Description |
|----------------------|--------|--|
| <code>newCasn</code> | String | The CASN of the requesting device. |
| <code>newNuid</code> | String | The PRM NUID of the requesting device. |

Return value

Type: `List<String>`

A list of LCIs, one for each device currently registered to the *Account*.

47.2.2 linkedStbService.pairNocsDevices

Allows the addressee device to access content encrypted by the owner device..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/linkedStbService/pairNocsDevices`

Parameters

| Name | Type | Description |
|-----------------------|--|---|
| <code>pRequest</code> | tv.quative.service.linkedstb.api:8080/requests | details of the pairing request ⁶⁵⁵ |

Return value

Type:

a LCI to control content access.

47.2.3 linkedStbService.refurbish

Allows a device to obtain the LCI which will allow it to decrypt content transferred from another device.

The application should install the LCI returned by this API on the DVL to enable it to decrypt the transferred content. See reference `CAK-PRM-API-DVL` for details of this operation.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/linkedStbService/refurbish`

Parameters

| Name | Type | Description |
|----------------------|--------|---|
| <code>newCasn</code> | String | The CASN of the requesting device. |
| <code>newNuid</code> | String | The PRM NUID of the requesting device. |
| <code>oldCasn</code> | String | The CASN of the device which encrypted the content. |
| <code>oldNuid</code> | String | The PRM NUID of the device which encrypted the content. |

Return value

Type:

An LCI for the device which encrypted the content.

48 mdrmService *New*

This Service allows the MDRM to interact with SDP.

48.1 mdrmService *New*

mdrmService implementation exposed methods for MDRM to call into SDP for various authorization calls.

48.1.1 mdrmService.isContentAuthorised *New*

HTTP URL

This API is called by sending a HTTP GET request to

`http://<server>:<port>/hue-gateway/gateway/http/js/mdrmService/isContentAuthorised`

The endpoint (server, port) is configurable in MDRM-Manager, see <https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pageId=27592037>

HTTP headers

| Name | Type | Description | Always Present |
|------------------|--------|---|----------------|
| x-correlation-id | string | A unique identifier to correlate logs corresponding to a request through the entire Multiscreen system. | Yes |
| Accept | string | Standard HTTP header. Will include <code>application/json</code> if supplied. | No |

Warning!

Http Basic Auth

This SDP service requires HTTP basic authentication.

URI Parameters

| Name | Type | Description | Always Present |
|-------|--------|---|----------------|
| arg0 | string | Content ID; The unique identifier assigned to the content by the CMS | Yes |
| token | string | A token acquired during client signon. The Portal decrypts the authToken and retrieve the account and/or device information required for license authorization. | Yes |

```
http://mySDPserver.com:80/hue-gateway/gateway/http/js/mdrmService/isContentAuthorized?token=zxcvb&arg0=12345678
```

Body

The body of this request is empty.

Response

HTTP 200 with JSON payload

| Name | Type | Description | Always Present |
|-------------------|---|---|--------------------------------------|
| isAuthorized | String, always "AUTHORIZED" or "NOT_AUTHORIZED" | the authorization status of the request. | Yes |
| accountNumber | string | unique identifier for the client account. | No (Yes for AUTHORIZED responses) |
| usageRules | UsageRulesType (see below) | The criteria for creating the license. Present only for Authorized responses. | No |
| keyDeliveryWindow | KeyDeliveryWindow (see below) | If 'Live PPV License' is enabled, when no live subscription exists for requested DRM ID, PPV event's timings will be included. For other scenarios, keyDeliveryWindow will be NULL. | yes |

| Name | Type | Description | Always Present |
|----------------|---------------------|---|----------------|
| additionalInfo | JSON map of strings | Optional parameter to convey additional information if required. The contents is a JSON map i.e key value pairs. SPD currently uses this field to present one parameter, "casProductId" | No |

UsageRulesType

| Name | Type | Description | Always Present |
|------------------------------------|-------------------------------------|--|---|
| ViewingNumber | integer | The viewing number | No |
| expiryDate | long | The time at which the license should expire. Milliseconds since Epoch | No |
| startDate | long | The start time of the license. Milliseconds since Epoch | No |
| consumption Window | string | The content consumption window. | No |
| LicenseExpiry Duration | integer | The expiry duration of the license. | No |
| isStorageAllowed | boolean | A flag stating whether persistence of the entitlement is allowed on the device. | No |
| fixedRuleset.digital OutputBitrate | float: a value in increments of 0.1 | The capped bitrate, when digital output is "Best EffortWithBitrateCapping" where the value is defined in Mbits/s in increments of 0.1. | Only when digital Output is "Best Effort With Bitrate Capping" Note: Currently this property is not part of |

| Name | Type | Description | Always Present |
|-----------------------------------|-------------------------------------|---|---|
| | | | SDP response |
| fixedRuleset.analog OutputBitrate | float: a value in increments of 0.1 | The capped bitrate, when analog output is "Best EffortWithBitrateCapping" where the value is defined in Mbits/s in increments of 0.1. | Only when analog Output is "Best Effort With Bitrate Capping" Note: Currently this property is not part of SDP response |
| isViewingWindow Floating; | boolean | Boolean value to enable/disable floating viewing window (usually possible when the licence is stored in the device) | no |

Key Delivery Window

| Name | Type | Description | Always Present |
|-----------|------|---|----------------|
| beginDate | long | PPV event's start time - guard padding time | Yes |
| endDate | long | PPV event's end time + guard padding time | Yes |

```
{
  "resultCode": "0",
  "result": {
    "isAuthorized" : "AUTHORIZED",
    "accountNumber" : "12345",
    "keydeliverywindow": {
      "beginDate" : "1234236612725",
      "endDate" : "1234556667775",
    }
  }
}
```

```

    },
    "usageRules": {
      "viewingNumber" : 12345,
      "expiryDate" : "1234556667775",
      "startDate" : "1234556667775",
      "consumptionWindow" : 1000,
      "licenseExpiryDuration" : 200,
      "isStorageAllowed" : true
    },
    "casProductId" : "IQ3-ASSET-VOD"
  }
}

```

```

{
  "resultCode": "0",
  "result": {
    "isAuthorized" : "NOT_AUTHORIZED"
  }
}

```

Error Response

HTTP error response with JSON payload

| Name | Type | Description | Always Present |
|---------------|--------|---|----------------|
| resultCode | string | SDP Error code. | Yes |
| result | string | A text message explaining the error | Yes |
| localeMessage | string | A localized text message explaining the error | No |

```

{
  "resultCode": "85102",
  "result": "Service method mdrmService.isDeviceAuthorised throw exception \"tv.quative.service.ServiceException: ErrorCode: [code=85102, severity=ERROR] Unable to decrypt the token, token is invalid\"",
  "localeMessage": "none"
}

```

48.1.2 mdrmService.isContentAuthorised *New*

HTTP URL

This API is called by sending a HTTP GET request to

`http://<server>:<port>/hue-gateway/gateway/http/js/mdrmService/isContentAuthorised`

The endpoint (server, port) is configurable in MDRM-Manager, see <https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pageId=27592037>

HTTP headers

| Name | Type | Description | Always Present |
|------------------|--------|---|----------------|
| x-correlation-id | string | A unique identifier to correlate logs corresponding to a request through the entire Multiscreen system. | Yes |
| Accept | string | Standard HTTP header. Will include <code>application/json</code> if supplied. | No |

Warning!

Http Basic Auth

This SDP service requires HTTP basic authentication.

URI Parameters

| Name | Type | Description | Always Present |
|-------|--------|---|----------------|
| arg0 | string | Content ID; The unique identifier assigned to the content by the CMS | Yes |
| arg1 | string | Private Data; Data passed from the client in the JSON format. This argument contains the user device zone locati and | No |
| token | string | A token acquired during client signon. The Portal decrypts the authToken and retrieve the account and/or device information required for license authorization. | Yes |

```
http://mySDPserver.com:80/hue-gateway/gateway/http/js/mdrmService/isContentAuthorized?
token=zxcvb&arg0=12345678&arg1={"locality": "XYZ", "isDownload": true}
```

Body

The body of this request is empty.

Response

HTTP 200 with JSON payload

| Name | Type | Description | Always Present |
|-------------------|---|---|--------------------------------------|
| isAuthorized | String, always "AUTHORIZED" or "NOT_AUTHORIZED" | the authorization status of the request. | Yes |
| accountNumber | string | unique identifier for the client account. | No (Yes for AUTHORIZED responses) |
| usageRules | UsageRulesType (see below) | The criteria for creating the license. Present only for Authorized responses. | No |
| additionalInfo | JSON map of strings | Optional parameter to convey additional information if required. The contents is a JSON map i.e key value pairs. SPD currently uses this field to present one parameter, "casProductId" | No |
| keyDeliveryWindow | KeyDeliveryWindow (see below) | If 'Live PPV License' is enabled, when no live subscription exists for requested DRM ID, PPV event's timings will be included. For other scenarios, keyDeliveryWindow will be NULL. | yes |

UsageRulesType

| Name | Type | Description | Always Present |
|---------------|---------|--------------------|----------------|
| ViewingNumber | integer | The viewing number | No |

| Name | Type | Description | Always Present |
|------------------------------------|-------------------------------------|--|--|
| expiryDate | long | The time at which the license should expire. Milliseconds since Epoch | No |
| startDate | long | The start time of the license. Milliseconds since Epoch | No |
| consumption Window | string | The content consumption window. | No |
| LicenseExpiry Duration | integer | The expiry duration of the license. | No |
| isStorageAllowed | boolean | A flag stating whether persistence of the entitlement is allowed on the device. | No |
| fixedRuleset.digital OutputBitrate | float: a value in increments of 0.1 | The capped bitrate, when digital output is "Best EffortWithBitrateCapping" where the value is defined in Mbits/s in increments of 0.1. | Only when digital Output is "Best Effort With Bitrate Capping" Note: Currently this property is not part of SDP response |
| fixedRuleset.analog OutputBitrate | float: a value in increments of 0.1 | The capped bitrate, when analog output is "Best EffortWithBitrateCapping" where the value is defined in Mbits/s in increments of 0.1. | Only when analog Output is "Best Effort With Bitrate Capping" Note: Currently |

| Name | Type | Description | Always Present |
|---------------------------|---------|---|---|
| | | | this property is not part of SDP response |
| isViewingWindow Floating; | boolean | Boolean value to enable/disable floating viewing window (usually possible when the licence is stored in the device) | no |

Key Delivery Window

| Name | Type | Description | Always Present |
|-----------|------|---|----------------|
| beginDate | long | PPV event's start time - guard padding time | Yes |
| endDate | long | PPV event's end time + guard padding time | Yes |

```
{
  "resultCode": "0",
  "result": {
    "isAuthorized" : "AUTHORIZED",
    "accountNumber" : "12345",
    "keydeliverywindow": {
      "beginDate" : "1234236612725",
      "endDate" : "1234556667775",
    },
    "usageRules": {
      "viewingNumber" : 12345,
      "expiryDate" : "1234556667775",
      "startDate" : "1234556667775",
      "consumptionWindow" : 1000,
      "licenseExpiryDuration" : 200,
    },
    "isStorageAllowed" : true
  },
  "casProductId" : "IQ3-ASSET-VOD"
}
```

```
{
  "resultCode": "0",
  "result": {
    "isAuthorized" : "NOT_AUTHORIZED"
  }
}
```

Error Response

HTTP error response with JSON payload

| Name | Type | Description | Always Present |
|---------------|--------|---|----------------|
| resultCode | string | SDP Error code. | Yes |
| result | string | A text message explaining the error | Yes |
| localeMessage | string | A localized text message explaining the error | No |

```
{
  "resultCode": "85102",
  "result": "Service method mdrmService.isDeviceAuthorised throw exception \"tv.quative.service.ServiceException: ErrorCode: [code=85102, severity=ERROR] Unable to decrypt the token, token is invalid\"",
  "localeMessage": "none"
}
```

48.1.3 mdrmService.isDeviceAuthorised *New*

HTTP URL

This API is called by sending a HTTP GET request to:

`http://<server>:port/hue-gateway/gateway/http/js/mdrmService/isDeviceAuthorised`

The endpoint (server, port) is configurable in MDRM-Manager, see <https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pageId=27592037>

HTTP headers

| Name | Type | Description | Always Present |
|------------------|--------|---|----------------|
| x-correlation-id | string | A unique identifier to correlate logs corresponding to a request through the entire Multiscreen system. | Yes |
| Accept | string | Standard HTTP header. Will include <code>application/json</code> if supplied. | No |

Warning!

Http Basic Auth

This SDP service requires HTTP basic authentication.

URI parameters

| Name | Type | Description | Always Present |
|-------|--------|--|----------------|
| token | string | SDP specific authentication token This value is taken from the <code>applicationData</code> input parameter of the inbound API | Yes |
| arg0 | string | MediaPlayerID, i.e Unique identifier of the device to be authorized. This value is taken from the <code>device.secretId</code> input parameter of the https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pagelD=27591604 inbound API | Yes |
| arg1 | string | Identifier for the platform (<code>"ios"</code> , <code>"Android"</code> ,...), if supplied by the client. This value is taken from the <code>device.platform</code> input parameter of the https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pagelD=27591604 inbound API If present, SDP uses this parameter to associate a "Device Classifier" to newly created devices, if one exists for the provided platform. | No |


```
http://mySDPserver.com:80/hue-gateway/gateway/http/js/mdrmService/isDeviceAuthorised?token=zxcvb&arg0=12345&arg1=iOS
```

Body

The body of this request is empty

Response

HTTP 200, with JSON payload.

| Name | Type | Description |
|---------------------|---|--|
| isDevice Authorized | String, always "AUTHORIZED" or "NOT_AUTHORIZED" | the authorization status of the request. |

```
{
  "resultCode": "0",
  "result": "AUTHORIZED"
}
```

```
{
  "resultCode": "0",
  "result": "NOT_AUTHORIZED"
}
```

Error Response

HTTP error response with JSON payload

| Name | Type | Description | Always Present |
|------------|--------|-------------------------------------|----------------|
| resultCode | string | SDP Error code. | Yes |
| result | string | A text message explaining the error | Yes |

| Name | Type | Description | Always Present |
|---------------|--------|---|----------------|
| localeMessage | string | A localized text message explaining the error | No |

```
{
  "resultCode": "85102",
  "result": "Service method mdrmService.isDeviceAuthorised throw exception \"tv.quative.service.ServiceException: ErrorCode: [code=85102, severity=ERROR] Unable to decrypt the token, token is invalid\"",
  "localeMessage": "none"
}
```

48.1.4 mdrmService.isDeviceAuthorised *New*

HTTP URL

This API is called by sending a HTTP GET request to:

`http://<server>:port/hue-gateway/gateway/http/js/mdrmService/isDeviceAuthorised`

The endpoint (server, port) is configurable in MDRM-Manager. This is available in the section **Configuring outbound API calls** in the document **MULTIDRM Installation and User Guide**

HTTP headers

| Name | Type | Description | Always Present |
|------------------|--------|---|----------------|
| x-correlation-id | string | A unique identifier to correlate logs corresponding to a request through the entire Multiscreen system. | Yes |
| Accept | string | Standard HTTP header. Will include <code>application/json</code> if supplied. | No |

Warning!

Http Basic Auth

This SDP service requires HTTP basic authentication.

URI parameters

| Name | Type | Description | Always Present |
|-------|--------|--|----------------|
| token | string | SDP specific authentication token This value is taken from the application Data input parameter of the /PrmCredentials Authorization/isDeviceAuthorized inbound API | Yes |
| arg0 | string | MediaPlayerID, i.e Unique identifier of the device to be authorized, or set top box CA SN. This value is taken from the device.secret Id input parameter of the /PrmCredentials Authorization/isDeviceAuthorized inbound API. If the platform argument has been provided as "NOCS" then this parameter is interpreted as the set top box CA SN. and this is used to identify the device. | Yes |
| arg1 | string | Identifier for the platform ("ios", "Android",...), if supplied by the client. This value is taken from the device.platform input parameter of the /PrmCredentials Authorization/isDeviceAuthorized inbound API If present, SDP uses this parameter to associate a "Device Classifier" to newly created devices, if one exists for the provided platform. | No |
| arg2 | string | the granularity of Device Information generated by a device/player {"device":{"screen":{"height":"1920","density":"480","width":"1080"},"hardware":{"model":"SM-G9006V","manufacturer":"samsung","type":"Handheld"},"OS":{"type":"android","version":"4.4.2"},"CPU":{"neon":"true","frequency":"2.4576","cores":"4"},"GPU":{"frequency":"0.45","cores":"32"},"secureplayer":{"DRMs":["PRM"],"codecs":["H264"],"streamings":["HLS"]}} If present, SDP uses this parameter to associate a "Device Classifier" to newly created devices | No |

```
http://mySDPserver.com:80/hue-gateway/gateway/http/js/mdrmService/isDeviceAuthorised?
token=zxcvbn&arg0=12345&arg1=iOS&arg2={"secureplayer":{"DRMs":["PRM"],"codecs":
["H264"],"streamings":["hls"]}}
```

Body

The body of this request is empty

Response

HTTP 200, with JSON payload.

| Name | Type | Description |
|---------------------|--|--|
| isDevice Authorised | Json Payload, with result as always "AUTHORIZED" or "NOT_AUTHORIZED" | the authorization status of the request. |

```
{
  "token":null,
  "resultCode":"0",
  "result":"AUTHORIZED",
  "requestId":1988700974
}
```

```
{
  "token":null,
  "resultCode":"0",
  "result":"NOT_AUTHORIZED",
  "requestId":1988700974
}
```

Error Response

HTTP error response with JSON payload

| Name | Type | Description | Always Present |
|------------|--------|-----------------|----------------|
| resultCode | string | SDP Error code. | Yes |

| Name | Type | Description | Always Present |
|---------------|--------|---|----------------|
| result | string | A text message explaining the error | Yes |
| localeMessage | string | A localized text message explaining the error | No |

```
{
  "resultCode": "85102",
  "result": "Service method mdrmService.isDeviceAuthorised throw exception \"tv.quative.service.ServiceException: ErrorCode: [code=85102, severity=ERROR] Unable to decrypt the token, token is invalid\"",
  "localeMessage": "none"
}
```

48.1.5 mdrmService.isHomeDomainAuthorised *New*

The Multi-DRM Manager uses this SDP API to check whether the specified devices are within the same Home Domain.

HTTP URL

This API is called by sending a HTTP GET request to:

`http://<server>:port/hue-gateway/gateway/http/js/mdrmService/isHomeDomainAuthorized`

The endpoint (server, port) is configurable in MDRM-Manager, see <https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pageId=27592037>

URI parameters

| Name | Type | Description | Always Present |
|-------|---|--|----------------|
| token | String | SDP specific authentication token (token must contain an account uid). | Yes |
| arg0 | URL encoded https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pageId=85892755 | Owner device ID* | Yes |

| Name | Type | Description | Always Present |
|------|---|------------------------|----------------|
| arg1 | URL encoded https://atlassian.hq.k.grp/confluence/pages/viewpage.action?pageId=85892755 | Destination device ID* | Yes |

* Only Device.secretId (Mpld), will be used for identifying the device in case of Open Devices.

Device JSON Object

The following JSON objects are shared between multiple MDRM outbound APIs:

Device

| Field name | Field type | Description | Multiplicity |
|-----------------------|------------|---|--------------|
| deviceUniqueId | String | <p>Unique identifier of the device to be authorized.</p> <p>This value is taken from the device.deviceUniqueId input parameter of the inbound API</p> <p>STBCASN for NOCS STB (allocated by the STB manufacturer) as a decimal form or hexadecimal form</p> <p>- Built by the PRM client upon platform information for open devices, such as MAC address or CPUID.</p> | 1..1 |
| secretId | String | <p>Identifier for the secrets assigned to a device, if supplied by the client.</p> <p>This value is taken from the device.secretId input parameter of the inbound API</p> <p>- NUID for NOCS STB as a decimal form or hexadecimal form</p> <p>- secretId for open devices. Several secret ids can be associated to the same device id</p> | 1..1 |
| platform | String | <p>Identifier for the platform ("NOCS", "ios", "Android",...), if supplied by the client.</p> <p>This value is taken from the device.platform input parameter of the inbound API</p> <p>- Platform name for open device</p> | 1..1 |

| Field name | Field type | Description | Multiplicity |
|-----------------------|--|--|--------------|
| | | - Hardcoded to "NOCS" for NOCS | |
| secretParams | String | CSAD List for NOCS as an hexadecimal pattern: "HHHH HHHH HHHH HHHH [CC]", checksum can be omitted. - Not relevant for other devices This field may be omitted only if the key package addressing is STBCASN | 0..1 |
| isPlatformCompromised | Boolean | true if the device or operating system is compromised (jailbroken or rooted), otherwise false. If not supplied, false must be assumed. This value is taken from the device.isPlatformCompromised input parameter of the inbound API | 0..1 |
| deviceTime | String: a date and time in ISO 8601 format ("YYYY-MM-DDThh:mm:ss.ffffffZ") | The current time recorded by the device. This value is taken from the device.deviceTime input parameter of the inbound API | 0..1 |

```
{
  "deviceUniqueId" : "100",
  "secretId" : "secret100",
  "platform" : "ios",
  "secretParams" : "0A01 0705 0A05 4CDE 66",
  "isPlatformCompromised" : false,
  "deviceTime" : "2013-10-11T13:00:00Z"
}
```

Example request URI

```
http://sdpserver.http://com:80/hue-gateway/gateway/http/js/mdrmService/
isHomeDomainAuthorized?token=zxcv&arg0=%7B%22deviceUniqueId%22%3A%2212345%22%2C
%22secretId%22%3A%2267890%22%2C%22platform%22%3A%22NOCS%22%7D&arg1=%7B
%22deviceUniqueId%22%3A%22abcdefg%22%2C%22secretId%22%3A%22123456%22%2C%22platform
%22%3A%22ios%22%7D
```

For clarity, the decoded form of the example URI:

```
http://mySDPServer.com:80/hue-gateway/gateway/http/js/mdrmService/isHomeDomain
Authorised?token=zxcv&arg0={"deviceUniqueId":"12345","secretId":"67890","platform":
"NOCS"}&arg1={"deviceUniqueId":"abcedfg","secretId":"123456","platform":"iOS"}
```

The body of this request is empty

Response

HTTP 200, with JSON payload.

| Name | Type | Description | Required |
|--------------|---|--|----------|
| isAuthorized | String, always "AUTHORIZED" or "NOT_AUTHORIZED" | The authorization status of the request. | Yes |

Example Authorized response

```
{
  "resultCode" : "0",
  "result" : "AUTHORIZED",
}
```

Example NOT Authorised

```
{
  "resultCode" : "0",
  "result" : "NOT_AUTHORIZED",
}
```

49 NmpExtendedService

This *Service* allows *Media Player* applications to initialize (personalize) themselves and also to obtain licenses from a NAGRA software PRM license server.

When SDP receives an initialization request through this *Service*, it contacts the PRM server to authenticate the application and obtain a unique media player ID for it. It then creates a *Media Player* Entity to represent the application, and returns details which enable it to sign on to SDP.

This *Service* can also be used to obtain licenses to watch content from the PRM server.

This *Service* should only be used by *Media Players*, not *Set Top Boxes*.

49.1 NmpExtendedService

amartens @author

49.1.1 NmpExtendedService.getEntitlements *Updated*

Allows building an NMP license by passing the asset (content) identifier.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/NmpExtendedService/getEntitlements>

Parameters

| Name | Type | Description |
|----------------------------|--------|-------------|
| <code>playerVersion</code> | String | |
| <code>playerType</code> | String | |
| <code>opaqueData</code> | String | |
| <code>contentId</code> | String | |

Return value

Type: `com.nagra.nmps.service.nmpextended.api.EntitlementsResponse<?>`

A request "result" object to be JSON serialized

Success Codes :

The NMPS success messages will be returned to Client in the response's status field.

| Success | Comment |
|----------------------------|----------------------------------|
| PLAYER_UPGRADE_REQUIRED | Critical update required for NMP |
| PLAYER_UPGRADE_RECOMMENDED | Recommended update for NMP |
| OK | NMP is upto date |

Error Codes :

If Upgrade Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for UGDM.

| Error | Comment |
|----------------------------|--|
| INVALID_RESPONSE_FROM_UGDM | Invalid Error response received from Upgrade Manager |
| INTERNAL_ERROR_FROM_UGDM | Internal error occurred in Upgrade Manager |
| UGDM_COMS | Problem occurred when communicating with Upgrade Manager |
| UGDM_ERROR | Unexpected error response received from Upgrade Manager |

If MDRM Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for MDRM.

| Error | Comment |
|------------------------------------|--|
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | SDP error when maximum MP device per account is reached. |
| MDRM_COMS | Problem occurred when communicating with MDRM Manager |
| MDRM_ERROR | Unexpected error response received from MDRM Manager |

If both MDRM and UGDM is ON, the precedence of Success and Error response from UGDM and MDRM is as follows :

Case 1 : If there is an Error from MDRM and Success from UGDM, then the response status will have MDRM_ERROR

Case 2 : If there is an Error from MDRM and Error from UGDM, then set response status with UGDM ERROR.

Case 3 : If Success from MDRM and Error from UGDM, then set response status with UGDM_ERROR.

Case 4 : If Success from MDRM and Success from UGDM, then set response status with UGDM Success.

The other error codes are as follows

| Error | Comment |
|------------------------------------|--|
| INTERNAL_ERROR | Error occured internally inside the server due to resouce unavailability or format mismatch or configuration issue |
| AUTHENTICATION_ERROR | Authentication issue, no token or data associated to token not found or expired |
| INACTIVE_ACCOUNT | The account associated to the request is not active. |
| INACTIVE_DEVICE | The NMP device that is linked to the given device id is not active. |
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | The maximum number of devices attached to the account has been reached. |
| NO_VALID_LICENSE | Unable to provide valid entitlements for the given asset due to expiry or non-subscription |
| UNDEFINED_CONTENT | The content/asset for which entitlements are requested does not exist in the system |
| UNDEFINED_GATEWAY | Target Gateway device not found |
| INACTIVE_GATEWAY | Target Gateway device is not enabled |
| INVALID_GATEWAY_ACCOUNT | Target Gateway and NMP devices not in the same home domain (account) |
| INVALID_GATEWAY_LICENSE | The LCM provided by the Gateway is invalid |
| POLICY_ENFORCEMENT_ERROR | One of the policies created is rejected due to the policy rule and an associated check failing. |
| DEVICE_LIMIT_REACHED | Max number of enabled devices reached. |

| Error | Comment |
|---------------------------------|--|
| DEVICE_CLASS_LIMIT_REACHED | Max number of enabled devices per device class reached |
| DEVICE_ACTIVATION_LIMIT_REACHED | Max number of device activations (enabling) within activation window |

49.1.2 NmpExtendedService.initializeDevice *Updated*

Allows device (re)personalization and initialization with a single entry point.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/NmpExtendedService/initializeDevice`

Parameters

| Name | Type | Description |
|---------------|--------|-------------|
| playerVersion | String | |
| playerType | String | |
| opaqueData | String | |

Return value

Type: `com.nagra.nmps.service.nmpextended.api.InitDeviceResponse`

A request "result" object to be JSON serialized

Success Code :

The NMPS success messages will be returned to Client in the response's status field.

| Success | Comment |
|----------------------------|----------------------------------|
| PLAYER_UPGRADE_REQUIRED | Critical update required for NMP |
| PLAYER_UPGRADE_RECOMMENDED | Recommended update for NMP |
| OK | NMP is upto date |

Error Code :

If Upgrade Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for UGDM.

| Error | Comment |
|------------------------------------|--|
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | SDP error when maximum MP device per account is reached. |
| INVALID_RESPONSE_FROM_UGDM | Invalid Error response received from Upgrade Manager |
| INTERNAL_ERROR_FROM_UGDM | Internal error occurred in Upgrade Manager |
| UGDM_COMS | Problem occurred when communicating with Upgrade Manager |
| UGDM_ERROR | Unexpected error response received from Upgrade Manager |

If MDRM Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for MDRM.

| Error | Comment |
|------------------------------------|--|
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | SDP error when maximum MP device per account is reached. |
| MDRM_COMS | Problem occurred when communicating with MDRM Manager |
| MDRM_ERROR | Unexpected error response received from MDRM Manager |

If both MDRM and UGDM is ON, the precedence of Success and Error response from UGDM and MDRM is as follows :

Case 1 : If there is an Error from MDRM and Success from UGDM, then the response status will have MDRM_ERROR

Case 2 : If there is an Error from MDRM and Error from UGDM, then set response status with UGDM ERROR.

Case 3 : If Success from MDRM and Error from UGDM, then set response status with UGDM_ERROR.

Case 4 : If Success from MDRM and Success from UGDM, then set response status with UGDM Success.

The other error codes are as follows

| Error | Comment |
|------------------------------------|--|
| INTERNAL_ERROR | Error occurred internally inside the server due to resource unavailability or format mismatch or configuration issue |
| AUTHENTICATION_ERROR | Authentication issue, no token or data associated to token not found or expired |
| INACTIVE_ACCOUNT | The account associated to the request is not active. |
| INACTIVE_DEVICE | The NMP device that is linked to the given device id is not active. |
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | The maximum number of devices attached to the account has been reached. |
| NO_VALID_LICENSE | Unable to provide valid entitlements for the given asset due to expiry or non-subscription |
| UNDEFINED_CONTENT | The content/asset for which entitlements are requested does not exist in the system |
| UNDEFINED_GATEWAY | Target Gateway device not found |
| INACTIVE_GATEWAY | Target Gateway device is not enabled |
| INVALID_GATEWAY_ACCOUNT | Target Gateway and NMP devices not in the same home domain (account) |
| INVALID_GATEWAY_LICENSE | The LCM provided by the Gateway is invalid |
| POLICY_ENFORCEMENT_ERROR | One of the policies created is rejected due to the policy rule and an associated check failing. |
| DEVICE_LIMIT_REACHED | Max number of enabled devices reached. |
| DEVICE_CLASS_LIMIT_REACHED | Max number of enabled devices per device class reached |

| Error | Comment |
|---------------------------------|--|
| DEVICE_ACTIVATION_LIMIT_REACHED | Max number of device activations (enabling) within activation window |

49.1.3 NmpExtendedService.initializeDevice *New*

Allows device (re)personalization and initialization with a single entry point.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/NmpExtendedService/initializeDevice`

Parameters

| Name | Type | Description |
|-------------------|--------|---|
| playerVersion | String | |
| playerType | String | |
| opaqueData | String | |
| deviceInformation | String | the granularity of Device Information generated by a device/player {"device":{"screen":{"height":"1920","density":"480","width":"1080"},"hardware":{"model":"SM-G9006V","manufacturer":"samsung","type":"Handheld"},"OS":{"type":"android","version":"4.4.2"},"CPU":{"neon":"true","frequency":"2.4576","cores":"4"},"GPU":{"frequency":"0.45","cores":"32"},"secureplayer":{"DRMs":["PRM"],"codecs":["H264"],"streamings":["HLS"]}} |

Return value

Type: `com.nagra.nmps.service.nmpextended.api.InitDeviceResponse`

A request "result" object to be JSON serialized

Success Code :

The NMPS success messages will be returned to Client in the response's status field.

| Success | Comment |
|----------------------------|----------------------------------|
| PLAYER_UPGRADE_REQUIRED | Critical update required for NMP |
| PLAYER_UPGRADE_RECOMMENDED | Recommended update for NMP |
| OK | NMP is upto date |

Error Code :

If Upgrade Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for UGDM.

| Error | Comment |
|----------------------------|--|
| INVALID_RESPONSE_FROM_UGDM | Invalid Error response received from Upgrade Manager |
| INTERNAL_ERROR_FROM_UGDM | Internal error occurred in Upgrade Manager |
| UGDM_COMS | Problem occurred when communicating with Upgrade Manager |
| UGDM_ERROR | Unexpected error response received from Upgrade Manager |

If MDRM Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for MDRM.

| Error | Comment |
|------------|---|
| MDRM_COMS | Problem occurred when communicating with MDRM Manager |
| MDRM_ERROR | Unexpected error response received from MDRM Manager |

If both MDRM and UGDM is ON, the precedence of Success and Error response from UGDM and MDRM is as follows :

Case 1 : If there is an Error from MDRM and Success from UGDM, then the response status will have MDRM_ERROR

Case 2 : If there is an Error from MDRM and Error from UGDM, then set response status with MDRM_ERROR.

Case 3 : If Success from MDRM and Error from UGDM, then set response status with UGDM_ERROR.

Case 4 : If Success from MDRM and Success from UGDM, then set response status with UGDM Success.

The other error codes are as follows

| Error | Comment |
|------------------------------------|--|
| INTERNAL_ERROR | Error occurred internally inside the server due to resource unavailability or format mismatch or configuration issue |
| AUTHENTICATION_ERROR | Authentication issue, no token or data associated to token not found or expired |
| INACTIVE_ACCOUNT | The account associated to the request is not active. |
| INACTIVE_DEVICE | The NMP device that is linked to the given device id is not active. |
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | The maximum number of devices attached to the account has been reached. |
| NO_VALID_LICENSE | Unable to provide valid entitlements for the given asset due to expiry or non-subscription |
| UNDEFINED_CONTENT | The content/asset for which entitlements are requested does not exist in the system |
| UNDEFINED_GATEWAY | Target Gateway device not found |
| INACTIVE_GATEWAY | Target Gateway device is not enabled |
| INVALID_GATEWAY_ACCOUNT | Target Gateway and NMP devices not in the same home domain (account) |
| INVALID_GATEWAY_LICENSE | The LCM provided by the Gateway is invalid |
| POLICY_ENFORCEMENT_ERROR | One of the policies created is rejected due to the policy rule and an associated check failing. |
| DEVICE_LIMIT_REACHED | Max number of enabled devices reached. |
| DEVICE_CLASS_LIMIT_REACHED | Max number of enabled devices per device class reached |

| Error | Comment |
|---------------------------------|--|
| DEVICE_ACTIVATION_LIMIT_REACHED | Max number of device activations (enabling) within activation window |

49.1.4 NmpExtendedService.transformEntitlements *Updated*

Allows building an NMP license by transforming an existing local content message (LCM).

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/NmpExtendedService/transformEntitlements`

Parameters

| Name | Type | Description |
|---------------|--------|-------------|
| playerVersion | String | |
| playerType | String | |
| opaqueData | String | |
| lcm | String | |

Return value

Type: `com.nagra.nmps.service.nmpextended.api.EntitlementsResponse<?>`

A request "result" object to be JSON serialized

Success Code :

The NMPS success messages will be returned to Client in the response's status field.

| Success | Comment |
|----------------------------|----------------------------------|
| PLAYER_UPGRADE_REQUIRED | Critical update required for NMP |
| PLAYER_UPGRADE_RECOMMENDED | Recommended update for NMP |
| OK | NMP is upto date |

Error Code :

If Upgrade Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for UGDM.

| Error | Comment |
|----------------------------|--|
| INVALID_RESPONSE_FROM_UGDM | Invalid Error response received from Upgrade Manager |
| INTERNAL_ERROR_FROM_UGDM | Internal error occurred in Upgrade Manager |
| UGDM_COMS | Problem occurred when communicating with Upgrade Manager |
| UGDM_ERROR | Unexpected error response received from Upgrade Manager |

If MDRM Manager is ON, the NMPS error will be returned to Client in the response's status field. The following NMPS errors for MDRM.

| Error | Comment |
|------------------------------------|--|
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | SDP error when maximum MP device per account is reached. |
| MDRM_COMS | Problem occurred when communicating with MDRM Manager |
| MDRM_ERROR | Unexpected error response received from MDRM Manager |

If both MDRM and UGDM is ON, the precedence of Success and Error response from UGDM and MDRM is as follows :

Case 1 : If there is an Error from MDRM and Success from UGDM, then the response status will have MDRM_ERROR

Case 2 : If there is an Error from MDRM and Error from UGDM, then set response status with UGDM ERROR.

Case 3 : If Success from MDRM and Error from UGDM, then set response status with UGDM_ERROR.

Case 4 : If Success from MDRM and Success from UGDM, then set response status with UGDM Success.

The other error codes are as follows

| Error | Comment |
|----------------|--|
| INTERNAL_ERROR | Error occurred internally inside the server due to resource unavailability or format mismatch or configuration issue |

| Error | Comment |
|------------------------------------|---|
| AUTHENTICATION_ERROR | Authentication issue, no token or data associated to token not found or expired |
| INACTIVE_ACCOUNT | The account associated to the request is not active. |
| INACTIVE_DEVICE | The NMP device that is linked to the given device id is not active. |
| MAXIMUM_DEVICE_PER_ACCOUNT_REACHED | The maximum number of devices attached to the account has been reached. |
| NO_VALID_LICENSE | Unable to provide valid entitlements for the given asset due to expiry or non-subscription |
| UNDEFINED_CONTENT | The content/asset for which entitlements are requested does not exist in the system |
| UNDEFINED_GATEWAY | Target Gateway device not found |
| INACTIVE_GATEWAY | Target Gateway device is not enabled |
| INVALID_GATEWAY_ACCOUNT | Target Gateway and NMP devices not in the same home domain (account) |
| INVALID_GATEWAY_LICENSE | The LCM provided by the Gateway is invalid |
| POLICY_ENFORCEMENT_ERROR | One of the policies created is rejected due to the policy rule and an associated check failing. |
| DEVICE_LIMIT_REACHED | Max number of enabled devices reached. |
| DEVICE_CLASS_LIMIT_REACHED | Max number of enabled devices per device class reached |
| DEVICE_ACTIVATION_LIMIT_REACHED | Max number of device activations (enabling) within activation window |

50 NmpOperatorService

This *Service* allows head-end systems to initialize (personalize) new *Media Player* applications and also to obtain licenses from a NAGRA software PRM license server.

It provides the same operations as the *NmpExtendedService*, but is used by Web Services clients.

50.1 NmpOperatorService

amartens @author

50.1.1 NmpOperatorService.getEntitlements

Get Entitlements: Allows building an NMP license by passing the asset (content) identifier..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|---|
| <code>playerVersion</code> | String | Player's version as sent by the NMPC |
| <code>playerType</code> | String | Player's operating system information (a.k.a. osType) |
| <code>opaqueData</code> | String | Protected data (carrying device attributes) |
| <code>contentId</code> | String | Asset identifier for which entitlements are required |
| <code>accountNumber</code> | String | User's account number |
| <code>deviceId</code> | String | NMP device identifier |

Return value

Type: `com.nagra.nmps.service.external.api.WSEntitlementResponse`

A request "result" object to be JSON serialized

Example

```
<getEntitlements>
  <playerVersion>stringValue</playerVersion>
  <playerType>stringValue</playerType>
  <opaqueData>stringValue</opaqueData>
  <contentId>stringValue</contentId>
  <accountNumber>ABC-54655</accountNumber>
```

```
<deviceId>stringValue</deviceId>
</getEntitlements>
```

50.1.2 NmpOperatorService.initializeDevice

Device Initialization: Allows device (re)personalization and initialization with a single entry point.. Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|---|
| playerVersion | String | Player's version as sent by the NMPC |
| playerType | String | Player's operating system information (a.k.a. osType) |
| opaqueData | String | Protected data (carrying device attributes) |
| accountNumber | String | User's account number |
| deviceId | String | NMP device identifier |

Return value

Type: `com.nagra.nmps.service.nmpextended.api.InitDeviceResponse`

A request "result" object to be JSON serialized

Example

```
<initializeDevice>
  <playerVersion>stringValue</playerVersion>
  <playerType>stringValue</playerType>
  <opaqueData>stringValue</opaqueData>
  <accountNumber>ABC-52177</accountNumber>
  <deviceId>stringValue</deviceId>
</initializeDevice>
```

50.1.3 NmpOperatorService.transformEntitlements

Transform Entitlements: Allows building an NMP license by transforming an existing local content message (LCM)..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|---|
| <code>playerVersion</code> | String | Player's version as sent by the NMPC |
| <code>playerType</code> | String | Player's operating system information (a.k.a. osType) |
| <code>opaqueData</code> | String | Protected data (carrying device attributes) |
| <code>lcm</code> | String | Base-64 encoded LCM, as recovered by the device |
| <code>accountNumber</code> | String | User's account number |
| <code>deviceId</code> | String | NMP device identifier |

Return value

Type: `com.nagra.nmps.service.external.api.WSEntitlementResponse`

A request "result" object to be JSON serialized

Example

```
<transformEntitlements>
  <playerVersion>stringValue</playerVersion>
  <playerType>stringValue</playerType>
  <opaqueData>stringValue</opaqueData>
  <lcm>stringValue</lcm>
  <accountNumber>ABC-37551</accountNumber>
  <deviceId>stringValue</deviceId>
</transformEntitlements>
```

51 nvodChannelService

This *Service* manages NVOD (Near Video On Demand) channels.

51.1 nvodChannelService

NvodChannel Service Implementation. Implementation which performs the compulsory methods defined in the Nvod ChannelService interface. NvodChannelService implemented object is used as WebServices to communicate with the Near Video On Demand (NVOD) system. The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true, instrument = false, roles = "<allowed-roles>") The methods used to create, delete or update items of the NVOD System, only could be accessible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Ashish Gupta @author

1.0 @version

51.1.1 nvodChannelService.addMultiLanguage

A Service method to expose the addition of nvodchannel multi language into the system.

Contains the business logic required for nvodchannel multi language addition.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---|
| c | Channel ^{p.133} | LocaleBaseModel model object of Channel |

Return value

Type:

Integer number of records added

51.1.2 nvodChannelService.create

A Service method to expose the creation of nvodchannel into the system.

Contains the business logic required for nvodchannel creation. The uid is generated internally via the DB sequence in insert procedure. If the locale is null then the default locale is considered by the insert procedure.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---|
| c | Channel ^{p.133} | LocaleBaseModel model object of Channel |

Return value

Type:

Long nvodchannel uid

51.1.3 nvodChannelService.delete

A Service method to expose the deletion of nvodchannel from the system.

Contains the business logic required for nvodchannel deletion. It will also delete all the records from multi language table.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---|
| c | Channel ^{p.133} | LocaleBaseModel model object of Channel |

Return value

Type:

Integer number of records deleted

51.1.4 nvodChannelService.getByOrigin

A Service method to get the nvod channel details by origin key.

Contains the business logic required to get nvod channel details.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/nvodChannelService/getByOrigin>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|--------------------------------|
| originKey | String | String nvod channel origin key |

| Name | Type | Description |
|-----------|---------|-------------------------------------|
| originUID | Integer | Integer nvod channel origin Integer |
| locale | String | String |

Return value

Type:

LocaleBaseModel object of Channel

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originUID>23</originUID>
  <locale>en_us</locale>
</getByOrigin>
```

51.1.5 nvodChannelService.getByOriginUID

A Service method to get the nvod channel details by origin uid.

Contains the business logic required to get nvod channel details.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|-------------|
| originUID | Integer | |
| locale | String | String |

Return value

Type: List<Channel>

List object of Channel

Example

```
<getByOriginUID>
  <originUID>46</originUID>
  <locale>en_us</locale>
</getByOriginUID>
```

51.1.6 nvodChannelService.getByUID

A Service method to get the nvod channel details by uid.

Contains the business logic required to get nvod channel details.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-----------------------|
| uid | Long | Long nvod channel uid |
| locale | String | String |

Return value

Type:

LocaleBaseModel object of Channel

Example

```
<getByUID>
  <uid>87</uid>
  <locale>en_us</locale>
</getByUID>
```

51.1.7 nvodChannelService.getList

A Service method to get the nvodchannel list for uid.

Contains the business logic required to get nvodchannel list.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-------------|
| locale | String | String |

Return value

Type: List<Channel>

List of Channel objects

Example

```
<getList>
  <locale>en_us</locale>
</getList>
```

51.1.8 nvodChannelService.getMultiLanguageList

A Service method to get the nvodchannel multi language details list for uid.
Contains the business logic required to get nvodchannel multi language details list.
Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | Long |

Return value

Type: List<Channel>

List of Channel objects

Example

```
<getMultiLanguageList>
  <uid>27</uid>
</getMultiLanguageList>
```

51.1.9 nvodChannelService.isNvodChannel

A Service method to know whether the channel is an nvod channel.
Contains the business logic required to know whether the channel is an nvod channel.
Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/nvodChannelService/isNvodChannel>
Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-----------------------|
| uid | Long | Long nvod channel uid |

Return value

Type:

Boolean true if channel is nvod channel

Example

```
<isNvodChannel>
  <uid>42</uid>
</isNvodChannel>
```

51.1.10 nvodChannelService.modifyMultiLanguage

A Service method to expose the modification of nvodchannel multi language into the system.

Contains the business logic required for nvodchannel multi language modification.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---|
| c | Channel ^{p.133} | LocaleBaseModel model object of Channel |

Return value

Type:

Integer number of records modified

51.1.11 nvodChannelService.removeMultiLanguage

A Service method to expose the removal of nvodchannel multi language from the system.

Contains the business logic required for nvodchannel multi language removal.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---|
| c | Channel ^{p.133} | LocaleBaseModel model object of Channel |

Return value

Type:

Integer number of records removed

51.1.12 nvodChannelService.update

A Service method to expose the updation of nvodchannel into the system.

Contains the business logic required for nvodchannel updation.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---|
| c | Channel ^{p.133} | LocaleBaseModel model object of Channel |

Return value

Type:

Integer number of records updated

52 nvodProgramService

This *Service* manages NVOD (Near Video On Demand) programmes.

52.1 nvodProgramService

NvodProgram Service Implementation. Implementation which performs the compulsory methods defined in the Nvod ProgramService interface. NvodProgramService implemented object is used as WebServices to communicate with the Near Video On Demand (NVOD) system The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true, instrument = false, roles = "<allowed-roles>") The methods used to create, delete or update items of the NVOD System, only could be accessible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Ashish Gupta @author

1.0 @version

52.1.1 nvodProgramService.addAltAudioPID

A Service method to expose the addition of nvod program alternate audio pid into the system.

Calls the corresponding overloaded method with ProgramAltAudioPID model object.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/nvodProgramService/addAltAudioPID>

Parameters

| Name | Type | Description |
|-----------|--------|-----------------------|
| uid | Long | Long nvod program uid |
| audioID | String | language id |
| language | String | Language |
| audioType | String | Audio type |

Return value

Type:

Integer number of records added

52.1.2 nvodProgramService.addAltAudioPID

A Service method to expose the addition of nvod program alternate audio pid into the system.
Contains the business logic required for nvod program alternate audio pid addition.
Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|---------------------------------|
| paap | tv.quative.service.program.model.ProgramAudioPid | ProgramAltAudioPID model object |

Return value

Type:

Integer number of records added

52.1.3 nvodProgramService.addMultiLanguage

A Service method to expose the addition of nvodprogram multi language into the system.
Calls the corresponding overloaded method with NvodProgram model object.

Parameters

| Name | Type | Description |
|-------------|--------|----------------------|
| uid | Long | Long nvodprogram uid |
| locale | String | String |
| name | String | String |
| description | String | String |
| synopsis | String | String |

Return value

Type:

Integer number of records added

52.1.4 nvodProgramService.addMultiLanguage

A Service method to expose the addition of nvodprogram multi language into the system.

Contains the business logic required for nvodprogram multi language addition.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|---|--|
| vp | VerboseProgram ^{p.734} | LocaleBaseModel model object of VerboseProgram |

Return value

Type:

Integer number of records added

52.1.5 nvodProgramService.create

A Service method to expose the creation of nvodprogram object into the system.

Calls the corresponding overloaded method with NvodProgram model object.

Parameters

| Name | Type | Description |
|-------------|---------|-----------------|
| locale | String | String |
| name | String | String |
| description | String | String |
| duration | Integer | Integer |
| ratingUID | Long | Long rating uid |
| recordable | Boolean | Boolean |
| timeshifted | Boolean | Boolean |
| audiomode | String | String |
| aspect | String | String |

| Name | Type | Description |
|-------------------------|---------|--|
| <code>definition</code> | String | String |
| <code>year</code> | String | String |
| <code>language</code> | String | String |
| <code>actor</code> | String | String |
| <code>director</code> | String | String |
| <code>producer</code> | String | String |
| <code>studio</code> | String | String |
| <code>promoImage</code> | String | String |
| <code>synopsis</code> | String | String |
| <code>originKey</code> | String | String origin key i.e. client identifier |
| <code>originUID</code> | Integer | Integer origin uid |

Return value

Type:

Long nvodprogram uid

52.1.6 nvodProgramService.create

A Service method to expose the creation of nvodprogram into the system.

Contains the business logic required for nvodprogram creation. The uid is generated internally via the DB sequence in insert procedure. If the locale is null then the default locale is considered by the insert procedure.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|---|--|
| vp | VerboseProgram ^{p.734} | LocaleBaseModel model object of VerboseProgram |

Return value

Type:

Long nvodprogram uid

52.1.7 nvodProgramService.delete

A Service method to expose the deletion of nvodprogram from the system.

Calls the corresponding overloaded method with NvodProgram model object.

Parameters

| Name | Type | Description |
|------|------|----------------------|
| uid | Long | Long nvodprogram uid |

Return value

Type:

Integer number of records deleted

52.1.8 nvodProgramService.delete

A Service method to expose the deletion of nvodprogram from the system.

Contains the business logic required for nvodprogram deletion. It will also delete all the records from multi language table.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|---|--|
| vp | VerboseProgram ^{p.734} | LocaleBaseModel model object of VerboseProgram |

Return value

Type:

Integer number of records deleted

52.1.9 nvodProgramService.getAltAudioPIDList

A Service method to get the nvod program alternate audio pid details list for nvod program uid.

Contains the business logic required to get nvod program alternate audio pid details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-----------------------|
| uid | Long | Long nvod program uid |

Return value

Type: List<ProgramAltAudioPID>

List of ProgramAltAudioPID object

Example

```
<getAltAudioPIDList>
  <uid>4</uid>
</getAltAudioPIDList>
```

52.1.10 nvodProgramService.getByOrigin

A Service method to get the nvod verbose program details by origin key.

Contains the business logic required to get nvod verbose program details.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|-------------------------------------|
| originKey | String | String nvod program origin key |
| originUID | Integer | Integer nvod program origin Integer |
| locale | String | String |

Return value

Type:

LocaleBaseModel object of VerboseProgram

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originUID>41</originUID>
  <locale>en_us</locale>
</getByOrigin>
```

52.1.11 nvodProgramService.getByOriginUID

A Service method to get the nvod verbose program details by origin uid.

Contains the business logic required to get nvod verbose program details.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---------------------------------|
| originUID | Integer | Integer nvod program origin uid |
| locale | String | String |

Return value

Type: List<VerboseProgram>

List object of VerboseProgram

Example

```
<getByOriginUID>
  <originUID>36</originUID>
  <locale>en_us</locale>
</getByOriginUID>
```

52.1.12 nvodProgramService.getByUID

A Service method to get the nvod verbose program details by uid.

Contains the business logic required to get nvod verbose program details.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-----------------------|
| uid | Long | Long nvod program uid |
| locale | String | String |

Return value

Type:

LocaleBaseModel object of VerboseProgram

Example

```
<getByUID>
  <uid>83</uid>
  <locale>en_us</locale>
</getByUID>
```

52.1.13 nvodProgramService.getMultiLanguageList

A Service method to get the nvodprogram multi language details list for uid.

Contains the business logic required to get nvodprogram multi language details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | Long |

Return value

Type: `List<VerboseProgram>`

List of VerboseProgram objects

Example

```
<getMultiLanguageList>
  <uid>11</uid>
</getMultiLanguageList>
```

52.1.14 nvodProgramService.getSimpleProgramByUID

A Service method to get the nvod simple program details by uid.

Contains the business logic required to get nvod simple program details.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/nvodProgramService/getSimpleProgramByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-----------------------|
| uid | Long | Long nvod program uid |
| locale | String | String |

Return value

Type:

SimpleProgram object

Example

```
<getSimpleProgramByUID>
  <uid>85</uid>
  <locale>en_us</locale>
</getSimpleProgramByUID>
```

52.1.15 nvodProgramService.modifyAltAudioPID

A Service method to expose the modification of nvod program alternate audio pid into the system.

Calls the corresponding overloaded method with ProgramAltAudioPID model object.

Parameters

| Name | Type | Description |
|----------|--------|-----------------------|
| uid | Long | Long nvod program uid |
| audioID | String | Integer language id |
| language | String | String |

| Name | Type | Description |
|--------------|--------|-------------|
| audioType | String | Audio type |
| modifiedDate | Date | Date |

Return value

Type:

Integer number of records modified

52.1.16 nvodProgramService.modifyAltAudioPID

A Service method to expose the modification of nvod program alternate audio pid into the system.

Contains the business logic required for nvod program alternate audio pid modification.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|---|---------------------------------|
| paap | tv.quative.service.program.model.ProgramAltAudioPID | ProgramAltAudioPID model object |

Return value

Type:

Integer number of records modified

52.1.17 nvodProgramService.modifyMultiLanguage

A Service method to expose the modification of nvodprogram multi language into the system.

Calls the corresponding overloaded method with NvodProgram model object.

Parameters

| Name | Type | Description |
|--------|--------|----------------------|
| uid | Long | Long nvodprogram uid |
| locale | String | String |

| Name | Type | Description |
|-----------------------------|--------|-------------|
| <code>name</code> | String | String |
| <code>description</code> | String | String |
| <code>synopsis</code> | String | String |
| <code>modifiedDateML</code> | Date | Date |

Return value

Type:

Integer number of records modified

52.1.18 `nvodProgramService.modifyMultiLanguage`

A Service method to expose the modification of nvodprogram multi language into the system.

Contains the business logic required for nvodprogram multi language modification.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|---|--|
| <code>vp</code> | VerboseProgram ^{p.734} | LocaleBaseModel model object of VerboseProgram |

Return value

Type:

Integer number of records modified

52.1.19 `nvodProgramService.removeAltAudioPID`

A Service method to expose the removal of nvodprogram alternate audio pid from the system.

Calls the corresponding overloaded method with ProgramAltAudioPID model object.

Parameters

| Name | Type | Description |
|-----------|--------|------------------------------|
| uid | Long | Long nvod program uid |
| language | String | description for the language |
| audioType | String | Audio type |

Return value

Type:

Integer number of records removed

52.1.20 nvodProgramService.removeAltAudioPID

A Service method to expose the removal of nvodprogram alternate audio pid from the system.

Contains the business logic required for nvodprogram alternate audio pid removal.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|---|---------------------------------|
| paap | tv.quative.service.program.model.program.AudioPid | ProgramAltAudioPID model object |

Return value

Type:

Integer number of records removed

52.1.21 nvodProgramService.removeMultiLanguage

A Service method to expose the removal of nvodprogram multi language from the system.

Calls the corresponding overloaded method with NvodProgram model object.

Parameters

| Name | Type | Description |
|------|------|----------------------|
| uid | Long | Long nvodprogram uid |

| Name | Type | Description |
|--------|--------|-------------|
| locale | String | String |

Return value

Type:

Integer number of records removed

52.1.22 nvodProgramService.removeMultiLanguage

A Service method to expose the removal of nvodprogram multi language from the system.

Contains the business logic required for nvodprogram multi language removal.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|---|--|
| vp | VerboseProgram ^{p.734} | LocaleBaseModel model object of VerboseProgram |

Return value

Type:

Integer number of records removed

52.1.23 nvodProgramService.update

A Service method to expose the updation of nvodprogram into the system.

Calls the corresponding overloaded method with NvodProgram model object.

Parameters

| Name | Type | Description |
|-----------|---------|----------------------|
| uid | Long | Long nvodprogram uid |
| duration | Integer | Integer |
| ratingUID | Long | Long rating uid |

| Name | Type | Description |
|---------------------------|---------|-------------|
| <code>recordable</code> | Boolean | Boolean |
| <code>timeshifted</code> | Boolean | Boolean |
| <code>audiomode</code> | String | String |
| <code>aspect</code> | String | String |
| <code>definition</code> | String | String |
| <code>year</code> | String | String |
| <code>language</code> | String | String |
| <code>actor</code> | String | String |
| <code>director</code> | String | String |
| <code>producer</code> | String | String |
| <code>studio</code> | String | String |
| <code>promoImage</code> | String | String |
| <code>modifiedDate</code> | Date | Date |

Return value

Type:

Integer number of records updated

52.1.24 `nvodProgramService.update`

A Service method to expose the updation of `nvodprogram` into the system.

Contains the business logic required for `nvodprogram` updation.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|---|--|
| <code>vp</code> | VerboseProgram ^{p.734} | LocaleBaseModel model object of VerboseProgram |

Return value

Type:

Integer number of records updated

53 ottSessionService

This is a primitive *Service* which provides the *OTTSession* Entity and allows clients to register and deregister over-the-top sessions.

53.1 OTT Session

An *OTTSession* represents an over-the-top session created by a client application.

When a client registers a session with SDP, it identifies

- ▶ the user's *Account*;
- ▶ the device (*Media Player* or *Set Top Box*) which is being used for the session; and
- ▶ the content (normally an *Asset*) which will be played out.

SDP can be configured to place a limit on the number of concurrent over-the-top sessions an *Account* is allowed to have. A conforming client will not attempt to create a new session if SDP tells it the limit has been reached.

SDP is not involved in the actual setup or teardown process of over-the-top sessions. It relies on the client to notify it before setup and after teardown.

OTTSession Entities are permanently deleted after the session ends. They are not used for reporting or auditing.

53.1.1 OTTSession

| Name | Type | Description |
|------------------------------|---|---|
| <code>accountId</code> | Long | id of the account thats associated with this session. |
| <code>allocatedDate</code> | Date | Date on which the session was created. |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| <code>contentId</code> | Long | The ID of the content being streamed by the sesion. |
| <code>contentType</code> | String | The type of content being streamed by the sesion. |
| <code>creationDate</code> | Date | The date created. |
| <code>deAllocatedDate</code> | Date | Date on which the session was deallocated. |
| <code>deviceId</code> | Long | The ID of the device using the session. |

| Name | Type | Description |
|--------------------------------|-------------------|---|
| <code>expiryDate</code> | Date | |
| <code>exportID</code> | Integer | The id to export object to third part systems. |
| <code>modifiedDate</code> | Date | The date last modified and persisted. |
| <code>originID</code> | Integer | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object derived by concatenating the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key of the model object in the origin system (3rd part system). |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>sessionType</code> | String | The type of session. |
| <code>status</code> | String | The status of the session. |
| <code>uID</code> | Long | The unique identifier. |

53.2 ottSessionService

Service implementation of OTTSessionService. OTTSessionService is an aggregate service with methods to setup, tear down and retrieve OTT sessions. SetUp and Tear down methods expect that is client is signed on and therefore the account id and device id are available in the request context.

RRajendram @author

53.2.1 ottSessionService.getMPDevicesWithOpenSessions

Retrieve a list of Media players with open OTT sessions for an account.

Accessible via HUE gateway only.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/ottSessionService/getMPDevicesWithOpenSessions>

Parameters

| Name | Type | Description |
|----------|------|---------------------------|
| startRow | Long | - Pagination - start row. |
| endRow | Long | - Pagination - end row. |

Return value

Type: `List<MediaPlayer>`

List - List of MediaPlayer objects.

53.2.2 ottSessionService.getSessionById

Retrieve a OTT session by session Id..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|------|--|
| ottSessionUid | Long | - Identifier of the session to be retrieved. |

Return value

Type:

OTTSession - The OTT session corresponding to the identifier passed in.

Example

```
<getSessionById>
  <ottSessionUid>54</ottSessionUid>
</getSessionById>
```

53.2.3 ottSessionService.getSessions

Retrieve a list of sessions for the given search criteria.

If the account number filter is specified, only open sessions belonging to the account are returned. If account number is not specified, returns list of all open OTT sessions. Accessible vis WS gateway only.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|--------|--------------------------------|
| accountNumber | String | - account number to filter by. |
| startRow | Long | - Pagination - start row. |
| endRow | Long | - Pagination - end row. |

Return value

Type: `List<OTTSession>`

List of open sessions.

Example

```
<getSessions>
  <accountNumber>ABC-95454</accountNumber>
  <startRow>52</startRow>
  <endRow>18</endRow>
</getSessions>
```

53.2.4 ottSessionService.setUp *Updated*

Given a content id and content type, sets up an OTT session.

The client must be signed on prior to invoking this method. The account and (or) device must be authorised to view the content, else a service exception is thrown. Only accessible via the HUE gateway.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/ottSessionService/setup`

Parameters

| Name | Type | Description |
|-------------|--------|---|
| contentId | String | - The id of the content that client wants to view. |
| contentType | String | - The type of content. Valid values are CHN and AST |

Return value

Type:

Session - the session that was set up.

53.2.5 ottSessionService.tearDown

Given a session Id, tears down the session.

Clients must be signed on prior to invoking this method and the request must be made from the same device that setup the session. Only accessible via the HUE gateway.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/ottSessionService/tearDown`

Parameters

| Name | Type | Description |
|------------------------|------|--|
| <code>sessionId</code> | Long | - the identifier of the session to be torn down. |

Return value

Type:

- the identifier of the session that was torn down.

53.2.6 ottSessionService.tearDownNoDeviceChecks

Tears down a session without validation if the tear down request originated from the device that setup the session..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|--|
| <code>sessionId</code> | Long | - The identifier of the session to be torn down. |

Return value

Type:

- the identifier of the session that was torn down.

Example

```
<tearDownNoDeviceChecks>
  <sessionId>61</sessionId>
</tearDownNoDeviceChecks>
```

54 ottUpgradeService *New*

This is a primitive **Service** which provides the **OTTUpgradeService** Entity and allows clients to upgrade.

54.1 ottUpgradeService *New*

Package Service Implementation. Object which implements the business logic of all the methods of the Integererface. The services are made visible to Hue or to WebServices by writing the annotations in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="packageService", hue = true, ws = true) For the Methods: @Qube Method(hue=false, ws = true) The methods used to create, delete or update items of the BTV System, only could be accesible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

jraya @author

54.1.1 ottUpgradeService.upgrade

Client Service for RolloutManager..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/ottUpgradeService/upgrade>

Parameters

| Name | Type | Description |
|----------|--------|--|
| deviceID | String | The identifier used by the Rollout Manager for the device, as stored in the <code>originKey</code> of the <i>Set Top Box</i> Entity, in the format <code><manufacturer-oui>,<model-id>,<hardware-revision-id>,<box-serial-number></code> |
| version | String | The current version of the firmware on the device. |

Return value

Type:

An object specifying the version and URL of the latest available firmware image.

55 packageService

This is a primitive *Service* which provides the BTV *Package* Entity.

It manages relationships between *Packages* and their constituent *Channels*.

55.1 packageService

Package Service Implementation. Object which implements the business logic of all the methods of the Integerinterface.

The services are made visible to Hue or to WebServices by writing the annotations in the Service Implementation class.

Ex: For the ServiceClass: @QubeService(name="packageService", hue = true, ws = true) For the Methods: @QubeMethod(hue=false, ws = true) The methods used to create, delete or update items of the BTV System, only could be accessible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

jraya @author

55.1.1 packageService.addChannel

Contains the business logic to add an existing Channel to the group of channels associated to the package.

This method creates the relationship between the channel and the package.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|---------|---|
| <code>pchanneluid</code> | Long | Integer channel uid with the information of the channel to add. |
| <code>ppkguid</code> | Long | Integer package uid with the information of the package. |
| <code>premium</code> | boolean | boolean that indicates it's a premium channel |

Return value

Type:

Long num rows added

55.1.2 packageService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

55.1.3 packageService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

55.1.4 packageService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--|-------------|
| <code>pkg</code> | Package ^{p.140} | |

Return value

Type:

55.1.5 packageService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows deleted.

55.1.6 packageService.getAllPackages

Gets all the packages for a specific language locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>locale</code> | String | String language which is running the application |

Return value

Type: `List<Package>`

List of `Package` objects with the information of the packages

Example

```
<getAllPackages>
  <locale>en_us</locale>
</getAllPackages>
```

55.1.7 packageService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>23</originID>
  <locale>en_us</locale>
</getByOrigin>
```

55.1.8 packageService.getByOriginUID

Gets the Package object information using the external system's id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|--|
| <code>originuid</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type: `List<Package>`

List of Package objects with the multilanguage information

Example

```
<getByOriginUID>
  <originuid>23</originuid>
  <locale>en_us</locale>
</getByOriginUID>
```

55.1.9 packageService.getByPackageOriginKeys

Gets the Package model objects for all given Origin Keys..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------------|-------------|
| originKeys | List<String> | |

Return value

Type: List<Package>

List of Package objects

55.1.10 packageService.getByPackageUIDs

Gets the Package model objects for all given Uids..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------------|-------------|
| packageUIDs | List<Long> | |

Return value

Type: List<Package>

List of Package objects

55.1.11 packageService.getByUID

Gets the *Package* with the specified *uid*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>uid</code> | Long | The <i>uid</i> of the <i>Package</i> . |
| <code>locale</code> | String | The locale to use for the localizable fields of the <i>Package</i> . |

Return value

Type:

The matching *Package*, or null if none is found.

Example

```
<getByUID>
  <uid>58</uid>
  <locale>en_us</locale>
</getByUID>
```

55.1.12 packageService.getChannels

Gets all the channels associated to the package.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>ppkgid</code> | Long | unique id of the <i>Package</i> |
| <code>locale</code> | String | object that contains the id of the current language. |

Return value

Type: `List<Channel>`

List of Channels objects with the information of the channels associated to the package.

Example

```
<getChannels>
  <ppkgid>93</ppkgid>
  <locale>en_us</locale>
</getChannels>
```

55.1.13 packageService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: List<M>

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>85</uid>
</getMultiLanguageList>
```

55.1.14 packageService.getPagedPackagesByChannelUID

getProductByChannel..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPagedPackagesByChannelUID>

Parameters

| Name | Type | Description |
|-------------|--------|----------------------------|
| channelUID | Long | unique channel identifier. |
| locale | String | Country code eg en_GB |
| startRowNum | Long | Long |

| Name | Type | Description |
|-----------|------|-------------|
| endRowNum | Long | Long |

Return value

Type: `List<Package>`

List of Packages.

55.1.15 packageService.getPagedPkgMapsByPkgID

Gets all the relationships between the channels and the packages..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPagedPkgMapsByPkgID`

Parameters

| Name | Type | Description |
|-------------|--------|--|
| pkguid | Long | Integer with the information of package UID |
| locale | String | object that contains the id of the current language. |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: `List<PackageMap>`

List of PackageMap objects with the information of the relationship between the package and the channels.

55.1.16 packageService.getPagedPkgMapsByPkgPatternAndCatUID

Gets all the relationships between the channels and the packages depending on a pattern and the category uid passed as a parameter.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pattern</code> | String | String with the information we want to retrieve. It should be like this: idpkg1 - idpkg2 - ... |
| <code>catuid</code> | Long | Integer category unique id. |
| <code>locale</code> | String | object that contains the id of the current language. |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<PackageMap>`

List of PackageMap objects with the information of the relationship between the package and the channels.

55.1.17 `packageService.getPagedPkgsByChanUID`

Retrieve a list of packages for a given channel..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPagedPkgsByChanUID`

Parameters

| Name | Type | Description |
|--------------------------|--------|--------------------------------|
| <code>chanuid</code> | Long | Integer - channel uid. |
| <code>locale</code> | String | String - locale of the channel |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<Package>`

List - a list containing the package information

55.1.18 packageService.getPagedPkgsByOriginUID

Gets the Package object information using the external system's id..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPagedPkgsByOriginUID>

Parameters

| Name | Type | Description |
|-------------|---------|--|
| originuid | Integer | Integer unique origin id for customer representation |
| locale | String | String reference for the language |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: List<Package>

List of Package objects with the multilanguage information

55.1.19 packageService.getPagedPkgsBySbscrProfUID

Returns a list of packages belonging to the given profile..

zlaposa @author

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPagedPkgsBySbscrProfUID>

Parameters

| Name | Type | Description |
|-------------|--------|--------------|
| profileUID | Long | Profile UID. |
| locale | String | String |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return valueType: `List<Package>`

List List of features belonging to the given profile.

55.1.20 `packageService.getPkgBySbscrProfUID`

Returns a list of packages belonging to the given profile..

zlaposa @author

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPkgBySbscrProfUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--------------|
| <code>profileUID</code> | Long | Profile UID. |
| <code>locale</code> | String | String |

Return valueType: `List<Package>`

List List of features belonging to the given profile.

Example

```
<getPkgBySbscrProfUID>  
<profileUID>17</profileUID>  
<locale>en_us</locale>  
</getPkgBySbscrProfUID>
```

55.1.21 `packageService.getPkgMapsByPkgID`

Gets all the relationships between the channels and the packages..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/packageService/getPkgMapsByPkgID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>pkguid</code> | Long | Integer with the information of package UID |
| <code>locale</code> | String | object that contains the id of the current language. * |

Return value

Type: `List<PackageMap>`

List of PackageMap objects with the information of the relationship between the package and the channels.

Example

```
<getPkgMapsByPkgID>
  <pkguid>41</pkguid>
  <locale>en_us</locale>
</getPkgMapsByPkgID>
```

55.1.22 packageService.getPkgMapsByPkgPattern

Gets all the relationships between the channels and the packages depending on a pattern passed as a parameter.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|--|
| <code>pattern</code> | String | String with the information we want to retrieve. It should be like this: idpkg1 - idpkg2 - ... |
| <code>locale</code> | String | object that contains the id of the current language. * |

Return value

Type: `List<PackageMap>`

List of PackageMap objects with the information of the relationship between the package and the channels.

Example

```
<getPkgMapsByPkgPattern>
  <pattern>stringValue</pattern>
  <locale>en_us</locale>
</getPkgMapsByPkgPattern>
```

55.1.23 packageService.getPkgMapsByPkgPatternAndCatUID

Gets all the relationships between the channels and the packages depending on a pattern and the category uid passed as a parameter..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|--|
| <code>pattern</code> | String | String with the information we want to retrieve. It should be like this: idpkg1 - idpkg2 - ... |
| <code>catuid</code> | Long | Integer category unique id. |
| <code>locale</code> | String | object that contains the id of the current language. |

Return value

Type: `List<PackageMap>`

List of PackageMap objects with the information of the relationship between the package and the channels.

Example

```
<getPkgMapsByPkgPatternAndCatUID>
<pattern>stringValue</pattern>
<catuid>71</catuid>
<locale>en_us</locale>
</getPkgMapsByPkgPatternAndCatUID>
```

55.1.24 packageService.getPkgsByChanUID

Retrieve a list of packages for a given channel.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|--------------------------------|
| <code>chanuid</code> | Long | Integer - channel uid. |
| <code>locale</code> | String | String - locale of the channel |

Return value

Type: `List<Package>`

List - a list containing the package information

Example

```
<getPkgsByChanUID>
  <chanuid>8</chanuid>
  <locale>en_us</locale>
</getPkgsByChanUID>
```

55.1.25 packageService.getQspDefaultPackage

Gets the default qsp package.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| locale | String | String language which is running the application |

Return value

Type:

Package object with the information of the package

Example

```
<getQspDefaultPackage>
  <locale>en_us</locale>
</getQspDefaultPackage>
```

55.1.26 packageService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

55.1.27 packageService.removeChannel

Contains the business logic to remove an existing relationship between the channel and the package.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|---|
| pchanneluid | Long | Integer channel uid with the information of the channel to add. |
| ppkguid | Long | Integer package uid with the information of the package. |

Return value

Type:

Integer num rows deleted

Example

```
<removeChannel>
  <pchanneluid>30</pchanneluid>
  <ppkguid>95</ppkguid>
</removeChannel>
```

55.1.28 packageService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

55.1.29 packageService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

55.1.30 packageService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--|-------------|
| <code>pkg</code> | Package ^{p.140} | |

Return value

Type:

55.1.31 packageService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

56 PAVService

This *Service* performs checks to see if a subscriber is permitted to access a specific content item.

56.1 tv.quative.service.boc.asset.model.BocAsset

| Name | Type | Description |
|--------------------------------|-------------------------------|---|
| <code>accessCriteriaHex</code> | <code>String</code> | Asset-specific access criteria, if any. Hex-string encoded octets. |
| <code>assetDefinition</code> | <code>String</code> | Definition (SD/HD/3D). |
| <code>assetFilename</code> | <code>String</code> | The asset filename. |
| <code>assetPlayType</code> | <code>BocAssetPlayType</code> | Asset playback type. |
| <code>assetUID</code> | <code>Long</code> | The underlying asset or event uid. |
| <code>assetUri</code> | <code>String</code> | Asset URI. |
| <code>bitRate</code> | <code>Long</code> | |
| <code>endPoint</code> | <code>Long</code> | |
| <code>event</code> | <code>boolean</code> | |
| <code>ownerId</code> | <code>String</code> | Asset or (BTV) channel owner ID. Ingested from the CMS. Required for NGOD R2. |
| <code>startPoint</code> | <code>Long</code> | |
| <code>titleAssetId</code> | <code>String</code> | |
| <code>titleProviderID</code> | <code>String</code> | |
| <code>trickPlay</code> | <code>TrickPlay</code> | The trick play status for the asset. |

56.2 tv.quative.service.boc.asset.model.BocAssetPlayType

| Name | Type | Description |
|---------------|------|--------------------------------|
| ADVERTISEMENT | | Asset play type Advertisement. |
| ENTERTAINMENT | | Asset play type Entertainment. |
| TRAILER | | Asset play type trailer. |

56.3 tv.quative.service.boc.asset.model.PlayList

| Name | Type | Description |
|-----------------|----------------|--|
| bocAssets | List<BocAsset> | The List of Assets. |
| firstByPlayType | BocAsset | |
| mainAssetIndex | Integer | The main Asset index of the list of BocAssets. |
| primaryAsset | BocAsset | The primary (entertainment or trailer) asset. |
| timePoint | Long | The time point of Event. |

56.4 PAVService

Playout Access Validity Service implementation. Implements the compulsory methods to deals with Playout Access Validity. The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="PAVService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The getters methods used to retrieve information using different data model could be accessible both the WebServices and the Hue.

Naeem Bilal @author

56.4.1 PAVService.getPlayList

This is method acts like facade of the advertisement sub system, and validate the SetTopBox, Account and purchased Asset before returning the PlayList..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--------------|
| <code>stb</code> | SetTopBox ^{p.416} | as SetTopBox |
| <code>asset</code> | Asset ^{p.156} | as Asset |

Return value

Type:

playList as PlayList

56.4.2 PAVService.getPlayListByMacAddress

This is method acts like facade of the advertisement sub system, and validate the SetTopBox, Account and purchased Asset before returning the PlayList.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|-------------|
| <code>macAddress</code> | String | as String |
| <code>assetUri</code> | String | as String |

Return value

Type:

playList as PlayList

Example

```
<getPlayListByMacAddress>
  <macAddress>stringValue</macAddress>
  <assetUri>stringValue</assetUri>
</getPlayListByMacAddress>
```

56.4.3 PAVService.getPlayListBySmartCard

This is method acts like facade of the advertisement sub system, and validate the SetTopBox, Account and purchased Asset before returning the PlayList.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|-------------|
| <code>smartcard</code> | String | as String |
| <code>assetUri</code> | String | as String |

Return value

Type:

playList as PlayList

Example

```
<getPlayListBySmartCard>  
  <smartcard>stringValue</smartcard>  
  <assetUri>stringValue</assetUri>  
</getPlayListBySmartCard>
```


57 PaymentOptionService

This is a primitive *Service* which provides the *Payment Method Provider* Entity.

This *Service* provides different methods for retrieving details of payment options. It cannot be used to create new payment options in SDP. To do this, you must inject them into the database directly. Scripts are available from Nagra to perform this task.

57.1 tv.quative.service.payment.model.PaymentProviderDetails

| Name | Type | Description |
|---|--------|--|
| <code>accountPaymentCredentialsUid</code> | Long | The <code>uid</code> of the <i>Account Payment Credentials</i> , if applicable, which will be used for payment with this option. |
| <code>accountPaymentOptionUid</code> | Long | The <code>uid</code> of the <i>Account Payment Credentials</i> , if applicable, which will be used for payment with this option. |
| <code>cardExpiry</code> | String | For card-based payment providers, the expiry date as printed on the card (09/12...) Used for display only, so that users can verify which card will be charged. |
| <code>cardPan</code> | String | For card-based payment providers, the card number, with the first 12 digits hidden. Used for display only, so that users can verify which card will be charged. <i>For example: **** * 3244</i> |
| <code>cardToken</code> | String | A token generated by the payment provider, which authorizes SDP to take payments from the card. |
| <code>cardType</code> | String | For card-based payment providers, the type of card. (Visa, Mastercard...) Used for display only, so that users can verify which card will be charged. |
| <code>customerRef</code> | String | The identifier used for the customer (that is, <i>Account holder</i>) in the payment provider's system. |
| <code>description</code> | String | A short description of this <i>Entity</i> . |
| <code>endpointUrl</code> | String | The URL that payment requests should be submitted to. |

| Name | Type | Description |
|---------------------------------|--------|---|
| <code>failureCallbackUrl</code> | String | The URL the client should be directed to if the payment fails, if applicable. |
| <code>gatewayBeanName</code> | String | The name of the <i>Service</i> that will process this payment option: either null, for the internal post-paid payment method, or one of <ul style="list-style-type: none"> ▶ <i>payWizardCardService</i> ▶ <i>payWizardEwalletService</i> |
| <code>merchantRef</code> | String | The username or reference representing the <i>Service Provider</i> in the payment provider's system. |
| <code>paymentType</code> | String | The type of payment: one of <ul style="list-style-type: none"> ▶ <code>Post Paid</code> ▶ <code>Card</code> ▶ <code>E-wallet</code> |
| <code>providerName</code> | String | The name of the payment provider (e.g. Pay Wizard) |
| <code>providerSpecific1</code> | String | A field holding implementation-specific data as required. <ul style="list-style-type: none"> ▶ For PayWizard Connect (PTE), this field holds the <i>System</i> identifier. Other payment types do not use this field. |
| <code>providerSpecific2</code> | String | A field holding implementation-specific data as required. <ul style="list-style-type: none"> ▶ For PayWizard Connect (PTE), this field holds the <i>MIDtef</i> identifier. Other payment types do not use this field. |
| <code>providerSpecific3</code> | String | A field holding implementation-specific data as required. <ul style="list-style-type: none"> ▶ For PayWizard Connect (PTE), this field holds the <i>Client Tef</i> identifier. Other payment types do not use this field. |
| <code>providerSpecific4</code> | String | A field holding implementation-specific data as required; not currently used by any supported implementations. |

| Name | Type | Description |
|--|---------|--|
| <code>providerSpecific5</code> | String | A field holding implementation-specific data as required; not currently used by any supported implementations. |
| <code>proxiedViaSdp</code> | Boolean | Does this payment option expect SDP to contact the payment provider to process transactions? For the PayWizard card and e-wallet payment methods, this will be <code>true</code> . For payment methods which require the client to contact the payment provider directly (for example, by redirection to secure screens provided by the payment provider), this will be <code>false</code> . In these cases SDP will expect either the client or the payment provider to notify it of a successful transaction before it will action the purchase. |
| <code>requiresAccountCredentials</code> | Boolean | Does this payment option require the user to have previously registered payment credentials with SDP? For the PayWizard Connect (PTE) card-based payment method, this will be <code>true</code> . For payment methods which allow the user to supply credentials at the time of purchase, or which do not require credentials at all (post paid, PayWizard E-wallet...), this will be <code>false</code> . |
| <code>serviceProviderID</code> | Long | The <i>Service Provider</i> that is offering the payment option. |
| <code>serviceProviderPaymentOptionUid</code> | Long | The <code>uid</code> of the <i>Service Provider Payment Option</i> which represents the payment option. |
| <code>successCallbackUrl</code> | String | The URL the client should be directed to if the payment succeeds, if applicable. |

57.2

tv.quative.service.payment.model.PaymentServiceProviderSession

| Name | Type | Description |
|-------------------------|------|--|
| <code>accountUid</code> | Long | The account that is involved in the session. |

| Name | Type | Description |
|---|--------|---|
| customerAccess Token | String | The token identifying the customer on the payment providers system. |
| merchantAccess Token | String | The token identifying the merchant on the payment providers system. |
| serviceProvider PaymentOptionUid | long | The pay option being used for the session. |
| sessionDate | Date | Date of session creation. |
| uid | Long | Unique DB ID. |

57.3 PaymentOptionService

Implementation of **PaymentOptionService** to provide data operations for payment entities that do not require their own CRUD service.

mharvey @author

57.3.1 PaymentOptionService.getPayMethodProvidersKeyData

Retrieve essential information about all *Payment Method Providers* in the deployment.

Available through Web Services.

Parameters

None.

Return value

Type: `List<PaymentMethodProviderKeyData>`

Details of each *Payment Method Provider*.

Example

```
<getPayMethodProvidersKeyData/>
```

57.3.2 PaymentOptionService.getPaymentOption

Fetch a specific payment option..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------------|---------|--|
| pAccountUID | Long | the account that is making the purchase. |
| pServiceProvider PaymentOptionUID | Long | the service provider payment option. |
| pEnabled | Boolean | |

Return value

Type:

PaymentProviderDetails aggregating all relevant data for the chosen payment option.

57.3.3

PaymentOptionService.getPaymentOptionByPaymentCredentialsUid

Fetch a specific payment option by id.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------------|---------|--|
| pPayment CredentialsUID | Long | the payment credentials to bill the purchase |
| pServiceProvider PaymentOptionUID | Long | the service provider payment option. |
| pEnabled | Boolean | |

Return value

Type:

PaymentProviderDetails aggregating all relevant data for the chosen payment option.

57.3.4 PaymentOptionService.getPaymentOptionsForAccount

Fetch the payment options that are available to the account.

The options returned are those supported by the accounts service provider.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|---------|---|
| pAccountUid | Long | The <code>uid</code> of the <i>Account</i> that is making the purchase. |
| pSpid | Long | The <code>uid</code> of the <i>Service Provider</i> to which the <i>Account</i> belongs. |
| pEnabled | Boolean | Should SDP return only the payment options which are currently enabled for the <i>Service Provider</i> ? (If you supplying <code>false</code> , SDP will return all configured options, not just the currently-enabled ones.) |

Return value

Type: `List<PaymentProviderDetails>`

Details of the payment options currently available to the *Account*.

57.3.5 PaymentOptionService.updatePaymentProviderSession

Update the session tokens..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--|--------------------------------------|
| pPayProvSession | tv.quative.service.payment.model.PaymentServiceProviderSession | the session tokens. ^{p.643} |

Return value

Type: `void`

58 PaymentTransactionService

This is a primitive *Service* which provides the *PaymentTransaction* Entity.

58.1 PaymentTransaction

A *PaymentTransaction* Entity represents an ongoing payment action.

It is created when SDP first receives a request to purchase content, to track the payment process. When the payment transaction completes (either successfully or unsuccessfully), this *Entity* is deleted.

Operator wishing to track payment actions should use the fields available in the *Billing* Entity for this purpose.

58.1.1 PaymentTransaction

| Name | Type | Description |
|---------------------------|---|---|
| accountUid | Long | The Account that is making the payment. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | |
| creationDate | Date | |
| exportID | Integer | |
| modifiedDate | Date | |
| originID | Integer | |
| originIdAndKey | <i>Read only.</i> String | |
| originKey | String | |
| payment CredentialsUid | Long | |
| policyGroupUid | Long | The content being purchased. |
| serviceProviderID | Long | |

| Name | Type | Description |
|--|------|--------------------------|
| <code>serviceProviderPayOptionUid</code> | Long | The payment method used. |
| <code>uID</code> | Long | |

58.2 PaymentTransactionService

Implementation of `PaymentTransactionService` to provide CRUD operations for `PaymentTransaction`.
mharvey @author

58.2.1 PaymentTransactionService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | |

Return value

Type:

58.2.2 PaymentTransactionService.createPaymentTransaction

Create a new `PaymentTransaction`..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------------|---|--|
| <code>pPayTransaction</code> | PaymentTransaction ^{p.647} | the <code>PaymentTransaction</code> details to create. |

Return value

Type:

the UID of the created PaymentTransaction - which should be used as the transaction ID in communication with a payment provider.

58.2.3 PaymentTransactionService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

58.2.4 PaymentTransactionService.deletePaymentTransaction

Deletes the PaymentTransaction identified by the UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|---|
| transactionUid | Long | the unique ID of the target PaymentTransaction. |

Return value

Type:

the number of rows deleted.

Example

```
<deletePaymentTransaction>
  <transactionUid>57</transactionUid>
</deletePaymentTransaction>
```

58.2.5 PaymentTransactionService.getByOrigin

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|-------------|
| originKey | String | |
| originID | Integer | |

Return value

Type:

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>23</originID>
</getByOrigin>
```

58.2.6 PaymentTransactionService.getByUID

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | |

Return value

Type:

Example

```
<getByUID>
  <uid>26</uid>
</getByUID>
```

58.2.7 PaymentTransactionService.getPaymentTransaction

Fetch the PaymentTransaction identified by the UID..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|------|---|
| pPaymentTransactionUid | Long | the UID of the PaymentTransaction to fetch. |

Return value

Type:

a PaymentTransaction with the requested unique ID.

Example

```
<getPaymentTransaction>
  <pPaymentTransactionUid>54</pPaymentTransactionUid>
</getPaymentTransaction>
```

58.2.8 PaymentTransactionService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

59 Ping Service

This *Service* is a simple primitive *Service* for demonstration and testing purposes, which returns the current time.

59.1 pingService

Service that returns a formatted string containing the current time. Used to check whether service tier is functioning correctly.

Dave Watson @author

59.1.1 pingService.ping

Returns a formatted string containing the current time..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/pingService/ping`

Available through Web Services.

Parameters

None.

Return value

Type:

a formatted string containing the current time.

Example

```
<ping/>
```

60 policyGroupService

This is a primitive *Service* which provides the *Policy Group* Entity.

It manages relationships between *Policy Groups* and their constituent *Devices*, and between *Policy Groups* and *Access Points*.

Relationships between *Policy Groups* and subscribable *Entities* are handled in the *Service* for the subscribable *Entity*.

60.1 policyGroupService

Policy Group Service Implementation. Implementation which implements the compulsory methods defined in the PolicyGroupService interface. The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The methods provide the functionality to create, retrieve, update, delete policy groups.

zlaposa @author

Naeem Bilal @author

Miguel Gil (NagraDVS) @author

60.1.1 policyGroupService.addAccessPoint

Adds an association of Access Point with Policy Group.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pAccessPointUid | Long | as accessPointUid |
| pPolicyGroup | Long | as policyGroup |

Return value

Type:

Integer as affected rows

Example

```
<addAccessPoint>
  <pAccessPointUid>82</pAccessPointUid>
  <pPolicyGroup>34</pPolicyGroup>
</addAccessPoint>
```

60.1.2 policyGroupService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

60.1.3 policyGroupService.addPolicy

Adds a policy to a policy group..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|------|---------------------------------|
| <code>policyGroupUID</code> | Long | UID of the parent policy group. |
| <code>policyUID</code> | Long | UID of the policy to add. |

Return value

Type:

Integer Number of inserted rows.

Example

```
<addPolicy>
  <policyGroupUID>36</policyGroupUID>
  <policyUID>43</policyUID>
</addPolicy>
```

60.1.4 policyGroupService.create

Create (persist) a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to persist. |

Return value

Type:

UID of the created object.

60.1.5 policyGroupService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows deleted.

60.1.6 policyGroupService.getByBasePriceUID

Returns policy groups with a given base price..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getByBasePriceUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|---------------------------------|
| <code>basePriceUID</code> | Long | UID of a BasePrice type policy. |
| <code>locale</code> | String | Locale |

Return value

Type: `List<PolicyGroup>`

List - List of policy groups with the given base price.

Example

```
<getByBasePriceUID>
  <basePriceUID>33</basePriceUID>
  <locale>en_us</locale>
</getByBasePriceUID>
```

60.1.7 policyGroupService.getByEvent

Gets the list of policy groups associated with the given event..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getByEvent`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--------|---------------|
| <code>eventId</code> | Long | Event UID |
| <code>locale</code> | String | locale string |

Return value

Type: `List<PolicyGroup>`

List of policy groups

Example

```
<getByEvent>
  <eventId>76</eventId>
  <locale>en_us</locale>
</getByEvent>
```


60.1.8 policyGroupService.getByName

Gets the *Policy Group* with the specified **name**.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getByName`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>name</code> | String | The name of the <i>Policy Group</i> . |
| <code>locale</code> | String | The locale to use when returning multi-language fields. |

Return value

Type: `List<PolicyGroup>`

A list of matching *Policy Groups* (may be empty.)

Example

```
<getByName>
  <name>stringValue</name>
  <locale>en_us</locale>
</getByName>
```

60.1.9 policyGroupService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getByOrigin`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |

| Name | Type | Description |
|-----------------------|---------|--|
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>16</originID>
  <locale>en_us</locale>
</getByOrigin>
```

60.1.10 policyGroupService.getByPolicyUID

Returns policy groups containing the given policy (not a base price policy)..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getByPolicyUID>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---|
| <code>policyUID</code> | Long | UID of a policy type other than base price. |
| <code>locale</code> | String | Locale |

Return value

Type: `List<PolicyGroup>`

List - List of policy groups with the given base price.

Example

```
<getByPolicyUID>
  <policyUID>54</policyUID>
  <locale>en_us</locale>
</getByPolicyUID>
```

60.1.11 policyGroupService.getByType

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|--|
| <code>basepricetype</code> | String | the type of the base price. The type of base price is determined by the frequency of the base price which is one of the following types 1.IMPULSE("IMP") 2.RECURRING("REC") 3.MULTIPLE("MUL") 4.SEASON_PASS("SEP") 5.ALL("ALL") - this is used in order to facilitate retrieval of all Frequencies |

Return value

Type: `List<PolicyGroup>`

List of Policy Group objects with each having the information of policy groups

Example

```
<getByType>
  <basepricetype>stringValue</basepricetype>
</getByType>
```

60.1.12 policyGroupService.getByTypeLocaleAndValidity *Updated*

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|--|
| <code>basepricetype</code> | String | the type of the base price. The type of base price is determined by the frequency of the base price which is one of the following types 1.IMPULSE("IMP") 2.RECURRING("REC") 3.MULTIPLE("MUL") 4.SEASON_PASS("SEP") 5.ALL("ALL") - this is used in order to facilitate retrieval of all Frequencies |
| <code>locale</code> | String | - locale of the base price |

| Name | Type | Description |
|-------------|---------|---|
| valid | Boolean | Determines if to return valid policies or both expired & valid policies. <ul style="list-style-type: none"> ▶ true returns only valid policies ▶ false returns all valid & expired policies |
| startRowNum | Long | - start row number |
| endRowNum | Long | - end row number |

Return value

Type: `List<PolicyGroup>`

List of Policy Group objects with each having the information of policy groups

60.1.13 policyGroupService.getByUID

Gets the *Policy Group* with the specified `uid`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|---|
| uid | Long | The <code>uid</code> of the <i>Policy Group</i> . |
| locale | String | The locale to use for the localizable fields of the <i>Policy Group</i> . |

Return value

Type:

The matching *Policy Group*, or null if none is found.

Example

```
<getByUID>
  <uid>75</uid>
  <locale>en_us</locale>
</getByUID>
```

60.1.14 policyGroupService.getDefaultQspZeroPrice

Gets the default qsp zero price policy group..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getDefaultQspZeroPrice`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| locale | String | String language which is running the application |

Return value

Type:

Policy Group object with the information of the policy Group.

Example

```
<getDefaultQspZeroPrice>
  <locale>en_us</locale>
</getDefaultQspZeroPrice>
```

60.1.15 policyGroupService.getList

Gets all defined **Policy Groups**.

Caution! In a production environment, this call will return a large amount of data. Use it sparingly.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|---|
| locale | String | The locale to use when returning multi-language fields. |

Return value

Type: `List<PolicyGroup>`

Example

```
<getList>
  <locale>en_us</locale>
</getList>
```

60.1.16 policyGroupService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: List<M>

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>81</uid>
</getMultiLanguageList>
```

60.1.17

policyGroupService.getPagedPolicyForCatAndAssetGroupAndAsset

Returns the list of polices depending upon many way the parameters passed.

This method throw exception on sending Catalogue Uid as null. If Catalogue Uid passed as not null and Asset Group and Asset passed as null then we return all the policy groups of non episodes assets filtered by policy group type. When Asset Group passed as not null and asset as null then policy groups return of all assets of that Asset Group within the Catalogue and filtered by the Policy Group Type. If Asset Group passed as null and Asset not null then policy groups of non episodes assets returns associated with the Catalogue filtered by the policy group type. Policy types can be passed as IMP, MUL, SEP or REC where combination of these can be passed as comma delimited values like "IMP, MUL, SEP". If Policy types passed as null then method returns all the policy groups of all policy types.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getPagedPolicyForCatAndAssetGroupAndAsset>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--------|--|
| catalogueUid | Long | as Long Catalogue Uid and should not passed as null. |
| assetGroupUid | Long | as Long Asset Group Uid and can be passed as null. |
| assetUid | Long | as Long Asset Uid and can be passed as null. |
| policyGroupType | String | as String and can be passed as null. |
| locale | String | as String and can be passed as null, default locale can be used if pass as null. |
| startRowNum | Long | as Long. |
| endRowNum | Long | as Long. |
| accessPointUID | Long | as Long. |

Return value

Type: `List<PolicyGroup>`

List of policy groups.

Example

```
<getPagedPolicyForCatAndAssetGroupAndAsset>
<catalogueUid>12</catalogueUid>
<assetGroupUid>4</assetGroupUid>
<assetUid>67</assetUid>
<policyGroupType>stringValue</policyGroupType>
<locale>en_us</locale>
<startRowNum>12</startRowNum>
<endRowNum>97</endRowNum>
<accessPointUID>11</accessPointUID>
</getPagedPolicyForCatAndAssetGroupAndAsset>
```

60.1.18 policyGroupService.getPolicyGroupsByAccept

Gets List of PolicyGroup associated to AccessPoint..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getPolicyGroupsByAccept`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pAccessPointUid | Long | as accessPointUid |

Return value

Type: List<PolicyGroup>

List of PolicyGroup

Example

```
<getPolicyGroupsByAccept>
  <pAccessPointUid>2</pAccessPointUid>
</getPolicyGroupsByAccept>
```

60.1.19 policyGroupService.getPolicyGroupsBySchedule

Gets the list of policy groups associated with the schedule..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getPolicyGroupsBySchedule>

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|--------------|
| scheduleUid | Long | Schedule UID |

Return value

Type: List<PolicyGroup>

List of policy groups

Example

```
<getPolicyGroupsBySchedule>
  <scheduleUid>27</scheduleUid>
</getPolicyGroupsBySchedule>
```


60.1.20

policyGroupService.getPolicyGroupsByScheduleAndAccessPoint

Return the list of policy groups associated with a given schedule and access point uid.

Note that this method uses the default locale.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getPolicyGroupsByScheduleAndAccessPoint`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|------|-----------------|
| <code>scheduleUid</code> | Long | Schedule UID |
| <code>accessPointUid</code> | Long | AccessPoint UID |

Return value

Type: `List<PolicyGroup>`

List of policy groups

Example

```
<getPolicyGroupsByScheduleAndAccessPoint>
<scheduleUid>43</scheduleUid>
<accessPointUid>8</accessPointUid>
</getPolicyGroupsByScheduleAndAccessPoint>
```

60.1.21

policyGroupService.getPolicyGroupsByScheduleAndAccessPointAndLocale

Return the list of policy groups associated with a given schedule, access point uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|------|-----------------|
| <code>scheduleUid</code> | Long | Schedule UID |
| <code>accessPointUid</code> | Long | AccessPoint UID |

| Name | Type | Description |
|--------|--------|---------------------------------|
| locale | String | - set to default locale if null |

Return value

Type: `List<PolicyGroup>`

List of policy groups

Example

```
<getPolicyGroupsByScheduleAndAccessPointAndLocale>
  <scheduleUid>83</scheduleUid>
  <accessPointUid>31</accessPointUid>
  <locale>en_us</locale>
</getPolicyGroupsByScheduleAndAccessPointAndLocale>
```

60.1.22 policyGroupService.getPolicyGroupsByScheduleAndLocale

Gets the list of policy groups associated with a specified schedule and locale..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getPolicyGroupsByScheduleAndLocale>

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---|
| scheduleUid | Long | Schedule UID |
| locale | String | - the locale string for returned policy groups. |

Return value

Type: `List<PolicyGroup>`

List of policy groups

Example

```
<getPolicyGroupsByScheduleAndLocale>
  <scheduleUid>30</scheduleUid>
  <locale>en_us</locale>
</getPolicyGroupsByScheduleAndLocale>
```

60.1.23 policyGroupService.getScheduledValidPolicyGroups

Gets the list of policy groups which are valid and are scheduled.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyGroupService/getScheduledValidPolicyGroups`

Available through Web Services.

Parameters

None.

Return value

Type: `List<PolicyGroup>`

List of policy groups

Example

```
<getScheduledValidPolicyGroups/>
```

60.1.24 policyGroupService.getTechsByBusKey

Gets a list of technical policy groups given a business key..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|-------------|
| <code>polgrpUID</code> | Long | |
| <code>accountUID</code> | Long | |
| <code>spid</code> | Long | |
| <code>casInstanceId</code> | String | |

Return value

Type: `List<PolicyGroup>`

Example

```
<getTechsByBusKey>
<polgrpUID>43</polgrpUID>
<accountUID>81</accountUID>
<spid>97</spid>
<casInstanceId>stringValue</casInstanceId>
```

</getTechsByBusKey>

60.1.25 policyGroupService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

60.1.26 policyGroupService.removeAccessPoint

Remove an association of Access Point with Policy Group.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|-------------------|
| pAccessPointUid | Long | as accessPointUid |
| pPolicyGroup | Long | as policyGroup |

Return value

Type:

Integer as affected rows

Example

```
<removeAccessPoint>
<pAccessPointUid>62</pAccessPointUid>
<pPolicyGroup>88</pPolicyGroup>
```

</removeAccessPoint>

60.1.27 policyGroupService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

60.1.28 policyGroupService.removePolicy

Removes a policy from a policy group..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|---------------------------------|
| policyGroupUID | Long | UID of the parent policy group. |
| policyUID | Long | UID of the policy to remove. |

Return value

Type:

Integer Number of deleted rows.

Example

```
<removePolicy>
<policyGroupUID>80</policyGroupUID>
<policyUID>43</policyUID>
```

</removePolicy>

60.1.29 policyGroupService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

60.1.30 policyGroupService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows updated.

61 policyGrpSubscriptionAggreService

61.1 policyGrpSubscriptionAggreService

Saci Santhana raj @author

61.1.1 policyGrpSubscriptionAggreService.subscribeToPolicyGroup

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|--------|-------------|
| accountUid | Long | |
| userId | Long | |
| policyGroupUid | Long | |
| itemLocale | String | |
| smartcardID | String | |

Return value

Type: List<String>

Example

```
<subscribeToPolicyGroup>
  <accountUid>91</accountUid>
  <userId>55</userId>
  <policyGroupUid>82</policyGroupUid>
  <itemLocale>en_us</itemLocale>
  <smartcardID>ABC-10137</smartcardID>
</subscribeToPolicyGroup>
```

61.1.2 policyGrpSubscriptionAggreService.unsubscribePolicyGroup

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--------|-------------|
| accountUID | Long | |
| policyGrpUid | Long | |
| smartcardID | String | |

Return value

Type:

Example

```
<unsubscribePolicyGroup>  
<accountUID>61</accountUID>  
<policyGrpUid>45</policyGrpUid>  
<smartcardID>ABC-31200</smartcardID>  
</unsubscribePolicyGroup>
```


62 policyService

This is a primitive *Service* which provides the *Policy* Entity.

62.1 Policy

A *Policy* represents a price, discount, or tariff which forms an element of a *Policy Group* offered to subscribers.

Figure 19 Policy and associated Entities.

There are three kinds of *Policy*:

- ▶ A **Flat Price** overrides the base price. In this case, the flat price value is used in place of the base price value.
- ▶ A **Percentage Discount** policy reduces the base price by a defined percentage.
- ▶ A **Flat Rate Discount** policy reduces the base price by a fixed amount.

Base Prices are related to *Policies*, but have additional fields.

Note | The CMS does not create *Policies* other than *Base Prices*.

62.1.1 Policy

| Name | Type | Description |
|--------------|---|--|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| locale | String | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |

| Name | Type | Description |
|--------------------------------|---|---|
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The name of this <i>Entity</i> . |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>policyGroupUid</code> | Long | The price, or percentage associated with this <i>Policy</i> . If the <i>Policy</i> has type Percentage Discount , this value is the percentage by which to reduce the price. <i>For example:</i> A value of 50 would reduce the price by half. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>type</code> | String | The type of the <i>Policy</i> . A <code>tv.quative.service.policy.model.PolicyType</code> value: one of FP , PD , or FRD . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>unit</code> | String | The unit type of the <i>Policy</i> . A <code>tv.quative.service.policy.model.UnitType</code> value: one of PRICE or PERC . |
| <code>validFrom</code> | Date | The date and time this <i>Policy</i> starts to be valid. |
| <code>validTo</code> | Date | The date and time this <i>Policy</i> stops being valid. |

| Name | Type | Description |
|-------|--------|---|
| value | Double | The value (price or percentage) of this <i>Policy</i> . |

62.2 Base Price

A *Base Price* defines the standard price for the item and the validity period for which the price applies. It is mandatory for any subscribable item.

It is a type of *Policy*, although SDP no longer uses *Policies* of other types.

Figure 20 Policy and associated Entities.

Caution!

In systems using a CMS, *Base Prices* are created automatically by the ingest process. Pricing should be defined in the CMS.

The CMS ingest creates separate *Base Prices* as necessary to represent the offers and promotions you have defined.

Base Prices can represent payment plans as well as single purchases. This is managed by the **Policy Type**.

| Value | Description |
|-------------|--|
| Impulse | <p>These <i>Base Prices</i> are charged once only, at the time of purchase. <i>Policy Groups</i> using this type of <i>Base Price</i> provide access to a single content object, which is normally viewed straight away.</p> <p>For VOD products, an Impulse <i>Base Price</i> corresponds to TVOD (transactional VOD.)</p> |
| Multiple | <p>These <i>Base Prices</i> are charged once only, at the time of purchase. <i>Policy Groups</i> using this type of <i>Base Price</i> provide access to multiple content objects during a defined time period. The selection of content does not change.</p> <p>For VOD products, a Multiple <i>Policy</i> corresponds to Bundled VOD.</p> |
| Season Pass | <p>These <i>Base Prices</i> are charged once only, at the time of purchase. <i>Policy Groups</i> using this type of <i>Base Price</i> provide access to multiple content objects during a defined time period. The selection of content may change over the lifetime of the subscription.</p> |
| Recurring | <p>These <i>Base Prices</i> are charged repeatedly, to a schedule. This is the classic subscription model. <i>Policy Groups</i> using this type of <i>Base Price</i> provide</p> |

| Value | Description |
|-------|--|
| | <p>access to multiple content objects, which may change over the lifetime of the subscription.</p> <p>For VOD products, a Recurring <i>Base Price</i> corresponds to SVOD (subscription VOD.)</p> <p>A <i>Policy Group</i> with a recurring <i>Base Price</i> may be unsubscribed. Some <i>Base Prices</i> define a minimum subscription interval: if this is the case, a request to unsubscribe will not be honoured until this interval has expired.</p> |

Base Prices and validity periods

The *Base Price* for any purchase has an **Expiry interval** which determines how long the viewer should have access to the content, after they have purchased it.

For impulse purchases, the **Expiry interval** is typically short, say 24 hours. For subscription purchases, the **Expiry interval** is typically much longer.

62.2.1 BasePrice

| Name | Type | Description |
|--------------------------------|---|--|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE. List<String></i> | A log of the changes to this <i>Entity</i> history. |
| <code>consumptionWindow</code> | Long | <p>The UID of the consumption <i>Interval</i> for the <i>Base Price</i>, if one is required.</p> <p>Consumption <i>Intervals</i> are used in deployments which use Nagra PRM to implement relative licensing. When a subscriber buys a <i>Policy Group</i> whose <i>Base Price</i> has a consumption <i>Interval</i>, their client is expected to obtain a license to view the content at some point between purchase and viewing. This license entitles the subscriber to begin viewing the content at any point within the consumption <i>Interval</i> of the license being issued.</p> <p>Once they have begun viewing, the subscriber has access to the content for the expiry duration of the license, as for absolute licensing. The consumption <i>Interval</i> is no longer used.</p> <p>When the consumption <i>Interval</i> has elapsed, the subscriber will no longer be able to access the content, even if they have not yet viewed it.</p> |

| Name | Type | Description |
|---------------------------|--------------------|--|
| | | If a <i>Base Price</i> does not have a consumption <i>Interval</i> , this restriction is not applied. |
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created. |
| <code>currency</code> | String | The ISO 4217 currency code of the currency in which the <i>Base Price</i> is denominated. <i>For example: USD = US dollar, EUR = euro</i> |
| <code>deleted</code> | Read only. boolean | Has this <i>Base Price</i> been marked for deletion? When the CMS tells SDP that a <i>Base Price</i> is no longer required, SDP does not remove it from the database, because clients may still need the information. For example, an application which lists a viewer's purchase history will need access to all the historical <i>Base Prices</i> . Tip If your applications will not need historical <i>Base Price</i> data, we recommend setting up a periodic database process to move <i>Base Prices</i> which have been marked as deleted into a separate archive table. This will improve performance. |
| <code>description</code> | String | A short description of this <i>Entity</i> . |
| <code>expiry</code> | Long | The <i>Interval</i> after which subscribers will no longer be able to access the content. When this <i>Interval</i> begins depends on the other fields of the <i>Policy</i> . <ul style="list-style-type: none"> ▶ If the <i>Policy</i> has no Consumption Interval, the Expiry Interval begins at the time of purchase. This is normally enforced on the server side (either by SDP, by checking the <i>Acquired Content List</i>, or by the CAS.) ▶ If the <i>Policy</i> has a Consumption Interval, the Expiry Interval begins at the time the subscriber first starts to view the content (known as "first visualization".) |

| Name | Type | Description |
|-----------------------------|---|---|
| | | <p>In deployments which use Nagra PRM to implement relative licensing, this is enforced by the client using information in the license.</p> <p>Note</p> <p>A <i>Policy</i> which is Recurring should not have an Expiry Interval.</p> <p>In deployments which use Nagra PRM to implement relative licensing, the expiry interval for subscription licenses is a global configuration option which should be managed in <code>qspconf</code>.</p> |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>frequency</code> | String | The frequency type of the <i>Base Price</i> . A <code>tv.quative.service.policy.model.FrequencyType</code> value: one of IMP , MUL , REC or SEP . |
| <code>locale</code> | String | When <i>Base Price</i> objects are returned, the locale used for localizable fields. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The name of this <i>Entity</i> . |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . The CMS ingest uses a fixed value of 1. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |

| Name | Type | Description |
|--------------------------------|--------|---|
| | | The CMS ingest sets this field to the ID of the <i>Product</i> . |
| <code>policyGroupUid</code> | Long | This field should not be used. |
| <code>recurrence</code> | Long | For a <i>Base Price</i> which has frequency type REC (recurring), the <i>Interval</i> after which the price should be reapplied. For other types of <i>Base Price</i> , this field is ignored. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>type</code> | String | This field is shared with <i>Policy</i> . For <i>Base Price</i> Entities, it is always BP . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>unit</code> | String | This field is shared with <i>Policy</i> . For <i>Base Price</i> Entities, it is always PRICE . |
| <code>validFrom</code> | Date | The date and time this <i>Policy</i> starts to be valid. The first <i>Billing Events</i> associated with this <i>Policy</i> will be generated at this time. The CMS ingest places the <i>Product</i> object's Start Purchase in this field. |
| <code>validTo</code> | Date | The date and time this <i>Policy</i> stops being valid. The CMS ingest places the <i>Product</i> object's End Purchase in this field. |
| <code>value</code> | Double | The value of the <i>Base Price</i> , in the specified currency. |

62.3 policyService.addMultiLanguage

A Service method to expose the addition of policy multi language data into the system.

Calls the corresponding overloaded method with a Policy/BasePrice model object.

Parameters

| Name | Type | Description |
|--------------------------|--------|--------------------|
| <code>policyUID</code> | Long | Policy UID |
| <code>name</code> | String | Policy name |
| <code>description</code> | String | Policy description |
| <code>locale</code> | String | Locale code |
| <code>type</code> | String | Policy type |

Return value

Type:

Integer - Number of records added

62.3.1 `policyService.addMultiLanguage`

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

62.3.2 `policyService.addMultiLanguage`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|---|-------------|
| policy | Policy ^{p.673} | |

Return value

Type:

62.3.3 policyService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|---------------------------------|
| model | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

62.3.4 policyService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|---|-------------|
| policy | Policy ^{p.673} | |

Return value

Type:

62.3.5 policyService.createBasePrice

A Service method to expose the creation of a base price into the system.

Calls the corresponding overloaded method with a BasePrice model object.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--|---|
| <code>basePrice</code> | BasePrice ^{p.676} | entity with all the data of the BasePrice |

Return value

Type:

Integer - Base price UID

62.3.6 `policyService.delete`

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------------|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

62.3.7 `policyService.delete`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|---|-------------|
| <code>policy</code> | Policy ^{p.673} | |

Return value

Type:

62.3.8 policyService.deleteBasePrice

A Service method to expose the deletion of base price from the system.

Calls the corresponding overloaded method with a BasePrice model object.

Parameters

| Name | Type | Description |
|--------------|------|-----------------------|
| basePriceUID | Long | Integer BasePrice UID |

Return value

Type:

Integer - Number of records deleted

62.3.9 policyService.getByname

Returns a policy with the given name.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|-------------------------|
| name | String | The name of the policy. |
| locale | String | Locale code |
| type | String | Policy type |

Return value

Type: List<Policy>

List - List of BasePolicy - The matching policy

Example

```
<getByName>
<name>stringValue</name>
<locale>en_us</locale>
```

```
<type>stringValue</type>
</getByName>
```

62.3.10 policyService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
<originKey>stringValue</originKey>
<originID>60</originID>
<locale>en_us</locale>
</getByOrigin>
```

62.3.11 policyService.getByPolicyGroupUID

Returns a list of policies belonging to the given policy group.

In case of a base price type the returned list could contain only one BasePrice policy model.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--------|------------------------------------|
| <code>policyGroupUID</code> | Long | Parent group of the policy. |
| <code>type</code> | String | Policy type |
| <code>locale</code> | String | String representation of a Locale. |

Return value

Type: `List<Policy>`

Policy - Policies belonging to the given policy group.

Example

```
<getByPolicyGroupUID>
  <policyGroupUID>5</policyGroupUID>
  <type>stringValue</type>
  <locale>en_us</locale>
</getByPolicyGroupUID>
```

62.3.12 policyService.getByType

Returns a list of policies with the given type..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyService/getByType`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|-------------------------|
| <code>type</code> | String | The type of the policy. |
| <code>locale</code> | String | Locale code |

Return value

Type: `List<Policy>`

List - Policy list returned by the query.

Example

```
<getByType>
```

```
<type>stringValue</type>
<locale>en_us</locale>
</getByType>
```

62.3.13 policyService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| uid | Long | Integer unique id for the LocaleBaseModel object |
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>60</uid>
  <locale>en_us</locale>
</getByUID>
```

62.3.14 policyService.getByUIDAndType

Get the specified *Policy* or *Base Price*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/policyService/getByUIDAndType`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|---|
| policyUID | Long | The uid of the <i>Policy</i> or <i>Base Price</i> . |

| Name | Type | Description |
|---------------------|--------|---|
| <code>locale</code> | String | The locale to use when representing localizable fields of the <i>Policy</i> or <i>Base Price</i> . |
| <code>type</code> | String | BP for <i>Base Prices</i> . Otherwise, the type of the <i>Policy</i> . A <code>tv.quative.service.policy.model.PolicyType</code> value: one of BP , FP , PD , or FRD . |

Return value

Type:

Or *Base Price*.

Example

```
<getByUIDAndType>
  <policyUID>87</policyUID>
  <locale>en_us</locale>
  <type>stringValue</type>
</getByUIDAndType>
```

62.3.15 policyService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|--------------------------------|
| <code>uid</code> | Long | Integer unique id of the Model |

Return value

Type: `List<M>`

List `LocaleBaseModel` with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>87</uid>
</getMultiLanguageList>
```

62.3.16 policyService.getMultiLanguageListByType

Returns the multilingual data of a policy.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|---------------------------------|
| policyUID | Long | UID of the policy to look for. |
| type | String | Type of the policy to look for. |

Return value

Type: List<Policy>

List - List of policies with the given UID.

Example

```
<getMultiLanguageListByType>
  <policyUID>71</policyUID>
  <type>stringValue</type>
</getMultiLanguageListByType>
```

62.3.17 policyService.isPolicyApplied

Checks if the policy is applied in a policy group and the policy group is applied in an ACL.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-----------------------------|
| policyUID | Long | UID of the policy to check. |
| type | String | Policy type. |

Return value

Type:

Boolean - True if the policy is applied.

Example


```
<isPolicyApplied>
  <policyUID>60</policyUID>
  <type>stringValue</type>
</isPolicyApplied>
```

62.3.18 policyService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

62.3.19 policyService.modifyMultiLanguage

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|---|-------------|
| policy | Policy ^{p.673} | |

Return value

Type:

62.3.20 policyService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

62.3.21 `policyService.removeMultiLanguage`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|---|-------------|
| <code>policy</code> | Policy ^{p.673} | |

Return value

Type:

62.3.22 `policyService.update`

Update, to storage, an existing `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

62.3.23 policyService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|---|-------------|
| policy | Policy ^{p.673} | |

Return value

Type:

62.3.24 policyService.updateBasePrice

A Service method to expose the update of a base price into the system.

Calls the corresponding overloaded method with a base price model object.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--|---|
| pbasePrice | BasePrice ^{p.676} | entity with all the data of the BasePrice |

Return value

Type:

Integer - Base price UID

62.3.25 policyService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

62.3.26 `policyService.updateModelAndML`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|---|-------------|
| <code>policy</code> | Policy ^{p.673} | |

Return value

Type:

63 PPVService

This *Service* allows clients to view and purchase pay-per-view *Events*.

63.1 PPVService

Service implementation for Pay per View. The PPV Service is an aggregate service with methods to retrieve PPV events based on account id and to subscribe to these events. The service exposes various methods to retrieve all ppv events, subscribed ppv events, channels having ppv events and to purchase ppv events (either by policy group or product). For a ws 'get' calls the account number or the smart card id should be provided. The account is derived from the account number if provided, else derived from the smart card id. If both account number and smart card id are provided, the smart card should be account related. For HUE calls the account is derived from the request context.

rrajendram @author

63.1.1 PPVService.getChannelListWithPPVEvents

Return a list of channels with PPV events for a given search criteria.

Date range filter logic- 1) If startDate and endDate provided - ppvEvents are filtered by supplied date range. 2) If only startDate provided - endDate is set to startDate 3) If only the endDate provided - startDate set to NOW, unless endDate before NOW. 4) If start and end dates not provided - startDate = endDate = NOW.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|--|
| accountId | String | unique account identifier |
| startDate | Date | optional date to start search from |
| endDate | Date | optional date to start search to |
| locale | String | optional country specific code eg en_GB (Defaults to Qsp locale) |

Return value

Type: `List<Channel>`

a list of purchasable BTV event identifiers

63.1.2 PPVService.getPPVEventList

Return a list of user purchasable and/or purchased PPV events for a given search criteria.

Date range filter logic- 1) If startDate and endDate provided - ppvEvents are filtered by supplied date range. 2) If only startDate provided - endDate is set to startDate 3) If only the endDate provided - startDate set to NOW, unless endDate before NOW. 4) If start and end dates not provided - startDate = endDate = NOW.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---|
| accountId | String | unique account identifier |
| smartcardId | String | optional smartcard identifier associated with an active device. Only return events associated with this smartcard |
| startDate | Date | optional date to start search from |
| endDate | Date | optional date to start search to |
| locale | String | optional country specific code eg en_GB (Defaults to Qsp locale) |

Return value

Type: List<PPVEvent>

a list of purchasable BTV event identifiers

63.1.3 PPVService.getPPVEventListByChannel

Return a list of user purchasable PPV and/or purchased ppvevents for a given search criteria by channel.

Date range filter logic- 1) If startDate and endDate provided - ppvEvents are filtered by supplied date range. 2) If only startDate provided - endDate is set to startDate 3) If only the endDate provided - startDate set to NOW, unless endDate before NOW. 4) If start and end dates not provided - startDate = endDate = NOW.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---|
| accountId | String | unique account identifier |
| smartcardId | String | optional smartcard identifier associated with an active device. Only return events associated with this smartcard |

| Name | Type | Description |
|------------------------------|--------|--|
| <code>startDate</code> | Date | optional date to start search from |
| <code>endDate</code> | Date | optional date to start search to |
| <code>channelNetCarry</code> | String | channel Identifier |
| <code>locale</code> | String | optional country specific code eg en_GB (Defaults to Qsp locale) |

Return value

Type: `List<PPVEvent>`

a list of purchasable BTV event identifiers

63.1.4 PPVService.getPagedChannelListWithPPVEvents

Get pages from the list of *Channels* which have pay-per-view events during a specified period available for purchase.

The search includes all *Events* which are in progress at any time between the `startDate` and `endDate`, if both are provided. You can also leave either date null.

- ▶ If only the `startDate` is provided, SDP considers only *Events* which are in progress at that time.
- ▶ If only the `endDate` is provided, and it is in the future, SDP considers all *Events* between now and the `endDate`.
- ▶ If only the `endDate` is provided, and it is in the past, SDP considers only *Events* which are in progress at that time.
- ▶ If no dates are provided SDP considers only *Events* which are currently in progress.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/PPVService/getPagedChannelListWithPPVEvents`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|--|
| <code>accountId</code> | String | The <code>uid</code> of the <i>Account</i> . Only required when using this API from a Web Services client. |
| <code>startDate</code> | Date | The start date and time of the time period to investigate. |
| <code>endDate</code> | Date | The end date and time of the time period to investigate. |

| Name | Type | Description |
|-----------------------|--------|---|
| <code>startRow</code> | Long | The index of the first result to return, or null to start at the beginning of the list. |
| <code>endRow</code> | Long | The index of the last result to return, or null to stop at the end of the list. |
| <code>locale</code> | String | The locale to use when returning localizable fields of the <i>Channels</i> . |

Return value

Type: `List<Channel>`

63.1.5 PPVService.getPagedPPVEventList

Get a page from the list of pay-per-view *Events* available in the specified time period.

The search includes all *Events* which are in progress at any time between the `startDate` and `endDate`, if both are provided. You can also leave either date null.

- ▶ If only the `startDate` is provided, SDP considers only *Events* which are in progress at that time.
- ▶ If only the `endDate` is provided, and it is in the future, SDP considers all *Events* between now and the `endDate`.
- ▶ If only the `endDate` is provided, and it is in the past, SDP considers only *Events* which are in progress at that time.
- ▶ If no dates are provided SDP considers only *Events* which are currently in progress.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/PPVService/getPagedPPVEventList`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>accountId</code> | String | The <code>uid</code> of the <i>Account</i> . This is only required when using this API from a Web Services client. It is used to determine which pay-per-view <i>Policy Groups</i> are available to the user, and to populate the <code>isSubscribed</code> field of the returned <i>Event</i> objects. |
| <code>smartcardId</code> | String | The ID of the <i>Smart Card</i> associated with the active <i>Set Top Box</i> . This is only required when using this API from a Web Services client. It is used to populate the <code>isSubscribed</code> field of the returned <i>Event</i> objects. |

| Name | Type | Description |
|------------------------|--------|---|
| <code>startDate</code> | Date | The start date and time of the time period to investigate. |
| <code>endDate</code> | Date | The end date and time of the time period to investigate. |
| <code>startRow</code> | Long | The index of the first result to return, or null to start at the beginning of the list. |
| <code>endRow</code> | Long | The index of the last result to return, or null to stop at the end of the list. |
| <code>locale</code> | String | The locale to use when returning localizable fields of the returned objects. |

Return value

Type: `List<PPVEvent>`

63.1.6 PPVService.getPagedPPVEventListByChannel

Get a page from the list of pay-per-view *Events* available in the specified time period on the specified *Channel*.

The search includes all *Events* which are in progress at any time between the `startDate` and `endDate`, if both are provided. You can also leave either date null.

- ▶ If only the `startDate` is provided, SDP considers only *Events* which are in progress at that time.
- ▶ If only the `endDate` is provided, and it is in the future, SDP considers all *Events* between now and the `endDate`.
- ▶ If only the `endDate` is provided, and it is in the past, SDP considers only *Events* which are in progress at that time.
- ▶ If no dates are provided SDP considers only *Events* which are currently in progress.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/PPVService/getPagedPPVEventListByChannel`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---|
| <code>accountId</code> | String | The <code>uid</code> of the <i>Account</i> . This is only required when using this API from a Web Services client. It is used to determine which pay-per-view <i>Policy Groups</i> are available to the user, and to populate the <code>isSubscribed</code> field of the returned <i>Event</i> objects. |

| Name | Type | Description |
|------------------------------|--------|--|
| <code>smartcardId</code> | String | The ID of the <i>Smart Card</i> associated with the active <i>Set Top Box</i> . This is only required when using this API from a Web Services client. It is used to populate the <code>isSubscribed</code> field of the returned <i>Event</i> objects. |
| <code>startDate</code> | Date | The start date and time of the time period to investigate. |
| <code>endDate</code> | Date | The end date and time of the time period to investigate. |
| <code>channelNetCarry</code> | String | The <code>carryingNetwork</code> of the <i>Channel</i> to investigate. |
| <code>startRow</code> | Long | The index of the first result to return, or null to start at the beginning of the list. |
| <code>endRow</code> | Long | The index of the first result to return, or null to stop at the end of the list. |
| <code>locale</code> | String | The locale to use when returning localizable fields of the returned objects. |

Return value

Type: `List<PPVEvent>`

63.1.7 PPVService.getPagedPurchasedPPVEventList

Get a page from the list of pay-per-view events which the user has already purchased, which will be shown during the specified time window.

The search includes all *Events* which are in progress at any time between the `startDate` and `endDate`, if both are provided. You can also leave either date null.

- ▶ If only the `startDate` is provided, SDP considers only *Events* which are in progress at that time.
- ▶ If only the `endDate` is provided, and it is in the future, SDP considers all *Events* between now and the `endDate`.
- ▶ If only the `endDate` is provided, and it is in the past, SDP considers only *Events* which are in progress at that time.
- ▶ If no dates are provided SDP considers only *Events* which are currently in progress.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/PPVService/getPagedPurchasedPPVEventList`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>accountId</code> | String | The <code>uid</code> of the <i>Account</i> . This is only required when using this API from a Web Services client. |
| <code>smartcardId</code> | String | The ID of the <i>Smart Card</i> associated with the active <i>Set Top Box</i> . This is only required when using this API from a Web Services client. |
| <code>startDate</code> | Date | The start date and time of the time period to investigate. |
| <code>endDate</code> | Date | The end date and time of the time period to investigate. |
| <code>startRow</code> | Long | The index of the first result to return, or null to start at the beginning of the list. |
| <code>endRow</code> | Long | The index of the last result to return, or null to stop at the end of the list. |
| <code>locale</code> | String | The locale to use when returning localizable fields of the returned objects. |

Return value

Type: `List<PPVEvent>`

63.1.8 PPVService.getPurchasedPPVEventList

Return a list of user subscribed PPV events for a given search criteria.

Date range filter logic- 1) If `startDate` and `endDate` provided - `ppvEvents` are filtered by supplied date range. 2) If only `startDate` provided - `endDate` is set to `startDate` 3) If only the `endDate` provided - `startDate` set to NOW, unless `endDate` before NOW. 4) If start and end dates not provided - `startDate` = `endDate` = NOW.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>accountId</code> | String | unique account identifier |
| <code>smartcardId</code> | String | optional smartcard identifier associated with an active device. Only return events associated with this smartcard |

| Name | Type | Description |
|------------------------|--------|--|
| <code>startDate</code> | Date | optional date to start search from |
| <code>endDate</code> | Date | optional date to start search to |
| <code>locale</code> | String | optional country specific code eg en_GB (Defaults to Qsp locale) |

Return value

Type: `List<PPVEvent>`

a list of purchasable BTV event identifiers

63.1.9 PPVService.purchasePPVEventByPolicyGroup

Subscribes an *Account* and optional *Smart Card* to a *Policy Group* providing a pay-per-view *Event*, where the `uid` of the *Policy Group* is known.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/PPVService/purchasePPVEventByPolicyGroup`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>accountId</code> | String | The <code>uid</code> of the <i>Account</i> to subscribe, when this API is called through Web Services. If called through HUE, the signed-on <i>Account</i> is always used, so this parameter should not be supplied. |
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID of the device to give access to the content, or null to leave this unspecified. If specified, the content will be made available to this device only; otherwise, the content will be made available across the whole <i>Account</i> . If this and the <i>Account</i> are both specified, the device must be provisioned to the <i>Account</i> supplied. |
| <code>userUID</code> | Long | The <code>uid</code> of the <i>User</i> making the subscription, or null to leave this unspecified. This is used to identify the subscribing user in subscription and billing records: it does not affect who is given access to the content. If a <i>User</i> is specified, it must be linked to the <i>Account</i> supplied, or to |

| Name | Type | Description |
|-----------------------------|--------|--|
| | | the <i>Account</i> associated with the <i>Smart Card</i> supplied (in the case where <code>accountUid</code> is null.) |
| <code>policyGroupUid</code> | Long | The <code>uid</code> of the <i>Policy Group</i> representing the product to subscribe. |
| <code>locale</code> | String | An optional locale code. This determines the locale used for localizable fields in <i>Acquired Content List</i> and <i>Billing</i> Entities created by this subscription. If this field is not supplied, SDP will use its configured default locale. |

Return value

Type: `List<Long>`

A list containing the `uid` of each created *Acquired Content List* Entity.

Example

```
<purchasePPVEventByPolicyGroup>
  <accountId>stringValue</accountId>
  <smartcardId>stringValue</smartcardId>
  <userId>44</userId>
  <policyGroupUid>83</policyGroupUid>
  <locale>en_us</locale>
</purchasePPVEventByPolicyGroup>
```

63.1.10 PPVService.purchasePPVEventByProduct

Subscribes an *Account* and optional *Smart Card* to a *Policy Group* providing a pay-per-view *Event*, where the `casId` of the *Policy Group* is known.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/PPVService/purchasePPVEventByProduct`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>accountId</code> | String | The <code>accountNumber</code> of the <i>Account</i> . |
| <code>smartcardId</code> | String | The <i>Smart Card</i> ID of the device to give access to the content, or null to leave this unspecified. If specified, |

| Name | Type | Description |
|------------------|--------|---|
| | | the content will be made available to this device only; otherwise, the content will be made available across the whole <i>Account</i> . If this and the <i>Account</i> are both specified, the device must be provisioned to the <i>Account</i> supplied. |
| userID | Long | The uid of the <i>User</i> making the subscription, or null to leave this unspecified. This is used to identify the subscribing user in subscription and billing records: it does not affect who is given access to the content. If a <i>User</i> is specified, it must be linked to the <i>Account</i> supplied, or to the <i>Account</i> associated with the <i>Smart Card</i> supplied (in the case where accountUID is null.) |
| productId | String | The casId of the <i>Policy Group</i> representing the product to subscribe. |
| locale | String | An optional locale code. This determines the locale used for localizable fields in <i>Acquired Content List</i> and <i>Billing Entities</i> created by this subscription. If this field is not supplied, SDP will use its configured default locale. |

Return value

Type: `List<Long>`

A list containing the **uid** of each created *Acquired Content List* Entity.

Example

```
<purchasePPVEventByProduct>
  <accountId>stringValue</accountId>
  <smartcardId>stringValue</smartcardId>
  <userID>22</userID>
  <productId>stringValue</productId>
  <locale>en_us</locale>
</purchasePPVEventByProduct>
```

64 preferenceService

This is a primitive *Service* which provides the *Preference* Entity.

64.1 Preference

The *Preference* Entity.

No content

64.1.1 Preference

| Name | Type | Description |
|----------------|---|---|
| accountUID | Long | Defines the account uid associated to the preference. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| creationDate | Date | The date created. |
| domain | String | Defines the domain associated to the preference. They are always in upper case and words are separated by underscores. Example: BTV, NVOD, COD, COR, AM etc. |
| exportID | Integer | The id to export object to third part systems. |
| modifiedDate | Date | The date last modified and persisted. |
| originID | Integer | The uid of the origin system (3rd part system). |
| originIdAndKey | <i>Read only.</i> String | A unique reference for the object derived by concatenating the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| originKey | String | The key of the model object in the origin system (3rd part system). |
| paramName | String | Defines the parameter name associated to the preference. They are always in upper case and words are separated |

| Name | Type | Description |
|--------------------------------|---------------------|--|
| | | by underscores. Example: LOCALE, COUNTRY, LANGUAGE, CURRENCY, BOOT_STATUS etc. |
| <code>paramType</code> | <code>String</code> | Defines the parameter type associated to the preference. Example: Long, String, Boolean etc. |
| <code>paramValue</code> | <code>String</code> | Defines the parameter value associated to the preference. |
| <code>serviceProviderID</code> | <code>Long</code> | The Service provider UID. |
| <code>uID</code> | <code>Long</code> | The unique identifier. |
| <code>userUID</code> | <code>Long</code> | Defines the user uid associated to the preference. |

64.2 preferenceService

Preference Service Implementation. Implementation which performs the compulsory methods defined in the Preference Service interface. PreferenceService implemented object is used as WebServices to communicate with the Account Management (AM) system The service is made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true, instrument = false, roles = "<allowed-roles>") The methods used to create, delete or update items of the AM System, only could be accesible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue.

Ashish Gupta @author

1.0 @version

64.2.1 preferenceService.create

Create (persist) a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

64.2.2 preferenceService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

64.2.3 preferenceService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
<originKey>stringValue</originKey>
```

```
<originID>64</originID>
</getByOrigin>
```

64.2.4 preferenceService.getByOriginUID

A Service method to get the preference details list for origin uid.

Contains the business logic required to get preference details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|---------|--------------------|
| originUID | Integer | Integer origin uid |
| accountUID | Long | Long account uid |
| domain | String | String |
| userID | Long | Long user uid |

Return value

Type: `List<Preference>`

List of Preference object

Example

```
<getByOriginUID>
<originUID>1</originUID>
<accountUID>12</accountUID>
<domain>stringValue</domain>
<userID>64</userID>
</getByOriginUID>
```

64.2.5 preferenceService.getByUID

Retrieve, from storage, the `BaseMode1` with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>68</uid>
</getByUID>
```

64.2.6 preferenceService.getList

A Service method to get the preference details list for user uid.

Contains the business logic required to get preference details list.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/preferenceService/getList>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|------------------|
| accountUID | Long | Long account uid |
| domain | String | String |
| userUID | Long | Long user uid |

Return value

Type: `List<Preference>`

List of Preference object

Example

```
<getList>
  <accountUID>35</accountUID>
  <domain>stringValue</domain>
  <userUID>89</userUID>
```

</getList>

64.2.7 preferenceService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

65 presenceService

This is an aggregate *Service* which manages the relationship of *Accounts* and *Users*.

Managing Users

The `addUserToAccount` method should be used to associate new *Users* to an *Account*.

`addUserToAccount` first checks the `status` of the *Account*, as only *Accounts* with a `status` of `Active`, `Restricted` or `suspended` may be associated with *Users*.

If the `status` is valid, the *User Service* is used to update the *User* with the UID of the specified *Account*.

The `removeUserFromAccount` method uses the *User Service* to update the *User* with a null `accountUID`.

65.1 tv.quative.service.user.model.PurchaseAbility

| Name | Type | Description |
|---------|------|-------------|
| ALLOWED | | Allow all |
| DENIED | | Deny all |

65.2 User

A *User* represents an end user who is entitled to view digital TV and media services.

Figure 21 User and associated Entities.

Every active *User* is attached to a single *Account*. Different *Users* on the same *Account* may have different privileges, e.g. to access rated content or other services.

Users can have either normal or super-user access privileges. Super-users of an *Account* may be able to add new *Users*, manage *Ratings*, or perform *Account* administration, if the application supports it. Every *Account* always has at least one super-user.

Every *Account* always has a default *User*. This is the *User* which is assumed to be active when the set-top box is first switched on. The application is responsible for determining what this means.

When an end user makes a purchase, the application may notify SDP which *User* is currently active. This information is passed through to billing and subscriber management systems to help them gather statistics. However, the purchase itself is registered to the *Account*, not to the individual *User*.

A *User* may have a *Rating* to denote their morality level for parental control purposes.

Note

Many deployments manage user information entirely on the client side, storing login details and enforcing parental control locally. In these deployments, there is only one *User* Entity for each *Account*.

65.2.1 User

| Name | Type | Description |
|---------------------------|---|---|
| <code>acctUID</code> | Long | The UID of the parent <i>Account</i> . Note In QSP Manager, this field is set by the Account Number control, which uses the <i>Account</i> number, rather than the UID. |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created. |
| <code>defaultUser</code> | Boolean | Is this <i>User</i> the default <i>User</i> for the associated <i>Account</i> ? Exactly one <i>User</i> of an <i>Account</i> must always be the default <i>User</i> for that <i>Account</i> . Note You cannot unset this field, because that would leave the <i>Account</i> without a default <i>User</i> . When you make another <i>User</i> the default, this field will be updated automatically. |
| <code>dob</code> | Date | The date of birth of the user, in the format <yyyy>-<mm>-<dd>. |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>hashSalt</code> | <i>Not visible to HUE.</i> String | The salt value used to encrypt the password in the SDP database. SDP protects passwords using the PBKDF2 hashing function defined in PKCS#5. The parameters of this function are |

| Name | Type | Description |
|-----------------------------|--|--|
| | | <ul style="list-style-type: none"> ▶ the iteration count, configured in <code>qspconf</code>; ▶ the hash length, configured in <code>qspconf</code>; and ▶ the hash salt provided in this field. <p>If this field is blank, the password will be stored in clear text.</p> |
| <code>loginID</code> | <i>Search criterion.</i> <code>String</code> | <p>The unique ID which a client device will use to identify this <i>User</i> when signing on, in deployments using authentication with a login ID and password. This field must be unique across all <i>Users</i> if present.</p> <p><i>For example: carsten.scheller@happytv.com</i></p> <p>NAGRA Media Player clients always use this type of authentication, so this field must be supplied in these deployments.</p> |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> <code>String</code> | The name of this <i>Entity</i> . |
| <code>originID</code> | <code>Integer</code> | <p>An identifier for the system which owns this <i>Entity</i>.</p> <p>Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above.</p> |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>password</code> | <i>Not visible to HUE.</i> <code>String</code> | <p>The password which a client device will use when this <i>User</i> signs on, in deployments using authentication with a login ID and password.</p> <p>NAGRA Media Player clients always use this type of authentication, so this field must be supplied in these deployments.</p> |

| Name | Type | Description | | | | | | |
|--------------------------------|---|---|-------|-------------|-----|---|-----|--|
| <code>pin</code> | <i>Not visible to HUE.</i> String | A number or string which users may be expected to provide to obtain access to rating-restricted content. This field is only used in deployments where users share a PIN across several devices. Where users only have one device, the PIN is more often stored locally. | | | | | | |
| <code>purchaseAbility</code> | String | Is this <i>User</i> permitted to make purchases? This field may be updated by HUE clients as long as they are signed in as a superuser of the <i>User's Account</i> . | | | | | | |
| <code>ratingID</code> | Long | The UID of the <i>Rating</i> which describes the content which it is appropriate for this <i>User</i> to view. | | | | | | |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . | | | | | | |
| <code>type</code> | String | The type of the <i>User</i> . One of: <table border="1" data-bbox="794 1064 1401 1581"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>SUP</td> <td>(Super) This <i>User</i> is a super-user for the <i>Account</i>. They may be able to access <i>Account</i> settings, add and remove other <i>Users</i>, or perform other administration depending on the deployment and application.</td> </tr> <tr> <td>NOR</td> <td>(Normal) The <i>User</i> is a normal user. They can access services provided to the <i>Account</i>, subscribe to content and view it, but they cannot perform <i>Account</i> administration.</td> </tr> </tbody> </table> | Value | Description | SUP | (Super) This <i>User</i> is a super-user for the <i>Account</i> . They may be able to access <i>Account</i> settings, add and remove other <i>Users</i> , or perform other administration depending on the deployment and application. | NOR | (Normal) The <i>User</i> is a normal user. They can access services provided to the <i>Account</i> , subscribe to content and view it, but they cannot perform <i>Account</i> administration. |
| Value | Description | | | | | | | |
| SUP | (Super) This <i>User</i> is a super-user for the <i>Account</i> . They may be able to access <i>Account</i> settings, add and remove other <i>Users</i> , or perform other administration depending on the deployment and application. | | | | | | | |
| NOR | (Normal) The <i>User</i> is a normal user. They can access services provided to the <i>Account</i> , subscribe to content and view it, but they cannot perform <i>Account</i> administration. | | | | | | | |

Note

At least one *User* of an *Account* must be a super-user. SDP will only allow you to set this field to **Normal** if the *Account* already has another *User* whose **Type** is **Super**.

| Name | Type | Description |
|------|------|---|
| uID | Long | The SDP internal UID for this <i>Entity</i> . |

65.3 presenceService

Provides all of the required functionality for the Presence service.

Steve @author

Ashish Gupta redefined this service apis @author

Naeem Bilal @author

Binod Singh @author

65.3.1 presenceService.addUserToAccount

Associate an existing User to an Account via HUE..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/presenceService/addUserToAccount>

Parameters

| Name | Type | Description |
|---------------|---------|--|
| acctUID | Long | is the UID of an account for the User. |
| loginID | String | the login identifier of the User |
| ratingID | Long | is the UID of the rating table. |
| pin | String | security string to access device. |
| isDefaultUser | boolean | |
| name | String | consider as user's name. |
| type | String | either be SUP (Super) or NOR (Normal) |
| dob | Date | is Date of Birth of the user |

| Name | Type | Description |
|------------------------|---------|--|
| <code>password</code> | String | this is the hashed password value. |
| <code>hashSalt</code> | String | this is the salt needed to compare the password. |
| <code>originUID</code> | Integer | |
| <code>originKey</code> | String | |

Return value

Type:

The uid of the newly created user record

65.3.2 presenceService.addUserToAccount

Associate an existing User to an Account via WS..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------------------------------------|--------------------------------------|
| <code>accountNo</code> | String | the identifier of the parent Account |
| <code>user</code> | User ^{p.710} | as an User object. |

Return value

Type:

The uid of the newly created user

65.3.3 presenceService.modifyUserOfAccount

Update an existing User details associated from an Account.

The account association can not be changed.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---------------------------------------|--------------------------------|
| <code>user</code> | User ^{p.710} | The User object to be updated. |

Return value

Type:

The number of rows updated

65.3.4 presenceService.removeUserByNameAndAccountNumber

Disassociates an existing User from an Account and deletes the user.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|--------|--|
| <code>name</code> | String | The User name. This is unique for a given account. |
| <code>accountNumber</code> | String | The user account number. |

Return value

Type:

The number of rows deleted.

Example

```
<removeUserByNameAndAccountNumber>
  <name>stringValue</name>
  <accountNumber>ABC-45513</accountNumber>
</removeUserByNameAndAccountNumber>
```

65.3.5 presenceService.removeUserFromAccount

Disassociates an existing User from an Account and deletes the user.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|----------------------------------|
| userLoginId | String | the login identifier of the User |

Return value

Type:

The number of rows updated

Example

```
<removeUserFromAccount>  
  <userLoginId>stringValue</userLoginId>  
</removeUserFromAccount>
```

66 profileService

This is a primitive *Service* which provides *Profile* Entities, including *Device Profile* and *Subscription Profile*.

All *Profile* Entities represent a logical grouping of other *Entities*:

- ▶ **Device Profiles** group *Features* and other *Device Profiles*.
- ▶ **Subscription Profiles** group *Packages*, *Catalogues* and other *Subscription Profiles*.

66.1 profileService

Provides functionalities to Create, Retrieve, Update and Delete Profile accounts on the system.

zlaposa @author

Naeem Bilal @author

66.1.1 profileService.addCatToSbscrProf

Maps a catalogue to a profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|------|----------------|
| profileUID | Long | Profile UID. |
| catalogueUID | Long | Catalogue UID. |

Return value

Type:

Integer Number of updated rows..

Example

```
<addCatToSbscrProf>
  <profileUID>93</profileUID>
  <catalogueUID>41</catalogueUID>
</addCatToSbscrProf>
```

66.1.2 profileService.addFeatureToDProf

Maps a feature to a device profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------|
| profileUID | Long | Profile UID. |
| featureUID | Long | Feature UID. |

Return value

Type:

Integer Number of updated rows.

Example

```
<addFeatureToDProf>
  <profileUID>14</profileUID>
  <featureUID>76</featureUID>
</addFeatureToDProf>
```

66.1.3 profileService.addFeatureToProf

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------|---|-------------|
| profileUID | Long | |
| featureUID | Long | |
| theRequired ProfileType | tv.quative.service.profile.model.ProfileType ^{p.404} | |

Return value

Type:

66.1.4 profileService.addPkgToSbscrProf

Maps a package to a profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------|
| profileUID | Long | Profile UID. |
| packageUID | Long | Package UID. |

Return value

Type:

Integer Number of updated rows.

Example

```
<addPkgToSbscrProf>
  <profileUID>79</profileUID>
  <packageUID>91</packageUID>
</addPkgToSbscrProf>
```

66.1.5 profileService.addProfile

Maps a profile to another profile..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|------------------------------------|
| profileUID | Long | Profile UID of the parent profile. |
| childProfileUID | Long | Profile UID of the child profile. |

Return value

Type:

Integer Number of updated rows.

Example

```
<addProfile>
  <profileUID>85</profileUID>
  <childProfileUID>58</childProfileUID>
</addProfile>
```

66.1.6 profileService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

66.1.7 profileService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------|
| <code>model</code> | Profile ^{p.144} | |

Return value

Type:

66.1.8 profileService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

66.1.9 profileService.getALaCarteProfiles

Returns a list of profiles `Profile` belonging marked as A La Carte.

Available through Web Services.

Parameters

None.

Return value

Type: `List<Profile>`

List List of child profiles belonging to the given profile.

Example

```
<getALaCarteProfiles/>
```

66.1.10 profileService.getByFeatureUIDAndProfileType

A Service method to get the profile details list for the feature uid and profile type.

Contains the business logic required to get device profile details list.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/profileService/getByFeatureUIDAndProfileType`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|-------------|
| <code>featureUID</code> | Long | Integer |
| <code>profileType</code> | String | String |

Return value

Type: `List<Profile>`

List of Profile objects

Example

```
<getByFeatureUIDAndProfileType>
  <featureUID>18</featureUID>
  <profileType>stringValue</profileType>
</getByFeatureUIDAndProfileType>
```

66.1.11 profileService.getByName

Returns a profile with the given name.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|--------------------------|
| <code>name</code> | String | The name of the profile. |

Return value

Type:

Profile The profile with the given name.

Example

```
<getByName>
  <name>stringValue</name>
</getByName>
```

66.1.12 profileService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>32</originID>
</getByOrigin>
```

66.1.13 profileService.getByType

Returns a list of profiles with the given type.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|--------------------------|
| type | String | The type of the profile. |

Return value

Type: `List<Profile>`

List Profile list returned by the query.

Example

```
<getByType>
  <type>stringValue</type>
</getByType>
```

66.1.14 profileService.getByUID

Retrieve, from storage, the `BaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/profileService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

`BaseModel` If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>63</uid>
</getByUID>
```

66.1.15 profileService.getEmptyProfiles

Returns a list of profiles `Profile` that have no features.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--------|-------------|
| type | String | |

Return value

Type: `List<Profile>`

List List of featureless profiles.

Example

```
<getEmptyProfiles>
  <type>stringValue</type>
</getEmptyProfiles>
```

66.1.16 profileService.getEmptyProfileUids

Returns a list of profile `Profile` uids for those that have no features.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--------|-------------|
| <code>type</code> | String | |

Return value

Type: `List<Long>`

List List of featureless profiles.

Example

```
<getEmptyProfileUids>
  <type>stringValue</type>
</getEmptyProfileUids>
```

66.1.17 profileService.getFeatureValueForAccount *New*

Overview

Get feature value for account from the profileService.

This will retrieve the profile feature value based on the account uid and features name key.

Description

This operation requires as mandatory inputs `accountUid` and `featureNameKey` and `token` but optionally `deviceInfo`, `drmDeviceId` and returns the profile feature value in the response.

This operation can be called in the following scenarios:

1. If both mandatory input passed then operation validates input then returns the valid profile feature value.
2. if any mandatory filed missing will the get the custom exception.

Definition

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/js/profileService/getFeatureValueForAccount`

Available through WS at `http://<server>:<port>/ws-gateway/gateway/ws/profileService/getFeatureValueForAccount`

Request input parameters

| Name | Type | Description | Cardinality |
|------|--------|----------------|-------------|
| arg0 | Long | account UID | mandatory |
| arg1 | String | featureNameKey | mandatory |

Requests examples

1. WS Request with all mandatory fields:

```
<soapenv:Body>
<prof:getFeatureValueForAccount>
<accountUID>1</accountUID>
<featureNameKey>FEATURE1</featureNameKey>
</prof:getFeatureValueForAccount>
</soapenv:Body>
```

2. Hue Request with all mandatory fields:

```
http://<server>/hue-gateway/gateway/http/js/profileService/getFeatureValueForAccount?
arg0=1&arg1=FEATURE1&token=fHcM7NZaZcWtiBif2HLrCeN57itdd3vuLHN1YnNjcmliZXItMSwxLEtCUk8s
MSw2NzQ1Mjg0OSwsLDEsLCxvYW0tZGVmYXVsdC1oaWRkZW4%3D
```

Response example

Ws valid Response:

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">1943554333</rqid:request
ID>
  </env:Header>
  <env:Body>
    <ns2:getFeatureValueForAccountResponse xmlns:ns2="http://quative.tv/ProfileService
Namespace">
      <return>BASIC</return>
    </ns2:getFeatureValueForAccountResponse>
  </env:Body>
</env:Envelope>
```

Hue valid response:

```
<Response xmlns="http://10.15.2.60/hue-gateway/gateway/http/js/profileService/getFeature
ValueForAccount">
```

```
<requestId>1250786843</requestId>
<result>BASIC</result>
<resultCode>0</resultCode>
<token null="true"/>
</Response>
```

3. Request with no mandatory input:

http://<server>/hue-gateway/gateway/http/js/profileService/getFeatureValueForAccount?token=fHcM7NZaZcWtiBif2HLrCeN57itdd3vuLHN1YnNjcmliZXItMSwxLEtCUk8sMSw2NzQ1Mjg0OSwsLDEsLCxvYW0tZGVmYXVsdC1oaWRkZW4%3D

Error response:

```
<Response xmlns="http://10.15.2.60/hue-gateway/gateway/http/js/profileService/getFeatureValueForAccount">
<localeMessage>none</localeMessage>
<requestId>1681503110</requestId>
<result>Service method profileService.getFeatureValueForAccount throw exception "tv.quative.service.Service
Exception: ErrorCode: [code=14038, severity=ERROR] Invalid parameters passed" </result>
<resultCode>14038</resultCode>
</Response>
```

66.1.18 profileService.getPagedAlacarteProfilesByChannelNetCarry

getPagedAlacarteProfilesByChannelNetCarry returns the alacarte Profiles.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|--------|----------------------------|
| channelNetCarry | String | unique channel identifier. |
| locale | String | Country code eg en_GB |
| startRowNum | Long | Long |
| endRowNum | Long | Long |

Return value

Type: List<Profile>

List of Packages.

Example

```
<getPagedAlacarteProfilesByChannelNetCarry>
<channelNetCarry>stringValue</channelNetCarry>
<locale>en_us</locale>
<startRowNum>13</startRowNum>
<endRowNum>54</endRowNum>
</getPagedAlacarteProfilesByChannelNetCarry>
```

66.1.19 profileService.getProfiles

Returns a list of profiles belonging to the specified profile.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|---------|---|
| profileUID | Long | UID of the profile. |
| recursive | boolean | If true , it also returns the children of the children and so on |

Return value

Type: List<Profile>

List List of child profiles belonging to the given profile.

66.1.20 profileService.removeCatFromSbscrProf

Removes catalogue-profile mapping..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|------|----------------|
| profileUID | Long | Profile UID. |
| catalogueUID | Long | Catalogue UID. |

Return value

Type:

Integer The number of the deleted rows.

Example

```
<removeCatFromSbscrProf>
  <profileUID>60</profileUID>
  <catalogueUID>19</catalogueUID>
</removeCatFromSbscrProf>
```

66.1.21 profileService.removeDeviceTypeFromDvcProf *New*

Removes an association of DeviceTypeList with Device Profile

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|----------------|
| profileUID | Long | as Profile UID |

Return value

Type:

Integer as affected rows

Example

```
<http://profremoveDeviceTypeFromDvcProf> <profileUID>21</profileUID></http://profremoveDeviceTypeFromDvcProf>
```

66.1.22 profileService.removeFeatureFromDProf

Removes feature-device profile mapping..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------|
| profileUID | Long | Profile UID. |
| featureUID | Long | Feature UID. |

Return value

Type:

Integer The number of the deleted rows.

Example

```
<removeFeatureFromDProf>
  <profileUID>23</profileUID>
  <featureUID>70</featureUID>
</removeFeatureFromDProf>
```

66.1.23 profileService.removeFeatureFromSrcvProf

Removes feature-service profile mapping..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------|
| profileUID | Long | Profile UID. |
| featureUID | Long | Feature UID. |

Return value

Type:

Integer The number of the deleted rows.

Example

```
<removeFeatureFromSrcvProf>
  <profileUID>53</profileUID>
  <featureUID>95</featureUID>
</removeFeatureFromSrcvProf>
```

66.1.24 profileService.removePkgFromSbscrProf

Removes package-profile mapping..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|------|--------------|
| profileUID | Long | Profile UID. |
| packageUID | Long | Package UID. |

Return value

Type:

Integer The number of the deleted rows.

Example

```
<removePkgFromSbscrProf>
  <profileUID>60</profileUID>
  <packageUID>70</packageUID>
</removePkgFromSbscrProf>
```

66.1.25 profileService.removeProfile

Removes a catalogue-subscription profile mapping..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|------|------------------------------------|
| profileUID | Long | Profile UID of the parent profile. |
| childProfileUID | Long | Profile UID of the child profile. |

Return value

Type:

Integer The number of the deleted rows.

Example

```
<removeProfile>
  <profileUID>45</profileUID>
  <childProfileUID>81</childProfileUID>
</removeProfile>
```

66.1.26 profileService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

66.1.27 profileService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------|
| <code>model</code> | Profile <small>p.144</small> | |

Return value

Type:

66.1.28 profileService.updateFeatureProfileValue

Maps a feature to a profile with a value..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|--------------|
| <code>profileUID</code> | Long | Profile UID. |
| <code>featureUID</code> | Long | Feature UID. |

| Name | Type | Description |
|--------------|--------|-------------|
| profileValue | String | String |

Return value

Type:

Integer Number of updated rows.

Example

```
<updateFeatureProfileValue>  
<profileUID>88</profileUID>  
<featureUID>88</featureUID>  
<profileValue>stringValue</profileValue>  
</updateFeatureProfileValue>
```

67 programService

This is a primitive *Service* which provides the BTV *Programme* Entity.

67.1 Verbose Program

A *Verbose Program* is an internal *Entity* which adds extra fields to a standard *Program*.

67.1.1 VerboseProgram

| Name | Type | Description |
|---------------|---|--|
| actor | String | The names of actors appearing in this <i>Program</i> , separated by semicolons. The CMS ingest places the <i>Program</i> object's Actors EPG attribute in this field. |
| aspect | String | The aspect ratio of this <i>Program</i> . <i>For example:</i> <ul style="list-style-type: none"> ▶ 16:9 ▶ 4:3 ▶ Widescreen The CMS ingest places the <i>Program</i> object's Aspect EPG attribute in this field. |
| audioMode | String | The audio properties of this <i>Program</i> . |
| audioPID | List<ProgramAltAudioPID> | The value of an alternative audio PID. This is used when playing a program that contains multiple audio streams, to correctly decode the correct stream. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| contentRating | AggregateRating | |
| copyright | String | The copyright status and/or owner of this <i>Program</i> . This field is not managed by SDP; operators may decide how to use it. |

| Name | Type | Description |
|---------------------|----------------|---|
| countries | String | <p>A list of 2-letter ISO codes of countries for the <i>Program</i>.</p> <p><i>For example: us, fr, cn</i></p> <p>The CMS ingest places the <i>Program</i> object's Country of origin EPG attribute in this field.</p> |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| definition | String | <p>The picture definition of this <i>Program</i>.</p> <p><i>For example:</i></p> <ul style="list-style-type: none"> ▶ HD ▶ SD <p>The CMS ingest places the <i>Program</i> object's Definition EPG attribute in this field.</p> |
| description | String | <p>A short description of this <i>Program</i>.</p> <p>The CMS ingest places the <i>Program</i> object's Description EPG attribute in this field.</p> |
| director | String | <p>The names of the directors of this <i>Program</i>, separated by semicolons.</p> <p>The CMS ingest places the <i>Program</i> object's Directors EPG attribute in this field.</p> |
| duration | Integer | <p>The duration of this <i>Program</i>, in milliseconds.</p> <p>The CMS ingest places the <i>Program</i> object's Duration in this field, or works it out from the Start Date and End Date.</p> |
| episode | String | |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| genre | String | <p>A list of genres to which this <i>Program</i> belongs, separated by semicolons.</p> <p>The CMS ingest sets this field from the Categories EPG attribute of the <i>Program</i> object.</p> |

| Name | Type | Description |
|------------------------------|--|---|
| <code>language</code> | <code>String</code> | The original language of this <i>Program</i> . The CMS ingest places the <i>Program</i> object's Language EPG attribute in this field. |
| <code>locale</code> | <code>String</code> | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>modifiedDate</code> | <code>Date</code> | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | <code>Date</code> | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> <code>String</code> | The full name of this <i>Program</i> . The CMS ingest places the <i>Program</i> object's Short Title EPG attribute in this field. |
| <code>originID</code> | <code>Integer</code> | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>ownerID</code> | <code>String</code> | A code representing the owner (content provider) for this <i>Program</i> . The CMS ingest sets this field to the Owner ID of the <i>Program</i> . In deployments using NAGRA Media CMS, it must be no longer than 3 characters. |
| <code>privateMetadata</code> | <code>String</code> | This field is provided for operators to use for their own purposes. SDP passes it to the client unchanged. Its format (string, JSON object...) and meaning should be decided by the operator. |

| Name | Type | Description |
|--------------------------------|----------------------|---|
| | | The CMS ingest places the <i>Program</i> object's Private Metadata EPG attribute in this field. |
| <code>producer</code> | <code>String</code> | <p>The names of the producers of this <i>Program</i>, separated by semicolons.</p> <p>The CMS ingest places the <i>Program</i> object's Producers EPG attribute in this field.</p> |
| <code>promoImage</code> | <code>String</code> | <p>The URL of a promotional image for this <i>Program</i>.</p> <p>The CMS ingest places the <i>Program</i> object's Promoting picture URL EPG attribute in this field.</p> |
| <code>ratingID</code> | <code>Long</code> | <p>The <i>Rating</i> appropriate to this <i>Program</i>.</p> <p>The CMS ingest uses the <i>Program</i> object's Parental Rating, MPAA Rating and TV Rating EPG attributes to determine a suitable value for this field.</p> |
| <code>recordable</code> | <code>Boolean</code> | May this <i>Program</i> be recorded by users? |
| <code>serviceProviderID</code> | <code>Long</code> | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>shortTitle</code> | <code>String</code> | <p>A shorter name for this <i>Program</i>.</p> <p>The CMS ingest places the <i>Program</i> object's Short Title EPG attribute in this field.</p> |
| <code>studio</code> | <code>String</code> | <p>The name of the studio which produced this <i>Program</i>: the content owner.</p> <p>The CMS ingest places the <i>Program</i> object's Studio EPG attribute in this field.</p> |
| <code>subTitles</code> | <code>String</code> | <p>Does this <i>Program</i> include subtitles (in any language)?</p> <p>The CMS ingest sets this field to false.</p> |
| <code>synopsis</code> | <code>String</code> | <p>A longer description of this <i>Program</i>.</p> <p>The CMS ingest places the <i>Program</i> object's Short Synopsis EPG attribute in this field.</p> |

| Name | Type | Description |
|--------------------------|---------------------------------|---|
| <code>timeshifted</code> | Boolean | May end users use time-shift or trick-play features during this program? |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>year</code> | <i>Search criterion.</i> String | The year of production of this <i>Program</i> , or another notable date defined by the operator. The CMS ingest places the <i>Program</i> object's Production year EPG attribute in this field. |

67.2 tv.quative.service.program.model.ProgramAltAudioPID

| Name | Type | Description |
|---------------------------|---------|--|
| <code>audioType</code> | String | Audio Type: 'AUDIO' or 'SUBTITLES'. |
| <code>businessKey</code> | String | |
| <code>defaultPid</code> | Boolean | Default Pid. |
| <code>langID</code> | String | langID language id for the available languages. |
| <code>language</code> | String | String representation of the language. |
| <code>modifiedDate</code> | Date | Date modified. |
| <code>oldAudioType</code> | String | old Audio Type: 'AUDIO' or 'SUBTITLES'. It is used when updating |
| <code>oldLanguage</code> | String | String representation of the old language. It is used when updating. |
| <code>programUID</code> | Long | Unique id for the programme. |

67.3 Program

A *Program* represents a content item which can be broadcast.

Figure 22 Program and associated Entities.

It is the broadcast equivalent of an *Asset*, and shares many of the fields of the *Asset* Entity.

When a *Program* is broadcast on a *Channel* at a particular time, this forms an *Event*. The same *Program* may be broadcast on different *Channels* at the same or different times.

Programs do not have *Policy Groups*. Pay-per-view content is subscribed through specific *Events*.

Tip

In SDP, a *Program* represents a complete multiplexed transport stream. This is different from the definition used in MPEG-2, where a "program" forms a single element of a "transport" (identified by a Program Identifier, or PID.)

Caution!

In systems using a CMS, *Programs* are created automatically by the ingest process. Broadcast content and pricing should be defined in the CMS.

The CMS ingestion process creates *Programs* from the CMS *Content* objects.

67.3.1 Program

| Name | Type | Description |
|--------------|---|--|
| audioPID | List<ProgramAltAudioPID> | The UID of an <i>Audio PID</i> . Used when playing a program that contains multiple audio streams, to correctly decode the correct stream. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of the program. The CMS ingest places the <i>Programme</i> object's Description EPG attribute in this field. |
| duration | Integer | The duration of this <i>Program</i> , in milliseconds. The CMS ingest places the Duration of the <i>Programme</i> 's child <i>Period</i> in this field, or works it out from the Start Date and End Date . |

| Name | Type | Description |
|-----------------------------|---|---|
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>locale</code> | String | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>modifiedDateML</code> | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |
| <code>name</code> | <i>Search criterion.</i> Quick <i>Search criterion.</i> String | The title of the program. The CMS ingest places the <i>Programme</i> object's Title EPG attribute in this field. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | <i>Read only.</i> String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>ownerID</code> | String | A code representing the owner (content provider) for this <i>Program</i> . The CMS ingest does not set this field. |
| <code>ratingID</code> | Long | The <code>uid</code> of the <i>Rating</i> appropriate to this program. The CMS ingest uses the <i>Service</i> object's Rating EPG attribute to link the appropriate <i>Rating</i> . |
| <code>recordable</code> | Boolean | May this <i>Program</i> be recorded by users? The CMS ingest sets this field to reflect the <i>Programme</i> object's IsRecordable EPG attribute. |

| Name | Type | Description |
|--------------------------------|---------|---|
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>timeshifted</code> | Boolean | May end users use time-shift or trick-play features during this program? The CMS ingest sets this field to reflect the <i>Programme</i> object's <code>IsTimeshifted</code> EPG attribute. |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |

67.4

tv.quative.service.contentrating.model.primitive.AggregateRating

| Name | Type | Description |
|--------------------------------|------------|-------------------------------------|
| <code>contentID</code> | Long | |
| <code>numberOfRatings</code> | Long | |
| <code>serviceProviderID</code> | Long | ServiceProvider of aggregate rating |
| <code>sumOfRatings</code> | BigDecimal | |
| <code>uID</code> | Long | UID of aggregate rating |

67.5 programService.addMultiLanguage

Creates a new Program description for multilanguage.

This method calls internally the `createMultiLanguage(locale, pgm)`.

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>pgmuid</code> | Long | unique id of the package |
| <code>pname</code> | String | The name of the package to create |
| <code>pdescr</code> | String | short description |
| <code>psynopsis</code> | String | long description |
| <code>locale</code> | String | object that contains the id of the current language. |
| <code>pcopyright</code> | String | |
| <code>pshorttitle</code> | String | |
| <code>pepisode</code> | String | |

Return value

Type:

Integer the number of records inserted

67.5.1 programService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|--|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

67.5.2 programService.addProgramAltAudioPID

Inserts a new relationship between a language and the programme..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---|--|
| audiopid | tv.quative.service.program.model.ProgramAltAudioPid | ProgramAltAudioPID with the information of the programme, the current language and the id of the language available to listen. |

Return value

Type:

Integer with the num rows added

67.5.3 programService.create

Create (persist) a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|--|
| model | <code>LocaleBaseModel</code> tv.quative.core.persistence.model.LocaleBaseModel | The <code>LocaleBaseModel</code> to persist. |

Return value

Type:

UID of the created object.

67.5.4 programService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------------|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

67.5.5 programService.getAudioPIDsByPgmUID

Gets the VerboseProgram object information using the unique id as the key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|---------------------------------|
| puid | Long | Long unique id of the programme |

Return value

Type: List<ProgramAltAudioPID>

VerboseProgram object with the extended information retrieved in the dao layer.

Example

```
<getAudioPIDsByPgmUID>
  <puid>29</puid>
</getAudioPIDsByPgmUID>
```

67.5.6 programService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/programService/getByOrigin>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>97</originID>
  <locale>en_us</locale>
</getByOrigin>
```

67.5.7 programService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/programService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>uid</code> | Long | Integer unique id for the LocaleBaseModel object |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel if found, the matching object, null otherwise.

Example

```
<getByUID>
```

```
<uid>83</uid>
<locale>en_us</locale>
</getByUID>
```

67.5.8 programService.getByUIDs

Retrieve, from storage, the **Program** with the unique ids 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|-------------------------------|---|
| <code>uids</code> | <code>List<Long></code> | List of unique ids for the LocaleBaseModel object |
| <code>locale</code> | String | String reference for the language |

Return value

Type: `List<VerboseProgram>`

LocaleBaseModel if found, the matching object, null otherwise.

67.5.9 programService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|--------------------------------|
| <code>uid</code> | Long | Integer unique id of the Model |

Return value

Type: `List<M>`

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
<uid>69</uid>
</getMultiLanguageList>
```

67.5.10 programService.getPagedPgmAltAudioPIDsByPgmUID

Gets the VerboseProgram object information using the unique id as the key.

Parameters

| Name | Type | Description |
|--------------------------|------|---------------------------------|
| <code>puid</code> | Long | Long unique id of the programme |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<ProgramAltAudioPID>`

VerboseProgram object with the extended information retrieved in the dao layer.

67.5.11 programService.getPagedVerboseProgramsByOriginUID

Gets the list of VerboseProgram objects using the external system's id.

Parameters

| Name | Type | Description |
|--------------------------|---------|--|
| <code>originuid</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |
| <code>startRowNum</code> | Long | Long |
| <code>endRowNum</code> | Long | Long |

Return value

Type: `List<VerboseProgram>`

VerboseProgram object with the verbose information retrieved in the dao layer.

67.5.12 programService.getSimpleProgramByUID

Gets the SimpleProgram object information using the unique id as the key.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>puid</code> | Long | Long unique id of the programme |
| <code>locale</code> | String | String default language id to create the 1st multilanguage |

Return value

Type:

SimpleProgram object with the simple information retrieved in the dao layer.

Example

```
<getSimpleProgramByUID>
  <puid>4</puid>
  <locale>en_us</locale>
</getSimpleProgramByUID>
```

67.5.13 programService.getVerboseProgramByEventUID

Gets the VerboseProgram object information using the unique id as the key..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/programService/getVerboseProgramByEventUID>

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|--|
| <code>pEventUID</code> | Long | Long unique id of the event |
| <code>pLocale</code> | String | String default language id to create the 1st multilanguage |

Return value

Type:

VerboseProgram object with the extended information retrieved in the dao layer.

Example

```
<getVerboseProgramByEventUID>
  <pEventUID>45</pEventUID>
  <pLocale>stringValue</pLocale>
</getVerboseProgramByEventUID>
```

67.5.14 programService.getVerboseProgramByOriginUID

Gets the list of VerboseProgram objects using the external system's id.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|--|
| originuid | Integer | Integer unique origin id for customer representation |
| locale | String | String reference for the language |

Return value

Type: List<VerboseProgram>

VerboseProgram object with the verbose information retrieved in the dao layer.

Example

```
<getVerboseProgramByOriginUID>
  <originuid>51</originuid>
  <locale>en_us</locale>
</getVerboseProgramByOriginUID>
```

67.5.15 programService.getVerboseProgramByUID

Gets the VerboseProgram object information using the unique id as the key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|---------------------------------|
| puid | Long | Long unique id of the programme |

| Name | Type | Description |
|--------|--------|--|
| locale | String | String default language id to create the 1st multilanguage |

Return value

Type:

VerboseProgram object with the extended information retrieved in the dao layer.

Example

```
<getVerboseProgramByUID>
  <puid>76</puid>
  <locale>en_us</locale>
</getVerboseProgramByUID>
```

67.5.16 programService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

67.5.17 programService.modifyProgramAltAudioPID

Updates a relationship between a language and the programme..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|--|--|
| <code>audiopid</code> | <code>tv.quative.service.program.model.ProgramAltAudioPID</code> | ProgramAltAudioPID with the information of the programme, the current language and the id of the language available to listen. |

Return value

Type:

Integer with the num rows added

67.5.18 `programService.removeMultiLanguage`

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---|
| <code>model</code> | <code>LocaleBaseModel tv.quative.core.persistence.model.LocaleBaseModel</code> | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

67.5.19 `programService.removeProgramAltAudioPID`

Removes a new relationship between a language and the programme.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|-----------------------|
| <code>programUid</code> | Long | uid of the programme. |

| Name | Type | Description |
|-----------|--------|------------------------------|
| language | String | description for the language |
| audioType | String | Audio type |

Return value

Type:

Integer with the num rows added

Example

```
<removeProgramAltAudioPID>
<programUid>49</programUid>
<language>stringValue</language>
<audioType>stringValue</audioType>
</removeProgramAltAudioPID>
```

67.5.20 programService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------------|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

67.5.21 programService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

68 provisionService

This is an aggregate *Service* which provisions *Set Top Boxes* and *Smart Cards* for use with *Accounts*.

It enables post-provisioning by MAC address or by serial number, in the case of non-CAS environments, and by CAS information in card-based or cardless CAS environments.

The *Provision Service* communicates with the CAS to enable the new account.

68.1 provisionService

The implementation for the Provision Service. Provisioning associates a device with a subscriber account. This service supports two separate provisioning models: Pre-Provisioning and Post-Provisioning. Pre-Provisioning requires both Device and Account records to exist prior to the provisioning request. Post-Provisioning requires an Account record to exist and optionally a Device record. If the Device record does not exist on the system, a new record will be created.

Steve @author

Ashish Gupta @author

68.1.1 provisionService.pcClientRegistration

Registers a new user by creating it's account and user records..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|---------------------------------------|-------------------------|
| account | Account | Account of the new user |
| user | User ^{p.710} | User to create |

Return value

Type:

Long - UID of the created user

68.1.2 provisionService.provisionMediaPlayer

Create a new Marlin or PlayReady client application and bind it to the currently signed-on account.

The client must have successfully invoked `signonService#signonInitialByUserAndSpid(String, String, String)` previously.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/provisionService/provisionMediaPlayer`

Parameters

| Name | Type | Description |
|--------------------------------|--------|--|
| <code>mpId</code> | String | The media player ID. Identifies the client app (rather than the physical device which runs it). |
| <code>drmClientId</code> | String | All player apps on a physical client device share a DRM client. This is (in effect) a library or module which manages the interaction with the DRM server, on behalf of all conformant apps on that device. Each of these has a unique ID. |
| <code>drmSystemId</code> | String | An ID for the DRM server the DRM client is bound to. |
| <code>deviceProfileName</code> | String | The name of the device profile. The device profile defines the playback or control features the app supports. The app will only be offered content for which all features of the content's profile are in the device profile. (Content profile "requires"; device profile "supports".) |

Return value

Type: void

69 QUMActivity Service

This is a primitive *Service* which provides *Activities* for *Role* Entities.
The set of *Activities* is automatically derived from the *Services* in use.

69.1 QUM Activity

A *QUM Activity* represents an area of OAM which users can access.
QUM Users are granted permission to access *QUM Activities* through their *Roles*.
QUM Activities include

- ▶ workflows
- ▶ Qubelets

The available *QUM Activities* are determined by the deployed *Services*. *QUM Activities* cannot be edited.

69.1.1 QUMActivity

| Name | Type | Description |
|-----------------------------|--|--|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |
| <code>name</code> | <code>String</code> | Defines the name associated to the activity (in general the SDP model class name). |
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |
| <code>parentUID</code> | <code>Long</code> | Defines the parent activity uid (null if no parent). |

| Name | Type | Description |
|--------------------------------|--------|-----------------------------------|
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>type</code> | String | Defines the type of the activity. |
| <code>typeCode</code> | String | |
| <code>uID</code> | Long | The unique identifier. |

69.2 tv.quative.service.qum.activity.model.ActivityType

| Name | Type | Description |
|-----------------------|------|---------------------------------|
| <code>ACTIVITY</code> | | Code for ACTIVITY ActivityType. |
| <code>G</code> | | ActivityType for 'G' Code. |
| <code>GROUP</code> | | Code for GROUP ActivityType. |
| <code>Q</code> | | ActivityType for 'Q' Code. |
| <code>W</code> | | ActivityType for 'W' Code. |
| <code>WORKFLOW</code> | | Code for WORKFLOW ActivityType. |

69.3 qumActivityService

Handles the persistence of authorisation data encapsulated within the `tv.quative.service.qum.activity.model.QUMActivity` class by delegating to the dependency-injected Data Access Object.

Binod Singh @author

69.3.1 qumActivityService.create

Create (persist) a `BaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

69.3.2 `qumActivityService.create`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------|
| <code>model</code> | QUMActivity <small>p.756</small> | |

Return value

Type:

69.3.3 `qumActivityService.delete`

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

69.3.4 qumActivityService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|-------------|
| model | QUMActivity <small>p.756</small> | |

Return value

Type:

69.3.5 qumActivityService.getActivityListByRoleName

Retrieves the list of activities for a role by name..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|--------|----------------------|
| roleName | String | the name of the role |

Return value

Type: `List<QUMActivityPermission>`

List the list of activities associated to the role.

Example

```
<getActivityListByRoleName>
  <roleName>stringValue</roleName>
</getActivityListByRoleName>
```

69.3.6 qumActivityService.getActivityListByRoleUID

Retrieves the list of activities for a role by uid..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------|---------------------|
| roleUID | Long | the uid of the role |

Return value

Type: `List<QUMActivityPermission>`

List the list of activities associated to the role.

Example

```
<getActivityListByRoleUID>
  <roleUID>65</roleUID>
</getActivityListByRoleUID>
```

69.3.7 qumActivityService.getActivityTreeNodesByRoleName

Retrieves the list of activity tree nodes for a role by name..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|--------|----------------------|
| roleName | String | the name of the role |

Return value

Type: `List<QUMActivityTreeNode>`

List the list of activity tree nodes associated to the role

Example

```
<getActivityTreeNodesByRoleName>
  <roleName>stringValue</roleName>
</getActivityTreeNodesByRoleName>
```

69.3.8 qumActivityService.getByActivityName

Retrieves a `QUMActivity` object based upon activityname..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|---------------------------------|
| <code>activityName</code> | String | The activityname to search for. |

Return value

Type:

Matching `<code>QUMActivity</code>`

 or `<code>>null</code>` if not found.

Example

```
<getByActivityName>
  <activityName>stringValue</activityName>
</getByActivityName>
```

69.3.9 qumActivityService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>23</originID>
</getByOrigin>
```

69.3.10 qumActivityService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>59</uid>
</getByUID>
```

69.3.11 qumActivityService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows updated.

69.3.12 qumActivityService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|-------------|
| model | QUMActivity ^{p.756} | |

Return value

Type:

70 QUM User Service

This is a primitive *Service* which provides the *QUM User* Entity.
It also manages relations between *QUM Users* and *Roles*.

70.1 QUM User

A *QUM User* represents a user of a SDP system, as exposed by applications such as OAM.

Figure 23 QUMUser and associated Entities.

Every *QUM User* has a set of *Roles* which determine their access privileges.
Your organisation may decide how to distribute these privileges.

Note

Different users of SDP do not need to use distinct *QUM Users* if there is no business reason to do so. For example, in the reference SDP Services, all end users (as in the final consumers of digital media services) share a single, pre-defined *QUM User* with minimal privileges. In this system, individual end users are represented by the *User* entity type.

70.1.1 QUMUser

| Name | Type | Description |
|-----------------------------|--|---|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |

| Name | Type | Description |
|--------------------------------|---|---------------------------|
| <code>password</code> | <code>String</code> | The user's password. |
| <code>serviceProviderID</code> | <code>Long</code> | The Service provider UID. |
| <code>uID</code> | <code>Long</code> | The unique identifier. |
| <code>username</code> | <i>Search criterion.</i> <code>Quick</code> <i>Search criterion.</i> <code>String</code> | The login ID of the user. |

70.2 qumUserService

Handles the persistence of authorisation data encapsulated within the `tv.quative.service.qum.user.model.QUMUser` class by delegating to the dependency-injected Data Access Object. Although the out-of-the-box configuration uses a DAO which communicates with the Qube database, alternative DAOs may be substituted to persist roles in other systems.

James Rayner @author

70.2.1 qumUserService.addRole

Associates an existing user with an existing role..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|-------------------|--|
| <code>userId</code> | <code>Long</code> | The identifier of the user to associate with the role. Must be non-null. |
| <code>roleId</code> | <code>Long</code> | The identifier of the role to be associated. Must be non-null. |

Return value

Type: `void`

Example

```
<addRole>
<userId>4</userId>
<roleId>37</roleId>
```

</addRole>

70.2.2 qumUserService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

70.2.3 qumUserService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--|-------------|
| <code>user</code> | QUMUser ^{p.764} | |

Return value

Type:

70.2.4 qumUserService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

70.2.5 qumUserService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>73</originID>
</getByOrigin>
```

70.2.6 qumUserService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>27</uid>
</getByUID>
```

70.2.7 qumUserService.getQUMUsersForRole

Retrieves all **QUMUser(s)** objects with the appropriate role. Note that **QUMUser(s)** which have no matching role will be omitted from the result set..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------|--|
| roleUid | Long | The role uid to retrieve QUMUser objects. |

Return value

Type: **List<QUMUser>**

List of matching QUMUser(s).

Example

```
<getQUMUsersForRole>
  <roleUid>94</roleUid>
</getQUMUsersForRole>
```

70.2.8 qumUserService.getRoles

Retrieves all **Role** objects associated with the specified user..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--|--|
| <code>user</code> | QUMUser <small>p.764</small> | The user to retrieve the roles for (only <code>uid</code> attribute is used from this object). |
| <code>locale</code> | String | The locale to retrieve the roles in. Leave <code>null</code> to retrieve roles in the default locale. Note that roles will be omitted from the result set if no appropriate localisation is found. |

Return value

Type: `List<Role>`

A list of matching `<code>Role</code>` objects.

70.2.9 qumUserService.removeRole

Removes the association between a user and a role..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|--|--|
| <code>user</code> | QUMUser <small>p.764</small> | The user to be disassociated (only <code>uid</code> attribute is used from this object). |
| <code>role</code> | Role <small>p.812</small> | The role to be disassociated (only <code>uid</code> attribute is used from this object). |

Return value

Type:

Number of rows affects (0 if no association found or 1 if found and successfully removed).

70.2.10 qumUserService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows updated.

70.2.11 qumUserService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|------|--|-------------|
| user | QUMUser ^{p.764} | |

Return value

Type:

71 ratingService

This is a primitive *Service* which provides the *Rating* Entity.

71.1 Rating

A *Rating* represents the *User* age group for whom a *Program* or *Asset* is considered appropriate.

Figure 24 Rating and associated Entities.

All subscribable *Entities* can have a *Rating*. Client applications are expected not to allow a user to view content if the *Rating* of that content is higher than the *Rating* of the user. The *User* Entity also has a *Rating*, to allow rating control to be implemented when *Users* are managed on the server side.

Caution!

In systems using a CMS, *Ratings* are created automatically by the ingest process. Content should be defined in the CMS.

The CMS ingest creates *Ratings* as required to represent the ratings used by the *Content* ingested.

71.1.1 Rating

| Name | Type | Description |
|----------------|---|--|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| description | String | A short description of this <i>Entity</i> . |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| locale | String | When <i>Entities</i> are returned as model objects, the locale for which localizable fields of the <i>Entity</i> have been returned. |
| modifiedDate | Date | The date and time this <i>Entity</i> was last modified in the database. |
| modifiedDateML | Date | The date and time the multi-language fields of this <i>Entity</i> were last modified in the database. |

| Name | Type | Description |
|--------------------------|---|---|
| mpaaRating | String | The MPAA rating code for this <i>Rating</i> , where appropriate. |
| name | <i>Search criterion.</i> Quick Search criterion. String | The name of this <i>Entity</i> . |
| originID | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| originIdAndKey | <i>Read only.</i> String | A unique reference for the object, derived by joining the originID and originKey , together, with a pipe character to separate them. |
| originKey | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| precedenceValue | Integer | The value of the <i>Rating</i> . This is a number which can be used to compare <i>Ratings</i> numerically. The operator is responsible for choosing and enforcing a suitable comparison scheme, for example one based on the age of the viewer. |
| ratingCode | <i>Search criterion.</i> String | The operator's rating code for this <i>Rating</i> . Because this is a text field, comparison operators such as < (less than) and > (greater than) cannot be used. To be able to compare <i>Ratings</i> by a numeric value (such as by permitted age), use the precedenceValue field in addition to this one. |
| serviceProviderID | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| tvRating | String | The local TV authority's rating code for this <i>Rating</i> , where appropriate. |
| uID | Long | The SDP internal UID for this <i>Entity</i> . |

71.2 ratingService

Zafar Ali @author

1.0 @version

71.2.1 ratingService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|--|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

71.2.2 ratingService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------------|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

71.2.3 ratingService.delete

Delete, from storage, a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows deleted.

71.2.4 ratingService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>50</originID>
  <locale>en_us</locale>
</getByOrigin>
```

71.2.5 ratingService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/ratingService/getByUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>uid</code> | Long | The <code>uid</code> of the <i>Rating</i> . |
| <code>locale</code> | String | The locale to return localizable fields. |

Return value

Type:

`LocaleBaseModel` if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>3</uid>
  <locale>en_us</locale>
</getByUID>
```

71.2.6 ratingService.getDefaultRating

Gets the default rating for the specified locale..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>locale</code> | String | String language which we want to get List of ratings |

Return value

Type:

Rating default object for the locale.

Example

```
<getDefaultRating>
```

```
<locale>en_us</locale>
</getDefaultRating>
```

71.2.7 ratingService.getList

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/ratingService/getList`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|--|
| locale | String | String language which we want to get List of ratings |

Return value

Type: `List<Rating>`

List Rating objects for the locale.

Example

```
<getList>
<locale>en_us</locale>
</getList>
```

71.2.8 ratingService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: `List<M>`

List LocaleBaseModel with the specific locale dependent data.

Example


```
<getMultiLanguageList>
  <uid>93</uid>
</getMultiLanguageList>
```

71.2.9 ratingService.getRatingByRatingCode

Gets a rating by rating code..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/ratingService/getRatingByRatingCode`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--------|--------------|
| <code>ratingCode</code> | String | Rating code. |
| <code>locale</code> | String | Locale. |

Return value

Type:

Rating.

Example

```
<getRatingByRatingCode>
  <ratingCode>stringValue</ratingCode>
  <locale>en_us</locale>
</getRatingByRatingCode>
```

71.2.10 ratingService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

71.2.11 ratingService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

71.2.12 ratingService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

71.2.13 ratingService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

72 recommendationService

This *Service* provides the client interface to SDP's recommendation features.

SDP maintains a list of static **GENERAL** recommendations, provided by the CMS, and also can interface with an external recommendation engine to provide dynamic recommendations.

Clients use a single API, *recommendationService.getRecommendations*, to retrieve recommendations of all types.

SDP also provides APIs allowing clients to submit user ratings to the recommendation engine and to request content or content item groups to be excluded from dynamic recommendations.

Currently, SDP supports the ThinkAnalytics external recommendation engine.

72.1

tv.quative.service.recommendation.api.ExcludeContentRequest

| Name | Type | Description |
|-----------------------------------|--|--|
| <code>contentItemGroup</code> | <code>String</code> | origin key of the group to be excluded. |
| <code>contentOriginKey</code> | <code>String</code> | Origin key. |
| <code>contentOriginKeyType</code> | <code>String</code> | The type of content represented by the 'contentOriginKey'. values - 'COD_SCHEDULE' or 'BTV_EVENT'. |
| <code>contentSources</code> | <code>List<RecommendationEngineContentSource></code> | The type of recommendations to return. |
| <code>context</code> | <code>Context</code> | context in which the request is being made. |
| <code>endRow</code> | <code>Long</code> | end row. |
| <code>startRow</code> | <code>Long</code> | start row. |
| <code>uid</code> | <code>Long</code> | Alternative to providing an origin key (do not have to provide both, just one) we can provide the uid. |
| <code>userLevel</code> | <code>boolean</code> | Are user based recommendations filtered at user or account level. |

72.2

tv.quative.service.recommendation.api.RecommendationFilter

| Name | Type | Description |
|--------------------------------------|---------------------------------|--|
| <code>assetProfileUid List</code> | <code>List<Long></code> | List of Asset Profile UIDs to filter by. |
| <code>disableDevice Filtering</code> | <code>boolean</code> | Flag to disable device filtering. |
| <code>features</code> | <code>List<String></code> | list of features (actors, directors, etc) to be forwarded to recommendation engine |
| <code>genres</code> | <code>List<String></code> | list of genres. |
| <code>productProfileUid List</code> | <code>List<Long></code> | List of Product Profile UIDs to filter by. |

72.3

tv.quative.service.recommendation.api.RecommendationRequest

| Name | Type | Description |
|--|----------------------|---|
| <code>age</code> | <code>String</code> | age. |
| <code>allowAdultContent</code> | <code>Boolean</code> | Flag indicates whether Adult content should be included in the recommendations. Default to FALSE as precaution! |
| <code>allowPrevious Recommendations</code> | <code>Boolean</code> | Flag indicates if previous recommendations can be returned in the result list. |
| <code>contentOriginKey</code> | <code>String</code> | Origin key. |
| <code>contentOriginKey Type</code> | <code>String</code> | The type of content represented by the 'contentOriginKey'. values - 'COD_SCHEDULE' or 'BTV_EVENT'. |

| Name | Type | Description |
|--|--|---|
| <code>contentSources</code> | <code>List<RecommendationEngineContentSource></code> | The type of recommendations to return. |
| <code>context</code> | <code>Context</code> | context in which the request is being made. |
| <code>endRow</code> | <code>Long</code> | end row. |
| <code>filter</code> | <code>RecommendationFilter</code> | Contains attributes to filter recommendations by. |
| <code>maximumRecommendationToReturn</code> | <code>Long</code> | The maximum number of recommendations that should be returned. |
| <code>recommendedContentEndsBefore</code> | <code>Date</code> | The schedule date of recommendations returned should be on or before this date. |
| <code>recommendedContentStartsAfter</code> | <code>Date</code> | The schedule date of recommendations returned should be on or after this date. |
| <code>region</code> | <code>String</code> | Country or region (i.e. UK, Taipei) |
| <code>startRow</code> | <code>Long</code> | start row. |
| <code>type</code> | <code>String</code> | Type of recommendations required. One of - 'GENERAL', 'SOCIAL', 'CONTEXT', 'PREFERENCE' or 'RELATED_CONTENT' etc. |
| <code>uid</code> | <code>Long</code> | Alternative to providing an origin key (do not have to provide both, just one) we can provide the uid. |
| <code>userLevel</code> | <code>boolean</code> | Are user based recommendations filtered at user or account level. |

72.4

tv.quative.service.recommendation.api.ContentRatingRequest

| Name | Type | Description |
|----------------------|---|--|
| contentOriginKey | String | Origin key. |
| contentOriginKeyType | String | The type of content represented by the 'contentOriginKey'. values - 'COD_SCHEDULE' or 'BTV_EVENT'. |
| contentSources | List<RecommendationEngineContentSource> | The type of recommendations to return. |
| context | Context | context in which the request is being made. |
| endRow | Long | end row. |
| rating | Integer | A numeric value between 1 and 10. |
| startRow | Long | start row. |
| uid | Long | Alternative to providing an origin key (do not have to provide both, just one) we can provide the uid. |
| userLevel | boolean | Are user based recommendations filtered at user or account level. |

72.5

tv.quative.service.recengintegration.enums.RecommendationEngineUs

| Name | Type | Description |
|------------|------|--------------------------|
| DOWNLOADED | | User downloaded content. |
| PURCHASED | | User purchased content. |
| RECORDED | | User recorded content |
| WATCHED | | User watched content. |

72.6

tv.quative.service.recommendation.api.ReportActionRequest

| Name | Type | Description |
|--|---|--|
| <code>actionType</code> | <code>RecommendationEngine UserAction</code> | The action performed by the user that the client wants to report. |
| <code>contentOriginKey</code> | <code>String</code> | Origin key. |
| <code>contentOriginKey Type</code> | <code>String</code> | The type of content represented by the 'contentOriginKey'. values - 'COD_SCHEDULE' or 'BTV_EVENT'. |
| <code>contentSources</code> | <code>List<Recommendation EngineContentSource></code> | The type of recommendations to return. |
| <code>context</code> | <code>Context</code> | context in which the request is being made. |
| <code>endRow</code> | <code>Long</code> | end row. |
| <code>startRow</code> | <code>Long</code> | start row. |
| <code>uid</code> | <code>Long</code> | Alternative to providing an origin key (do not have to provide both, just one) we can provide the uid. |
| <code>userLevel</code> | <code>boolean</code> | Are user based recommendations filtered at user or account level. |

72.7 recommendationService

The recommendation service allows clients (STB, Media Players, etc) to retrieve and manage recommendations.

Recommendations can be general 'content-based' or personalised 'user-based'. General 'content-based' recommendations are content that are somehow related (Star wars Episode II may be linked to Star Wars Episode II and therefore recommended). User based recommendations on the other hand are made based on user preference, history etc. There are various types of user based recommendations denoted by the 'type' attribute of the request. The recommendation 'types' supported depends on the underlying RecommendationEngine. The service also allows clients to manage recommendations by exposing methods to exclude content from future recommendations, clear user cache, provide a rating for content etc.

RRajendram @author

72.7.1 recommendationService.clearCache

Clear the user's personalised content recommendation cache.

If userLevel flag is true and user id is available in the context, cache will be cleared at a user level. Else cache is cleared at an account level.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/recommendationService/clearCache`

Parameters

| Name | Type | Description |
|---------|-------------------|---------------------------------|
| request | ClearCacheRequest | the request to clear the cache. |

Return value

Type: void

72.7.2 recommendationService.excludeRecommendation

Exclude one or more recommendations with origin key matching the origin key in the exclusion request from future recommendations.

Note: content origin key refers to either Event origin-key or Schedule origin-key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/recommendationService/excludeRecommendation`

Parameters

| Name | Type | Description |
|---------------|---|---|
| exclusionList | tv.quative.service.recommendation.api.ExcludeContentRequest | the list of recommendations to exclude. |

Return value

Type: void

72.7.3 recommendationService.getRawRecommendations

Gets a list of general 'content-based' or personalised 'user-based' recommendations.

General content-based recommendations are content somehow related to the content origin key. Note: content origin key refers to either Event origin-key or Schedule origin-key. User-based recommendations are based on account or user and the 'type' of user-based recommendation being requested. Please note, that unlike 'getRecommendations', this simply returns raw results from the recommendation engine and doesn't filter based on scheduling or profile applicability.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/recommendationService/getRawRecommendations`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--|---|
| <code>request</code> | <code>tv.quative.service.recommendation.api.RecommendationRequest</code> | the recommendation request ^{D.781} |

Return value

Type: `List<RawRecommendation>`

list of raw (unprocessed) recommendations for the content passed in

72.7.4 recommendationService.getRecommendations

Get a list of general 'content-based' or personalised 'user-based' recommendations.

General content-based recommendations are content somehow related to the content origin key. Note: content origin key refers to either Event origin-key or Schedule origin-key. User-based recommendations are based on account or user and the 'type' of user-based recommendation being requested. The type attribute of the request must specify the type of recommendations being requested, and should be one of - 'GENERAL'(content-based) or one of the configured user-based types.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/recommendationService/getRecommendations`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--|---|
| <code>request</code> | <code>tv.quative.service.recommendation.api.RecommendationRequest</code> | the recommendation request ^{D.781} |

Return value

Type: `List<Recommendation>`

list of recommendations for the content passed in.

72.7.5 recommendationService.rate

Provide a rating for one or more Assets (COD) or Programs (BTV).

Note: content origin key refers to either Asset origin-key or Program origin-key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/recommendationService/rate`

Parameters

| Name | Type | Description |
|----------------------|---|--|
| <code>ratings</code> | <code>tv.quative.service.recommendation.api.ConsentRatingRequest</code> | a list of ratings for corresponding content. |

Return value

Type: void

72.7.6 recommendationService.reportAction

Report an action performed by the user.

Note: content origin key refers to either Event origin-key or Schedule origin-key.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/recommendationService/reportAction`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|--|--|
| <code>userAction</code> | <code>tv.quative.service.recommendation.api.ReportActionRequest</code> | the request object used to report user action. |

Return value

Type: void

73 Report Engine Service

This *Service* processes a report and delivers it to its destination.

SDP automatically triggers this *Service* when the current time matches an active schedule for a *Report*.

73.1 reportEngineService

Report Engine Service Interface. Interface which declares the compulsory methods that must be implemented in all the ReportEngineServiceImpl classes which implements this interface. ReportEngineServiceImpl implemented objects are used as WebServices to communicate with the Reporting system The services are made visible to Hue or to Web Services using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="asset Service", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) Since it is an aggregate service, all of its methods are exposed to everyone. The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue. IMPORTANT NOTE: We do not need Reporting framework at this moment at the ws-gateway, so we will remove ws and hue annotations and use the default value which is false. That implies that we do not have to set up quartz in ws-gateway container.

Ashish Gupta @author

Miguel Gil @author

1.0 @version

73.1.1 reportEngineService.cancelRunningReport

A Service method to cancel a running/processing report.

Contains the logic required to cancel the running/processing report.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|------------------|
| runLogUID | Long | Long run log uid |

Return value

Type:

Boolean tells whether the report has been cancelled or not

Example

```
<cancelRunningReport>
  <runLogUID>60</runLogUID>
</cancelRunningReport>
```

73.1.2 reportEngineService.processAdHocReport

A Service method to process the ad hoc report (without scheduling).

Contains the logic required to process the ad hoc report (without scheduling). It is a static method which will act as a parent to create a report processing thread.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-----------------|
| reportUID | Long | Long report uid |
| locale | String | String locale |

Return value

Type:

Boolean tells whether the report has been processed or not

Example

```
<processAdHocReport>  
  <reportUID>27</reportUID>  
  <locale>en_us</locale>  
</processAdHocReport>
```

74 Report Scheduler Service

This *Service* is responsible for scheduling *Reports*.

Note

This is an aggregate *Service* which depends on the *Report Service*. Its methods `getReportService` and `setReportService` are used by the dependency injection process and are not intended for use by clients.

74.1 reportSchedulerService

Report Scheduler Service Interface. Interface which declares the compulsory methods that must be implemented in all the `ReportSchedulerServiceImpl` classes which implements this interface. `ReportSchedulerServiceImpl` implemented objects are used as `WebServices` to communicate with the Reporting system. The services are made visible to Hue or to `WebServices` using annotation in the Service Implementation class. Ex: For the ServiceClass: `@QubeService(name="assetService", hue=true, ws=true)` For the Methods: `@QubeMethod(hue=false, ws=true)` Since it is an aggregate service, all of its methods are exposed to everyone. The getters methods used to retrieve information using different datamodel could be accessible both the `WebServices` and the Hue. IMPORTANT NOTE: We do not need Reporting framework at this moment at the ws-gateway, so we will remove ws and hue annotations and use the default value which is false. That implies that we do not have to set up quartz in ws-gateway container.

Ashish Gupta @author

Miguel Gil - Integrated with Spring Quartz Scheduler @author

1.0 @version

74.1.1 reportSchedulerService.scheduleAllActiveReports

A Service method to schedule all active reports.

Contains the logic required to remove the report run log.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------|--------|---------------|
| locale | String | String locale |

Return value

Type:

Integer number of reports scheduled

Example

```
<scheduleAllActiveReports>
  <locale>en_us</locale>
</scheduleAllActiveReports>
```

74.1.2 reportSchedulerService.scheduleAllActiveSchedulesOfReport

A Service method to schedule all active reports.

Contains the logic required to remove the report run log.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-------------------------------|
| runSpUID | Long | Long run service provider uid |
| reportUID | Long | Long report uid |
| locale | String | String locale |

Return value

Type:

Integer number of report schedules scheduled

Example

```
<scheduleAllActiveSchedulesOfReport>
  <runSpUID>9</runSpUID>
  <reportUID>1</reportUID>
  <locale>en_us</locale>
</scheduleAllActiveSchedulesOfReport>
```

74.1.3 reportSchedulerService.scheduleReportSchedule

A Service method to schedule a report schedule.

Contains the logic required to schedule a report schedule.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|------|-------------------------------|
| runSpUID | Long | Long run service provider uid |

| Name | Type | Description |
|----------------|--------|------------------------|
| reportUID | Long | Long report uid |
| cronExpression | String | String cron expression |
| locale | String | String locale |

Return value

Type:

Boolean scheduling status

Example

```
<scheduleReportSchedule>
  <runSpUID>21</runSpUID>
  <reportUID>87</reportUID>
  <cronExpression>stringValue</cronExpression>
  <locale>en_us</locale>
</scheduleReportSchedule>
```

74.1.4 reportSchedulerService.unscheduleReport

A Service method to unschedule all the schedules of the report.

Contains the logic required to schedule a report schedule.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-----------------|
| reportUID | Long | Long report uid |
| locale | String | String locale |

Return value

Type:

Boolean scheduling status

Example

```
<unscheduleReport>
  <reportUID>18</reportUID>
```



```
<locale>en_us</locale>
</unscheduleReport>
```

74.1.5 reportSchedulerService.unscheduleReportSchedule

A Service method to unschedule a report schedule.

Contains the logic required to schedule a report schedule.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|--------|------------------------|
| reportUID | Long | Long report uid |
| cronExpression | String | String cron expression |
| locale | String | String locale |

Return value

Type:

Boolean scheduling status

Example

```
<unscheduleReportSchedule>
<reportUID>50</reportUID>
<cronExpression>stringValue</cronExpression>
<locale>en_us</locale>
</unscheduleReportSchedule>
```

75 Report Service

This is a primitive *Service* which provides the *Report* Entity.
It also manages the schedules and destinations for a created *Report*.

75.1 Report

A *Report* Entity is an instruction to SDP to generate a series of reports to a schedule.

While a *Report* exists and is fully enabled, SDP will continue generating reports and sending them to the report's configured destination.

The *Report* is associated with subsidiary *Entities* which contain information about

- ▶ what kind of report to produce (the *Report Type*)
- ▶ what data to use to generate the report (*Report Criteria*)
- ▶ what schedule to follow (*Report Schedule*) and
- ▶ where to send it (*Report Destination*.)

Every *Report* has its own *Report Type*, *Report Criteria*, *Report Schedules* and *Report Destinations*. All these *Entities* should be created together to form a single scheduled report.

75.1.1 Report

| Name | Type | Description |
|---------------------------|--|--|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>criteria</code> | <code>String</code> | Defines the criteria associated to the report. |
| <code>description</code> | <code>String</code> | Defines the description associated to the report. |
| <code>enabled</code> | <i>Read only.</i> <code>Boolean</code> | Defines whether the report is enabled or disabled. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>localeBased</code> | <i>Read only.</i> <code>Boolean</code> | Defines whether the report is locale based or not. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |

| Name | Type | Description |
|--------------------------------|--|---|
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> String | Defines the name associated to the report. |
| <code>originID</code> | Integer | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> String | |
| <code>originKey</code> | String | The key of the model object in the origin system (3rd part system). |
| <code>serviceName</code> | String | Defines the service name associated to the report. |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>uID</code> | Long | The unique identifier. |
| <code>useErrorDir</code> | <i>Read only.</i> Boolean | Defines whether the report has to use error directory. |

75.2 tv.quative.service.report.model.Protocol

| Name | Type | Description |
|-----------------------------|------|--------------------------|
| <code>FTP</code> | | FTP Protocol. |
| <code>LOCAL_TRANSFER</code> | | Local Transfer Protocol. |
| <code>SFTP</code> | | |

75.3 tv.quative.service.report.model.ReportDestination

| Name | Type | Description |
|---------------------------|----------------|--|
| <code>binaryOutput</code> | Boolean | Defines the transfer mode for the generated output . |

| Name | Type | Description |
|--------------------------------|---------|---|
| <code>bodyTemplate</code> | String | Defines the body template name associated to the report destination. |
| <code>compressed</code> | Boolean | Defines whether the report destination is compressed or uncompressed. |
| <code>compressionScheme</code> | String | |
| <code>destUID</code> | Long | Defines the destination uid to which the report destination is associated to. |
| <code>directory</code> | String | Defines the directory associated to the report destination. |
| <code>email</code> | String | Defines the email associated to the report destination. |
| <code>enabled</code> | Boolean | Defines whether the report destination is enabled or disabled. |
| <code>filenameFormat</code> | String | Defines the format of the filename associated to the report destination. |
| <code>footerTemplate</code> | String | Defines the footer template name associated to the report destination. |
| <code>formatter</code> | String | Defines the formatter class name associated to the report destination. |
| <code>headerTemplate</code> | String | Defines the header template name associated to the report destination. |
| <code>hostname</code> | String | Defines the hostname associated to the report destination. |
| <code>modifiedDate</code> | Date | Defines the last modified date for the report destination. |
| <code>pMaxPSeqNo</code> | Integer | |
| <code>pSeqNo</code> | Integer | |
| <code>padPSeqNo</code> | Boolean | |

| Name | Type | Description |
|-----------------------------|----------------------|--|
| <code>password</code> | <code>String</code> | Defines the password associated to the report destination. |
| <code>port</code> | <code>Integer</code> | Defines the port associated to the report destination. |
| <code>protocol</code> | <code>String</code> | Defines the protocol associated to the report destination. |
| <code>recordsPerFile</code> | <code>Integer</code> | Defines the records per file associated to the report destination. |
| <code>reportUID</code> | <code>Long</code> | Defines the report uid to which the report destination is associated to. |
| <code>streamed</code> | <code>Boolean</code> | Defines whether the report destination is to be byte streamed. |
| <code>suppressZipExt</code> | <code>Boolean</code> | |
| <code>username</code> | <code>String</code> | Defines the username associated to the report destination. |

75.4

tv.quative.service.report.model.ReportDestination.CompressionScheme

| Name | Type | Description |
|-------------------|------|-------------|
| <code>GZIP</code> | | |
| <code>ZIP</code> | | |

75.5 tv.quative.service.report.model.ReportRunLog

| Name | Type | Description |
|----------------------|-------------------|--|
| <code>endTime</code> | <code>Date</code> | Defines the end time associated to the report run log. |

| Name | Type | Description |
|---------------------------|--------|---|
| <code>locale</code> | String | Defines the locale associated to the report run log. |
| <code>modifiedDate</code> | Date | Defines the last modified date for the report run log. |
| <code>reportName</code> | String | Defines the report name associated to the report run log. |
| <code>reportUID</code> | Long | Defines the report uid to which the run log is associated to. |
| <code>runLogUID</code> | Long | Defines the run log uid to which the report run log is associated to. |
| <code>startTime</code> | Date | Defines the cron start time associated to the report run log. |
| <code>status</code> | String | Defines the run status associated to the report run log. |
| <code>statusCode</code> | String | |

75.6 tv.quative.service.report.model.Status

| Name | Type | Description |
|----------------------------|------|--------------------------------|
| <code>C</code> | | Status for 'C' Code. |
| <code>CANCELLED</code> | | Code for CANCELLED Status. |
| <code>F</code> | | Status for 'F' Code. |
| <code>FAILED</code> | | Code for FAILED Status. |
| <code>FORCED_FAILED</code> | | Code for FORCED_FAILED Status. |
| <code>P</code> | | Status for 'P' Code. |

| Name | Type | Description |
|-----------------------------|------|---------------------------------------|
| PARTIALLY_SUCCESSFUL | | Code for PARTIALLY_SUCCESSFUL Status. |
| R | | Status for 'R' Code. |
| RUNNING | | Code for RUNNING Status. |
| S | | Status for 'S' Code. |
| SUCCESSFUL | | Code for SUCCESSFUL Status. |
| X | | Status for 'X' Code. |

75.7 tv.quative.service.report.model.ReportSchedule

| Name | Type | Description |
|-----------------------|----------------|---|
| cronExpression | String | Defines the cron expression associated to the report schedule. |
| enabled | Boolean | Defines whether the report schedule is enabled or disabled. |
| locale | String | Defines the locale associated to the report schedule. |
| modifiedDate | Date | Defines the last modified date for the report schedule. |
| reportUID | Long | Defines the report uid to which the report schedule is associated to. |
| schdUID | Long | Defines the schedule uid to which the report schedule is associated to. |

75.8 reportService

Report Service Interface. Interface which declares the compulsory methods that must be implemented in all the Report Service classes which implements this interface. ReportService implemented objects are used as WebServices to communicate with the Reporting system. The services are made visible to Hue or to WebServices using annotation in the Service Implementation class. Ex: For the ServiceClass: @QubeService(name="assetService", hue=true, ws=true) For the Methods: @QubeMethod(hue=false, ws=true) The methods used to create, delete or update items of the Reporting System, only could be accessible to WebServices (used by 3rd part Systems). The getters methods used to retrieve information using different datamodel could be accessible both the WebServices and the Hue. IMPORTANT NOTE: We do not need Reporting framework at this moment at the ws-gateway, so we will remove ws and hue annotations and use the default value which is false. That implies that we do not have to set up quartz in ws-gateway container. Regarding this particular service and because it is extending AbstractCRUDService, we should keep annotations here.

Ashish Gupta @author

Miguel Gil @author

1.0 @version

75.8.1 reportService.addReportDestination

A Service method to add a report destination to the report.

Contains the logic required to add the report destination.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|---------------------------|
| reportDestination | tv.quative.service.report.model.ReportDestination | Report destination object |

Return value

Type:

Long the uid of the created report destination

75.8.2 reportService.addReportRunLog

A Service method to add a report run log to the report.

Contains the logic required to add the report run log.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|---|-------------|
| reportRunLog | tv.quative.service.report.model.ReportRunLog ^{p.797} | |

Return value

Type:

Long the uid of the created report run log

75.8.3 reportService.addReportSchedule

A Service method to add a report schedule to the report.

Contains the logic required to add the report schedule.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|--|---------------------------------------|
| reportSchedule | tv.quative.service.report.model.ReportSchedule | ReportSchedule report schedule object |

Return value

Type:

Long the uid of the created report schedule

75.8.4 reportService.create

Create (persist) a BaseModel..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

75.8.5 reportService.delete

Delete, from storage, a BaseModel..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

75.8.6 reportService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>72</originID>
</getByOrigin>
```

75.8.7 reportService.getByOriginID

A Service method to get the report details list for origin uid.

Contains the logic required to get report details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------|---------|-------------|
| originID | Integer | Integer |

Return value

Type: List<Report>

List of Report object

Example

```
<getByOriginID>
  <originID>14</originID>
</getByOriginID>
```

75.8.8 reportService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>20</uid>
</getByUID>
```

75.8.9 reportService.getLastReportRunLogByStatus

A Service method to get the last report run log details object for the report uid and status.

Contains the logic required to get last report run log details object by status.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|--------|-----------------|
| reportUID | Long | Long report uid |
| status | String | String status |

Return value

Type:

List of ReportRunLog object

Example

```
<getLastReportRunLogByStatus>
  <reportUID>28</reportUID>
  <status>stringValue</status>
</getLastReportRunLogByStatus>
```

75.8.10 reportService.getLatestReportRunLog

A Service method to get the latest (running or successful) report run log details object for the report uid.

Contains the logic required to get latest (running or successful) report run log details object.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-----------------|
| reportUID | Long | Long report uid |

Return value

Type:

List of ReportRunLog object

Example

```
<getLatestReportRunLog>
  <reportUID>33</reportUID>
</getLatestReportRunLog>
```

75.8.11 reportService.getList

A Service method to get the report details list for user uid.

Contains the logic required to get report details list.

Available through Web Services.

Parameters

None.

Return value

Type: `List<Report>`

List of Report object

Example

```
<getList/>
```

75.8.12 reportService.getReportDestination

A Service method to get the report destination details object for the report destination uid.

Contains the logic required to get report destination details object.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------|-----------------------------|
| destUID | Long | Long report destination uid |

Return value

Type:

ReportDestination report destination object

Example

```
<getReportDestination>
  <destUID>68</destUID>
</getReportDestination>
```

75.8.13 reportService.getReportDestinationList

A Service method to get the report destination details list for the report uid.

Contains the logic required to get report destination details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-----------------|
| reportUID | Long | Long report uid |

Return value

Type: `List<ReportDestination>`

List of ReportDestination object

Example

```
<getReportDestinationList>
  <reportUID>97</reportUID>
</getReportDestinationList>
```

75.8.14 reportService.getReportRunLog

A Service method to get the report run log details object for the report run log uid.

Contains the logic required to get report run log details object.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-------------------------|
| runLogUID | Long | Long report run log uid |

Return value

Type:

ReportRunLog report run log object

Example

```
<getReportRunLog>
  <runLogUID>56</runLogUID>
</getReportRunLog>
```

75.8.15 reportService.getReportRunLogList

A Service method to get the report run log details list for the report uid.

Contains the logic required to get report run log details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-----------------|
| reportUID | Long | Long report uid |

Return value

Type: List<ReportRunLog>

List of ReportRunLog object

Example

```
<getReportRunLogList>
  <reportUID>64</reportUID>
</getReportRunLogList>
```

75.8.16 reportService.getReportSchedule

A Service method to get the report schedule details object for the report schedule uid.

Contains the logic required to get report schedules details object.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------|--------------------------|
| schdUID | Long | Long report schedule uid |

Return value

Type:

ReportSchedule report schedule object

Example

```
<getReportSchedule>
  <schdUID>3</schdUID>
</getReportSchedule>
```

75.8.17 reportService.getReportScheduleList

A Service method to get the report schedule details list for the report uid.

Contains the logic required to get report schedules details list.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-----------------|
| reportUID | Long | Long report uid |

Return value

Type: `List<ReportSchedule>`

List of ReportSchedule object

Example

```
<getReportScheduleList>
  <reportUID>70</reportUID>
</getReportScheduleList>
```

75.8.18 reportService.modifyReportDestination

A Service method to modify a report destination to the report.

Contains the logic required to modify the report destination.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|---------------------------|
| reportDestination | tv.quative.service.report.model.ReportDestination | Report destination object |

Return value

Type:

Integer number of records modified

75.8.19 reportService.modifyReportRunLog

A Service method to modify a report run log to the report.

Contains the logic required to modify the report run log.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--|------------------------------------|
| <code>reportRunLog</code> | tv.quative.service.report.model.ReportRunLog | ReportRunLog report run log object |

Return value

Type:

Integer number of records modified

75.8.20 reportService.modifyReportSchedule

A Service method to modify a report schedule to the report.

Contains the logic required to modify the report schedule.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--|---------------------------------------|
| <code>reportSchedule</code> | tv.quative.service.report.model.ReportSchedule | ReportSchedule report schedule object |

Return value

Type:

Integer number of records modified

75.8.21 reportService.removeReportDestination

A Service method to remove a report destination to the report.

Contains the logic required to remove the report destination.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|---|---------------------------|
| <code>reportDestination</code> | tv.quative.service.report.model.ReportDestination | Report destination object |

Return value

Type:

Integer number of records removed

75.8.22 reportService.removeReportRunLog

A Service method to remove a report run log to the report.

Contains the logic required to remove the report run log.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--|-----------------------|
| <code>reportRunLog</code> | tv.quative.service.report.model.ReportRunLog | Report run log object |

Return value

Type:

Integer number of records removed

75.8.23 reportService.removeReportSchedule

A Service method to remove a report schedule to the report.

Contains the logic required to remove the report schedule.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--|------------------------|
| <code>reportSchedule</code> | tv.quative.service.report.model.ReportSchedule | Report schedule object |

Return value

Type:

Integer number of records removed

75.8.24 reportService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

76 Role Service

This is a primitive *Service* which provides the *Role* Entity.
It also manages the relationships between *Roles* and *Activities*.

76.1 Role

A *Role* represents a set of *Activities* and *Services* which a *QUM User* may access.

Figure 25 Role and associated Entities.

Roles in OAM

Permissions can be granted for each *Activity* at three levels:

| Value | Description |
|-------|--|
| None | <i>Users</i> with this <i>Role</i> are not permitted to use this <i>Activity</i> . When these <i>Users</i> log in to OAM, these <i>Activities</i> are not included in menus. |
| Read | <i>Users</i> with this <i>Role</i> are permitted to use this <i>Activity</i> for viewing purposes. When these <i>Users</i> log in to OAM, the <i>Activities</i> are available, but any controls which would alter data held in SDP are disabled. |
| Write | <i>Users</i> with this <i>Role</i> have full access to this <i>Activity</i> and can perform any action which it provides. |

Users with a particular *Role* are permitted to carry out all the associated *Activities* at the level defined by the *Role*.

Roles for Services

Roles also determine whether a user is permitted to make calls to a *Service* method.
These *Roles* are defined in the *Service* code.

76.1.1 Role

| Name | Type | Description |
|--------------------------------|--|---|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>description</code> | <code>String</code> | The short description of the model element. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>locale</code> | <code>String</code> | The locale description of the language. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |
| <code>modifiedDateML</code> | <code>Date</code> | The date last modified and persisted. |
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> <code>String</code> | The name of the model element. |
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |
| <code>serviceProviderID</code> | <code>Long</code> | The Service provider UID. |
| <code>uID</code> | <code>Long</code> | The unique identifier. |

76.2 roleService

Handles the persistence of authorisation data encapsulated within the `tv.quative.service.qum.role.model.Role` class by delegating to the dependency-injected Data Access Object. Although the out-of-the-box configuration uses a DAO which communicates with the Qube database, alternative DAOs may be substituted to persist roles in other systems.

James Rayner @author

76.2.1 roleService.addActivity

Add an activity to the role..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---------------------------------|
| roleUID | Long | the uid of the account |
| activityUID | Long | the uid of the role |
| permission | String | the permission for the activity |

Return value

Type:

Integer number of activities added to the role.

Example

```
<addActivity>
  <roleUID>51</roleUID>
  <activityUID>84</activityUID>
  <permission>stringValue</permission>
</addActivity>
```

76.2.2 roleService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|--|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

76.2.3 roleService.addMultiLanguage

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---------------------------------------|-------------|
| model | Role ^{p.812} | |

Return value

Type:

76.2.4 roleService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|---------------------------------|
| model | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to persist. |

Return value

Type:

UID of the created object.

76.2.5 roleService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---------------------------------------|-------------|
| model | Role ^{p.812} | |

Return value

Type:

76.2.6 roleService.delete

Delete, from storage, a `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|--|--|
| model | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows deleted.

76.2.7 roleService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---------------------------------------|-------------|
| model | Role ^{p.812} | |

Return value

Type:

76.2.8 roleService.getAllRoles

Retrieves all `Role` objects in the appropriate locale. Note that `Roles` which have no matching localisation will be omitted from the result set..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>locale</code> | String | The locale to retrieve <code>Role</code> objects in. |

Return value

Type: `List<Role>`

List of matching roles.

Example

```
<getAllRoles>
  <locale>en_us</locale>
</getAllRoles>
```

76.2.9 `roleService.getByOrigin`

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

`LocaleBaseModel` for this external origin uid and origin key.

Example

```
<getByOrigin>
```

```
<originKey>stringValue</originKey>
<originID>75</originID>
<locale>en_us</locale>
</getByOrigin>
```

76.2.10 roleService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|--------|---|
| <code>uid</code> | Long | Integer unique id for the <code>LocaleBaseModel</code> object |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

`LocaleBaseModel` if found, the matching object, null otherwise.

Example

```
<getByUID>
<uid>23</uid>
<locale>en_us</locale>
</getByUID>
```

76.2.11 roleService.getMultiLanguageList

Retrieves all `Role` objects with the specified uid from different locales..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|--|
| <code>uid</code> | Long | The role uid for which to retrieve <code>Role</code> objects in. |

Return value

Type: `List<M>`

List of matching roles.

Example

```
<getMultiLanguageList>
  <uid>46</uid>
</getMultiLanguageList>
```

76.2.12 roleService.getRoleListByAccountUID

Retrieves the list of roles for an account..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|--------|--------------------------|
| accountUID | Long | the uid of the account |
| locale | String | the locale for the roles |

Return value

Type: `List<Role>`

List<Role> the list of roles associated to the account.

Example

```
<getRoleListByAccountUID>
  <accountUID>90</accountUID>
  <locale>en_us</locale>
</getRoleListByAccountUID>
```

76.2.13 roleService.modifyActivity

Modify an activity to the role..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|---------------------------------|
| roleUID | Long | the uid of the account |
| activityUID | Long | the uid of the role |
| permission | String | the permission for the activity |

Return value

Type:

Integer number of activities modified.

Example

```
<modifyActivity>
  <roleUID>14</roleUID>
  <activityUID>87</activityUID>
  <permission>stringValue</permission>
</modifyActivity>
```

76.2.14 roleService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

76.2.15 roleService.modifyMultiLanguage

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---------------------------------------|-------------|
| model | Role ^{p.812} | |

Return value

Type:

76.2.16 roleService.removeActivity

Remove an activity from the role..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|------|------------------------|
| roleUID | Long | the uid of the account |
| activityUID | Long | the uid of the role |

Return value

Type:

Integer number of activities removed from the role.

Example

```
<removeActivity>
  <roleUID>21</roleUID>
  <activityUID>82</activityUID>
</removeActivity>
```

76.2.17 roleService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

76.2.18 roleService.removeMultiLanguage

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---------------------------------------|-------------|
| model | Role ^{p.812} | |

Return value

Type:

76.2.19 roleService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------------|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

76.2.20 roleService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---------------------------------------|-------------|
| model | Role ^{p.812} | |

Return value

Type:

76.2.21 roleService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------------|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

77 ServiceProvider Service

This is a primitive *Service* which provides the *Service Provider* Entity.

77.1 Service Provider

A *Service Provider* represents a media service operator using the SDP system.

All *Entities* have a single *Service Provider*, which represents that this service operator owns that *Entity*.

There is always at least one *Service Provider* in a SDP deployment, termed the *Global Service Provider*. The Global Service Provider can sublet services on their installation of SDP to other organisations by creating *Service Providers* for their clients.

Subsidiary service operators are allowed to create *Entities* as they like. When they do this, the *Entities* are automatically associated with their *Service Provider*. This allows them to modify and delete the *Entities* they have created. Subsidiary service operators do not have access to *Entities* belonging to any other subsidiary service operator.

All service operators have read access to all the *Entities* owned by the *Global Service Provider*, but subsidiary service operators may not edit or delete them.

77.1.1 ServiceProvider

| Name | Type | Description |
|---|--|--|
| <code>casInstanceId</code> | <code>String</code> | |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>creditLimit</code> | <code>Double</code> | Account's credit limit. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>marlinCustomer Authenticator</code> | <code>String</code> | |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |
| <code>mopPPID</code> | <i>Search criterion.</i> <code>String</code> | Management Operator ID. |

| Name | Type | Description |
|---|--|---|
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> String | The Name of this ServiceProvider. |
| <code>originID</code> | Integer | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <i>Read only.</i> String | |
| <code>originKey</code> | String | The key of the model object in the origin system (3rd part system). |
| <code>ottSessionLimit</code> | Integer | |
| <code>ottSessionTimeout Secs</code> | Integer | |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>uID</code> | Long | The unique identifier. |

77.2 serviceproviderService

Provides all of the required functionality to Create, Retrieve, Update and Delete 'ServiceProvider' objects on the system.
Steve @author

77.2.1 serviceproviderService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---------------------------|
| <code>model</code> | tv.quative.core.persistence. model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

77.2.2 serviceproviderService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

77.2.3 serviceproviderService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>61</originID>
</getByOrigin>
```

77.2.4 serviceproviderService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>10</uid>
</getByUID>
```

77.2.5 serviceproviderService.getGlobalSPID

Retrieves the Global Service Provider ID from the configuration..

Available through Web Services.

Parameters

None.

Return value

Type:

The Global Service Provider ID. Items belonging to the Global Service Provider can be read by all Service Providers.

Example

```
<getGlobalSPID/>
```

77.2.6 serviceproviderService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

78 ServiceProviderPaymentOptionService

This is a primitive *Service* which provides the *ServiceProviderPaymentOption* Entity.

78.1 ServiceProviderPaymentOption

A *ServiceProviderPaymentOption* Entity represents the relationship between a *Service Provider* and a *Payment Method Provider*.

To use the services of a *Payment Method Provider*, a *Service Provider* will need to set up their own merchant account with the provider and store connection and account details in this *Entity*.

Once a *ServiceProviderPaymentOption* has been created, *Accounts* with that *Service Provider* will be able to select that option for payment immediately. Some *Payment Method Providers* require credentials for individual *Accounts* to be stored in SDP as *AccountPaymentCredentials* Entities. Where this happens, the option will still be presented to the end user, but clients are expected to make it clear where further action is required.

78.1.1 ServiceProviderPaymentOption

Represents a merchant account held by a *Service Provider* with a payment provider (*Payment Method Provider*)

| Name | Type | Description |
|---------------------------|---|---|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| <code>creationDate</code> | Date | The date and time this <i>Entity</i> was created. |
| <code>enabled</code> | Boolean | Is the <i>Payment Method Provider</i> currently available for use by the <i>Service Provider</i> ? |
| <code>exportID</code> | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |
| <code>merchantKey</code> | String | The key which will identify the <i>Service Provider</i> to the payment provider, used in digest creation. |
| | | Note This field is stored in the database in encrypted form. |
| <code>merchantRef</code> | String | The username or reference representing the <i>Service Provider</i> in the payment provider's system. |

| Name | Type | Description |
|--|-------------------|---|
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>password</code> | String | The password used for communications with the payment provider, if applicable. Note This field is stored in the database in encrypted form. |
| <code>paymentMethod ProviderUid</code> | Long | The UID of the <i>Payment Method Provider</i> Entity which represents the payment provider. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>uID</code> | Long | The SDP internal UID for this <i>Entity</i> . |
| <code>username</code> | String | The username used for communications with the payment provider, if applicable. Note This field is stored in the database in encrypted form. |

78.2 ServiceProviderPaymentOptionService

Implementation of `ServiceProviderPaymentOptionService` to provide CRUD operations for `ServiceProviderPaymentOption`.

mharvey @author

78.2.1 ServiceProviderPaymentOptionService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | |

Return value

Type:

78.2.2

ServiceProviderPaymentOptionService.createServiceProviderPaymentOption

Create a new *Service Provider Payment Option*.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--|---|
| <code>pSpPayOption</code> | ServiceProviderPaymentOption | An object specifying the fields of the <i>Service Provider Payment Option</i> . |

Return value

Type:

The `uid` of the created *Service Provider Payment Option*.

78.2.3 ServiceProviderPaymentOptionService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | |

Return value

Type:

78.2.4

ServiceProviderPaymentOptionService.deleteServiceProviderOption

Delete the specified ServiceProviderPaymentOption..

Available through Web Services.

Parameters

| Name | Type | Description |
|---|------|---------------------------------|
| <code>pServiceProviderPayOptionUid</code> | Long | the ID of the option to delete. |

Return value

Type: `void`

Example

```
<deleteServiceProviderOption>
  <pServiceProviderPayOptionUid>16</pServiceProviderPayOptionUid>
</deleteServiceProviderOption>
```

78.2.5 ServiceProviderPaymentOptionService.getByOrigin

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|-------------|
| <code>originKey</code> | String | |

| Name | Type | Description |
|----------|---------|-------------|
| originID | Integer | |

Return value

Type:

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>98</originID>
</getByOrigin>
```

78.2.6 ServiceProviderPaymentOptionService.getByUID

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|-------------|
| uid | Long | |

Return value

Type:

Example

```
<getByUID>
  <uid>50</uid>
</getByUID>
```

78.2.7

ServiceProviderPaymentOptionService.getPaymentOptionsForServiceProvider

Fetch all payment options supported by the service provider.

Note returns those that both enabled and disabled.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|---------------------------------|
| pServiceProvider Uid | Long | the ID of the service provider. |

Return value

Type: List<ServiceProviderPaymentOption>

a List containing all payment options supported by the service provider.

Example

```
<getPaymentOptionsForServiceProvider>
  <pServiceProviderUid>52</pServiceProviderUid>
</getPaymentOptionsForServiceProvider>
```

78.2.8

ServiceProviderPaymentOptionService.getServiceProviderPaymentOption

Fetch a particular service provider payment option..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------------------|------|--|
| pServiceProvider PayOptionUid | Long | the ID of the service provider pay option. |

Return value

Type:

the requested ServiceProviderPaymentOption.

Example

```
<getServiceProviderPaymentOption>
  <pServiceProviderPayOptionUid>62</pServiceProviderPayOptionUid>
</getServiceProviderPaymentOption>
```

78.2.9 ServiceProviderPaymentOptionService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| model | tv.quative.core.persistence.model.BaseModel | |

Return value

Type:

78.2.10

ServiceProviderPaymentOptionService.updateServiceProviderPaymentOption

Update the specified ServiceProviderPaymentOption..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--|-----------------------|
| pspPayOption | ServiceProviderPaymentOption | the option to update. |

Return value

Type:

the number of rows updated.

79 signonService

This is an aggregate *Service* which allows set-top boxes and *Media Players* to sign on to gain access to other SDP *Services*.

Clients may initiate the sign-on process in several ways, depending on the information they have to identify the *Account*.

- ▶ Set-top box applications may supply the CA serial number of the *Set Top Box* (recommended), its MAC address, or the ID of an associated *Smart Card*.
- ▶ Media Player applications may supply the *Media Player's mediaPlayerId*, plus optionally the login ID and password of an associated *User*.
- ▶ All applications may supply the login ID and password of a *User*.

Caution!

If you sign on with the login ID and password of a *User*, SDP will not be able to identify what device is being used, so some functionality will be lost.

Depending on the deployment, some of these methods may not be available to clients.

When signing on using *Set Top Box* details, the *Set Top Box* must have been already provisioned to an *Account* in SDP. Likewise, before signing on by *Smart Card* ID, the *Smart Card* must be associated with a fully provisioned *Set Top Box*.

When a device successfully signs on, they are provided with a token which identifies their session. All further calls to SDP *Services* via the HUE gateway should include this token.

The token identifies the *Account*, its *Service Provider*, and if available the *User* and device, in encrypted form. SDP can use the token to provide personalized responses to subsequent API calls.

79.1 signonService

The Signon Service allows devices to sign on to the Qube system. Devices can currently be identified by MAC address or by Smartcard ID, although this class may be extended to handle other types of identifier. The services is made visible to the HUE and WebServices gateways by default.

zlaposa @author

Naeem Bilal @author

79.1.1 signonService.doSomeMongoLogging

Available through Web Services.

Parameters

None.

Return value

Type: `void`

Example

```
<doSomeMongoLogging/>
```

79.1.2 signonService.signonByCASN *Updated*

Signs on a device by its CASN.

Caution!

As installed, SDP requires the `casn` field to be unique across **Set Top Boxes**. However, some deployments require this constraint to be removed. Clients of an SDP which allows duplicate `casn` should avoid this API, because if a client uses it to sign on, a token will be generated which identifies one of the **Set Top Boxes** with the supplied `casn`, but which one is undefined.

Caution!

When Geoblocking feature (flag - `isEnabledGeoBlocking`) is enabled, sign-on of CASN devices will be blocked unless IP address of the device is within the allowed range of IP addresses for the region to which it belongs to.

Please refer to [Geo Blocking](#) section for more details

Parameters available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByCASN`

| Name | Type | Description |
|-------------------|--------|------------------------------------|
| <code>casn</code> | String | The CASN of the device to sign on. |

Return value

Type: `void`

The signon token is returned in the `token` field of the response object, not in the `result`.

79.1.3 signonService.signonByMACAddress

Signs on a client by its MAC address.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByMACAddress`

Parameters

| Name | Type | Description |
|-------------------------|--------|---|
| <code>macAddress</code> | String | The MAC address of the device to sign on. |

Return value

Type: `void`

The signon token is returned in the `token` field of the response object, not in the `result`.

79.1.4 signonService.signonByMpDeviceIdAndUser

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByMpDeviceIdAndUser`

Parameters

| Name | Type | Description |
|------------------------|--------|-------------|
| <code>mpId</code> | String | |
| <code>userLogin</code> | String | |
| <code>password</code> | String | |

Return value

Type: void

79.1.5 signonService.signonByPublicDeviceID *New*

Signs on a device by its `publicDeviceID`. The `publicDeviceId` is stored as `OriginKey` inside the SDP.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByPublicDeviceID`

Parameters

| Name | Type | Description |
|-----------------------------|--------|---|
| <code>publicDeviceID</code> | String | The <code>publicDeviceId</code> of the MediaPlayer/Device to sign on. |

Return value

Type: void

The signon token is returned in the `token` field of the response object, not in the `result`.

79.1.6 signonService.signonBySmartcardID

Signs on a device by its `Smartcard ID`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonBySmartcardID`

Parameters

| Name | Type | Description |
|--------------------------|--------|--|
| <code>smartcardID</code> | String | The <i>Smartcard</i> ID associated with the device to sign on. |

Return value

Type: `void`

The signon token is returned in the `token` field of the response object, not in the `result`.

79.1.7 signonService.signonByUser

Signs on using username and password.

No device is identified and hence no device id will be present in the token. A default value of 0 is used instead.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByUser`

Parameters

| Name | Type | Description |
|------------------------|--------|----------------------|
| <code>userLogin</code> | String | login |
| <code>password</code> | String | password (plaintext) |

Return value

Type: `void`

79.1.8 signonService.signonByUserMpldAndSpid

Signon for a Marlin or PlayReady DRM client for which the account has been previously created (by the CRM system), and for which the client application has already provisioned itself.

Please note that this call is very similar to `signonByMpDeviceIdAndUser`. However, we use a separate call because `signonByMpDeviceIdAndUser` is intended primarily for NMP signons. Keeping the flows separate in this way allows for them to diverge in future, should this become necessary.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByUserMpldAndSpid`

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>userLogin</code> | String | The username. |
| <code>password</code> | String | Password (in clear). This will be compared to the hashed and salted version kept in the DB. |
| <code>spOriginKey</code> | String | The origin key of the service provider to register for. The same user-login may be used across multiple accounts (one per service provider) so it's necessary to know which one the user is attempting to log into. |
| <code>mpId</code> | String | The media-player ID. This identifies the application. |

Return value

Type: void

79.1.9 `signonService.signonInitialByUserAndSpid`

Signon for a Marlin or PlayReady DRM client for which the account has been previously created (by the CRM system), but for which the client application has not yet provisioned itself.

If this call completes successfully, the client must then call `ProvisionService#provisionMediaPlayer`.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonInitialByUserAndSpid`

Parameters

| Name | Type | Description |
|--------------------------|--------|---|
| <code>userLogin</code> | String | The username. |
| <code>password</code> | String | Password (in clear). This will be compared to the hashed and salted version kept in the DB. |
| <code>spOriginKey</code> | String | The origin key of the service provider to register for. The same user-login may be used across multiple accounts (one per service provider) so it's necessary to know which one the user is attempting to log into. |

Return value

Type: void

79.1.10 signonService.updateTokenWithMediaPlayer

Update the signon token with a new media player device id attached to the same account The media player must already be signed in with a valid token to use this method..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/updateTokenWithMediaPlayer`

Parameters

| Name | Type | Description |
|-------------------|--------|----------------------------|
| <code>mpId</code> | String | the media player device id |

Return value

Type: `void`

79.1.11 signonService.registerDeviceAndGetToken *New*

Overview

Signon for ALL devices except the PRM ones (HW PRM and SW PRM devices should continue to use initialise service on MDRM).

This operation will create the device (if not existing) and return a token.

Description

This operation requires as mandatory inputs user login and user password but optionally deviceInfo, drmDeviceId and token can be given.

Note that if the deviceinfo and drmdeviceid are not given then these values will be stored as null for the new device that gets created.

This operation can be called in the following scenarios:

1. For a new signon request (in which case token should not be present as input) Operation validates user credentials (login/password) then creates a new device and returns a token and mediaPlayerID (generated by SDP) in the response section.
2. For already signedon devices (where token should be present as input)
Operation validates the given token against user credentials and extracts the mediaPlayer Id from the token and returns a new refreshed token and still return the mediaPlayerID in the response section.

Important Note: If the device already gets a **drmdeviceID** given by the local security lib, then it's better for security to use it during the register request.

Such **drmDeviceId** availability depends on the platform, some offers it some others don't...

If not provided then CLM will create one on behalf of the device during the first license request and SDP will use it.

Definition

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/registerDeviceAndGetToken`

Request input parameters

| Name | Type | Description | Cardinality |
|-------|--|--|---|
| arg0 | String | User login | mandatory |
| arg1 | String | User password (in clear) Note: This will be compared to the hashed and salted version kept in the DB. | mandatory |
| arg2 | <u>DeviceInformation</u> (<u>DeviceInformation</u>) | Device Information generated by a device/ player Note: If presents, SDP uses this device information to associate the new device to a "Device Classifier" accordingly to rules defined. | optional |
| arg3 | String | drmdeviceid defined by device | optional |
| token | String | Encrypted token string | optional (can be set also in the header) Note that header will prime over parameter |

Requests examples

1.Request with only mandatory fields:

```
http://<server>/hue-gateway/gateway/http/js/signonService/registerDeviceAndGetToken?
arg0=username&arg1=password
```

2.Request with deviceinfo and drmdeviceid:

```
http://<server>/hue-gateway/gateway/http/js/signonService/registerDeviceAndGetToken?
arg0=username&arg1=password&arg2={"device": {"CPU":{"cores":2,"frequency":2.5329999923706055},"GPU":
{"cores": 0,"frequency": 0,"model": "VMware SVGA 3D"},"OS":{"type": "iOS", "version": "iOS1.1"},"hardware":
{"manufacturer": "VMware, Inc.,"model": "VMware Virtual Platform", "type": "OTT"},"screen": {"density": 0,"height":
1200,"width": 1920},"securePlayer":{"DRMs": ["WV"],"codecs":["H264"],"streamings":["DASH"]}}&arg3=nagra1234
```

3.Request from already signed-on device (Token will be in request parameter)

```
http://<server>/hue-gateway/gateway/http/js/signonService/registerDeviceAndGetToken?
arg0=username&arg1=password&arg2={"device": {"CPU":{"cores":2,"frequency":2.532999923706055},"GPU":
{"cores": 0,"frequency": 0,"model": "VMware SVGA 3D"},"OS":{"type": "iOS", "version": "iOS1.1"},"hardware":
{"manufacturer": "VMware, Inc.", "model": "VMware Virtual Platform", "type": "OTT"},"screen": {"density":
0,"height": 1200,"width": 1920}}, "securePlayer":{"DRMs": ["WV"],"codecs":["H264"],"streamings":
["DASH"]}}}&arg3=nagra1234&token=OZlZ1q_Gr9S3d6KLX1whHDkeKZdscpx2LHN1YnNjcmliZXItNjA4LDEs
S0JSTyw2MDGsMiwYMDMscXdlcnR5MTlyMywwLCwsb2FtLWRIZmF1bHQtaGlkZGVu
```

Response example

```
<Response xmlns="http://10.15.2.140/hue-gateway/gateway/http/js/signonService/register
DeviceAndGetToken">
  <requestId>846452539</requestId>
  <result>
    <mediaPlayerId>4294967903</mediaPlayerId>
  </result>
  <resultCode>0</resultCode>
  <token>qaTjXs0MftKvDy9ZT6ZWwwTvkFZaFBCdNDI5NDk2NzkwMyxzdWJzY3JpYmVyLTE5ODIsMSxLQl
JPLDE5ODIsTkLEMTIzNDUsMTY1MSwsMTkxOSwsLG9hbS1kZWZhdWx0LWwhpZGRlbg==</token>
</Response>
```

79.1.12 signonService.getCastToken New

Overview

Signon for ChromeCast devices.

This operation will create the device and return a token.

Description

This operation requires as mandatory inputs user login and user password and token but optionally deviceInfo, drm DeviceId and returns the token as out put in the response.

Note that if the deviceinfo and drmdeviceid are not given then these values will be stored as null for the new device that gets created.

This operation can be called in the following scenarios:

1. For a new signon getCastToken request Operation validates user credentials (login/password) then creates a new MpDevice with mediaPlayer Id prefix as 'castDev-' and returns a token in the response section.
2. For the subsequent getCastToken request (where token should be present as input)
Operation validates the given token against user credentials and creates the new MpDevice with 'castDev-' as mediaPlayerId prefix and returns a new refreshed token in the response section.

Important Note: If the device already gets a **drmdeviceID** given by the local security lib, then it's better for security to use it during the getCastToken request.

Such **drmDeviceId** availability depends on the platform, some offers it some others don't...

If not provided then CLM will create one on behalf of the device during the first license request and SDP will use it.

Definition

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/getCastToken`

Request input parameters

| Name | Type | Description | Cardinality |
|-------|---|--|---|
| arg0 | String | User login | mandatory |
| arg1 | String | User password (in clear) Note: This will be compared to the hashed and salted version kept in the DB. | mandatory |
| arg2 | <u>DeviceInformation</u> (DeviceInformation) | Device Information generated by a device/ player Note: If presents, SDP uses this device information to associate the new device to a "Device Classifier" accordingly to rules defined. | optional |
| arg3 | String | drmdeviceid defined by device | optional |
| token | String | Encrypted token string | Mandatory (need to pass token which is generated by registerDeviceAndGetToken api) |

Requests examples

1.Request with only mandatory fields:

```
http://<server>/hue-gateway/gateway/http/js/signonService/getCastToken?arg0=username&arg1=password
```

2.Request with deviceinfo and drmdeviceid:

```
http://<server>/hue-gateway/gateway/http/js/signonService/getCastToken?arg0=username&arg1=password&arg2={"device":{"CPU":{"cores":2,"frequency":2.5329999923706055},"GPU":{"cores":0,"frequency":0,"model":"VMware SVGA 3D"},"OS":{"type":"iOS","version":"iOS1.1"},"hardware":
```

```
{"manufacturer": "VMware, Inc.", "model": "VMware Virtual Platform", "type": "OTT"}, "screen": {"density": 0, "height": 1200, "width": 1920}}, "securePlayer": {"DRMs": ["WV"], "codecs": ["H264"], "streamings": ["DASH"]}]&arg3=nagra1234
```

3. Request from already signed-on device (Token will be in request parameter)

```
http://<server>/hue-gateway/gateway/http/js/signonService/getCastToken?
arg0=username&arg1=password&arg2={"device": {"CPU":{"cores":2,"frequency":2.5329999923706055},"GPU":
{"cores": 0,"frequency": 0,"model": "VMware SVGA 3D"},"OS":{"type": "iOS","version": "iOS1.1"},"hardware":
{"manufacturer": "VMware, Inc.", "model": "VMware Virtual Platform", "type": "OTT"}, "screen": {"density":
0, "height": 1200, "width": 1920}}, "securePlayer": {"DRMs": ["WV"], "codecs": ["H264"], "streamings":
["DASH"]}]&arg3=nagra1234&token=OZlZ1q_Gr9S3d6KLX1whHDkeKZdscpx2LHN1YnNjcmliZXItNjA4LDEs
S0JSTyw2MDGsMiwYMDMscXdlcnR5MTIyMywwLCwsb2FtLWRIZmF1bHQtaGlkZGVu
```

Response example

```
<Response xmlns="http://10.15.2.60/hue-gateway/gateway/http/js/signonService/getCast
Token">
  <requestId>285679058</requestId>
  <result null="true"/>
  <resultCode>0</resultCode>
  <token>0ysMKLaQcKQqBph_-C5nutaAchRdTpGVY2FzdERldi130XB4YkpmSm9TdE1BbVk5VCxzdWJzY3Jp
YmVyLTUxLDEsS0JSTyw1MSxjYXN0ZGV2aWNlYWNjMDAwNSwzMSwsNTMsLCxvYW0tZGVmYXVsdC1oaWRkZW4=</
token>
</Response>
```

79.1.13 signonService.signonByDASDeviceandUser New

Overview

Signon for signonByDASDeviceandUser devices.

This operation will create the device and return a token.

Description

This operation requires as mandatory inputs user login and user password and token but optionally deviceInfo, drm DeviceId and returns the token as out put in the response.

Note that if the deviceinfo and drmdeviceid are not given then these values will be stored as null for the new device that gets created.

This operation can be called in the following scenarios:

1. For a new signonByDASDeviceandUser request Operation validates user credentials (login/password) then creates a new MpDevice with mediaplayer Id prefix as 'castDev-' and returns a token in the response section.
2. For the subsequent signonByDASDeviceandUser request (where token should be present as input)
Operation validates the given token against user credentials and creates the new MpDevice with 'castDev-' as mediaplayerId prefix and returns a new refreshed token in the response section.

Important Note: If the device already gets a **drmdeviceID** given by the local security lib, then it's better for security to use it during the `signonByDASDeviceandUser` request.

Such **drmDeviceId** availability depends on the platform, some offers it some others don't...

If not provided then CLM will create one on behalf of the device during the first license request and SDP will use it.

Definition

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/signonService/signonByDASDeviceandUser`

Request input parameters

| Name | Type | Description | Cardinality |
|-------|--|--|---|
| arg0 | String | User login | mandatory |
| arg1 | String | User password (in clear) Note: This will be compared to the hashed and salted version kept in the DB. | mandatory |
| arg2 | DeviceInformation (DeviceInformation) | Device Information generated by a device/player Note: If presents, SDP uses this device information to associate the new device to a "Device Classifier" accordingly to rules defined. | optional |
| arg3 | String | drmdeviceid defined by device | optional |
| token | String | Encrypted token string | Mandatory (need to pass token which is generated by registerDeviceAndGetToken api) |

Requests examples

1.Request with only mandatory fields:

```
http://<server>/hue-gateway/gateway/http/js/signonService/signonByDASDeviceandUser?arg0=username&arg1=password
```

2.Request with deviceinfo and drmdeviceid:

```
http://<server>/hue-gateway/gateway/http/js/signonService/signonByDASDeviceandUser?
arg0=username&arg1=password&arg2={"device": {"CPU":{"cores":2,"frequency":2.5329999923706055},"GPU":
{"cores": 0,"frequency": 0,"model": "VMware SVGA 3D"},"OS":{"type": "iOS", "version": "iOS1.1"},"hardware":
{"manufacturer": "VMware, Inc.", "model": "VMware Virtual Platform", "type": "OTT"},"screen": {"density": 0,"height":
1200,"width": 1920}}, "securePlayer":{"DRMs": ["WV"],"codecs":["H264"],"streamings":["DASH"]}}&arg3=nagra1234
```

3. Request from already signed-on device (Token will be in request parameter)

```
http://<server>/hue-gateway/gateway/http/js/signonService/signonByDASDeviceandUser?
arg0=username&arg1=password&arg2={"device": {"CPU":{"cores":2,"frequency":2.5329999923706055},"GPU":
{"cores": 0,"frequency": 0,"model": "VMware SVGA 3D"},"OS":{"type": "iOS", "version": "iOS1.1"},"hardware":
{"manufacturer": "VMware, Inc.", "model": "VMware Virtual Platform", "type": "OTT"},"screen": {"density":
0,"height": 1200,"width": 1920}}, "securePlayer":{"DRMs": ["WV"],"codecs":["H264"],"streamings":
["DASH"]}}&arg3=nagra1234&token=OZlZ1q_Gr9S3d6KLX1whHDkeKZdscpx2LHN1YnNjcmliZXItNjA4LDEs
S0JSTyw2MDGsMiwYMDMscXdlcnR5MTlyMywwLCwsb2FtLWRIZmF1bHQtaGikZGVu
```

Response example

```
<Response xmlns="http://10.15.2.60/hue-gateway/gateway/http/js/signonService/signonBy
DASDeviceandUser">
  <requestId>285679058</requestId>
  <result null="true"/>
  <resultCode>0</resultCode>
  <token>0ysMKLaQcKQqBph_-C5nutaAchRdTpqVY2FzdERldi130XB4YkpmSm9TdE1BbVk5VCxzdWJzY3Jp
YmVyLTUxLDEsS0JSTyw1MSxjYXN0ZGV2aWN1YWNjMDAwNSwzMSwsNTMsLCxvYW0tZGVmYXVsdC1oaWRkZW4=</
token>
</Response>
```

80 smartCardService

This is a primitive *Service* which provides the *Smart Card* Entity.

Caution!

If you create or update a *Smart Card* using this *Service*, the CAS will not be notified. The *DeviceAggregate* Service provides *Smart Card* methods which integrate with the SMS.

80.1 Smart Card

A *Smart Card* represents the component in a client device which identifies it to the CAS.

Figure 26 SmartCard and associated Entities.

In a card-based CAS environment, the *Smart Card* corresponds to a physical card which handles cryptographic authentication for an *Account*.

Cardless CAS systems have a concept of a virtual smartcard which identifies the device. A *Smart Card* Entity is still used in SDP to represent this virtual smartcard.

Note

SDP can manage *Smart Cards* on its own: there is no need to create or delete them explicitly. A *Smart Card* Entity is automatically created when you refer to it by its smartcard ID.

80.1.1 SmartCard

| Name | Type | Description |
|---------------|---|--|
| casInstanceId | String | The ID of the <i>CAS Instance</i> which manages this <i>Smart Card</i> . For cardless CAS systems, the <i>Smart Card</i> represents the "virtual smartcard" created to represent the device to the CAS. |
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | A log of the changes to this <i>Entity</i> history. |
| creationDate | Date | The date and time this <i>Entity</i> was created. |
| exportID | Integer | The identifier to use when exporting this <i>Entity</i> to third-party systems. |

| Name | Type | Description |
|--------------------------------|--|--|
| <code>modifiedDate</code> | Date | The date and time this <i>Entity</i> was last modified in the database. |
| <code>originID</code> | Integer | An identifier for the system which owns this <i>Entity</i> . Identifiers up to 9999 are reserved for identifying systems pre-integrated with SDP. Operators may define their own meanings for values of 10000 and above. |
| <code>originIdAndKey</code> | Read only. String | A unique reference for the object, derived by joining the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | String | The key or identifier which the system which owns this <i>Entity</i> uses to identify it. |
| <code>serviceProviderID</code> | Long | The SDP internal UID for the <i>Service Provider</i> which created this <i>Entity</i> . |
| <code>smartcardID</code> | Search criterion. Quick Search criterion. String | The unique ID of the <i>Smart Card</i> , as used by the CAS. In deployments using a Nagra CAS, this ID must be a decimal number with either 10 or 12 digits. If it has 12 digits, the last 2 digits are a checksum of the first 10 digits. |
| | | <p>Note</p> <p>SDP treats this field as text, even if your CAS uses numeric smart card IDs. Any leading zeroes must be included in the value like any other character.</p> <p>For cardless CAS systems, this field holds the VUA assigned by SDP to the device for the purposes of the CAS.</p> |
| <code>status</code> | Search criterion. Quick Search criterion. String | Whether the <i>Smart Card</i> is available for pairing with a <i>Set Top Box</i> . One of |

| Value | Description |
|-----------|---|
| AVAILABLE | The <i>Smart Card</i> is ready to be paired with a <i>Set Top Box</i> . |

| Name | Type | Description | | | | | | |
|--------------------|---|--|-------|-------------|--------------------|---|------------------|---|
| | | <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>UNAVAILABLE</td> <td>The <i>Smart Card</i> is not available for pairing, either because it is already in use or for some other reason.</td> </tr> <tr> <td>CANCELLED</td> <td>The <i>Smart Card</i> has been cancelled, and is no longer valid for accessing content controlled by the CAS.</td> </tr> </tbody> </table> <p>The CAS is automatically notified of the status change.</p> <p>If you change the status of a <i>Smart Card</i> to AVAILABLE or CANCELLED when it is already paired with a <i>Set Top Box</i>, the association with the <i>Set Top Box</i> is automatically broken.</p> <p>In the case of AVAILABLE, the <i>Smart Card</i> can now be used for pairing with a different <i>Set Top Box</i>.</p> <p>Caution!</p> <ul style="list-style-type: none"> ▶ Do not set the Status of a <i>Smart Card</i> to AVAILABLE if it is in use. The <i>Set Top Box</i> will no longer be able to access any content controlled by the CAS. ▶ Only set the Status of a <i>Smart Card</i> to CANCELLED if you mean to prevent the <i>Set Top Box</i> from accessing any content controlled by the CAS. <p>To reassociate a <i>Set Top Box</i> with a <i>Smart Card</i> in QSP Manager, set its Status to AVAILABLE and use the <i>Set Top Box</i> Qubelet to restore the association.</p> | Value | Description | UNAVAILABLE | The <i>Smart Card</i> is not available for pairing, either because it is already in use or for some other reason. | CANCELLED | The <i>Smart Card</i> has been cancelled, and is no longer valid for accessing content controlled by the CAS. |
| Value | Description | | | | | | | |
| UNAVAILABLE | The <i>Smart Card</i> is not available for pairing, either because it is already in use or for some other reason. | | | | | | | |
| CANCELLED | The <i>Smart Card</i> has been cancelled, and is no longer valid for accessing content controlled by the CAS. | | | | | | | |
| statusCode | String | A short code which mirrors the value of the status . One of | | | | | | |

| Value | Description |
|----------|------------------|
| A | AVAILABLE |

| Name | Type | Description | | | | | | |
|-------------|---------------------------|--|-------|-------------|---|-------------|---|-----------|
| | | <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>U</td> <td>UNAVAILABLE</td> </tr> <tr> <td>C</td> <td>CANCELLED</td> </tr> </tbody> </table> | Value | Description | U | UNAVAILABLE | C | CANCELLED |
| Value | Description | | | | | | | |
| U | UNAVAILABLE | | | | | | | |
| C | CANCELLED | | | | | | | |
| stbUid | Long | | | | | | | |
| uID | Long | The SDP internal UID for this <i>Entity</i> . | | | | | | |
| unavailable | <i>Read only.</i> boolean | | | | | | | |

80.2 smartCardService

Provides all of the required functionality to Create, Retrieve, Update and Delete 'Device' entities on the system.

Saci @author

80.2.1 smartCardService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

80.2.2 smartCardService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

80.2.3 smartCardService.getAvailableSmartcardList

Retrieve the list of available smart cards from the DB starting with smart card id (AJAX).

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|---------|---|
| smartCardId | String | |
| startRow | Integer | the row count from which the row starts |
| endRow | Integer | the row count from which the row ends |

Return value

Type: `List<SmartCard>`

a List Of SmartCards

Example

```
<getAvailableSmartcardList>
  <smartCardId>stringValue</smartCardId>
  <startRow>30</startRow>
  <endRow>97</endRow>
</getAvailableSmartcardList>
```

80.2.4 smartCardService.getAvailableSmartcardListCount

Retrieve the count of smart cards..

Available through Web Services.

Parameters

None.

Return value

Type:

the count of smart cards

Example

```
<getAvailableSmartcardListCount/>
```

80.2.5 smartCardService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>87</originID>
</getByOrigin>
```

80.2.6 smartCardService.getBySmartCardId

Retrieve, from storage, a SmartCard identified by its Id..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|--------|-------------|
| smartcardID | String | |

Return value

Type:

The Smart Card with the id that is same as smartcardID

Example

```
<getBySmartCardId>
  <smartcardID>ABC-57357</smartcardID>
</getBySmartCardId>
```

80.2.7 smartCardService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>79</uid>
</getByUID>
```

80.2.8 smartCardService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

81 spChannelMapService

This is a primitive *Service* which manages the relation between *Service Providers* and *Channels*.

It is used in cases where different *Service Providers* use different channel numbers for the same channel, as identified by its UID.

To map a channel number for a particular *Service Provider*, an *SPChannelMap* Entity is created to override the default choice of channel number as set by the *Global Service Provider*.

81.1 sPChannelMapService

Provides all of the required functionality to Create, Retrieve, Update and Delete **sPChannelMapService** objects on the system.

zlaposa @author

81.1.1 sPChannelMapService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to persist. |

Return value

Type:

UID of the created object.

81.1.2 sPChannelMapService.delete

Delete, from storage, a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

81.1.3 sPChannelMapService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|---|
| originKey | String | String the primary key of the record in a customer system |
| originID | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>60</originID>
</getByOrigin>
```

81.1.4 sPChannelMapService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>63</uid>
</getByUID>
```

81.1.5 sPChannelMapService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------------------|
| model | tv.quative.core.persistence.model.BaseModel | The BaseModel to delete |

Return value

Type:

the number of rows updated.

82 statisticsReportingService

This *Service* is used by various report types to obtain data about users' usage of SDB (switched digital broadcast) channels

82.1

tv.quative.service.statisticsreporting.model.AggregationLevel

| Name | Type | Description |
|--------|------|-------------|
| DAILY | | Daily. |
| HOURLY | | hourly. |
| WEEKLY | | Weekly. |

82.2

tv.quative.service.statisticsreporting.model.DomainAggregationLevel

| Name | Type | Description |
|------------|------|--|
| ALL | | All domain objects are aggregated into a single graph. |
| ERM | | Service groups (only) are aggregated by ERM. |
| INDIVIDUAL | | Each domain object (service group, channel) has its own graph. |

82.3 statisticsReportingService

Implementation for Statistics Reporting service. CAUTION - Horrible hack alert! When originally written, SGs were identified by origin since those were compulsory, and unique. Now they are neither, but SG names are both. Due to scheduling constraints it has been necessary to bodge this change into the Java layer without changing the DB (except very minimally). To do this, we use a fixed, fake origin ID, and an "origin key" which is actually the SG name.

Prabhu Subramanian @author

wdickson @author

82.3.1 statisticsReportingService.getActivity

Historical activity level (whether you need a bigger datacentre).

NB. domain aggregation does not apply. to this.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|---|---|
| startDate | Date | Start of period |
| endDate | Date | End of period |
| timeAggregation | tv.quative.service.statisticsreporting.model.AggregationLevel | Granularity over time (number of data points on each chart) |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw. (There will always be 3: one each for joins, voluntary leaves and evictions).

82.3.2 statisticsReportingService.getChannelPopularity

Historical channel popularity (tuner count)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------|---|---|
| startDate | Date | Start of period |
| endDate | Date | End of period |
| timeAggregation | tv.quative.service.statisticsreporting.model.AggregationLevel | Granularity over time (number of data points on each chart) |

| Name | Type | Description |
|--------------------------------|--|---|
| <code>domainAggregation</code> | <code>tv.quative.service.statisticsreporting.model.DomainAggregationLevel</code> | Granularity over domain (number of charts). NB. you cannot aggregate channels by ERM. |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw.

82.3.3 `statisticsReportingService.getLiveActivity`

Current activity level (most recent recorded minute)..

Available through Web Services.

Parameters

None.

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw. (There will always be 3: one each for joins, voluntary leaves and evictions).

Example

```
<getLiveActivity/>
```

82.3.4 `statisticsReportingService.getLiveChannelPopularity`

Current channel popularity (most recent recorded minute)..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------------|--|--|
| <code>domainAggregation</code> | <code>tv.quative.service.statisticsreporting.model.DomainAggregationLevel</code> | Granularity over domain (number of charts) |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw.

82.3.5 statisticsReportingService.getLiveServiceGroupPopularity

Current service group popularity (most recent recorded minute)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|--|
| domainAggregation | tv.quative.service.statisticsreporting.model.DomainAggregationLevel | Granularity over domain (number of charts) |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw.

82.3.6 statisticsReportingService.getLiveServiceGroupUtilisation

Current service group utilisation (most recent recorded minute)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|--|
| domainAggregation | tv.quative.service.statisticsreporting.model.DomainAggregationLevel | Granularity over domain (number of charts) |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw.

82.3.7 statisticsReportingService.getServiceGroupPopularity

Historical service group popularity (tuner count)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|------|-----------------|
| startDate | Date | Start of period |

| Name | Type | Description |
|-------------------|---|---|
| endDate | Date | End of period |
| timeAggregation | tv.quative.service.statisticsreporting.model.AggregationLevel | Granularity over time (number of data points on each chart) |
| domainAggregation | tv.quative.service.statisticsreporting.model.DomainAggregationLevel | Granularity over domain (number of charts) |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw.

82.3.8 statisticsReportingService.getServiceGroupUtilisation

Historical service group utilisation (channel count / bandwidth usage)..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|---|
| startDate | Date | Start of period |
| endDate | Date | End of period |
| timeAggregation | tv.quative.service.statisticsreporting.model.AggregationLevel | Granularity over time (number of data points on each chart) |
| domainAggregation | tv.quative.service.statisticsreporting.model.DomainAggregationLevel | Granularity over domain (number of charts) |

Return value

Type: `List<CategoryDataset>`

The list of data sets. Each defines a single chart to draw.

83 subscriptionProfileService

This is an aggregate *Service* which provides the methods required to subscribe to *Packages*, *Catalogues* and *Subscription Profiles*.

Caution!

The *SubscriptionProfile* Service does not provide the *Subscription Profile* Entity. *Subscription Profiles* are defined by the *Profile* Service.

This *Service* also provides internal methods for subscribing to broadcast *Entities*, but these methods only affect the SDP system. To subscribe to any subscribable *Entity*, clients should use the *Subscription* Service so that the SMS is also notified.

83.1

tv.quative.service.subscriptionprofile.impl.ProfileSmartcardSubscribab

| Name | Type | Description |
|--------------------------------------|---------------------------------|--|
| <code>anyAssociatedSmartCards</code> | <code>boolean</code> | This is an internal field which should not be set. |
| <code>deferredExpiryDate</code> | <code>Date</code> | See bug #9929 |
| <code>profile</code> | <code>Profile</code> | A reference to Profile. |
| <code>smartcardList</code> | <code>List<String></code> | A list containing the <code>smartcardId</code> of each <i>Smart Card</i> to be subscribed or unsubscribed to this <i>Catalogue</i> , or an empty list to subscribe or unsubscribe the whole <i>Account</i> . |
| | | Tip In Web Services, omit the <code><smartcardList></code> element to denote the empty list. |
| <code>userSubscribable</code> | <code>boolean</code> | See bug #9929 |
| <code>wrappedModelObject</code> | <code>BaseModel</code> | This is an internal field which should not be set. |

83.2 subscriptionProfileService

The Subscription service handles subscription profiles in the Qube system. The services are made visible to Hue or to WebServices using annotations. For example: For the ServiceClass: @QubeService(name="SubscriptionProfileService", hue=true, ws=true) For the Methods: @QubeMethod(hue=true, ws=true)

dhanumandla @author

83.2.1 subscriptionProfileService.getProfilesByAccountUID

Gets the list of subscription profiles belonging to the specified account.

The list contains only the currently active subscriptions.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|---------|-------------|
| accountUID | Long | Account UID |
| recursive | boolean | boolean |

Return value

Type: List<Profile>

List<Profile> - List of profiles belonging to the given account

83.2.2 subscriptionProfileService.subscribeCatalogue

Subscribes an **Account**, or a specified set of **Set Top Boxes** belonging to the same **Account**, to one or more **Catalogues**.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|---|--|
| accountUID | Long | The uid of the Account . |
| userUID | Long | The uid of the User , or null to subscribe without specifying a User . |
| catalogueList | List<Catalogue Smartcard SubscribableModel> | The Catalogues to subscribe, and (optionally) the devices to subscribe to them. |

Return value

Type: `List<Long>`

A list containing the `uid` of each **Acquired Content List** Entity that was created.

83.2.3 subscriptionProfileService.subscribePackage

Subscribes an **Account**, or a specified set of **Set Top Boxes** belonging to the same **Account**, to one or more **Packages**.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|--|---|
| <code>accountUID</code> | Long | The <code>uid</code> of the Account . |
| <code>userUID</code> | Long | The <code>uid</code> of the User , or null to subscribe without specifying a User . |
| <code>packageListModel</code> | <code>List<PackageSmartcardSubscribableModel></code> | The Packages to subscribe, and (optionally) the devices to subscribe to them. |

Return value

Type: `List<Long>`

A list containing the `uid` of each **Acquired Content List** Entity that was created.

83.2.4 subscriptionProfileService.subscribeProfile

Subscribes an **Account**, or a specified set of **Set Top Boxes** belonging to the same **Account**, to one or more **Subscription Profiles**.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|---|
| <code>accountUID</code> | Long | The <code>uid</code> of the Account . |
| <code>userUID</code> | Long | The <code>uid</code> of the User , or null to subscribe without specifying a User . |

| Name | Type | Description |
|---------------------------|---|--|
| parentProfileModel | tv.quative.service.subscriptionprofileimpl.ProfileSmartcardSubscribableModel ^{p.864} | Profile object |
| start | boolean | boolean flag indicates whether the method should be called again (in case of 'true') |
| locale | String | String |

Return value

Type: `List<Long>`

List of Integers which indicates the ACL UID's created in the system

83.2.5 subscriptionProfileService.unsubscribeCatalogue

unsubscribe Catalogue's for a given account id.

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|--|--|
| accountUID | Long | Long account UID |
| catalogueList | <code>List<CatalogueSmartcardSubscribableModel></code> | List of CatalogueSmartcardSubscribableModel objects to be unsubscribed |

Return value

Type: `void`

83.2.6 subscriptionProfileService.unsubscribePackage

unsubscribe a package for a given account id.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------|---|--|
| accountUID | Long | account UID |
| packageList | List<PackageSmartcardSubscribableModel> | List of PackageSmartcardSubscribableModel objects to be unsubscribed |

Return value

Type: void

83.2.7 subscriptionProfileService.unsubscribeProfile

unsubscribe a profile for a given account id.

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|---|--|
| accountUID | Long | account UID |
| parentProfile | tv.quative.service.subscriptionprofileapi.ProfileSmartcardSubscribableModel | ProfileSmartcardSubscribableModel object ^{p.864} |
| start | boolean | boolean flag to indicate whether to call the method recursively or not |

Return value

Type: void

84 subscriptionService

This internal *Service* provides common subscription logic for multiple product types.

Clients should not use this *Service* directly. Instead, they should use the *bocPurchaseService* or *workflowSubscriptionService* to purchase content.

84.1 tv.quative.service.subscription.model.PriceInfo

| Name | Type | Description |
|---------------------------------|--------|--|
| <code>casId</code> | String | CAS ID. |
| <code>currency</code> | String | Currency of the price. |
| <code>expirationDate</code> | Date | The expiration date of the price. |
| <code>frequency</code> | String | Price frequency. tv.quative.service.policy.model.BasePrice @see tv.quative.service.policy.model.FrequencyType @see |
| <code>intervalValue</code> | Long | Value of the expiry interval. |
| <code>intervalValueType</code> | String | Type of the expiry interval. |
| <code>name</code> | String | Name of the price's policy group. |
| <code>nextRecurrenceDate</code> | Date | Defines the next recurrence date. |
| <code>offerPrice</code> | Double | |
| <code>originKey</code> | String | Origin key. |
| <code>price</code> | Double | The computed price of a subscribable item. |
| <code>productType</code> | String | The Product Type used by the cas. |

| Name | Type | Description |
|-----------|--------|--|
| status | String | Indicates the price availability, it can be expired, current or available in the future. |
| validFrom | Date | Name of the price's policy group. The start date of the price's validity. |

84.2 tv.quative.service.subscription.model.SubscribableModel

| Name | Type | Description |
|-----------------------|---|---|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| creationDate | Date | The date created. |
| exportID | Integer | The id to export object to third part systems. |
| modifiedDate | Date | The date last modified and persisted. |
| originID | Integer | The uid of the origin system (3rd part system). |
| originIdAndKey | String | |
| originKey | String | The key of the model object in the origin system (3rd part system). |
| pPVPurchase | boolean | |
| parentAclUID | Long | UID of the parent ACL, this is used for Multiple, Season pass and Subscription ACLs. It refers to the ACI with the policy group, used when we need to unsubscribe (event or asset). |
| parentPolicyGroup UID | Long | UID of the parent policy group, if will be used if policy GroupUID is null andd profile UID is not null. |
| policyGroupUID | Long | Policy Group UID. |

| Name | Type | Description |
|--------------------------|---------|--|
| profileUID | Long | Profile UID. |
| serviceProviderID | Long | The Service provider UID. |
| subscribableItem Type | String | Indicates the type of the subscribable item. |
| subscribed | boolean | Flag indicating whether the user is already subscribed to this item. |
| uID | Long | The unique identifier. |

84.3

tv.quative.service.subscription.model.LocaleSubscribableModel

| Name | Type | Description |
|----------------|---|--|
| changeLog | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| creationDate | Date | The date created. |
| description | String | The short description of the model element. |
| exportID | Integer | The id to export object to third part systems. |
| locale | String | The locale description of the language. |
| modifiedDate | Date | The date last modified and persisted. |
| modifiedDateML | Date | The date last modified and persisted. |
| name | <i>Search criterion. Quick Search criterion.</i> String | The name of the model element. |

| Name | Type | Description |
|------------------------------------|----------------------|---|
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | <code>String</code> | |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |
| <code>pVPPurchase</code> | <code>boolean</code> | |
| <code>parentAclUID</code> | <code>Long</code> | UID of the parent ACL, this is used for Multiple, Season pass and Subscription ACLs. It refers to the ACI with the policy group, used when we need to unsubscribe (event or asset). |
| <code>parentPolicyGroup UID</code> | <code>Long</code> | UID of the parent policy group. It will be used if policy GroupUID is null and profile UID is not null. |
| <code>policyGroupUID</code> | <code>Long</code> | Policy Group UID. |
| <code>privateMetadata</code> | <code>String</code> | |
| <code>profileUID</code> | <code>Long</code> | Profile UID. |
| <code>serviceProviderID</code> | <code>Long</code> | The Service provider UID. |
| <code>subscribableItem Type</code> | <code>String</code> | Indicates the type of the subscribable item as stored in the database. |
| <code>subscribed</code> | <code>boolean</code> | Flag indicating whether the user is already subscribed to this item. |
| <code>uID</code> | <code>Long</code> | The unique identifier. |

84.4 subscriptionService

The Subscription service handles subscriptions in the Qube system. The services are made visible to Hue or to Web Services using annotations. For example: For the ServiceClass:

hue=true, ws=true) For the Methods: `@QubeService(name="SubscriptionService",`

ws=true) @QubeMethod(hue=false,
zlaposa @author
Naeem Bilal @author
Ashish Gupta @author
iahmedhadzajeva @author

84.4.1 subscriptionService.getCalculatedPrice

Calculates the price defined in the a policygroup at a given time..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/subscriptionService/getCalculatedPrice`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|--------|---------------------------------|
| policyGroupUID | Long | The UID of the policy group |
| calcDate | Date | Date to calculate the price for |
| locale | String | Locale code |

Return value

Type:

Double - The calculated price

84.4.2 subscriptionService.getCalculatedPriceWithOffer

Calculates the price and offerPrice .

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------|------|---------------------------------|
| policyGroupUID | Long | The UID of the policy group |
| calcDate | Date | Date to calculate the price for |

| Name | Type | Description |
|-------------|---|-------------|
| locale | String | Locale code |
| offerDetail | tv.quative.service.acquiredcontentlist.model.PurchasedOffer | |

Return value

Type:

PriceInfo - The calculated price

84.4.3 subscriptionService.subscribe

Subscribes an account to the given purchased item..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|--|
| accountUID | Long | Account unique identifier |
| userUID | Long | User unique identifier |
| subscribableModel | tv.quative.service.subscription.model.SubscribableModel | Model object of the item to subscribe to |
| smartcardID | String | ID of the smartcard if exists |

Return value

Type:

Long - The UID of the created Acquired Content List entry

84.4.4 subscriptionService.subscribeLocaleBased

Subscribes a local based account to the given purchased item..

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|--|---|
| accountUID | Long | Account unique identifier |
| userUID | Long | User unique identifier |
| locale SubscribableModel | tv.quative.service.subscription.Model.LocalSubscribableModel | Locale based model object of the item to subscribe to |
| smartcardID | String | ID of the smartcard if exists |

Return value

Type:

Long - The UID of the created Acquired Content List entry

84.4.5 subscriptionService.unsubscribe

Removes the account's subscription from the item..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|--|
| accountUID | Long | UID of the account |
| subscribableModel | tv.quative.service.subscription.Model.SubscribableModel | Model object of the item to subscribe to |
| smartcardID | String | ID of the smartcard if exists |

Return value

Type:

Integer - Number of unsubscribed items

84.4.6 subscriptionService.unsubscribeByAclUID

Removes the account's subscription from the item..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------------|--------|-------------------------------------|
| <code>aclUID</code> | Long | UID of the ACL entry to unsubscribe |
| <code>smartcardID</code> | String | ID of the smartcard if exists |

Return value

Type:

Integer - Number of unsubscribed items

Example

```
<unsubscribeByAclUID>
  <aclUID>9</aclUID>
  <smartcardID>ABC-88806</smartcardID>
</unsubscribeByAclUID>
```

84.4.7 subscriptionService.unsubscribeLocaleBased

Removes the account's subscription from a local based item..

Available through Web Services.

Parameters

| Name | Type | Description |
|---|---|---|
| <code>accountUID</code> | Long | UID of the account |
| <code>locale</code> <code>SubscribableModel</code> | tv.quative.service.subscription.model.LocaleSubscribableModel | Locale based model object of the item to subscribe to |
| <code>smartcardID</code> | String | ID of the smartcard if exists |

Return value

Type:

Integer - Number of unsubscribed items

85 targetAreaService

This is a primitive *Service* which provides the *Target Area* Entity.

85.1 Target Area

A *Target Area* represents a set of postal codes which are treated as a unit for geographically-targeted advertising.

Figure 27 TargetArea and associated Entities.

Target Areas may overlap; that is, the same postal code may be included in several *Target Areas*.

Each *Target Area* is associated with a set of advertisement *Assets* which are to be provided to viewers in that area.

85.1.1 TargetArea

| Name | Type | Description |
|-----------------------------|--|---|
| <code>assetList</code> | <code>List<Long></code> | The list of Asset UID's associated with the Target Area |
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> <code>List<String></code> | The change history. |
| <code>creationDate</code> | <code>Date</code> | The date created. |
| <code>description</code> | <code>String</code> | The short description of the model element. |
| <code>exportID</code> | <code>Integer</code> | The id to export object to third part systems. |
| <code>locale</code> | <code>String</code> | The locale description of the language. |
| <code>modifiedDate</code> | <code>Date</code> | The date last modified and persisted. |
| <code>modifiedDateML</code> | <code>Date</code> | The date last modified and persisted. |
| <code>name</code> | <i>Search criterion. Quick Search criterion.</i> <code>String</code> | The name of the model element. |
| <code>originID</code> | <code>Integer</code> | The uid of the origin system (3rd part system). |

| Name | Type | Description |
|--------------------------------|---|---|
| <code>originIdAndKey</code> | <i>Read only.</i> <code>String</code> | A unique reference for the object derived by concatenating the <code>originID</code> and <code>originKey</code> , together, with a pipe character to separate them. |
| <code>originKey</code> | <code>String</code> | The key of the model object in the origin system (3rd part system). |
| <code>postCodeList</code> | <code>List<String></code> | The list of Post Codes associated with the Target Area |
| <code>serviceProviderID</code> | <code>Long</code> | The Service provider UID. |
| <code>shortCode</code> | <i>Quick Search criterion.</i> <code>String</code> | The short code for a targeting area |
| <code>uID</code> | <code>Long</code> | The unique identifier. |

85.2 targetAreaService

Interface for the primitive Target Area Service. This service encapsulates all the CRUD database functionality needed to create, delete and update target area entities. It also encompasses the functionality needed to associate or disassociate assets and post codes with individual target areas. The services are made visible to HUE or to WebServices using annotation in the Service Implementation class. For example: For the ServiceClass: `@QubeService(name="targetArea ServiceImpl", hue=true, ws=true)` For the Methods: `@QubeMethod(hue=false, ws=true)`

mdouthwaite @author

85.2.1 targetAreaService.addMultiLanguage

Create (persist) multi language fields.

The record is identified by uid and locale.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The LocaleBaseModel to add multilanguage |

Return value

Type:

The UID of the created entity

85.2.2 targetAreaService.associateAsset

Associates targeting Area to advert asset entry for the supplied values..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/associateAsset`

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|---------------------------------------|
| targetingAreaUid | Long | The targeting area unique identifier. |
| assetUid | Long | The advert asset unique identifier. |

Return value

Type: void

Example

```
<associateAsset>
  <targetingAreaUid>63</targetingAreaUid>
  <assetUid>24</assetUid>
</associateAsset>
```

85.2.3 targetAreaService.associateAssetList

Associates targeting Area to advert List if asset entry for the supplied values.

The asset list is comma separated list

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|---------------------------------------|
| targetingAreaUid | Long | The targeting area unique identifier. |

| Name | Type | Description |
|------------------------|------------|---------------------------|
| <code>assetUids</code> | List<Long> | list of advert asset uids |

Return value

Type: void

85.2.4 targetAreaService.associateAssetList

Associates targeting Area to advert List if asset entry for the supplied values.

The asset list is comma separated list

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/associateAssetList`

Parameters

| Name | Type | Description |
|-------------------------------|--------|--|
| <code>targetingAreaUId</code> | Long | The targeting area unique identifier. |
| <code>assetUids</code> | String | comma separated list of advert asset uids. |

Return value

Type: void

85.2.5 targetAreaService.associatePostCode

Associates targeting Area to PostCode entry for the supplied values..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/associatePostCode`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|--------|---------------------------------------|
| <code>targetingAreaUId</code> | Long | The targeting area unique identifier. |
| <code>postCode</code> | String | The postcode string literals. |

Return value

Type: void

Example

```
<associatePostCode>
  <targetingAreaUid>76</targetingAreaUid>
  <postCode>stringValue</postCode>
</associatePostCode>
```

85.2.6 targetAreaService.associatePostCodeList

Associates targeting Area to list of PostCode entry for the supplied values.

The post code list is comma separated list.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|--------------|---------------------------------------|
| targetingAreaUid | Long | The targeting area unique identifier. |
| postCodes | List<String> | list of post code literals. |

Return value

Type: void

85.2.7 targetAreaService.associatePostCodeList

Associates targeting Area to list of PostCode entry for the supplied values.

The post code list is comma separated list.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/associatePostCodeList>

Parameters

| Name | Type | Description |
|------------------|--------|---------------------------------------|
| targetingAreaUid | Long | The targeting area unique identifier. |
| postCodes | String | comma separated list of postCodes. |

Return value

Type: void

85.2.8 targetAreaService.create

Create (persist) a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to persist. |

Return value

Type:

UID of the created object.

85.2.9 targetAreaService.delete

Delete, from storage, a `LocaleBaseModel..`

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel</code> | The <code>LocaleBaseModel</code> to delete |

Return value

Type:

the number of rows deleted.

85.2.10 targetAreaService.disassociateAsset

Disassociate an advert asset entry from a target area..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/disassociateAsset`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|------|---------------------------------------|
| <code>targetingAreaUid</code> | Long | The targeting area unique identifier. |
| <code>assetUid</code> | Long | The advert asset unique identifier. |

Return value

Type: void

Example

```
<disassociateAsset>
<targetingAreaUid>33</targetingAreaUid>
<assetUid>50</assetUid>
</disassociateAsset>
```

85.2.11 targetAreaService.disassociateAssetList

Disassociate a list of asset entries from a target area.

The asset list is comma separated list

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|------------|---------------------------------------|
| <code>targetingAreaUid</code> | Long | The targeting area unique identifier. |
| <code>assetUids</code> | List<Long> | list of asset Uids. |

Return value

Type: void

85.2.12 targetAreaService.disassociatePostCode

Disassociate a post code from a targeting area..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/disassociatePostCode`

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|--------|---|
| <code>targetingAreaUid</code> | Long | The targeting area's unique identifier. |
| <code>postCode</code> | String | The post code string literals. |

Return value

Type: void

Example

```
<disassociatePostCode>
<targetingAreaUid>63</targetingAreaUid>
<postCode>stringValue</postCode>
</disassociatePostCode>
```

85.2.13 targetAreaService.disassociatePostCodeList

Disassociate a list of post codes entry from a target area.

The post code list is a comma separated list.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------------|--------------|---------------------------------------|
| <code>targetingAreaUid</code> | Long | The targeting area unique identifier. |
| <code>postCodes</code> | List<String> | list of post code literals. |

Return value

Type: void

85.2.14 targetAreaService.disassociatePostCodeList

Disassociate a list of post codes entry from a target area.

The post code list is a comma separated list.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/disassociatePostCodeList>

Parameters

| Name | Type | Description |
|------------------|--------|---|
| targetingAreaUid | Long | The targeting area unique identifier. |
| postCodes | String | comma separated list of post code literals. |

Return value

Type: void

85.2.15 targetAreaService.getAssetUids

Return a list of all the asset uids associated with a list of target areas.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------------|-------------|
| targetingAreaIds | List<Long> | |

Return value

Type: List<Long>

List<Long> The list of advert asset uids associated with the target areas.

85.2.16 targetAreaService.getByAssetUid

Gets List of targeting Area for the supplied advert asset uid..

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/getByAssetUid>

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|--------|--|
| <code>assetUid</code> | Long | The advert asset area unique identifier. |
| <code>locale</code> | String | |

Return value

Type: `List<TargetArea>`

List of TargetingArea objects.

Example

```
<getByAssetUid>
<assetUid>66</assetUid>
<locale>en_us</locale>
</getByAssetUid>
```

85.2.17 targetAreaService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |
| <code>locale</code> | String | String reference for the language |

Return value

Type:

LocaleBaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
<originKey>stringValue</originKey>
<originID>62</originID>
```

```
<locale>en_us</locale>
</getByOrigin>
```

85.2.18 targetAreaService.getByPostCode

Gets List of targeting Area for the supplied post code string literal..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/getByPostCode`

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------|--------|-------------|
| <code>postCode</code> | String | |
| <code>locale</code> | String | |

Return value

Type: `List<TargetArea>`

List of TargetingArea objects.

Example

```
<getByPostCode>
<postCode>stringValue</postCode>
<locale>en_us</locale>
</getByPostCode>
```

85.2.19 targetAreaService.getByUID

Retrieve, from storage, the `LocaleBaseModel` with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|---|
| <code>uid</code> | Long | Integer unique id for the <code>LocaleBaseModel</code> object |

| Name | Type | Description |
|--------|--------|-----------------------------------|
| locale | String | String reference for the language |

Return value

Type:

LocaleBaseModel if found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>3</uid>
  <locale>en_us</locale>
</getByUID>
```

85.2.20 targetAreaService.getMultiLanguageList

Retrieve all the multilanguage data for an specific Unique id..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--------------------------------|
| uid | Long | Integer unique id of the Model |

Return value

Type: List<M>

List LocaleBaseModel with the specific locale dependent data.

Example

```
<getMultiLanguageList>
  <uid>72</uid>
</getMultiLanguageList>
```

85.2.21 targetAreaService.getPostCodes

Retrieves a list of all the post codes associated with a given target area uid.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/targetAreaService/getPostCodes>

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------|------|-------------|
| targetAreaUid | Long | |

Return value

Type: `List<String>`

list of post codes

Example

```
<getPostCodes>
  <targetAreaUid>6</targetAreaUid>
</getPostCodes>
```

85.2.22 targetAreaService.modifyMultiLanguage

Update (persist) multi language fields.

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|---|
| model | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to modify multilanguage |

Return value

Type:

The number of rowa updated

85.2.23 targetAreaService.removeMultiLanguage

Delete a multi language entry .

The record is identified by uid and locale

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|---|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete multilanguage |

Return value

Type:

The number of rows deleted

85.2.24 targetAreaService.update

Update, to storage, an existing `LocaleBaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

85.2.25 targetAreaService.updateModelAndML

Method to update the business entity along with its locale information..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------------------------|
| <code>model</code> | LocaleBaseModel tv. quative.core.persistence. model.LocaleBaseModel | The LocaleBaseModel to delete |

Return value

Type:

the number of rows updated.

86 tunerService

This is a primitive *Service* which provides the *Tuner* Entity.

86.1 tv.quative.service.tuner.model.Tuner

| Name | Type | Description |
|--------------------------------|---|--|
| <code>changeLog</code> | <i>Deprecated. Not visible to HUE.</i> List<String> | The change history. |
| <code>clientPort</code> | Integer | When tuned, the network (UDP or TCP) port on the STB to which subsequent announcements should be sent. This varies per STB session at the discretion of the STB. |
| <code>connected</code> | boolean | |
| <code>creationDate</code> | Date | The date created. |
| <code>exportID</code> | Integer | The id to export object to third part systems. |
| <code>modifiedDate</code> | Date | The date last modified and persisted. |
| <code>originID</code> | Integer | The uid of the origin system (3rd part system). |
| <code>originIdAndKey</code> | String | |
| <code>originKey</code> | String | The key of the model object in the origin system (3rd part system). |
| <code>rtspSessionUID</code> | Long | When tuned, the foreign key to the RTSPSession (from the ERM) to which we are tuned. |
| <code>serviceProviderID</code> | Long | The Service provider UID. |
| <code>setTopBoxUID</code> | Long | Foreign key to the owning STB. |
| <code>stbSessionID</code> | String | When tuned, the RTSP session ID of the session with the STB. Different from the session ID of the RTSPSession - that's the session ID we share with the ERM. The |

| Name | Type | Description |
|--------------|---------|---|
| | | separation is required both for security reasons, and also because, by dfn. of the Telenet STB spec, Teardowns identify the STB by session ID, and do not supply the client ID. |
| tunerIDInSTB | Integer | The ID of this tuner within the STB. Usually 1 - 4 or so. |
| uID | Long | The unique identifier. |

86.2 tunerService

wdickson @author

86.2.1 tunerService.countByRtspSessionUID

Count the tuner objects which are tuned to the specified RTSP session (ERM)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|--|
| uid | Long | The primary key of the RTSPSession object. |

Return value

Type:

How many tuners are tuned to this session.

Example

```
<countByRtspSessionUID>
  <uid>37</uid>
</countByRtspSessionUID>
```

86.2.2 tunerService.create

Create (persist) a **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to persist. |

Return value

Type:

UID of the created object.

86.2.3 tunerService.create

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------|
| <code>tuner</code> | tv.quative.service.tuner.model.Tuner ^{p.892} | |

Return value

Type:

86.2.4 tunerService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows deleted.

86.2.5 tunerService.delete

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|-------------|
| tuner | tv.quative.service.tuner.model.Tuner ^{p.892} | |

Return value

Type:

86.2.6 tunerService.getActiveBySetTopBoxUID

Obtain the list of tuners registered to a given STB, which are currently active (connected to a session)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | The primary key of the STB object. |

Return value

Type: `List<Tuner>`

The tuners for that STB. Typically there will only be a handful.

Example

```
<getActiveBySetTopBoxUID>
  <uid>81</uid>
</getActiveBySetTopBoxUID>
```

86.2.7 tunerService.getActiveOlderThan

Obtain the list of tuner objects which are active (ie.

tuned to a session) and were last modified prior to the specified cutoff date. Such tuner objects are ripe to be de-allocated since they haven't been sending keep-alives (and can safely be assumed to be dead).

Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------|---------|--|
| <code>cutoff</code> | Date | The cutoff date. |
| <code>n</code> | Integer | The max number of rows to return. Need this to provide an upper bound on our memory usage. |

Return value

Type: `List<Tuner>`

The list of zombie tuners.

86.2.8 tunerService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>10</originID>
</getByOrigin>
```

86.2.9 tunerService.getByRtspSessionUID

Obtain the list of all tuner objects which are tuned to the specified RTSP session (ERM)..

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------|------|--|
| <code>uid</code> | Long | The primary key of the RTSPSession object. |

Return value

Type: `List<Tuner>`

The tuners which are tuned to this session. We assume that we won't need to page this - very popular channels will be static rather than switched.

Example

```
<getByRtspSessionUID>
  <uid>84</uid>
</getByRtspSessionUID>
```

86.2.10 tunerService.getBySetTopBoxSessionID

Obtain the Tuner, if any, which is mapped by the specified session ID.

This session ID is generated by us and is (with probability unity - epsilon) unique.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|--------|---------------------------|
| <code>sessionID</code> | String | The session to check for. |

Return value

Type:

The Tuner, or null if there's no such tuner. (This might represent a session-guessing attack; in future, we might want to blacklist STBs which generate a lot of duff session IDs. However, such an attack has a vanishingly small probability of succeeding on any timescale we need to care about.

Example

```
<getBySetTopBoxSessionID>
  <sessionID>ABC-26020</sessionID>
</getBySetTopBoxSessionID>
```

86.2.11 tunerService.getBySetTopBoxUID

Obtain the list of tuners registered to a given STB..

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | The primary key of the STB object. |

Return value

Type: List<Tuner>

The tuners for that STB. Typically there will only be a handful.

Example

```
<getBySetTopBoxUID>
  <uid>60</uid>
</getBySetTopBoxUID>
```

86.2.12 tunerService.getByUID

Retrieve, from storage, the **BaseModel** with the unique id 'uid'.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------------|
| uid | Long | Unique id for the BaseModel object |

Return value

Type:

BaseModel If found, the matching object, null otherwise.

Example

```
<getByUID>
  <uid>77</uid>
</getByUID>
```

86.2.13 tunerService.update

Update, to storage, an existing `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

86.2.14 tunerService.update

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|---|-------------|
| <code>tuner</code> | tv.quative.service.tuner.model.Tuner ^{p.892} | |

Return value

Type:

87 userManagementService

This *Service* provides methods which allow set-top box clients to work with *User* Entities.

Clients are restricted in what they can do with a *User* Entity for security reasons. The current version of SDP only permits clients to change the PIN of a *User*.

Note

Clients are expected to verify that the user who initiates this operation has the appropriate privileges. For example, a deployment may mandate that only *Users* with superuser access can change the PIN of another *User*. This must be implemented on the client side, because SDP does not know which user is active: clients sign on by *Account*, and do not notify SDP when the user changes.

87.1 userManagementService

User Management Service Implementation. Implements methods available for managing User through User service.

David Hendy @author

87.1.1 userManagementService.changePinByAccNumAndUserName

Change the PIN of a *User* identified by username

Because usernames are only unique to their *Account*, the *Account* must also be identified, either from the request context or explicitly.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/changePinByAccNumAndUserName>

Parameters

| Name | Type | Description |
|----------------------------|--------|--|
| <code>accountNumber</code> | String | The <code>uid</code> of the <i>Account</i> hosting the <i>User</i> , if called through Web Services. HUE clients should not supply this parameter, as the API will use the signed-on <i>Account</i> ; the specified <i>User</i> must belong to this <i>Account</i> . |
| <code>userName</code> | String | The <code>name</code> of the <i>User</i> to be updated. |
| <code>newPIN</code> | String | The new PIN for the <i>User</i> . |

Return value

Type:

The number of *User* Entities updated: 1 for success, 0 for failure.

87.1.2 userManagementService.changePinByUserId

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/changePinByUserId`

Parameters

| Name | Type | Description |
|---------------------|--------|--|
| <code>userId</code> | Long | The <i>uid</i> of the <i>User</i> to be updated. |
| <code>newPIN</code> | String | The new PIN for the <i>User</i> . |

Return value

Type:

The number of *User* Entities updated: 1 for success, 0 for failure.

87.1.3 userManagementService.createUser

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/createUser`

Parameters

| Name | Type | Description |
|------------------------|--------|-------------|
| <code>name</code> | String | |
| <code>userLogin</code> | String | |
| <code>password</code> | String | |
| <code>type</code> | String | |
| <code>dob</code> | Date | |
| <code>ratingId</code> | Long | |

Return value

Type:

87.1.4 userManagementService.removeUser

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/removeUser`

Parameters

| Name | Type | Description |
|--------------|------|-------------|
| targetUserId | Long | |

Return value

Type:

87.1.5 userManagementService.updateDateOfBirth

Update a users date of birth..

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/updateDateOfBirth`

Parameters

| Name | Type | Description |
|--------|------|-----------------------|
| userId | Long | the user to update |
| newDob | Date | the new date of birth |

Return value

Type:

number of rows updated.

87.1.6 userManagementService.updateLogin

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/updateLogin`

Parameters

| Name | Type | Description |
|--------------|------|-------------|
| targetUserId | Long | |

| Name | Type | Description |
|------------|--------|-------------|
| newLoginId | String | |

Return value

Type:

87.1.7 userManagementService.updatePassword

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/updatePassword`

Parameters

| Name | Type | Description |
|--------------|--------|-------------|
| targetUserId | Long | |
| newPassword | String | |

Return value

Type:

87.1.8 userManagementService.updatePurchaseAbility

Update whether a *User* is permitted to make their own purchases.

To use this method, the client must have identified a *User* at sign-on which is a superuser of its *Account*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/updatePurchaseAbility`

Parameters

| Name | Type | Description |
|--------------------|--------|---|
| targetUserId | Long | The <i>uid</i> of the <i>User</i> to update. This <i>User</i> must belong to the same <i>Account</i> as the signed-on <i>User</i> . |
| newPurchaseAbility | String | The new purchase ability code; one of A (ALLOWED) or D (DENIED). |

Return value

Type:

The number of *Users* updated.

87.1.9 userManagementService.updateRatingId

Update the *Rating* applied to a *User*.

To use this method, the client must have identified a *User* at sign-on.

- ▶ If the *User* is a superuser of its *Account*, any *User* of the *Account* may be updated, and any *Rating* may be specified.
- ▶ Otherwise, only the signed-on *User* may be updated, and only *Ratings* with an equal or lower `precedenceValue` to the *Rating* of the signed-on *User* may be specified. (A lower `precedenceValue` implies a more restrictive *Rating*.)

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/updateRatingId`

Parameters

| Name | Type | Description |
|---------------------------|--------|---|
| <code>targetUserId</code> | Long | The <code>uid</code> of the <i>User</i> to update. |
| <code>newRatingId</code> | Long | The <code>uid</code> of the <i>Rating</i> to apply. |
| <code>locale</code> | String | The current locale. |

Return value

Type:

The number of rows updated.

87.1.10 userManagementService.updateType

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userManagementService/updateType`

Parameters

| Name | Type | Description |
|---------------------------|------|-------------|
| <code>targetUserId</code> | Long | |

| Name | Type | Description |
|----------------------|--------|-------------|
| <code>newType</code> | String | |

Return value

Type:

88 userService

This is a primitive *Service* which provides the *User* Entity.

88.1 userService

Provides all of the required functionality to Create, Retrieve, Update and Delete 'User' accounts on the system.

Steve @author

Miguel @author

Naeem Bilal (NMPS) @author

88.1.1 userService.create

Creates a new *User*.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|---|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | A model object containing all the fields of the <i>User</i> , |

Return value

Type:

The `uid` of the created *User*.

88.1.2 userService.delete

Delete, from storage, a `BaseModel`..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--------------------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The <code>BaseModel</code> to delete |

Return value

Type:
the number of rows deleted.

88.1.3 userService.getByAcctUID

Gets all the *Users* associated with the specified *Account*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userService/getByAcctUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|------|---------------------------------|
| <code>acctuid</code> | Long | The uid of the <i>Account</i> . |

Return value

Type: `List<User>`

A list of matching *User* objects (may be empty.)

Example

```
<getByAcctUID>
  <acctuid>74</acctuid>
</getByAcctUID>
```

88.1.4 userService.getByOrigin

Retrieve all the models related to the external origin uid and origin key.

Both makes a unique composite key.

Available through Web Services.

Parameters

| Name | Type | Description |
|------------------------|---------|---|
| <code>originKey</code> | String | String the primary key of the record in a customer system |
| <code>originID</code> | Integer | Integer unique origin id for customer representation |

Return value

Type:

BaseModel for this external origin uid and origin key.

Example

```
<getByOrigin>
  <originKey>stringValue</originKey>
  <originID>31</originID>
</getByOrigin>
```

88.1.5 userService.getByOriginUID

Retrieve all the models related to the external origin uid.

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------|---------|--|
| originUID | Integer | Integer unique origin id for customer representation |

Return value

Type: List<User>

List All the BaseModels for this external origin uid.

Example

```
<getByOriginUID>
  <originUID>56</originUID>
</getByOriginUID>
```

88.1.6 userService.getByUID

Gets the **User** with the specified **uid**.

Available through Web Services.

Parameters

| Name | Type | Description |
|------|------|------------------------------|
| uid | Long | The uid of the User . |

Return value

Type:

The matching **User**, or null if none is found.

Example

```
<getByUID>
  <uid>27</uid>
</getByUID>
```

88.1.7 userService.getDefaultUserByAcctUID

Gets the default *User* of the specified *Account*.

This call will always return a *User*, because all *Accounts* must have a single default *User*.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userService/getDefaultUserByAcctUID`

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|------|---------------------------------|
| acctUID | Long | The uid of the <i>Account</i> . |

Return value

Type:

Example

```
<getDefaultUserByAcctUID>
  <acctUID>19</acctUID>
</getDefaultUserByAcctUID>
```

88.1.8 userService.getUserPIN *New*

Gets the **UserPIN** associated with default user of the specified **Account**.

Available through HUE at `http://<server>:<port>/hue-gateway/gateway/http/js/userService/getUserPIN`

This API should be called with a valid token consisting of a valid user id of the default user.

Return value

Type: `string`

A list of matching **User** objects (may be empty.)

Example

```
<getUserPIN>  
  <userPIN>74</userPIN>  
</getUserPIN>
```

88.1.9 userService.update

Update, to storage, an existing **BaseModel**..

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|-------------------------|
| <code>model</code> | <code>tv.quative.core.persistence.model.BaseModel</code> | The BaseModel to delete |

Return value

Type:

the number of rows updated.

89 visitorService *Updated*

Visitor Service

The visitor may be used as a generic mechanism to add shared properties to a host table. It is loosely inspired by the Go F visitor pattern.

For example suppose that COR_ACCOUNT requires a field whose value is shared between many records. Rather than creating dedicated tables that contain profile information for COR_ACCOUNT (and then the same again for say, COR_DEVICE), we can instead link to a VISITOR table that holds the shared information.

Goals

- ▶ Intended for write rarely, read heavy data.
- ▶ Simplicity
- ▶ Reuse
- ▶ The impact on SDP is to be as low as possible.
- ▶ Code clarity

Concepts

There is a host table and a visitor table. Any table could be the host table and its job is to host a property that may contain one of several values. Typically, the host property is shared between a large number of records.

This job of the Visitor table is hold values that are referenced by the value in the host table. This allows for updates to be made to the visitor table in a single place and the changes will be indirectly reflected in the host table.

Static View

Dynamic View

You must pass the host table name, the host property and the SPID in addition to value that you wish to dereference. This has the following benefits.

- ▶ Keeps things organised.
- ▶ Allow for the possibility that several tables might share the same property.
- ▶ Will aid traceability of the data.

To use the visitor service you must first get the value of the property in the host table and then look up that value in the visitor table.

Standalone Value

A single value may be returned.

Grouped Values

Groups of values may be returned in one hit.

Consequences

1. A caching mechanism is required on the client side to prevent overload.
2. The service is intended to be exposed as a Web Service which means that a Map structure is not supported. The client must manipulate a list returned by the Web Service. See helper code below.

Uses

- To implement profile data shared between many records.

Example Code - JAXWS

Standalone Value

```
public int getMaxRecordingQuotaHoursFromSDP( final Long spid, final String account
Number) throws QuotaUsageServiceException {
    Account account;
    try {
        account = accountService.getByAccountNumber(accountNumber);
        String indirectValue = account.getMaxRecordingQuotaHours();
        if (indirectValue == null) {
            throw new QuotaUsageServiceException( "Cannot find a max recording quota for the
account: " + accountNumber);
        }
        return visitorService.getValueAsInt( "COR_ACCOUNT" , "MAX_REC_QUOTA_HRS" , spid,
indirectValue);
    } catch (Exception e) {
        throw new QuotaUsageServiceException(e.getMessage(), e);
    }
}
```

Grouped Values

```
/**
 * Ask the SDP for the maximum number of hours available to the account for recording.
 * Not exposed RESTfully - intended for usage within Locker.
 * @param accountNumber The ID of the account.
 * @return The maximum number of hours available to the account for recording.
 * @throws QuotaUsageServiceException If we are unable to obtain the figure from the
SDP.
 */
private int getMaxRecordingQuotaHoursFromSDP(final Long spid, final String account
Number) throws QuotaUsageServiceException {
    Account account;
    try {
        account = accountService.getByAccountNumberAndServiceProviderID(accountNumber, spid);
    } catch (Exception e) {
        throw new QuotaUsageServiceException("Problem looking up account number", e);
    }
    if (account == null) {
```



```

        throw new QuotaUsageServiceException("Account not found for account number : " +
accountNumber);
    }
    String profileValue = account.getNpvrProfile();
    Assertions.checkInternalError(profileValue == null, MessageFormat.format("Account
Number [{0}] does not have a profile.", account.getAccountNumber()));
    try {
        int maxQuota = visitorHelper.findGroupPropertyAsInt("COR_ACCOUNT", "NPVR_PROFILE",
profileValue, spid, "MAX_REC_QUOTA_HRS");
        return maxQuota;
    } catch (Exception e) {
        throw new QuotaUsageServiceException("Problem getting account quota for account
number : " + accountNumber, e);
    }
}
private long getMaxRecordingQuotaSecondsFromSDP(final Long spid, final String account
Number) throws QuotaUsageServiceException {
    return getMaxRecordingQuotaHoursFromSDP(spid, accountNumber) * 3600L;
}

```

Helper Code

```

package com.nagra.npvrlocker.client.visitor;
import java.text.MessageFormat;
import java.util.List;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import tv.quative.visitorservicenamespace.ServiceException_Exception;
import tv.quative.visitorservicenamespace.VisitorPair;
import tv.quative.visitorservicenamespace.VisitorService;
import com.google.inject.Inject;
import com.google.inject.Singleton;
import com.nagra.npvrlocker.exception.InternalServerErrorException;
/**
 * The visitor service that allows read-rarely data to be added to the SDP with minimal
impact and retrieved
 * dynamically using this API.
 *
 * Data of various types may be arranged into groups and associated with a key that is be
used to access the group.
 *
 * The ideal data structure for such a group would be a 'Map', but since WS does not
support Maps, we will have to
 * make do with a List instead.
 *
 * This class searches sequentially though the list to access the required property. Not
ideal, but since the
 * underlying service is cached, it's still *much* faster than going to the SDP every
time.
 *
 */

```

```

* @author lewis
*
*/
@Singleton
public class VisitorHelper {
    private static final Logger LOG = LoggerFactory.getLogger(VisitorHelper.class);
    @Inject private VisitorService visitorService;
    /**
     * @param hostName
     * @param hostProperty
     * @param hostValue
     * @param spid
     * @param groupProperty
     * @return
     */
    public Object findGroupProperty(String hostName, String hostProperty, String hostValue,
    Long spid, String groupProperty) {
        try {
            List<VisitorPair> pairs = visitorService.getGroup(hostName, hostProperty, hostValue,
            spid);
            for (VisitorPair pair : pairs) {
                if (pair.getName().equals(groupProperty)) {
                    return pair.getValue();
                }
            }
        } catch (ServiceException e) {
            String text = MessageFormat.format("Could not get Visitor Group for HostName[{0}],
            HostProperty[{1}], HostValue(Group)[{2}], spid [{3}]",
            hostName, hostProperty, hostValue, spid);
            LOG.debug(text);
            throw new InternalServerErrorException(text);
        }
        String text = MessageFormat.format("Could not get property [{0}] from Visitor Group for
            HostName[{1}], HostProperty[{2}], HostValue(Group)[{3}], spid [{4}]",
            groupProperty, hostName, hostProperty, hostValue, spid);
        LOG.debug(text);
        throw new InternalServerErrorException(text);
    }
    public Boolean findGroupPropertyAsBoolean(String hostName, String hostProperty, String
    hostValue, Long spid, String groupProperty) {
        return (Boolean) findGroupProperty(hostName, hostProperty, hostValue, spid, group
        Property);
    }
    public Integer findGroupPropertyAsInteger(String hostName, String hostProperty, String
    hostValue, Long spid, String groupProperty) {
        return (Integer) findGroupProperty(hostName, hostProperty, hostValue, spid, group
        Property);
    }
    public int findGroupPropertyAsInt(String hostName, String hostProperty, String host
    Value, Long spid, String groupProperty) {

```

```
return findGroupPropertyAsInteger(hostName, hostProperty, hostValue, spid, group
Property).intValue();
}
}
```

Example Code - HUE

HUE Methods:

```
VisitorService.prototype.getGroup = function(jsCaller, jsSuccessCallback, jsFailure
Callback, hostName, hostProperty, grp, spid) {
    this.invokeMethod(jsCaller, jsSuccessCallback, jsFailureCallback, "getGroup",
    hostName, hostProperty, grp, spid);
};
```

HUE call:

```
cwm-sdp-vdev12:8180/qsp/gateway/http/js/VisitorService/getGroup?arg0=COR_
ACCOUNT&arg1=NPVR_PROFILE&arg2=NPVR_SMALL&arg3=3&token=AAABRbbuAIuQB0uBEQNzn3h4jyN0dqo
OFPuQmtv/vawvFaK59pcB58vD3HMU1Zxwbu7zG/rPQsLfYHTF8PExQQ==
```

89.1 visitorService

Visitor Service Implementation.

A Visitor is some data that may be shared with a generic table, and is therefore suitable for creating generic profile data relating to a particular table.

89.1.1 visitorService.create

Creates a new Visitor.

Available through Web Services.

Parameters

| Name | Type | Description |
|-------|---|--|
| model | tv.quative.service.visitor. api.VisitorService | A model object containing all the fields of the Visitor |

Return value

Type:

The `uid` of the created **Visitor**.

89.1.2 visitorService.update

Updates a Visitor.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------------|--|--|
| <code>model</code> | <code>tv.quative.service.visitor.api.VisitorService</code> | A model object containing all the fields of the Visitor |

89.1.3 visitorService.getGroup

Obtains a group of values from a Visitor. Available through Web Services.

Parameters

| Name | Type | Description |
|---------------------------|--------|----------------------------|
| <code>hostName</code> | String | The name of the host table |
| <code>hostProperty</code> | String | The name of the host field |
| <code>grp</code> | String | The name of the group |
| <code>spid</code> | Long | A service provider id |

Example

Suppose that the host table is `COR_ACCOUNT` and that it contains a host property called `NPVR_PROFILE` with value `NPVR_UNLIMITED`

```
Insert into COR_ACCOUNT (ACC_UID,ACC_NUM,ACC_SPID,ACC_STATUS,PPV_STATUS,CREATION_DATE,
ACCP_UID,NAME_TITLE,INITIALS_FORENAME,LAST_NAME,ADDRESS_1,ADDRESS_2,LOCALITY,POST_TOWN,
COUNTY,POST_CODE,COUNTRY,ACC_PWD,MODIFIED_DATE,ORIGIN_UID,ORIGIN_KEY,EXPORT_KEY,CREDIT_
LIMIT,CREDIT_SPENT,CREDIT_SPENT_RST,ROLLPROF_UID,MAX_MPDEV_ALLOWED,MAX_USER_ALLOWED,ACC_
PROFILE_UID,NPVR_PROFILE) values (552,'910-5754355',1,'I','0',to_date('09-JUN-14','DD-
```

```
MON-RR'),null,'?','?','?','P','?','?','?','?','?','?','?','?','to_date('09-JUN-14','DD-MON-RR'),1,'910-5754355',null,null,null,'?',null,null,null,null,'NPVR_UNLIMITED');
```

We can attach a group of visitor data, NPVR_UNLIMITED to this table that is filtered by SPID, 3 and retrieve the group of data by issuing the following API call.

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:vis="http://quative.tv/VisitorServiceNamespace">
  <soapenv:Header/>
  <soapenv:Body>
    <vis:getGroup>
      <!--Optional:-->
      <hostName>COR_ACCOUNT</hostName>
      <!--Optional:-->
      <hostProperty>NPVR_PROFILE</hostProperty>
      <!--Optional:-->
      <grp>NPVR_UNLIMITED</grp>
      <!--Optional:-->
      <spid>3</spid>
    </vis:getGroup>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <rqid:requestID xmlns:rqid="http://quative.tv/requestId">471504209</rqid:requestID>
  </env:Header>
  <env:Body>
    <ns2:getGroupResponse xmlns:ns2="http://quative.tv/VisitorServiceNamespace">
      <return>
        <name>NPVR_STATUS</name>
        <value xsi:type="xs:string" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xs="http://www.w3.org/2001/XMLSchema">ENABLED</value>
      </return>
      <return>
        <name>MAX_REC_QUOTA_HRS</name>
        <value xsi:type="xs:int" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xs="http://www.w3.org/2001/XMLSchema">999999999</value>
      </return>
    </ns2:getGroupResponse>
  </env:Body>
</env:Envelope>
```

Please note that there is currently no way to add groups of visitor data other than by direct table manipulation or via DLM.

89.1.4 visitorService.getValueAsInt

Obtains a property value from a Visitor.

Available through Web Services.

Parameters

| Name | Type | Description |
|--------------|--------|--|
| hostName | String | The name of the host table |
| hostProperty | String | The name of the host field |
| spid | Long | A service provider id |
| name | String | The name of the property for which the value will be returned. |

90 workflowSubscriptionService

This *Service* provides high-level functions required to manage subscriptions in SDP and connected systems.

It is used by the *Create* and *Modify Subscription* workflows in QSP Manager, and by subscriber management systems over Web Services.

90.1 workflowSubscriptionService

dhanumandla @author

Naeem Bilal @author

Miguel Gil (BNET Conax Cas) @author

Ashish Gupta @author

Will Dickson @author

Ben Beazley (re-factoring and clean up for BNET SmartCard multiplicity) @author

90.1.1 workflowSubscriptionService.createAccountUserSubscription

Aggregate strongly typed method to create a new subscription with account and its users..

Available through Web Services.

Parameters

| Name | Type | Description |
|----------------------|-------------------------------|--|
| <code>account</code> | <code>Account</code> | Account object to be created |
| <code>users</code> | <code>List<User></code> | List of User objects that are to be associated to an account |

Return value

Type:

Long The UID of the created account

90.1.2 workflowSubscriptionService.createNewSubscription

Available through Web Services.

Parameters

| Name | Type | Description |
|------------|-----------------|-------------|
| account | <u>Account</u> | |
| users | List<User> | |
| profiles | List<Profile> | |
| stbs | List<SetTopBox> | |
| packages | List<Package> | |
| catalogues | List<Catalogue> | |

Return value

Type:

90.1.3 workflowSubscriptionService.createSmartCardSubscription

Creates a fully-populated *Account* for an end user, including device provisioning and initial subscriptions.

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/workflowSubscriptionService/createSmartCardSubscription>

Available through Web Services.

Parameters

| Name | Type | Description |
|---------|-----------------|---|
| account | <u>Account</u> | A model object containing all the fields of the <i>Account</i> to be created. |
| users | List<User> | A list of model objects containing all the fields of each <i>User</i> to be associated with the <i>Account</i> . If a <i>User</i> with a supplied <i>uid</i> does not already exist, it will be created. |
| stbs | List<SetTopBox> | A list of model objects containing all the fields of each <i>Set Top Box</i> to be provisioned to the <i>Account</i> . If a <i>Set Top Box</i> with a supplied <i>uid</i> does not already exist, it will be created. |

| Name | Type | Description |
|---------------------------|--|---|
| <code>profiles</code> | <code>List<ProfileSmartcardSubscribableModel></code> | The <i>Profiles</i> to subscribe, and (optionally) the devices to subscribe to them. |
| <code>packages</code> | <code>List<PackageSmartcardSubscribableModel></code> | The <i>Packages</i> to subscribe, and (optionally) the devices to subscribe to them. |
| <code>catalogues</code> | <code>List<CatalogueSmartcardSubscribableModel></code> | The <i>Catalogues</i> to subscribe, and (optionally) the devices to subscribe to them. |
| <code>policyGroups</code> | <code>List<PolicyGroupSmartcardSubscribableModel></code> | The <i>Policy Groups</i> to subscribe, and (optionally) the devices to subscribe to them. |

Return value

Type:

The `uid` of the created *Account*.

90.1.4 workflowSubscriptionService.createSubscription

Aggregate method to create a new subscription.

Compatibility method. New code should prefer the strongly typed `#createNewSubscription(Account, List, List, List, List, List)`.

Parameters

| Name | Type | Description |
|---------------------|-----------------------|--|
| <code>models</code> | <code>Object[]</code> | a list of objects that will fulfil the creation of a new subscription. They follow the same order and type requirements as the strongly typed version. |

Return value

Type:

The *Account* UID of the account created.

90.1.5 workflowSubscriptionService.updateExistingSubscription

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|-----------------|-------------|
| accountUID | Long | |
| subscribeProfiles | List<Profile> | |
| unsubscribeProfiles | List<Profile> | |
| subscribePackagesInternal | List<Package> | |
| unsubscribePackagesInternal | List<Package> | |
| subscribeCatalogues | List<Catalogue> | |
| unsubscribeCatalogues | List<Catalogue> | |
| subscribePackagesCAS | List<Package> | |
| unsubscribePackagesCAS | List<Package> | |

Return value

Type:

90.1.6 workflowSubscriptionService.updateExistingSubscription

Available through Web Services.

Parameters

| Name | Type | Description |
|-----------------------------|-----------------|-------------|
| accountUID | Long | |
| subscribeProfiles | List<Profile> | |
| unsubscribeProfiles | List<Profile> | |
| subscribePackagesInternal | List<Package> | |
| unsubscribePackagesInternal | List<Package> | |
| subscribeCatalogues | List<Catalogue> | |
| unsubscribeCatalogues | List<Catalogue> | |

Return value

Type:

90.1.7 workflowSubscriptionService.updateSmartCardSubscription

Aggregate strongly typed modify subscription..

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------|---|---|
| accountUID | Long | Account UID of the account to modify. |
| subscribeProfiles | List<ProfileSmartcardSubscribableModel> | List of ProfileSmartcardSubscribableModel profiles to be subscribed |

| Name | Type | Description |
|--------------------------------------|--|---|
| <code>unsubscribeProfiles</code> | <code>List<ProfileSmartcardSubscribableModel></code> | List of ProfileSmartcardSubscribableModel profiles to be unsubscribed |
| <code>subscribePackages</code> | <code>List<PackageSmartcardSubscribableModel></code> | List of PackageSmartcardSubscribableModel to be subscribed internally. These will not be propagated to the CAS. |
| <code>unsubscribePackages</code> | <code>List<PackageSmartcardSubscribableModel></code> | List packages to be unsubscribed internally. These will not be propagated to the CAS. |
| <code>subscribeCatalogues</code> | <code>List<CatalogueSmartcardSubscribableModel></code> | List of CatalogueSmartcardSubscribableModel catalogues to be subscribed |
| <code>unsubscribeCatalogues</code> | <code>List<CatalogueSmartcardSubscribableModel></code> | List of CatalogueSmartcardSubscribableModel catalogues to be unsubscribed |
| <code>subscribePolicyGroups</code> | <code>List<PolicyGroupSmartcardSubscribableModel></code> | TODO |
| <code>unsubscribePolicyGroups</code> | <code>List<PolicyGroupSmartcardSubscribableModel></code> | TODO |

Return value

Type:

The account UID provided.

90.1.8 workflowSubscriptionService.updateSubscription

Available through Web Services.

Parameters

| Name | Type | Description |
|-------------------------|------|-------------|
| <code>accountUID</code> | Long | |

| Name | Type | Description |
|--|------------------------------------|-------------|
| <code>subscribeProfiles</code> | <code>List<Profile></code> | |
| <code>unsubscribeProfiles</code> | <code>List<Profile></code> | |
| <code>subscribePackagesInternal</code> | <code>List<Package></code> | |
| <code>unsubscribePackagesInternal</code> | <code>List<Package></code> | |
| <code>subscribeCatalogues</code> | <code>List<Catalogue></code> | |
| <code>unsubscribeCatalogues</code> | <code>List<Catalogue></code> | |

Return value

Type:

90.1.9 workflowSubscriptionService.updateSubscription

Aggregate method to update an existing subscription.

Compatibility method. New code should prefer the strongly typed `#updateExistingSubscription(Long, List, List, List, List, List, List)`

Available through HUE at <http://<server>:<port>/hue-gateway/gateway/http/js/workflowSubscriptionService/updateSubscription>

Parameters

| Name | Type | Description |
|---------------------|-----------------------|--|
| <code>models</code> | <code>Object[]</code> | a list of objects that will full fill the modification of subscription They follow the same order and type requirements as the strongly typed version. |

Return value

Type:

Long Account UID created