



Front-end Integration API

9.1



Table of Contents

Introduction	12
Copyright	12
Trademarks	12
Overview	12
Intended Audience	12
Required Skills	12
Conventions	12
Parameter Type Conventions	13
Overview of Front-end Integration API	14
General Overview	14
Actions	14
Entities	14
Protocol	14
GET Method Sample	14
POST Method Sample	14
Response Types	15
Response Format	15
Authentication	15
Household and Member Identity	16
Multilingual Support	16
Paging	16
Client Session ID	16
General Error Codes	17
HTTP Response Codes	17
HTTP Caching Headers	17
ETag Caching	18
Images	18
Deprecation	19
XML Formatter Reference	21
General	21
Single Entity Response	21
List of Entities Response	21
Result Object Response	22
Exception Response	22
JSON Formatter Reference	24
General	24
General Response Format	24
Single Entity Response	24
List of Entities Response	24
Result Object Response	24
Exception Response	25
JavaScript Code Sample	25

Remoting Infrastructure Reference	27
General	27
Remoting APIs	27
What's New	28
BasicRecordingTicket	28
Channel.isNpvrAllowed	28
DeleteRecording	28
DeleteRecording	28
DownloadLimitsAndCounters	28
ExtendedChannel.isNpvrAllowed	28
ExtendedProgram.contentProviderName	28
ExtendedProgram.genreEntityList	28
GetBasicRecordingTicketList	29
GetChannelList	29
GetChannelMetaDataTable	29
GetClearLivePlayingInfo	29
GetDownloadLimitsAndCounters	29
GetExtendedChannelList	29
GetHouseholdPackageList	29
GetHouseholdSubscribedChannelList	29
GetLivePlayingInfo	29
GetLivePlayingInfo	29
GetOrderList	29
GetPackageChannelList	29
GetPackageList	30
GetRecordingTicket	30
GetRecordingTicketList	30
GetRecordingTicketList	30
GetRecordingTicketList	30
GetRecordingTicketList	30
GetSeriesTicketList	30
GetTvShowEpisodePlayingInfo	30
GetTvShowEpisodePlayingInfo	30
GetTvShowEpisodePlayingInfo	30
GetTvShowEpisodePlayingInfo	30
GetTvShowEpisodePlayingInfo	30
GetTvShowEpisodePlayingInfo	30
GetVideoPlayingInfo	30
GetVideoTicketList	30
GetVideoTvShowEpisodeList	30
OrderRecording	30
OrderRecording	30
OrderRecording	30

OrderRecording	30
OrderRecording	30
OrderTVShow	31
RecordingTicket.duration	31
RecordingTicket.failureReason	31
RecordingTicket.lagDuration	31
RecordingTicket.leadDuration	31
RecordingTicket.resumePosition	31
RecordingTicket.userComment	31
Result.id	31
Terminal.activationCode	31
UpdateRecording	31
UpdateTerminal	31
UpdateTerminal	31
UpdateTerminal	31
UpdateTerminal	31
VideoTicket.duration	31
Entities	32
AggregatedTickets	32
AggregatedTvShowTicket	32
AggregatedVideo	32
AggregatedVideoPackageTicket	34
AssetChapter	34
Attachment	35
AttachmentMediaFile	35
Award	36
BasicRecordingTicket	36
Bookmark	37
Bouquet	38
CaptchaKey	39
Channel	39
ChannelMetaData	41
ChannelPackageSubscriptionInfo	42
ChannelReference	42
ChannelTechnicalFormat	43
CharIndex	44
ClientData	44
Cluster	44
Config	44
Contributor	45
Country	45
Discount	46
DiscountPurchaseOption	46
DownloadLimitsAndCounters	47
Dubbing	48
Encoding	48

ExtendedAggregatedVideo	48
ExtendedChannel	50
ExtendedPricingOption	51
ExtendedProgram	52
ExtendedUniqueVideoInfo	55
ExtendedVideo	56
ExternalAccount	60
Extrafield	60
Folder	60
FolderReference	61
Genre	61
Group	62
Household	62
HouseholdSubscriptions	64
HouseholdTerminals	64
Language	65
LimitedPage	65
LimitedVideo	65
LiveLicenseAcquisitionInfo	66
LivePlayingInfo	66
MediaFile	67
Member	68
Message	69
Metadata	70
NPVRQuotaStatus	70
NPVRUsedQuota	71
OffersInfo	71
Order	72
PPVEvent	72
PPVEventTicket	73
Package	74
PackageChannel	74
PackageSubscription	75
Page	75
PageReference	76
ParentalRating	77
PaymentMethodDetailsInfo	77
PaymentRegistrationProperty	78
PercentageDiscountedOption	78
PersoTokenEntity	78
PlayingInfo	79
Position	79
PreOrder	79
PricingMatrix	80
PricingOption	80
Program	82

ProgramTicket	83
RechargeOption	84
RecordingParentalRating	84
RecordingPlayingInfo	85
RecordingTicket	85
Reference	89
Result	89
Review	89
SearchedProgram	90
SearchedRecording	91
SeasonOfferOption	93
Series	94
SeriesTicket	95
Service	95
ServicePlan	97
ServicePlanReference	99
ServicePlanSubscriptionInfo	99
ServiceSubscription	99
ServiceSubscriptionInfo	100
SubscriptionTransaction	100
Subtitle	101
Terminal	101
TerminalCategory	102
TerminalModel	103
TimeShiftingServicesConfig	104
TimeWindow	105
TimezoneOffset	105
TssBookmark	105
TvShowCategoryInfo	106
TvShowReference	106
TvShowSeason	107
TvShowSeasonPage	108
TvShowSeries	108
TvShowSeriesPage	109
TvShowTicket	110
UniqueVideoInfo	112
Video	113
VideoAsset	116
VideoCard	118
VideoCardTicket	120
VideoDownloadInfo	121
VideoPackage	122
VideoPackageInfo	124
VideoPlayingInfo	125
VideoPurchaseInfo	126
VideoTicket	127

VideoTvShowSeason	130
VideoTvShowSeries	131
VideoUrlInfo	132
VideoWish	133
VodGift	133
Actions	136
ActivateTicket	136
AddBookmark	136
AddPackageSubscription	137
AddServicePlanSubscription	140
AddServiceSubscription	142
AddToWishList	144
CancelPreOrder	145
CheckSmartCard	145
CloseSession	146
CloseViewingSessions	147
CompletePaymentMethodRegistration	148
DeleteBookmark	149
DeleteMessage	149
DeletePaymentMethod	150
DeleteRecording	150
FinishPackageSubscription	153
FinishServicePlanSubscription	154
FinishServiceSubscription	155
GetAggregatedVideo	156
GetAggregatedVideoTicketList	157
GetAttachmentDownloadInfo	158
GetAttachmentMediaFileList	161
GetAttachmentPlayingInfo	162
GetBasicRecordingTicketList	164
GetBookmarkList	166
GetBouquetList	167
GetCaptchaKey	167
GetChannelList	167
GetChannelMetaDataTable	169
GetChannelSupportedModelList	169
GetClearLivePlayingInfo	170
GetClientData	171
GetConfig	171
GetContentDefinitionList	172
GetDiscount	172
GetDiscountList	173
GetDownloadLimitsAndCounters	173
GetEstDownloadingTerminalList	174
GetExtendedChannelList	175
GetExtendedProgram	176

GetExtendedProgramList	177
GetExternalAccount	178
GetFilteredProgramList	178
GetGenreList	179
GetHousehold	180
GetHouseholdBouquetList	180
GetHouseholdPackageList	180
GetHouseholdServiceList	181
GetHouseholdSubscribedChannelList	182
GetHouseholdSubscriptionTransactionList	183
GetHouseholdSubscriptions	183
GetHouseholdTerminalCategoryList	183
GetLanguageList	184
GetLiveLicenseAcquisitionInfo	184
GetLivePlayingInfo	186
GetMarkedPosition	189
GetMediaFile	190
GetMemberList	191
GetMessageList	191
GetMsg	192
GetMsgToken	193
GetNPVRProgramSupportedModelList	194
GetNPVRQuotaStatus	194
GetNPVRUsedQuota	194
GetOrderList	195
GetPPVEvent	196
GetPPVEventList	196
GetPPVEventTicketList	197
GetPackageChannelList	198
GetPackageList	199
GetPage	200
GetPageList	200
GetPageReferenceList	202
GetParentalRatingList	203
GetPaymentMethodDetailsInfo	203
GetPaymentRegistrationPropertyList	204
GetPersoToken	204
GetPlayingInfo	205
GetPreOrderList	206
GetPricingMatrix	206
GetPricingMatrixList	207
GetProgramList	208
GetProgramPlayingInfo	209
GetProgramTicketList	210
GetRechargeOptionList	210
GetRecordingMarkedPosition	211

GetRecordingTicket	212
GetRecordingTicketList	213
GetReferenceList	215
GetSeriesTicketList	216
GetServiceList	217
GetServicePlan	217
GetServicePlanList	218
GetSessionList	218
GetSpecificVideoList	219
GetSubscriptionOffers	220
GetTVShowTicketList	221
GetTerminalList	222
GetTerminalModelList	222
GetTimeShiftingServicesConfig	223
GetTimezoneOffsetList	223
GetTvShowEpisodePlayingInfo	224
GetUnifiedIndexList	227
GetUnifiedList	230
GetVideo	233
GetVideoAsset	234
GetVideoCardList	235
GetVideoCardTicketList	236
GetVideoDownloadInfo	237
GetVideoList	241
GetVideoPackage	243
GetVideoPackageList	244
GetVideoPlayingInfo	245
GetVideoPurchaseOptions	249
GetVideoSupportedModelList	250
GetVideoTicketList	251
GetVideoTvShowEpisode	253
GetVideoTvShowEpisodeList	254
GetVideoTvShowSeason	256
GetVideoTvShowSeasonList	257
GetVideoTvShowSeries	259
GetVideoTvShowSeriesList	259
GetVodGiftList	261
GetWishList	261
InsertMessage	262
IsAlive	263
Login	263
Logout	265
MarkPosition	266
MarkRecordingPosition	266
MarkViewedVodGift	268
OpenSession	268

OpenViewingSession	270
OrderGenericService	271
OrderPPVEvent	272
OrderProgram	273
OrderRecording	275
OrderTVShow	278
OrderVideo	280
OrderVideoCard	286
OrderVideoGift	287
OrderVideoPackage	290
PollMessage	293
PreOrderVideo	294
RechargeInternalPrepaid	296
RegisterBouquet	297
RegisterExternalToken	298
RegisterHousehold	299
RegisterTerminal	302
RemoveFromWishList	304
RemovePackageSubscription	305
RemoveServicePlanSubscription	306
RemoveServiceSubscription	306
ResendConfirmationEmail	307
ResetMemberPin	308
ResetParentalPin	309
ResetPassword	309
ReuseMsgToken	310
SendMessage	311
SendMsg	312
SendResponse	314
SetClientData	315
SetEntitlement	316
SetLanguage	317
TechnicianLogin	317
UnregisterBouquet	318
UnregisterExternalToken	319
UnregisterTerminal	320
UpdateHousehold	321
UpdateHouseholdSpendingLimit	322
UpdateMemberPin	322
UpdateMemberSpendingLimit	323
UpdateParentalPin	323
UpdatePassword	324
UpdatePaymentAuthentication	325
UpdateRecording	325
UpdateTerminal	326
UpdateTerminalIp	328

UpdateTransactionsPaymentMode	328
UpdateUsername	329
ValidateUsername	330
Appendixes	331
Appendix A - Action Attributes	331
Appendix B - Purple DRM Play Ready License Acquisition API Service	336
Appendix E - Purple DRM VO API Service	338
Appendix F - Purple DRM VO Personalization API Service	341
Appendix G - Captcha Image Servlet	344
Appendix I - VO Widevine DRM API Service	345
Appendix K - Fairplay DRM API Service	348
Appendix M - CSP Personalization API Service	352

Introduction

This chapter describes this manual's intended audience, specifies the required skills, and lists the available documentation.

Copyright

The contents of this documentation are strictly confidential and the receiver is obliged to use them exclusively for his or her own purposes as defined in the contractual relationship. No part of Viaccess-Orca applications or this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from Viaccess S.A. The information in this document is subject to change without notice. Viaccess S.A does not warrant that this document is error free. If you find any problems in this documentation or wish to make any comment, please report them to Viaccess-Orca in writing at documentation@viaccess-orca.com.

Trademarks

Viaccess-Orca is a trademark of Viaccess S.A ® in France and/or other countries. All other products and company names mentioned herein are the trademarks of their respective owner. Viaccess S.A may hold patents, patent applications, trademarks, copyrights or other intellectual property rights over the products described in this document. Unless expressly specified otherwise in a written license agreement, the delivery of this document does not imply the concession of any license over these patents, trademarks, copyrights or other intellectual property rights.

Overview

The purpose of this document is to define the interface and the mechanism whereby external clients or portals can interact with the RiGHTv runtime platform in order to enable the implementation of the subscriber user interface (front-end application) using various client technologies.

Intended Audience

This manual is designed for developers implementing interactive TV web portals or client applications which are based on or integrated with RiGHTv's runtime APIs.

Required Skills

The Front-end Integration API is designed for developers who have the following skills:

- Experience in using the HTTP protocol.
- Experience in using XML or JSON data formats

Conventions Used in This Guide

The following type and symbol conventions are used in this document:

- Square brackets ([]) are used after an entity name to identify arrays (Foo[]), or around optional arguments ([bar]).

- A hyphen (-) used before properties indicates object properties.
- **Bold** used in functions indicates mandatory parameters.
- Gray used in objects indicates inherited parameters.
- Monospace indicates code examples and references.

Parameter Type Conventions

Number Format

Throughout all APIs, the type Number refers to type Long, unless otherwise specified.

Overview of Front-end Integration API

This chapter describes the main concepts related to the use of Viaccess-Orca's Front-end Integration API.

General Overview

Viaccess-Orca's Front-end Integration is the main interface through which subscriber client applications gain access to all services provided by the RiGHTv platform. The Front-end Integration API is exposed by the RTE (Run Time Engine), a J2EE based application which is accessed via an HTTP-based interface.

Actions

The APIs exposed by the Front-end Integration API are referred to throughout this document as actions. The following are the main types of actions:

- Authentication - These actions are used for authentication (Login) or changing the identity of an active member.
- Data retrieval - These actions are used for the retrieval of a single data entity or a list of data entities from the server. These action names usually start with the "Get" or "Search" prefixes.
- Data modification actions - These actions are used for updating data in the server or execution of business logic, such as performing orders.

Entities

RiGHTv's data model is constructed from entity types which define the interactive TV domain. These entities, such as Household, Member and Video, which are described in the entities section of this document, are retrieved using the data retrieval actions mentioned above.

Protocol

The Front-end Integration API is based on the HTTP 1.1 protocol. All actions can be accessed using HTTP GET or POST, although GET is the preferred access method for data retrieval actions. POST should be used only in cases where data is sent to the server as parameters of data modification actions, such as the SetClientData.

GET Method Sample

`http://<host name>:<port>/RTEFacade/<action name>?parameters
parameterName=parameterValue` **Example:**

`http://172.31.150.15:3198/RTEFacade/Login?username=1&password=1`

See the following sections for more details regarding general or specific action parameters.

POST Method Sample

POST /RTEFacade/Login HTTP/1.1

Host: xxx-dev.yyy.com

Cache-Control: no-cache

Postman-Token: fa1eafef-f4cc-03c5-8fdb-e64a52f191ff

Content-Type: application/x-www-form-urlencoded

username=1&password=1&client=json

Response Types

Each action call results in one of the following responses:

- A single entity - when calling a data retrieval action defined as returning a single entity.
- A list of entities - when calling a data retrieval action defined as returning a list (array) of entities.
- A result object - when calling authentication or data modification actions.
- An exception object - in case a system or applicative error occurs during the action call.

Response Format

The Front-end Integration API can support multiple response formats used for supporting multiple client types. By default, the following formats are supported:

- JSON - used mostly for JavaScript based clients.
- XML - used for Java or other technology-based clients.

Additional formats also can be supported per specific integration requirement. This requires additional development by Viaccess-Orca. The required response format is defined by adding the client parameter to any action call. The value of the parameter should be either JSON or XML. In case a specific response format is defined for a specific client integration, the name of the client should be used. See XML and JSON format sections for specific details.

Authentication

Prior to any personal/private action call, the client should be first authenticated (for the list of actions and their privacy/login attributes, see Appendix A - Action Attributes section).

The authentication should be performed by calling the Login action with one of the supported combinations of authentication parameters (e.g. username and password). Combinations of authentication parameters are configured in the login-params-config.properties file.

Once the authentication is performed successfully, an encrypted session cookie is generated by RiGHTv and is used for identification of the calling client in all subsequent action calls. This cookie is referred to as the identity cookie throughout this document.

The identity cookie will expire after a configurable time has elapsed since the creation of the cookie (configured in the authenticator plugin - "cookie-expiration-interval" parameter).

In case it is configured to ReLogin (configured in the authenticator plugin - "is-relogin-required" parameter) and the identity cookie has expired, a NOT_AUTHENTICATED error will be returned as a result to an action call, until the client logs in again.

In case it is **not** configured to ReLogin and the identity cookie has expired, RiGHTv will check that the household's status is "Active" (the same check is done during the Login action) and if it is active, update the cookie creation time and invoke the action; otherwise a HOUSEHOLD_NOT_ACTIVE error will be returned.

For more details regarding configuration of the Authentication plugin, see the RiGHTv Configuration Guide.

Household and Member Identity

Once the authentication process is successfully performed, the household identity is cached within the identity cookie. In addition to the identity of the calling household, the identity cookie also contains the identity of the active member. By default, the active member identified by the server is the main member. In order to change the identity of the active member temporarily in a specific call (e.g. in order to update the spent amount of a specific member while ordering a movie), the **pin_code** parameter is added. The value of this parameter is the PIN of the specific member.

Multilingual Support

All data retrieval actions can return the data in one of the languages defined in the system. It is possible to specify the language for the returned information by providing the **language_code** in the request. For the case when the language is not provided, the information is returned according to the language defined for the logged in household, but when the language is not defined for the household, or when the action does not require to login, the response is returned in the system default language.

Paging

In order to support the retrieval of large lists, all data retrieval actions which return a list of entities support the paging mechanism which allows the retrieval of a specific part of the list (sublist). In order to enable the paging option, the following parameters should be added to the action call (both parameters must be supplied):

- from - a zero-based index of the item in the list that should be returned as the first item of the sublist.
- count - the number of items to be returned in the sublist.

In the case when the number of items that appear in the list from the index specified by the from parameter until the end of the list, is less than the value specified by the count parameter. The sublist's item count is less than that specified by the count parameter. Note: Using the paging mechanism is strongly recommended to ensure optimal performance.

Client Session ID

The parameter **url_session_id** allows the client to differentiate between URLs from different households. A new ID is generated for each user login and is passed with the URL.

RiGHTv ignores the parameter and no exception is thrown when passing this parameter with the URL.

From RiGHTv 8.0 SR4 Fix002 this parameter is used to identify different users logging in cases where the cache is not cleared between logins. The parameter can be used for other purposes as well.

General Error Codes

As mentioned above any error that occurs during an action execution results in an exception object response. The exception object contains the code and message properties. The valid values for the code property per action type are listed in the Throws section of each action's documentation. In addition, the following general error codes may be thrown:

- GENERAL_SERVER_ERROR - a general server error occurred.
- rte.facade.action.NoSuchAction - when calling a non-existent action.
- INVALID_CREDENTIALS - a non-existent PIN was passed to an action or the user is not logged-in.
- MISSING_PARAMETER - a mandatory parameter was not passed to an action.
- INVALID_PARAMETER - an invalid parameter was passed to an action.
- INVALID_PARAMETER_SET - some parameters that cannot be combined together were passed to an action.
- INVALID_PARAMETER_VALUE - an invalid parameter value was passed to an action.
- NOT_AUTHENTICATED - a private action was called, but the user is not logged-in, or the user's identity cookie has expired.
- HOUSEHOLD_NOT_ACTIVE - the user's identity cookie has expired and the household is not "Active" any more.
- INVALID_CONSENT - the action received cannot be processed for the user due to unfulfilled consent requirements. The user has not granted consent for the latest version of the consent bundle that governs the use of the action sent (neededConsentBundleName), or the granted status notification period has since expired. In addition, the grace period for the relevant bundle has expired.

HTTP Response Codes

By default, all action calls are responded to with the 200 OK response code, including cases where the response is of an exception type.

There may be some cases where an unexpected error occurs in the server which results in one of the 5xx HTTP error codes.

In some of the action calls the server may respond with a 304 Not modified response code. For more details see the ETag section.

HTTP Caching Headers

By default, all action responses include HTTP caching headers. These headers can be used by the client or by the web-cache layer to control their caching behavior. The caching headers are: Cache-Control, Expires, Surrogate-Control (optional) and Pragma (optional). The default values of these headers are as following:

For public actions and for images:

- Cache-Control = "public"
- Expires = "now plus 1 hour"

For private actions:

- Cache-Control = "private"
- Expires = "now plus 1 hour"

For action that should not be cached at all (such as in case of exceptions):

- Cache-Control = "no-cache"
- Expires = "0"
- Surrogate-Control = "no-store"
- Pragma = "no-cache"

in addition, custom headers can be configured per each action.

The headers and their values can be configured in invalidation-config.xml.

For more details, see "Appendix A - Action Attributes" in this document and "RiGHTv Configuration Guide" document.

ETag Caching

As of version 7.0 RiGHTv supports HTTP ETag headers.

HTTP ETag headers is one of several mechanisms that HTTP provides for cache validation, and which allows a client to make conditional requests.

If the STB browser supports ETags, the front-end application should leave ETag handling to the browser, otherwise it should handle it on its own.

Here are the main flows of the protocol:

- 1 . When a client requests some data from RiGHTv for the first time, it may receive a HTTP Etag header along with the response data.
- 2 . In this case, the client should cache the data along with the request URL and keep the associated ETag with it.
- 3 . The next time the client wants to use the same URL, it should first look for the associated Etag and add it to the request, as 'if-none-match' HTTP request header.
- 4 . If the server responds with HTTP 304 - Not modified, the client should use the data from its cache.
- 5 . If the server returns new data with a new ETag, the client should update its cache.

For more details see HTTP 1.1 RFC2616 <http://www.w3.org/Protocols/rfc2616/rfc2616.html>

Note that usually the ETag cache will be used only for personalized / private data.

In order to use ETag caching properly, http-caching-policies in invalidation-config.xml should be customized per private actions.

For more details, see "Appendix A - Action Attributes" in this document and "RiGHTv Configuration Guide" document.

Images

All images that are loaded to RiGHTv as attachments to programs, assets, channels, TV show series, TV show seasons, PPV events, service plans, services or pages can be retrieved using HTTP GET to the RTE URL, with the suffix "/images/" and the image attachment value.

The image attachment value is the relative URL of the image in the RiGHTv repository.

RiGHTv support on the fly resizing of images, the required dimensions will be specified by the client application when requesting the image.

When resizing images dimensions parameters "width" and "height" should be supplied.

The URL format for image retrieval is:

`http://<host name>:<port>/RTEFacade/images/<attachment url>`

Example:

`http://<host name>:<port>/RTEFacade/images/jamesbond1.jpg` The URL format for resized image retrieval is:

`http://<host name>:<port>/RTEFacade/images/<attachment url>?height=<height_val>&width=<width_val>`

Example:

`http://<host name>:<port>/RTEFacade/images/jamesbond1.jpg?height=200&width=200`

Deprecation

Functionality which is deprecated is marked with a special label as part of its description. For example, **Deprecated since version 6.2.**

This means that this functionality currently exists, but will be supported in only one more major version, after the deprecation version. For example, a function deprecated in version 6.2 will still be available in version 7.0 but will not be available in version 8.0. Processes that include this functionality should be modified and include the new relevant APIs instead.

What's deprecated

Name	Description
RecordingTicket.id	Deprecated since 9.0. For NPVR 2.0, use RecordingTicket.otfRecordingId instead.
RecordingTicket.advisories	Deprecated since 9.0.
RecordingTicket.supportedTerminalModels	Deprecated since 9.0.
RecordingTicket.prName	Deprecated since 9.0.
RecordingTicket.prLevel	Deprecated since 9.0.
Service.quotaSizeGB	Deprecated since 9.0. For NPVR 2.0, use Service.quotaInHours instead.
recording_id parameter in DeleteRecording action	Deprecated since 9.0. For NPVR 2.0, use the otf_recording_id parameter instead.
past parameter in DeleteRecording action	Deprecated since 9.0.
recording_id parameter in GetLivePlayingInfo action	Deprecated since 9.0. For NPVR 2.0, use the otf_recording_id instead.

Name	Description
GetNPVRUsedQuota	Deprecated since 9.0. For NPVR 2.0, use GetNPVRQuotaStatus action instead.
recording_id parameter in GetRecordingMarkedPosition action	Deprecated since 9.0. For NPVR 2.0, use the otf_recording_id instead.
recording_ids parameter in GetRecordingTicketList action	Deprecated since 9.0.
recording_id parameter in MarkRecordingPosition action	Deprecated since 9.0. For NPVR 2.0, use the otf_recording_id instead.
allow_scheduling_over_quota parameter in OrderRecording	Deprecated since 9.0.
RecordingTicket.programid	Deprecated since 9.1.
RecordingTicket.channelid	Deprecated since 9.1.
GetNPVRProgramSupportedModelList	Deprecated since 9.0.
Program.id	Deprecated since 9.1.
Program.price	Deprecated since 9.1.
ExtendedProgram.genre	Deprecated since 9.1.
ExtendedProgram.contentProvider	Deprecated since 9.1.
ExtendedProgram.genres	Deprecated since 9.1.
Genre.id	Deprecated since 9.1.
Series.genres	Deprecated since 9.1.
GetExtendedProgram filter_deleted	Deprecated since 9.1.
GetExtendedProgramList filter_deleted	Deprecated since 9.1.

XML Formatter Reference

General

The XML format is intended to be used by clients, implemented by various technologies, with the capability of XML parsing. Each entity in the response is represented as an XML element. Simple properties (string or numeric) of the entity are represented by attributes of the element. In the case when the type of property is an array of another entity type, the property is represented by a child element. The child element contains child elements representing the items in the array. See the examples below.

Single Entity Response

A single entity response always contains a single entity as the root element of the response.

Example

```
http://127.0.0.1:7778/RTEFacade/GetPage?id=30&client=xml&language_code=en
```

```
<Page id="30" name="VOD" description="Root page for VOD services" parentId="1">
<extrafields>
<Extrafield name="color" value="Blue">
</Extrafield>
</extrafields>
<attachments>
</attachments>
</Page>
```

List of Entities Response

The root element of an entity list response is always a LIST having child elements representing the entities in the list. The LIST element contains a "fullLength" attribute which represents the number of items in the full list, regardless of whether paging parameters were applied.

Example

```
http://127.0.0.1:7778/RTEFacade/GetPageList?
parent_id=30&client=xml&language_code=en

<LIST fullLength="3" timestamp="1232081333579">
<Page id="413" name="Action" description="" parentId="30">
```

```

<extrafields>

<Extrafield name="color" value="Blue">
</Extrafield>

<Extrafield name="color" value="Blue">
</Extrafield>

</extrafields>

<attachments>
</attachments>

</Page>

<Page id="613" name="Comedy" description="" parentId="30">

<extrafields>

<Extrafield name="color" value="Blue">
</Extrafield>

</extrafields>

<attachments>
</attachments>

</Page>

</LIST>

```

Result Object Response

A result response contains a single **Result** element having the following attributes:

- status - the result status which is usually SUCCESS
- message - an additional informative message

Example

```

http://127.0.0.1:7778/RTEFacade/Login?username=1&password=1&client=xml

<Result status="SUCCESS" message="Login action finished successfully"/>

```

Exception Response

An exception response contains a single **Exception** element having the following attributes:

- code - one of the valid error codes as defined in the Throws section of each action.

- message - an additional informative message
- properties - additional data in a key-value format

Example

```
http://127.0.0.1:7778/RTEFacade/Login?username=1&password=2&client=xml
```

```
<Exception code="INVALID_CREDENTIALS" message="Login failed." />
```

Example of exception with properties

```
<Exception Code="My CODE" Message="simple test message">

<Properties>

<Property Key="MyKey1" Value="MyValue1" />
<Property Key="MyKey2" Value="MyValue2" />

</Properties>

</Exception>
```

JSON Formatter Reference

General

The JSON format is commonly used for loading data from the server and converting it directly into JavaScript objects (AJAX).

For the JSON format specifications, see <http://www.json.org/>

General Response Format

When using the JSON response format, a JavaScript object is created by applying the **eval** function on the `responseText`. The object created always contains the following properties: "metadata" and "response". The response property contains the actual response to the action call, which can be a single entity, a list of entities, a result object or an exception. See the following sections for specific formats.

Single Entity Response

The response property contains a single entity as defined for the return type of the called action.

Example

```
http://127.0.0.1:7778/RTEFacade/GetPage?id=30&client=json &language_code=en

{ "metadata": { "timestamp": 1232081133811, "request": "GetPage" }, "response": {
  "attachments": [], "parentId": 1, "description": "Root page for VOD services", "name": "VOD", "extrafields": [], "id": 30 } }
```

List of Entities Response

The response property contains an array of entities as defined for the return type of the called action.

Example

```
http://127.0.0.1:7778/RTEFacade/GetPageList?parent_id=30&client=json
&language_code=en

{ "metadata": {
  "fullLength": 3, "timestamp": 1232081240212, "request": "GetPageList" }, "response": [
    { "attachments": [], "parentId": 30, "description": "", "name": "Action", "extrafields": [], "id": 413 },
    { "attachments": [], "parentId": 30, "description": "", "name": "Comedy", "extrafields": [], "id": 613 },
    { "attachments": [], "parentId": 30, "description": "", "name": "Drama", "extrafields": [], "id": 813 } ] }
```

Result Object Response

The response property contains a single result object which contains the following properties:

- status - the result status which is usually SUCCESS
- message - an additional informative message

Example

```
http://127.0.0.1:7778/RTEFacade/Login?username=1&password=1&client=json
```

```
{ "metadata":{ "timestamp":1232081521236, "request":"Login" }, "response":{ "message":"'Login' action finished successfully", "status":"SUCCESS" } }
```

Exception Response

The response property contains a single exception object which contains the following properties:

- code - one of the valid error codes as defined in the Throws section of each action
- message - an additional informative message
- properties - an additional data in a key-value format

Example

```
http://127.0.0.1:7778/RTEFacade/Login?username=1&password=2&client=json
```

```
{ "metadata":{ "timestamp":1232081494364, "request":"Login" }, "response":{ "code": "INVALID_CREDENTIALS", "message": "Login failed.", "status": "FAILURE" } }
```

Example of exception with properties

```
{ "message": "simple test message", "status": "FAILURE", "properties": [ { "value": "MyValue2", "key": "MyKey2" }, { "value": "MyValue1", "key": "MyKey1" } ], "code": "My CODE" }
```

JavaScript Code Sample

```
var url = " http://127.0.0.1:7778/RTEFacade/GetPage?
id=1&client=json&language_code=en" ;

var request = new XMLHttpRequest();

request.open( "GET", url, true);

request.onreadystatechange = function() {

    if (request.readyState == 4) {

        if (request.status == 200 || request.status == 304) {

            var response = eval(request.responseText); // convert the response text to
javascript objects

            // do something with response

        }
    }
}
```

```
}
```

```
request.send(null);
```

Remoting Infrastructure Reference

General

The Remoting Infrastructure is a communication framework between two devices that belong to the same household. It can be used to develop various feature that enhance user experience.

The most common use case is to develop a remote client PVR mechanism, to provide a subscriber the ability to control the recordings of it's Set-Top Box from a remote device such as a smartphone.

Remoting APIs

The Remoting Infrastructure is constructed of six Actions and one entity, as following:

- **ProxySession** - An entity representing information about the established proxy session between remote client and terminal.
- **OpenSession** - An action used to open a session for remote terminal communication.
- **CloseSession** - An action used to close remote terminal session.
- **GetSessionList** - An action used to get all household remote terminal sessions.
- **SendMessage** - An action used to send a message to the remote terminal.
- **PollMessage** - An action used to poll pending messages from the remote terminal.
- **SendResponse** - An action used to send response message to the remote terminal.

In addition to the above actions, when using the remoting infrastructure in offline mode, the following additional actions can be used. For example, in case of Remote Client PVR, the recording device can send a message to the remote device, informing it about the status of the requested recordings. The actions are:

- **InsertMessage** - An action used to insert a message to a household (can be used to pass messages between devices of the same household).
- **GetMessageList** - An action used to return the message list associated with the logged-in household. Asynchronous processes in the server that require the users attention are added to the message list (e.g. a failure to record a program).

These actions can be accessed by using the same URL as the standard RTE actions appended with "/remote/".

For example: <http://172.31.150.15:3198/RTEFacade/remote/OpenSession>

For more information on developing remote client PVR using the remoting infrastructure, see "RiGHTv Remote Client PVR Developers Guide".

What's New

This chapter describes the changes made in the current version over the previous one.

Name	Description
BasicRecordingTicket	An entity that represents reduced information for a recording ticket.
Channel.isNpvrAllowed	Indicates whether the channel is allowed for NPVR, used for NPVR 3.0. Used only when isNPVRHybridSupported feature is enabled in the Configuration. This parameter is used instead of the flag isRecordable in Channel and in ChannelTechnicalFormat, which are used for NPVR 1.0 and NPVR 2.0. Since 9.1 SR21.
DeleteRecording	Added parameter otf_recording_ids. Since 9.1 SR5
DeleteRecording	Added parameter personal Since 9.1 SR25
DownloadLimitsAndCounters	An entity that represents the download limits and counters.(Since 9.1 SR21)
ExtendedChannel.isNpvrAllowed	Indicates whether the channel is allowed for NPVR, used for NPVR 3.0. Used only when isNPVRHybridSupported feature is enabled in the Configuration. This parameter is used instead of the flag isRecordable in Channel and in ChannelTechnicalFormat, which are used for NPVR 1.0 and NPVR 2.0. Since 9.1 SR21.
ExtendedProgram.contentProviderName	The program content provider name.
ExtendedProgram.genreEntityList	A list of program genres. Relevant for NPVR 2.0 only, when ExtendedProgram is contained in the RecordingTicket entity.

Name	Description
GetBasicRecordingTicketList	An action used to return a list of recordings tickets for the logged-in account containing reduced information. It is recommended to use pagination with this API for improved performance. When filtering recordings by start_date and end_date, for EPG based recordings if only the lead/lag duration falls in the requested time range specified, but the recorded program is outside the range, the recording will not be returned. One of the following permutations may be used for filtering by a specific season recording: <ul style="list-style-type: none"> • series_id, season_name, channel_external_id • series_ticket_id If an episode was recorded as a stand-alone recording and not as part of a season recording, the recording will not be returned.
GetBasicRecordingTicketList	A new parameter is introduced: channel_external_id Since 9.1 SR11
GetBasicRecordingTicketList	A new parameter is introduced: series_id Since 9.1 SR11
GetBasicRecordingTicketList	A new parameter is introduced: season_name Since 9.1 SR11
GetChannelList	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetChannelMetaDataTable	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetClearLivePlayingInfo	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetDownloadLimitsAndCounters	Gets the downloads limits and the account counters. See the RiGHTv Service Guide for more information. (Since 9.1 SR21)
GetExtendedChannelList	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetHouseholdPackageList	Added sector parameter. Since 9.1 SR22.
GetHouseholdSubscribedChanne lList	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetLivePlayingInfo	New error code is introduced: TARGETED_ADS_ENCRYPTION_FAILED Since 9.1 SR23.
GetLivePlayingInfo	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetOrderList	Added support for TV show season pass orders. (Since 9.1 SR1)
GetPackageChannelList	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.

Name	Description
GetPackageList	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
GetRecordingTicket	Gets the specified single recording ticket for the logged-in account. Cannot be used to return all recorded episodes in a season. Relevant for NPVR 2.0 only.
GetRecordingTicketList	A new parameter is introduced: personal Since 9.1 SR25
GetRecordingTicketList	A new parameter is introduced: channel_external_id Since 9.1 SR11
GetRecordingTicketList	A new parameter is introduced: season_name Since 9.1 SR11
GetRecordingTicketList	A new parameter is introduced: series_id Since 9.1 SR11
GetSeriesTicketList	A new parameter is introduced: personal Since 9.1 SR25
GetTvShowEpisodePlayingInfo	A new parameter is introduced: chapter_id. Since 9.1 SR17.
GetTvShowEpisodePlayingInfo	A new parameter is introduced: bookmark_id. Since 9.1 SR17.
GetTvShowEpisodePlayingInfo	A new parameter is introduced: terminal_id. Since 9.1 SR17.
GetTvShowEpisodePlayingInfo	A new parameter is introduced: resume. Since 9.1 SR17.
GetTvShowEpisodePlayingInfo	A new parameter is introduced: encoding. Since 9.1 SR17.
GetTvShowEpisodePlayingInfo	A new error code is introduced: INVALID_ENCODING. Since 9.1 SR17.
GetVideoPlayingInfo	New error code is introduced: TARGETED_ADS_ENCRYPTION_FAILED Since 9.1 SR23.
GetVideoTicketList	A new parameter is introduced: external_asset_id Since 9.1 SR10
GetVideoTvShowEpisodeList	A new parameter is introduced: include_removed_from_service. (Since 9.1SR3)
OrderRecording	Added sector parameter and a new exception: INVALID_SECTOR Since 9.1 SR22.
OrderRecording	Added parameters npvr_time_based_recording_duration_limit
OrderRecording	Updated the description of the exception PAST_PROGRAM. Since 9.1 SR18
OrderRecording	Added recording id to result, The ID of the recording will be returned on program based non recursive recording in NPVR 2.0 model
OrderRecording	Added parameters user_comment

Name	Description
OrderTVShow	Added parameter payment_mode. The payment method for the season pass order. (Since 9.1 SR1)
RecordingTicket.duration	The duration of the recording, in milliseconds. For EPG based recordings, the duration includes the leadDuration and lagDuration. Relevant for NPVR 2.0 only.
RecordingTicket.failureReason	The reason the recording failed. (Values: MAX_QUOTA_EXCEEDED, CONCURRENT_RECORDINGS_LIMIT_EXCEEDED, MISSING_PARAMETER, RECORDING_CHANNEL_NOT_FOUND, PROGRAM_DELETED, GENERAL_ERROR). PROGRAM_DELETED corresponds to status of Deleted (5) and is relevant for NPVR 2.0 only. All other failure reasons correspond to status of Failed (4). PROGRAM_DELETED occurs when a program is deleted from the EPG. MAX_QUOTA_EXCEEDED and CONCURRENT_RECORDINGS_LIMIT_EXCEEDED occur when a user has recorded a season and recording a new episode causes the user to exceed its quota or concurrent recordings limit. The following failure reasons indicate System Integration errors: MISSING_PARAMETER, RECORDING_CHANNEL_NOT_FOUND, GENERAL_ERROR.
RecordingTicket.lagDuration	Creates a buffer after the scheduled recording time to take minor scheduling shifts into account, in minutes. Relevant only for EPG-based recordings. Relevant for NPVR 2.0 only.
RecordingTicket.leadDuration	Creates a buffer before the scheduled recording time to take minor scheduling shifts into account, in minutes. Relevant only for EPG-based recordings. Relevant for NPVR 2.0 only.
RecordingTicket.resumePosition	The position value in milliseconds. Relevant for NPVR 2.0 only. Since 9.1 SR7.
RecordingTicket.userComment	User comment for the recording. Relevant for NPVR 2.0 only. Updatable from 9.1 SR1.
Result.id	Place holder for id if exist.
Terminal.activationCode	The terminal activationCode.
UpdateRecording	Added new exception: PAST_PROGRAM. Since 9.1 SR18.
UpdateTerminal	Added a new exception: {INVALID_TERMINAL}. Since 9.1 SR20
UpdateTerminal	The name parameter is now optional. Since 9.1 SR20
UpdateTerminal	Added new optional parameter: ef_values. Since 9.1 SR20
UpdateTerminal	Added new optional parameter: ef_names. Since 9.1 SR20
VideoTicket.duration	The duration of movie attached to media file of the video related the video ticket.

Entities

This chapter describes the entities generated in the RTE.

AggregatedTickets since 8.3

An entity that represents an aggregated ticket that includes all active tickets for movies ordered through VOD/SVOD/VOD card, aggregated TV Show tickets and their episode tickets and aggregated movie package order information and their movie tickets.

Properties

Name	Type	Description
videoTickets	VideoTicket[]	A list of all the active VOD/SVOD/VOD card movie tickets.
aggregatedTvShowTickets	AggregatedTvShowTicket[]	A list of all the active TV Show tickets and related episode tickets (active and expired).
aggregatedVideoPackageTickets	AggregatedVideoPackageTicket[]	Video package order information for all active video packages, and tickets for all movies in those video packages (active and expired).

AggregatedTvShowTicket since 8.3

An entity that represents an aggregated list of TV Show tickets.

Properties

Name	Type	Description
tvShowTicket	TvShowTicket	The TV Show ticket.
episodeTickets	VideoTicket[]	A list of all the TV Show episode tickets. Tickets only exist for episodes that were already watched.

AggregatedVideo since 7.0

An entity that represents an aggregated video page. The aggregated video includes an aggregated result of one or more pages/shortcuts (services attribute) that has the same asset and exists in the same VOD category (same path); and the movies are returned consecutively (one after the other) according to the sorting option (it is essential that the sorting will be by name, that way the videos will be consecutive).

Properties

Name	Type	Description
name	String	The name of the video.
shortName	String	The short name of the video.
description	String	The description of the video.
windowStart	Number	The license start date (in milliseconds from 1/1/1970).
windowEnd	Number	The license end date (in milliseconds from 1/1/1970).
duration	Number	The duration of the asset in milliseconds.
categoryExternalId	String	The external ID of the Category (parent page in VO Management Console).
prLevel	Number	The parental rating level.
prName	String	The parental rating name.
year	Number	The year this content was produced.
contentProvider	String	The content provider name.
assetExternalId	String	The external ID of the asset as defined by the content provider.
externalChannelId	String	The external channel ID associated with the asset.
series	Series	The series to which the video asset is part.
genres	Number[]	The list of the video asset's genre IDs.
chapters	AssetChapter[]	The list of chapters.
extrafields	Extrafield[]	The list of extra fields. This list contains the extra fields of the video asset and of the series to which the asset is part.
attachments	Attachment[]	The list of attachments. This list contains the attachments of the video asset and of the series to which the asset is part.
reviews	Review[]	The list of reviews attached to the video.
awards	Award[]	The list of awards attached to the video.

Name	Type	Description
uniqueVideos	UniqueVideoInfo[]	The list of unique video information (distinct pages information) for the video asset.
tvShowReference	TvShowReference	The reference to a TV show series if this asset is linked to a TV show series.
genreEntityList	Genre[]	An list of genres of the aggregated video. Since 8.0 SR4.
contentProviderExternalId	String	The content provider external ID.
allowedTerminalCategories	TerminalCategory[]	A list of allowed terminal categories.

See Also

- [GetAggregatedVideo](#)
- [GetUnifiedList](#)
- [UniqueVideoInfo](#)

AggregatedVideoPackageTicket*since 8.3*

An entity that represents an aggregated list of video package order information and related movie tickets.

Properties

Name	Type	Description
videoPackageInfo	VideoPackageInfo	The video package order information.
moviePackageTickets	VideoTicket[]	A list of tickets for movies in the video package.

AssetChapter*since 5.3*

An entity that represents a chapter in an asset.

Properties

Name	Type	Description
id	Number	The ID of the chapter.
name	String	The name of the chapter.
position	Number	The position of the chapter, in milliseconds, from the start of the asset.
description	String	The description of the chapter.

Name	Type	Description
attachments	Attachment[]	A new attribute is introduced: An array of image attachments of the asset chapter.

Attachment

since 5.3

An entity that represents an attachment for one of the following entities:

- Page, Discount in RiGHTv
- Asset, Asset Chapter and Series in RiGHTv
- Channel, Program and Series in RiGHTv
- Service (Catch-Up, Start Over, NPVR)
- Message in RiGHTv
- Video Ticket

The attachment can be any file that you want to attach to the entity. In most cases, it is an image or a movie file.

Properties

Name	Type	Description
name	String	The name of the attachment.
value	String	The relative URL in case of image attachment. "true" in case of movie attachment.
assetId	Number	The ID of the attached asset (from RiGHTv). This applies only to movie attachments.
assetName	String	The name of the attached asset. This applies only to movie attachments.

See Also

- [Page](#)
- [Video](#)
- [VideoPackage](#)
- [AggregatedVideo](#)
- [VideoAsset](#)
- [AssetChapter](#)
- [VideoTicket](#)
- [Channel](#)
- [ExtendedProgram](#)
- [Series](#)
- [Discount](#)
- [Message](#)

AttachmentMediaFile

since 7.0

An entity that represents a media file of an attachment (bonuses or trailers).

Properties

Name	Type	Description
attachmentName	String	The name of the video attachment (of the page or the asset in this order) to which the media is attached.
mediaFileExternalId	String	The external ID of the media file.
definition	String	The stream definition of the video.
deliveryMode	String	The possible values for Delivery mode of the Media File: "Streaming" or "Download" or "Streaming and Download".
encoding	Encoding	The encoding format of the media file.

See Also

- [GetAttachmentMediaFileList](#)

Award

since 6.2

An entity that represents an award item.

Properties

Name	Type	Description
title	String	The title of the award.
year	String	The year the award was given.

See Also

- [VideoAsset](#)
- [ExtendedProgram](#)
- [GetVideoAsset](#)
- [GetExtendedProgram](#)

BasicRecordingTicket

since 9.1

An entity that represents reduced information for a recording ticket.

Properties

Name	Type	Description
programExternalId	Number	The program ID of the recording.
channelExternalId	Number	The channel ID of the recording.

Name	Type	Description
name	String	For NPVR 2.0, for EPG based recording this is the name of the program, for time-based recording, this is the name that was provided. If no name was provided the name will be empty. For NPVR 1.0 the name of the recording (the program name).
startDate	Number	The recording start date (in milliseconds from 1/1/1970). For EPG based recordings, the start date includes the leadDuration. Relevant for both NPVR 2.0 and 1.0.
endDate	Number	The recording end date (in milliseconds from 1/1/1970). For EPG based recordings, the end date includes the lagDuration. Relevant for both NPVR 2.0 and 1.0.
status	Number	The recording status. [Values: 1 - Scheduled, 2 - In progress, 3 - Recorded, 4 - Failed (for NPVR 2.0, relevant for series recording only), 5 - Deleted, 6 - Pending (relevant for NPVR 1.0 only)].
flags	Number	A bitwise description of the recording properties: <ul style="list-style-type: none">• isPartOfSeries (See the example below)
seriesId	String	The ID of the series. Empty if the recording is not part of a season recording.
seasonName	String	The name of the series season. Empty if the recording is not part of a season recording.

Examples

```

1 Check if a recording is part of series:
2     isPartOfSeries = ((flags | 1) == flags);

```

See Also

- [GetBasicRecordingTicketList](#)

Bookmark

since 5.3

An entity that represents a bookmark for an asset. The bookmark remains even after the ticket is expired

Properties

Name	Type	Description
id	Number	The ID of the bookmark.
name	String	The name of the bookmark.
author	String	The name of the author for this bookmark.
position	Number	The position of the bookmark in milliseconds from the beginning of the video.
videoId	Number	The video ID as defined in RiGHTv.
videoName	String	The video name with which this bookmark is associated.
timestamp	Number	The bookmark creation time (in milliseconds from 1/1/1970).

See Also

- [AddBookmark](#)
- [DeleteBookmark](#)
- [GetBookmarkList](#)

Bouquet

since 6.0

An entity that represents a group of channels that are provided by a specific content provider and are grouped in the front-end application as a distinctive content world.

Properties

Name	Type	Description
id	Number	The ID of the bouquet.
name	String	The name of the bouquet.
providerName	String	The name of the bouquet provider name.
extrafields	Extrafield[]	An array of extra fields attached to this bouquet.
defaultPackageId	Number	The bouquet default package. Each bouquet may have a default package.
externalId	String	The external ID of the bouquet.

See Also

- [GetBouquetList](#)
- [GetHouseholdBouquetList](#)
- [UnregisterBouquet](#)
- [RegisterBouquet](#)
- [GetHouseholdPackageList](#)

CaptchaKey*since 7.3*

An entity that contains the CAPTCHA key.

Properties

Name	Type	Description
key	String	CAPTCHA key.

See Also

- [GetCaptchaKey](#)

Channel*since 5.3*

An entity that represents a single channel. Channels in status Off and all related applications are not visible to the end user.

Properties

Name	Type	Description
id	Number	The ID of the channel.
name	String	The name of the channel.
number	Number	The number of the channel.
description	String	The description of the channel.
ip	String	The IP in which the channel is multicast.
port	Number	The port in which the channel is multicast.
category	String	The category of the channel.
prLevel	Number	The level of the parental rating set for this channel.
prName	String	The name of the parental rating set for this channel.
identifier	String	The identifier for CA/DRM integration.
affiliation	String	The unique station sign (e.g. "NBC", "HBO", etc.).

Name	Type	Description
longName	String	The full channel name.
bitrate	Number	The bitrate at which this channel is broadcast. (Number Type: Double).
sourceType	String	The type of channel ("IP" / "DVB_SI" / "MHP" / "HLS" / "SmoothStreaming" / "MPEGDash").
transportStreamId	String	The transport stream ID.
originalNetworkId	String	The original network ID.
serviceId	String	The service ID.
encoding	String	The encoding format for this channel.
contentDefinition	String	The channel content definition.
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the channel allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific channel, that limit will be returned, otherwise the global system default limit for channels will be returned. When session management is not handled by RiGHTv (i.e SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.
isNpvrAllowed	boolean	Indicates whether the channel is allowed for NPVR, used for NPVR 3.0. Used only when isNPVRHybridSupported feature is enabled in the Configuration. This parameter is used instead of the flag isRecordable in Channel and in ChannelTechnicalFormat, which are used for NPVR 1.0 and NPVR 2.0. Since 9.1 SR21.
externalChannelId	String	The external ID of the channel used to identify the channel from which programs were recorded. Relates to the channel reference ID (defined through VO Management Console).
extrafields	Extrafield[]	An array of extra fields attached to the channel.

Name	Type	Description
url	String	The channel URL address. Relevant for OTT formats.
interactiveUrl	String	The channel interactive URL.
attachments	Attachment[]	An array of attachments attached to the channel.
flags	Number	A bitwise description of the channel properties: <ul style="list-style-type: none">• isVSChannel (Live) (1)• isPaymentBased (2)• isAudioEnabled (4)• isVideoEnabled (8)• isRecordable (16)• isClientRecordable (32)• isEncrypted (64)• isCatchUpEnabled (128)• isStartOverEnabled (256)• isInteractiveChannel (1024)

Examples

```

1 // check if the channel is recordable (flags is the channel flags)
2 isRecordable = ((flags | 16) == flags);
3
4 // check if the channel is a PPV channel (flags is the channel flags)
5 isPpvChannel = ((flags | 2) == flags);
6
7 // check if the channel is a live channel (flags is the channel flags)
8 isLiveChannel = ((flags | 1) != flags);

```

See Also

- [GetChannelList](#)
- [GetPackageChannelList](#)
- [PackageChannel](#)

ChannelMetaData

since 8.4

A simplified entity presenting limited information on a Channel. Returned by GetChannelMetaDataTable.

Properties

Name	Type	Description
id	Number	The ID of the channel.
name	String	The name of the channel.
number	Number	The number of the channel.
description	String	The description of the channel.

Name	Type	Description
category	String	The category of the channel.
prLevel	Number	The level of the parental rating set for this channel.
prName	String	The name of the parental rating set for this channel.
affiliation	String	The unique station sign (e.g. "NBC", "HBO", etc.).
longName	String	The full channel name.
attachments	Attachment[]	An array of attachments attached to the channel.
extrafields	Extrafield[]	An array of extra fields attached to the channel.
externalChannelId	String	The external ID of the channel used to identify the channel from which programs were recorded.

See Also

- [GetChannelMetaDataList](#)

ChannelPackageSubscriptionInfo

since 8.1

An entity that represents a subscription to a channel package. (Since 8.1 SR1)

Properties

Name	Type	Description
channelPackage	Package	The package to which the account (household) is subscribed.
activationDate	Number	The date from which the channel package is activated for the subscriber.
expirationDate	Number	The date the subscription to the channel package expires.
storeType	String	The type of App store.

ChannelReference

since 8.4

An entity that represents a reference to a channel.

Properties

Name	Type	Description
externalChannelId	String	The external ID of the channel.

ChannelTechnicalFormat

since 8.6

An entity that represents a single technical format of a channel.

Properties

Name	Type	Description
id	Number	The ID of the technical format.
externalId	String	The external ID of the technical format.
sourceType	String	The type of technical format ("IP" / "DVB_SI" / "MHP" / "HLS" / "SmoothStreaming" / "MPEGDash").
url	String	The channel's URL address. Relevant for OTT formats.
bitrate	Number	The bitrate at which the channel is broadcast. (Number Type: Double).
ip	String	The IP on which a channel is multicast.
port	Number	The port on which a channel is multicast.
transportStreamId	String	The transport stream ID.
originalNetworkId	String	The original network ID.
serviceId	String	The service ID.
encoding	String	The encoding format for this channel.
contentDefinition	String	The channel content definition.
flags	Number	A bitwise description of the channel and technical format properties: <ul style="list-style-type: none"> • isPaymentBased (2) • isAudioEnabled (4) • isVideoEnabled (8) • isRecordable (16) • isClientRecordable (32) • isEncrypted (64) • isCatchUpEnabled (128) • isStartOverEnabled (256)

CharIndex

since 7.1

An entity that represents a key value pair of the index of each unique first letter (char) of the entities that are returned by GetUnifiedList action.

Properties

Name	Type	Description
character	String	The key of the index.
index	Number	The value of the index. The first occurrence of the character.

See Also

- [GetUnifiedIndexList](#)

ClientData

since 5.3

An entity that represents a value of client data for a requested key.

Properties

Name	Type	Description
value	String	The value of client data for a requested key.

See Also

- [GetClientData](#)
- [SetClientData](#)

Cluster

since 6.0

An entity that represents a video server cluster.

Properties

Name	Type	Description
id	Number	The ID of the Cluster.
externalId	String	The external ID of the Cluster.
name	String	The name of the Cluster.

See Also

- [Household](#)

Config

since 5.3

An entity that represents a configuration object.

Properties

Name	Type	Description
timestamp	Number	The current server time.

See Also

- [GetConfig](#)

Contributor

since 6.0

An entity that represents a contributor to the content item.

Properties

Name	Type	Description
role	String	The role of the contributor in the referenced content item.
firstName	String	The first name of the contributor.
lastName	String	The last name of the contributor.
popularity	Number	The numeric value indicating the popularity of the contributor in the context of the specific content item.
inRole	String	Provides information about the contributor role in the specific content. (Maximum length: 150). since 7.4 SR6

See Also

- [ExtendedProgram](#)
- [VideoAsset](#)

Country

since 6.0

An entity that represents a country related to the production of the content item.

Properties

Name	Type	Description
code	String	A two letter country code according to ISO-3166-1.
name	String	The country translated name. Not relevant for NPVR 2.0, when contained in RecordingTicket entity.

See Also

- [ExtendedProgram](#)
- [VideoAsset](#)

Discount

since 6.2

An entity that represents a discounted pricing matrix including its pricing options and valid time windows.

Properties

Name	Type	Description
externalId	String	The external ID of the discount.
name	String	The name of the discount.
description	String	The description of the discount.
validityTimePeriods	TimeWindow[]	An array of TimeWindow entities.
pricingOptions	PricingOption[]	An array of PricingOption entities, filled in case discount type is FIXED.
attachments	Attachment[]	An array of attachment entities (only image attachments will be supported).
discountType	String	The discount type. (Values: FIXED, PERCENTAGE)
percentageDiscountedOptions	PercentageDiscountedOption[]	An array of percentage discount options, filled if the discount type is PERCENTAGE.

Comments

- In order to allow the operator to attract as much attention as possible to the existing discounts, the front-end application should display a discount indication for the video that is connected to a currently available discount. In order to identify the available discounted videos the front-end application should filter out from the list of discounts (retrieved by GetDiscountList action), all the discounts that are not currently available (according to the discount availability periods).

See Also

- [GetDiscount](#)
- [GetDiscountList](#)
- [GetVideo](#)
- [GetVideoList](#)

DiscountPurchaseOption

since 8.3

An entity that represents a discount purchase option.

Properties

Name	Type	Description
discountOptionId	Number	The discount option ID.
discountExternalId	String	The discount external ID.
discountType	String	The discount type. (Values: FIXED or PERCENTAGE)
percentage	Number	The percentage discount.
price	Number	The price after the discount was applied. (Number Type: Double).
rentalPeriod	Number	The rental period offered. Relevant only for fixed discount options.
rentalPeriodUnit	Number	The unit of time in which the rental period is specified. Values: <ul style="list-style-type: none"> • (2) Minutes • (3) Hours • (4) Days (0 - Can be used when the rental period is not specified)
downloadPeriod	Number	The download period offered. Relevant only for fixed discount options.
downloadPeriodUnit	Number	The download period unit. Values: <ul style="list-style-type: none"> • (2) Minutes • (3) Hours • (4) Days (0 - Can be used when the download period is not specified)

DownloadLimitsAndCounters

since 9.1

An entity that represents the download limits and counters.(Since 9.1 SR21)

Properties

Name	Type	Description
monthlyLimit	Number	The number of downloads allowed for an account in a calendar month.
assetLimit	Number	The number of downloads allowed for an account per asset.
monthlyCounter	Number	The monthly downloads counter of the account.

Name	Type	Description
assetCounter	Number	The downloads counter of the specified asset for the account.

Dubbing

since 5.3

An entity that represents a dubbing option for an asset.

Properties

Name	Type	Description
flags	Number	The Bitwise description of the dubbing properties: is visually impaired (1).
audioMode	String	An audio mode of this dubbing.
audioFormat	String	An audio format of this dubbing.
language	String	Two or three characters code of the dubbing language (Depending on the installation).
order	Number	The order defined for this dubbing.

See Also

- [MediaFile](#)
- [ExtendedProgram](#)

Encoding

since 5.3

An entity that represents an encoding entity.

Properties

Name	Type	Description
id	Number	The ID for the format.
name	String	The name of the format (e.g. "MPEG-4").

See Also

- [Video](#)

ExtendedAggregatedVideo

since 8.3

An entity that represents extended aggregated video page. (Since 8.3 SR1)

Properties

Name	Type	Description
name	String	The name of the video.
shortName	String	The short name of the video.
description	String	The description of the video.
windowStart	Number	The license start date (in milliseconds from 1/1/1970).
windowEnd	Number	The license end date (in milliseconds from 1/1/1970).
duration	Number	The duration of the asset in milliseconds.
categoryExternalId	String	The external ID of the Category (parent page in VO Management Console).
prLevel	Number	The parental rating level.
prName	String	The parental rating name.
year	Number	The year this content was produced.
contentProvider	String	The content provider name.
contentProviderExternalId	String	The content provider external ID.
assetExternalId	String	The external ID of the asset as defined by the content provider.
externalChannelId	String	The external channel ID associated with the asset.
series	Series	The series to which the video asset is part.
genres	Number[]	The list of the video asset's genre IDs.
chapters	AssetChapter[]	The list of chapters.
extrafields	Extrafield[]	The list of extra fields. This list contains the extra fields of the video asset and of the series to which the asset is part.
attachments	Attachment[]	The list of attachments. This list contains the attachments of the video asset and of the series to which the asset is part.
reviews	Review[]	The list of reviews attached to the video.
awards	Award[]	The list of awards attached to the video.

Name	Type	Description
allowedTerminalCategories	TerminalCategory[]	A list of allowed terminal categories.
genreEntityList	Genre[]	An list of genres of the aggregated video.
tvShowReference	TvShowReference	The reference to a TV show series if this asset is linked to a TV show series.
extendedUniqueVideoInfoList	ExtendedUniqueVideoInfo[]	Information on each video (page) through which the specified asset is offered. Includes purchase information for each page.

See Also

- [GetAggregatedVideo](#)
- [GetVideoTvShowEpisodeList](#)

ExtendedChannel

since 8.6

An entity that represents a single channel with all its matching technical formats.

Properties

Name	Type	Description
id	Number	The ID of the channel.
name	String	The name of the channel.
number	Number	The number of the channel.
description	String	The description of the channel.
category	String	The category of the channel.
parentalRatingLevel	Number	The level of the parental rating set for this channel.
parentalRatingName	String	The name of the parental rating set for this channel.
affiliation	String	The unique station sign (e.g. "NBC", "HBO", etc.).
longName	String	The full channel name.
interactiveUrl	String	The channel interactive URL.
externalChannelId	String	The external ID of the channel used to identify the channel from which programs were recorded. Relates to the channel reference ID (defined through VO Management Console).

Name	Type	Description
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the channel allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific channel, that limit will be returned, otherwise the global system default limit for channels will be returned. When session management is not handled by RiGHTv (i.e. SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.
attachments	Attachment[]	An array of attachments attached to the channel.
extraFields	Extrafield[]	An array of extra fields attached to the channel.
technicalFormats	ChannelTechnicalFormat[]	An array of channel technical formats attached to the channel, ordered by priority
isNpvrAllowed	boolean	Indicates whether the channel is allowed for NPVR, used for NPVR 3.0. Used only when isNPVRHybridSupported feature is enabled in the Configuration. This parameter is used instead of the flag isRecordable in Channel and in ChannelTechnicalFormat, which are used for NPVR 1.0 and NPVR 2.0. Since 9.1 SR21.

ExtendedPricingOption extends [PricingOption](#)

since 8.3

An entity that extends the PricingOption entity.

Properties

Name	Type	Description
id	Number	The ID of the pricing option.
name	String	The name of the pricing option.
rentalPeriod	Number	The rental period offered, in the specified unit.
downloadPeriod	Number	The download period offered, in the specified unit.

Name	Type	Description
price	Number	The Price for this rental period, without any discount. Default value is 0.0. (Number Type: double).
rentalPeriodUnit	Number	Rental period unit, valid options: <ul style="list-style-type: none">• (2) Minutes• (3) Hours• (4) Days (0 - can be a valid option when rental period is not specified)
allowedViews	Number	The allowed number of views per VOD order.
estOnlineAvailabilityPeriod	Number	The period of time during which the content is available online for streaming and/or for download for an eST order from when the order is placed.
downloadPeriodUnit	Number	Download period unit, valid options: <ul style="list-style-type: none">• (2) Minutes• (3) Hours• (4) Days (0 - can be a valid option when download period is not specified)
estOnlineAvailabilityPeriodUnit	Number	The time unit of the eST Online Availability period. Values: <ul style="list-style-type: none">• 4 = days• 6 = months• 7 = years
type	String	The pricing option type. Possible values: streaming, download, converged, est, Est-Pre-Order (Since 8.0 SR3).
discountPurchaseOption	DiscountPurchaseOption	The discount purchase option.
storeProductReferenceId	String	The ID of the App store product reference.

ExtendedProgram extends Program

since 5.3

An entity specifying extended information on a program.

Properties

Name	Type	Description
id	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use referenceProgramId instead. The ID of the program.

Name	Type	Description
name	String	The name of the program.
description	String	The description of the program.
startDate	Number	The start time of the program (in milliseconds from 1/1/1970).
endDate	Number	The end time of the program (in milliseconds from 1/1/1970).
price	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty because the PPV channel feature was deprecated. The price of the program. (Number Type: Double).
prLevel	Number	The program parental rating level.
prName	String	The program parental rating name.
advisories	String	The program parental rating advisories.
seriesNumberOfEpisodes	Number	A series number of episodes, if the program is part of a series.
seriesSeason	String	The program series season, if it is part of a series.
referenceProgramId	String	Reference program ID as defined by the EPG provider (This is the program external ID). Added in 7.3 SR1.
seriesName	String	The program Series name, if it is part of a series.
genre	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use genreEntityList instead. The program genre ID.
contentId	String	The program content ID. From 9.1, for NPVR 2.0, when ExtendedProgram is contained in the RecordingTicket entity, if contentId was not provided when the program was created or modified, a unique internally generated content ID will be returned only.

Name	Type	Description
contentProvider	String	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use contentProviderName instead. The program content provider, when ExtendedProgram is contained in the RecordingTicket entity.
seriesId	String	The ID for the series in case the program is part of a series. Can be used for series recording.
dubbing	Dubbing[]	The available Dubbings for this program.
subtitles	Subtitle[]	The available Subtitles for this program.
shortDescription	String	The program short description.
contributors	Contributor[]	A list of contributors related to this program.
extrafields	Extrafield[]	An array of extra fields attached to the program.
contentProviderName	String	The program content provider name.
attachments	Attachment[]	An array of attachments attached to the program.
awards	Award[]	A list of awards attached to the program.
definition	String	The program stream definition.
flags	Number	A bitwise description of the program properties: <ul style="list-style-type: none"> • isPartOfSeries (1) • isRecordable (2) • isCatchupEnabled (4) • isStartOverEnabled (8) • isVisuallyImpaired (16) • isHearingImpaired (32)
countries	Country[]	A list of countries related to this program.
genreEntityList	Genre[]	A list of program genres. Relevant for NPVR 2.0 only, when ExtendedProgram is contained in the RecordingTicket entity.
genres	Number[]	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use genreEntityList instead. An array of genre IDs for the program.

Name	Type	Description
series	Series	The program series information.
shortName	String	The program short name.
reviews	Review[]	A list of reviews attached to the program.
episodeId	String	An ID of the episode, in case the video asset is part of a series.
year	Number	The year this content was produced.

Examples

```

1 // check if the program is part of series (flags is the program flags)
2 isPartOfSeries = ((flags | 1) == flags);
3 // check if the program is recordable (flags is the program flags)
4 isRecordable = ((flags | 2) == flags);

```

See Also

- [GetExtendedProgram](#)
- [Contributor](#)

ExtendedUniqueVideoInfo**since 8.3**

Information on each video (page) through which the specified asset is offered. Includes purchase information for each page. (Since 8.3SR1)

Properties

Name	Type	Description
id	Number	The page internal ID.
externalId	String	The external ID of the video page.
name	String	The page name.
status	Number	The video status: <ul style="list-style-type: none"> • Published (2) • Announced (3) • Removed from Service (4)
definition	String	The page definition.
template	String	The name of the template for the page.
attachments	Attachment[]	The list of attachments. This list contains the attachments of the video page and of the series (if this page is part of a series).
extrafields	Extrafield[]	The list of extra fields. This list contains the extra fields of the video page and of the series (if this page is part of a series).

Name	Type	Description
removalDate	Number	The date and time at which the service will be removed from service.
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the movie allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific movie page, that limit will be returned, otherwise the global system default limit for VOD will be returned. When session management is not handled by RiGHTv (i.e. SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.
pricingOptions	ExtendedPricingOption[]	The list of pricing options available for the page.
supportedTerminalModels	TerminalModel[]	The list of devices supported for the page.
vodCards	VideoCard[]	The list of VOD Cards associated with the page.
services	Service[]	The list of services associated with the page.
seasonOfferOptions	SeasonOfferOption[]	A list of all Season Passes associated with the season page.
servicePlans	ServicePlan[]	The list of Service Plans that include the provided services.
videoPackage	VideoPackage	The video package associated with the movie page that is part of the video package page.

See Also

- [ExtendedAggregatedVideo](#)
- [GetVideoTvShowEpisodeList](#)
- [GetAggregatedVideo](#)

ExtendedVideo

since 8.1

An extended entity holding information on a video.

Properties

Name	Type	Description
id	Number	The ID of the video page (from RiGHTv).

Name	Type	Description
name	String	The name of the video (from the RiGHTv).
template	String	The template of the video (vod_service / svod_service / etc.).
description	String	The description of the video (from RiGHTv).
duration	Number	The duration of the asset in milliseconds.
categoryId	Number	The category ID (parent page ID from RiGHTv).
prLevel	Number	The video parental rating level.
prName	String	The video parental rating name.
advisories	String	The video parental rating advisories.
windowStart	Number	The license start date (in milliseconds from 1/1/1970).
windowEnd	Number	The license end date (in milliseconds from 1/1/1970).
assetId	Number	The ID of the asset (from the VO Management Console).
pricingMatrixId	Number	The ID of the pricing matrix attached to this video. If the video is a video package movie, the value of pricingMatrixId will be empty.
chapters	AssetChapter[]	An array of chapters of the video.
metadata	extraFields	An entity that represents an extra field for an asset in the asset manager (RiGHTv). Entity properties: <ul style="list-style-type: none"> • name - The name of the extra field, as defined in RiGHTv • value - The Value of the extra field
extrafields	Extrafield[]	An array of extra fields of the video.
attachments	Attachment[]	An array of attachments of the video. This list contains the attachments of the page, video asset and series.
encodings	Encoding[]	An array of encodings.
genres	Number[]	An array of genre IDs of the video.

Name	Type	Description
reviewerRating	Number	The external reviewer rating for this content.
year	Number	The year this content was produced.
shortName	String	A short name of the video.
adsInfo	String	Any advertisement related information for the video.
flags	Number	A bitwise description of the video properties: <ul style="list-style-type: none"> • Ads Pre-Roll (1) • Ads Post-Roll (2)
status	Number	The video's status: <ul style="list-style-type: none"> • Published (2) • Announced (3) • Removed from Service (4)
externalChannelId	String	The external channel ID associated with the asset associated to this video service.
broadcastTime	Number	The time the video was broadcast.
contentProvider	String	The video content provider name.
contentProviderExternalId	String	The video content provider external ID.
removalDate	Number	The time the video will be removed from service.
reviews	Number	A list of reviews attached to the video.
awards	Award[]	A list of awards attached to the video.
externalId	String	The external ID of the video.
type	String	The video type, possible values: <ul style="list-style-type: none"> • "movie" • "vod package movie"
rentalPeriod	Number	The rental period of the video package movie (empty otherwise).
rentalPeriodUnit	Number	The time unit of the video package movie rental period (empty when this is not a video package movie): <ul style="list-style-type: none"> • Minutes (2) • Hours (3) • Days (4)

Name	Type	Description
seriesName	String	The series name, if the video is part of a series.
seriesSeason	String	The series season name, if the video is part of a series.
discountId	String	The external ID of the discount attached to the video. For more details about the use of this parameter see the Discount entity specification.
assetExternalId	String	The external ID of the asset attached to the video.
plannedPublishDate	Number	The time when the video is planned to be published.
definition	String	The video definition.
keywords	String	The keywords attached to the video. Keywords are free text separated by semicolons.
isSecured	Boolean	A boolean that indicates whether the video is secured.
seriesNumberOfEpisodes	Number	The number of episodes in the series, if the video is part of a series.
securityGroups	Group[]	A list of security groups/services.
episodeId	String	The ID of the episode, in case the video is part of a series. Introduced in 7.0 SR1.
allowedTerminalCategories	TerminalCategory[]	A list of terminal categories that are allowed to watch the video.
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the movie allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific movie page, that limit will be returned, otherwise the global system default limit for VOD will be returned. When session management is not handled by RiGHTv (i.e. SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.

Name	Type	Description
genreEntityList	Genre[]	An list of genres of the video. Since 8.0 SR4.
terminalModels	TerminalModel[]	List of supported terminal models

See Also

- [LimitedVideo](#)
- [GetVideo](#)
- [GetVideoList](#)
- [OrderVideo](#)
- [Discount](#)

ExternalAccount*since 5.3*

An entity that represents an external account for the household; used for Prepaid systems.

Properties

Name	Type	Description
balance	Number	The account balance. (Number Type: double).

See Also

- [GetExternalAccount](#)

Extrafield*since 5.3*

An entity that represents an extra field defined for entity customization purposes.

Properties

Name	Type	Description
name	String	The name of the extra field.
value	String	The value of the extra field.

See Also

- [Page](#)
- [Video](#)
- [ExtendedProgram](#)

Folder*since 6.1*

An entity that represents a folder in which the page is located. A folder is actually a general page that has child pages.

Properties

Name	Type	Description
id	Number	The ID of the folder.
name	String	The name of the folder.
template	String	The name of the folder's template.
externalId	String	The folder external ID.

See Also

- [GetPageReferenceList](#)

FolderReference

since 8.0

An entity that represents a reference to a folder (a page in the VOD Catalog). Since 8.0 SR4.

Properties

Name	Type	Description
id	Number	The ID of the page.
externalId	String	The external ID of the page.

See Also

- [GetServiceList](#)

Genre

since 5.3

An entity that represents a genre.

Properties

Name	Type	Description
id	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use externalId instead. The ID of the genre.
name	String	The name of the genre.
parentName	String	The name of the genre's parent in case of sub genre.
externalId	String	The external ID of the genre.

See Also

- [GetGenreList](#)

Group

since 7.0

An entity that represents a security group/service.

Properties

Name	Type	Description
externalId	String	The external ID of the service/group.
type	String	The group security type, can be 'Service' or 'General'.

See Also

- [Page](#)
- [UniqueVideoInfo](#)
- [Video](#)
- [Household](#)

Household

since 5.3

An entity that represents a household (account). A household usually contains several members (users).

Properties

Name	Type	Description
id	Number	The ID of the household.
spendingLimit	Number	The purchase limit set for the household for all types of transactions (orders, not subscriptions).(Number Type: double).
spentTotal	Number	The total amount of money spent.(Number Type: double).
isHouseholdScope	Boolean	A boolean value indicating whether the spending scope is set to Household.
languageCode	String	The language code as defined in the language list.
parentalPin	String	The parental PIN for the Household, to be used whenever the content has parental rating.
discount	Number	The discount for which the household has entitled. The value here is in a percentage value (e.g. 10 means 10% discount).
phone	String	The phone number for this household.
extrafields	Extrafield[]	An array of extra fields attached to the Household.

Name	Type	Description
transactionsPaymentMode	String	Default payment mode in which transactions can be made: Normal (Postpaid/Prepaid) or internal prepaid. In the order, the client indicates which payment method to use - it may retrieve the default of the household and use it or provide another payment method.
paymentAuthentication	String	Required authentication rule when performing monetary transactions. Authentication is performed by the client. Valid values: <ul style="list-style-type: none"> • never • always • normal only • internal prepaid only
internalPrepaidBalance	Number	The internal account balance. (Number Type: double).
externalId	String	The external ID of the household (might be null/empty, depending on Supported functionality option: HIDE_HOUSEHOLD_EXTERNAL_ID_IN_SUI_SDK).
billingDay	Number	Household billing day. Not mandatory if not provided, default value will be set from global configuration(7.4SR2)
cluster	Cluster	The household's video server cluster information (optional - since 7.2).
maxTerminals	Number	The maximum number of terminals allowed in this household.
plannedInactivationDate	Number	The planned inactivation date (in milliseconds). Used to deactivate a household on a specific date.(Since 7.4SR7 merged into 8.0 SR3).
parentalPinLastAdminResetTime	Number	The date (in milliseconds) at which the parental PIN was last reset by customer support (using the CRM API). Parental PIN changes performed using the Front-end API do not affect this value.
thirdPartyAuth	String	The name of the third party authenticating the Household, if any. [Values: null, facebook] Since 9.0 SR1.

Name	Type	Description
type	Number	The household's type (1 - operator subscriber / 2 - non-operator subscriber).
groups	Group[]	A list of security groups/services authorized for the household.
username	String	The household main member user name.
billingMethod	String	Basic payment method of the Household. Household may have in addition an Internal Prepaid Account. (postpaid / prepaid / online payment method).
timezone	String	The name of the household's time zone.

See Also

- [Member](#)
- [GetHousehold](#)

HouseholdSubscriptions

since 8.1

An entity that represents the subscriptions of an account (household).(Since 8.1 SR1)

Properties

Name	Type	Description
channelPackageSubscriptionInfoList	ChannelPackageSubscriptionInfo[]	The list of channel packages to which the account is subscribed to directly (not through a Service Plan).
serviceSubscriptionInfoList	ServiceSubscriptionInfo[]	The list of services to which the account is subscribed to directly (not through a Service Plan).
servicePlanSubscriptionInfoList	ServicePlanSubscriptionInfo[]	The list of Service Plans to which the account is subscribed.

HouseholdTerminals

since 6.2

An entity that represents a household's terminals (devices). Represents a Household's terminals

Properties

Name	Type	Description
householdId	Number	The ID of the household.
terminals	Terminal[]	An array of terminals belonging to the household.

See Also

- [GetTerminalList](#)

Language*since 5.3*

An entity that represents a language defined in RiGHTv.

Properties

Name	Type	Description
code	String	The language code in the format of two English letters (e.g "en" for "English", "es" for "Espanol").
name	String	The language name as defined in the DB (e.g "English").

See Also

- [Household](#)
- [GetLanguageList](#)
- [SetLanguage](#)

LimitedPage*since 5.3*

An entity that represents a limited information about a page in RiGHTv.

Properties

Name	Type	Description
id	Number	The unique ID of the page.
name	String	The name of the page.

See Also

- [Page](#)
- [GetPageList](#)

LimitedVideo*since 5.3*

An entity that represents limited information for a video.

Properties

Name	Type	Description
id	Number	The ID of the video page (from RiGHTv).
name	String	The name of the video (from RiGHTv).
template	String	The template of the video (vod_service / svod_service / etc.).

Name	Type	Description
external_channel_id	String	The external channel ID of the related asset.
broadcast_time	Number	The time when the video was broadcast.
seriesSeason	String	The series season name, if the video is part of a series.
seriesNumberOfEpisodes	Number	A series number of episodes, if the limited video is part of a series.
seriesName	String	The series name, if the video is part of a series.
removalDate	Number	The page's scheduled removal date.
plannedPublishDate	Number	The page's scheduled publish date.

See Also

- [Video](#)
- [GetVideo](#)
- [GetVideoList](#)
- [OrderVideo](#)
- [SearchVideos](#)

LiveLicenseAcquisitionInfo

since 7.4

An entity that represents information needed for live license acquisition.

Properties

Name	Type	Description
casToken	String	A string with encrypted CAS token.
rightsObject	String	An XML string containing the rights object.

See Also

- [GetLiveLicenseAcquisitionInfo](#)

LivePlayingInfo

since 7.2

An entity that represents information needed for playing a channel, PPV(deprecated since 8.4)/STOV/Catch-Up program or an NPVR recording.

Properties

Name	Type	Description
channelId	Number	The ID of the channel.
playingUrl	String	The URL for playing the channel.

Name	Type	Description
casToken	String	The CAS token.
casTokenExpiration	Number	The expiration time in milliseconds of the CAS token
urlExpiration	Number	Expiration time in milliseconds of the CDN token. (As returned from the CDN plugin).
parentalRatings	RecordingParentalRating[]	The parental ratings of the program with the parental rating of programs in lead and lag time. relevant for NPVR 1.0 only.
resumePosition	Number	The last position marked by the subscriber, in milliseconds, from the beginning of the program/recording. Since 8.0 SR2 "position lead time" configuration is introduced and the resumePosition for recordings is calculated according to "position lead time".
duration	Number	The duration of the program/recording in milliseconds.

See Also

- [GetLivePlayingInfo](#)

MediaFile

since 5.3

An entity that represents a media file. Each asset can hold several media files. For example, if the operator wants to support both HD and non-HD subscribers, the operator can add two media files to the same asset, one in High Definition, and another in Standard Definition.

Properties

Name	Type	Description
bitrate	Number	The bit rate of the media file.
encryption	String	The media file encryption.
ratio	String	The display type (e.g. "4:3"/"16:9").
dubbing	Dubbing[]	A list of dubbings available to this media file.
subtitles	Subtitle[]	A list of subtitles available to this media file.
horizontalSize	Number	The media file horizontal size in pixels.

Name	Type	Description
deliveryMode	String	The media file delivery mode. Possible values: <ul style="list-style-type: none">• "Streaming"• "Download"• "Streaming and Download"
definition	String	The media file definition (e.g. "HD", "SD", "3D").
adaptiveStreaming	String	The media file adaptive streaming type. Possible values: <ul style="list-style-type: none">• "Not Adaptive"• "HLS"• "Smooth Streaming"• "MPEGDash"
externalId	String	The external ID of the media file.
verticalSize	Number	The media file vertical size in pixels.
extrafields	Map	The media file extra fields (not translatable).

See Also

- [GetMediaFile](#)
- [VideoAsset](#)

Member

since 5.3

An entity that represents a member (user) in the logged-in household (account).

Properties

Name	Type	Description
id	Number	The ID of the Member.
firstName	String	The member first name.
lastName	String	The member surname.
email	String	The member email.
spendingLimit	Number	The purchase limit set for the member for all types of transactions (orders, not subscriptions). (Number Type: double).
spentTotal	Number	The total amount of money the household spent. (Number Type: double).

Name	Type	Description
pin	String	A private PIN of the member, to be used whenever a purchase is made, or settings are modified.
isMainMember	Boolean	A boolean that indicates whether this is the main member (there is only one main member in a household).
extrafields	Extrafield[]	An array of extra fields attached to the member.
externalId	String	The external ID of the member.

See Also

- [Household](#)
- [GetMemberList](#)

Message*since 5.3*

An entity that represents a message for the subscriber.

Properties

Name	Type	Description
id	Number	The ID of the message.
timestamp	Number	The creation time stamp of the message.
properties	Object	The message properties. For type: "NPVR_SUBSCRIPTION_EXPIRATION_NOTIFICATION" the remaining amount of days before service subscription expiration will be provided in the value of property named "days_before_expiration".
title	String	The message title.
expirationDate	Number	The message expiration date.
publishDate	Number	The message publishing date.
text	String	The message body.
code	String	The message code. Deprecated since 7.1
attachments	Attachment[]	An array of image attachments attached to the message.
externalId	String	The external ID of the message.

Name	Type	Description
type	String	<p>The message type. Used instead of the property 'code' which is deprecated.</p> <p>Possible system values:</p> <ul style="list-style-type: none"> • PROGRAM_DELETED • HOUSEHOLD_PROGRAM_RECORDING_DELETED • PROGRAM_RECORDING_DELETED • ALL_HOUSEHOLD_RECORDINGS_DELETED • NPVR_DISABLED_FOR_THE_PROGRAM • USED_QUOTA_EXCEEDED • USED_QUOTA_THRESHOLD_REACHED • RECORDING_ORDER_DELETED • NON_PVR_PROGRAM • NON_PVR_CHANNEL • USER_NOT_SUBSCRIBE_TO_CHANNEL • HOUSEHOLD_NOT_SUBSCRIBE_TO_NPVR • GENERAL_ERROR • EXPIRED_RECORDING_ORDER • NPVR_SUBSCRIPTION_EXPIRATION_NOTIFICATION

See Also

- [GetMessageList](#)
- [DeleteMessage](#)

Metadata

since 5.3

An entity that represents an extra field for an asset in the asset manager (RiGHTv).

Properties

Name	Type	Description
name	String	The name of the extra field, as defined in RiGHTv.
value	String	The Value of the extra field.

See Also

- [Video](#)

NPVRQuotaStatus

since 9.0

The status of NPVR quota, including the quota used by completed and scheduled recordings, and the total quota assigned. Relevant only for NPVR 2.0.

Properties

Name	Type	Description
recorded	Double	The total duration of recordings already recorded, in hours.
scheduled	Double	The total duration of recordings in progress and recordings scheduled for the future, in hours.
subscribedQuota	Integer	The total quota of the NPVR service to which the account is subscribed, in hours. 0 if the account is not subscribed to NPVR.

See Also

- [GetNPVRQuotaStatus](#)

NPVRUsedQuota*since 7.4*

Represents the NPVR quota used by the household.

Properties

Name	Type	Description
recordedUsedQuotaBytes	Number	The quota for recorded programs already used (in bytes). (READ ONLY).
scheduledUsedQuotaBytes	Number	The estimated quota for programs that are scheduled to be recorded (in bytes). (READ ONLY).

See Also

- [GetNPVRUsedQuota](#)

OffersInfo*since 8.1*

An entity that represents the available subscription offers.(Since 8.1 SR1)

Properties

Name	Type	Description
channelPackageList	Package[]	Channel package list offered for sale.
servicePlanList	ServicePlan[]	Published Service Plan list offered for sale.
serviceList	Service[]	Service list offered for sale.

Order

since 5.3

An entity that represents a payment transaction of a content (e.g. VOD, VOD Package, VOD Card, PVR, PPV, etc).

Properties

Name	Type	Description
id	Number	The unique ID of the order.
name	String	The name for the order.
price	Number	The price paid for this order. (Number Type: double).
timestamp	Number	The time the order was purchased.
type	String	The order type ("VOD" / "PPV" (PPV is deprecated since 8.4) / "PVR" / "VOD PACKAGE" / "VOD CARD" / "VOD GIFT" / "Recharge" / "Deduction" / "SEASON PASS" / generic order name). (Deduction was added since 7.4 SR5, SEASON PASS was added since 9.1 SR1).
bouquetId	Number	A bouquet identifier the order belongs to, if relevant.
prName	String	The order parental rating name.
pageExternalId	String	The external ID of ordered video or video package.
flags	Number	A bit representation for billing method for an account(0 = postpaid, 1 = internal prepaid, 2 = prepaid).
exTransactionId	String	The external transaction ID set by external payment system.
advisories	String	The order parental rating advisories.
preOrderDate	Number	This field indicates whether this order emerged from pre-order.(Since 8.0 SR3).
prLevel	Number	The order parental rating level.

See Also

- [GetOrderList](#)

PPVEvent

since 8.4

An entity that represents PPV event.

Properties

Name	Type	Description
id	Number	The unique ID for the PPV event.
name	String	The name of the PPV event.
externalId	String	The externalId of the PPV event.
promotionText	String	The promotion text of the PPV event.
startTime	Number	The time from which this PPV event is available (in milliseconds from 1/1/1970).
endTime	Number	The time at which this PPV event expires (in milliseconds from 1/1/1970).
price	Number	The price paid for this PPV event. (Number Type: double).
attachments	Attachment[]	An array of attachments of the PPV event.
channelExternalId	String	The channel external ID of the PPV event.
channelNumber	Number	The channel number of the PPV event.
channelName	String	The channel name of the PPV event.
status	String	The status of the PPV event. Possible values are: active, future.

See Also

- [GetPPVEventList](#)

PPVEventTicket

since 8.4

An entity that represents a PPV event ticket. A PPV event ticket is created for an account when a PPV event is ordered.

Properties

Name	Type	Description
ticketExternalId	String	The external ID of the PPV event ticket.
orderExternalId	String	The external ID of the order.
householdId	Number	The ID of the account (household) that created the order.
ppvEventExternalId	Number	The external ID of the PPV event.

Name	Type	Description
startDate	Number	The time and date at which the PPV event ticket is activated (in milliseconds from 1/1/1970).
endDate	Number	The time and date at which the PPV event ticket is no longer active (in milliseconds from 1/1/1970).
status	String	The status of the PPV event ticket.1 - active, 2 - expired, 3 - future.

Package

since 5.3

An entity that represents a package of channels.

Properties

Name	Type	Description
id	Number	The ID of the package.
name	String	The name of the package.
description	String	The description of the package.
price	Number	The price of the package, in the defined currency (in RiGHTv). (Number Type: float).
extrafields	Extrafield[]	An array of extra fields attached to this package. Since 7.4 SR4
servicePlanReferenceList	ServicePlanReference[]	An array of Service Plan reference list related to this package. Since 8.1 SR1
externalId	String	The package external ID.
storeProductReferenceId	String	The ID of the App store product reference.

See Also

- [PackageChannel](#)
- [PackageSubscription](#)
- [GetPackageList](#)

PackageChannel

since 5.3

An entity that represents the minimal information about a channel that is associated to a package.

Properties

Name	Type	Description
id	Number	The ID of the channel.
name	String	The name of the channel.
number	Number	The number of the channel. Relates to the channel reference ID (defined through VO Management Console).
externalId	String	The external ID of the channel. Used to identify the channel from which the programs were recorded. relates to the channel reference id (defined through VO Management Console).

See Also

- [Package](#)
- [PackageSubscription](#)
- [Channel](#)
- [GetPackageChannelList](#)

PackageSubscription

since 5.3

An entity that represents the package to which the household subscribed.

Properties

Name	Type	Description
id	Number	The ID of the package subscription.
name	String	The name of the package. Since 8.0 SR4 fix005, merged into 8.2.
subscriptionExpiration	Number	The package subscription expiration date. If the account is subscribed to the package directly and through a service plan, the latest expiration date is returned. Since 8.0 SR4 fix005, merged into 8.2.
externalId	String	The external ID of the package. Since 8.0 SR4 fix005, merged into 8.2.

See Also

- [Package](#)
- [PackageChannel](#)
- [GetHouseholdPackageList](#)

Page extends LimitedPage

since 5.3

An entity that represents the page information.

Properties

Name	Type	Description
id	Number	The unique ID of the page.
name	String	The name of the page.
description	String	The description of the page.
parentId	Number	The ID of the parent page.
template	String	The name of the template the page is based on.
extrafields	Extrafield[]	An array of extra fields attached to this page.
attachments	Attachment[]	An array of attachments attached to this page.
isSecured	Boolean	A boolean indicating whether the page is secured.
isBlockedBrowsing	Boolean	A boolean indicating whether the page is blocked for browsing by unauthorized users. Relevant only for general pages.
externalId	String	The page external ID.
securityGroups	Group[]	A list of security groups/services.
status	Number	The page status: <ul style="list-style-type: none"> • Published (2) • Announced (3) • Removed from Service (4)

See Also

- [GetPageList](#)
- [GetPage](#)

PageReference

since 6.1

An entity that represents a reference to a page (including the folder in which it exists) and an indication if it is a shortcut.

Properties

Name	Type	Description
id	Number	The ID of the page.
name	String	The name of the page.
template	String	The name of the page's template.

Name	Type	Description
isShortcut	Boolean	A boolean indicating whether the page is shortcut.
folder	Folder	The folder in which the page is located.
externalId	String	The page external ID.

See Also

- [GetPageReferenceList](#)

ParentalRating*since 5.3*

An entity that represents the parental rating option.

Properties

Name	Type	Description
id	Number	The ID of the parental rating option.
name	String	The name of the parental rating option.
level	Number	The level of the parental rating option. Higher level means stronger rating. A level of 0 means No Rating, while 9999 is the highest rating and means Blocked.

See Also

- [GetParentalRatingList](#)

PaymentMethodDetailsInfo*since 7.3*

The payment information details that are presented to the user.

Properties

Name	Type	Description
ccExpiryMonth	int	The credit card expiry month.
ccExpiryYear	int	The credit card expiry year.
ccHolderName	String	The credit card holder first and last name.
ccLastFourDigits	int	The last 4 digits of the credit card.
ccType	String	The credit card type. For example: Visa, Mastercard.
paymentSystemReferenceId	String	The payment system reference ID to the payment method.

See Also

- [GetPaymentMethodDetailsInfo](#)

PaymentRegistrationProperty*since 7.3*

A name/value object that holds the payment registration properties.

Properties

Name	Type	Description
name	String	Payment property name.
value	String	Payment property value.

See Also

- [GetPaymentRegistrationPropertyList](#)

PercentageDiscountedOption*since 8.3*

An entity that represents a percentage discounted option.

Properties

Name	Type	Description
discountOptionId	Number	The discount option ID.
type	String	The type of the percentage discount option. RENT applies to all rental delivery methods (Streaming, Download and Converged) in the pricing matrix (Values: RENT, BUY).
percentage	Number	The discount percentage.

Comments

- In order to allow the operator to attract as much attention as possible to the existing discounts, the front-end application should display a discount indication for the video that is connected to a currently available discount. In order to identify the available discounted videos the front-end application should filter out from the list of discounts (retrieved by GetDiscountList action), all the discounts that are not currently available (according to the discount availability periods).

See Also

- [GetDiscount](#)
- [GetDiscountList](#)

PersoTokenEntity*since 7.3*

An entity that represents the personalization token.

Properties

Name	Type	Description
persoTokenString	String	The personalization string token
dataHash	String	The data hash of the persoTokenString

PlayingInfo*since 5.4*

An entity that represents the playing information for a content item.

Properties

Name	Type	Description
url	String	The full URL address to play the content item.
position	Number	The last position marked by the subscriber, in milliseconds, from the beginning of the content item.
duration	Number	The duration of the content item in milliseconds.

See Also

- [GetPlayingInfo](#)

Position*since 7.3*

An entity that represents a position within a movie that is kept on the ticket.

Properties

Name	Type	Description
position	Number	The position value in milliseconds

See Also

- [GetMarkedPosition](#)

PreOrder*since 8.0*

An entity that represents a pre-order of an announced page (usually for a discounted price). A pre-order can be created for a movie page that is announced but not yet published (usually for a discounted price). (Since 8.0 SR3)

Properties

Name	Type	Description
preOrderId	Number	The unique ID of the pre-order.

Name	Type	Description
pageExternalId	String	The external ID of corresponding movie page.
price	Number	The price of the pre-order. (Number Type: double).
failureReason	String	The reason the pre-order failed.
preOrderDate	Number	The date the pre-order was created.
toOrderDate	Number	The date of the attempt to convert a pre-order to regular order.
cancelDate	Number	The date the pre-order was cancelled.
paymentMethod	String	The payment method of the pre-order: NORMAL or INTERNAL-PREPAID.
status	String	The status of the pre-order: PENDING(1), FAILED(3), TO_BE_CANCELLED(4), CANCELLED(5).

See Also

- [GetPreOrderList](#)

PricingMatrix

since 6.1

An entity that represents a pricing matrix. A pricing matrix includes a list of one or more pricing options, and is associated to VOD service pages to indicate the possible price options of certain content.

Properties

Name	Type	Description
id	Number	The ID of the pricing matrix.
pricingOptions	PricingOption[]	An array of pricing options which construct the matrix.

See Also

- [GetPricingMatrixList](#)

PricingOption

since 6.1

An entity represents a pricing option.

Properties

Name	Type	Description
id	Number	The ID of the pricing option.
name	String	The name of the pricing option.
rentalPeriod	Number	The rental period offered, in the specified unit.
downloadPeriod	Number	The download period offered, in the specified unit.
price	Number	The Price for this rental period, without any discount. Default value is 0.0. (Number Type: double).
rentalPeriodUnit	Number	Rental period unit, valid options: <ul style="list-style-type: none">• (2) Minutes• (3) Hours• (4) Days (0 - can be a valid option when rental period is not specified)
allowedViews	Number	The allowed number of views per VOD order.
estOnlineAvailabilityPeriod	Number	The period of time during which the content is available online for streaming and/or for download for an eST order from when the order is placed.
downloadPeriodUnit	Number	Download period unit, valid options: <ul style="list-style-type: none">• (2) Minutes• (3) Hours• (4) Days (0 - can be a valid option when download period is not specified)
estOnlineAvailabilityPeriodUnit	Number	The time unit of the eST Online Availability period. Values: <ul style="list-style-type: none">• 4 = days• 6 = months• 7 = years
type	String	The pricing option type. Possible values: streaming, download, converged, est, Est-Pre-Order (Since 8.0 SR3).

See Also

- [GetPricingMatrix](#)
- [Discount](#)
- [ExtendedPricingOption](#)

Program

since 5.3

An entity that represents the program details.

Properties

Name	Type	Description
id	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use referenceProgramId instead. The ID of the program.
name	String	The name of the program.
description	String	The description of the program.
startDate	Number	The start time of the program (in milliseconds from 1/1/1970).
endDate	Number	The end time of the program (in milliseconds from 1/1/1970).
price	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty because the PPV channel feature was deprecated. The price of the program. (Number Type: Double).
prLevel	Number	The program parental rating level.
prName	String	The program parental rating name.
advisories	String	The program parental rating advisories.
seriesNumberOfEpisodes	Number	A series number of episodes, if the program is part of a series.
seriesSeason	String	The program series season, if it is part of a series.
flags	Number	A bitwise description of the program properties: <ul style="list-style-type: none"> • isPartOfSeries (1) • isRecordable (2) • isCatchupEnabled (4) • isStartOverEnabled (8) • isVisuallyImpaired (16) • isHearingImpaired (32)

Name	Type	Description
referenceProgramId	String	Reference program ID as defined by the EPG provider (This is the program external ID). Added in 7.3 SR1.
shortName	String	The program short name.
episodeId	String	The ID of the episode, in case the program is part of a series. Introduced in 7.0 SR1.
genres	Genre[]	A list of genres of the program. Since 8.0 SR4.
attachments	Attachment[]	An array of attachments attached to the program. Since 8.0 SR4.
seriesName	String	The program Series name, if it is part of a series.

See Also

- [ExtendedProgram](#)
- [GetProgramList](#)

ProgramTicket*since 5.3*

An entity that represents a ticket for a paid program.

Properties

Name	Type	Description
id	Number	The unique ID for the program ticket.
channelId	Number	The channel ID for which this ticket was opened.
programId	Number	The program ID for which this ticket was opened.
name	String	The name of the ticket (Program name).
description	String	The description of the program.
windowStart	Number	The time from which this ticket is available (in milliseconds from 1/1/1970).
windowEnd	Number	The time at which this ticket expires (in milliseconds from 1/1/1970).
price	Number	The price paid for this program. (Number Type: float).

Name	Type	Description
prLevel	Number	The level of the parental rating set for this channel.
prName	String	The name of the parental rating set for this channel.
advisories	String	The rating advisories.
bouquetId	Number	The bouquet identifier the ticket belongs to, if it is a part of a bouquet.

See Also

- [GetProgramTicketList](#)

RechargeOption*since 6.0*

An entity that represents an option to recharge an internal prepaid account.

Properties

Name	Type	Description
id	Number	The unique ID for the option.
amount	Number	The recharge amount. (Number Type: double).
bonusAmount	Number	The amount of bonus to which the customer is entitled. (Number Type: double).

See Also

- [GetRechargeOptionList](#)

RecordingParentalRating*since 7.4*

An entity that represents the parental rating of a program.

Properties

Name	Type	Description
shiftFromStartMillis	Number	The amount of milliseconds from the beginning of the recording from where this parental rating is relevant.
durationMillis	Number	The duration in milliseconds after the deflection point that this parental rating is relevant.
prLevel	Number	The parental rating level.

Name	Type	Description
prName	String	The parental rating name.

See Also

- [LivePlayingInfo](#)

RecordingPlayingInfo

since 5.3

An entity that represents the playing information for a recording.

Properties

Name	Type	Description
ip	String	The IP where the recording can be played from.
port	String	The port where the recording can be played from.
duration	Number	The duration of the recording in milliseconds.
url	String	The full URL address to play the recording.
position	Number	The last position marked by the subscriber, in milliseconds, since the beginning of the recording.

See Also

- [GetProgramPlayingInfo](#)

RecordingTicket

since 7.4

An entity that represents a ticket for a program recording.

Properties

Name	Type	Description
id	Number	Deprecated since 9.0. The unique ID for the recording. Relevant for NPVR 1.0 only. (Unique)
programId	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use programExternalId instead. The program ID of the recording.

Name	Type	Description
channelId	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use channelExternalId instead. The channel ID of the recording.
name	String	For NPVR 2.0, for EPG based recording this is the name of the program, for time-based recording, this is the name that was provided. If no name was provided the name will be empty. For NPVR 1.0 the name of the recording (the program name).
description	String	The description of the recording (the program description).
startDate	Number	The recording start date (in milliseconds from 1/1/1970). For EPG based recordings, the start date includes the leadDuration. Relevant for both NPVR 2.0 and 1.0.
endDate	Number	The recording end date (in milliseconds from 1/1/1970). For EPG based recordings, the end date includes the lagDuration. Relevant for both NPVR 2.0 and 1.0.
windowEnd	Number	The expiration date for the recording. Relevant for NPVR 1.0 only.
advisories	String	Deprecated since 9.0. The parental rating advisories of the recording. Relevant for NPVR 1.0 only.
status	Number	The recording status. [Values: 1 - Scheduled, 2 - In progress, 3 - Recorded, 4 - Failed (for NPVR 2.0, relevant for series recording only), 5 - Deleted, 6 - Pending (relevant for NPVR 1.0 only)].
flags	Number	A bitwise description of the recording properties: <ul style="list-style-type: none"> • isPartOfSeries (See example below)
size	Number	The size of the recording in bytes. When isLiveOTFSupported is set to true, the size of the recording will be in milliseconds.
supportedTerminalModels	TerminalModel[]	Deprecated since 9.0. Terminal models of the household on which this recording can be played. Relevant for NPVR 1.0 only.

Name	Type	Description
prName	String	For NPVR 2.0 - The name of the maximum (highest restriction) parental rating of all programs included within the recording. For NPVR 1.0 - the parental rating name of the main program only (without the parental rating of programs in lead and lag time).
prLevel	Number	The numeric value that represents the parental rating. The higher the number, the more restricted the rating. The primary purpose of this number is to order between the different ratings. This is mainly used for setting thresholds, above which the user is required to enter a PIN code in order to watch the content. [Valid Values: 0-9999]. For NPVR 1.0 - the parental level of the main program. For NPVR 2.0 - the level of the maximum (highest restriction) parental of all programs included within the recording.
channelExternalId	String	The external ID of the channel. If the channel is turned off or deleted, it is possible that the channel external ID returned now belongs to another channel if the same ID was reused. For NPVR 2.0 this parameter is filled for both time-based and EPG based recordings.
failureReason	String	The reason the recording failed. (Values: MAX_QUOTA_EXCEEDED, CONCURRENT_RECORDINGS_LIMIT_EXCEEDED, MISSING_PARAMETER, RECORDING_CHANNEL_NOT_FOUND, PROGRAM_DELETED, GENERAL_ERROR). PROGRAM_DELETED corresponds to status of Deleted (5) and is relevant for NPVR 2.0 only. All other failure reasons correspond to status of Failed (4). PROGRAM_DELETED occurs when a program is deleted from the EPG. MAX_QUOTA_EXCEEDED and CONCURRENT_RECORDINGS_LIMIT_EXCEEDED occur when a user has recorded a season and recording a new episode causes the user to exceed its quota or concurrent recordings limit. The following failure reasons indicate System Integration errors: MISSING_PARAMETER, RECORDING_CHANNEL_NOT_FOUND, GENERAL_ERROR.

Name	Type	Description
otfRecordingId	String	The ID of the recording. Relevant for NPVR 2.0 only. (Unique)
lagDuration	Number	Creates a buffer after the scheduled recording time to take minor scheduling shifts into account, in minutes. Relevant only for EPG-based recordings. Relevant for NPVR 2.0 only.
userComment	String	User comment for the recording. Relevant for NPVR 2.0 only. Updatable from 9.1 SR1.
leadDuration	Number	Creates a buffer before the scheduled recording time to take minor scheduling shifts into account, in minutes. Relevant only for EPG-based recordings. Relevant for NPVR 2.0 only.
parentalRatings	RecordingParentalRating[]	A list of parental ratings for the recording. Relevant only when NPVR is set to 2.0.
resumePosition	Number	The position value in milliseconds. Relevant for NPVR 2.0 only. Since 9.1 SR7.
extendedProgram	ExtendedProgram	The metadata for the first program in a time-based recording. For EPG based recordings, contains the metadata of the requested program. Relevant only when NPVR is set to 2.0.
programExternalId	String	The program external ID (reference_program_id) since 7.4SR5. For NPVR 2.0 if the parameter is empty then indicates this is a time-based recording.
duration	Number	The duration of the recording, in milliseconds. For EPG based recordings, the duration includes the leadDuration and lagDuration. Relevant for NPVR 2.0 only.

Examples

```

1 Check if a recording is part of series:
2 isPartOfSeries = ((flags | 1) == flags);

```

See Also

- [GetRecordingTicketList](#)
- [GetRecordingTicket](#)

Reference

since 5.3

An entity that represents a reference item. Used whenever you need an ID-value pair (for example, in the genre list).

Properties

Name	Type	Description
id	Number	The ID of the reference item.
value	String	The reference value matching the ID.

See Also

- [GetReferenceList](#)

Result

since 5.3

The response object for authentication and data modification actions.

Properties

Name	Type	Description
status	String	The Status of the action ("SUCCESS" / "FAILURE").
message	String	A short message from the server, indicating the result.
id	String	Place holder for id if exist.

Review

since 6.2

An entity that represents a review item.

Properties

Name	Type	Description
reviewText	String	The review text.
reviewerFirstName	String	The reviewer first name.
reviewerLastName	String	The reviewer last name.
reviewerOrganization	String	The reviewer organization.
reviewerRating	Number	The reviewer rating. (Number Type: Float).

See Also

- [VideoAsset](#)
- [ExtendedProgram](#)
- [GetVideoAsset](#)
- [GetExtendedProgram](#)

SearchedProgram extends [Program](#)*since 5.3*

An entity that represents a searched program results.

Properties

Name	Type	Description
id	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use referenceProgramId instead. The ID of the program.
name	String	The name of the program.
description	String	The description of the program.
startDate	Number	The start time of the program (in milliseconds from 1/1/1970).
endDate	Number	The end time of the program (in milliseconds from 1/1/1970).
price	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty because the PPV channel feature was deprecated. The price of the program. (Number Type: Double).
prLevel	Number	The program parental rating level.
prName	String	The program parental rating name.
advisories	String	The program parental rating advisories.
seriesNumberOfEpisodes	Number	A series number of episodes, if the program is part of a series.
seriesSeason	String	The program series season, if it is part of a series.

Name	Type	Description
flags	Number	A bitwise description of the program properties: <ul style="list-style-type: none"> • isPartOfSeries (1) • isRecordable (2) • isCatchupEnabled (4) • isStartOverEnabled (8) • isVisuallyImpaired (16) • isHearingImpaired (32)
referenceProgramId	String	Reference program ID as defined by the EPG provider (This is the program external ID). Added in 7.3 SR1.
episodeId	String	The ID of the episode, in case the program is part of a series. Introduced in 7.0 SR1.
genres	Genre[]	A list of genres of the program. Since 8.0 SR4.
attachments	Attachment[]	An array of attachments attached to the program. Since 8.0 SR4.
seriesName	String	The program Series name, if it is part of a series.
channelId	Number	The ID of the channel on which this program is broadcast.
channelName	String	The name of the channel on which this program is broadcast.
channelNumber	Number	The number of the channel on which this program is broadcast.
shortName	String	The program short name.

SearchedRecording extends [Program](#) (deprecated) since 7.4

Deprecated since 9.0.

An entity that represents a searched recording results.

Properties

Name	Type	Description
id	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use referenceProgramId instead. The ID of the program.
name	String	The name of the program.

Name	Type	Description
description	String	The description of the program.
startDate	Number	The start time of the program (in milliseconds from 1/1/1970).
endDate	Number	The end time of the program (in milliseconds from 1/1/1970).
price	Number	Deprecated since 9.1. From 9.1 this parameter is returned empty because the PPV channel feature was deprecated. The price of the program. (Number Type: Double).
prLevel	Number	The program parental rating level.
prName	String	The program parental rating name.
advisories	String	The program parental rating advisories.
seriesNumberOfEpisodes	Number	A series number of episodes, if the program is part of a series.
seriesSeason	String	The program series season, if it is part of a series.
flags	Number	A bitwise description of the program properties: <ul style="list-style-type: none"> • isPartOfSeries (1) • isRecordable (2) • isCatchupEnabled (4) • isStartOverEnabled (8) • isVisuallyImpaired (16) • isHearingImpaired (32)
referenceProgramId	String	Reference program ID as defined by the EPG provider (This is the program external ID). Added in 7.3 SR1.
shortName	String	The program short name.
episodeId	String	The ID of the episode, in case the program is part of a series. Introduced in 7.0 SR1.
genres	Genre[]	A list of genres of the program. Since 8.0 SR4.
attachments	Attachment[]	An array of attachments attached to the program. Since 8.0 SR4.
seriesName	String	The program Series name, if it is part of a series.

Name	Type	Description
recordingId	Number	The ID of the recording.

See Also

- [ExtendedRecordingSearch](#)
- [SearchedProgram](#)

SeasonOfferOption**since 8.2**

An entity holding information on a Season Pass offer. A Season Pass can be used to offer a complete season of a series as on-demand content in a single order. Multiple offers with different options can be offered for a single season. Offers can differ in terms of business model (e.g. eST or rental), content definition (e.g. SD/HD), rental consumption period, availability period and allowed devices.

Properties

Name	Type	Description
seasonOfferOptionId	String	A concatenation of the Season Pass offer ID and the pricingOptionId with an underscore (_) as a delimiter. For example, 55_98.
definition	String	The Season Pass offer option content definition (for example, "HD", "SD", "3D").
rentalPeriod	Number	The period of time during which the user can watch an episode in the season starting from when the user first watches the episode. After this time has elapsed the user can no longer watch the episode. The time unit is specified by rentalPeriodUnit. (When 0 is returned, indicates that the rental period was not specified).
downloadPeriod	Number	The period of time during which the user can download an episode starting from when the user first downloads the episode. After this time has elapsed the user can't download the episode any more. The time unit is specified by downloadPeriodUnit.
rentalPeriodUnit	Number	The unit of measurement used for the rental period. Values: <ul style="list-style-type: none"> • Minutes (2) • Hours (3) • Days (4) (0 - Can be used when the rental period is not specified)

Name	Type	Description
downloadPeriodUnit	Number	The unit of measurement used for the download period. Values: <ul style="list-style-type: none">• Minutes (2)• Hours (3)• Days (4) (0 - Can be used when the download period is not specified)
type	String	The season offer option delivery mode. [Values: streaming, download, converged, eST]
offerEndDate	Number	The last date at which the season offer option can be purchased.
rentalConsumptionPeriod	Number	The number of days during which the season can be watched from the time of purchase as opposed to the rentalPeriod which starts from the time the user first watches an episode (relevant for rental only). Access to episodes included in the Season Pass will be blocked once either of these two limits is reached.
price	Number	The price for the Season Pass. The default is 0.0. (Number Type: double).
restrictedToTerminalCategories	TerminalCategory[]	The device categories allowed for the Season Pass offer.

Series

since 5.4

An entity that represents a season of a series. Each season of the series will have its own entity.

Properties

Name	Type	Description
name	String	The name of the series.
seasonName	String	The name of the series season.
description	String	The description of the series.
genreEntityList	Genre[]	An list of genres of the series. Since 8.0 SR4.
shortName	String	The short name of the series.
seriesId	String	The ID of the series as provided by the content provider.

Name	Type	Description
shortDescription	String	The short description of the series.
genres	Number[]	Deprecated since 9.1. From 9.1 this parameter is returned empty if NPVR 2.0 is enabled. Use genreEntityList instead. An array of genre IDs of the series.
numberOfEpisodes	Number	The number of episodes per season in the series. (Number type: Integer).

See Also

- [ExtendedProgram](#)
- [VideoAsset](#)

SeriesTicket

since 7.4

An entity that represents a ticket for an ordered series recording.

Properties

Name	Type	Description
id	Number	The unique identifier of the series.
seriesName	String	The series name.
seasonName	String	The series season name.

See Also

- [OrderRecording](#)
- [GetSeriesTicketList](#)

Service

since 5.3

An entity that represents a service.

Properties

Name	Type	Description
id	Number	The ID of the service.
name	String	The name of the service.
price	Number	The price of the service. (Number Type: double).
attachments	Attachment[]	An array of attachments attached to this service. Since 8.0 SR4.

Name	Type	Description
quotaInHours	Integer	The NPVR quota in hours. The quota cannot be unlimited. Relevant only for NPVR 2.0.
serviceCatalogFolders	FolderReference[]	The folders in the VOD Catalog tree to which the SVOD service is linked. Relevant for SVOD business mode. Since 8.0 SR4.
servicePlanReferenceList	ServicePlanReference[]	An array of Service Plan reference list related to this service. Since 8.1 SR1
quotaSizeGB	Number	Deprecated since 9.0. Relevant for NPVR 1.0 only. Quota NPVR size in GB. Value must be positive fraction or -1 = unlimited quota size. (Number Type: Double).
storeProductReferenceId	String	The ID of the App store product reference.
externalId	String	The external ID of the service.
concurrentRecordingsLimit	Integer	The maximum number of recordings allowed at the same time. Relevant only for NPVR 2.0. Changes to the maximum number of concurrent recordings only affects recordings that have not yet been scheduled.
model	String	The model of the service, possible values: <ul style="list-style-type: none"> • "SVOD" • "TSTV" • "STOV" • "NPVR"
description	String	The description of the service.
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the service (Live or Catch-Up) allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific service, that limit will be returned, otherwise the global system default limit for the relevant service type will be returned. When session management is not handled by RiGHTv (i.e. SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.
extrafields	Extrafield[]	An array of extra fields attached to this service. Since 7.4 SR4.

See Also

- [ServiceSubscription](#)
- [GetServiceList](#)

ServicePlan**since 8.1**

An entity that represents a Service Plan.(Since 8.1 SR1)

Properties

Name	Type	Description
externalId	String	The Service Plan external ID.
name	String	The Service Plan name.
description	String	The Service Plan description.
price	Number	The Service Plan price. (Number Type: double).
removalDate	Number	From this date it is no longer possible to subscribe to the Service Plan.
activationDate	Number	The date from which the Service Plan is activated for any subscriber that subscribes to this Service Plan. Only relevant and mandatory if activationOption is set to ON_DATE.
activationOption	String	Indicates how consumption of the Service Plan is activated for a subscriber. Possible values are: <ul style="list-style-type: none"> • 1 = IMMEDIATE. Consumption is activated immediately when the account subscribes to the Service Plan. • 2 = ON_DATE. Consumption is activated on the date specified in activationDate. • 3 = ON_FIRST_DEVICE_REGISTERATION. Consumption is activated the first time the account registers its first device. • 4 = ON_FIRST_LOGIN. Consumption is activated when the account first logs in.

Name	Type	Description
servicePlanOption	String	<p>Specifies how the duration of an account's access to the Service Plan is defined. Possible values are:</p> <ul style="list-style-type: none"> • 1 = RENEWED. The Service Plan can be consumed until the account unsubscribes from the Service Plan. • 2 = FIXED_BY_DURATION. The Service Plan can be consumed by the number of days specified in consumptionDurationInDays until the beginning of the last day (00:00:01). • 3 = FIXED_BY_DURATION_END_DAY. The Service Plan can be consumed by the number of days specified in the property consumptionDurationInDays until the end of the last day (at 23:59:59). • 4 = FIXED_BY_DATE. The Service Plan can be consumed until the date set in fixedEndDate.
consumptionDurationInDays	Number	The number of days from subscription activation date, that the Service Plan can be consumed. Only relevant and mandatory when servicePlanOption is set to FIXED_BY_DURATION.
fixedEndDate	Number	The date the Service Plan ends. From this date the services in the Service Plan will not be available for consumption to accounts (households) that are subscribed to the Service Plan. Only relevant and mandatory when servicePlanOption is set to FIXED_BY_DATE.
extrafields	Extrafield[]	The Service Plan extra fields.
attachments	Attachment[]	The Service Plan image attachments.
channelPackages	Package[]	The channel packages included in the Service Plan.
bouquets	Bouquet[]	The bouquets offered by the Service Plan.
services	Service[]	SVOD services offered by the Service Plan.
excludedFromSelfService	Boolean	Indicates whether the Service Plan can be purchased by the customer. If true, the service plan cannot be purchased by the customer. The default is false.

Name	Type	Description
storeProductReferenceId	String	The ID of the App store product reference.

ServicePlanReference

since 8.1

An entity that represents a reference to a Service Plan.(Since 8.1 SR1)

Properties

Name	Type	Description
externalId	String	The Service Plan external ID.
name	String	The Service Plan name.

ServicePlanSubscriptionInfo

since 8.1

An entity that represents a subscription to a Service Plan. (Since 8.1 SR1)

Properties

Name	Type	Description
servicePlan	ServicePlan	The Service Plan to which the account (household) is subscribed.
startDate	Number	The date the subscriber requested a subscription to the Service Plan.
removalRequestDate	Number	The date when the subscriber requested to be removed from the Service Plan.
activationDate	Number	The date from which the Service Plan is activated for the subscriber.
expirationDate	Number	The date the subscription to the Service Plan expires.
storeType	String	The type of App store.

ServiceSubscription

since 5.3

An entity that represents a subscription to a service.

Properties

Name	Type	Description
id	Number	The ID of the Service.
externalId	String	The external ID of the service. Since 8.0 SR4.

Name	Type	Description
model	String	The model of the service, possible values: <ul style="list-style-type: none">• "SVOD"• "TSTV"• "STOV"• "NPVR" Since 8.0 SR4.
serviceName	String	The name of the service. Since 8.0 SR4.
subscriptionExpiration	Number	The end date of the service subscription. In future versions, for NPVR Service 2.0, on service subscription expiration all the household recordings are deleted. Since 8.0 SR4.

See Also

- [Service](#)
- [GetHouseholdServiceList](#)

ServiceSubscriptionInfo

since 8.1

An entity that represents a subscription to a service. (Since 8.1 SR1)

Properties

Name	Type	Description
service	Service	The service to which the account (household) is subscribed.
activationDate	Number	The date from which the service is activated for the subscriber.
expirationDate	Number	The date the subscription to the service expires.
storeType	String	The type of App store.

SubscriptionTransaction

since 7.3

The subscription transaction details.

Properties

Name	Type	Description
externalId	String	The service/package/service plan external ID.
name	String	The service/package/service plan name.

Name	Type	Description
subscriptionType	String	The product type: package, service, service plan
transactionDate	Number	The transaction date.
status	Number	The transaction status: 1 - success, 2 - failed
paymentSystemTransactionId	String	The payment system reference ID to the payment method.
price	Number	The price of the subscription. (Number Type: double).
errorMessage	String	The transaction failure message.

See Also

- [GetHouseholdSubscriptionTransactionList](#)

Subtitle

since 5.3

An entity that represents a subtitle option.

Properties

Name	Type	Description
name	String	The subtitle language name for this subtitle. Deprecated
flags	Numeric	A bitwise description of the subtitle properties: <ul style="list-style-type: none"> • isHearingImpaired (1) • isForced (2) • isInVideo (4)
language	String	A two or three character code for the subtitle language (Depending on the installation).
type	String	The subtitle's type.

See Also

- [MediaFile](#)
- [ExtendedProgram](#)

Terminal

since 6.2

An entity that represents a terminal (device).

Properties

Name	Type	Description
id	Number	The terminal's internal ID.
macAddress	String	The terminal's MAC address.
serialNumber	String	The terminal's serial number.
smartCardId	String	The terminal's smart card identifier.
manufacturerId	String	The terminal's manufacturer ID .
terminalModelId	String	The terminal's model ID. Free Text, not related to RiGHTv's terminal model.
operatorId	String	The terminal's operator ID.
ip	String	The terminal ip address.
port	Number	The terminal port.
registrationDate	Number	The terminal registration date(in milliseconds).
extraFields	Extrafield[]	The terminal's extra fields (not translatable).
name	String	The terminal name (as provided by the user during self-service, or by CSR).
activationCode	Number	The terminal activationCode.
model	TerminalModel	The terminal's model.

See Also

- [GetTerminalList](#)
- [UpdateTerminalIp](#)

TerminalCategory

since 7.1

An entity that represents a terminal category (device category).

Properties

Name	Type	Description
externalId	String	The terminal category external ID.
name	String	The terminal category name.
maxTerminals	Number	The maximum allowed terminals for operator subscribers for this category per household (-1 = unlimited).

Name	Type	Description
maxTerminalsOfNonOperator	Number	The maximum allowed terminals for non-operator subscribers for this category per household (-1 = unlimited).

See Also

- [GetHouseholdTerminalCategoryList](#)

TerminalModel*since 7.1*

Defines a device (terminal model).

Properties

Name	Type	Description
externalId	String	The device model external ID.
name	String	The device model name.
categoryExternalId	String	The device category external ID.
vodCasName	String	The name of the VOD CAS of the device model. Deprecated since 7.4 SR7, merged into 8.0 SR3. Use TerminalModel.vodCasNames instead.
liveCasName	String	The name of the Live CAS of the device model.
vodVsName	String	The name of the VOD VS/CDN of the device model. Deprecated since 7.4 SR7, merged into 8.0 SR3. Use TerminalModel.vodVsNames instead.
vodVsNames	String[]	The list of VOD VS names of the device model. Since 7.4 SR7, merged in 8.0 SR3.
vodCasNames	String[]	The list of VOD CAS names of the device model. Since 7.4 SR7, merged in 8.0 SR3.
liveVsName	String	The Live VS/CDN name of the device model.

Name	Type	Description
availableDefinitions	String[]	The list of content definitions for the specified video that are supported by the device model, sorted by the definition level. Relevant for GetVideoSupportedModelList, GetVideoPurchaseOptions, GetVideoTvShowEpisodeList, GetAggregatedVideo and GetExtendedUnifiedList actions, null for all other functions. Since 8.3 SR3.
highestDefinitionForVideo	String	The highest definition supported by the device model. (in use only for GetVideoSupportedModelList, GetVideoPurchaseOptions and GetExtendedUnifiedList actions, null in all other cases).

See Also

- [GetTerminalModelList](#)

TimeShiftingServicesConfig*since 7.3*

An entity that represents a time shifting service configuration.

Properties

Name	Type	Description
leadTimeInMinutes	Number	Time-shifting services lead time in minutes.
lagTimeInMinutes	Number	Time-shifting services lag time in minutes.
cpvrAvailabilityPeriodInHours	Number	CPVR availability period in hours.
catchUpAvailabilityPeriodInHours	Number	Catch-Up availability period in hours, up to a maximum of 576 hours (24 days).
npvrAvailabilityPeriod	Number	The allowable subscriber retention period for NPVR.
npvrExpirationPeriod	Number	The allowable subscriber expiration period for NPVR.
npvrExpirationPeriodUnit	Number	The allowable subscriber expiration period unit. For NPVR 2.0 the only valid unit is <ul style="list-style-type: none"> • (4) Days . For NPVR 1.0 the valid options are: <ul style="list-style-type: none"> • (2) Minutes • (3) Hours • (4) Days

Name	Type	Description
npvrAvailabilityPeriodUnit	Number	The allowable subscriber retention period unit. For NPVR 2.0 the only valid unit is • (4) Days . For NPVR 1.0 the valid options are: • (2) Minutes • (3) Hours • (4) Days

See Also

- [GetTimeShiftingServicesConfig](#)

TimeWindow*since 6.2*

An entity that represents a time window.

Properties

Name	Type	Description
startTime	Number	The window start time in milliseconds format.
endTime	Number	The window end time in milliseconds format.

TimezoneOffset*since 6.0*

An entity that represents an offset from GMT in a specified range. Used for DST support.

Properties

Name	Type	Description
startDate	Number	The offset start date (in milliseconds from 1/1/1970).
endDate	Number	The offset end date (in milliseconds from 1/1/1970).
offset	Number	The number of milliseconds to offset from GMT between startDate and endDate.

See Also

- [GetTimezoneOffsetList](#)

TssBookmark*since 7.3*

An entity that represents a program bookmark per household.

Properties

Name	Type	Description
id	Number	The bookmark ID.
householdId	Number	The household ID.
programId	Number	The program ID.
resumePosition	Number	The bookmark position in miliseconds.
createDate	Number	The bookmark creation date.

TvShowCategoryInfo

since 8.1

An entity that represents a TV show category.

Properties

Name	Type	Description
externalId	String	The page external ID.
description	String	The page description.
status	Number	The status of the page: Published (2), Announced (3), Removed from Service (4).
template	String	The page template.
extrafields	Extrafield[]	The page extra fields.
attachments	Attachment[]	The page attachments.

TvShowReference

since 8.1

An entity that represents a reference to a season of a TV show series.

Properties

Name	Type	Description
seriesExternalId	String	The external ID of the TV show series.
seasonExternalId	String	The external ID of the TV show season.
seriesName	String	The name of the TV show series.
seasonNumber	Number	The TV show season number.
seasonName	String	The TV show season name.

Name	Type	Description
episodeNumber	Number	The TV show episode number.

TvShowSeason

since 8.1

An entity that represents a TV show season.

Properties

Name	Type	Description
externalId	String	The TV show season external ID.
name	String	The TV show season name.
seriesExternalId	String	The TV show series external ID.
seriesName	String	The TV show series name.
contentProviderExternalId	String	The TV show season content provider external ID.
contentProviderName	String	The TV show season content provider name.
shortName	String	The TV show season short name.
description	String	The TV show season description.
shortDescription	String	The TV show season short description.
numberOfEpisodes	Number	The number of episodes in the TV show season.
seasonNumber	Number	The TV show season number.
year	Number	The production year of the TV show season.
studio	String	The studio that produced the TV show season.
countries	Country[]	The countries which produced the TV show season.
genres	Genre[]	The TV show season genres.
contributors	Contributor[]	The list of contributors to the TV show season (director, actors etc.).
prName	String	The TV show season parental rating name.
prLevel	Number	The TV show season parental rating level.

Name	Type	Description
parentalAdvisories	String	Warnings on content that is offensive, violent or otherwise considered unsuitable for children.
extrafields	Extrafield[]	The TV show season extra fields.
attachments	Attachment[]	The TV show season attachments.

TvShowSeasonPage

since 8.1

An entity that represents a TV show series page.

Properties

Name	Type	Description
id	Number	The page internal id.
externalId	String	The page external ID.
name	String	The page name.
tvShowSeason	TvShowSeason	The TV show season entity.
template	String	The template name for the TV show season page.
isSecured	Boolean	Indicates whether the TV show season page has security restrictions.
securityGroups	Group[]	A list of security groups/services associated with the page.
status	Number	The status of the page: Published (2), Announced (3), Removed from Service (4).
description	String	The page description.
parentId	String	The parent ID of the page.
extrafields	Extrafield[]	The TV show season page extra fields.
attachments	Attachment[]	The TV show season page attachments.

TvShowSeries

since 8.1, 8.1

An entity that represents a TV show series.

Properties

Name	Type	Description
externalId	String	The TV show series external ID.
name	String	The TV show series name.
description	String	The TV show series description.
contentProviderExternalId	String	The TV show series content provider external ID.
contentProviderName	String	The TV show series content provider name.
shortName	String	The TV show series short name.
isMiniSeries	boolean	Indicates whether the series is a mini-series and links to one season only.
shortDescription	String	The TV show series short description.
numberOfSeasons	Number	The number of seasons in the TV show series.
year	Number	The production year of the TV show series.
studio	String	The studio that produced the TV show series.
countries	Country[]	The countries which produced the TV show series.
genres	Genre[]	The TV show series genres.
contributors	Contributor[]	The list of contributors to the TV show series (director, actors etc.).
prName	String	The TV show series parental rating name.
prLevel	Number	The TV show series parental rating level.
parentalAdvisories	String	Warnings on content that is offensive, violent or otherwise considered unsuitable for children.
extrafields	Extrafield[]	The TV show series extra fields.
attachments	Attachment[]	The TV show series attachments.

TvShowSeriesPage

since 8.1

An entity that represents a TV show series page.

Properties

Name	Type	Description
id	Number	The page internal id.
externalId	String	The page external ID.
name	String	The page name.
tvShowSeries	TvShowSeries	The TV show series entity.
template	String	The template name for the TV show series page.
isSecured	Boolean	Indicates whether the TV show series page has security restrictions.
securityGroups	Group[]	A list of security groups/services associated with the page.
status	Number	The status of the page: Published (2), Announced (3), Removed from Service (4).
description	String	The page description.
parentId	String	The parent ID of the page.
extrafields	Extrafield[]	The TV show series page extra fields.
attachments	Attachment[]	The TV show series page attachments.

TvShowTicket

since 8.2

An entity that represents a TV Show ticket. A TV Show ticket is created for an account when a Season Pass for a TV Show season is ordered. The TV Show ticket specifies the TV Show season page ordered as well as the dates during which the season's episodes can be accessed. Whenever a user attempts to access a TV Show episode, the account's TVShowTickets are checked to determine whether the account has the right to do so.

Properties

Name	Type	Description
id	Number	The ID of the ticket.
orderId	Number	The ID of the order.
householdId	Number	The ID of the account (household) that created the order.
tvShowPageId	Number	The ID of the TV Show page that was ordered.

Name	Type	Description
startDate	Number	The start date of the TV Show ticket (calculated in milliseconds from 1/1/1970).
endDate	Number	The end date of the TV Show ticket (calculated in milliseconds from 1/1/1970).
rentalPeriod	Number	The period of time during which the user can watch an episode in the season starting from when the user first watches the episode. After this time has elapsed the user can no longer watch the episode. The time unit is specified by rentalPeriodUnit. (When 0 is returned, indicates that the rental period was not specified).
rentalPeriodUnit	Number	The unit of measurement used for the rental period. Values: <ul style="list-style-type: none"> • 2 - Minutes • 3 - Hours • 4 - Days (0 - Can be used when the rental period is not specified)
downloadPeriod	Number	The period of time during which the user can download an episode starting from when the user first downloads the episode. After this time has elapsed the user can't download the episode any more. The time unit is specified by downloadPeriodUnit.
downloadPeriodUnit	Number	The unit of measurement used for the download period. Values: <ul style="list-style-type: none"> • 2 - Minutes • 3 - Hours • 4 - Days (0 - Can be used when the download period is not specified)
deliveryMode	String	The TV Show ticket delivery mode. [Values: streaming, download, converged, est]
status	String	The TV Show ticket status. [Values: active, expired]. Expired is only relevant for tickets where deliveryMode = est
restrictedToTerminalCategories	TerminalCategory[]	The device categories allowed for the TV Show ticket.
pageExternalId	String	The TV show season page external ID through which the season was ordered.

Name	Type	Description
seasonExternalId	String	The external ID of a TV show season that was ordered regardless of the page through which it was ordered.
definition	String	The pricing option content definition. Replaces the parameter definitionId.

See Also

- [GetTVShowTicketList](#)

UniqueVideoInfo**since 7.0**

An entity that represents the specific details of a video that is part of an AggregatedVideo.

Properties

Name	Type	Description
externalId	String	The external ID of the video page.
pricingMatrixId	Number	The ID of the pricing matrix attached to this video.
discountId	String	The external ID of a discount attached to the video. For more details about the use of this parameter see the Discount entity specification.
definition	String	The stream definition of the video.
encodings	Encoding[]	A list of encoding formats filtered by the definitions.
status	Number	The video status: <ul style="list-style-type: none"> Published (2) Announced (3) Removed from Service (4)
removalDate	Number	The time the service will be removed from service.
securityGroups	Group[]	A list of security groups/services.
isSecured	Boolean	A boolean that indicates whether the video is secured.

Name	Type	Description
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the movie allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific movie page, that limit will be returned, otherwise the global system default limit for VOD will be returned. When session management is not handled by RiGHTv (i.e. SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.
extrafields	Extrafield[]	The list of extra fields. This list contains the extra fields of the video page and of the series (if this page is part of a series).
attachments	Attachment[]	The list of attachments. This list contains the extra fields of the video page and of the series (if this page is part of a series).

See Also

- [GetUnifiedList](#)
- [AggregatedVideo](#)

Video

since 5.3

An entity that represents information about a video page.

Properties

Name	Type	Description
id	Number	The ID of the video page (from RiGHTv).
name	String	The name of the video (from the RiGHTv).
template	String	The template of the video (vod_service / svod_service / etc.).
description	String	The description of the video (from RiGHTv).
duration	Number	The duration of the asset in milliseconds.
categoryId	Number	The category ID (parent page ID from RiGHTv).
prLevel	Number	The video parental rating level.
prName	String	The video parental rating name.

Name	Type	Description
advisories	String	The video parental rating advisories.
windowStart	Number	The license start date (in milliseconds from 1/1/1970).
windowEnd	Number	The license end date (in milliseconds from 1/1/1970).
assetId	Number	The ID of the asset (from the VO Management Console).
pricingMatrixId	Number	The ID of the pricing matrix attached to this video. If the video is a video package movie, the value of pricingMatrixId will be empty.
chapters	AssetChapter[]	An array of chapters of the video.
metadata	Metadata[]	An array of metadata of the video.
extrafields	Extrafield[]	An array of extra fields of the video.
attachments	Attachment[]	An array of attachments of the video. This list contains the attachments of the page, video asset and series.
encodings	Encoding[]	An array of encodings.
tvShowReference	TvShowReference	The reference to a TV show series if this asset is linked to a TV show series.
externalChannelId	String	The external channel ID associated with the asset associated to this video service.
reviewerRating	Number	The external reviewer rating for this content.
isSecured	Boolean	A boolean that indicates whether the video is secured.
definition	String	The video definition.
adsInfo	String	Any advertisement related information for the video.
contentProvider	String	The video content provider name.
removalDate	Number	The time the video will be removed from service.

Name	Type	Description
simultaneousViewsLimit	Number	The maximum number of simultaneous views of the movie allowed for the account. Valid values: positive whole numbers. When SimultaneousViewsRiGHTvControl is set to "On" in the Configuration, if a limit has been applied to the specific movie page, that limit will be returned, otherwise the global system default limit for VOD will be returned. When session management is not handled by RiGHTv (i.e. SimultaneousViewsRiGHTvControl is set to "Off" or "Customized"), a null value will be returned.
keywords	String	The keywords attached to the video. Keywords are free text separated by semicolons.
broadcastTime	Number	The time the video was broadcast.
discountId	String	The external ID of the discount attached to the video. For more details about the use of this parameter see the Discount entity specification
genreEntityList	Genre[]	An list of genres of the video. Since 8.0 SR4.
genres	Number[]	An array of genre IDs of the video.
seriesName	String	The series name, if the video is part of a series.
assetExternalId	String	The external ID of the asset attached to the video.
type	String	The video type, possible values: <ul style="list-style-type: none"> • "movie" • "vod package movie"
shortName	String	A short name of the video.
rentalPeriodUnit	Number	The time unit of the video package movie rental period (empty when this is not a video package movie): <ul style="list-style-type: none"> • Minutes (2) • Hours (3) • Days (4)
rentalPeriod	Number	The rental period of the video package movie (empty otherwise).

Name	Type	Description
seriesNumberOfEpisodes	Number	The number of episodes in the series, if the video is part of a series.
year	Number	The year this content was produced.
plannedPublishDate	Number	The time when the video is planned to be published.
reviews	Review[]	A list of reviews attached to the video.
flags	Number	A bitwise description of the video properties: <ul style="list-style-type: none"> • Ads Pre-Roll (1) • Ads Post-Roll (2)
contentProviderExternalId	String	The video content provider external ID.
awards	Award[]	A list of awards attached to the video.
seriesSeason	String	The series season name, if the video is part of a series.
externalId	String	The external ID of the video.
episodeId	String	The ID of the episode, in case the video is part of a series. Introduced in 7.0 SR1.
status	Number	The video's status: <ul style="list-style-type: none"> • Published (2) • Announced (3) • Removed from Service (4)
allowedTerminalCategories	TerminalCategory[]	A list of terminal categories that are allowed to watch the video.
securityGroups	Group[]	A list of security groups/services.

See Also

- [LimitedVideo](#)
- [GetVideo](#)
- [GetVideoList](#)
- [OrderVideo](#)
- [Discount](#)

VideoAsset

since 6.0

An entity that represents information about the video asset in RiGHTv.

Properties

Name	Type	Description
id	Number	The ID of video asset.
name	String	The name of the video asset.
description	String	The description of the video (from RiGHTv).
contentProvider	String	The content provider Name.
duration	Number	The duration of the asset in seconds.
prLevel	Number	The video asset parental rating level.
prName	String	The video asset parental rating name
advisories	String	The video asset parental rating advisories.
windowStart	Number	The License start date (in milliseconds from 1/1/1970).
windowEnd	Number	The License end date (in milliseconds from 1/1/1970).
year	Number	The video asset production year.
series	Series	The series related to the video asset, in case the video asset is part of a series.
episodeId	String	The ID of the episode, in case the video asset is part of a series.
externalId	String	The video asset identifier as provided by the content provider.
mediaFiles	MediaFile[]	An array of media files of the video asset.
chapters	AssetChapter[]	An array of chapters of the video asset.
extrafields	Extrafield[]	An array of extra fields of the video asset.
attachments	Attachment[]	An array of attachments of the video asset. In case the video asset is part of a series, the list also contains the series' attachments.
countries	Country[]	An array of countries of the video asset.
contributors	Contributor[]	An array of contributors of the video asset.
externalChannelId	String	The external ID of the live channel on which the content was broadcast.

Name	Type	Description
broadcastTime	Number	The broadcast time of the asset in the live channel, in milliseconds.
genres	Number[]	The list of the video asset's genre IDs.
reviews	Review[]	A list of reviews attached to the video.
tvShowReference	TvShowReference	The reference to a TV show series if this asset is linked to a TV show series.
genreEntityList	Genre[]	An list of genres of the video asset. Since 8.0 SR4.
keywords	String	The keywords attached to the asset. Keywords are free text separated by semicolons.
awards	Award[]	A list of awards attached to the video.
contentProviderExternalId	String	The content provider external ID.
shortDescription	String	A short description of the video asset.
shortName	String	The short name of the video asset.
allowedTerminalCategories	TerminalCategory[]	A list of terminal categories that are allowed to watch the video asset.

See Also

- [GetVideoAsset](#)

VideoCard**since 5.3**

An entity that represents a video card in the system. The video card defines a number of movies that can be purchased, and may be restricted to specific categories. It can be sold until its inactivation date, and used until:

- Its expiration date passed.
- The number of days set as duration have passed since the purchase date.

A video card cannot be updated via administration. A video card is a multiple entry ticket with which you can order several videos.

Properties

Name	Type	Description
id	Number	The ID of the video card.
name	String	The name of the video card.
description	String	The description of the video card.

Name	Type	Description
price	Number	VOD Card's rental price. Contains -1 if rental mode is not supported. (Number Type: double).
originalPrice	Number	VOD Card's rental price before discount. (Number Type: Double).
windowEnd	Number	The expiration date (in milliseconds from 1/1/1970). Can be -1 if not set.
numVideos	Number	The number of videos that can be purchased using this card (only after purchasing this card).
duration	Number	The duration from the moment of purchase (in milliseconds) in which the card can be used. Can be -1 if not set.
limitOrderingOnlyOnce	Boolean	Indicates whether to prevent the end user from ordering the content again from the same VOD card. Since 8.0 SR3.
deliveryMode	String	The VOD card delivery mode type. Possible values: streaming, download, converged. (Not relevant for eST. eST support is managed through estPrice).
inactivationDate	Number	The date in which the video card becomes inactive (in milliseconds from 1/1/1970). Can be -1 if not set.
rentalPeriod	Number	The video card rental period in the specified rental period unit. (When 0 is returned, indicates that the rental period was not specified).
categoriesExternalIds	String[]	The external IDs of the categories for which this video card ticket applies (page in RiGHTv).
categoryId	Number	The category ID for which this video card ticket applies (page in VO Management Console). Deprecated since 7.1
rentalPeriodUnit	Number	Rental period unit, valid options: (2) Minutes (3) Hours (4) Days. (0 - Can be used when the rental period is not specified)
estPrice	Number	VOD Card's eST price. Contains -1 if eST mode is not supported. (Number Type: double). Since 8.0 SR3.

Name	Type	Description
downloadPeriod	Number	The video card download period in the specified download period unit.
estOriginalPrice	Number	VOD Card's eST price before discount. (Number Type: Double). Since 8.0 SR3.
downloadPeriodUnit	Number	Download period unit, valid options: (2) Minutes (3) Hours (4) Days. (0 - Can be used when the download period is not specified)
activationDate	Number	The date in which the video card becomes active (in milliseconds from 1/1/1970). Can be -1 if not set.
externalId	String	The external ID of the video card.

See Also

- [GetVideoCardList](#)

VideoCardTicket

since 5.3

An entity that represents a video card ticket. Once a video card is ordered, it is assigned to a specific subscriber using the ticket. The ticket's start and end date already take into consideration the video card definitions of expiration date and duration. The number of consumed movies is saved in the ticket, determining how many movies have been ordered on the card thus far. An entity that represents a ticket for a video card.

Properties

Name	Type	Description
id	Number	The ID of the ticket.
videoCardId	Number	The ID of the video card.
name	String	The name of the video card.
description	String	The description of the video card.
price	Number	The price paid for the video card. (Number Type: double).
windowStart	Number	The availability start date (in milliseconds from 1/1/1970).
windowEnd	Number	The expiration date (in milliseconds from 1/1/1970).
numVideos	Number	The number of videos that can be purchased using this ticket.

Name	Type	Description
numConsumed	Number	The number of videos already purchased using this ticket.
categoriesExternalIds	String[]	The external IDs of the categories for which this video card ticket applies (page in RiGHTv).
businessModel	String	The VOD card business model. Possible values: rental, eST. Since 8.0 SR3.
rentalPeriod	Number	The video card rental period in the specified rental period unit. (When 0 is returned, indicates that the rental period was not specified).
rentalPeriodUnit	Number	Rental period unit, valid options: (2) Minutes (3) Hours (4) Days. (0 - Can be used when the rental period is not specified)
downloadPeriodUnit	Number	Download period unit, valid options: (2) Minutes (3) Hours (4) Days. (0 - Can be used when the download period is not specified)
downloadPeriod	Number	The video card download period in the specified download period unit.
categoryId	Number	The category ID for which this video card ticket applies (page in RiGHTv). Deprecated since 7.1

See Also

- [GetVideoCardTicketList](#)
- [OrderVideoCard](#)

VideoDownloadInfo

since 6.1

An entity that represents the video download information.

Properties

Name	Type	Description
id	Number	The ID of the movie item to be played. In case of attachment of TV show series/TV show season it will be empty.
ip	String	The IP of the unicast stream containing the video required.

Name	Type	Description
port	Number	The port of the unicast stream containing the video required.
mediaFileExternalId	String	The video media file external ID.
mediaImageUrl	String	The full URL for playing the video.
identifier	String	The CAS identifier of the media file in some CAS integrations.
token	String	A token that should be used for retrieving the Play-Ready license.
mediaFileDuration	Number	The duration of the media file.

Comments

- Notice that the content of this entity relies on the video server. While some return the full URL, some return the URL components (protocol, IP, port, etc.).

See Also

- [GetVideoDownloadInfo](#)

VideoPackage

since 6.2

An entity that represents a video package entity.

Properties

Name	Type	Description
id	Number	The internal ID of the video package page.
externalId	String	The external ID of the video package page.
name	String	The translated name of the video package page.
description	String	The translated description of the video package page.
price	Number	The price of the video package. (Number Type: double).
status	Number	The video package's status, possible values: <ul style="list-style-type: none"> Published (2) Announced (3) Removed from Service (4)
removalDate	Number	The planned (calculated) removal date of the package.

Name	Type	Description
packagePeriod	Number	The period of the video package. Calculated according to the packagePeriodUnit.
packagePeriodUnit	Number	The time unit of the package period, possible values: <ul style="list-style-type: none">• Minutes (2)• Hours (3)• Days (4)
rentalPeriod	Number	The period of the video package movie (unless overridden by the movie itself). Calculated according to the rentalPeriodUnit.
rentalPeriodUnit	Number	The time unit of the package movie rental period, possible values: <ul style="list-style-type: none">• Minutes (2)• Hours (3)• Days (4)
extrafields	Extrafield[]	An array of the video package extra fields.
attachments	Attachment[]	An array of the video package attachments.
estPrice	Number	The eST price of the video package. From 7.4 SR3. (Number Type: double).
deliveryMode	String	The video package delivery mode. Possible values: <ul style="list-style-type: none">• "Streaming"• "Download"• "Streaming and Download"• "Converged" (From RiGHTv version 7.3)
definition	String	The video package definition.
estStoreProductReferenceId	String	The ID of the App store eST product reference.
plannedPublishDate	Number	The time the package is planned to be published (in milliseconds from 1/1/1970).
storeProductReferenceId	String	The ID of the App store product reference.
securityGroups	Group[]	A list of the video package security groups/services.

Name	Type	Description
estOnlineAvailabilityPeriodUnit	Number	The time unit of the eST Online Availability period. Values: <ul style="list-style-type: none">• 4 = days• 6 = months• 7 = years
estOnlineAvailabilityPeriod	Number	The period of time during which the content is available online for streaming and/or for download for an eST order from when the order is placed.

See Also

- [GetVideoPackage](#)
- [GetVideoPackageList](#)
- [OrderVideoPackage](#)

VideoPackageInfo

since 8.3

An entity that represents the order information for a video package.

Properties

Name	Type	Description
orderId	Number	The ID of the order.
orderDate	Number	The video package order date.
packageId	Number	The internal ID of the video package page.
packageExternalId	String	The external ID of the video package page.
packageName	String	The translated name of the video package page.
description	String	The translated description of the video package page.
price	Number	The price of the video package. (Number Type: double).
downloadPeriod	Number	The period during which movies in the video package can be downloaded.
downloadPeriodUnit	Number	The time unit of the download period. Values: <ul style="list-style-type: none">• 2 - Minutes• 3 - Hours• 4 - Days (0 - Can be used when the download period is not specified)

Name	Type	Description
rentalPeriod	Number	The period during which movies in the video package can be rented. The rental period of an individual movie in the video package overrides the rental period of the video package. (When 0 is returned, indicates that the rental period was not specified).
rentalPeriodUnit	Number	The unit of time of the ordered package in which the rental period is specified. Values: <ul style="list-style-type: none"> • 2 - Minutes • 3 - Hours • 4 - Days (0 - Can be used when the rental period is not specified)
deliveryMode	String	The video package delivery mode. Values: <ul style="list-style-type: none"> • Streaming • Streaming and Download • Download • Converged
definition	String	The movie package content definition (SD, HD, etc).

VideoPlayingInfo

since 5.3

An entity that represents the playing information of a video.

Properties

Name	Type	Description
id	Number	The ID of the movie page to be played. In case of attachment of TV show series/TV show season it will be empty.
ip	String	The IP of the unicast stream containing the video required.
port	String	The port of the unicast stream containing the video required.
protocol	String	The protocol in which this unicast is transmitted.
filename	String	The asset file name.
url	String	The full URL for playing the video.
identifier	String	The string used for some CA integrations.

Name	Type	Description
token	String	A token that is used for retrieving the Microsoft PlayReady license.
position	Number	The current position of the video. Since 8.0 SR2 "position lead time" configuration is introduced and the value is calculated according to "position lead time".
adsUrl	String	A relative URL for advertisement content. For future use.

Comments

- The content of this entity relies on the video server. While some return the full URL, some return the URL components (protocol, IP, port, etc.)

See Also

- [GetVideoPlayingInfo](#)
- [GetAttachmentPlayingInfo](#)

VideoPurchaseInfo

since 8.1

An entity that represents the information for purchasing a page.

Properties

Name	Type	Description
id	Number	The page internal ID.
externalId	String	The page external ID.
name	String	The page name.
definition	String	The page definition. Relevant only for video page.
template	String	The template name for the page.
pricingOptions	ExtendedPricingOption[]	The list of pricing options available for the page. Relevant only for video page.
supportedTerminalModels	TerminalModel[]	The list of devices supported for the page. Relevant only for video page.
vodCards	VideoCard[]	The list of VOD Cards associated with the page.
services	Service[]	The list of services associated with the page.

Name	Type	Description
status	Number	The page status: <ul style="list-style-type: none">• Published (2)• Announced (3)
videoPackage	VideoPackage	The video package associated with the movie page that is part of the video package.(Since 8.3SR1)
seasonOfferOptions	SeasonOfferOption[]	A list of all Season Passes associated with the season page.
servicePlans	ServicePlan[]	The list of Service Plans that include the provided services. (Since 8.1 SR1)

VideoTicket

since 5.3

An entity that represents a ticket for a video.

Properties

Name	Type	Description
id	Number	The ID of the ticket.
assetId	Number	The ID of the asset (from RiGHTv).
videoId	Number	The ID of the video page (from RiGHTv).
name	String	The name of the video (taken from the RiGHTv).
template	String	The template of the video (vod_service / svod_service / etc.).
windowStart	Number	The availability start date (in milliseconds from 1/1/1970). If filled, indicates the ticket is active.
windowEnd	Number	The ticket expiration date (in milliseconds from 1/1/1970).
downloadPeriod	Number	The download period of the ticket. For eST VOD gift tickets the value returned is '-1' This value is not passed when replacing an eST gift.
currentEstDownloadingDevices	Number	The number of terminals which have downloaded the eST content.
resumePosition	Number	The last played position. For a movie that has not been watched yet, the position will be zero.

Name	Type	Description
deliveryMode	String	Parameter to distinguish the ticket's delivery mode, possible values are: <ul style="list-style-type: none">• "streaming"• "download"• "converged"• "est"
rentalPeriod	Number	The rental period of the ticket. For eST VOD gift tickets the value returned is '-1'. This value is not passed when replacing an eST gift. (When 0 is returned, indicates that the rental period was not specified).
videoPackageExternalId	String	The external ID of the video package that was ordered.
allowedViews	Number	The allowed number of views per video order.
videoExternalId	String	The external ID of the video related the video ticket.
rentalPeriodUnit	Number	The rental period unit of the ticket. Valid options: <ul style="list-style-type: none">• (2) Minutes• (3) Hours• (4) Days . (0 - Can be used when the rental period is not specified). For eST VOD gift tickets the return value will be '-1'. This value is not passed when replacing an eST gift.
downloadWindowEnd	Number	The ticket download end date.
attachment	Attachment	The attachment connected to the video.
downloadWindowStart	Number	The ticket download start date.
assetExternalId	String	The external ID of the asset related the video ticket.
allowedEstDownloadingDevices	Number	The number of allowed eST downloads (to unique terminals).
tvShowTicketId	Number	For a TV show episode that was bought as part of a season pass offer, the ID of the season pass. Null for a movie that is not a TV show episode, and for an episode that was not bought as part of a season pass offer.

Name	Type	Description
type	String	The VOD type, possible values are: <ul style="list-style-type: none">• "transaction"• "subscription"
prLevel	Number	The video parental rating level.
duration	Number	The duration of movie attached to media file of the video related the video ticket.
downloadPeriodUnit	Number	The download period unit of the ticket. Valid options: <ul style="list-style-type: none">• (2) Minutes• (3) Hours• (4) Days . (0 - Can be used when the download period is not specified). For eST VOD gift tickets the return value will be '-1'. This value is not passed when replacing an eST gift.
videoPackageName	String	The name of the video package that was ordered.
prName	String	The video parental rating name.
advisories	String	The video parental rating advisories.
status	String	Ticket status. Possible values are: <ul style="list-style-type: none">• "active"• "expired" . "expired" status is relevant only for tickets of video package movies. Any ticket that has not expired will be in "active" status. If the ticket has an active status but the windowStart date is empty, the ticket must be activated before the movie can be downloaded, using ActivateTicket.
tvShowSeasonExternalId	String	For a TV show episode, the external ID of a TV show season. Null for a movie that is not a TV show episode.

Comments

- It is recommended to use start_date/download_start_date and end_date/download_end_date and not relying on the rental/download periods. The download/rental periods can be rounded when adjusting the ticket.
- The ticket is active when windowStart is filled and the windowEnd was not yet reached.

See Also

- [GetVideoTicketList](#)
- [OrderVideo](#)

VideoTvShowSeason

since 8.1

An entity that represents a video TV show season.

Properties

Name	Type	Description
externalId	String	The TV show season external ID.
name	String	The TV show season name.
seriesExternalId	String	The TV show series external ID.
seriesName	String	The TV show series name.
description	String	The TV show season description.
contentProviderExternalId	String	The TV show season content provider external ID.
contentProviderName	String	The TV show season content provider name.
shortName	String	The TV show season short name.
shortDescription	String	The TV show season short description.
numberOfEpisodes	Number	The number of episodes in the TV show season.
seasonNumber	Number	The TV show season number.
year	Number	The year the TV show season was produced.
studio	String	The studio that produced the TV show season.
countries	Country[]	The countries which produced the TV show season.
genres	Genre[]	The TV show season genres.
contributors	Contributor[]	The list of contributors to the TV show season (directors, actors, etc.).
prLevel	Number	The TV show season parental rating level.
prName	String	The TV show season parental rating name.

Name	Type	Description
parentalAdvisories	String	Warnings on content that is offensive, violent or otherwise considered unsuitable for children.
extrafields	Extrafield[]	The TV show season extra fields.
attachments	Attachment[]	The TV show season attachments.
seasonCategoryInfoList	TvShowCategoryInfo[]	The TV show series page in the VOD Catalog linked to this season.

VideoTvShowSeries

since 8.1

An entity that represents a video TV show series.

Properties

Name	Type	Description
externalId	String	The TV show series external ID.
name	String	The TV show series name.
description	String	The TV show series description.
contentProviderExternalId	String	The TV show series content provider external ID.
contentProviderName	String	The TV show series content provider name.
shortName	String	The TV show series short name.
isMiniSeries	Boolean	Indicates whether this TV show series is a miniseries. A miniseries contains only one season and links directly to the episodes.
shortDescription	String	The TV show series short description.
numberOfSeasons	Number	The number of seasons in the TV show series.
year	Number	The year the TV show series was produced.
studio	String	The studio that produced the TV show series.
countries	Country[]	The countries which produced the TV show series.
genres	Genre[]	The TV show series genres.

Name	Type	Description
contributor	Contributor[]	The list of contributors to the TV show series (directors, actors, etc.).
prLevel	Number	The TV show series parental rating level.
prName	String	The TV show series parental rating name.
parentalAdvisories	String	Warnings on content that is offensive, violent or otherwise considered unsuitable for children.
extrafields	Extrafield[]	The TV show series extra fields.
attachments	Attachment[]	The TV show series attachments.
seriesCategoryInfoList	TvShowCategoryInfo[]	The TV show series page in the VOD Catalog linked to this series.

VideoUrlInfo

since 8.2

An entity that represents video playing or download information.

Properties

Name	Type	Description
id	Number	The ID of the movie page to be played. If the movie page is an attachment to a TV show series/season page, the ID will be empty.
url	String	The full URL for playing or downloading the video.
identifier	String	The string used for some CA integrations.
position	Number	The current position of the video. Since 8.0 SR2 "position lead time" configuration is introduced and the value is calculated according to "position lead time".
token	String	A token that is used for retrieving the Microsoft Play-Ready license.
urlType	Number	Type of the URL. <ul style="list-style-type: none"> • 1 - streaming • 2 - download
adsUrl	String	A relative URL for advertisement content. For future use.

See Also

- [GetTvShowEpisodePlayingInfo](#)

VideoWish (deprecated)**since 7.0**

Deprecated since 8.5. Use the Wishlist functionality provided in The COMPASS Content Discovery Platform.

An entity that represents a video/videoPackage wish item.

Properties

Name	Type	Description
item	Page	The wish page. Can be a video or video package.
creationDate	Number	The date the wish was added to the wish list.
originalStatus	Number	The original status of the wish when the wish was added to the wish list. The original status can be: <ul style="list-style-type: none"> • (2) published • (3) announced
statusChangeDate	Number	The date the wish status changed.
source	String	The source from where the user adds the page to the wish list.
currentStatus	Number	The current status of the wish. Can be: <ul style="list-style-type: none"> • (1) created • (2) published • (3) announced • (4) removed • (5) deleted
name	String	The wish name.
type	Number	The wish type. Possible values: <ul style="list-style-type: none"> • (3) movie (Video) • (5) package (VideoPackage)

See Also

- [AddToWishList](#)
- [RemoveFromWishList](#)
- [GetWishList](#)

VodGift**since 6.2**

An entity that represents a video gift granted to a subscriber.

Properties

Name	Type	Description
id	Number	The ID of the gift (internal).
creationDate	Number	The gift creation date (in milliseconds from 1/1/1970).
expirationDate	Number	The gift expiration date (in milliseconds from 1/1/1970).
receiverName	String	The name of the person to whom the gift was granted.
buyerName	String	The name of the gift's buyer.
vendorId	String	The identifier of the granting vendor (in case of external vendor gift).
message	String	The text message sent along with the gift.
videoId	Number	The ID of the video page (from RiGHTv).
externalVideoId	String	The external ID of the video page.
videoName	String	The name of the video (taken from the RiGHTv).
rentalPeriod	Number	The rental period of the gift. (When 0 is returned, indicates that the rental period was not specified).
rentalPeriodUnit	Number	The time unit for rental period: <ul style="list-style-type: none">• Minutes (2)• Hours (3)• Days(4) (0 - Can be used when the rental period is not specified)
downloadPeriod	Number	The download period of the gift.
downloadPeriodUnit	Number	The time unit for download period: <ul style="list-style-type: none">• Minutes (2)• Hours (3)• Days(4) (0 - Can be used when the download period is not specified)

Name	Type	Description
status	Number	The gift status: <ul style="list-style-type: none"> • New (1) - A gift was granted but receiver has not yet viewed its details. • Viewed (2) - Receiver has viewed the details of the gift but not yet consumed it. Use the MarkViewedVodGift API to inform the system that the user viewed the gift. This will change the status of the VOD gift to Viewed. • In consumption (3) - Receiver consumed the gift and it is still available for watching. • Consumed (4) - Receiver consumed the gift and it is no longer available for watching. • Expired (5) - Receiver didn't consume the gift and it is no longer possible to consume it. • Refused (6) - Receiver has viewed the details of the gift and refused it. Use the MarkViewedVodGift API with refuse parameter set to true to inform the system that the user refused the gift. This will change the status of the VOD gift to Refused.
type	String	Free text defining the gift's type.
price	Number	The price value of the gift. (Number Type: Double).
replacementGiftPrice	Number	Replacement gift video price. Filled with value only when VOD gift was replaced. (Number Type: Double). Since 8.0 SR3.
canBeReplaced	boolean	A boolean indicates whether this gift can be replaced or not.
replacementGiftVideoName	String	Replacement gift video name. Filled with value only when VOD gift was replaced. Since 8.0 SR3.
deliveryMode	String	The VOD gift delivery mode type. Possible values: streaming, download, converged, est (est is supported for VOD gift since 7.4 SR3).

See Also

- [GetVodGiftList](#)
- [MarkViewedVodGift](#)
- [OrderVideoGift](#)

Actions

This chapter describes the actions exposed by the RTE.

ActivateTicket since 6.1

An action used to activate the ticket of a downloaded movie for viewing or of a video package movie, in case the video package was not activated during the order.

Parameters

Name	Type	Description
ticket_id	Number	ID of the ticket to activate.

Returns

Name	Description
Result	

Throws

Name	Description
CP_HOUSEHOLD_NO_TERMINALS	Household has no terminals defined in the CAS.
INVALID_TICKET	No ticket was found to be activated, or the ticket has expired, or the ticket does not have the proper delivery mode.

Examples

```
1 | ActivateTicket?ticket_id=123
```

See Also

- [VideoTicket](#)
- [VideoAsset](#)

AddBookmark since 5.3

An action used to add a bookmark to an asset.

Parameters

Name	Type	Description
video_id	Number	The ID of the video page as defined in RiGHTv.

Name	Type	Description
position	Number	The position of the bookmark in milliseconds from the beginning of the video.
name	String	The name for the bookmark. The default is "Bookmark".
author	String	The author of the bookmark. The default is the current member name.
pin_code	String	The purchase PIN code of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_VIDEO_ID	An invalid video ID (video_id) was supplied.

Examples

```
1 AddBookmark?video_id=123&position=12000
```

See Also

- [Bookmark](#)
- [GetBookmarkList](#)
- [DeleteBookmark](#)

AddPackageSubscription*since 5.3*

An action used to add a channel package subscription to the logged-in household. When adding a channel package subscription that includes channels to which the account was previously subscribed, if the npvrDeletionDelay period has not passed since the account unsubscribed from those channels, deleted recordings belonging to those channels are revived, as long as the account is subscribed to an NPVR 2.0 service. All recordings will be revived even if the account's quota is exceeded.

Parameters

Name	Type	Description
package_id	Number	The ID of the package to add.
external_id	String	The external id of the package to add.

Name	Type	Description
pin_code	String	The purchase PIN of the Member performing the action. The default is the logged-in member.
planned_end_date	Number	A future planned end date for which the subscription will end (in milliseconds since 1/1/1970).
store_id	String	The ID of the app store.
purchase_reference	String	The reference to a purchase made for the specified video at the App store. Use the POST method to pass this parameter. (Mandatory if store_id is specified)
store_product_reference	String	The reference assigned to the product purchased at a particular App store.

Returns

Name	Description
Result	

Throws

Name	Description
CHANNEL_PACKAGE_BELONGS_TO_UNREGISTERED_BOUQUET	Channel package belongs to unregistered bouquet.
EXCESS_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is higher than the product price.
EXPIRED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference has expired.
FAILED_TO_DEBIT_IN_EXTERNAL_SYSTEM	Debit failure of external system.
GENERAL_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) A general error which is not related to any of the other error codes.
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INSUFFICIENT_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is lower than the price of the purchased item.

Name	Description
INVALID_PACKAGE_ID	An invalid package ID (package_id) was supplied.
INVALID_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference is invalid (rejected either by the integration micro service or by the store).
INVALID_STORE_ID	The supplied store id does not exist.
PACKAGE_SUBSCRIPTION_ALREADY_EXIST	Household already subscribed to the specified channel package.
PAST_PLANNED_END_DATE	The Provided planned end date was already passed.
PAYMENT_METHOD_NOT_FOUND	Client payment details are not found in payment system. (For Non operator users only)
REDEEMED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference was already used.
SYNTAX_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) At least one of the input parameters is invalid or a syntax error. More information is detailed in the "inputErrors" of the error body.
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Comments

- There are several permutations available for using this API:
 - package_id
 - external_id

Examples

```
1 AddPackageSubscription?package_id=123
```

```
1 AddPackageSubscription?external_id=abc
```

See Also

- [Package](#)
- [PackageChannel](#)
- [PackageSubscription](#)
- [RemovePackageSubscription](#)
- [GetPackageList](#)
- [GetHouseholdPackageList](#)
- [GetPackageChannelList](#)

AddServicePlanSubscription

since 8.1

Subscribes the account to a Service Plan.(Since 8.1 SR1). When adding a Service Plan with a channel package that includes channels that were previously removed from the account, deleted recordings belonging to channels are not revived

Parameters

Name	Type	Description
external_id	String	The Service Plan external ID.
validateServicesSubscription	Boolean	When set to true, checks whether the account is already subscribed to one of the services in the Service Plan (SVOD or channel package). The default is false.
activation_date	Calendar	The date from which the Service Plan is activated for the subscriber. Only relevant when the parameter activationOption in the selected ServicePlan is set to ON_DATE.
store_id	String	The ID of the app store.
purchase_reference	String	The reference to a purchase made for the specified video at the App store. Use the POST method to pass this parameter. (Mandatory if store_id is specified)
store_product_reference	String	The reference assigned to the product purchased at a particular App store.

Returns

Name	Description
Result	

Throws

Name	Description
EXCESS_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is higher than the product price.
EXPIRED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference has expired.
FAILED_TO_DEBIT_IN_EXTERNAL_SYSTEM	Debit failure of external system.
GENERAL_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) A general error which is not related to any of the other error codes.
HOUSEHOLD_NOT_ACTIVE	Household is not active.

Name	Description
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INSUFFICIENT_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is lower than the price of the purchased item.
INVALID_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference is invalid (rejected either by the integration micro service or by the store).
INVALID_STORE_ID	The supplied store id does not exist.
PACKAGE_SUBSCRIPTION_ALREADY_EXIST	The account is already subscribed to a channel package within the Service Plan.
REDEEMED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference was already used.
SERVICE_PLAN_NOT_FOUND	The servicePlan requested could not be found.
SERVICE_PLAN_NOT_YET_PUBLISHED	Service plan publish date is in the future.
SERVICE_PLAN_OUT_OF_DATE	Service plan removal date is in the past or option is FIXED_BY_DATE and the fixed end date is in the past.
SERVICE_SUBSCRIPTION_ALREADY_EXIST	The account is already subscribed to a service within the Service Plan.
SUBSCRIPTION_ALREADY_EXIST	The servicePlan requested is already in use by the household.
SYNTAX_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) At least one of the input parameters is invalid or a syntax error. More information is detailed in the "inputErrors" of the error body.
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Examples

```
1 AddServicePlanSubscription?external_id=123
```

```
1 AddServiceSubscription?external_id=123&validateServicesSubscription=true
```

See Also

- [ServicePlan](#)
- [GetServicePlan](#)

AddServiceSubscription

since 5.3

Adds a service subscription to the logged-in account in RiGHTv. The account may be subscribed to only one service of NPVR, thus if it is already subscribed to an NPVR service, subscribing to another NPVR service will replace the old service with the new one. For NPVR 2.0, when replacing an NPVR service with a new NPVR service, if the quota used by the account is greater than the quota in the new NPVR service, replacement is allowed. No recordings will be deleted, however new recordings will not be possible until there is sufficient available quota. If the account was previously subscribed to an NPVR service and recordings were deleted when the account unsubscribed from the service, deleted recordings belonging to channels to which the account is subscribed are revived, if the npvrDeletionDelay period has not passed since the NPVR 2.0 service subscription was previously removed. All recordings will be revived even if the account's quota is exceeded. Deprecated behavior since 9.0. For NPVR 1.0 when replacing an NPVR service with a new NPVR service, the old NPVR service can't be replaced if the quota used by the account exceeds the maximum quota of the new service.

Parameters

Name	Type	Description
service_id	Number	The ID of the service to add.
pin_code	String	The purchase PIN of the Member performing the action. The default is the logged-in member.
external_service_id	String	The external ID of the service to add.
planned_end_date	Number	A future planned end date for which the subscription will end (in milliseconds since 1/1/1970)..
store_id	String	The ID of the app store.
purchase_reference	String	The reference to a purchase made for the specified video at the App store. Use the POST method to pass this parameter. (Mandatory if store_id is specified)
store_product_reference	String	The reference assigned to the product purchased at a particular App store.

Returns

Name	Description
Result	

Throws

Name	Description
EXCESS_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is higher than the product price.

Name	Description
EXPIRED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference has expired.
FAILED_TO_DEBIT_IN_EXTERNAL_SYSTEM	Debit failure of external system.
GENERAL_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) A general error which is not related to any of the other error codes.
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INSUFFICIENT_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is lower than the price of the purchased item.
INVALID_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference is invalid (rejected either by the integration micro service or by the store).
INVALID_SERVICE_ID	An invalid service ID (service_id) was supplied.
INVALID_STORE_ID	The supplied store id does not exist.
MAX_QUOTA_EXCEEDED	For NPVR service - Can't replace the old NPVR service with the new one as the used quota by this household is more than the max quota of the new service.
PAYMENT_METHOD_NOT_FOUND	Client payment details are not found in payment system. (For Non operator users only)
REDEEMED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference was already used.
SERVICE_SUBSCRIPTION_ALREADY_EXIST	Service subscription already exists.
SYNTAX_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) At least one of the input parameters is invalid or a syntax error. More information is detailed in the "inputErrors" of the error body.
UNSUPPORTED_OPERATION	In-App purchases are not supported. OR NPVR is not supported for In-App purchases

Comments

- There are several permutations available for using this API:
 - service_id
 - external_service_id

Examples

```
1 AddServiceSubscription?service_id=123
```

```
1 AddServiceSubscription?external_service_id=ldsa23
```

See Also

- [Service](#)
- [ServiceSubscription](#)
- [RemoveServiceSubscription](#)
- [GetServiceList](#)
- [GetHouseholdServiceList](#)

AddToWishList (deprecated)

since 7.0

Deprecated since 8.5. Use the Wishlist functionality provided in The COMPASS Content Discovery Platform.

An action used to add an item (video or video package) to the wish list for a specific household or member.

Parameters

Name	Type	Description
item_external_id	String	The external ID of the item (video or package) to be added to the wish list.
scope	String	Possible options: household or member. Wish list can be updated on the household or member level. The default is household.
source	String	Free text used as an indication of the source of the wish. For example a recommendation engine (such as COMPASS).

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_PAGE_ID	The supplied page external ID is invalid.
INVALID_PAGE_STATUS	The supplied page status is not announced or published.
INVALID_PAGE_TYPE	The supplied page is not of type video or package.

Name	Description
WISH_ALREADY_EXIST	The supplied wish is already in the household/member list.
WISH_LIST_IS_FULL	The supplied wish list has reached its maximum size and another item can not be added.

Examples

1 AddToWishList?item_external_id=123

1 AddToWishList?item_external_id=123&scope=member

1 AddToWishList?item_external_id=123&scope=household

See Also

- [VideoWish](#)
- [RemoveFromWishList](#)
- [GetWishList](#)

CancelPreOrder*since 8.0*

An action that allows users to cancel a pre-order. (Since 8.0 SR3)

Parameters

Name	Type	Description
pre_order_id	Number	The internal ID of the pre-order.

Returns

Name	Description
Result	

Throws

Name	Description
PRE_ORDER_NOT_EXISTS	Pre order in status PENDING doesn't exist for household.

Examples

1 CancelPreOrder?pre_order_id=123

See Also

- [PreOrder](#)

CheckSmartCard*since 5.3*

An action used to check if a specific smart card is registered with a terminal.

Parameters

Name	Type	Description
smart_card_id	String	The ID of the smart card.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_SMART_CARD_ID	Specified smart card (smart_card_id) is not registered with the calling terminal.

Examples

```
1 CheckSmartCard?smart_card_id=123456789ABCDEF
```

See Also

- [Terminal](#)

CloseSession (deprecated)*since 7.1*

since 7.4

An action used to close remote terminal session.

Parameters

Name	Type	Description
session_id	String	The session ID to close.
terminal_id	String	The ID of the terminal for which to close the session.
mac_address	String	The MAC address of the terminal for which to close the session.
serial_number	String	The serial number of the terminal for which to close the session.
smart_card_id	String	The smart card ID of the terminal for which to close the session.

Returns

Name	Description
Result	

Throws

Name	Description
SESSION_NOT_EXIST	The requested session does not exist.

Comments

- At least one of the terminal identifiers is mandatory.
- For more details on how to use the remote messaging mechanism please read "RiGHTv Remote Client PVR Developers Guide".

Examples

```
1 CloseSession?session_id=1234
```

See Also

- [OpenSession](#)
- [GetSessionList](#)
- [SendMessage](#)
- [PollMessage](#)
- [SendResponse](#)

CloseViewingSessions**since 7.3**

An action used to close an existing viewing session or sessions by the device which opened it (or them). When position and video_external_id parameters are provided, position will be marked for that video. When RiGHTv handles the viewing sessions (On/customized mode) it will implement the following logic for sessions on the device indicated in the request:

- When video external ID and/or channel ID are provided - RiGHTv will close either all the sessions of the video/channel or only the oldest session of the video/channel, depending on the "sessions" parameter. Both play and download sessions will be closed.
- When neither video external ID nor channel ID are provided - RiGHTv will close either all the play sessions or only the oldest play session of the device, depending on the "sessions" parameter. Download sessions will not be closed.

When RiGHTv doesn't manage the viewing sessions (Off mode) - it will not perform any logic, and will call a plugin to handle this request. Plugin implementation is customizable per project.

Parameters

Name	Type	Description
mac_address	String	The MAC address of the terminal that requested the viewing session.
serial_number	String	The serial number of the terminal that requested the viewing session.
smart_card_id	String	The smart card ID of the terminal that requested the viewing session.
video_external_id	String	The external ID of the movie page for which to close the viewing session. (When not provided, position will not be marked.)

Name	Type	Description
sessions	String	The number of sessions to close. Possible values: One, All. Default = One.
position	Number	The position in milliseconds from the start of the movie. (When not provided, position is not marked.)
channel_id	Number	The ID of the live channel for which to close the viewing session.

Returns

Name	Description
Result	

Throws

Name	Description
GENERAL_SERVER_ERROR	The limit simultaneous views feature is not enabled
INVALID_CHANNEL	The supplied live channel ID is invalid.
INVALID_TERMINAL	Either the terminal is not valid / registered for the logged in household, or that the passed terminal identifier do not match the logged in terminal.
INVALID_VIDEO_ID	The supplied video page ID is invalid.
TERMINAL_IDENTIFIER_MISMATCH	Terminal parameter doesn't match the logged in terminal.

Comments

- At least one of the terminal identification parameters (mac_address, serial_number, smart_card_id) must be supplied or user should login with terminal.

Examples

```
1 CloseViewingSessions?video_external_id=1234&mac_address=1234&sessions=All
```

```
1 CloseViewingSessions?serial_number=1234
```

CompletePaymentMethodRegistration

since 7.3

This action will validate and complete client payment registration.

Parameters

Name	Type	Description
transaction_id	String	Represents the transaction ID. This will be used to cancel the transaction.

Throws

Name	Description
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.

Examples

```
1 | CompletePaymentMethodRegistration?transaction_id=Product_872387
```

DeleteBookmark*since 5.3*

An action used to delete a bookmark belonging to an asset.

Parameters

Name	Type	Description
bookmark_id	Number	The ID of the bookmark.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_BOOKMARK_ID	An invalid bookmark ID was supplied.
PERMISSION_DENIED	The logged-in member does not have permission to delete another member's bookmarks.

Examples

```
1 | DeleteBookmark?bookmark_id=123
```

See Also

- [Bookmark](#)
- [GetBookmarkList](#)
- [AddBookmark](#)

DeleteMessage*since 5.3*

An action used to delete a message from the message list.

Parameters

Name	Type	Description
message_id	Number	The message ID.
message_external_id	String	The external ID of the message.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_MESSAGE_ID	An invalid message ID was supplied.

Examples

```
1 DeleteMessage?message_id=123
```

See Also

- [Message](#)
- [GetMessageList](#)

DeletePaymentMethod*since 7.3*

This action deletes the payment method from the external system.

Throws

Name	Description
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household.Relevant for non operator subscribers.

Examples

```
1 DeletePaymentMethod
```

DeleteRecording*since 7.4*

Deletes recordings from the logged-in account.For NPVR 1.0, if Personal Recording is enabled, deletes recordings from the logged-in profile.

For NPVR 2.0, one of the following permutations must be provided:

- Any combination of otf_recording_id, past, future
- otf_recording_ids

Recordings are permanently deleted immediately and are not affected by the npvrDeletionDelayperiod.

For NPVR 2.0:

- When only the otf_recording_id is specified, the specified recording for the logged-in account is deleted.

- When otf_recording_id indicates a recording that is part of a series and the future value specified is true all future/ongoing recordings of the series identified with this recording are deleted. No more recordings are created for this season. If the future value specified is true and the recording is not part of a series, then an exception is thrown.
- When otf_recording_id indicates a recording that is part of a series and the past value specified is true all past recordings of the series identified with this recording are deleted. If the recording specified is not part of a series, then an exception is thrown.
- When the future value specified is true, and otf_recording_id is not specified, all of the logged-in account's future/ongoing recordings are deleted.
- When the past value specified is true, and otf_recording_id is not specified, all of the logged-in account's past recordings are deleted.
- When both the past value and future value specified is true and otf_recording_id is not specified, all the recordings for the logged-in account are deleted.
- When otf_recording_ids is specified and either the past or future values are specified as true or otf_recording_id is specified, an exception is thrown.

When otf_recording_id or otf_recording_ids are specified and indicate an episode that is part of a season recording, and the future and past values specified are false, the episode will not be recorded again by the series recording job or if the episode is broadcasted again.

For NPVR 1.0 one of the following must be provided: recording_id, past, future.

For NPVR 1.0:

- When only the recording_id is specified, the specified recording for the logged-in account is deleted.
- When the future value specified is true, and recording_id is not specified, all of the logged-in account's future/ongoing recordings are deleted.
- When the future value specified is true and recording_id is specified, all future/ongoing recordings of the series identified with this recording are deleted. No more recordings are created for this season. If the recording specified is not part of a series, then an exception is thrown.
- When the past value specified is true, and recording_id is not specified, all of the logged-in account's past recordings are deleted.
- When the past value specified is true and recording_id is specified - all past recordings of the series identified with this recording are deleted. If the recording specified is not part of a series, than just the recording specified is deleted.
- If isPersonalRecordingSupported is to true in the Configuration, only recording belonging to the profile are deleted.

Parameters

Name	Type	Description
recording_id	Number	Deprecated since 9.0. The ID of the recording to delete. Mandatory for NPVR 1.0.
otf_recording_id	String	The ID of the recording. Relevant for NPVR 2.0 only. Since 9.0.
future	Boolean	Default value is false. Must not be provided when providing otf_recording_ids. Since 9.1 SR5.

Name	Type	Description
past	Boolean	Default value is false. Must not be provided when providing otf_recording_ids. Since 9.1 SR5.
otf_recording_ids	String	A list of recording IDs separated by commas. Season recordings cannot be deleted using this parameter. Relevant for NPVR 2.0 only. Since 9.1 SR5.
personal	Boolean	For NPVR 1.0, if isPersonalRecordingSupported is set to true in the Configuration, this parameter is set to true by default and only recordings belonging to the profile are deleted.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_RECORDING_ID	No program was found based on recording with the provided ID.
INVALID_SERIES	Recording ID provided is not part of a series. Can happen only if providing the recording ID and one or both flags.
MISSING_PARAMETER	One of the parameters must be provided: recording_id, future, past
PROFILE_DATA_UNAVAILABLE	Personal data cannot be displayed as the COMPASS profile is unavailable.
SERVICE_UNSUBSCRIBED	Household not subscribed to NPVR service
TICKET_NOT_FOUND	Household has no ticket for the provided recording ID.
UNSUPPORTED_OPERATION	Account has no NPVR model or for NPVR 1.0, if personal is set to true and isPersonalRecordingSupported is set to false in the Configuration.
UNSUPPORTED_OPERATION	Household has no NPVR model.

Examples

```
1 DeleteRecording?recording_id=123
```

```
1 DeleteRecording?recording_id=123&past=true
```

```
1 DeleteRecording?recording_id=123&future=true
```

```
1 DeleteRecording?future=true
```

```
1 DeleteRecording?past=true
```

```
1 DeleteRecording?past=true&past=true
```

See Also

- [RecordingTicket](#)
- [GetRecordingTicketList](#)

FinishPackageSubscription

since 7.4

An action used to finish a channel package subscription. It is available only for online customers. This action sets the planned end date of the channel package subscription to the nearest debit date (in the future). However, if the original channel package end date occurs before the next debit date, the service plan subscription will end at that service plan end date instead. The customer will be able to consume the package until that day because the customer has already paid for it. Example 1: Customer registered on 8/1; cancelled on 20/2 - Customer will still have the package until 8/3 00:00:00, as the customer was billed for the whole month on 8/2. Example 2: Customer registered on 9/1; cancelled on 8/2 - package will be available until 9/2 00:00:00.

Parameters

Name	Type	Description
external_package_id	String	The external ID of the package to finish its subscription .
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_PACKAGE_ID	An invalid package ID (package_id) was supplied.
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for online customers.

Name	Description
SERVICE_UNSUBSCRIBED	The logged-in household is not subscribed to the package.

Examples

```
1 FinishPackageSubscription?external_package_id=abc
```

See Also

- [Package](#)
- [PackageSubscription](#)
- [AddPackageSubscription](#)
- [RemovePackageSubscription](#)
- [GetPackageList](#)
- [GetHouseholdPackageList](#)

FinishServicePlanSubscription

since 8.2

An action used to end a service plan subscription. It is available only for online customers. This action sets the end date of the service plan subscription to the next debit date. However, if the original service plan end date occurs before the next debit date, the service plan subscription will end at that service plan end date instead. The customer will be able to consume the service plan until that date.

Parameters

Name	Type	Description
external_id	String	The external ID of the service plan to remove.
pin_code	String	The purchase PIN of the user performing the action. The default is the logged-in user.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	The account is not active.
HOUSEHOLD_PENDING_INACTIVATION	The account is in pending inactivation process (household planned inactivation date is in the past).
OPERATOR_SUBSCRIBER_NOT_ALLOWED	This action is only valid for online customers.
SERVICE_PLAN_NOT_FOUND	The supplied external service plan ID not found.

Name	Description
SERVICE_UNSUBSCRIBED	The logged-in household is not subscribed to the service plan, or the household's subscription to the service plan has not been activated (in which case removeServicePlanSubscription should be used instead).

Examples

1	FinishServicePlanSubscription?external_service_id=123abc
---	--

FinishServiceSubscription**since 7.4**

An action used to end a service subscription. It is available only for online customers. This action sets the planned end date of the service subscription to the nearest debit date (in the future). However, if the original service end date occurs before the next debit date, the service plan subscription will end at that service plan end date instead. The customer will be able to consume the service until that day as is already paid for.

Parameters

Name	Type	Description
external_service_id	String	The external ID of the service to remove.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_SERVICE_ID	The supplied external service ID is invalid.
OPERATOR_SUBSCRIBER_NOT_ALLOWED	This action is only valid for a online customer
SERVICE_UNSUBSCRIBED	The logged-in household is not subscribed to the service.

Examples

1	FinishServiceSubscription?external_service_id=123abc
---	--

See Also

- [Service](#)
- [ServiceSubscription](#)
- [AddServiceSubscription](#)
- [GetServiceList](#)
- [GetHouseholdServiceList](#)
- [RemoveServiceSubscription](#)

GetAggregatedVideo**since 7.1**

Returns the movie pages of the asset and purchase information in an aggregated video entity. Where an asset is associated with a movie page that is part of a movie package, and include_extended_info is set to true, that movie package is also returned as part of the extended object. If isMultiDefinitionSupported is set to true in the Configuration, then for each device model, the availableDefinitions parameter will contain all available content definitions for that device model including definitions that are lower than the content definition for the page. If isMultiDefinitionSupported is set to false, then the availableDefinitions parameter will be null. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.

Parameters

Name	Type	Description
external_asset_id	String	The external ID of the video asset as defined by the content provider.
external_content_provider_id	String	The external ID of the content provider.
external_root_ids	String	A semicolon separated list of external IDs of root pages, under which the movie pages of the asset will retrieved. Default ID is 1 (meaning all of the RiGHTv catalog).
statuses	String	A filtering option. Filtering according to statuses. A semicolon separated list of the following valid values: <ul style="list-style-type: none"> • published" • "announced"
definitions	String	A filtering option. Filtering according to definitions. A semicolon separated list of definitions. No limitation. The definitions cannot contain semicolons, because it is used as the delimiter.
delivery_mode	String	A filtering option. Filtering according to delivery mode. Single allowed value: "streaming" or "download".
services	String	A filtering option. Filtering according to services/groups (by external ID). A semicolon separated list of security groups (services).

Name	Type	Description
include_extended_info	Boolean	A boolean that indicates whether to include video purchase information and movie packages associated with the movie pages in the result. If true, returns ExtendedAggregatedVideo entity. If omitted or false (default), returns AggregatedVideo instead of ExtendedAggregatedVideo (used for backward compatibility). (Since 8.3 SR1).

Returns

Name	Description
AggregatedVideo	The action will return the movie pages of the asset in an aggregated video Entity.
ExtendedAggregatedVideo	The movie pages of the asset, movie packages associated with the movie pages, as well as purchase information in an aggregated video entity.

Throws

Name	Description
INVALID_VIDEO_ASSET_ID	The supplied asset doesn't exist in the system.
NO_MOVIE_WAS_FOUND	When no movie is found for the action call. If given parameters such as definitions, encodings etc. and the parameter does not exist, no movie page will be found, which will cause this error to be thrown. An invalid parameter exception is not thrown.

Comments

- When the supplied external_asset_id is not unique in the system external_content_provider_id should be supplied in order to identify the unique asset. if the external_content_provider_id is not supplied and the external_asset_id is not unique in the system GENERAL_SERVER_ERROR exception will be thrown.

Examples

```
1 GetAggregatedVideo?external_asset_id=123
```

```
1 GetVideoList?
  external_root_ids=abc&external_asset_id=1234&external_content_provider_id=1
```

See Also

- [AggregatedVideo](#)
- [ExtendedAggregatedVideo](#)

GetAggregatedVideoTicketList

since 8.3

Returns active video tickets purchased through different options (VOD package, TV Show Season Pass, SVOD, TVOD) in an aggregated list. The response will include:

- Active movie tickets which were ordered individually through VOD, SVOD or VOD cards.
- Active tickets for movies that were ordered using gifts, including tickets that were opened by the gift recipient and tickets that were opened automatically when the gift was granted by the operator.
- Active TV Show Season Pass tickets and their related active and expired episode tickets nested within the season. A TV Show ticket is active according to the rental availability period (rental) and to the eST online availability period (eST).
- Video (movie) packages and their related active and expired movies tickets nested within each movie package. A video package is considered active if it contains at least one active movie.

Returns

Name	Description
AggregatedTickets	An aggregated ticket that contains the following entities in the following order: <ul style="list-style-type: none"> • VideoTickets: A list of active video tickets for movies ordered through SVOD, TVOD, or VOD card). The order of the tickets will be according to the expiration date (soonest first). • AggregatedTvShowTickets : A list of active TV Show tickets and their active and expired episode tickets. The order of the tickets will be according to the expiration date (soonest first). The nested episode tickets will be ordered by episode number (ascending). • AggregatedVideoPackageTickets: Order information for active video packages and a list of active and expired tickets for videos in each video package. The order of the tickets will be according to the expiration date (soonest first). The order of the movies within the package will be according to the display order in the VOD catalog.

Examples

```
1 | GetAggregatedVideoTicketList?
```

See Also

- [AggregatedTickets](#)

GetAttachmentDownloadInfo

since 6.1

Returns the download information for an attachment. Video attachments are assets; either encrypted or unencrypted, and are used for example, as movie trailers or bonus movies.
Precedence: Page (1), TV show (2), PPV event (3).

Parameters

Name	Type	Description
page_id	Number	The page ID. (Mandatory - see comments)
external_page_id	String	The external page ID. (Mandatory - see comments)

Name	Type	Description
series_external_id	String	The TV show series external ID. (Mandatory - see comments)
season_external_id	String	The TV show season external ID. (Mandatory - see comments)
attachment_name	String	The name of the attachment.
encoding	String	The encoding format of the attachment (e.g. "MPEG-2").
definition	String	The content definition of the media file for which to return the attachment download info that must be equal to or lower than the definition specified for the page. If no ticket exists with a matching media file of the specified definition: if Multi Definition Playback is not enabled in the Configuration, the content definition is ignored and the highest priority media file will be returned, if Multi Definition Playback is enabled, an error will be returned. If content definition is not specified, the highest priority media file will be returned. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.
model_external_id	String	The external ID of the device model.
mac_address	String	The MAC address of the device that requested the playing info. Used to keep the terminal ID in the token.
serial_number	String	The serial number of the device that requested the playing info. Used to keep the terminal ID in the token.
smart_card_id	String	The smart card ID of the device that requested the playing info. Used to keep the terminal ID in the token.
ppv_event_external_id	String	The ppv event external ID. (Mandatory - see comments)

Returns

Name	Description
VideoDownloadInfo	The attachment download information.

Throws

Name	Description
INVALID_ATTACHMENT	Attachment name supplied does not exist or is not a video.
INVALID_DEFINITION	The supplied definition is invalid.
INVALID_ENCODING	The supplied encoding format is invalid.
INVALID_PAGE_ID	The supplied page ID is invalid.
INVALID_TERMINAL	Either the terminal is not valid / registered for the logged in household, or the passed terminal identifier does not match the logged in terminal.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.
NO_MATCHING_MEDIA_FILE_FOR_DEFINITION	There is no media file that matches the definition specified.
NO_SUITABLE_MEDIA_FILE	No suitable media file for the given properties.
PPV_EVENT_EXPIRED	PPV Event was expired.
PPV_EVENT_NOT_FOUND	PPV Event was not found.
TV_SHOW_NOT_EXISTS	The TV show series/season does not exist.
UNSUPPORTED_OPERATION	PPV Event feature is not supported.

Comments

- Mandatory: To retrieve attachment download info for a:
 - Page, specify page_id or external_page_id
 - TV Show Series, specify series_external_id
 - TV Show Season, specify season_external_id
 - PPV Event, specify ppv_event_external_id
- If more than one of the identifiers are supplied, precedence will be 1) Page, 2) TV Show Season, 3) TV Show Series, 4) PPV Event.
- There are several permutations available for using this API. At least one of the following must be provided:
 - model_external_id - recommended
 - encoding - if this parameter is specified, the Multi Definition Playback will not work correctly.
- New integrations are encouraged to use model_external_id in all VOD related APIs. Encoding and definition parameters should not be used since model_external_id encapsulates both parameters and more. However, If the system was upgraded from an older version, model_external_id should not be used, unless the client (e.g. front-end application) is updated to use model_external_id in all relevant APIs.

Examples

```
1 GetAttachmentDownloadInfo?page_id=123&attachment_name=preview&encoding=MPEG-2
1 GetAttachmentDownloadInfo?external_page_id=abc&attachment_name=preview&encoding=MPEG-2
1 GetAttachmentDownloadInfo?page_id=123&attachment_name=preview&model_external_id=PC
1 GetAttachmentDownloadInfo?
page_id=123&attachment_name=preview&model_external_id=PC&definition=HD
1 GetAttachmentDownloadInfo?page_id=123&attachment_name=preview&model_external_id=tablet
1 GetAttachmentDownloadInfo?
page_id=123&attachment_name=preview&model_external_id=PC&definition=HD&encoding
=MPEG-2
```

See Also

- [GetVideoDownloadInfo](#)
- [GetVideoPlayingInfo](#)
- [VideoDownloadInfo](#)
- [GetAttachmentPlayingInfo](#)

GetAttachmentMediaFileList

since 7.0

An action used to return all media files information related to the provisioning quality of the media files of all attachments of a specific video. The action is used to retrieve information about media files of attachments (bonuses or trailers) in order to play the media file that best fits the STB capabilities.

Parameters

Name	Type	Description
external_video_id	String	The external ID of video page.
attachment_name	String	The attachment name to filter by.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
AttachmentMediaFile[]	

Throws

Name	Description
INVALID_ATTACHMENT	The supplied attachment name is invalid.
INVALID_VIDEO_ID	The supplied video ID is invalid.

Examples

```
1 | GetAttachmentMediaFileList?external_video_id=abc
```

```
1 | GetAttachmentMediaFileList?external_video_id=abc&attachment_name=trailer
```

GetAttachmentPlayingInfo

since 5.3

Returns the playing information for an attachment. Video attachments are assets; either encrypted or unencrypted, and are used for example, as movie trailers or bonus movies.

Parameters

Name	Type	Description
page_id	Number	The internal page ID. (Mandatory - see comments)
external_page_id	String	The external page ID. (Mandatory - see comments)
series_external_id	String	The TV show series external ID. (Mandatory - see comments)
season_external_id	String	The season external ID. (Mandatory - see comments)
attachment_name	String	The name of the attachment.
encoding	String	The encoding format of the attachment (e.g. MPEG-2).
terminal_id	Number	The device ID.
definition	String	The content definition of the media file for which to return the attachment playing info, that must be equal to or lower than the definition specified for the page. If no ticket exists with a matching media file of the specified definition: if Multi Definition Playback is not enabled in the Configuration, the content definition is ignored and the highest priority media file will be returned, if Multi Definition Playback is enabled, an error will be returned. If content definition is not specified, the highest priority media file will be returned. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.
model_external_id	String	The external ID of the device model.

Name	Type	Description
mac_address	String	The MAC address of the device that requested the playing info. Used to keep the terminal ID in the token.
serial_number	String	The serial number of the device that requested the playing info. Used to keep the terminal ID in the token.
smart_card_id	String	The smart card id of the device that requested the playing info. Used to keep the terminal ID in the token.
ppv_event_external_id	String	The ppv event external ID. (Mandatory - see comments)

Returns

Name	Description
VideoPlayingInfo	

Throws

Name	Description
INVALID_ATTACHMENT	The attachment name supplied does not exist or is not a video.
INVALID_DEFINITION	The supplied definition is invalid.
INVALID_ENCODING	The supplied encoding format is invalid.
INVALID_PAGE_ID	The supplied page ID is invalid.
INVALID_TERMINAL	Either the terminal is not valid or not registered for the logged in household, or the passed terminal identifier does not match the logged in household.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.
NO_MATCHING_MEDIA_FILE_FOR_DEFINITION	There is no media file that matches the definition specified.
NO_SUITABLE_MEDIA_FILE	There are no suitable media files for the given properties.
PPV_EVENT_EXPIRED	PPV Event was expired.
PPV_EVENT_NOT_FOUND	PPV Event was not found.
TV_SHOW_NOT_EXISTS	The TV show series/season does not exist.
UNSUPPORTED_OPERATION	PPV Event feature is not supported.

Comments

- Mandatory: To retrieve attachment playing info for a:
 - Page, specify page_id or external_page_id
 - TV Show Series, specify series_external_id
 - TV Show Season, specify season_external_id
 - PPV Event, specify ppv_event_external_id
- If more than one of the identifiers are supplied, precedence will be 1) Page, 2) TV Show Season, 3) TV Show Series, 4) PPV Event.
- There are several permutations available for using this API. At least one of the following must be provided:
 - model_external_id - recommended
 - encoding - if this parameter is specified, the Multi Definition Playback will not work correctly.
- New integrations are encouraged to use model_external_id in all VOD related APIs. Encoding and definition parameters should not be used since model_external_id encapsulates both parameters and more. However, If the system was upgraded from an older version, model_external_id should not be used, unless the client (e.g. front-end application) is updated to use model_external_id in all relevant APIs.

Examples

```
1 GetAttachmentPlayingInfo?page_id=123&attachment_name=preview&encoding=MPEG-2
```

```
1 GetAttachmentPlayingInfo?
external_attachment_item_id=123&attachment_name=preview&encoding=MPEG-2
```

```
1 GetAttachmentPlayingInfo?
external_attachment_item_id=123&attachment_name=preview&model_external_id=PC
```

```
1 GetAttachmentPlayingInfo?
external_attachment_item_id=123&attachment_name=preview&model_external_id=PC&
definition=SD
```

```
1 GetAttachmentPlayingInfo?
external_attachment_item_id=123&attachment_name=preview&model_external_id=tab
let&encoding=MPEG-2
```

See Also

- [GetVideoPlayingInfo](#)
- [VideoPlayingInfo](#)
- [GetAttachmentDownloadInfo](#)

GetBasicRecordingTicketList

since 9.1

An action used to return a list of recordings tickets for the logged-in account containing reduced information. It is recommended to use pagination with this API for improved performance. When filtering recordings by start_date and end_date, for EPG based recordings if only the lead/lag duration falls in the requested time range specified, but the recorded program is outside the range, the recording will not be returned. One of the following permutations may be used for filtering by a specific season recording:

- series_id, season_name, channel_external_id
- series_ticket_id

If an episode was recorded as a stand-alone recording and not as part of a season recording, the recording will not be returned.

Parameters

Name	Type	Description
start_date	Number	The lowest time limit for the recording list, in milliseconds.
end_date	Number	The highest time limit for the recording list, in milliseconds.
order_by	String	An indicator on how to sort the list. Possible options are: "date" or "name". The "date" refers to the recording start date. For EPG based recordings, when sorting by date, the lead time and lag time are ignored. For EPG based recordings, when sorting by name, sorting is done by the name of the program in the EPG when the recording was created. Sorting ignores changes to the program name in the EPG. For time-based recordings sorting is done by the recording name. The default is order by "date".
order	String	An indicator whether to sort the list by ascending("asc") or descending ("desc"). The default is "asc".
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
series_ticket_id	Number	The ID of the ticket of the season recording order. When supplied, only recordings that belong to the supplied season recording are returned. series_ticket_id is returned by GetSeriesTicketList - SeriesTicket.id.
series_id	String	The ID of the series, used for filtering by a specific season recording. Must be supplied together with a channel_external_id, and season_name.
season_name	String	The name of the series season, used for filtering by a specific season recording. Must be supplied together with a channel_external_id, and series_id.

Name	Type	Description
channel_external_id	String	A single channel ID or multiple channel IDs using a comma delimiter. When used for filtering by a specific season recording, a single channel ID must be supplied together with series_id, and season_name. When used for filtering recordings by channel/s, provide either a single channel ID or multiple channel IDs without series_id, season_name, and series_ticket_id.

Returns

Name	Description
BasicRecordingTicket[]	

Examples

1 GetBasicRecordingTicketList?client=xml

1 GetBasicRecordingTicketList?channel_external_id={channelId1},{channelId2}&client=xml

See Also

- [OrderRecording](#)

GetBookmarkList

since 5.3

An action used to return list of bookmarks.

Parameters

Name	Type	Description
video_id	Number	The ID of the movie page as defined in RiGHTv. If the video exists, only the video bookmarks are returned. If the video does not exist or no video_id is provided, all the bookmarks of the logged-in household are returned.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Bookmark[]	

Examples

1 GetBookmarkList?video_id=123

See Also

- [AddBookmark](#)
- [DeleteBookmark](#)

GetBouquetList*since 6.0*

An action used to return a list of all the bouquets in RiGHTv.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Bouquet[]	

Examples

```
1 GetBouquetList
```

GetCaptchaKey*since 7.3*

An action used to create new CAPTCHA object in the system and return a new CAPTCHA key. The key along with the challenge text will be used by the server for confirmation.

Returns

Name	Description
CaptchaKey	

Examples

```
1 GetCaptchaKey
```

See Also

- [CaptchaKey](#)

GetChannelList*since 5.3*

An action used to return the channel list. Only Started channels are returned. In a multi-device environment, the client must provide its model ID in order to allow RiGHTv to validate that the channel can be viewed on the device model. This means that the device model is allowed to view the content (according to the rights permissions set on the channel) and its suitable technical information is defined for the channel. All Channels with the status started will be returned when model_external_id is supplied. Channel will be returned even if it is not viewable on the device model (model_external_id). In such a case, its technical information will be empty (sourceType

without a value is a good indication of this). In order to filter out channels which are not viewable for the terminal model passed, filter_unsupported_channels parameter should be provided. Backward compatibility is supported in old client interfaces where the model (model_external_id) is not provided (only channels that have one of IP/ DVB-SI / MHP technical format will be returned), yet the validation of rights management per device model is not carried out.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
bouquet_id	Number	The bouquet ID number. (PRECEDENCE 1)
bouquet_external_id	String	The bouquet external ID. (PRECEDENCE 2)
model_external_id	String	The model external ID. Mandatory in case filter_unsupported_channels is set to true.
filter_unsupported_channels	Boolean	A Boolean that indicates whether to filter out channels not viewable on the specified terminal model. If set to <code>true</code> and a <code>model_external_id</code> is provided, only channels that can be viewed on the specified terminal will be returned. The default is <code>false</code> . If set to <code>true</code> , <code>model_external_id</code> should be supplied.
max_pr_level	Number	The maximum level of parental rating allowed. When provided, returns channels with no parental rating or with a parental rating equal to or less than this level..
sector	String	The external ID of the sector. Must match the sector ID of the account. <code>isSectorsManagementSupported</code> in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
<code>Channel[]</code>	

Throws

Name	Description
<code>INVALID_BOUCLET_ID</code>	The bouquet ID is invalid.
<code>INVALID_SECTOR</code>	The sector ID specified does not match the sector ID of the account.

Name	Description
INVALID_TERMINAL_MODEL	The model ID is invalid.

Examples

1 GetChannelList

GetChannelMetaDataTable

since 8.4

Returns a selection of metadata of all channels, including channels that are turned off and channels that have been deleted. This API should only be used to get metadata for displaying channel recording and should not be used to get metadata for displaying live channels.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
ChannelMetaData[]	

Throws

Name	Description
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.

Examples

1 GetChannelMetaDataTable

GetChannelSupportedModelList

since 7.2

This action is used to return a list of supported terminal models for a specific channel.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
channel_external_id	String	The channel external ID.

Returns

Name	Description
TerminalModel[]	

Throws

Name	Description
INVALID_CHANNEL	The channel ID is invalid, or channel is off.

Examples

```
1 GetChannelSupportedModelList?channel_external_id=2552
```

GetClearLivePlayingInfo*since 7.4*

This action returns the channel information needed for playing an unencrypted channel. This operation does not require login, and therefore is allowed for anonymous users. Since 7.4 SR5.

Parameters

Name	Type	Description
channel_id	Number	The channel ID. Mandatory only if channel external ID is not provided
channel_external_id	String	The external ID of the channel. Mandatory only if channel_id is not provided.
model_external_id	String	The terminal model of the playing device. The model is used to calculate the best match channel format for the provided device. Mandatory.
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
LivePlayingInfo	

Throws

Name	Description
CHANNEL_NOT_CLEAR	Channel encrypted. For encrypted channels use GetLivePlayingInfo.
INVALID_CHANNEL	Channel doesn't exist, or channel is 'off'.
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.
NO_MATCHING_SOURCE_TYPE	No channel source type matches the terminal model.
SOURCE_TYPE_NOT_ALLOWED	Channel source type DVB-SI, MHP, IP is not allowed.

Examples

```
1 | GetClearLivePlayingInfo?channel_id=777&model_external_id=1234&client=xml
```

See Also

- [GetLivePlayingInfo](#)

GetClientData*since 5.3*

An action used to return client information saved in the server.

Parameters

Name	Type	Description
client_code	String	The unique client identifier (e.g. MAC address).
key	String	The "Key" of which getting the client data "value".

Returns

Name	Description
ClientData	

Examples

```
1 | GetClientData?client_code=00:11:22:33:44:55:66&key=BLOCKED_CHANNELS
```

See Also

- [SetClientData](#)

GetConfig*since 5.3*

An action used to return configuration values from the server.

Returns

Name	Description
Config	

Examples

1	GetConfig
---	-----------

GetContentDefinitionList

since 7.4

An action that returns all content definitions defined in the system. The definitions are returned sorted from the highest level to the lowest.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
String[]	

Examples

1	GetContentDefinitionList
1	GetContentDefinitionList?from=0&count=5

GetDiscount

since 6.2

Gets the discount entity according to the specified external ID.

Parameters

Name	Type	Description
external_id	String	The external ID of the discount.

Returns

Name	Description
Discount	

Throws

Name	Description
INVALID_DISCOUNT_ID	The supplied discount ID is invalid.

Examples

```
1 | GetDiscount?external_id=123
```

See Also

- [GetDiscountList](#)

GetDiscountList*since 6.2*

Gets the list of all discounts including current and future discounts.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Discount	

Examples

```
1 | GetDiscountList
```

See Also

- [GetDiscount](#)

GetDownloadLimitsAndCounters*since 9.1*

Gets the downloads limits and the account counters. See the RiGHTv Service Guide for more information. (Since 9.1 SR21)

Parameters

Name	Type	Description
external_asset_id	String	The asset external ID.
external_content_provider_id	String	The content provider external ID. Mandatory if external_asset_id does not uniquely identify the asset.

Returns

Name	Description
DownloadLimitsAndCounters	The monthly limit and counters are always returned. The asset limit and counters are only returned if the external_asset_id is provided.

Throws

Name	Description
GENERAL_SERVER_ERROR	The external_asset_id does not uniquely identify the asset.
INVALID_VIDEO_ASSET_ID	The supplied asset doesn't exist in the system.

Comments

- When supplied external_asset_id is not unique in the system external_content_provider_id should be supplied in order to identify the unique asset. If the external_content_provider_id is not supplied and the external_asset_id is not unique in the system. GENERAL_SERVER_ERROR exception will be thrown.

Examples

```
1 GetDownloadLimitsAndCounters?
  external_asset_id=asset&external_content_provider_id=contentProvider
```

```
1 GetDownloadLimitsAndCounters?external_asset_id=asset
```

```
1 GetDownloadLimitsAndCounters
```

See Also

- [DownloadLimitsAndCounters](#)

GetEstDownloadingTerminalList*since 7.3*

An action used to retrieve the list of terminals that downloaded a specific eST movie.

Parameters

Name	Type	Description
ticket_id	Number	The ID of the video ticket.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Terminal[]	The terminal List

Throws

Name	Description
INVALID_TICKET	The logged-in household has no ticket for this video, or the ticket has expired.

Examples

```
1 | GetEstDownloadingTerminalList?ticket_id=1234
```

See Also

- [GetTerminalList](#)

GetExtendedChannelList*since 8.6*

An action used to return a list of channels. Only channels that are in Started status are returned. Optionally, the list of channels can be filtered to include only channels that can be viewed on the requesting device, meaning that two conditions are met: a.) the channel's rights permissions allow the channel to be viewed on the device model, and b.) the channel has a technical format that is suitable for the device model. To apply that filter the `filter_unsupported_channels` must be set to true. On-the-fly technical formats will be filtered out from the returned Channels.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
bouquet_external_id	String	The bouquet's external ID.
max_pr_level	Number	The maximum level of parental rating allowed. When provided, channels with no parental rating are returned or channels with a parental rating equal to or less than this level.
filter_unsupported_channels	Boolean	A Boolean that indicates whether to filter out channels that are not viewable on the specified terminal model. If set to <code>true</code> only channels that can be viewed on the specified terminal model will be returned. The default is <code>false</code> .
model_external_id	String	The model external ID.
sector	String	The external ID of the sector. Must match the sector ID of the account. <code>isSectorsManagementSupported</code> in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
ExtendedChannel[]	

Throws

Name	Description
INVALID_BOUQUET_ID	The bouquet ID is invalid.
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.
INVALID_TERMINAL_MODEL	The model ID is invalid.

GetExtendedProgram*since 5.3*

An action used to return an extended program object.

Parameters

Name	Type	Description
program_id	Number	The program ID (internal).
program_external_id	String	The program external ID.
channel_external_id	String	The channel external ID.
filter_deleted	Boolean	Deprecated since 9.1. From RiGHTv 9.1, for NPVR 2.0, the program information is returned in RecordingTicket also for programs that are no longer active in the EPG. (For NPVR 1.0: Use false value to indicate that programs that are no longer active in the EPG should be returned. Default value is true.)

Returns

Name	Description
ExtendedProgram[]	

Throws

Name	Description
INVALID_CHANNEL_ID	The supplied channel ID is invalid.
INVALID_PROGRAM_ID	The supplied program ID is invalid.

Comments

- There are several permutations available:
 - program_external_id + channel_external_id
 - program_id
 - program_external_id

Examples

```
1 | GetExtendedProgram?program_id=123
```

```
1 | GetExtendedProgram?program_external_id=abc&channel_external_id=1
```

See Also

- [GetExtendedProgramList](#)
- [GetProgramList](#)

GetExtendedProgramList

since 7.0

An action used to return a list of extended programs according to one of two sets of parameters:

1. the external program ID and channel ID (channel ID is optional). or 2. the channel external id, and a time range represented by start date and end date

Parameters

Name	Type	Description
ext_program_channel_ids	String	A specific list of programs external IDs combined with channel IDs, separated by a configurable delimiter (default = ";"). The combined IDs will be separated by a different configurable delimiter (default = " "). Channel ID is optional. If channel ID was not delivered and there is more than one extended program matching the external program ID a GENERAL_SERVER_ERROR (too many rows in getEntity) exception will be thrown. No limitation, the external IDs cannot contain the delimiters.
channel_external_id	String	The channel external ID.
start_date	Number	The lowest time limit for the extended program list, in milliseconds.
end_date	Number	The highest time limit for the extended program list, in milliseconds.
filter_deleted	boolean	Deprecated since 9.1. From RiGHTv 9.1, for NPVR 2.0, the program information is returned in RecordingTicket also for programs that are no longer active in the EPG. (For NPVR 1.0: Use false value to indicate that programs that are no longer active in the EPG should be returned. Default value is true.)
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
ExtendedProgram[]	

Comments

- The result returns only the valid programs; if one or more of the IDs is invalid these programs will not be included in the result list. If all the IDs are invalid the result will be an empty list.
- The delimiters are configured and can be changed in the op_config.xml. See the Configuration Guide for more details.

Examples

```
1 | GetExtendedProgramList?ext_program_channel_ids=program1|channel1
```

```
1 | GetExtendedProgramList?ext_program_channel_ids=program1|channel1;video3|channel3
```

See Also

- [GetExtendedProgram](#)
- [GetProgramList](#)

GetExternalAccount*since 5.3*

An action used to return the logged-in household external account.

Returns

Name	Description
ExternalAccount	

Examples

```
1 | GetExternalAccount
```

GetFilteredProgramList*since 6.0*

An action used to return an array of searched programs entities in the specified channel(s), according to a specified period.

Parameters

Name	Type	Description
channel_list	Number	A list of channel IDs on which the programs reside, separated by semicolons.

Name	Type	Description
external_channel_list	String	A list of channel external IDs on which the programs reside, separated by semicolons. No limitation. The external IDs cannot contain semicolons, because it is used as the delimiter.
start_date	Number	The lowest time limit for the program list, in milliseconds.
end_date	Number	The highest time limit for the program list, in milliseconds.
genre_list	Number	A list of genre IDs that are defined for a program, separated by semicolons.
flags	Number	A bitwise description of the program properties: <ul style="list-style-type: none"> • isRecordable (2) • isCatchupEnabled (4) • isStartOverEnabled (8)
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
SearchedProgram[]	

Examples

```
1 GetFilteredProgramList?
  channel_list=123&start_date=1262323232231&end_date=1262325232231
```

GetGenreList

since 5.3

An action used to return a list of genres.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Genre[]	

Examples

1	GetGenreList
---	------------------------------

GetHousehold*since 5.3*

An action used to return the logged-in household information.

Returns

Name	Description
Household	

Examples

1	GetHousehold
---	------------------------------

GetHouseholdBouquetList*since 6.0*

An action used to return a list of bouquets related to the logged-in household.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Bouquet[]	

Examples

1	GetHouseholdBouquetList
---	---

GetHouseholdPackageList*since 5.3*

An action used to return the subscribed package list for the logged-in household.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
bouquet_id	Number	The bouquet ID.
bouquet_external_id	String	The bouquet external ID.
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
PackageSubscription[]	

Throws

Name	Description
INVALID_BOUQUET_ID	The bouquet ID is invalid.

Comments

- There are several permutations available for using this API:
 - bouquet_external_id
 - bouquet_id

Examples

1 GetHouseholdPackageList

1 GetHouseholdPackageList?bouquet_id=1

1 GetHouseholdPackageList?bouquet_external_id=ABC

See Also

- [Package](#)
- [GetPackageList](#)

GetHouseholdServiceList*since 5.3*

An action used to return the subscribed service list for the logged-in household including services he is subscribed to directly or through service plan.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
ServiceSubscription[]	

Examples

```
1 GetHouseholdServiceList
```

See Also

- [Service](#)
- [GetServiceList](#)

GetHouseholdSubscribedChannelList*since 8.4*

An action used to return all channels subscribed by logged in user. Channels of ordered PPV events are not included.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
ChannelReference[]	

Throws

Name	Description
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.

Examples

```
1 GetHouseholdSubscribedChannelList
```

GetHouseholdSubscriptionTransactionList

since 7.3

This action retrieves household subscription transactions.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
SubscriptionTransaction[]	

Throws

Name	Description
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households.

Examples

```
1 | GetHouseholdSubscriptionTransactionList
```

GetHouseholdSubscriptions

since 8.1

Gets all the subscriptions to which the account (household) is subscribed (services, packages and Service Plans). (Since 8.1 SR1)

Returns

Name	Description
HouseholdSubscriptions	The list of subscriptions to which the account is subscribed.

GetHouseholdTerminalCategoryList

since 7.1

An action used to return a list of device categories. A device category appears in the list if the account (household) has at least one registered device belonging to that category. The information returned includes the maximum number of devices (terminals) the account can have in this category. This value is calculated as the lower of the following values: -maximum number of allowed devices for an account defined system wide for all the accounts -maximum number of allowed devices for the specified account -maximum number of allowed devices for the category, system wide for all the accounts -maximum number of allowed devices for the category for the specified account In order to determine whether the account is allowed to register a new device for a specific category, the value of maximum devices relevant to the account type (maxTerminals/maxTerminalsOfNonOperator) should be used.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
TerminalCategory[]	

Examples

```
1 GetHouseholdTerminalCategoryList
```

See Also

- [TerminalCategory](#)

GetLanguageList*since 5.3*

An action used to get the language list supported by the scheme.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Language[]	

Examples

```
1 GetLanguageList
```

See Also

- [SetLanguage](#)

GetLiveLicenseAcquisitionInfo*since 7.4*

Returns the license acquisition information needed for getting a live and PPV event (if PPV event feature is supported) license. If cas_ids is not provided, the license information for all the channel packages/subscriptions for the logged-in user are returned. If cas_ids is provided but its value is empty or it only includes a partial list, then license information will be returned only for the cas_ids that were provided. If the account does not currently have an active subscription to the

specified Channel Package (or to a Channel Package that includes the specified channel), but has subscribed to a relevant Service Plan that has a future activation date, licensing information will nevertheless be returned. The returned information includes all relevant information that enables the client application to request and obtain a license for included Channel Packages in advance. (Only if service_delivery_type is specified.)

Parameters

Name	Type	Description
channel_id	Number	The channel ID.
channel_external_id	String	The channel external ID.
package_id	Number	The package ID.
package_external_id	String	The package external ID.
cas_ids	String	The CAS IDs of the packages/channels. Comma delimited values.
mac_address	String	Terminal identifier. The MAC address of the terminal that requested the license acquisition info.
serial_number	String	Terminal identifier. The serial number of the terminal that requested the license acquisition info.
smart_card_id	String	Terminal identifier. The smart card ID of the terminal that requested the license acquisition info.
service_delivery_type	String	The service delivery type. Possible values: OTT/IPTV.
is_renew	Boolean	A flag that indicates the license mode. For renew mode, set as true. For new mode, set as false. Default is false.
license_type	String	Requested license type. Possible values: VODRM/PLAY_READY. Default is VODRM.

Returns

Name	Description
LiveLicenseAcquisitionInfo	

Throws

Name	Description
INVALID_CHANNEL	Channel doesn't exist, or it was turned 'off'.
INVALID_CHANNEL_MODE	The license acquisition for channel mode is invalid.
INVALID_TERMINAL	Terminal is not registered for the logged in household.

Name	Description
SERVICE_UNSUBSCRIBED	The logged in household is not subscribed to the required service.
TERMINAL_IDENTIFIER_MISMATCH	One of the terminal parameters doesn't match the logged in terminal.
UNSUBSCRIBED_CAS_ID	No packages/channels subscriptions for the account were found based on the requested CAS_IDs.
USER_NOT_SUBSCRIBED_TO_CHANNEL	User is not register to the channel
USER_NOT_SUBSCRIBED_TO_PACKAGE	The logged in household is not subscribed to the specified package.

Comments

- One of terminal identifiers must be provided when the terminal identity has not provided before.

Examples

```
1 | GetLiveLicenseAcquisitionInfo?channel_id=777&client=xml
```

GetLivePlayingInfo**since 7.2**

Returns the channel information needed for joining a channel or playing time-shifted programs.
 Note: In order to play time-shifted programs, the account must be subscribed to a relevant service (i.e. Start Over, Catch-Up, or NPVR) and the time-shifted service type must be specified in time_shifting_service. However, the operator may choose to allow all accounts to use the Start Over and/or Catch-Up feature for free, without having to subscribe to a Start Over/Catch-Up service (requires configuration).

Parameters

Name	Type	Description
channel_id	Number	Unique channel ID. (Mandatory only for joining a channel if channel_external_id is not provided.)
program_id	Number	Unique program ID.(Relevant for PPV (PPV is deprecated since 8.4) when CAS token is calculated and for Catch-Up and Start Over (STOV) services.)
mac_address	String	Terminal identifier. The MAC address of the terminal that requested the playing info.
serial_number	String	Terminal identifier. The serial number of the terminal that requested the playing info.

Name	Type	Description
smart_card_id	String	Terminal identifier. The smart card ID of the terminal that requested the playing info.
include_cas_token	Boolean	Indicates whether the CAS token should be calculated and included in the result. (Default value is false)
time_shifting_service	String	The type of requested time-shifting service. Possible values: npvr, catchup, stov.
recording_id	Number	Deprecated since 9.0. The recording ID. Relevant for NPVR 1.0 only.
channel_external_id	String	The channel external ID. Mandatory only for joining a channel when channel_id is not provided. For Start Over and Catch-Up services channel_external_id can be used together with program_external_id.
program_external_id	String	The program external ID. Will be ignored if program_id is provided. Unique within a channel only. When supplying only program_external_id it is possible that more than one program is returned causing an exception. Use program_external_id together with channel_id or channel_external_id to get exact results.
otf_recording_id	String	The ID of the recording. Relevant for NPVR 2.0 only. Since 9.0.
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
LivePlayingInfo	

Throws

Name	Description
CHANNEL_NOT_VIEWABLE	The household is not subscribed to channel, and does not have any purchased PPV events.
FAILED_TO_GET_RECORDING_P LAYBACK_INFO	The specified recording failed.

Name	Description
INVALID_CHANNEL	The channel specified in the recording does not exist.
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
INVALID_PROGRAM_ID	Program not found for the supplied program ID. (Relevant for PPV when CAS token is calculated and for Catch-up and STOV services.)
INVALID_RECORDING_ID	No completed successful recording was found.
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account
INVALID_TERMINAL	Terminal is not registered for the logged-in household.
INVALID_TICKET	No valid ticket was found for PVR or PPV playing info.
MAX_SIMULTANEOUS_VIEWS_REACHED	The maximum number of simultaneous viewing sessions has been reached.
MISSING_ARGUMENT	A mandatory parameter was not specified.
NO_MATCHING_SOURCE_TYPE	No channel source type matches the device model.
SERVICE_NOT_AVAILABLE_ON_DEVICE	The device's category model is not enabled for NPVR or the On-the-fly technical format is not enabled for NPVR.
SERVICE_UNSUBSCRIBED	The logged in household is not subscribed to the required service.
SOURCE_TYPE_NOT_ALLOWED	Channel source type DVB-SI, MHP, IP is not allowed.
TARGETED_ADS_ENCRYPTION_FAILED	The encryption of the targeted advertising contexts has failed.
TERMINAL_IDENTIFIER_MISMATCH	One of the terminal parameters doesn't match the logged in terminal.
UNSUPPORTED_OPERATION	NPVR is not enabled in the system.
USER_NOT_SUBSCRIBED_TO_CHANNEL	The user is not subscribed to the channel.

Comments

- One of the terminal identifiers must be provided for time-shifting services and for retrieving channel information in case the CAS token is required. For more details see Appendix B - Purple DRM API Service.
- When the SimultaneousViewsRiGHTvControl is set to "On", RiGHTv fully controls the simultaneous views limit feature. In this configuration, RiGHTv decides whether to allow or refuse a new viewing session, based on the current session count and the maximum session count configured in the system.
- When SimultaneousViewsRiGHTvControl is set to "Off", RiGHTv does not manage the simultaneous views limit feature, it is managed by the CDN plugin or not managed at all.
- When SimultaneousViewsRiGHTvControl is set to "Customized", RiGHTv partially manages the simultaneous views limit feature, while it stores allowed sessions, the CDN plugin decides whether to allow or refuse a new viewing session request - only for sessions that were created successfully. Refused sessions are not stored. "Customized" mode allows project customization for this feature.
- If simViewingEnforceSingleSessionPerDevice is enabled, it assumes that only one playback is possible per device at a certain time. Therefore when a new playback is requested, existing sessions are automatically closed. This configuration is only relevant in case SimultaneousViewsRiGHTvControl is set to "On"/"Customized".

Examples

```
1 GetLivePlayingInfo?channel_id=777&include_cas_token=true&client=xml
```

GetMarkedPosition

since 7.3

An action to return the position of a movie.

Parameters

Name	Type	Description
ticket_id	Number	The ID of the ticket to return its position.
chapter_id	Number	The ID of the chapter to return its position.

Returns

Name	Description
Position	The marked position in milliseconds

Throws

Name	Description
INVALID CHAPTER	No chapter was found for the provided ticket ID.
INVALID TICKET	No ticket was found to return its position, or the ticket has expired.

Examples

```
1 | GetMarkedPosition?ticket_id=123
```

```
1 | GetMarkedPosition?ticket_id=123&chapter_id=456
```

See Also

- [MarkPosition](#)

GetMediaFile

since 5.3

An action used to return the media file associated with a specific movie item (by supplying asset_id and format) or a media file associated with a VOD program (by supplying a program_id).

Parameters

Name	Type	Description
asset_id	Number	The ID of the asset as defined in RiGHTv.
encoding	String	The encoding format of the asset.
program_id	Number	The ID of the VOD program matching the required media file. This argument should not be used in conjunction with the other arguments.

Returns

Name	Description
MediaFile	

Throws

Name	Description
INVALID_ASSET_ID	The supplied asset ID is invalid, or the media file for this asset does not exist.
INVALID_ENCODING	The supplied encoding is invalid.
INVALID_PROGRAM_ID	The supplied program ID is invalid, or the media file for this program does not exist.

Comments

- There are several permutations available using this API:
 - asset_id + encoding
 - program_id

Examples

```
1 | GetMediaFile?asset_id=123&encoding=MPEG-2
```

```
1 | GetMediaFile?program_id=123
```

GetMemberList

since 5.3

An action used to return the list of members associated with the logged-in household.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Member[]	

Examples

```
1 | GetMemberList
```

GetMessageList

since 5.3

An action used to return the message list associated with the logged-in household. Asynchronous processes in the server that require the users attention are added to the message list (e.g. a failure to record a program).

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
order	String	An indicator for sorting the list according to the message publish date. Allowed values: "asc"(ascending) or "desc"(descending). The default is "asc".

Returns

Name	Description
Message[]	

Examples

```
1 | GetMessageList
```

See Also

- [DeleteMessage](#)

GetMsg

since 7.4

External system API (Messaging Dispatcher). This API is used for getting messages that were sent to the terminal of the logged-in household.

Parameters

Name	Type	Description
from_terminal_id	String	The ID of the terminal from which to get message.
from_mac_address	String	The MAC address of the terminal from which to get message.
from_serial_number	String	The serial number of the terminal from which to get message.
from_smart_card_id	String	The smart card ID of the terminal from which to get message.
message_id	String	The message ID.

Returns

Name	Description
Msg	

Throws

Name	Description
ENTITY_NOT_FOUND	Message ID does not exist.
HOUSEHOLD_NOT_ACTIVE	Household is currently not active (deleted or suspended).
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
MISSING_PARAMETER	One of mandatory parameters is missing.
NO_HOUSEHOLD_TERMINALS	Household has no terminals.

Comments

- RiGHTv cookies contain household_id and terminal_id identifiers. If RiGHTv login was done only with household (and does not contain terminal information) then one of 4 'from' parameters can be used to specify the terminal. Terminal 'from' parameters are relevant when login was done only with household.

Examples

```
1 GetMsg?message_id=1
```

See Also

- [GetMsgToken](#)

GetMsgToken

since 7.4

External system API (Messaging Dispatcher). A device which is logged in to RiGHTv and wants to connect the ConnectionManager (CM), first needs to receive an access token by calling this API. RiGHTv will call the Dispatcher API GetAccessToken with the relevant arguments and will return the token to the device.

Parameters

Name	Type	Description
from_terminal_id	String	The ID of the terminal from which to register token.
from_mac_address	String	The MAC address of the terminal from which to register token.
from_serial_number	String	The serial number of the terminal from which to register token.
from_smart_card_id	String	The smart card ID of the terminal from which to register token.

Returns

Name	Description
AccessToken	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is currently not active (deleted or suspended).
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
MAX_TIME_TO_LIVE_EXCEEDED	Verify op-config for MessagingTokenTimeToLive parameter. Default is 1 day. Maximum is 1 year.
MISSING_PARAMETER	One of the mandatory parameters is missing.
NO_HOUSEHOLD_TERMINALS	Household has no terminals.

Comments

- RiGHTv cookies contain household_id and terminal_id identifiers. If RiGHTv login was done only with household (and does not contain terminal information) then one of 4 'from' parameters can be used to specify the terminal. Terminal 'from' parameters are relevant when login was done only with household.

Examples

```
1 GetMsgToken
```

```
1 GetMsgToken?from_terminal_id=1
```

See Also

- [ReuseMsgToken](#)

GetNPVRProgramSupportedModelList*since 7.3*

Deprecated since 9.0. This action used to return a list of terminal models on which NPVR is supported of a specific program. Not relevant for NPVR 2.0.

Parameters

Name	Type	Description
program_id	Number	The ID of the program.

Returns

Name	Description
TerminalModel[]	

Throws

Name	Description
INVALID_PROGRAM_ID	Program was not found.

Examples

```
1 GetNPVRProgramSupportedModelList?program_id=1456
```

GetNPVRQuotaStatus*since 9.0*

Gets the status of the account's NPVR quota and the quota used by completed, in progress and scheduled recordings. Relevant for NPVR 2.0 only. For NPVR 1.0 use GetNPVRUsedQuota.

Returns

Name	Description
NPVRQuotaStatus	NPVRQuotaStatus The status of NPVR quota, including the quota used by completed and scheduled recordings, and the total quota assigned.

Throws

Name	Description
UNSUPPORTED_OPERATION	NPVR is not set to NPVR 2.0.

GetNPVRUsedQuota*since 7.4*

Deprecated since 9.0. Gets the total of the NPVR quota already used by existing recordings and scheduled recordings for the specified account. Relevant for NPVR 1.0 only. For NPVR 2.0 use GetNPVRQuotaStatus.

Returns

Name	Description
NPVRUsedQuota	

Examples

1	GetNPVRUsedQuota
---	------------------

GetOrderList*since 5.3*

An action used to return the order history list.

Parameters

Name	Type	Description
duration	Number	The duration in days, for which to generate the order history. The default is 30 days.
order_by	String	An indicator on how to sort the list. Possible options are "date" or "name". The default is order by "date".
order	String	An indicator on whether to sort the list by ascending ("asc") or descending ("desc"). The default is "desc".
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
type	String	<p>The requested order type. Possible values:</p> <ul style="list-style-type: none"> • all • VOD • PPV • VOD PACKAGE • VOD CARD • VOD GIFT • Recharge • Deduction • SEASON PASS <p>The default type is "all". PPV is deprecated since 8.4.</p>
payment_mode	String	<p>The order payment mode. Possible values:</p> <ul style="list-style-type: none"> • all • normal • internal prepaid <p>The default payment mode is "all".</p>

Returns

Name	Description
Order[]	

Examples

```
1 GetOrderList?order_by=date&order=asc
```

GetPPVEvent*since 8.4*

An action used to return a single PPV event

Parameters

Name	Type	Description
external_id	String	The ppv event external ID. (Mandatory)

Returns

Name	Description
PPVEvent	

Throws

Name	Description
PPV_EVENT_EXPIRED	PPV Event was expired.
PPV_EVENT_NOT_FOUND	PPV Event was not found.
UNSUPPORTED_OPERATION	PPV Event feature is not supported.

Examples

```
1 GetPPVEvent?external_id=11
```

See Also

- [PPVEvent](#)

GetPPVEventList*since 8.4*

An action used to return a list of current or future PPV events. The list will be sorted by start time - nearest first

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
PPVEvent[]	List of PPVEvent[] (current or future).

Throws

Name	Description
UNSUPPORTED_OPERATION	PPV Event feature is not supported.

Examples

```
1 GetPPVEventList?
```

See Also

- [PPVEvent](#)

GetPPVEventTicketList*since 8.4*

Returns all of the logged-in account's PPV event tickets for PPV events currently playing and to be played in the future. The PPV event tickets are returned in the order in which the PPV events will be shown.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
PPVEventTicket[]	The PPVEventTicket list.

Throws

Name	Description
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households..
PERMISSION_DENIED	The user is not permitted to access the page.
UNSUPPORTED_OPERATION	PPV Event feature is not supported.

Examples

```
1 GetPPVEventTicketList?
```

See Also

- [OrderPPVEvent](#)
- [PPVEventTicket](#)

GetPackageChannelList

since 5.3

An action used to return the channel list for a specific package.

Parameters

Name	Type	Description
package_id	Number	The ID of the package.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
PackageChannel[]	

Throws

Name	Description
INVALID_PACKAGE_ID	The supplied package ID (package_id) is invalid.
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.

Comments

- GetPackageChannelList returns all channels including channels that cannot be played on the device, unlike GetChannelList which only returns channels that can be played on the device according to the model_external_id that is supplied.

Examples

```
1 GetPackageChannelList?package_id=123
```

See Also

- [Package](#)
- [PackageSubscription](#)
- [GetPackageList](#)
- [GetHouseholdPackageList](#)

GetPackageList

since 5.3

An action used to return the list of available packages. When channel_id or channel_external_id are provided, returns the list of channel packages to which the provided channel belongs. When bouquet_id is provided, returns the list of channel packages to which the provided bouquet belongs.

Parameters

Name	Type	Description
bouquet_id	Number	The bouquet ID number.
channel_id	Number	The channel ID number.
channel_external_id	String	The channel external ID.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
store_id	String	The ID of the app store.
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
Package[]	

Throws

Name	Description
INVALID_BOUCLET_ID	The bouquet ID is invalid.
INVALID_CHANNEL	The channel ID is invalid.
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Examples

```
1 GetPackageList
```

```
1 GetPackageList?bouquet_id=111
```

See Also

- [PackageChannel](#)
- [PackageSubscription](#)
- [GetPackageChannelList](#)
- [GetHouseholdPackageList](#)

GetPage*since 5.3*

An action used to return a page from RiGHTv.

Parameters

Name	Type	Description
page_id	Number	The ID of the page as defined in RiGHTv.
external_id	String	The external ID of the page.

Returns

Name	Description
Page	

Throws

Name	Description
INVALID_PAGE_ID	The supplied page ID is invalid.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Comments

- There are several permutations available for using this API:
 - page_id
 - external_id

Examples

```
1 GetPage?page_id=123
```

```
1 GetPage?external_id=abc
```

See Also

- [GetPageList](#)

GetPageList*since 5.3*

An action used to return an array of pages under a specified parent page.

Deprecated since RiGHTv 7.1. Please use GetUnifiedList instead.

Parameters

Name	Type	Description
parent_id	Number	The ID of the parent page as defined in RiGHTv.
external_parent_id	String	The external ID of the parent page.
order_by	String	An indicator on how to sort the list. Possible options are: "date" or "name". The default is order by "date" (the order in which the pages are sorted in VO Management Console). Sorting by name is case insensitive.
order	String	An indicator on whether to sort the list by ascending ("asc") or descending ("desc"). The default is "asc".
limited	Boolean	A Boolean that indicates whether to retrieve limited page ("true") or page ("false"). The default is false.
staging	Boolean	A Boolean that indicates whether to return staging state of the pages.
filter_empty_categories	Boolean	A Boolean that indicates whether to filter categories that do not have any videos as direct or indirect descendants.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Page[]	When limited parameter is false.
LimitedPage[]	When limited parameter is true.

Throws

Name	Description
INVALID_PAGE_ID	The supplied page ID is invalid.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Comments

- There are several permutations available for using this API:
 - parent_id
 - external_parent_id
- Pages which are defined as secured and for which the "hide in list" option is selected will not be included in the list for non permitted users.

Examples

```
1 | GetPageList?parent_id=123&order_by=name
```

```
1 | GetPageList?external_parent_id=abc&order_by=name
```

See Also

- [GetPage](#)

GetPageReferenceList

since 6.1

An action used to return an array of pageReferences for a specified page. Only announced or published pages are returned.

Parameters

Name	Type	Description
page_id	Number	The ID of the page as defined in RiGHTv.
root_ids	Number[]	A list of root folder IDs, separated by semicolons. The results are filtered by these root IDs. For empty root_ids, default value is 1 (main root).
external_page_id	String	The external ID of the page.
external_root_ids	String[]	A list of external root folder IDs, separated by semicolons. The results are filtered by these root IDs.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
PageReference[]	List of page references.

Comments

- There are several permutations available for using this API:
 - page_id
 - external_page_id

Examples

```
1 | GetPageReferenceList?page_id=123&root_ids=413;233;221
```

```
1 | GetPageReferenceList?external_page_id=abc&external_root_ids=aaa;bbb;ccc
```

See Also

- [GetPageList](#)

GetParentalRatingList*since 5.3*

An action used to return an array of parental ratings.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
ParentalRating[]	

Examples

```
1 | GetParentalRatingList
```

GetPaymentMethodDetailsInfo*since 7.3*

This action will retrieve the household payment info from the payment system (for 7.3 credit card information).

Returns

Name	Description
PaymentMethodDetailsInfo	

Throws

Name	Description
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.

Examples

1	GetPaymentMethodDetailsInfo
---	-----------------------------

GetPaymentRegistrationPropertyList*since 7.3*

This action retrieves payment registration info. The payment registration info is used by the client to register to the payment system.

Parameters

Name	Type	Description
payment_registration_property_names	String	String that holds all registration field names, separated by semicolons. For example: skinCode;orderData;recurringContract
payment_registration_property_values	String	String that holds all registration field values, separated by semicolons. For example: 6fC6PeJz;;RECURRING

Returns

Name	Description
PaymentRegistrationProperty[]	

Throws

Name	Description
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households.

Examples

1	GetPaymentRegistrationPropertyList? payment_registration_property_names=skinCode;orderData;recurringContract& payment_registration_property_values=6fC6PeJz;;RECURRING
---	--

GetPersoToken*since 7.3*

An action used to get the personalization token for the VODRM client.

Parameters

Name	Type	Description
mac_address	String	Terminal identifier. The MAC address of the terminal that requested the playing info.
serial_number	String	Terminal identifier. The serial number of the terminal that requested the playing info.

Name	Type	Description
smart_card_id	String	Terminal identifier. The smart card ID of the terminal that requested the playing info.

Returns

Name	Description
PersoTokenEntity	

Throws

Name	Description
INVALID_TERMINAL	Terminal is not registered for the logged-in household.
TERMINAL_IDENTIFIER_MISMATCH	Terminal parameter doesn't match the logged-in terminal.

Comments

- One and only one terminal identifier must be provided

Examples

```
1 | GetPersoToken?smart_card_id=AA:BB:CC:DD:EE:FF
```

GetPlayingInfo (deprecated)

since 5.4

- Instead use [GetLivePlayingInfo](#) An action used to return the playing information of a content item.

Parameters

Name	Type	Description
type	String	The type of the business model. Possible values: <ul style="list-style-type: none"> • "TSTV" • "STOV"
program_id	Number	The ID of the program.

Returns

Name	Description
PlayingInfo	

Throws

Name	Description
INVALID_PROGRAM_ID	The supplied program ID is invalid.
INVALID_TICKET	No ticket was found for PVR playing info.

Name	Description
SERVICE_UNSUBSCRIBED	The logged in household is not subscribed to the required service.

Examples

1 GetPlayingInfo?type=TSTV&program_id=123

1 GetPlayingInfo?type=STOV&program_id=123

See Also

- [GetVideoPlayingInfo](#)
- [GetProgramPlayingInfo](#)
- [GetAttachmentPlayingInfo](#)

GetPreOrderList*since 8.0*

An action used to return a list of all the pre-orders associated with the household.

Returns

Name	Description
PreOrder[]	Pre Orders of the household in status PENDING, FAILED, TO_BE_CANCELLED or CANCELLED.

See Also

- [PreOrder](#)
- [PreOrderVideo](#)

GetPricingMatrix*since 5.3*

An action used to return the pricing matrix for a given page in the VO Management Console.

Parameters

Name	Type	Description
video_id	Number	The ID of the video page as defined in VO Management Console to which the pricing matrix is assigned.
pricing_matrix_id	Number	The ID of the pricing matrix attached to the page.

Name	Type	Description
pre_order_pricing_supported	Boolean	Indicates whether to include the pre-order pricing option. When set to true and video_id is provided, the pre-order eST pricing option is returned only for movie pages in status Announced. When set to true and pricing_matrix_id is provided, all pricing options are returned including the pre-order eST pricing option. When set to false or not provided, the pre-order eST pricing option is not returned to the client.

Returns

Name	Description
PricingOption[]	

Throws

Name	Description
INVALID_VIDEO_ID	The supplied video page ID is invalid.

Comments

- Only one of the parameters must be supplied.

Examples

```
1 GetPricingMatrix?video_id=123
```

```
1 GetPricingMatrix?pricing_matrix_id=123
```

See Also

- [GetPricingMatrixList](#)

GetPricingMatrixList

since 6.1

An action used to return the list of available pricing matrixes.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Name	Type	Description
pre_order_pricing_supported	Boolean	Indicates whether to include the pre-order pricing option. When set to true all pricing options are returned including the pre-order eST pricing option. When set to false or not provided, the pre-order eST pricing option is not returned to the client.

Returns

Name	Description
PricingMatrix[]	

Examples

```
1 GetPricingMatrixList
```

See Also

- [GetPricingMatrix](#)

GetProgramList*since 5.3*

An action used to return an array of programs in a specified channel for a specified period.

Parameters

Name	Type	Description
channel_id	Number	The ID of the channel on which the programs reside.
channel_external_id	String	The external ID of the channel on which the programs reside.
start_date	Number	The lowest time limit for the program list, in milliseconds.
end_date	Number	The highest time limit for the program list, in milliseconds.
gaps	Boolean	A boolean that indicates whether to fill gaps in programs. The default behavior is false and it will return a consecutive program list, by filling gaps with virtual programs. Set this parameter to true to disable the default behavior.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Name	Type	Description
partial	Boolean	A boolean that indicates whether to return partial information for the program. If true retrieves the program list with an empty description to reduce memory cost. The default is false.

Returns

Name	Description
Program[]	

Comments

- There are several permutations available:
 - channel_id
 - channel_external_id

Examples

1	<code>GetProgramList?channel_id=123&start_date=1262323232231&end_date=1262325232231</code>
1	<code>GetProgramList?channel_external_id=aaa&start_date=1262323232231&end_date=1262325232231</code>

GetProgramPlayingInfo (deprecated)**since 5.3**

- Instead use [GetLivePlayingInfo](#) An action used to return the playing information of a past program. Used for Catch-Up TV. Relies on the user's being subscribed to Catch-Up (TSTV) service.

Parameters

Name	Type	Description
program_id	Number	The ID of the program.

Returns

Name	Description
RecordingPlayingInfo	

Throws

Name	Description
INVALID_PROGRAM_ID	The supplied program ID is invalid.
SERVICE_UNSUBSCRIBED	The logged in household is not subscribed to the required service.

Examples

1	<code>GetProgramPlayingInfo?program_id=123</code>
---	---

See Also

- [GetVideoPlayingInfo](#)
- [GetProgramPlayingInfo](#)
- [GetAttachmentPlayingInfo](#)

GetProgramTicketList

since 5.3

An action used to return a list of program tickets (paid programs) for the logged-in household.

Parameters

Name	Type	Description
start_date	Number	The lowest time limit for the ticket list, in milliseconds.
end_date	Number	The highest time limit for the ticket list, in milliseconds.
order_by	String	An indicator on how to sort the list. Two sorting options "date" or "name". The default is order by "date".
order	String	An indicator on sort the list ascending("asc") or descending ("desc"). The default is "asc".
model_external_id	String	The model external ID.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
ProgramTicket[]	

Throws

Name	Description
INVALID_TERMINAL_MODEL	The model ID is invalid.

Examples

```
1 GetProgramTicketList
```

See Also

- [ProgramTicket](#)
- [OrderProgram](#)

GetRechargeOptionList

since 6.0

An action used to return a list of recharge options.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
RechargeOption[]	

Examples

```
1 GetRechargeOptionList
```

GetRecordingMarkedPosition*since 8.2*

Returns the last saved position of an NPVR recording or program consumed through the Catch-Up or Start Over service.

Parameters

Name	Type	Description
recording_id	Number	The recording ID. Relevant for NPVR 1.0 only. Deprecated since 9.0.
otf_recording_id	String	The ID of the recording. Relevant for NPVR 2.0 only. Since 9.0.
program_id	Number	The internal ID of the program (relevant for Catch-up and Start Over).
program_external_id	String	The program external ID. Will be ignored if program_id is provided. The program_external_id is unique within a channel only. For Start Over and Catch-Up services when supplying only the program_external_id, it is possible that the same ID is used for multiple programs in different channels, resulting in an exception. Use program_external_id together with channel_external_id to ensure the marked position is saved for the desired program.

Name	Type	Description
channel_external_id	String	The channel external ID of the channel on which the program was broadcast. For Start Over and Catch-Up services when supplying only the program_external_id, it is possible that the same ID is used for multiple programs in different channels, resulting in an exception. Use channel_external_id together with program_external_id to ensure the marked position is saved for the desired program.

Returns

Name	Description
Position	The marked position in the recording/program in milliseconds.

Throws

Name	Description
INVALID_PROGRAM_ID	The supplied program ID does not exist.
INVALID_RECORDING_ID	The specified recording ID does not exist.
UNSUPPORTED_OPERATION	NPVR is not set to NPVR 2.0.

Examples

```
1 GetRecordingMarkedPosition?program_id=123
```

See Also

- [GetRecordingMarkedPosition](#)

GetRecordingTicket*since 9.1*

Gets the specified single recording ticket for the logged-in account. Cannot be used to return all recorded episodes in a season. Relevant for NPVR 2.0 only.

Parameters

Name	Type	Description
otf_recording_id	String	The recording ID. (Mandatory)

Returns

Name	Description
RecordingTicket	The recording ticket.

Throws

Name	Description
INVALID_RECORDING_ID	The recording specified does not exist.
MISSING_PARAMETER	The parameter otf_recording_id was not supplied.
SERVICE_UNSUBSCRIBED	The logged-in account is not subscribed to the NPVR service.
UNSUPPORTED_OPERATION	NPVR is not set to NPVR 2.0.

Examples

```
1 GetRecordingTicket?client=xml&otf_recording_id=123
```

See Also

- [OrderRecording](#)

GetRecordingTicketList

since 7.4

An action used to return a list of recording tickets for the logged-in account. For NPVR 1.0, if Personal Recording is enabled, returns a list of recording tickets for the logged-in profile. It is recommended to use pagination with this API for improved performance. When filtering recordings by start_date and end_date, for EPG based recordings if only the lead/lag duration falls in the requested time range specified, but the recorded program is outside the range, the recording will not be returned. One of the following permutations may be used for filtering by a specific season recording:

- series_id, season_name, channel_external_id
- series_ticket_id

If an episode was recorded as a stand-alone recording and not as part of a season recording, the recording will not be returned.

Parameters

Name	Type	Description
start_date	Number	The lowest time limit for the recording list, in milliseconds.
end_date	Number	The highest time limit for the recording list, in milliseconds.

Name	Type	Description
order_by	String	An indicator on how to sort the list. Possible options are: "date" or "name". The "date" refers to the recording start date. For EPG based recordings, when sorting by date, the lead time and lag time are ignored. For EPG based recordings, when sorting by name, sorting is done by the name of the program in the EPG when the recording was created. Sorting ignores changes to the program name in the EPG. For time-based recordings sorting is done by the recording name. The default is order by "date".
order	String	An indicator whether to sort the list by ascending("asc") or descending ("desc"). The default is "asc".
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
series_ticket_id	Number	(The G_ID of the T_SERIES_ORDER) The ID of the ticket of the season recording order. When supplied, only recordings that belong to the supplied season recording are returned. series_ticket_id is returned by GetSeriesTicketList - SeriesTicket.id.
recording_ids	String	Deprecated since 9.0. A list of live program recording IDs, separated by a configurable separator (default = ";"). If supplied, the resulting list of the recording tickets will be filtered by the given recording IDs.
series_id	String	(The G_ID of aT_SERIES)The ID of the series, used for filtering by a specific season recording. Must be supplied together with channel_external_id, and season_name.
season_name	String	The name of the series season, used for filtering by a specific season recording. Must be supplied together with channel_external_id, and series_id.

Name	Type	Description
channel_external_id	String	The ID of the channel, used for filtering by a specific season recording. Must be supplied together with series_id, and season_name.
personal	Boolean	For NPVR 1.0, if isPersonalRecordingSupported is set to true in the Configuration, then this parameter is set to true by default and only recordings for the profile are returned. * @return {RecordingTicket[]}

Throws

Name	Description
PROFILE_DATA_UNAVAILABLE	Personal data cannot be displayed as the COMPASS profile is unavailable.
UNSUPPORTED_OPERATION	If personal is set to true and isPersonalRecordingSupported is set to false in the Configuration.

Comments

- The delimiter is configured and can be changed in the op_config.xml. See the Configuration Guide for more details.

Examples

```
1 | GetRecordingTicketList?client=xml
```

```
1 | GetRecordingTicketList?recording_ids=1;2;&client=xml
```

See Also

- [OrderRecording](#)

GetReferenceList

since 5.3

An action used to return a reference list.

Parameters

Name	Type	Description
type	String	The type of reference list wanted. Possible values: <ul style="list-style-type: none"> "genres" "categories" "roles" "countries"
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Reference[]	Returns a reference list.

Examples

```
1 GetReferenceList?type=genres
```

See Also

- [GetGenreList](#)

GetSeriesTicketList

since 7.4

An action used to return the recorded series list.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
personal	Boolean	For NPVR 1.0, if isPersonalRecordingSupported is set to true in the Configuration, then this parameter is set to true by default and only series season recordings for the profile are returned.

Returns

Name	Description
SeriesTicket[]	

Throws

Name	Description
PROFILE_DATA_UNAVAILABLE	Personal data cannot be displayed as the COMPASS profile is unavailable.
UNSUPPORTED_OPERATION	if personal is set to true and isPersonalRecordingSupported is set to false in the Configuration.

Examples

```
1 | GetSeriesTicketList
```

See Also

- [OrderRecording](#)

GetServiceList*since 5.3*

An action used to return the available services list.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
store_id	String	The ID of the app store.

Returns

Name	Description
Service[]	

Throws

Name	Description
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Examples

```
1 | GetServiceList
```

See Also

- [ServiceSubscription](#)
- [GetHouseholdServiceList](#)

GetServicePlan*since 8.1*

Gets the specified Service Plan.(Since 8.1 SR1).

Parameters

Name	Type	Description
external_id	String	The Service Plan external ID.

Returns

Name	Description
ServicePlan	The Service Plan entity.

Throws

Name	Description
SERVICE_PLAN_NOT_EXIST	The specified Service Plan does not exist.

Examples

```
1 |.GetServicePlan?external_id=abc
```

GetServicePlanList*since 8.3 SR1*

Gets all the published Service Plans.

Parameters

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
store_id	String	The ID of the app store.

Returns

Name	Description
ServicePlan[]	List of Service Plans.

Throws

Name	Description
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Examples

```
1 |GetServicePlanList
```

GetSessionList (deprecated)*since 7.1***since 7.4**

An action used to get all household remote terminal sessions. If any of the terminal identifiers is passed, the returned list will include only the sessions that are open for that terminal.

Parameters

Name	Type	Description
terminal_id	String	The ID of the terminal for which to return the list.
mac_address	String	The MAC address of the terminal for which to return the list.
serial_number	String	The serial number of the terminal for which to return the list.
smart_card_id	String	The smart card ID of the terminal for which to return the list.

Returns

Name	Description
ProxySession[]	

Comments

- For more details on how to use the remote messaging mechanism please read "RiGHTv Remote Client PVR Developers Guide".

Examples

```
1 | GetSessionList
```

See Also

- [OpenSession](#)
- [CloseSession](#)
- [SendMessage](#)
- [PollMessage](#)
- [SendResponse](#)

GetSpecificVideoList**since 7.0**

Gets a list of videos according to the specified external video IDs. Gets the tvShowReference parameter if the asset is an episode in a TV show series.

Parameters

Name	Type	Description
video_external_ids	String	A list of videos external IDs, separated by configurable separator (default = ";") No limitation. The external IDs cannot contain semicolons, as it is used as the delimiter.
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
Video[]	

Comments

- The delimiter is configured and can be changed in the op_config.xml. See configuration guide for more details.

Examples

```
1 | GetSpecificVideoList?video_external_ids=video1;video2;video3
```

```
1 | GetSpecificVideoList?video_external_ids=video1
```

See Also

- [GetVideo](#)
- [GetVideoList](#)

GetSubscriptionOffers*since 8.1*

Gets all offers available for subscription (services, packages and Service Plans).(Since 8.1 SR1)

Parameters

Name	Type	Description
bouquet_external_id	String	The bouquet external ID. If this parameter is supplied, returns services, channel packages that belongs to the specified bouquet, Service Plans that contain the specified bouquet or Service Plans that do not contain any bouquets at all (i.e that only contain SVOD services and channel packages).
channel_id	Number	The channel ID. If specified, only Channel Packages and Service Plans that contain the specified channel will be returned. (Precedence 1 - see comments). Since 8.3 SR2.
channel_external_id	String	The channel external ID. If specified, only Channel Packages and Service Plans that contain the specified channel will be returned. (Precedence 2 - see comments). Since 8.3 SR2.
store_id	String	The ID of the app store.

Returns

Name	Description
OffersInfo	

Throws

Name	Description
INVALID_BOUQUET_ID	The specified bouquet does not exist.
INVALID_CHANNEL	The specified channel does not exist.
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Comments

- Precedence: Priority order if multiple channel identifiers specified: channel_id, channel_external_id.

Examples

```
1 | GetSubscriptionOffers?bouquet_external_id=abc
```

GetTVShowTicketList**since 8.2**

Returns a list of TV show tickets for the season/s ordered by the logged-in account (household). For rental, returns active tickets and for eST, returns active and expired tickets. An active ticket indicates that it is possible to watch the episodes online/offline or to download them (based on the purchase conditions). An eST expired ticket indicates that it is no longer possible to watch the episodes online or to download them, but it is still possible to watch previously downloaded episodes offline. To request tickets for a specific season or season page supply one of the identifiers: season_external_id or page_external_id. When season_external_id is supplied, page_external_id is ignored.

Parameters

Name	Type	Description
page_external_id	String	The TV show season page external ID through which the season was ordered.
season_external_id	String	The TV Show season external ID. If not supplied, valid tickets for all the account's TV Show seasons are returned.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
TvShowTicket[]	The list of TV show tickets.

Throws

Name	Description
INVALID_PAGE_ID	The specified season page ID does not exist.
INVALID_PAGE_TYPE	The specified page is not a season page.
PERMISSION_DENIED	The user is not permitted to access the page.
TV_SHOW_NOT_EXISTS	The specified TV show does not exist.

Examples

```
1 GetTVShowTicketList?external_id=page123
```

See Also

- [OrderTVShow](#)
- [TvShowTicket](#)

GetTerminalList*since 6.2*

An action used to return a list of the household's terminals.

Returns

Name	Description
HouseholdTerminals	

Examples

```
1 GetTerminalList
```

See Also

- [Terminal](#)

GetTerminalModelList*since 7.1*

An action used to return the list of terminal (device) models in RiGHTv. It can return a filtered list by category and/or by alias.

Parameters

Name	Type	Description
category_external_id	String	The external ID of the terminal category.
alias	String	Used as search phrase to return the models that fits the specified alias.

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
TerminalModel[]	

Throws

Name	Description
INVALID_CATEGORY	The passed category external id is invalid (not exist)

Examples

```
1 | GetTerminalModelList
```

See Also

- [TerminalModel](#)

GetTimeShiftingServicesConfig*since 7.3*

An action used to return time-shifting services configuration values from the server.

Returns

Name	Description
TimeShiftingServicesConfig	

Examples

```
1 | GetTimeShiftingServicesConfig
```

GetTimezoneOffsetList*since 6.0*

An action used to return the time zone offset list for the logged-in household. Should be used to support DST.

Parameters

Name	Type	Description
timezone	String	The name of the timezone for which the offsets are requested.
from	Number	The first index requested. Used for paging. (Number type: Integer).

Name	Type	Description
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
TimezoneOffset	

Throws

Name	Description
INVALID_TIMEZONE	The supplied timezone is invalid.

Examples

```
1 GetTimezoneOffsetList
```

GetTvShowEpisodePlayingInfo**since 8.2**

Returns the playing information of an episode that is part of a season that was ordered, also for an episode that has the status Removed from Service. Creates a ticket for the episode if no ticket exists. If the episode was on the wish list, it is removed.

Parameters

Name	Type	Description
video_external_id	String	The episode's movie page external ID.
tv_show_ticket_id	Number	The TV show ticket ID.
viewing_method	String	The viewing method for the episode. [Values: streaming, download]
terminal_id	Number	The internal ID of the account's device. This parameter should be used when integrating with the video server in order to retrieve the video viewing sessions per device.
mac_address	String	The MAC address of the device that requested the playing information.
serial_number	String	The serial number of the device that requested the playing information.
smart_card_id	String	The smart card ID of the device that requested the playing information.

Name	Type	Description
is_est_download_temporary	Boolean	Set to true to return video download info for a temporary eST download. A temporary eST download video can be played during a limited period of time (specified by estTemporaryDownloadPeriod in est-config.xml), but not beyond the end date of the ticket. The number of times a temporary eST download video can be downloaded and played, and the number of devices on which the video can be downloaded and played, is not limited by RiGHTv. For each temporary eST download, a new license is granted (the license period is specified by estTemporaryDownloadPeriod in est-config.xml).
definition	String	The content definition of the media file for which to return the TV show episode playing info, that must be equal to or lower than the definition specified for the TV show ticket. If no matching media file of the specified definition exists: if Multi Definition Playback is not enabled in the Configuration, this parameter is ignored and the highest priority media file will be returned, if Multi Definition Playback is enabled, an error will be returned. If content definition is not specified, the highest priority media file will be returned. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.
encoding	String	The encoding format of the media file to be played (e.g. "MPEG-2").
resume	Boolean	When set to true, calculates the resume position in the video and returns it as part of the video playing information. When set to false, the returned position is 0. The resume position can be calculated based on a specific chapter or bookmark (if the relevant ID is provided) or based on the last marked position. The default is false.
bookmark_id	Number	The ID of the bookmark for which the playing information is requested. resume=true parameter must be passed as well with this parameter.

Name	Type	Description
chapter_id	Number	The ID of the chapter for which the playing information is requested. resume=true parameter must be passed as well with this parameter.

Returns

Name	Description
VideoUrlInfo	The episode playing or download information.

Throws

Name	Description
DEFINITION_HIGHER_THAN_TICKET_DEFINITION	The definition specified is higher than the definition for the TV show ticket.
DELIVERY_MODE_NOT_EST_DOWNLOAD	The ticket delivery mode is not eST and is_est_download_temporary is set to true, or is_est_download_temporary is set to true, but the ticket delivery mode is not eST and/or the viewing method requested is not Download.
EPISODE_ALREADY_CONSUMED	This episode was already watched or downloaded by the logged-in account.
EST_LIMIT_REACHED	eST downloads have been performed on the maximum number of devices allowed. eST downloads cannot be performed on additional devices.
INVALID_ENCODING	The supplied encoding is invalid.
INVALID_TERMINAL	Device is not valid / not registered for the logged-in account.
INVALID_TV_SHOW_SEASON	The episode does not belong to the season specified in the TV Show ticket.
INVALID_VIDEO_ID	The specified movie page does not exist.
NO_MATCHING_MEDIA_FILE_FOR_DEFINITION	There is no media file that matches the definition specified.
NO_SUITABLE_MEDIA_FILE	There is no media file with a matching definition and delivery mode for the specified device.
PERMISSION_DENIED	The logged-in account does not have permission to access the required content due to security restrictions.
TV_SHOW_TICKET_EXPIRED	The TV show ticket specified has expired.
TV_SHOW_TICKET_NOT_FOUND	The logged-in account does not have a TV show ticket for the specified episode movie page.

Name	Description
VIDEO_ID_MISMATCH	The episode playing info for the specified episode has previously been retrieved for this TV Show Ticket for a different movie page ID.
VIEWING_METHOD_MISMATCH	The viewing method does not match the delivery mode.

Comments

- At least one terminal identifier must be specified (mac_address, serial_number, smart_card_id, terminal_id).

Examples

```
1 GetTvShowEpisodePlayingInfo?video_external_id=MySeriesSE3EP1&tv_show_ticket_id=3&
  viewing_method=streaming&mac_address=22:33:44:55
```

See Also

- [OrderTVShow](#)

GetUnifiedIndexList

since 7.1

An action used to retrieve an alphanumeric indexed list, according to the results from GetUnifiedList.

Parameters

Name	Type	Description
category_id	Number	The ID of the parent page as defined in RiGHTv.
external_category_id	String	The external ID of the parent page.
encoding	String	Filtering according to encodings (does not apply for categories - all categories will be returned). A comma separated list of encodings. No limitation. The encodings cannot contain commas, as commas are used as the delimiter.
filter_empty_categories	Boolean	A boolean that indicates whether to filter categories that do not have any videos as direct or indirect descendants(true), or not(false). The default is false.
order	String	An indicator whether to sort the list by ascending("asc") or descending ("desc"). The default is "asc".

Name	Type	Description
aggregate	Boolean	A boolean that indicates whether the video is aggregated. Possible values: true or false (the default is false). The aggregated video includes an aggregated result of one or more pages/shortcuts(services attribute) that have the same asset and exists in the same video category (same path); and the movies are returned consecutively (one after the other) according to the sorting option. It is essential that the sorting is done by name to ensure that the videos will be consecutive.
statuses	String	Filter according to statuses (does not apply for categories, all categories will be returned). A semicolon separated list of the following valid values: <ul style="list-style-type: none"> • "published" • "announced"
definitions	String	Filter according to definitions. A semicolon separated list of definitions. No limitation. The definitions cannot contain semicolons, as they are used as the delimiter.
delivery_mode	String	Filter according to delivery mode (does not apply for categories, all categories will be returned). Single allowed value: "streaming", "download" or "converged" (Since RiGHTv 7.3). This parameter is ignored if the availability filter contains the value: "for_download".
svod_min_duration	Number	Filter the list in order to return only SVOD movies whose publish time is more than the minimum duration given. min_duration is given in hours (according to the removal date).
services	String	Filter according to services/groups. A semicolon separated list of security groups (services).
model_external_id	String	Filter the list in order to return only content that can be viewed on specified terminal model.
external_channel_id	String	Filter according to the channel from which the program was recorded.

Name	Type	Description
max_pr_level	Number	The maximum level of parental rating allowed. When provided, returns movie pages with no parental rating or with a parental rating equal to or less than this level.
availability	String	Filter according to availability. This filter can be left empty for no filtering, or contain the value: "for_download", to limit returns to videos and pages that are available for download.

Returns

Name	Description
CharIndex[]	List of CharIndexes. The action will return a list that represents the indexes of the unified list.

Throws

Name	Description
INVALID_PAGE_ID	Supplied page ID is invalid.
NO_PROFILES_FOR_TERMINAL_MODEL	The provided terminal model has no profiles.
UNSUPPORTED_DELIVERY_MODE_BY_TERMINAL_MODEL	The provided delivery mode is not supported by provided terminal model, or the provided terminal model doesn't support availability for download.

Comments

- There are several permutations available for using this API:
 - `category_id`
 - `external_category_id`

Examples

```
1 GetUnifiedIndexList?category_id=123&encoding=MPEG-2,MPEG-4
```

```
1 GetUnifiedIndexList?external_category_id=aaa&encoding=MPEG-2
```

See Also

- [GetVideo](#)
- [Page](#)
- [Video](#)
- [VideoPackage](#)
- [AggregatedVideo](#)
- [GetUnifiedList](#)

GetUnifiedList

since 6.0

Gets a list of videos and pages from the VOD catalog, optionally filtered according to the requesting device. When filtering according to service, the filtered content is returned together with any content that is allowed to all users. Gets the tvShowReference parameter if the asset is an episode in a TV show series.

Parameters

Name	Type	Description
category_id	Number	The ID of the parent page as defined in RiGHTv. Will be ignored if external ID of the parent page is supplied.
external_category_id	String	The external ID of the parent page.
encoding	String	Filter according to encodings (does not apply for categories or TV Shows, all categories will be returned). A comma separated list of encodings. No limitation. The encodings cannot contain commas, as they are used as the delimiter.
filter_empty_categories	Boolean	A boolean that indicates whether to filter categories that do not have any videos as direct or indirect descendants (true), or not (false). The default is false.
staging	Boolean	A boolean that indicates whether to retrieve video information in prior to publication stage (true), or not (false). The default is false.
order_by	String	An indicator on how to sort the list. Possible options are: "tree", "name" or "broadcast_time". The default is order by "tree". Sorting by name is case insensitive.
order	String	An indicator on whether to sort the list by ascending ("asc") or descending ("desc"). The default is "asc".
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Name	Type	Description
aggregate	Boolean	A boolean that indicates whether the video is aggregated. Possible values: true or false (the default is false). The aggregated video includes an aggregated result of one or more pages/shortcuts(services attribute) that have the same asset and exists in the same video category (same path); and the movies are returned consecutively (one after the other) according to the sorting option. It is essential that the sorting will be by name to ensure that the videos will be consecutive.
statuses	String	Filter according to statuses. A semicolon separated list of the following valid values: <ul style="list-style-type: none"> • "published" • "announced"
definitions	String	Filter according to definitions (does not apply for categories, all categories will be returned). A semicolon separated list of definitions. No limitation. The definitions cannot contain semicolons, as they are used as the delimiter.
delivery_mode	String	Filter according to delivery mode (does not apply for categories, all categories will be returned). Single allowed value: "streaming", "download" or "converged"(Since RiGHTv 7.3). This parameter is ignored if the availability filter contains the value: "for_download".
svod_min_duration	Number	Filter the list in order to return only SVOD movies whose publish time is more than the minimum duration given. min_duration is given in hours (according to the removal date).
services	String	Filter according to services/groups. A semicolon separated list of security groups (services).
model_external_id	String	Filter the list in order to return only content that can be viewed on the specified terminal model.
external_channel_id	String	Filter the list according to the channel from which the program was recorded. Equivalent to the channel reference ID that is defined through in RiGHTv.

Name	Type	Description
limited	Boolean	A boolean that indicates whether to retrieve LimitedVideo(true) or Video(false). The default is false.
max_pr_level	Number	The maximum level of parental rating allowed. When provided, returns movie and TV show series/season pages with no parental rating or with a parental rating equal to or less than this level.
availability	String	Filter according to availability. This filter can be left empty for no filtering, or contain the value: "for_download", to limit returns to videos and pages that are available for download.
profile_type	String	The type of profile associated with the logged-in profile (user). Must match a profile type defined in the configuration. Only results that conform with the rules specified for the profile type in the configuration (e.g. maximum parental rating) will be returned.

Returns

Name	Description
Page[]	
Video[]	
LimitedVideo[]	LimitedVideo is returned when the limited parameter is true.
VideoPackage[]	
AggregatedVideo[]	
TvShowSeriesPage[]	
TvShowSeasonPage[]	

Throws

Name	Description
INVALID_PAGE_ID	Supplied page ID is invalid.
NO_PROFILES_FOR_TERMINAL_MODEL	The provided terminal model has no profiles.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions (can only occur when the VALIDATE_SECURITY_ON_PAGE_RETRIEVAL from supported functionality is true).

Name	Description
UNSUPPORTED_DELIVERY_MODE_BY_TERMINAL_MODEL	The provided delivery mode is not supported by the provided terminal model, or the provided terminal model doesn't support availability for download.

Comments

- There are several permutations available for using this API:
 - category_id
 - external_category_id

Examples

```
1 GetUnifiedList?category_id=123&encoding=MPEG-2,MPEG-4
1 GetUnifiedList?external_category_id=aaa&encoding=MPEG-2
1 GetUnifiedList?category_id=123&model_external_id=PC
1 GetUnifiedList?external_category_id=123&model_external_id=PC
1 GetUnifiedList?external_category_id=123&model_external_id=PC&definitions=HD;SD
1 GetUnifiedList?external_category_id=123&model_external_id=PC&definitions=HD;SD&
  statuses=published
1 GetUnifiedList?external_category_id=123&model_external_id=PC&definitions=HD;SD&
  statuses=published&aggregate=true
```

See Also

- [GetVideo](#)
- [Page](#)
- [Video](#)
- [VideoPackage](#)
- [AggregatedVideo](#)

GetVideo

since 5.3

Gets a video page from RiGHTv. Gets the tvShowReference parameter if the asset is an episode in a series.

Parameters

Name	Type	Description
video_id	Number	The ID of the video as defined in RiGHTv.
external_id	String	The external ID of the page.

Returns

Name	Description
Video	

Throws

Name	Description
INVALID_VIDEO_ID	The supplied page ID is invalid.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Comments

- There are several permutations available for using this API:
 - video_id
 - external_id

Examples

```
1 | GetVideo?video_id=123
```

```
1 | GetVideo?external_id=abc
```

See Also

- [GetVideoList](#)

GetVideoAsset*since 6.0*

Gets the video asset entity. Gets the tvShowReference parameter if the asset is an episode in a TV show series.

Parameters

Name	Type	Description
video_asset_id	Number	The video asset ID.
external_id	String	The video asset external ID.
provider_asset_id	Number	The video asset external ID (of the content provider).
content_provider_external_id	String	The content provider external ID.

Returns

Name	Description
VideoAsset	

Throws

Name	Description
INVALID_VIDEO_ASSET_ID	The supplied video asset ID is invalid.

Comments

- There are several permutations available for using this API:
 - video_asset_id
 - provider_asset_id
 - provider_asset_id + content_provider_external_id

Examples

```
1 GetVideoAsset?video_asset_id=123
```

```
1 GetVideoAsset?provider_asset_id=123
```

```
1 GetVideoAsset?provider_asset_id=123&content_provider_external_id=aaa
```

GetVideoCardList

since 5.3

An action used to return a list of active video cards.

Parameters

Name	Type	Description
page_id	Number	The ID of the page for which the list of cards applies. Only valid (and active) cards for the supplied page are included in the list.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
business_model_supported	String	Supported business model of VOD cards to return. Possible values: rental, est, all. Default value: all.

Returns

Name	Description
VideoCard[]	

Throws

Name	Description
INVALID_PAGE_ID	The supplied page ID is invalid.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Examples

```
1 GetVideoCardList?page_id=123
```

See Also

- [OrderVideoCard](#)
- [GetVideoCardTicketList](#)

GetVideoCardTicketList

since 5.3

An action used to return a list of video tickets rented by the logged-in household.

Parameters

Name	Type	Description
page_id	Number	The ID of the page for which the list of card tickets apply. Only valid tickets for the supplied page are included in the list. If not provided all user's VOD cards return
complete	Boolean	A boolean that indicates whether to include all tickets for VOD cards where all the movies were consumed. Used mainly for display purposes, not for ordering. The default is false.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
VideoCardTicket[]	

Throws

Name	Description
INVALID_PAGE_ID	The supplied page ID is invalid.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Comments

- There may be more than one ticket for a specific video card, since the subscriber can order the same card more than once.

Examples

```
1 | GetVideoCardTicketList?page_id=123
```

See Also

- [OrderVideoCard](#)
- [GetVideoCardList](#)

GetVideoDownloadInfo

since 6.1

Returns the download information for an asset.

Parameters

Name	Type	Description
video_id	Number	The ID of the movie page as defined in RiGHTv.
video_external_id	String	The external ID of the movie page as defined in RiGHTv.
encoding	String	The encoding format of the attachment (e.g. "MPEG-2"). Considered as a recommendation. If more than one media file is found according to the model ID, the media files are filtered according to the encoding.
terminal_identifier	String	The device (terminal) identifier used when single terminal download validation is enforced.
definition	String	The content definition of the media file for which to return the video download info, that must be equal to or lower than the definition specified on the page. If no ticket exists with a matching media file of the specified definition: if Multi Definition Playback is not enabled in the Configuration, the content definition is ignored and the highest priority media file will be returned, if Multi Definition Playback is enabled, an error will be returned. If content definition is not specified, the highest priority media file will be returned. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.
model_external_id	String	The external ID of the device's (terminal's) model.
mac_address	String	The MAC address of the terminal that requested the playing info. Shall be used to keep the terminal id in the token.
serial_number	String	The serial number of the terminal that requested the playing info. Used to keep the terminal ID in the token.

Name	Type	Description
smart_card_id	String	The smart card ID of the terminal that requested the playing info. Used to keep the terminal ID in the token.
is_est_download_temporary	Boolean	Set to true to return video download info for a temporary eST download. A temporary eST download video can be played during a limited period of time (specified by estTemporaryDownloadPeriod in est-config.xml), but not beyond the end date of the ticket. The number of times a temporary eST download video can be downloaded and played, and the number of devices on which the video can be downloaded and played, is not limited by RiGHTv. For each temporary eST download, a new license is granted (the license period is specified by estTemporaryDownloadPeriod in est-config.xml).

Returns

Name	Description
VideoDownloadInfo	

Throws

Name	Description
ACCOUNT_HAS_REACHED_THE_MAXIMUM_ALLOWED_DOWNLOADS_FOR_THIS_ASSET	The account has reached the maximum number of downloads allowed for this asset.
ACCOUNT_HAS_REACHED_THE_MAXIMUM_LIMIT_OF_DOWNLOADS_FOR_THIS_MONTH	The account has reached the maximum number of downloads allowed for this month.
DELIVERY_MODE_NOT_EST_DOWNLOAD	The ticket delivery mode is not eST and is_est_download_temporary is set to true.
EST_LIMIT_REACHED	eST downloads have been performed on the maximum number of devices allowed. eST downloads cannot be performed on additional devices.
INVALID_DEFINITION	The supplied definition is invalid.
INVALID_ENCODING	The supplied encoding is invalid.
INVALID_TERMINAL	Either the terminal is not valid / registered for the logged-in household, or that the passed terminal identifier does not match the logged-in terminal.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.

Name	Description
INVALID_TICKET	The logged-in household has no ticket for this video, or the ticket has expired.
INVALID_VIDEO_ID	The supplied video page ID is invalid.
MAX_SIMULTANEOUS_VIEWS_REACHED	The maximum number of simultaneous viewing sessions has been reached.
NO_MATCHING_MEDIA_FILE_FOR_DEFINITION	There is no media file that matches the definition specified.
NO_SUITABLE_MEDIA_FILE	No suitable media file for the given properties.
TERMINAL_IDENTIFIER_MISMATCH	The terminal identifier is empty or does not match the ordered ticket terminal identifier.

Comments

- There are several permutations available for using this API:
 - video_id
 - video_external_id
- There are several permutations available for using this API. At least one of the following must be provided:
 - model_external_id - recommended
 - encoding - if this parameter is specified, the Multi Definition Playback will not work correctly
- When an eST pricing option is passed, at least one of the following must be provided:
 - mac_address
 - serial_number
 - smart_card_id
- New integrations are encouraged to use model_external_id in all VOD related APIs. Encoding and definition parameters should not be used since model_external_id encapsulates both parameters and more. However, if the system was upgraded from an older version, model_external_id should not be used, unless the client (e.g. front-end application) is updated to use model_external_id in all relevant APIs.
- At least one terminal identifier must be specified (mac_address, serial_number, smart_card_id). If no terminal identifier is passed and a new viewing session is opened, the client will be unable to close it since the CloseViewingSession action requires at least one terminal identifier.
- When opening a session to SVOD which has access to all, no sessions on the service will be opened.
- When the SimultaneousViewsRiGHTvControl is set to "On", RiGHTv fully controls the simultaneous views limit feature. In this configuration, RiGHTv decides whether to allow or refuse a new viewing session, based on the current session count and the maximum session count configured in the system.
- When SimultaneousViewsRiGHTvControl is set to "Off", RiGHTv does not manage the simultaneous views limit feature, it is managed by the CDN plugin or not managed at all.
- When SimultaneousViewsRiGHTvControl is set to "Customized", RiGHTv partially manages the simultaneous views limit feature, while it stores allowed sessions, the CDN plugin decides whether to allow or refuse a new viewing session request - only for sessions that were created successfully. Refused sessions are not stored. "Customized" mode allows project customization for this feature.
- If simViewingEnforceSingleSessionPerDevice is enabled, then that only one playback is possible per device at a specific time. Therefore when a new playback is requested, existing sessions are automatically closed. This configuration is only relevant in case SimultaneousViewsRiGHTvControl is set to "On"/"Customized".
- Multi Definition Playback is not supported for pre-order eST.

Examples

```
1 GetVideoDownloadInfo?video_id=123&encoding=MPEG-2&terminal_identifier=td123
1 GetVideoDownloadInfo?video_external_id=abc&encoding=MPEG-2&terminal_identifier=td123
1 GetVideoDownloadInfo?video_id=123&encoding=MPEG-2&model_external_id=PC
1 GetVideoDownloadInfo?video_id=123&encoding=MPEG-2&model_external_id=PC&encoding=MPEG-2
1 GetVideoDownloadInfo?
  video_id=123&encoding=MPEG-2&model_external_id=PC&encoding=MPEG-2&definition=HD
```

See Also

- [GetVideoPlayingInfo](#)
- [GetAttachmentPlayingInfo](#)
- [GetAttachmentDownloadInfo](#)

GetVideoList

since 5.3

An action used to return a list of video pages from RiGHTv.

Deprecated since 7.1

Parameters

Name	Type	Description
category_id	Number	The ID of the parent page as defined in RiGHTv.
external_category_id	String	The external ID of the parent page (category).
root_id	Number	The ID of the root page as defined in RiGHTv, under which the list should be retrieved. This parameter should be used in combination with the external_asset_id parameter.
external_root_id	String	The external ID of the root page, under which the list should be retrieved. This parameter should be used in combination with the external_asset_id parameter.
external_asset_id	String	The external ID of the video asset as defined by the content provider. This parameter should be used in combination with the root_id/external_root_id parameter.
content_provider	Number	The ID of the content provider. This parameter should be used in combination with the root_id/external_root_id parameter.

Name	Type	Description
external_content_provider_id	String	The external ID of the content provider. This parameter should be used in combination with the root_id/external_root_id parameter.
encoding	String	Filter according to encodings. A comma separated list of encodings. No limitation. The encodings cannot contain commas, as they are used as the delimiter.
limited	Boolean	A boolean that indicates whether to retrieve LimitedVideo(true) or Video(false). The default is false.
order_by	String	An indicator on how to sort the list. Possible options are: "tree", "name" or "broadcast_time". The default is order by "tree". When sorting by name it is case insensitive.
order	String	An indicator whether to sort the list by ascending ("asc") or descending ("desc"). The default is "desc".
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
external_channel_id	String	A filtering option. Filter by this external channel ID. Identifies the channel from which the program was recorded. Equivalent to the channel reference ID that is defined in RiGHTv.

Returns

Name	Description
Video[]	When limited parameter is false.
LimitedVideo[]	When limited parameter is true.

Throws

Name	Description
INVALID_CATEGORY_ID	The supplied category ID is invalid.

Name	Description
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions (can only occur when the VALIDATE_SECURITY_ON_PAGE_RETRIEVAL from supported functionality is true).

Comments

- There are several permutations available using this API:
 - external_category_id/category_id
 - root_id/external_root_id + external_asset_id
- This API can be used by using internal IDs which is deprecated and not recommended or by using external IDs.

Examples

1	GetVideoList?category_id=123&encoding=MPEG-2,MPEG-4
1	GetVideoList?external_category_id=abc&encoding=MPEG-2,MPEG-4
1	GetVideoList?external_root_id=abc&external_asset_id=HBO&external_content_provider_id=1
1	GetVideoList?root_id=30&external_asset_id=HBO&external_content_provider_id=1
1	GetVideoList?external_root_id=abc&external_asset_id=HBO&external_content_provider_id=1

See Also

- [GetVideo](#)

GetVideoPackage

since 6.2

An action used to return a video package according to the external ID given.

Parameters

Name	Type	Description
external_id	String	The external ID of the video package page.

Returns

Name	Description
VideoPackage	

Throws

Name	Description
INVALID_PAGE_ID	The supplied video package page ID is invalid.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Examples

1	GetVideoPackage?external_id=123
---	---------------------------------

GetVideoPackageList

since 7.1

An action used to retrieve all video packages that contain the asset.

Parameters

Name	Type	Description
external_asset_id	String	The external ID of the video asset as defined by the content provider.
external_content_provider_id	String	The external ID of the content provider. This parameter should be used in combination with the external_asset_id parameter.
external_root_ids	String	A semicolon separated list of external IDs of root pages, under which the video packages of the asset will be retrieved. Default ID is 1 (meaning all of the RiGHTv catalog).
statuses	String	Filter according to statuses. A semicolon separated list of the following valid values: <ul style="list-style-type: none"> • "Published" • "Announced" Default is "Published".
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
store_id	String	The ID of the app store.

Returns

Name	Description
VideoPackage[]	

Throws

Name	Description
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Comments

- When the supplied external_asset_id is not unique in the system external_content_provider_id should be supplied in order to identify the unique asset. if the external_content_provider_id is not supplied and the external_asset_id is not unique in the system GENERAL_SERVER_ERROR exception will be thrown.
- If one of the supplied parameters external_asset_id/external_content_provider_id/ external_root_ids don't exists in the system an empty list will be returned.

Examples

1	GetVideoPackageList?external_asset_id=123
---	---

GetVideoPlayingInfo*since 5.3*

Returns the playing information for an asset.

Parameters

Name	Type	Description
video_id	Number	The movie page ID.
video_external_id	String	The movie page external ID.
encoding	String	The encoding format of the media file to be played (e.g. "MPEG-2"). If more than one media file is found according to the model ID, the media files are filtered according to the encoding.
resume	Boolean	When set to true, calculates the resume position in the video and returns it as part of the video playing information. When set to false, the returned position is 0. The resume position can be calculated based on a specific chapter or bookmark (if the relevant ID is provided) or based on the last marked position. The default is false.
bookmark_id	Number	The ID of the bookmark for which the playing information is requested. resume=true parameter must be passed as well with this parameter.
chapter_id	Number	The ID of the chapter for which the playing information is requested. resume=true parameter must be passed as well with this parameter.

Name	Type	Description
terminal_id	Number	The internal ID of the account's device. This parameter should be used when integrating with the video server in order to retrieve the video viewing sessions per device.
definition	String	The content definition of the media file for which to return the video playing info, that must be equal to or lower than the definition specified for the page. If no ticket exists with a matching media file of the specified definition: if Multi Definition Playback is not enabled in the Configuration, the content definition is ignored and the highest priority media file will be returned, if Multi Definition Playback is enabled, an error will be returned. If content definition is not specified, the highest priority media file will be returned. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.
model_external_id	String	The external ID of the device model.
mac_address	String	The MAC address of the device that requested the playing info.
serial_number	String	The serial number of the device that requested the playing info.
smart_card_id	String	The smart card ID of the device that requested the playing info.

Returns

Name	Description
VideoPlayingInfo	

Throws

Name	Description
INVALID_DEFINITION	The supplied definition is invalid.
INVALID_ENCODING	The supplied encoding is invalid.
INVALID_TERMINAL	Either the terminal is not valid / registered for the logged-in household, or the passed terminal identifier does not match the logged-in terminal.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.

Name	Description
INVALID_TICKET	The logged-in household has no ticket for this video, or the ticket expired.
INVALID_VIDEO_ID	The supplied video page ID is invalid.
MAX_SIMULTANEOUS_VIEWS_REACHED	The maximum number of simultaneous viewing sessions has been reached.
NO_MATCHING_MEDIA_FILE_FOR_DEFINITION	There is no media file that matches the definition specified.
NO_SUITABLE_MEDIA_FILE	No suitable media file for the given properties.
TARGETED_ADS_ENCRYPTION_FAILED	The encryption of the targeted advertising contexts has failed.
TERMINAL_MODEL_MISMATCH	The passed model is not matched the terminal model.
TICKET_NOT_ACTIVATED	The ticket was not activated for viewing (only for download).

Comments

- There are several permutations available for using this API:
 - video_id
 - video_external_id
- At least one of the following must be provided:
 - model_external_id - recommended
 - encoding - if this parameter is specified, the Multi Definition Playback will not work correctly.
- New integrations are encouraged to use model_external_id in all VOD related APIs. Encoding and definition parameters should not be used since model_external_id encapsulates both parameters and more. However, if the system was upgraded from an older version, model_external_id should not be used, unless the client (e.g. front-end application) is updated to use model_external_id in all relevant APIs.
- At least one terminal identifier must be specified (mac_address, serial_number, smart_card_id). If no terminal identifier is passed and a new viewing session is opened, the client will be unable to close it since the CloseViewingSession action requires at least one terminal identifier.
- When opening a session to SVOD which has access to all, no sessions on the service will be opened.
- In case of OTT, one of the following parameters is mandatory in order to get a valid token (inside the VideoPlayingInfo): mac_address, serial_number, smart_card_id and terminal_id. For more details see Appendix B - Purple DRM API Service.
- When the SimultaneousViewsRiGHTvControl is set to "On", RiGHTv fully controls the simultaneous views limit feature. In this configuration, RiGHTv decides whether to allow or refuse a new viewing session, based on the current session count and the maximum session count configured in the system.
- When SimultaneousViewsRiGHTvControl is set to "Off", RiGHTv does not manage the simultaneous views limit feature, it is managed by the CDN plugin or not managed at all.
- When SimultaneousViewsRiGHTvControl is set to "Customized", RiGHTv partially manages the simultaneous views limit feature, while it stores allowed sessions, the CDN plugin decides whether to allow or refuse a new viewing session request - only for sessions that were created successfully. Refused sessions are not stored. "Customized" mode allows project customization for this feature.
- If simViewingEnforceSingleSessionPerDevice is enabled, then only one playback is possible per device at a specific time. Therefore when a new playback is requested, existing sessions are automatically closed. This configuration is only relevant in case SimultaneousViewsRiGHTvControl is set to "On"/"Customized".
- Multi Definition Playback is not supported for pre-order eST.

Examples

```

1 GetVideoPlayingInfo?video_id=123&encoding=MPEG-2
1 GetVideoPlayingInfo?video_external_id=abc&encoding=MPEG-2
1 GetVideoPlayingInfo?video_external_id=abc&encoding=MPEG-2&definition=HD
1 GetVideoPlayingInfo?video_external_id=abc&encoding=MPEG-2&model_external_id=PC
1 GetVideoPlayingInfo?video_external_id=abc&model_external_id=PC
1 GetVideoPlayingInfo?
  video_external_id=abc&model_external_id=PC&encoding=MPEG-2&definition=HD

```

See Also

- [GetAttachmentPlayingInfo](#)
- [GetVideoDownloadInfo](#)
- [GetAttachmentDownloadInfo](#)

GetVideoPurchaseOptions**since 8.1**

Gets all available purchase options for the specified asset/TV show series/TV show season. For assets gets all announced/published movie pages including SVOD services, Service Plans that include the SVOD services and active VOD cards. For an asset that is an episode, returns all available Season Offer options for the episode's parent season page. Where an asset is associated with a movie page that is part of a movie package, that movie package is also returned. Supported device models are returned for each asset page. For TV show series/season gets all published pages, including the page or its parent related SVOD services and active VOD cards. For seasons, also returns all available season offer options for the pages through which they can be purchased. For assets that are episodes, returns all available season offer options for the parent season page of the asset through which they can be purchased. Discount information is only returned if the discount is valid within the purchase option's availability period. If store_id is provided, gets a list of store product reference IDs for the specified asset at that App store (isInAppPurchaseSupported must be set to true in the Configuration). In this version only TVOD and SVOD content are available for purchase with In-App purchases. Offers and pricing options that are not related to the specified App store are not returned. If isMultiDefinitionSupported is set to true in the Configuration, then for each device model, the availableDefinitions parameter will contain all available content definitions for that device model including definitions that are lower than the content definition for the page. If isMultiDefinitionSupported is set to false, then the availableDefinitions parameter will be null. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.

Parameters

Name	Type	Description
series_external_id	String	The TV show series external ID.
season_external_id	String	The TV show season external ID.
external_asset_id	String	The asset external ID.
external_content_provider_id	String	The content provider external ID.
store_id	String	The ID of the App store.

Returns

Name	Description
VideoPurchaseInfo[]	A list of available video purchase options.

Throws

Name	Description
INVALID_VIDEO_ASSET_ID	The supplied asset doesn't exist in the system.
TV_SHOW_NOT_EXISTS	The supplied series/season doesn't exist in the system.

Name	Description
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Comments

- One of the following parameters must be provided: series_external_id, season_external_id or external_asset_id.

Examples

```
1 GetVideoPurchaseOptions?series_external_id=123
```

```
1 GetVideoPurchaseOptions?season_external_id=123
```

```
1 GetVideoPurchaseOptions?external_asset_id=123&external_content_provider_id=1
```

```
1 GetVideoPurchaseOptions?external_asset_id=123
```

See Also

- [VideoPurchaseInfo](#)

GetVideoSupportedModelList**since 7.1**

Returns a list of device models that are supported for the specified video. If isMultiDefinitionSupported is set to true in the Configuration, then for each device model, the availableDefinitions parameter will contain all available content definitions for that device model including definitions that are lower than the content definition for the page. If isMultiDefinitionSupported is set to false, then the availableDefinitions parameter will be null. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled.

Parameters

Name	Type	Description
video_external_id	String	The external ID of the video
attachment_name	String	The attachment name. If specified, then only device models that are supported for the attachment media files are returned.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
TerminalModel[]	

Throws

Name	Description
INVALID_ATTACHMENT	The attachment name specified is not valid.

Name	Description
INVALID_VIDEO_ID	The video ID specified is not valid.

Examples

```
1 GetVideoSupportedModelList?video_external_id=123
```

```
1 GetVideoSupportedModelList?video_external_id=123&attachment_name=trailer
```

See Also

- [TerminalModel](#)

GetVideoTicketList

since 5.3

Returns a list of video tickets belonging to the logged-in account (household). The returned list includes tickets that were opened for movies received as gifts. If Kids' Profile functionality is enabled, and the logged-in profile has been assigned a kids' profile, the response will only include tickets for movies that are located in the kids' profile catalog category and that do not exceed the maximum parental rating level set for the relevant kids' profile.

Parameters

Name	Type	Description
video_card_ticket_id	Number	The ID of the video card ticket. (Mandatory when filter_vod_card_expired is false)
order_by	String	indicates how to sort the list. [Values: date (default), name]
order	String	indicates whether to sort the list by ascending or descending. [Values: asc(default), desc]
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).
video_package_id	String	The external ID of the ordered video package for which to return its tickets.
type	String	Used to filter the returned video tickets by their type. Values: <ul style="list-style-type: none"> subscription - SVOD only transaction - all other tickets
attachment_name	String	Indicates the attachment name of the video that the ticket belongs to. If supplied will return the attachment according to the specified name. if not supplied the attachment will be empty.

Name	Type	Description
est_filter	String	indicates whether to return eST tickets, non-eST tickets or all tickets. [values: EST, NON_EST, ALL (default)]
est_expired_filter	String	indicates whether to return expired eST tickets, non-expired eST tickets or all eST tickets. Ignored if filter_vod_card_expired is supplied. [Values: EXPIRED, NOT_EXPIRED, ALL (default)]
filter_video_package_expired	Boolean	Indicates whether to return expired tickets for video packages. Ignored for eST movie package. <ul style="list-style-type: none"> • true - Returns only active tickets of the supplied VOD package. (Default) • false - Both active and expired tickets are returned for the specified video package. Expired tickets of all eST content (package and non-package related) are returned if est_expired_filter and est_filter values are supplied.
filter_vod_card_expired	Boolean	Indicates whether to return expired tickets for the specified VOD card. Relevant only when video_card_ticket_id is specified. Ignored for eST movie package. If a value is specified, est_expired_filter is ignored. If video_card_ticket_id is specified and filter_vod_card_expired and est_expired_filter are not specified, then only active tickets for the specified VOD card are returned. When set to false - active and expired tickets are returned relevant for the specified video card ticket. When set to false - video_card_ticket_id is mandatory. <ul style="list-style-type: none"> • true - Returns only active tickets for the specified VOD card. (Default) • false - Both active and expired tickets are returned for the specified VOD card. When set to false video_card_ticket_id is mandatory.
filter_no_resume_position	Boolean	Indicates whether to return only tickets with a resume position. <ul style="list-style-type: none"> • true - Returns only tickets with a resume position. • false - Both tickets with and without a resume position are returned. (Default)

Name	Type	Description
tv_show_ticket_id	Number	The season pass ticket ID. If supplied, returns all active and expired tickets of the episodes belonging to the specified season pass.
external_asset_id	String	The external ID of the video asset as defined by the content provider.

Returns

Name	Description
VideoTicket[]	A list of video tickets.

Throws

Name	Description
INVALID_PACKAGE_ID	There is no video package in the system with the specified ID.
MISSING_PARAMETER	One of the mandatory parameters is missing.

Examples

```
1 | GetVideoTicketList?order_by=name
```

See Also

- [OrderVideo](#)

GetVideoTvShowEpisode*since 8.1*

Gets the aggregated video for an episode. Use one of the following permutations:

- **season_external_id** and **episode_number**: Gets the aggregated video related to the specified episode.
- **series_external_id**, **season_number** and **episode_number**: Gets the aggregated video related to the specified episode.

Parameters

Name	Type	Description
season_external_id	String	The TV show season external id.
episode_number	Number	The episode number.
series_external_id	String	The TV show series external ID.
season_number	Number	The TV show season number.

Returns

Name	Description
AggregatedVideo	The aggregated video for the episode.

Throws

Name	Description
ENTITY_NOT_AVAILABLE	The episode is not available due to missing movie pages.
EPISODE_NOT_EXIST	The episode does not exist.
TV_SHOW_NOT_EXISTS	The TV show series/season does not exist.

Examples

1 GetVideoTvShowEpisode?season_external_id=123&episode_number=123

1 GetVideoTvShowEpisode?series_external_id=123&season_number=123&episode_number=123

See Also

- [AggregatedVideo](#)

GetVideoTvShowEpisodeList

since 8.1

Gets the aggregated video list ordered by the episode number. When include_removed_from_service is set to true, episodes whose page status is Removed from Service, are also returned. Where an episode is associated with a movie page that is part of a movie package, and include_extended_info is set to true, that movie package is also returned as part of the extended object. If isMultiDefinitionSupported is set to true in the Configuration, and include_extended_info is set to true, then for each device model, the availableDefinitions parameter will contain all available content definitions for that device model including definitions that are lower than the content definition for the page. If isMultiDefinitionSupported is set to false, than the availableDefinitions parameter will be null. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled. Use one of the following permutations:

- season_external_id: Gets the aggregated video list of all pages related to the specified season.
- series_external_id and season_number: Gets the aggregated video list of all pages related to the specified season.
- external_category_id: Gets the aggregated video list of all pages under the TV show season folder.

Parameters

Name	Type	Description
season_external_id	String	The TV show season external ID.
series_external_id	String	The TV show series external ID.
season_number	Number	The TV show season number.
external_category_id	String	The external folder ID. When this parameter is supplied, returns AggregatedVideo instead of ExtendedAggregatedVideo.

Name	Type	Description
max_pr_level	Number	The maximum level of parental rating allowed. When provided, returns TV show episodes that do not have parental ratings or that have parental ratings equal to or less than this level.
include_extended_info	Boolean	A boolean that indicates whether to include video purchase information and movie packages associated with the movie pages in the result. If true, returns a list of ExtendedAggregatedVideo entities. If omitted or false (default), returns a list of AggregatedVideo entities instead of a list of ExtendedAggregatedVideo entities (used for backward compatibility). (Since 8.3 SR1).
availability	String	Filter according to availability. This filter can be left empty for no filtering, or contain the value "for_download" to limit returns to TV Shows Episodes that are available for download. An episode is considered available for download if at least one of the following conditions is fulfilled: <ul style="list-style-type: none"> • The episode is linked to a pricing matrix that includes a pricing option of type Download, Converged, or EST. • The episode is part of a season that is linked to a pricing matrix that includes a pricing option of type Download, Converged, or EST. • The episode is part of a video package that has an EST price or that has a delivery mode of type Download, Converged, or Download-And-Streaming.
include_removed_from_service	Boolean	Indicates whether to return episodes whose page status is Removed from Service (4). The default is false.

Returns

Name	Description
AggregatedVideo[]	The aggregated video list.
ExtendedAggregatedVideo[]	The movie pages of the asset, movie packages associated with the movie pages, as well as purchase information in an aggregated video entity.

Throws

Name	Description
INVALID_PAGE_ID	The TV show season page does not exist.
INVALID_PAGE_TYPE	The page type is not TV show season.
TV_SHOW_NOT_EXISTS	The TV show series or season does not exist.

Examples

```
1 GetVideoTvShowEpisodeList?season_external_id=123&include_extended_info=true
```

```
1 GetVideoTvShowEpisodeList?external_category_id=123
```

```
1 GetVideoTvShowEpisodeList?series_external_id=123&season_number=123
```

GetVideoTvShowSeason*since 8.1*

Gets a video TV show season entity with related pages. To retrieve the TV show season entity that is related to a page whose status is Removed from Service, specify the page_external_id parameter, using TvShowTicket.tvShowPageId returned by GetTVShowTicketList. This is possible only for seasons that were ordered by the user. Use one of the following permutations:

- season number and series_external_id: Gets video TV show season entity with all season pages related to the series. If there are several pages for a season, all pages are returned.
- season_external_id: Gets video TV show season entity with all season pages related to the season. If there are several pages for a season, all pages are returned.
- page_external_id: Gets video TV show season entity with specified season page.

Parameters

Name	Type	Description
page_external_id	String	The TV show season page external ID.
season_external_id	String	The TV show season external ID.
season_number	Number	The TV show season number.
series_external_id	String	The TV show series external ID.

Returns

Name	Description
VideoTvShowSeason	Video TV show season entity.

Throws

Name	Description
ENTITY_NOT_AVAILABLE	The TV show season is not available due to missing pages.
INVALID_PAGE_ID	The TV show season page does not exist.
INVALID_PAGE_TYPE	The page type is not TV show season.

Name	Description
PERMISSION_DENIED	The logged-in user does not have permission to access the required content due to security restrictions.
TV_SHOW_NOT_EXISTS	The TV show season does not exist.

Examples

```
1 GetVideoTvShowSeason?page_external_id=123
```

```
1 GetVideoTvShowSeason?series_external_id=123&season_number=123
```

```
1 GetVideoTvShowSeason?season_external_id=123
```

See Also

- [VideoTvShowSeason](#)

GetVideoTvShowSeasonList

since 8.1

Gets all the TV show season entities with related pages by specified parent folder or by specified TV show series sorted by season_number. Use one of the following permutations:

- **series_external_id**: Gets all video TV show season entities related to the specified series. If there are several pages for a season, all pages are returned.
- **external_category_id**: Gets all video TV show season entities under this folder.

Parameters

Name	Type	Description
external_category_id	String	The folder external ID.
series_external_id	String	The TV show series external ID.
max_pr_level	Number	The maximum level of parental rating allowed. When provided, returns TV show seasons that do not have parental ratings or that have parental ratings equal to or less than this level.

Name	Type	Description
availability	String	<p>Filter according to availability. This filter can be left empty for no filtering, or contain the value "for_download" to limit returns to TV Shows Seasons that are available for download. A season is considered available for download if at least one of the following conditions is fulfilled:</p> <ul style="list-style-type: none"> • An offer for the season exists that has a pricing matrix that includes a pricing option of type Download, Converged, or EST. • The season includes an episode that is linked to a pricing matrix that includes a pricing option of type Download, Converged, or EST. • The season includes an episode that is part of a video package that has an EST price or that has a delivery mode of type Download, Converged, or Download-And-Streaming.

Returns

Name	Description
VideoTvShowSeason[]	A list of video TV show season entities.

Throws

Name	Description
INVALID_PAGE_ID	The folder does not exist.
INVALID_PAGE_TYPE	The folder type is not General or TV show series.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.
TV_SHOW_NOT_EXISTS	The TV show series does not exist.

Examples

```
1 GetVideoTvShowSeasonList?external_category_id=123
```

```
1 GetVideoTvShowSeasonList?series_external_id=123
```

See Also

- [VideoTvShowSeason](#)

GetVideoTvShowSeries

since 8.1

Gets video TV show series entity with related pages by series_external_id or page_external_id. One of the parameters must be provided.

Parameters

Name	Type	Description
page_external_id	String	The TV show series page external ID.
series_external_id	String	The TV show series external ID.

Returns

Name	Description
VideoTvShowSeries	Video TV show series entity.

Throws

Name	Description
ENTITY_NOT_AVAILABLE	The TV show series is not available due to missing pages.
INVALID_PAGE_ID	The folder does not exist.
INVALID_PAGE_TYPE	The folder type is not General.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.
TV_SHOW_NOT_EXISTS	The TV show series does not exist.

Examples

```
1 | GetVideoTvShowSeries?page_external_id=123
```

```
1 | GetVideoTvShowSeries?series_external_id=123
```

See Also

- [VideoTvShowSeries](#)

GetVideoTvShowSeriesList

since 8.1

Returns all the TV show series entities with related pages under the specified parent folder in the VOD Catalog, sorted by name.

Parameters

Name	Type	Description
external_category_id	String	The external ID of the parent folder.

Name	Type	Description
max_pr_level	Number	The maximum level of parental rating allowed. When provided, returns TV show series that do not have parental ratings or that have parental ratings equal to or less than this level.
availability	String	Filter according to availability. This filter can be left empty for no filtering, or contain the value "for_download" to limit returns to TV Shows Series that are available for download. A series is considered available for download if at least one of the following conditions is fulfilled: <ul style="list-style-type: none"> An offer for the series exists that has a pricing matrix that includes a pricing option of type Download, Converged, or EST. The series includes an episode that is linked to a pricing matrix that includes a pricing option of type Download, Converged, or EST. The series includes an episode that is part of a video package that has an EST price or that has a delivery mode of type Download, Converged, or Download-And-Streaming.

Returns

Name	Description
VideoTvShowSeries[]	A list of TV show series entities data and related pages under specified parent folder.

Throws

Name	Description
INVALID_PAGE_ID	The folder does not exist.
INVALID_PAGE_TYPE	The folder type is not General.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Examples

```
1 GetVideoTvShowSeriesList?external_category_id=123
```

See Also

- [VideoTvShowSeries](#)

GetVodGiftList**since 6.2**

An action used to return a list of video gifts for the logged-in household.

Parameters

Name	Type	Description
status	String	The Status to filter the list by. Possible values: <ul style="list-style-type: none">• "new"• "viewed"• "in consumption" See VodGift.status description for more details.
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
VodGift[]	

Examples

```
1 | GetVodGiftList?status=new
```

GetWishList (deprecated)**since 7.0**

Deprecated since 8.5. Use the Wishlist functionality provided in The COMPASS Content Discovery Platform.

An action used for retrieval wish list for a specific household or member. Wish items can be video or video package.

Parameters

Name	Type	Description
order_by	String	An indicator on how to sort the list. Possible options are: "insertion_time", "name" or "removal_time". The default is order by "insertion_time".
order	String	An indicator on whether to sort the list by ascending ("asc") or descending ("desc"). The default is "desc".
scope	String	Possible values: household or member . Wish lists can be returned on the household or the member level. The default is household.

Name	Type	Description
from	Number	The first index requested. Used for paging. (Number type: Integer).
count	Number	The number of elements requested. Used for paging. (Number type: Integer).

Returns

Name	Description
VideoWish[]	

Examples

```
1 GetWishList
1 GetWishList?order_by=name&order=asc
1 GetWishList?order_by=name&order=desc&scope=member
```

See Also

- [VideoWish](#)
- [AddToWishList](#)
- [RemoveFromWishList](#)

InsertMessage*since 7.2*

An action used to send a self message to the logged-in household (can be used to pass messages between devices of the same household).

Parameters

Name	Type	Description
type	String	The message type.
title	String	The message title.
body	String	The message body.
message_external_id	String	The message external ID. When not provided, the message internal ID will be sent.
properties_names	String	The message properties names delimited by semicolon.
properties_values	String	The message properties values delimited by semicolon.

Returns

Name	Description
Result	

Throws

Name	Description
EXTERNAL_ID_ALREADY_EXIST	The message external ID already exists in the system.

Comments

- Semicolon and tabs should be filtered from the properties names and values by the client.

Examples

```
1 InsertMessage?type=type1
```

See Also

- [DeleteMessage](#)
- [GetMessageList](#)

IsAlive*since 8.0*

Checks if the system is alive. Returns the current date and time. If the system is down, nothing is returned. Since 8.0 SR4.

Returns

Name	Description
String	"RTE - I am alive. Current Date = [value of current date]"

Throws

Name	Description
GENERAL_SERVER_ERROR	Internal Server error occurred: Database is not responding.

Examples

```
1 IsAlive
```

Login*since 5.3*

An action used to log in to the Front-end Integration.

Parameters

Name	Type	Description
username	String	The username as defined in the household in RiGHTv.
password	String	The password as defined in the household in RiGHTv.
pin_code	String	The purchase PIN code of the member performing the action. The default is the main member.

Name	Type	Description
household_id	Number	The household ID.
household_external_id	String	The household external ID.
mac_address	String	The MAC address as defined in the terminals tab of the household in RiGHTv.
serial_number	String	The serial number as defined in the terminals tab of the household in RiGHTv.
smart_card_id	String	The smart card ID as defined in the terminals tab of the household in RiGHTv.
ef_name	String	An extra field name (e.g. "IP").
ef_value	String	An extra field value (e.g. "1.2.3.4").
meta_names	String	list of metadata names delimited by semicolons. Should be used with meta_values parameter. Will not be validated in the login process. (e.g. name1;name2;name3)
meta_values	String	list of metadata values delimited by semicolons. Should be used with meta_names parameter. Will not be validated in the login process. (e.g. value1;value2;value3)
user_token	String	The user token provided by an external system.
third_party	String	The name of the third party used for user authentication, if any. [Valid values: facebook] Since 9.0 SR1
first_name	String	The user first name.
last_name	String	The user last name.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_CREDENTIALS	In case the arguments supplied do not match a valid user. Or when a suspended/inactive user is trying to login.

Name	Description
LOGIN_NOT_ALLOWED	In case the login permutation is not supported, or in case the maximum number of login attempts was reached.

Comments

- There are several permutations available for login:
 - username, password
 - username, password, mac_address, ef_name, ef_value
 - username, password, mac_address
 - mac_address, smart_card_id
 - mac_address, serial_number
 - household_id, pin_code
 - household_external_id, pin_code
 - serial_number
 - smart_card_id
 - household_external_id, user_token, third_party, first_name, last_name
 - household_external_id, user_token, third_party, serial_number
- Each of the above permutations can be accompanied by pin_code, in order to set the corresponding member as active.
- It is also possible to extend the login action on a project basis.
 - The permutations which are supported for login from the list above are as configured in login-params-config.properties.
 - For security reasons it is recommended to use POST method instead of GET method in order not to expose the login details.

Examples

```
1 | Login?username=1&password=1
```

See Also

- [Logout](#)
- [TechnicianLogin](#)

Logout

since 5.3

An action uses to log out from the Front-end Integration.

Returns

Name	Description
Result	

Examples

```
1 | Logout
```

See Also

- [Login](#)

MarkPosition

since 5.3

An action used to set a resume position for a movie. It is not possible to mark a position on a recording with status Failed (4) or Deleted (5) in the RecordingTicket.

Parameters

Name	Type	Description
video_id	Number	The ID of the movie page as defined in RiGHTv.
video_external_id	String	The external ID of the movie page as defined in RiGHTv.
position	Number	The position in milliseconds from the start of the movie.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_VIDEO_ID	The supplied video ID is invalid.

Comments

- There are several permutations available for using this API:
 - video_id
 - video_external_id

Examples

```
1 MarkPosition?video_id=123&position=10000
```

```
1 MarkPosition?video_external_id=abc&position=10000
```

See Also

- [VideoPlayingInfo](#)
- [GetMarkedPosition](#)

MarkRecordingPosition

since 5.3

Saves a resume position for an NPVR recording or for a program consumed through the Catch-Up or Start Over service. This enables a user to resume watching a recording or program from a specific point. For NPVR 1.0, if isPersonalRecordingSupported is set to true in the Configuration, the recording position is marked at the profile level.

Parameters

Name	Type	Description
recording_id	Number	The recording ID. Relevant for NPVR 1.0 only. Deprecated since 9.0.
program_id	Number	The internal ID of the program (relevant for Catch-up and Start Over).
position	Number	The number of milliseconds that elapsed since the start of the recording or program.
program_external_id	String	The program external ID. Will be ignored if program_id is provided. The program_external_id is unique within a channel only. For Start Over and Catch-Up services when supplying only the program_external_id, it is possible that the same ID is used for multiple programs in different channels, resulting in an exception. Use program_external_id together with channel_external_id to ensure the marked position is saved for the desired program.
channel_external_id	String	The channel external ID of the channel on which the program was broadcast. For Start Over and Catch-Up services when supplying only the program_external_id, it is possible that the same ID is used for multiple programs in different channels, resulting in an exception. Use channel_external_id together with program_external_id to ensure the marked position is saved for the desired program.
otf_recording_id	String	The ID of the recording. Relevant for NPVR 2.0 only. Since 9.0.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_PROGRAM_ID	The supplied program ID does not exist.
INVALID_RECORDING_ID	The recording specified does not exist.
UNSUPPORTED_OPERATION	NPVR is not set to NPVR 2.0.

Comments

- recording_id or program_id must be supplied, but not both of them.
- For NPVR specify the recording_id. For Catch-Up and StartOver specify either program_id, or program_external_id and channel_external_id.

Examples

```
1 | MarkRecordingPosition?recording_id=123&position=10000
```

See Also

- [RecordingPlayingInfo](#)

MarkViewedVodGift*since 6.2*

An action used to set the status of a new VOD gift to "viewed".

Parameters

Name	Type	Description
vod_gift_id	Number	The internal identifier of the gift.
refuse	Boolean	A boolean indicating whether the gift was refused. If set to "true" the gift is marked as refused. If set to false (or not provided) the gift is marked as viewed. The default is false.

Returns

Name	Description
Result	

Examples

```
1 | MarkViewedVodGift?vod_gift_id=123
```

```
1 | MarkViewedVodGift?vod_gift_id=123&refuse=true
```

See Also

- [VodGift](#)

OpenSession (deprecated)*since 7.1, 7.2*

since 7.4

An action used to open a session for remote terminal communication. For example, can be used to implement Remote Client PVR.

Parameters

Name	Type	Description
terminal_id	String	The ID of the terminal for which to open the session.
mac_address	String	The MAC address of the terminal for which to open the session.
serial_number	String	The serial number of the terminal for which to open the session.
smart_card_id	String	The smart card ID of the terminal for which to open the session.
timeout	Number	Maximum time in milliseconds to wait for the session ID. Default is 120000 millis.
offline	Boolean	Open offline session. Default is false.

Returns

Name	Description
ProxySession	

Throws

Name	Description
ALREADY_CONNECTED	There is already an active session between the two terminals.
SESSION_ALREADY_EXIST	A session for the requested terminal already exists (but the remote terminal did not connect to it yet).

Comments

- At least one of the terminal identifiers is mandatory (terminal_id, mac_address, serial_number, smart_card_id).
- For more details on how to use the remote messaging mechanism please read "RiGHTv Remote Client PVR Developers Guide".

Examples

```
1 | OpenSession?terminal_id=1234&timeout=180000
```

```
1 | OpenSession?terminal_id=1234&offline=true
```

See Also

- [CloseSession](#)
- [GetSessionList](#)
- [SendMessage](#)
- [PollMessage](#)
- [SendResponse](#)

OpenViewingSession

since 7.3

An action used to open a new viewing session both for VOD and SVOD.

Parameters

Name	Type	Description
video_external_id	String	The external ID of the movie page for which to open the viewing session
mac_address	String	The MAC address of the terminal that requested the viewing session.
serial_number	String	The serial number of the terminal that requested the viewing session.
smart_card_id	String	The smart card ID of the terminal that requested the viewing session.

Throws

Name	Description
GENERAL_SERVER_ERROR	The limit simultaneous views feature is not enabled.
INVALID_TERMINAL	Either the terminal is not valid / registered for the logged-in household, or the passed terminal identifier does not match the logged-in terminal.
INVALID_TICKET	The logged-in household has no ticket for this video, or the ticket has expired.
INVALID_VIDEO_ID	The supplied video page ID is invalid.
MAX_SIMULTANEOUS_VIEWS_REACHED	The maximum number of simultaneous viewing sessions has been reached.

Comments

- At least one of the terminal identification parameters (mac_address, serial_number, smart_card_id) must be supplied or user should login with terminal.
- When opening a session to SVOD which has access to all, no sessions on the service will be opened.
- When the SimultaneousViewsRiGHTvControl is set to "On", RiGHTv fully controls the simultaneous views limit feature. In this configuration, RiGHTv decides whether to allow or refuse a new viewing session, based on the current session count and the maximum session count configured in the system.
- When SimultaneousViewsRiGHTvControl is set to "Off", RiGHTv does not manage the simultaneous views limit feature, it is managed by the CDN plugin, Session Manager, or not managed at all.
- When SimultaneousViewsRiGHTvControl is set to "Customized", RiGHTv partially manages the simultaneous views limit feature, while it stores allowed sessions, the CDN plugin decides whether to allow or refuse a new viewing session request - only for sessions that were created successfully. Refused sessions are not stored. "Customized" mode allows project customization for this feature.
- If simViewingEnforceSingleSessionPerDevice is enabled, then only one playback is possible per device at a specific time. Therefore when a new playback is requested, existing sessions are automatically closed. This configuration is only relevant in case SimultaneousViewsRiGHTvControl is set to "On"/"Customized".

Examples

```
1 OpenViewingSession?video_external_id=1234&mac_address=1234
```

```
1 OpenViewingSession?video_external_id=1234&serial_number=1234
```

```
1 OpenViewingSession?video_external_id=1234&smart_card_id=1234
```

OrderGenericService

since 5.3

An action used to order a service as a generic transaction. The service may be a part of RiGHTv, or an external service registered in an external system.

Parameters

Name	Type	Description
service_name	String	The name of the service.
business_model	String	The model under which the service is consumed by the subscriber, for example: "last X days" or "start over".
price	Number	The price the subscriber pays. (Number Type: Double).
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Name	Type	Description
service_id	Number	The external ID of the service being ordered.
payment_mode	String	The order payment mode. Possible values: <ul style="list-style-type: none">• "normal"• "internal prepaid" The default is "normal".

Returns

Name	Description
Result	

Throws

Name	Description
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INSUFFICIENT_CREDIT	There is insufficient credit for subscriber to make the purchase due to reaching the spending limit or exceeding the internal/external prepaid amount.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. (Relevant for non operator subscribers).

Examples

```
1 OrderGenericService?service_name=Foo&business_model=last X days&price=5
```

OrderPPVEvent

since 8.4

An action used to order a PPV event.

Parameters

Name	Type	Description
external_id	String	The PPV event external ID. (Mandatory)
price	Number	The price of the PPV event. (Number Type: Double).
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INVALID_PRICE	The price supplied does not match the actual price.
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. (Relevant for non operator subscribers).
PPV_EVENT_EXPIRED	PPV event with supplied external id expired.
PPV_EVENT_NOT_FOUND	The PPV event specified does not exist.
TICKET_EXISTS	A ticket for the same PPV event already exists.
UNSUPPORTED_OPERATION	PPV event feature is not supported.

Examples

```
1 OrderPPVEvent?external_id=ppvevent123&price=90
```

See Also

- [PPVEvent](#)

OrderProgram

since 5.3

An action used to order a Pay-Per-View program. deprecated since 8.4

Parameters

Name	Type	Description
program_id	Number	The ID of the program.
price	Number	The price shown to the subscriber; used to verify that the subscriber pays as shown. (Number Type: Double).
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.
bouquet_id	Number	The bouquet ID number.
model_external_id	String	The model external ID.

Returns

Name	Description
Result	

Throws

Name	Description
CHANNEL_BOUQUET_NOT_ASSOCIATE	The channel is not associated with a bouquet.
CHANNEL_NOT_VIEWABLE	Program is not viewable on specified terminal or not viewable on any household terminal.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INSUFFICIENT_CREDIT	There is insufficient credit for subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_BOUQUET_ID	The bouquet ID is invalid.
INVALID_CHANNEL_ID	The given program ID does not belong to a PPV channel.
INVALID_PRICE	The price supplied does not match the actual price.
INVALID_PROGRAM_ID	The given program ID does not exist.
INVALID_TERMINAL_MODEL	The model ID is invalid.
PAST_PROGRAM	The given program is in the past.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.
PLUGIN_NOT_FOUND_ERROR	The content protection plugin was not found.
TICKET_EXISTS	The ticket already exists for the household for the given program.

Comments

- The program must reside on a PPV channel.

Examples

```
1 OrderProgram?program_id=123
```

See Also

- [Program](#)
- [Channel](#)
- [ProgramTicket](#)
- [GetProgramTicketList](#)

since 7.4

OrderRecording

Creates an NPVR order from the specified program (EPG recording) for the logged-in account. NPVR 2.0 supports creating an NPVR order for a specified time (time-based recording) for the logged-in account. Use GetRecordingTicketList to get the recording IDs for the logged-in account. The NPVR quota for an account is defined in the Service entity. The calculation of the account's NPVR quota includes completed recordings, recordings in progress and scheduled recordings for the account. For NPVR 1.0, if isPersonalRecordingSupported is to true in the Configuration, the recording is set at the profile level.

One of the following permutations must be provided:

- program_id
- program_external_id: To avoid the possibility of the program_external_id not being unique, it is recommended to always provide the channel_external_id when providing the program_external_id. The internal program ID may be used, however it is recommended to provide the external program ID. Relevant for EPG based recordings.
- start_time, end_time and channel_external_id: Relevant only for NPVR 2.0 time-based recordings.

Parameters

Name	Type	Description
program_id	Number	The ID of the program to record. Mandatory for NPVR 1.0.
program_external_id	String	The external program ID.
recursive	Boolean	If the program is part of a series, set the value to true to order recording for the entire season. Relevant only for EPG based recordings. The default is false.
channel_external_id	String	The external ID of the channel on which to record. Mandatory for time-based recordings. Not relevant for NPVR 1.0. Since 9.0.

Name	Type	Description
allow_scheduling_over_quota	Boolean	Deprecated since 9.0. Indicates whether to apply the NPVR quota to scheduled recordings. When set to true, the recording will be scheduled without taking the NPVR quota into consideration. When set to false, the recording being requested will only be scheduled if existing recordings, previously scheduled recordings, and the recording being requested will together not exceed the NPVR quota. When set to false and ordering the recording of a series, only those episodes that already exist in the EPG and that are within the EPG searchable period (NPVR availability period) are taken into account when measuring the NPVR quota that will be used. The default value is false. This parameter is ignored for NPVR 2.0.
start_time	Number	The time to start the recording (in milliseconds from 1/1/1970). Relevant only for NPVR 2.0 time-based recordings. Since 9.0.
end_time	Number	The time to stop the recording (in milliseconds from 1/1/1970). Relevant only for NPVR 2.0 time-based recordings. Since 9.0.
name	String	The name to assign the time-based recording. The name is left empty if no name is provided. For EPG-based recordings the recording is assigned the program name as listed in the EPG. Relevant only for NPVR 2.0. Since 9.0.
lead_duration	Number	Creates a buffer before the scheduled recording time to take minor scheduling shifts into account, in minutes. Cannot be a negative number. Relevant only for NPVR 2.0 and for EPG-based recordings. If no lead duration is set for the specific recording, then the global lead duration setting is used. Since 9.0.

Name	Type	Description
lag_duration	Number	Creates a buffer after the scheduled recording time to take minor scheduling shifts into account, in minutes. Cannot be a negative number. Relevant only for NPVR 2.0 and for EPG-based recordings. If no lag duration is set for the specific recording, then the global lag duration setting is used. Since 9.0.
user_comment	String	User comment for the recording. Relevant for NPVR 2.0 only. Not relevant for series recording. Since 9.1.
sector	String	The external ID of the sector. Must match the sector ID of the account. isSectorsManagementSupported in the Configuration must be set to true. Since 9.1SR22.

Returns

Name	Description
Result	The status of the action. For NPVR 2.0 returns also the ID of the recording. Not relevant for series recording. Since 9.1.

Throws

Name	Description
CONCURRENT_RECORDINGS_LI MIT_EXCEEDED	The recording cannot be ordered as doing so would exceed the concurrent recordings limit set for the accounts NPVR service.
INVALID_CHANNEL	The channel technical format is invalid or not NPVR enabled. For NPVR 2.0 the given channel ID does not exist.
INVALID_PROGRAM_ID	The given program ID does not exist or no NPVR enabled channel technical format was found or no household device was found that matches NPVR enabled channel technical format or operation is not allowed on this device. For NPVR 2.0 the given program ID does not exist.
INVALID_SECTOR	The sector ID specified does not match the sector ID of the account.
INVALID_SERIES	The recording order is recursive and the program_id is represents a non series program.
MAX_QUOTA_EXCEEDED	There is insufficient NPVR quota remaining to schedule this recording.
MISSING_PARAMETER	One of the mandatory parameters is missing.

Name	Description
NPVR_NOT_ENABLED_FOR_ANY_TERMINAL_CATEGORY	NPVR must be enabled on at least one device category.
OPERATION_NOT_ALLOWED	There is no On-the-fly technical format with NPVR enabled.
PAST_PROGRAM	The specified program is in the past and outside the availability period. Applies to both time-based and EPG recordings. Relevant for NPVR 2.0 from RiGHTv 9.1 SR18.
RECORDING_FAILED	The recording failed due to video server plugin failure.
SCHEDULED_QUOTA_EXCEEDED	The requested recording would cause quota overflow and allow_scheduling_over_quota is set to false. Relevant for NPVR 1.0.
SERVICE_UNSUBSCRIBED	The logged-in household is not subscribed to the NPVR service.
TICKET_EXISTS	The ticket already exists for the household for the given recording. Relevant for all recording types. For NPVR 2.0 EPG based recordings, time-based recordings and series recordings. For NPVR 1.0 for EPG based recordings and series recordings.
TICKET_EXISTS	The ticket already exists for the household for the given recording. Relevant for all recording types. For NPVR 2.0 EPG based recordings, time-based recordings and series recordings. For NPVR 1.0 for EPG based recordings and series recordings.
UNSUPPORTED_OPERATION	The NPVR model is set to None - NPVR related operations are unsupported.
USER_NOT_SUBSCRIBED_TO_CHANNEL	The user is not subscribed to the channel related to the recording.

Examples

```
1 OrderRecording?program_id=123
```

See Also

- [Program](#)
- [Channel](#)
- [RecordingTicket](#)
- [GetRecordingTicketList](#)
- [GetSeriesTicketList](#)

OrderTVShow

since 8.2

An action used to order a TV show season. The action creates a TvShowTicket according to the details of the specified season pass offer. Online customers are not supported.

Parameters

Name	Type	Description
external_id	String	The TV show season page external ID. (Mandatory)
purchase_option_id	String	A concatenation of the offer ID and the pricingOptionId with an underscore (_) as a delimiter. For example, 55_98. Returned by GetVideoPurchaseOptions for the season. (Mandatory).
price	Number	The price of the TV Show season offer. (Number Type: Double).
payment_mode	String	The order payment mode. Possible values: <ul style="list-style-type: none"> • "normal" • "prepaid" The default is "normal". Since 9.1 SR1.
ignore_existing_tickets	Boolean	Whether to permit an order when a ticket with the same external ID already exists for the requesting account. Values: <ul style="list-style-type: none"> • true - Permit an order when a ticket with the same external ID already exists. • false - Do not allow an order when a ticket with the same external ID already exists. (Default)
terminal_identifier	String	Identifier of the terminal performing the order.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_PENDING_INACTIVATION	The specified household is in the process of inactivation.
INACTIVE_HOUSEHOLD	The specified household is inactive or suspended.
INSUFFICIENT_CREDIT	There is insufficient credit for the subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_OFFER_ID	The offer specified does not exist or is not linked to the specified page.
INVALID_PAGE_ID	The season page ID specified does not exist.

Name	Description
INVALID_PAGE_TYPE	The page referenced by the page ID is not of type season.
INVALID_PRICE	The price specified does not match the actual price of the offer.
NON_OPERATOR_SUBSCRIBER_NOT_ALLOWED	OrderTVShow season is not supported for online customers.
NO_DEVICES_ALLOWED_BY_OFFER	None of the account's devices are allowed by the season pass offer option.
OFFER_IS_NOT_ACTIVE	The offer specified is not active.
PERMISSION_DENIED	The user is not permitted to access the page.
TICKET_EXISTS	A ticket for the same TV show already exists.

Comments

- If the operator updates the price of an offer and the old price is still displayed to the user still shows the old price when the user attempts to order the season and OrderTVShow is called, the INVALID_PRICE exception will be thrown.
- If the operator updates the price of an offer and the old price is still displayed to the user, when the user attempts to order the season and OrderTVShow is called, the INVALID_PRICE exception will be thrown.

Examples

```
1 OrderTVShow?external_id=tvshow123&purchase_option_id=75_89&price=90
```

See Also

- [TvShow](#)
- [GetVideoPurchaseOptions](#)

OrderVideo

since 5.3

Orders a video and creates tickets for all relevant media files for play/download. The relevant media files are selected by 1) selecting only media files with content definition levels equal to or lower than that specified for the page, and then 2) selecting only media files that match the account's devices' matching profiles or according to device specified by the model_external_id (if provided) 3) If Multi Definition Playback feature is not enabled in the Configuration, for each of the account's devices, a ticket is created for the remaining matching media file of the highest definition. If Multi Definition Playback feature is enabled, for each of the account's devices, tickets are created for all remaining matching media files, with their priorities noted. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled. If ordering a TVOD or SVOD video that was purchased as an In-App purchase through an App store, the store_id together with the purchase_reference and the store_product_reference (for Google App store only) provided by the App store must be specified. RiGHTv validates the purchase_reference with the App store and sends the confirmation to the application. The application then completes the purchase with the App store.

Parameters

Name	Type	Description
video_id	Number	The ID of the movie page from RiGHTv.
external_video_id	String	The external ID of the movie page.
video_card_ticket_id	Number	The ID of the video card ticket to use for ordering the video.
pricing_id	Number	The ID of the pricing option. Viewing method is ignored when using eST pricing_id.
subscription	Boolean	A boolean indicator. In case the household is subscribed to SVOD, a value of <code>true</code> uses the subscription model to order the video. The default is <code>false</code> .
price	Number	The price that was shown to the customer; used to verify that the customer will pay as shown. The validation will be on the final price after discount calculation (if a discount exists). If a percentage discount is applied, the price is calculated by reducing the original price by the discount percentage. (Number Type: Double).
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.
payment_mode	String	The order payment mode. Possible values: <ul style="list-style-type: none"> • "normal" • "prepaid" • "internal_prepaid" The default is "normal".
viewing_method	String	The viewing method. Possible values: <ul style="list-style-type: none"> • "streaming" • "download" • "converged" • "est" The default is "converged" from 7.3 (streaming is the default in previous versions). viewing_method parameter is ignored when using eST pricing_id.
vod_gift_id	Number	The ID of the video gift to be used for ordering this video.
discounted_option_id	Number	The ID of the discounted pricing option (of the discount that id attached to the video).

Name	Type	Description
terminal_identifier	String	An identifier of the device used, e.g. its MAC address. No validation is performed on this parameter.
model_external_id	String	The external ID of the device model. If there is no matching media file for the device, an exception is thrown.
validateUserSubscriptions	Boolean	Deprecated since version 8.3 SR2. Functionality is now provided by the validateExistingRights flag. A boolean indicator which when set to true performs a validation that the asset does not exist in one of the account's SVOD subscriptions. Default is false.
validateUserSeasonPasses	Boolean	Deprecated since version 8.3 SR2. Functionality is now provided by the validateExistingRights flag. A boolean indicator which when set to true performs a validation that the episode/asset is available through one of the account's Season Passes. Default is false.
validateExistingRights	Boolean	Indicates whether to check if the logged-in account has a VOD gift in New/Viewed status for the specified movie or if the movie is available through one of the account's SVOD subscriptions or Season Passes (true). If set to true, validateUserSubscriptions and validateUserSeasonPasses are ignored. Default is false.
store_id	String	The ID of the app store. (Mandatory if purchase_reference is specified)
purchase_reference	String	The reference to a purchase made for the specified video at the App store. Use the POST method to pass this parameter. (Mandatory if store_id is specified)
store_product_reference	String	The reference assigned to the product purchased at a particular App store.

Returns

Name	Description
Result	

Throws

Name	Description
CANNOT_BE_ORDERED_MORE_THAN_ONCE	This video has been ordered already using this VOD Card and cannot be ordered again.
CONSUMED_VOD_GIFT	The specified video gift was already consumed.
CP_HOUSEHOLD_NO_TERMINALS	The household has no terminals defined in the CAS.
DISCOUNT_AND_PRICING_OPTION_MISMATCH	The discount option ID does not match the pricing option ID.
DISCOUNT_NOT_AVAILABLE	The specified discounted option is not available at the current time.
DOWNLOAD_PERIOD_IS_NOT_NULL	The download period in pricing option is not null when the viewing method is streaming, thus it cannot be ordered for streaming (only for download).
DOWNLOAD_PERIOD_IS_NULL	Download period in pricing option can't be null for viewing method converged or download.
EXCESS_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is higher than the product price.
EXISTING_VOD_GIFT	The account has a VOD gift in New/Viewed status for the specified video.
EXISTS_IN_USER_SEASONPASSES	The episode is part of a Season Pass that was already purchased by the user. The episode can be watched without an order by calling call GetTvShowEpisodePlayingInfo API with video_external_id (the value is returned in args0 of the response) and tv_show_ticket_id (the value is returned in args1 of the response).
EXISTS_IN_USER_SUBSCRIPTIONS	The asset also exists in one of the users SVOD subscriptions. GetAggregatedVideo gets all the VOD and SVOD offers that include this asset. Then the client should order the desired movie within the relevant SVOD service. This way the user will not be debited.
EXPIRED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference has expired.
EXPIRED_VOD_GIFT	The specified video gift has already expired.
FAILED_TO_DEBIT_IN_EXTERNAL_SYSTEM	The order could not be executed.
GENERAL_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) A general error which is not related to any of the other error codes.

Name	Description
GIFT_CAN_NOT_BE_REPLACED	Trying to replace a gift that cannot be replaced.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INSUFFICIENT_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is lower than the price of the purchased item.
INSUFFICIENT_CREDIT	There is insufficient credit for subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_CARD_ID	The video cannot be ordered with the supplied video card.
INVALID_CARD_TICKET	The video card ticket supplied is invalid (inactive or for a different category).
INVALID_CARD_TICKET_ID	The video card ticket supplied does not exist.
INVALID_DISCOUNT_OPTION_ID	The specified discounted option does not exist for the video.
INVALID_PRICE	The price supplied does not match the actual price. The validation is on the final price, after discount calculation.
INVALID_PRICING_ID	There is no pricing in the system with the given pricing ID. Or the pricing option is not supported for the requested VOD e.g ordering VOD card with eST pricing option. If vod_card_id is passed and vod_gift_id is passed an exception will be thrown.
INVALID_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference is invalid (rejected either by the integration micro service or by the store).
INVALID_TEMPLATE	The page with the given ID has an invalid template.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.
INVALID_VIDEO_ID	There is no video in the system with the given ID or the video is not published.
INVALID_VOD_GIFT_ID	The specified gift was not found for the household.
IN_APP_PURCHASE_DISCOUNT_NOT_SUPPORTED	Discounts are not supported for In-App purchases.
MISSING_PRICING_OPTION	Pricing option ID is mandatory when discount type is percentage.

Name	Description
NO_MATCHING_DELIVERY_MODEL_FOR_VIEWING_METHOD	The VOD card/gift delivery mode (if specified) does not match the requested viewing method.
NO_MATCHING_MEDIA_FILE	There is no matching distributed media file for the requested model, definition and viewing method.
NO_MATCHING_PRICING_ID_FOR_VIEWING_METHOD	The pricing option passed does not match the requested viewing method. In case viewing method is streaming and the download period is not null, or viewing method is download and the download period is null a DOWNLOAD_PERIOD_IS_NOT_NULL exception will be thrown. This is done to not break backward compatibility. In future versions this exception will be dropped and only DOWNLOAD_PERIOD_IS_NOT_NULL will be used.
NO_PROFILES_FOR_TERMINAL_MODEL	The provided terminal model has no profiles.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.
REDEEMED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference was already used.
REPLACED_GIFT_IS_MORE_EXPENSIVE	The replaced gift is more expensive than the original gift.
SYNTAX_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) At least one of the input parameters is invalid or a syntax error. More information is detailed in the "inputErrors" of the error body.
TICKET_EXISTS	The asset/episode was already purchased by the user in a VOD or VOD Package or purchased and watched by the user in a Season Pass offer. GetVideoTicketList returns the relevant asset ticket.
UNSUPPORTED_OPERATION	In-App purchases are not supported.
VIEWING_METHOD_NOT_SUPPORTED_BY_BUSINESS_MODEL	Viewing method not supported by the VOD Card business model.

Comments

- This API accepts the following permutations:
 - video_id | external_video_id + subscription + [pin_code] + [viewing_method] + [payment_mode]
 - video_id | external_video_id + pricing_id + [price] + [pin_code] + [viewing_method] + [payment_mode]
 - video_id | external_video_id + discounted_option_id + [pricing_id] + [price] + [pin_code] + [viewing_method] + [payment_mode]
 - video_id | external_video_id + video_card_ticket_id + [pin_code] + [viewing_method] + [payment_mode]
 - video_id | external_video_id + vod_gift_id + [pin_code] + [viewing_method] + [pricing_id] | [discounted_option_id]
- When replacing Video gift, the video_id/external_video_id should be different from the original video id and pricing_id/discounted_pricing_id should be delivered.

Examples

```
1 OrderVideo?video_id=123&pricing_id=1
```

```
1 OrderVideo?external_video_id=abc&pricing_id=1
```

See Also

- [Video](#)
- [VideoTicket](#)
- [GetVideoTicketList](#)
- [Order](#)

OrderVideoCard

since 5.3

An action used to order a video card.

Parameters

Name	Type	Description
video_card_id	Number	The ID of the video card.
video_card_external_id	String	The external ID of the video card.
price	Number	The price that was shown to the subscriber; used to verify that the subscriber will pay as shown. (Number Type: Double).
pin_code	String	Purchase PIN of the member performing the action. The default is the logged-in member.
terminal_identifier	String	The household terminal identifier (generally it should be the MAC address ID).

Name	Type	Description
business_model	String	Business model of the VOD card ticket. Possible values: rental or eST. Default value: rental.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INSUFFICIENT_CREDIT	There is insufficient credit for subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_CARD	The video card supplied is inactive.
INVALID_CARD_ID	The video card supplied does not exist.
INVALID_PRICE	The price supplied does not match the actual price.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.
UNSUPPORTED_BUSINESS_MODEL	Specified VOD card does not support this business model.
VODCARD_IS_NOT_ACTIVE	Specified VOD card is not active.

Comments

- Either video_card_external_id or video_card_id should be supplied

Examples

```
1 OrderVideoCard?video_card_id=123
```

See Also

- [VideoCard](#)
- [VideoCardTicket](#)
- [GetVideoCardTicketList](#)

OrderVideoGift

since 7.0

An action used to order a video gift.

Parameters

Name	Type	Description
external_video_id	String	The external ID of the movie page.
receiver_hh_external_id	String	The external ID of household that receives the gift (in case the household phone number is used, otherwise mandatory).
receiver_hh_phone_number	String	The phone number of the household that receives the gift (in case the household external ID is used, otherwise mandatory).
pricing_id	Number	The ID of the pricing option the buyer selected. (Note: eST options may not be selected when ordering a Video Gift). (Mandatory if discounted_option_id is not specified or if discounted_option_id specifies a percentage discount.)
discounted_option_id	Number	The ID of a discounted option ID (if pricing_id is not passed - it is mandatory).
price	Number	The price that was shown to the customer; used to verify that the customer will pay as shown. The validation will be on the final price after discount calculation (if a discount exists). If a percentage discount is applied, the price is calculated by reducing the original price by the discount percentage. (Number Type: Double).
pin_code	String	The purchase PIN of the Member performing the action. The default is the logged-in member.
payment_mode	String	The order payment mode. Possible values: <ul style="list-style-type: none">• "normal"• "prepaid"• "internal_prepaid" The default is "normal".
buyer_name	String	The buyer name. If not provided the buyer main member first name + last name shall be used.
buyer_message	String	Free text to appear for the receiver to identify the buyer.
receiver_name	String	The receiver name. if not provided the receiver main member first name + last name shall be used.

Name	Type	Description
can_be_replaced	Boolean	A boolean that indicates whether the given gift is replaceable (true), or not (false). The default is true.
terminal_identifier	String	The household terminal identifier (generally it should be the MAC address ID).

Returns

Name	Description
Result	

Throws

Name	Description
DISCOUNT_NOT_AVAILABLE	The specified discounted option is not available at the current time.
GIFT_IS_NOT_IN_PARENTAL_CONTROL_LEVEL	The requested gift is not in the allowed parental control level.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INSUFFICIENT_CREDIT	There is insufficient credit for subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_DISCOUNT_OPTION_ID	The specified discounted option does not exist for the video.
INVALID_HOUSEHOLD_STATE	The receiver household is invalid to receive a gift.
INVALID_PAGE_ID	Gift validity period exceeds page availability.
INVALID_PRICE	The price supplied does not match the actual price.
INVALID_PRICING_ID	There is no pricing in the system with the given pricing id. Or if passed pricing_id with an eST option
INVALID_RECEIVER_HOUSEHOLD_DETAILS	The household ID or phone number did not match any of the registered households.
INVALID_TEMPLATE	The page with the given ID has an invalid template.
INVALID_VIDEO_ID	There is no video in the system with the given ID or the video is not published.
MISSING_PRICING_OPTION	Pricing option ID is mandatory when discount type is percentage.

Name	Description
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.

Examples

1	OrderVideoGift?external_video_id=123&receiver_hh_external_id=10000&pricing_id=15
1	OrderVideoGift? external_video_id=123&receiver_hh_phone_number=98854684&discounted_option_id=2

See Also

- [Video](#)
- [VodGift](#)
- [Order](#)
- [OrderVideo](#)

OrderVideoPackage

since 5.3

Orders a video package and adds all relevant media files for play/download to the ticket. The media files to add to the ticket are selected by 1) selecting only media files with content definition levels equal to or lower than that specified for the page, and then 2) selecting only media files that match the account's devices' matching profiles or according to device specified by the model_external_id (if provided) 3) If Multi Definition Playback is not enabled in the Configuration, for each of the account's devices, a ticket is created for the remaining matching media file of the highest definition. If Multi Definition Playback is enabled, for each of the account's devices, tickets are created for all remaining matching media files, with their priorities noted. Multi Definition Playback is supported only when Converged Definition in the Settings is enabled. If ordering a video package that was purchased as an In-App purchase through an App store, the store_id together with the purchase reference or store_product_reference provided by the App store must be specified. RiGHTv validates the purchase_reference with the App store and sends the confirmation to the application. The application then completes the purchase with the App store.

Parameters

Name	Type	Description
package_id	Number	The ID of the package page.
package_external_id	String	The external ID of the package page.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.
price	Number	The price shown to the subscriber; used to verify that the subscriber will pay as shown. The validation will be on the final price after discount calculation (if a discount exists). (Number Type: Double).

Name	Type	Description
payment_mode	String	The order payment mode. Possible values: <ul style="list-style-type: none">• "normal"• "prepaid"• "internal_prepaid" The default is "normal".
viewing_method	String	The viewing method. Possible values: <ul style="list-style-type: none">• "streaming"• "download"• "converged"• "est" The default is "converged" from 7.3 (streaming is the default in previous versions).
terminal_identifier	String	An identifier of the device used, e.g. its MAC address. No validation is performed on this parameter.
model_external_id	String	The external ID of the device model. If there is no matching media file for the device, an exception is thrown.
validateExistingRights	Boolean	Indicates whether to check if the account has either an open ticket or a VOD gift in New/Viewed status for all movies in the package (true). Default is false.
store_id	String	The ID of the app store. (Mandatory if purchase_reference is specified)
purchase_reference	String	The reference to a purchase made for the specified video at the App store. Use the POST method to pass this parameter. (Mandatory if store_id is specified)
store_product_reference	String	The reference assigned to the product purchased at a particular App store.

Returns

Name	Description
Result	

Throws

Name	Description
CP_HOUSEHOLD_NO_TERMINALS	The household has no terminals defined in the CAS.
DOWNLOAD_PERIOD_IS_NULL	Download period in pricing option can't be null for viewing method converged or download.

Name	Description
EXCESS_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is higher than the product price.
EXISTING_RIGHTS_FOR_ALL_MOVIES	The account already has an open ticket or VOD gift for every movie in the package.
EXPIRED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference has expired.
FAILED_TO_DEBIT_IN_EXTERNAL_SYSTEM	The order could not be executed.
GENERAL_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) A general error which is not related to any of the other error codes.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INACTIVE_HOUSEHOLD	The household is no longer active (Inactive or Suspended).
INSUFFICIENT_BALANCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The user balance is lower than the price of the purchased item.
INSUFFICIENT_CREDIT	There is insufficient credit for the subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_PACKAGE_ID	There is no video package in the system with the given ID.
INVALID_PRICE	The price supplied does not match the actual price. The validation is on the final price, after discount calculation.
INVALID_PRICING_ID	There is no pricing in the system with the given ID/ The pricing ID is not supported for this action.
INVALID_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference is invalid (rejected either by the integration micro service or by the store).
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.
IN_APP_PURCHASE_DISCOUNT_NOT_SUPPORTED	Discounts are not supported for In-App purchases.
NO_MATCHING_MEDIA_FILE	No matching distributed media file for the requested model, definition and viewing method.
NO_PROFILES_FOR_TERMINAL_MODEL	The provided terminal model has no profiles.

Name	Description
PACKAGE_DELIVERY_MODE_DIF_F_FROM_VIEWING_METHOD	The passed viewing method does not match the video package delivery mode.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household. Relevant for non operator subscribers.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.
REDEEMED_PURCHASE_REFERENCE	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) The provided purchase reference was already used.
SYNTAX_ERROR	(Micro-service error code. Received as property DETAILED_ERROR_CODE.) At least one of the input parameters is invalid or a syntax error. More information is detailed in the "inputErrors" of the error body.
TICKET_EXISTS	There is an open ticket for this package for the logged-in household.
UNSUPPORTED_OPERATION	In-App purchases are not supported.

Comments

- This API accepts the following permutations:
 - package_id | package_external_id

Examples

```
1 OrderVideoPackage?package_id=227&viewing_method=streaming
1 OrderVideoPackage?package_id=230&viewing_method=download
1 OrderVideoPackage?package_id=227
1 OrderVideoPackage?package_external_id=abc
```

See Also

- [Order](#)
- [VideoPackage](#)

PollMessage (deprecated)

since 7.1

since 7.4

An action used to poll pending messages from the remote terminal.

Parameters

Name	Type	Description
session_id	String	The session ID.

Name	Type	Description
terminal_id	String	The ID of the terminal that is waiting for the message.
mac_address	String	The MAC address of the terminal that is waiting for the message.
serial_number	String	The serial number of the terminal that is waiting for the message.
smart_card_id	String	The smart card ID of the terminal that is waiting for the message.
timeout	Number	Maximum time in milliseconds to wait for the message. Default is 120000 millis.

Returns

Name	Description
String	Message from the remote terminal

Throws

Name	Description
ALREADY_CONNECTED	Another PollMessage is already waiting for a response.
SESSION_NOT_EXIST	The requested session does not exist.

Comments

- For more details on how to use the remote messaging mechanism please read "RiGHTv Remote Client PVR Developers Guide"

Examples

```
1 PollMessage?session_id=1234&terminal_id=5678
```

See Also

- [OpenSession](#)
- [CloseSession](#)
- [GetSessionList](#)
- [SendMessage](#)
- [SendResponse](#)

PreOrderVideo

since 8.0

An action that allows users to pre-order a movie page that is announced but not yet published (usually for a discounted price). (Since 8.0 SR3)

Parameters

Name	Type	Description
external_video_id	String	The external ID of the movie page.
price	Number	The price that was shown to the user; used to verify that the user will pay as shown. (displayed price). (Number Type: Double).
payment_mode	String	The pre-order payment mode. Possible values: <ul style="list-style-type: none">• "normal"• "internal prepaid" The default is "normal".

Returns

Name	Description
Result	

Throws

Name	Description
BILLING_METHOD_INVALID	Prepaid household billing method is invalid for a pre-order request.
EXISTS_IN_USER_SUBSCRIPTIONS	The asset already exists in one of the users SVOD subscriptions.
INVALID_PAGE_STATUS	The movie page status is not ANNOUNCED.
INVALID_PAGE_TYPE	The page status is not MOVIE.
INVALID_PAYMENT_MODE	The payment mode is invalid. Valid values are normal(postpaid) or internal prepaid.
INVALID_PRICE	The price supplied does not match the actual price.
INVALID_VIDEO_ID	There is no video in the system with the given ID.
MISSING_PRICING_OPTION	The pricing option EST_PRE_ORDER is missing.
PAYMENT_METHOD_NOT_FOUND	There is no payment method registered with the payment system for the household.Relevant for non operator subscribers.
PERMISSION_DENIED	The logged-in household does not have permission to access the required content due to security restrictions.
PRE_ORDER_EXISTS	Pre order already exists.
TICKET_EXISTS	The asset was already purchased by the user in a VOD or VOD Package offer. GetVideoTicketList returns the asset relevant ticket.

Comments

- This API accepts the following permutations:
 - external_video_id + [price] + [payment_mode]
 - external_video_id + [payment_mode]
 - external_video_id + [price]
 - external_video_id
- Multi Definition Playback is not supported for pre-order eST.

Examples

```
1 PreOrderVideo?external_video_id=123&price=1
```

See Also

- [PreOrder](#)

RechargeInternalPrepaid

since 6.0

An action used to recharge the internal prepaid account of the household.

Parameters

Name	Type	Description
option_id	Number	The ID of the option.
price	Number	The pricing amount to recharge. For validation purpose only. (Number Type: Double).
bonus	Number	The amount of bonus to recharge. For validation purpose only. (Number Type: Double).
name	String	The name of the recharge order that will be reflected in the Order list. If not provided, the order name is "price + bonus". If provided - the order name is "name price + bonus".
no_billing	Boolean	A boolean that indicates whether to bill the subscribers account. When set to 'true', recharge is done outside of RiGHTv by an external system. RiGHTv is only informed after the successful recharge is done. RiGHTv does not communicate the recharge to any billing system, postpaid or prepaid. When set to 'false' the subscriber is billed for the recharge using the subscriber's billing method (prepaid/postpaid). Default value: false.

Returns

Name	Description
Result	

Throws

Name	Description
INSUFFICIENT_CREDIT	There is insufficient credit for subscriber to make the purchase due to reaching the spending limit or exceeding the prepaid amount.
INVALID_OPTION_ID	The option ID is invalid.
INVALID_PRICE	The price or bonus supplied does not match the actual price or bonus.
NON_OPERATOR_SUBSCRIBER_NOT_ALLOWED	The operation is invalid for non operator subscriber.

Examples

```
1 | RechargeInternalPrepaid?option_id=12&price=10&bonus=2
```

See Also

- [GetOrderList](#)

RegisterBouquet*since 6.0*

An action used to register a bouquet to a household.

Parameters

Name	Type	Description
bouquet_id	Number	The bouquet ID to register {precedence 1}.
bouquet_external_id	Number	The external ID to register {precedence 2}.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).

Name	Description
INVALID_BOUQUET_ID	The bouquet id is invalid.

Comments

- This API accepts the following permutations:
 - bouquet_id | bouquet_external_id

Examples

1 RegisterBouquet?bouquet_id=12345

1 RegisterBouquet?bouquet_external_id=abc

See Also

- [Bouquet](#)
- [UnregisterBouquet](#)

RegisterExternalToken

since 7.4

External system API (Messaging Dispatcher). This API is used for registering an external application for receiving message notifications. GCM (Google Cloud Messaging) and APNS (Apple Push Notification Service) are supported. Every time the user logs into the application this API should be used in case the application wants to receive message notifications.

Parameters

Name	Type	Description
gcm_reg_id	String	The unique GCM registration ID that was received from the GCM service.
apns_device_token	String	The unique device token that was received from APNS. It contains information that enables APNS to locate the device on which the client application is installed. A device token is binary array of 32 bytes length. It must be in hexadecimal representation.
from_terminal_id	String	The ID of the terminal from which to register GCM token.
from_mac_address	String	The MAC address of the terminal from which to register GCM token.
from_serial_number	String	The serial number of the terminal from which to register GCM token.
from_smart_card_id	String	The smart card ID of the terminal from which to register GCM token.
apns_app_id	String	The Apple Application ID to which apns_device_token is connected.

Name	Type	Description
gcm_app_id	String	The Google Application ID to which it is connected.

Returns

Name	Description
GcmToken	or {ApnsToken}

Throws

Name	Description
ENTITY_NOT_FOUND	Registration ID is not exists.
HOUSEHOLD_NOT_ACTIVE	Household is currently not active (deleted or suspended).
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
MISSING_PARAMETER	One of mandatory parameter is missing.
NO_HOUSEHOLD_TERMINALS	Household has no terminals.

Comments

- RIGHTV cookies contains household_id and terminal_id identifiers. If RiGHTv Login was done only with household (and does not contain terminal information) then one of 4 'from' parameters can be used to specify terminal. Terminal 'from' parameters are relevant in case that login was done only with household.
- At least one of the 'gcm_reg_id' or 'apns_device_token' parameter is mandatory.

Examples

```
1 RegisterExternalToken?gcm_reg_id=12345
```

```
1 RegisterExternalToken?apns_device_token=0xAABBCCDD
```

See Also

- [GcmToken](#)
- [ApnsToken](#)
- [UnregisterExternalToken](#)

RegisterHousehold**since 7.3**

Registers an account (household). Relevant for online customers only. Operator subscriber registration is done via the CRM API or the Admin GUI application.

Parameters

Name	Type	Description
username	String	The username of the new household (valid email).

Name	Type	Description
password	String	The password of the new household.
pin_code	String	The PIN code of the new household. (if not supplied, 0000 will be used)
first_name	String	The first name of the main member.
last_name	String	The last name of the main member.
address_country	String	The country of the new household.
address_state	String	The state of the new household.
address_city	String	The city of the new household.
address_street	String	The street of the new household.
address_house_number	String	The house number of the new household.
address_apartment_number	String	The apartment number of the new household.
address_postal_code	Number	The postal code of the new household.
phone_number	String	The phone number of the new household.
spending_limit	Number	The spending limit of the new household.
language_code	String	The language code of the new household. If not inserted, the system default will be used.
ef_names	String	Each of the name parameters is a delimited list of household extra field names (default delimiter is semicolon: ";").
ef_values	String	Each of the name parameters is a delimited list of household extra field values (default delimiter is semicolon: ";").
member_ef_names	String	Each of the name parameters is a delimited list of member extra field names (default delimiter is semicolon: ";").
member_ef_values	String	Each of the name parameters is a delimited list of member extra field values (default delimiter is semicolon: ";").

Name	Type	Description
captcha_key	String	CAPTCHA key received by GetCaptchaKey method. Mandatory if skipCaptchaValidation is false in terminal category of specified terminal model. Mandatory if terminal model is not specified.
captcha_challenge	String	CAPTCHA challenge text filled by user. Mandatory if skipCaptchaValidation is false in terminal category of specified terminal model. Mandatory if terminal model is not specified.
model_external_id	String	The device model's external ID. Used to determine whether a CAPTCHA is required and whether the account email needs to be confirmed.

Returns

Name	Description
Result	

Throws

Name	Description
CAPTCHA_FAILED	Passed CAPTCHA is invalid.
CAPTCHA_KEY_NOT_FOUND	Passed CAPTCHA key could not be found.
ILLEGAL_PASSWORD	The password is illegal according to the RiGHTv password policy.
INVALID_EMAIL	Specified email(username) structure doesn't meet RiGHTv policy.
INVALID_PARAMETER	One of the passed parameters is invalid.
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
INVALID_PIN_CODE	The PIN code is illegal according to the RiGHTv pin policy.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.
MISSING_PARAMETER	One of the mandatory parameters is missing.
NON_OPERATOR_SUBSCRIBERS_NOT_SUPPORTED	Non-operator subscribers are not supported since there is no integration with an external payment system.
USERNAME_ALREADY_EXISTS	Specified username already exists in RiGHTv.

Examples

1	RegisterHousehold?username=test@test.com&password=tstPassword&pin_code=1234&first_name=tstFirstName&last_name=tstLastName&ef_names=name1;name2&ef_values=value1;value2&member_ef_names=name1;name2&member_ef_values=value1;value2&captcha_key=12345&captcha_challenge=1234
---	--

RegisterTerminal

since 5.3

An action used to register a terminal (device) with the logged-in household.

Parameters

Name	Type	Description
mac_address	String	The MAC address of the terminal to register.
serial_number	String	The serial number of the terminal to register.
smart_card_id	String	The smart card ID of the terminal to register.
manufacturer_id	String	The manufacturer ID of the terminal to register.
terminal_model_id	String	The terminal model ID of the terminal to register.
operator_id	String	The operator ID of the terminal to register
ef_names	String	Each of the name parameters is a delimited list of extra field names (default delimiter is semicolon: ";").
ef_values	String	Each of the name parameters is a delimited list of extra field values (default delimiter is semicolon: ";").
name	String	Friendly name of the device, as provided by the end-user. In case the end-user doesn't provide it, the client can insert here the device name, as retrieved from the device.
model_external_id	String	The external id of RiGHTv's terminal model. If not passed the default model will be used. In order to identify the model, client may use the GetTerminalModelList

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_TERMINAL_MODEL	The supplied model was not found
MAX_TERMINALS_EXCEEDED	The maximum number of terminals per household exceeded. In the system we have four max terminals parameters: Global Household - one for all households. Specific Household - for specific household. Global Category - for each category for all households. Specific Category - for each category for specific household. The applicable max terminal per household is determined by taking the minimum value between the global and the specific parameters.
MAX_TERMINAL_REMOVALS_EXCEEDED	The maximum allowed number of terminal unregistrations for current period has exceeded. Device in the set-top box category will be ignored by the removal counter and device removal limit validation if "Ignore Set-top box removal in the device removal limit" flag is set to ON. Number of days left till next terminal removal counters reset will be returned as a value of exception Property with key POSSIBLE_REMOVAL_IN_DAYS.
TERMINAL_CANNOT_BE_REPLACED	exception is thrown on device registration when the user limit is reached and none of the household devices can be removed. A future date for device replacement (in milliseconds) will be returned as a value of exception Property with the key PARAMETER_FUTURE_DATE. The client can initiate a new registration date on this future date. When allowSelfReplace flag is enabled, and the number of user devices limit is reached, RiGHTv checks with the CAS/DRM if the device can be replaced. If the device cannot be replaced (because the device license has not expired) then throw TERMINAL_CANNOT_BE_REPLACED exception.
TERMINAL_EXISTS	A terminal with the same mac-address, or same serial-number or same smart-card, already registered in the system. Notice that if the exact same terminal already exist in the current household, operation will be ignored with success result.
TERMINAL_NAME_ALREADY_EXISTS	Terminal with the specified name already registered with the same household.
TERMINAL_REPLACEMENT_FAILED_TOO_MANY_MATCHES	When ALLOW_REPLACE_BY_TERMINAL_MATCHING flag is enabled on STB terminal category and more than one matching terminal (mac-address, or same serial-number or same smart-card of STB category) were found for the household.

Comments

- At least one of the following parameters must not be empty: mac_address, serial_number, smart_card_id in the request.

Examples

```
1 RegisterTerminal?
mac_address=00:11:22:33:44:55&ef_names=name1;name2&ef_values=value1;value2
```

See Also

- [UpdateTerminalIp](#)
- [Terminal](#)

RemoveFromWishList (deprecated)

since 7.0

Deprecated since 8.5. Use the Wishlist functionality provided in The COMPASS Content Discovery Platform.

An action used to remove an item from the wish list for a specific household or member.

Parameters

Name	Type	Description
item_external_id	String	The external ID of the item (video or package) to be removed from the wish list.
scope	String	Possible values are: household or member. Wish list can be updated on the household or the member level. The default is household.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_PAGE_ID	The supplied page ID is invalid.
WISH_NOT_EXISTS	The supplied wish is not in the household/member list.

Examples

```
1 RemoveFromWishList?item_external_id=123
```

```
1 RemoveFromWishList?item_external_id=123&scope=member
```

```
1 RemoveFromWishList?item_external_id=123&scope=household
```

See Also

- [VideoWish](#)
- [AddToWishList](#)
- [GetWishList](#)

RemovePackageSubscription

since 5.3

Removes a package subscription from the specified account. If the account is not subscribed to the specified package, it will be ignored and the operation will succeed. For NPVR 2.0, when a package subscription is removed from an account, past and scheduled recordings belonging to channels in the removed package are deleted unless the account has an additional subscription to the channel through another channel package. Recordings are removed from the list of recordings displayed in the client, but can be revived during the npvrDeletionDelay period. After this period recordings are permanently deleted and cannot be revived. For NPVR 1.0 future recordings of channels in the removed package are deleted unless the account has an additional subscription to the channel.

Parameters

Name	Type	Description
package_id	Number	The ID of the package to remove.
external_id	String	The external ID of the package to remove.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_PACKAGE_ID	An invalid package ID (package_id) was supplied.

Comments

- This API accepts the following permutations:
 - package_id | external_id

Examples

```
1 RemovePackageSubscription?package_id=123
```

```
1 RemovePackageSubscription?external_id=abc
```

See Also

- [Package](#)
- [PackageChannel](#)
- [PackageSubscription](#)
- [AddPackageSubscription](#)
- [GetPackageList](#)
- [GetHouseholdPackageList](#)
- [GetPackageChannelList](#)

RemoveServicePlanSubscription*since 8.1*

This action unsubscribes the specified account from the specified service plan. Access to the service plan is ended immediately. (Since 8.1 SR1). In the current version, for NPVR 2.0, recordings of channels to which the account is no longer subscribed are not deleted.

Parameters

Name	Type	Description
external_id	String	The Service Plan external ID.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	The specified account is not active.
HOUSEHOLD_NOT_FOUND	The specified account could not be found.
HOUSEHOLD_PENDING_INACTIVATION	The account is pending inactivation (household planned inactivation date is in the past).
SERVICE_PLAN_NOT_FOUND	The specified service plan could not be found.
SERVICE_PLAN_SUBSCRIPTION_NOT_FOUND	The account is not subscribed to the specified service plan.

Examples

```
1 RemoveServicePlanSubscription?external_id=123
```

See Also

- [ServicePlan](#)
- [GetServicePlan](#)
- [AddServicePlanSubscription](#)

RemoveServiceSubscription*since 5.3*

Removes a service subscription from the logged-in account. If the account is not subscribed to the service, it will be ignored and no error will be thrown. When removing an NPVR Service,

recordings are deleted. For NPVR 2.0, deleted recordings are removed from the list of recordings displayed in the client, but can be revived during the npvrDeletionDelay period. After this period recordings are permanently deleted. For NPVR 1.0 recordings are permanently deleted.

Parameters

Name	Type	Description
service_id	Number	The ID of the service to remove.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.
external_service_id	String	The external ID of the service to remove.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_SERVICE_ID	The supplied service ID is invalid.

Comments

- This API accepts the following permutations:
 - service_id | external_service_id

Examples

```
1 RemoveServiceSubscription?service_id=123
```

```
1 RemoveServiceSubscription?external_service_id=123
```

See Also

- [Service](#)
- [ServiceSubscription](#)
- [AddServiceSubscription](#)
- [GetServiceList](#)
- [GetHouseholdServiceList](#)

ResendConfirmationEmail

since 7.3

An action used to send the confirmation email again to a non-operator household.

Parameters

Name	Type	Description
username	String	username (household's email).
captcha_key	String	CAPTCHA key received by GetCaptchaKey method. Mandatory if skipCaptchaValidation is false in terminal category of specified terminal model. Mandatory if terminal model is not specified.
captcha_challenge	String	CAPTCHA challenge text filled by user. Mandatory if skipCaptchaValidation is false in terminal category of specified terminal model. Mandatory if terminal model is not specified.
model_external_id	String	The device model's external ID. Used to determine whether a CAPTCHA is required.

Returns

Name	Description
Result	

Throws

Name	Description
CAPTCHA_FAILED	Passed CAPTCHA is invalid.
CAPTCHA_KEY_NOT_FOUND	Passed CAPTCHA key could not be found.
HOUSEHOLD_NOT_FOUND	Non-operator household with specified email was not found in RiGHTv.
HOUSEHOLD_STATUS_NOT_VALID	The household is not in "inactive" status
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.

Examples

1	ResendConfirmationEmail? username=test@test.com&captcha_key=12345&captcha_challenge=1234
---	---

ResetMemberPin

since 7.3

An action used to reset the household's purchase PIN code. The action sends an email to the household with a link to a page which generates a 4 digit purchase PIN code.

Returns

Name	Description
Result	

Throws

Name	Description
EMAIL_WAS_NOT_SET	Regular subscriber without email set in RiGHTv cannot perform this operation

Examples

1	ResetMemberPin
---	----------------

ResetParentalPin*since 7.3*

An action used to reset household's parental PIN code. The action sends an email to the household with a link to a page which generates a 4 digit parental P.

Returns

Name	Description
Result	

Throws

Name	Description
EMAIL_WAS_NOT_SET	Regular subscriber without email set in RiGHTv cannot perform this operation

Examples

1	ResetMemberPin
---	----------------

ResetPassword*since 7.3*

An action used to reset the password. Relevant for online customers only.

Parameters

Name	Type	Description
username	String	The username of the household for whose password will be reset.
captcha_key	String	CAPTCHA key received by GetCaptchaKey method. Mandatory if skipCaptchaValidation is false in terminal category of specified terminal model. Mandatory if terminal model is not specified.

Name	Type	Description
captcha_challenge	String	CAPTCHA challenge text filled by user. Mandatory if skipCaptchaValidation is false in terminal category of specified terminal model. Mandatory if terminal model is not specified.
model_external_id	String	The device model's external ID. Used to determine whether a CAPTCHA is required.

Returns

Name	Description
Result	

Throws

Name	Description
CAPTCHA_FAILED	Passed CAPTCHA is invalid.
CAPTCHA_KEY_NOT_FOUND	Passed CAPTCHA key could not be found.
INVALID_EMAIL	The email structure is invalid.
INVALID_TERMINAL_MODEL	The passed terminal model does not exist.
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households.

Examples

```
1 | ResetPassword?username=test@test.com&captcha_key=12345&captcha_challenge=1234
```

ReuseMsgToken

since 7.4

External system API (Messaging Dispatcher). A device which already has an open connection to ConnectionManager (e.g. the token was received from COMPASS) and wants to receive messages from the RiGHTv domain to the same connection should use this API.

Parameters

Name	Type	Description
token	String	The existing token to associate the device with.
from_terminal_id	String	The ID of the terminal(s) from which to reuse token.
from_mac_address	String	The MAC address of the terminal from which to reuse token.
from_serial_number	String	The serial number of the terminal from which to reuse token.

Name	Type	Description
from_smart_card_id	String	The smart card ID of the terminal from which to reuse token.

Returns

Name	Description
AccessToken	

Throws

Name	Description
ENTITY_NOT_FOUND	Token ID does not exists.
HOUSEHOLD_NOT_ACTIVE	Household is currently not active (deleted or suspended).
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
MISSING_PARAMETER	One of the mandatory parameters is missing.
NO_HOUSEHOLD_TERMINALS	Household has no terminals.

Comments

- RIGHtv cookies contains household_id and terminal_id identifiers. If RIGHtv login was done only with household (and does not contain terminal information) then one of 4 'from' parameters can be used to specify terminal. Terminal 'from' parameters are relevant if login was done only with household.

Examples

```
1 ReuseMsgToken?token=1
```

See Also

- [GetMsgToken](#)

SendMessage (deprecated)*since 7.1*

since 7.4

An action used to send a message to the remote terminal.

Parameters

Name	Type	Description
session_id	String	The ID of the session to which to send the message.
terminal_id	String	The ID of the terminal to which to send the message.
mac_address	String	The MAC address of the terminal to which to send the message.

Name	Type	Description
serial_number	String	The serial number of the terminal to which to send the message.
smart_card_id	String	The smart card ID of the terminal to which to send the message.
timeout	Number	Maximum time in milliseconds to wait for the response. Default is 120000 millis.
message	String	The message text. Not relevant when using POST. the message text should be in the POST body.

Returns

Name	Description
String	Response from the remote terminal

Throws

Name	Description
ALREADY_CONNECTED	Another SendMessage is already waiting for a response.
NOT_CONNECTED	A requested terminal is not connected to the requested session.
SESSION_NOT_EXIST	The requested session does not exist.

Comments

- At least one of the terminal identifiers is mandatory.
- The message should be sent in the body of the **POST** request.
- For more details on how to use the remote messaging mechanism please read "RiGHTv Remote Client PVR Developers Guide".

Examples

```
1 | SendMessage?session_id=1234&message=message_text
```

See Also

- [OpenSession](#)
- [CloseSession](#)
- [GetSessionList](#)
- [PollMessage](#)
- [SendResponse](#)

SendMsg

since 7.4

Sends a message through the VO OMP (Online Messaging Platform) to other devices (terminals) in the RiGHTv domain. For more information see the RiGHTv OMP Developers Guide.

Parameters

Name	Type	Description
to_terminal_id	String	The ID of the terminal to which to send the message.
to_address	String	The messaging address to which to send the message. (E.g. hh1.t10000@rightv). Can be used to reply to a message received. See 'fromAddress' in Message.
from_terminal_id	String	The ID of the terminal from which the message was sent.
from_mac_address	String	The MAC address of the terminal from which message is sending.
from_serial_number	String	The serial number of the terminal from which the message was sent.
from_smart_card_id	String	The smart card ID of the terminal from which the message was sent.
time_to_live	Number	Maximum time in seconds that the message will be stored in the database (relevant only if offline=true). Default is 86400 seconds (1 day).
offline	Number	True if message must be persisted in database. Default false.
message	String	The message text. Not relevant when using POST. The message text should be in the POST body.

Returns

Name	Description
Msg	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is currently not active (deleted or suspended).
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
MAX_TIME_TO_LIVE_EXCEEDED	time_to_live parameter is greater than maximum (1 year). Default is 1 day.
MISSING_PARAMETER	One of the mandatory parameters is missing.
NO_HOUSEHOLD_TERMINALS	Household has no terminals.

Comments

- RIGHTV cookies contains household_id and terminal_id identifiers. If RIGHTV Login was done only with household (and does not contain terminal information) then one of 4 'from' parameters can be used to specify terminal. Terminal 'from' parameters are relevant in case that Login was done only with household.
- At the same time only one of 'to_terminal_id' or 'address' parameters can be used.
- 'to_terminal_id' parameter can receive multiple values.
- 'to_address' parameter can receive only a single value.
- At least one of the 'to_terminal_id' or 'to_address' parameter is mandatory.
- The large message must be sent in the body of the **POST** request in URL encoded form with content-type: 'application/x-www-form-urlencoded'

Examples

```
1 SendMsg?message=hi&to_terminal_id=1
1 SendMsg?message=hi&to_terminal_id=1&to_terminal_id=2 (specify multiple terminals)
1 SendMsg?message=hi&to_address=hh1.t10000@rightv
1 SendMsg?message=hi&to_address=hh1.t10000@rightv&from_terminal_id=1 (specify to_address and from which terminal message is sent)
```

See Also

- [GetMsgToken](#)

SendResponse (deprecated)

since 7.1

since 7.4

An action used to send response message to the remote terminal.

Parameters

Name	Type	Description
session_id	String	The session ID.
terminal_id	String	The ID of the terminal from which the response is sent.
mac_address	String	The MAC address of the terminal from which the response is sent.
serial_number	String	The serial number of the terminal from which the response is sent.
smart_card_id	String	The smart card ID of the terminal from which the response is sent.
message	String	The message text. Not relevant when using POST. the message text should be in the POST body.

Returns

Name	Description
Result	

Throws

Name	Description
NOT_CONNECTED	A requested terminal is not connected to the requested session.
SESSION_NOT_EXIST	The requested session does not exist.

Comments

- At least one of the terminal identifiers is mandatory.
- The message should be sent in the body of the **POST** request.
- For more details on how to use the remote messaging mechanism please read "RiGHTv Remote Client PVR Developers Guide".

Examples

```
1 | SendMessage?session_id=1234&terminal_id=6789&message=message_text
```

See Also

- [OpenSession](#)
- [CloseSession](#)
- [GetSessionList](#)
- [SendMessage](#)
- [PollMessage](#)

SetClientData*since 5.3*

An action used to set client information in the server.

Parameters

Name	Type	Description
client_code	String	The unique client identifier (e.g. MAC address).
key	String	The "key" of the "value" you want to set.
value	String	The value you want to pair with the given "key". When not supplied the value is set to an empty string.

Returns

Name	Description
Result	

Comments

- Consider sending this request as POST due to its possible length.

Examples

```
1 SetClientData?client_code=00:11:22:33:44:55&key=BLOCKED_CHANNELS&value=1,2,3,4
```

See Also

- [GetClientData](#)

SetEntitlement*since 5.3*

An action used to set the CA/DRM entitlement for the current household on the given service.

Parameters

Name	Type	Description
program_id	Number	The program ID to which to set the entitlement.
package_id	Number	The channel package ID to which to set the entitlement.
video_id	Number	The video ID to which to set the entitlement.
encoding	String	The encoding format for the supplied video (only in conjunction with video_id).

Returns

Name	Description
Result	

Throws

Name	Description
CHANNEL_NOT_VIEWABLE	Program is not viewable on specified terminal or not viewable on any household terminal.
INVALID_CHANNEL_ID	The channel that connected to the given program ID does not exist.
INVALID_ENCODING	The supplied encoding format is invalid.
INVALID_PROGRAM_ID	The given program ID does not exist.
PLUGIN_NOT_FOUND_ERROR	The content protection plugin was not found.

Comments

- Only one service ID should be passed in the arguments list: program_id, package_id or video_id.

Examples

```
1 SetEntitlement?video_id=123&encoding=MPEG-2
```

SetLanguage*since 5.3*

An action used to set the current language for the logged-in household.

Parameters

Name	Type	Description
new_language_code	String	The language code to be set for the logged-in household.

Returns

Name	Description
Result	

Examples

```
1 SetLanguage?new_language_code=en
```

See Also

- [Language](#)
- [Household](#)

TechnicianLogin*since 5.3*

An action used to log in using technician credentials.

Parameters

Name	Type	Description
username	String	The username of the technician (less than 60 characters).
password	String	The password of the technician (less than 30 characters).

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_CREDENTIALS	An invalid username or password was given. Or user has no technician privileges.

Examples

```
1 TechnicianLogin?username=1&password=1
```

UnregisterBouquet*since 6.0*

An action used to de-register a bouquet from a household.

Parameters

Name	Type	Description
bouquet_id	Number	The bouquet ID to unregister.
bouquet_external_id	Number	The bouquet external ID to unregister.

Returns

Name	Description
Result	

Throws

Name	Description
CHANNEL_PACKAGE_IS_SUBSCRIBED_TO_HOUSEHOLD	Channel package is subscribed to household.
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_BOUQUET_ID	Bouquet ID is invalid.

Comments

- This API accepts the following permutations:
 - bouquet_id | bouquet_external_id

Examples

```
1 UnregisterBouquet?bouquet_id=12345
```

```
1 UnregisterBouquet?bouquet_external_id=abc
```

See Also

- [Bouquet](#)
- [UnregisterBouquet](#)

since 7.4

UnregisterExternalToken

External system API (Messaging Dispatcher). Used for de-registering an external application from receiving message notifications. GCM (Google Cloud Messaging) and APNS (Apple Push Notification Service) are supported.

Parameters

Name	Type	Description
gcm_reg_id	String	The unique GCM registration ID that was received from the GCM service.
apns_device_token	String	The unique device token that was received from APNS. Device token is a binary array of 32 bytes length. It must be in hexadecimal representation.
from_terminal_id	String	The ID of the terminal from which to unregister GCM token.
from_mac_address	String	The MAC address of the terminal from which to unregister GCM token.
from_serial_number	String	The serial number of the terminal from which to unregister GCM token.
from_smart_card_id	String	The smart card ID of the terminal from which to unregister GCM token.

Returns

Name	Description
Result	

Throws

Name	Description
ENTITY_NOT_FOUND	Registration ID is not exists.
HOUSEHOLD_NOT_ACTIVE	Household is currently not active (deleted or suspended).
INVALID_PARAMETER_VALUE	A parameter value specified is invalid.
MISSING_PARAMETER	One of mandatory parameter is missing.
NO_HOUSEHOLD_TERMINALS	Household has no terminals.

Comments

- RiGHTv cookies contains household_id and terminal_id identifiers. If RiGHTv login was done only with household (and does not contain terminal information) then one of 4 'from' parameters can be used to specify the terminal. Terminal 'from' parameters are relevant if login was done only with household.
- At least one of the 'gcm_reg_id' or 'apns_device_token' parameter is mandatory.

Examples

```
1 | UnregisterExternalToken?gcm_reg_id=12345
```

```
1 | UnregisterExternalToken?apns_device_token=0xAABBCCDD
```

See Also

- [GcmToken](#)
- [ApnsToken](#)
- [RegisterExternalToken](#)

UnregisterTerminal

since 7.1

Removes a device from the logged-in account upon request via self-service. The identity of the device to be removed must be specified by its MAC address, serial number or smart card ID.

Note: All stored NPVR recordings for the device's recording format are deleted. If the customer deletes all of its devices, all recordings will be deleted.

Parameters

Name	Type	Description
mac_address	String	The MAC address of the device to be removed.
serial_number	String	The serial number of the device to be removed.
smart_card_id	String	The smart card ID of the device to be removed.

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	The account is not active.
HOUSEHOLD_PENDING_INACTIVATION	The account is pending inactivation (the account's planned inactivation date is in the past).
INVALID_TERMINAL	The specified device to be removed is not found or does not belong to the account.
MAX_TERMINAL_REMOVALS_EXCEEDED	The maximum allowed number of terminal unregistrations for current period has exceeded. Device in the set-top box category will be ignored by the removal counter and device removal limit validation if "Ignore Set-top box removal in the device removal limit" flag is set to ON. Number of days left till next terminal removal counters reset will be returned as a value of exception Property with key POSSIBLE_REMOVAL_IN_DAYS.

Examples

```
1 | UnregisterTerminal?mac_address=00:11:22:33:44:55
```

See Also

- [Terminal](#)
- [RegisterTerminal](#)

UpdateHousehold**since 7.3**

An action used to update a household. Not all fields are updated through this action (e.g username, password etc.) Relevant for online customers only.

Parameters

Name	Type	Description
first_name	String	The first name of main member.
last_name	String	The last name of main member.
address_country	String	The country of new household.
address_state	String	The state of new household.
address_city	String	The city of new household.
address_street	String	The street of new household.
address_house_number	String	The house number of new household.
address_apartment_number	String	The apartment number of new household.
address_postal_code	Number	The postal code of new household.
phone_number	String	The phone number of new household.
language_code	String	The language code of new household. If not inserted, system default will be used.
ef_names	String	Each of the name parameters is a delimited list of extra field names (default delimiter is semicolon: ";").
ef_values	String	Each of the name parameters is a delimited list of extra field values (default delimiter is semicolon: ";").

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_PARAMETER	One of the passed parameters is invalid.
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households
PHONE_NUMBER_ALREADY_EXISTS	Specified phone number already exists.

Examples

```
1 UpdateHousehold?first_name=tstFirstName&last_name=tstLastName&ef_names=name1;name2
  &ef_values=value1;value2
```

UpdateHouseholdSpendingLimit*since 5.3*

An action used to update the logged-in household's spending limit.

Parameters

Name	Type	Description
spending_limit	Number	The new spending limit.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Examples

```
1 UpdateHouseholdSpendingLimit?spending_limit=100
```

See Also

- [Household](#)

UpdateMemberPin*since 5.3*

An action used to update the logged-in member's purchase PIN.

Parameters

Name	Type	Description
new_pin_code	String	The new purchase PIN (4 characters).
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_PIN_CODE	The passed PIN code length is not valid.

Examples

```
1 UpdateMemberPin?new_pin_code=1234
```

See Also

- [Member](#)

UpdateMemberSpendingLimit*since 5.3*

An action used to update the logged-in member's spending limit.

Parameters

Name	Type	Description
spending_limit	Number	The new spending limit.
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
NON_OPERATOR_SUBSCRIBER_NOT_ALLOWED	The operation is invalid for non operator subscriber.

Examples

```
1 UpdateMemberSpendingLimit?spending_limit=100
```

See Also

- [Member](#)

UpdateParentalPin*since 5.3*

An action used to update the logged-in household's parental PIN.

Parameters

Name	Type	Description
parental_pin	String	The new parental PIN (4 characters).
pin_code	String	The purchase PIN of the member performing the action. The default is the logged-in member.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_PIN_CODE	The passed pincode length is not valid.

Examples

```
1 | UpdateParentalPin?parental_pin=1234
```

See Also

- [Household](#)

UpdatePassword

since 7.3

An action used to update household's password. Relevant for online customers only.

Parameters

Name	Type	Description
old_password	String	The current password.
new_password	String	The new password.
new_password_retyped	String	The new password again.

Returns

Name	Description
Result	

Throws

Name	Description
ILLEGAL_PASSWORD	The password is illegal according to the RiGHTv password policy.
INVALID_PASSWORD	The current passed password is invalid.
INVALID_PASSWORD RETYPE	The password was not retyped successfully.

Name	Description
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households

Examples

```
1 UpdatePassword?old_password=1234&new_password=12345&new_password_retyped=12345
```

UpdatePaymentAuthentication*since 6.0*

An action used to update household payment authentication.

Parameters

Name	Type	Description
value	String	The payment authentication value. Possible values: <ul style="list-style-type: none">• "never"• "always"• "normal only"• "internal prepaid only"

Returns

Name	Description
Result	

Throws

Name	Description
NON_OPERATOR_SUBSCRIBER_NOT_ALLOWED	The operation is invalid for non-operator subscribers.

Examples

```
1 UpdatePaymentAuthentication?value=always
```

UpdateRecording*since 9.0*

Updates a recording order for the logged-in account. Relevant for NPVR 2.0 recordings only. Use GetRecordingTicketList to get the recording IDs (otf_recording_id) for the logged-in account. If the account has exceeded its NPVR quota or concurrent recordings limit, it will only be possible to reduce the recording's end_time and lag_duration. Only parameters that are specified are updated. If a parameter is left blank, the existing parameter remains unchanged. One of the following permutations must be provided:

- For EPG based recordings: otf_recording_id and at least one of lag_duration or user_comment.
- For time-based recordings: otf_recording_id and at least one of name or end_time.

Parameters

Name	Type	Description
otf_recording_id	String	The ID of the recording. (Mandatory).

Name	Type	Description
name	String	The updated name for the recording. Relevant for time-based recordings.
end_time	Number	The updated end time for the recording (in milliseconds from 1/1/1970). Relevant only for time-based recordings (both ongoing and future recordings).
lag_duration	Number	The buffer after the scheduled recording time to take minor scheduling shifts into account, in minutes. Cannot be a negative number. If no lag duration is set for the specific recording, then the global lag duration setting is used.
user_comment	String	User comment for the recording. Relevant for NPVR 2.0 only. Not relevant for series recording. Since 9.1 SR1.

Throws

Name	Description
FAILED_TO_UPDATE_RECORDING	Unable to update the recording.
INVALID_CHANNEL	The channel specified in the recording does not exist.
INVALID_RECORDING_ID	The recording specified does not exist.
INVALID_RECORDING_TIMES	The updated recording end time is in the past or the start time specified is after the end time.
MAX_QUOTA_EXCEEDED	The account has used all available quota.
MISSING_PARAMETER	One of the mandatory parameters is missing.
PAST_PROGRAM	The specified program is in the past and outside the availability period. Applies to both time-based and EPG recordings. Relevant from RiGHTv 9.1 SR18.
SERVICE_UNSUBSCRIBED	The logged-in account is not subscribed to an NPVR service.
UNSUPPORTED_OPERATION	NPVR is not set to NPVR 2.0.

UpdateTerminal

since 7.1

An action used to update the logged-in account's device identified by its MAC address/serial number/smart card ID via self-service. Extra fields values can be cleared by providing an empty string, however extra fields cannot be deleted. If an extra field is not provided, its value will remain unchanged.

Parameters

Name	Type	Description
mac_address	String	The MAC address of the device.
serial_number	String	The serial number of the device.
smart_card_id	String	The smart card ID of the device.
name	String	The device name to be updated. At least one of the following parameters is mandatory: name parameter or ef_names.
ef_names	String	A delimited list of extra field names (the delimiter is semicolon: □;□). At least one of the following parameters is mandatory: name parameter or ef_names. The number of ef_names items in the list should match the number of ef_values items. The first ef_names item will be associated with the first ef_values item and so on. Since 9.1 SR20
ef_values	String	A delimited list of extra field values (the delimiter is semicolon: □;□). The number of ef_names items in the list should match the number of ef_values items. The first ef_names item will be associated with the first ef_values item and so on. Since 9.1 SR20

Returns

Name	Description
Result	

Throws

Name	Description
HOUSEHOLD_NOT_ACTIVE	Household is not active.
HOUSEHOLD_PENDING_INACTIVATION	Household is in pending inactivation process (household planned inactivation date is in the past).
INVALID_PARAMETER	The device extra field name {0} does not exists
INVALID_TERMINAL	The specified terminal to be updated is not found or does not belong to the household.
MISSING_PARAMETER	"The number of extra fields names and extra field values should match."
TERMINAL_NAME_ALREADY_EXISTS	Terminal with the specified name already registered with the same household.

Examples

```
1 UpdateTerminal?mac_address=00:11:22:33:44:55&name=newTerminalName
```

```
1 UpdateTerminal?
mac_address=00:11:22:33:44:55&name=newTerminalName&ef_names=name1;name2&ef_values=value1;val
```

See Also

- [Terminal](#)
- [RegisterTerminal](#)

UpdateTerminalIp*since 5.3*

An action used to update the logged-in household's terminal IP identified by its MAC address.

Parameters

Name	Type	Description
mac_address	String	The MAC address as defined in the terminals tab of the household in RiGHTv.
ip	String	The terminal IP to update.
port	Number	The terminal port to update.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_TERMINAL	The specified terminal is not registered for the subscriber.

Comments

- It is important to use this method when integrating with CA/DRM and using DHCP in the terminals.

Examples

```
1 UpdateTerminalIp?mac_address=00:11:22:33:44:55&ip=1.2.3.4&port=1234
```

See Also

- [Household](#)
- [Terminal](#)
- [RegisterTerminal](#)

UpdateTransactionsPaymentMode*since 6.0*

An action used to update the household's payment mode.

Parameters

Name	Type	Description
value	String	The payment mode value. Possible values: <ul style="list-style-type: none">• "normal"• "internal prepaid"

Returns

Name	Description
Result	

Throws

Name	Description
NON_OPERATOR_SUBSCRIBER_NOT_ALLOWED	The operation is invalid for non-operator subscribers.

Examples

```
1 UpdateTransactionsPaymentMode?value=normal
```

UpdateUsername*since 7.3*

An action used to update household's username. Relevant for online customers only.

Parameters

Name	Type	Description
password	String	The user password.
new_username	String	The new username.
new_username_retyped	String	The new username again.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_NEW_USERNAME	The passed username is not valid.
INVALID_PASSWORD	The passed password is invalid.
OPERATOR_SUBSCRIBER_NOT_ALLOWED	The action is only allowed for non-operator households.
USERNAME_ALREADY_EXISTS	Specified new username already exists in RiGHTv.

Examples

1	UpdateUsername? password=Passw0rd&new_username=bob@mail.com&new_username_retyped=bob@mail.com
---	--

ValidateUsername

since 7.3

An action used to validate username for a new household.

Parameters

Name	Type	Description
username	String	The username of new household to be validated.

Returns

Name	Description
Result	

Throws

Name	Description
INVALID_EMAIL	Specified email structure is not valid.
USERNAME_ALREADY_EXISTS	Specified username already exists in RiGHTv.

Examples

1	ValidateUsername?username=tstUsername
---	---------------------------------------

Appendixes

Appendix A - Action Attributes

Following is a list of all of RiGHTv's supported actions. For each action, the following attributes are detailed:

- **Public** - Indicates if this action is public or private.
- **Requires Login** - Indicates if the user is required to login to RiGHTv prior to calling it.
- **ETag supported** - Indicates if this action support ETag headers.

For more details see the "ETag caching" and the "HTTP Caching Headers" sections in the overview of this document.

Action Name	Public	Requires Login	Etag supported
ActivateTicket	X	V	X
AddBookmark	X	V	X
AddPackageSubscription	X	V	X
AddServiceSubscription	X	V	X
AddToWishList	X	V	X
CancelPreOrder	X	V	X
CheckSmartCard	X	V	X
CloseSession	X	V	X
CloseViewingSessions	X	V	X
CompletePaymentMethod Registration	X	V	X
DeleteBookmark	X	V	X
DeleteMessage	X	V	X
DeletePaymentMethod	X	V	X
DeleteRecording	X	V	X
FinishPackageSubscription	X	V	X
FinishServiceSubscription	X	V	X
GetAggregatedVideo	V	X	X
GetAttachmentDownload Info	X	X	X
GetAttachmentMediaFile List	V	X	X
GetAttachmentPlayingIn fo	X	X	X
GetBookmarkList	X	V	X
GetBouquetList	V	X	X

Action Name	Public	Requires Login	Etag supported
GetCaptchaKey	X	X	X
GetChannelList	V	X	X
GetChannelMetaDataList	V	X	X
GetChannelSupportedModelList	V	X	X
GetClearLivePlayingInfo	V	X	X
GetClientData	X	V	X
GetConfig	X	V	X
GetContentDefinitionList	V	X	X
GetDiscount	V	X	X
GetDiscountList	V	X	X
GetEstDownloadingTerminalList	X	V	X
GetExtendedProgram	V	X	X
GetExtendedProgramList	X	X	X
GetExternalAccount	X	V	X
GetFilteredProgramList	V	X	X
GetGenreList	V	X	X
GetHousehold	X	V	V
GetHouseholdBouquetList	X	V	V
GetHouseholdPackageList	X	V	V
GetHouseholdServiceList	X	V	V
GetHouseholdSubscriptionTransactionList	X	V	X
GetHouseholdTerminalCategoryList	X	V	V
GetLanguageList	V	X	X
GetLiveLicenseAcquisitionInfo	X	V	X
GetLivePlayingInfo	X	V	X
GetMarkedPosition	X	V	X
GetMediaFile	V	X	X
GetMemberList	X	V	X
GetMessageList	X	V	V
GetMsg	X	V	X
GetMsgToken	X	V	X
GetNPVRProgramSupportedModelList	V	X	X

Action Name	Public	Requires Login	Etag supported
GetNPVRUsedQuota	X	V	X
GetOrderList	X	V	X
GetPackageChannelList	V	X	X
GetPackageList	V	X	X
GetPage	V	X	X
GetPageList	V	X	X
GetPageReferenceList	V	X	X
GetParentalRatingList	V	X	X
GetPaymentMethodDetailsInfo	X	V	X
GetPaymentRegistrationPropertyList	X	V	X
GetPersoToken	X	V	X
GetPlayingInfo	X	V	X
GetPreOrderList	X	V	X
GetPricingMatrix	X	X	X
GetPricingMatrixList	V	X	X
GetProgramList	V	X	X
GetProgramPlayingInfo	X	V	X
GetProgramTicketList	X	V	V
GetRechargeOptionList	V	X	X
GetRecordingTicketList	X	V	X
GetReferenceList	V	X	X
GetSeriesTicketList	X	V	X
GetServiceList	V	X	X
GetSessionList	X	V	X
GetSpecificVideoList	X	X	X
GetTerminalList	X	V	X
GetTerminalModelList	V	X	X
GetTimeShiftingServicesConfig	V	X	X
GetTimezoneOffsetList	V	X	X
GetUnifiedIndexList	V	X	X
GetUnifiedList	V	X	X
GetVideo	V	X	X
GetVideoAsset	V	X	X
GetVideoCardList	V	X	X

Action Name	Public	Requires Login	Etag supported
GetVideoCardTicketList	X	V	X
GetVideoDownloadInfo	X	V	X
GetVideoList	V	X	X
GetVideoPackage	V	X	X
GetVideoPackageList	V	X	X
GetVideoPlayingInfo	X	V	X
GetVideoSupportedMode IList	V	X	X
GetVideoTicketList	X	V	X
GetVodGiftList	X	V	V
GetWishList	X	V	V
InsertMessage	X	V	X
IsAlive	X	X	X
Login	X	V	X
Logout	X	V	X
MarkPosition	X	V	X
MarkRecordingPosition	X	V	X
MarkViewedVodGift	X	V	X
OpenSession	X	V	X
OpenViewingSession	X	V	X
OrderGenericService	X	V	X
OrderProgram	X	V	X
OrderRecording	X	V	X
OrderVideo	X	V	X
OrderVideoCard	X	V	X
OrderVideoGift	X	V	X
OrderVideoPackage	X	V	X
PollMessage	X	V	X
PreOrderVideo	X	V	X
RechargeInternalPrepaid	X	V	X
RegisterBouquet	X	V	X
RegisterExternalToken	X	V	X
RegisterHousehold	V	X	X
RegisterTerminal	X	V	X
RemoveFromWishList	X	V	X
RemovePackageSubscription	X	V	X

Action Name	Public	Requires Login	Etag supported
RemoveServiceSubscription	X	V	X
ResendConfirmationEmail	V	X	X
ResetMemberPin	X	V	X
ResetParentalPin	X	V	X
ResetPassword	V	X	X
ReuseMsgToken	X	V	X
SendMessage	X	V	X
SendMsg	X	V	X
SendResponse	X	V	X
SetClientData	X	V	X
SetEntitlement	X	V	X
SetLanguage	X	V	X
TechnicianLogin	X	V	X
UnregisterBouquet	X	V	X
UnregisterExternalToken	X	V	X
UnregisterTerminal	X	V	X
UpdateHousehold	X	V	X
UpdateHouseholdSpendingLimit	X	V	X
UpdateMemberPin	X	V	X
UpdateMemberSpendingLimit	X	V	X
UpdateParentalPin	X	V	X
UpdatePassword	X	V	X
UpdatePaymentAuthentication	X	V	X
UpdateTerminal	X	V	X
UpdateTerminalIp	X	V	X
UpdateTransactionsPaymentMode	X	V	X
UpdateUsername	X	V	X
ValidateUsername	V	X	X

Appendix B - Purple DRM Play Ready License Acquisition API Service

Overview

Purple DRM Play Ready License Acquisition API Service is a propriety service that uses Microsoft Play Ready (wrapped by Purple DRM) in order to give a license to a movie that should be played on a secure player that supports Play Ready.

It is also used to give license for live channels. The protocol between the service and the player is SOAP.

URL

`http://<host name>:<port>/RTEFacade/drm`

Service Request Structure

The SOAP body contains a license request, named challenge. The challenge is in xml format, and should contain an encrypted token including various RiGHTv information

Encrypted Token Path in Challenge

`//Envelope/SOAP/AcquireLicense/challenge/Challenge/LA/CustomData/`

The encrypted token is returned from one of the following RTE APIs:

- GetVideoPlayingInfo
- GetAttachmentPlayingInfo
- GetVideoDownloadInfo
- GetAttachmentDownloadInfo
- GetLivePlayingInfo

Sending Token as Parameter

The token can be sent also as parameter *token*, using GET. If the token parameter is sent, it will not be taken from the Challenge.

Service Response Structure

The service is used as a proxy of Play-Ready. It wraps the license with the SOAP envelope before returning it.

In case of an internal error it returns an error in SOAP fault format.

Playing a Movie Flow

- User orders a movie (OrderVideo).
- User requests movie playing information (including the CAS token) (GetVideoPlayingInfo / GetAttachmentPlayingInfo / GetVideoDownloadInfo / GetAttachmentDownloadInfo).
- User requests the Player to play the movie with the playing URL (including the CAS token).
- Player tries to play and identifies that it is missing a license for the movie.
- Player requests a license from Purple DRM Service (token is part of the request).
- Purple DRM Service authenticates the request, using the CAS token.

- Purple DRM Service passes the request to Play-Ready via purple DRM.
- Play Ready returns the license.
- Player plays the movie.

Channel Zapping Flow

- User asks to view a channel (GetLivePlayingInfo).
- User receives the channel info for viewing (including the CAS token).
- User requests the Player to play the channel with the playing URL (including the CAS token).
- Player tries to play and identifies that it is missing a license for the channel.
- Player requests a license from Purple DRM Service (token is part of the request).
- Purple DRM Service authenticates the request, using the CAS token.
- Purple DRM Service passes the request to Play-Ready via purple DRM.
- Play Ready returns a license.
- Player plays the channel.

Validations

The following validations are performed by the Purple DRM Service:

- That the CAS token is not expired.
- That the CAS token is not reused (can be disabled by a configuration file).
- That the challenge is not reused (can be disabled by a configuration file).

Error Messages

The following error messages are sent by the VO DRM Service:

- Channel doesn't exist, or it was turned 'off'.
- Terminal is not registered for the logged in household.
- Service not available on the device.
- The logged in household is not subscribed to the required service.

Notes

- In a clustered environment, if two requests are sent with the same token to different servers, there is a chance that token reuse will not be discovered, in case the time difference between the two request is very short.

Appendix E - Purple DRM VO API Service

Overview

Purple DRM VO API service is a proprietary service that uses the VO DRM in order to give a license to a movie that should be played on a secure player that supports VO DRM.

It is also used to give a license for live channels. The protocol between the service and the player is XML-based. VO DRM service is used for getting licensing both on post-delivery or pre-delivery. The license on post-delivery will be requested when playing a movie or zapping to a channel, whereas the license on pre-delivery will be requested prior to the channel zapping (relevant only on live content). Furthermore, package mode is supported additionally to the channel mode (default). Package mode allows getting the license per package instead of per channel.

URL

`http://<host name>:<port>/RTEFacade/vodrm`

Service Request Structure

The body contains a license request, named challenge. The challenge is in XML format, and should contain an encrypted token including various RiGHTv information.

Encrypted Token Path in Challenge

`//VoDrm/LicenseChallenge/CustomData/`

Service Response Structure

The service is used as a proxy of VO DRM.

In case of an internal error it returns an error in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<VoDrm version="{vodrmRequestVersion}" xmlns="http://viaccess-orca.com/vodrm">
<Error>
<Code>errorCode</Code>
<Message>errorMessage</Message>
</Error>
</VoDrm>
```

- Response version (`//VoDrm@version`) is identical to the version provided on the request.

Getting live license acquisition info (pre-delivery)

- Starting the STB on the first time.

- STB client sends license acquisition info request using GetLiveLicenseAcquisitionInfo RTE action.
- Input to this action can be: channel, package or none (when no input parameters passed all the subscriptions will be considered).
- The RTE action will authenticate and validate the request.
- Receives encrypted token and rights object.

Getting Live License Acquisition Info (Pre-delivery Renewal)

- STB checks licensing renewals time stamps.
- STB client sends license acquisition information request using GetLiveLicenseAcquisitionInfo action.
- Input to this action can be a list of CAS IDs.
- The RTE action will authenticate and validate the request.
- Receives encrypted token.

Getting License

- VO DRM agent requests license from the VO DRM Service (token is part of the body).
- VO DRM Service authenticates the request, using the CAS token.
- VO DRM Service passes the request to the licensor.
- The licensor returns the license.

Channel Zapping on Pre-delivery

- User asks to view a channel (GetLivePlayingInfo).
- User receives the channel information for viewing (including the CAS token).
- User requests the Player to play the channel with the playing URL (including the CAS token).
- Player plays the channel using the license saved before.

Channel Zapping on Post-delivery

- User receives the channel information for viewing (including the CAS token).
- User requests the Player to play the channel with the playing URL (including the CAS token).
- Player tries to play and identifies that it is missing a license for the channel.
- Player requests a license from VO DRM Service (token is part of the body).
- VO DRM Service returns a license.
- Player plays the channel.

Validations

The following validations are performed by the VO DRM Service on post-delivery:

- That the CAS token is not expired.
- That the CAS token is not reused (can be disabled by a configuration file).
- That the challenge is not reused (can be disabled by a configuration file).

The following validations are performed by the VO DRM Service on pre-delivery:

- That the CAS token is not expired.

Errors

The following error messages are sent by the VO DRM Service:

Error code	Description
CHALLENGE_REUSE	The provided challenge is reused.

Error code	Description
EXPIRED_TOKEN	The token has expired.
INVALID_TOKEN	The token is invalid.
REUSED_TOKEN	The license token was reused.
GENERAL_ERROR	A general error has occurred.

Notes

- In a clustered environment, if two requests are sent with the same token to different servers, there is a chance that token reuse will not be discovered, if the time difference between the two request is very short.
- In the configuration, the token expiration time needs to be limited to a short period in order to prevent forgeries.

Appendix F - Purple DRM VO Personalization API Service

Overview

Purple DRM VO personalization API service is a propriety service that uses VO DRM in order to give personalization to a VODRM agent.

URL

`http://<host name>:<port>/RTEFacade/perso`

Service Request Structure

The SOAP body contains the personalization request, named PersoChallenge. The PersoChallenge is in xml format, and should contain an encrypted token including various RiGHTv information. The PersoChallenge is generated by the client application via VODRM Agent.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>

<VoDrm version="3.0" xmlns="http://viaccess-orca.com/vodrm/3">

<PersoChallenge>

<PersoChallengeData>HUYNh7xKK/4gIjaEwDFTHfQkAMC9Mchikwq+fAegYbg=</
PersoChallengeData>

<CustomData>sxOPODs2fnm8dN2CLFKw9c4cdhpvxIlgGbUATz74meEOgo/1aYnTqVculYQT4LnmGbJTU7
WY1dy9wCq5ZtY2LysNpEdISGHuoKAetx4/dPw=</CustomData>

</PersoChallenge>

</VoDrm>
```

Encrypted Token Path for PersoChallenge

`//VoDrm/PersoChallenge/CustomData/`

Personalization token for VODRM client is returned from the following RTE APIs:

- GetPersoToken

Example: `<?xml version='1.0' encoding='UTF-8'?>`

```
<PersoTokenEntity dataHash='HUYNh7xKK/4gIjaEwDFTHfQkAMC9Mchikwq+fAegYbg='
persoTokenString='sxOPODs2fnm8dN2CLFKw9c4cdhpvxIlgGbUATz74meEOgo/1aYnTqVculYQT4LnmGbJT
Cq5ZtY2LysNpEdISGHuoKAetx4/dPw='>
</PersoTokenEntity>
```

Service Response Structure

The service is used as a proxy of Personalization service.

Personalization Response

```
<?xml version="1.0" encoding="UTF-8"?>
<VoDrm version="1.0" xmlns="http://viacces-orca.com/vodrm">
  <PersoResponse>
    <PersoData>AQACAAEDAAMAAQIH</PersoData>
    <Signature>098ybbI30JnrgbNDYFaU</Signature>
    <Certificates>
      <VpsCertificate>098ybbI30JnrgbNDYFaU</VpsCertificate>
    </Certificates>
  </PersoResponse>
</VoDrm>
```

Personalization Flow

1. The end-user passes the RiGHTv authentication phase (login/password).
2. If the terminal is not personalized, the Client Application requests RiGHTv to provide the personalization DRM token.
3. RTE generates a personalization DRM token information containing:
 - a. The encrypted personalization DRM token.
 - b. The personalization DRM token hash.
4. The Client Application generates the personalization challenge providing mainly:
 - a. The encrypted personalization DRM token.
 - b. The personalization DRM token hash.
5. The Client Application sends the VODRM personalization challenge in http POST mode to the RTE Facade:

`http://<host name>:<port>/RTEFacade/perso`
6. The VO personalization server receives the personalization request from RiGHTv.

7. Once having checked that the personalization challenge is correct, including checking the custom data integrity, the Personalization Server fills a personalization response container with DRM keys allocated for the VODRM agent and forwards it the response through RiGHTv.
8. The Client Application receives the personalization response, forwards it to the VODRM agent and receives the status of the personalization.
9. VODRM agent is personalized and has obtained its DRM keys.

Validations

The following validations are performed by the Personalization Service:

- That the Perso token is not expired.

Errors

The following error codes are sent by RiGHTv DRM personalization Service:

Error code	Description
EXPIRED_TOKEN	The token has expired.
GENERAL_ERROR	A general error has occurred.

Personalization Error Response

```
<?xml version="1.0" encoding="UTF-8"?>

<VoDrm version="1.0" xmlns="http://viacces-orca.com/vodrm">

<Error>

<Code>code</Code>

<Message>message</Message>

</Error>

</VoDrm>
```

Appendix G - Captcha Image Servlet

Overview

Captcha Image Servlet is designated to retrieve captcha images. Once captcha key was requested, Captcha Image Servlet is able to generate a relevant captcha image.

Usage

`http://<host name>:<port>/RTEFacade/captcha-image?key=<key>`

Where key is a captcha key.

Appendix I - VO Widevine DRM API Service

Overview

The VO Widevine DRM API service is a propriety service that uses Widevine DRM to provide licenses for content to be played on a Widevine DRM enabled secure player.

The service provides licenses for live channels and VOD content.

The protocol between the service and the player is HTTP/S.

Workflows:

The following post-delivery workflows are outlined below:

- Playing a movie
- Channel zapping

Playing a Movie on Post-delivery:

The workflow for playing a movie on post-delivery is as follows:

- The user orders a movie (OrderVideo).
- The user requests movie playing information (GetVideoPlayingInfo / GetAttachmentPlayingInfo / GetVideoDownloadInfo / GetAttachmentDownloadInfo / GetTvShowEpisodePlayingInfo).
- The playing URL is returned together with a license token.
- The user requests the player to play the movie with the playing URL.
- The player tries to play the movie but identifies that it is missing a VOD license.
- The player triggers the application to request a license.
- The application requests a license from the Widevine DRM Service (Servlet) in RiGHTv. (The license token is included in the request URL. The challenge is sent in the body of the request.)
- The Widevine DRM Service authenticates the request using the license token.
- The Widevine DRM Service passes the request through the Purple DRM plugin to the Purple DRM Licenser (which performs additional validation of the client token).
- The license is returned to the application.
- The player plays the movie.

Channel Zapping on Post-delivery:

The workflow for playing a movie on post-delivery is as follows:

- The user requests to view a channel (GetLivePlayingInfo).
- The user receives the channel info for viewing (including the license token).
- The user requests the player to play the channel with the playing URL.
- The player tries to play the channel but identifies that it is missing a license for the channel.
- The player triggers the application to request a license.
- The application requests a license from the Widevine DRM Service (Servlet) in RiGHTv. (The license token is included in the request URL. The challenge is sent in the body of the request.)
- The Widevine DRM Service authenticates the request using the license token.
- The Widevine DRM Service passes the request through the Purple DRM plugin to the Purple DRM Licenser (which performs additional validation of the client token).
- The license is returned to the application.
- The player plays the channel.

Interface

URL

`https://<host name>:<port>/RTEFacade/widevinedrm?token=<license token>`

Method

HTTP POST.

Arguments

Name	Type	Mandatory	Comments
<license token> (url param)	String	Y	The token containing the information required to generate the license, as provided by RiGHTv before hand.
Challenge (in POST body content-type:application/octet-stream)	binary	Y	The value of the challenge generated by the client.

Return value

The return value is the license in xml form (content-type:application/octet-stream).

Name	Value	Comments
License -content-type:application/octet-stream	binary	Widevine License

Validations

The following validations are performed by the involved entities on post-delivery:

- That the license token has not expired.
- That the passed client token is assigned to the requesting terminal.

Error Messages

The following error messages are sent by the Widevine DRM Service:

HTTP Response Code	Error Code	Description
400	client-token-mismatch-error	The client token does not match the client token registered for the terminal. This error is generated by the Licenser.
400	expired-token-error	The license token passed has expired.
500	general-error	All other errors.

Examples

An example of a request sent by the client is shown below:

```
POST https://<host name>:<port>/RTEFacade/widevinedrm?token=<license token>
HTTP/1.1
```

Request header

Content-Type: application/octet-stream

Content-Length: <length>

Request body

<challenge(byte content)>

An example of a response sent by the server is shown below:

Response header

Content-Type: application/octet-stream

Content-Length: <length>

Response body

<license (byte content)>

A template example of an error response sent by the server is shown below:

HTTP/1.1 400 Bad Request

Date: Thu, 20 Dec 2012 01:09:06 GMT

Content-Type: application/octet-stream

<ERROR>

<CODE>

[ERROR CODE]

</CODE>

<TEXT>

[TEXT DESCRIPTION]

</TEXT>

</ERROR>

Appendix K - Fairplay DRM API Service

Overview

The VO Fairplay DRM API service is a propriety service that uses Fairplay DRM to provide licenses for content to be played using a Fairplay DRM enabled secure player. The service provides licences for live channels and VOD content.

The protocol between the service and the player is HTTP/S.

Workflows

The following post-delivery workflows are outlined below:

- Playing a movie
- Channel zapping

Playing a Movie on Post-delivery

The workflow for playing a movie on post-delivery is as follows:

- The user orders a movie (OrderVideo).
- The user requests movie playing information (GetVideoPlayingInfo / GetAttachmentPlayingInfo / GetVideoDownloadInfo / GetAttachmentDownloadInfo / GetTvShowEpisodePlayingInfo).
- The playing URL is returned together with a license token.
- The user requests the player to play the movie with the playing URL.
- The player tries to play the movie but identifies that it is missing a VOD license.
- The player triggers the application to request a license.
- The application requests a license from the Fairplay DRM Service (Servlet) in RiGHTv. (The license token is included in the request URL. The challenge is sent in the body of the request.)
- The Fairplay DRM Service authenticates the request using the license token.
- The Fairplay DRM Service passes the request through the Purple DRM plugin to the Purple DRM Licenser (which performs additional validation of the client token).
- The license is returned to the application.
- The player plays the movie.

Channel Zapping on Post-delivery

The workflow for channel zapping on post-delivery is as follows:

- The user requests to view a channel (GetLivePlayingInfo).
- The user receives the channel info for viewing (including the license token).
- The user requests the player to play the channel with the playing URL.
- The player tries to play the channel but identifies that it is missing a license for the channel.
- The player triggers the application to request a license.
- The application requests a license from the Fairplay DRM Service (Servlet) in RiGHTv. (The license token is included in the request URL. The challenge is sent in the body of the request.)
- The Fairplay DRM Service authenticates the request using the license token.
- The Fairplay DRM Service passes the request through the Purple DRM plugin to the Purple DRM Licenser (which performs additional validation of the client token).
- The license is returned to the application.
- The player plays the channel.

Interface

URL

`https://<host name>:<port>/RTEFacade/fairplaydrm?token=<license token>`

Method

HTTP POST.

Arguments

Name	Type	Mandatory	Comments
<license token> (url param)	String	Y	The token containing the information required to generate the license, as provided by RiGHTv beforehand.
Challenge (in POST body either content-type:application/octet-stream or content-type:text/plain)	binary or text base64 encoded	Y	The value of the challenge generated by the client

Return value

The return value is the license in either in binary form (content-type:application/octet-stream) or xml form (content-type:application/xml) depending upon the request content-type.

Name	Value	Comments
if request content-type was application/octet-stream the license response is binary - content-type:application/octet-stream. if request content-type was text/plain License response is text - content-type:application/xml. <FAIRPLAY_RESPONSE> <LICENSE> [FairPlay base 64 CKC message] </LICENSE> </FAIRPLAY_RESPONSE>	binary or xml	Fairplay License

Validations

The following validations are performed by the involved entities on post-delivery:

- That the license token has not expired.
- That the passed client token is assigned to the requesting terminal.

Error Messages

The following error messages are sent by the Fairplay DRM Service:

HTTP Response Code	Error Code	Description
400	client-token-mismatch-error	The client token does not match the client token registered for the terminal. This error is generated by the Licenser.

HTTP Response Code	Error Code	Description
400	expired-token-error	The license token passed has expired.
500	general-error	All other errors.

Examples

An example of a request sent by the client is shown below:

```
POST https://<host name>:<port>/RTEFacade/fairplaydrm?token=<license token>
HTTP/1.1
```

Request header

Content-Type: text/plain

Content-Length: <length>

Request body

<challenge(byte content)>

An example of a response sent by the server is shown below:

Response header

Content-Type: application/xml

Content-Length: <length>

Response body

<FAIRPLAY_RESPONSE>

<LICENSE>

[FairPlay base 64 CKC message]

</LICENSE>

</FAIRPLAY_RESPONSE>

A template example of an error response sent by the server is shown below:

HTTP/1.1 400 Bad Request

Date: Thu, 20 Dec 2012 01:09:06 GMT

Content-Type: application/xml

<ERROR>

```
<CODE>  
[ ERROR_CODE ]  
</CODE>  
  
<TEXT>  
[ TEXT_DESCRIPTION ]  
</TEXT>  
  
</ERROR>
```

Appendix M - CSP Personalization API Service

Overview

CSP Personalization API Service is a propriety service that uses Purple DRM in order to give a personalization license to Connected Sentinel Player (CSP).

The protocol between the service and the player is HTTP/S.

URL

`http://<host name>:<port>/RTEFacade/cspperso`

Service Response Structure

The service is used as a proxy of Purple DRM.

Pre-condition for successful personalization

- Post login, the application performs register terminal to the logged-in household using the terminal identifier in RiGHTv, with the unique value identifying the terminal provided by the Connected Sentinel Player (CSP).

Personalization Flow

- The CSP invokes CPS personalization service
- RiGHTv passes the perso-challenge to the PPDRM Plugin
- PPDRM Plugin calls getTerminalInfo on the PersoServer using the perso-challenge.
- PPDRM Plugin checks if terminal identifier exists in RiGHTv, using configuration to determine the terminal identifier type (SERIAL_NUMBER | SMART_CARD_ID | MAC_ADDRESS). (Default type is SERIAL_NUMBER).
- If the verification passes the perso response is returned to RiGHTv, otherwise an exception is thrown
- If an exception is thrown, RiGHTv will return 500 Internal Server Error response.

Validations

The following validations are performed by the CSP Personalization Service on post-delivery:

- Validates that terminal id is registered to any household in RiGHTv

Error Messages

The error message will always be empty with HTTP error code 500

