

opentext™

Dimensions RM

Software version: 24.4 (12.12.1)

Web Service / Rest Service Reference



Copyright © 2024 Open Text.

The only warranties for products and services of Open Text and its affiliates and licensors ("Open Text") are as may be set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Open Text shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Product version: 24.4 (12.12.1)

Last updated: October 18, 2024

Table of Contents

Preface	11
Objective	11
Audience.	11
Contacting Technical Support	11
License and Copyright Information for Third-Party Software	12
Chapter 1	
RM Web Service	13
General	14
Methods	15
AddObjectToCollection	19
AddSyncXrefLink	20
AppendRMDocumentEx	21
AppendRtmDocument	23
ApproveChangeRequest	25
AsyncAppendRtmDoc.	26
AsyncCreateRtmDoc	27
AsyncQueryRtmDoc.	28
AsyncReplaceRtmDoc.	29
ChangeRequest.	30
CollectionAssociationWithCMPProject.	31
CreateBaseline	32
CreateCategory	34
CreateChildDocument	35
CreateCollection	36
CreateObject	37
CreateObjectEx	38
CreateObjectWithAttachment	39
CreateProduct.	41
CreateRMDocumentEx	42
CreateRtmDocSnapshot	45
CreateRtmDocument	46
CreateTemplate	49
DeleteBaseline	51
DeleteObjects	52
DeleteRtmDocument	53
DeleteSyncXrefLink	54
DeleteTemplate.	55
DocumentAddChapter	56
DocumentMoveObject	57
DocumentRemoveObject	58
FindAndReplaceObject	59
FindLinkedObjects	60

FindModifiedObjects	62
FindObjects	63
FindObjectsInClasses	65
FindObjectsInCollections	67
FindRtmDocument	69
FindTemplate	70
getAsyncAppendRtmDocStatus	71
getAsyncCreateRtmDocStatus	72
getAsyncQueryRtmDocStatus	73
getAsyncReplaceRtmDocStatus	74
getAsyncStatus	75
GetAttributeDefInfo	76
GetAvailableTransitions	77
GetCategoryByDisplayName	79
GetCategoryById	80
GetClassDefInfo	82
GetCollectionById	83
GetCollectionByName	84
GetCollectionInfo	85
GetObjectHierarchyInfo	86
GetObjectsInCollection	87
GetObjectsInCollections	88
GetObjectsInRMDocument	89
GetProjectInfo	90
GetRMDocSnapshotDiffUrl	91
GetRMDocSnapshots	92
GetRtmDocSnapshotById	93
GetRtmDocSnapshotByName	94
GetSharedFile	95
GetTemplateFile	96
getTestAsyncStatus	97
GetUserAttributeDefInfo	98
IsSSOLoginEnabled	99
LinkObjects	100
LinkObjectsEx	101
LinkObjectsExById	102
ListAllAllowedCategories	103
ListCategories	104
ListCollections	105
ListCollectionsEx	106
ListCollectionsEx2	107
ListObjectLinks	108
ListRootCategories	109
ListRtmDocuments	110
ListRtmDocumentsEx	112
ListTemplates	114
Login	115
LoginByProjectId	118

LoginEx	120
Logout	122
ModifyObjectsByScript	123
MoveObjectInHierarchy	126
ProposeNewRequirement	127
PutSharedFile	129
QueryAttributes	130
QueryAttributesEx	132
QueryAttributesEx2	133
QueryClasses	135
QueryClassesEx	136
QueryDatabases	137
QueryLinkClasses	138
QueryProjects	139
QueryProjectsEx	140
QueryRtmDocumentStructure	141
QuerySystemInfo	143
ReadObject.	144
ReadObjectAttachment	145
ReadObjectEx	147
ReadObjects	148
ReadSyncXrefLink	149
RejectChangeRequest	150
RemoveObject	151
RemoveObjectFromCollection	152
ReplaceObject.	153
ReplaceObjectEx	154
ReplaceObjectWithAttachment	155
ResolveSuspicion.	156
RTMDocumentAddChapter	157
RTMDocumentAddRequirement	158
RTMDocumentExchangeRequirement.	159
RTMDocumentRemoveRequirement	160
runTestAsync	161
SendCMNotifications	162
TransitionObject	163
UnlinkObjects	165
UnlinkObjectsEx	166
UnlinkObjectsExById	167
UpdateObject	168
UpdateObjectAttachment	169
UpdateObjectEx	171
UpdateTemplate	172
UpdateTemplateFile	173
UploadImageForHTMLAttribute	174
Complex Types	176
addObjectToCollectionResponse	180
AppendRMDocumentExResult	181

AppendRtmDocumentResult	182
AsyncAppendRtmDocResponse	183
AsyncCreateRtmDocResponse	184
AsyncReplaceRtmDocResponse	185
AttributeDef	186
AttributeDefEx	187
AttributeDefEx2	188
AttributeDefExList	189
AttributeDefEx2List	190
AttributeDefInfo	191
AttributeDefList	192
AttributeDefSpecifier	193
AttributeFormatDef	194
Category	195
CategoryInfo	196
CategoryAssign	197
CategoryList	198
CategoryReq	199
ClassDef	200
ClassDefEx	201
ClassDefExList	202
ClassDefInfo	203
ClassDefList	204
ClassSpecifier	205
CmdStatus	206
CMProjectInfo	207
CollectionEx	208
CollectionEx2	209
CollectionEx2List	210
CollectionExList	211
CollectionInfo	212
CollectionList	213
CollectionSpecifier	214
CollectionTransferRules	215
CollectionTransferRulesEx	216
CommandStatus	217
CreateCollectionEx	218
CreateChildDocumentData	219
CreateRMDocumentExResult	220
CreateRtmDocumentResult	221
DatabaseList	222
DocumentObject	223
DocumentsFilter	224
ECPObjInfo	225
ECPsObjectInfo	226
FileAttachment	227
FindRequirementOut	228
FormatDef	229

getAsyncAppendRtmDocStatusResponse	230
getAsyncCreateRtmDocStatusResponse	231
getAsyncQueryRtmDocResponse	232
getAsyncQueryRtmDocumentStatusResponse	233
getAsyncReplaceRtmDocStatusResponse	234
getAsyncStatusResponse	235
GetAttributeDefInfoReply	236
GetAttributeDefInfoRequest	237
GetClassDefInfoReply	238
GetClassDefInfoRequest	239
GetCollectionInfoReply	240
GetCollectionInfoRequest	241
GetProjectInfoReply	242
GetProjectInfoRequest	243
GetUserAttributeDefInfoReply	244
GetUserAttributeDefInfoRequest	245
HierarchyInfo	245
IdList	246
ImageList	247
ImageUploadResult	248
ImageUploadResultList	249
InRequirement	250
InTransition	251
Link	252
LinkClassDef	253
LinkClassDefList	254
LinkList	255
ListItemDef	256
ListItems	257
LoginExResultInfo	257
NameValue	258
NameValueList	259
ns-chapterDetails	260
Object	261
ObjectDef	262
ObjectDetails	263
ObjectDetailsList	264
ObjectsList	265
ProjectEx	266
ProjectExList	267
ProjectInfo	268
ProjectList	269
ProjectSpecifier	270
ProposeNewRequirementResult	271
RelationList	272
RelationWithObjects	273
removeObjectFromCollectionResponse	274
ReplaceRequirementList	275

ReplaceRequirementOut	276
Report	277
Requirement	278
RequirementFindList	279
RequirementIdentifier	280
RequirementIdentifierList	281
RequirementList	282
RequirementOut	283
Result	284
RtmDocSnapshot	285
RtmDocSnapshotOut	286
RtmDocSnapshotOutList	287
RtmDocument	288
RtmDocumentList	289
RtmDocumentListEx	290
RtmDocumentProperties	291
RtmDocumentPropertiesEx	292
RtmDocumentWithCategory	293
RtmSnapshotListEx	294
RtmSnapshotPropertiesEx	295
runTestAsyncInputData	296
runTestAsyncResponse	297
runTestAsyncResponseData	298
Script	299
SearchRequirement	300
StatusMessage	301
StatusMessageList	302
StringList	303
SyncXrefLink	304
SyncXrefLinkList	305
SystemInfo	306
Template	307
TemplateList	308
testAsyncStatusResponse	309
Transition	310
Transitions	311
TransitObject	312
UserAttributeDefInfo	313
UserValue	314
UserValueList	315
Simple Types	316
cmdStatusCode	317
IdNameEnumeration	318
IdNameUnion	319
reqStatusCode	320
statusCode	321

RM Admin Web Service	323
General	324
Methods	325
AssignCategoryToGroup	326
AssignGroupToProject	327
AssignUsersInGroupPerCategory	328
AssignUserToGroup	330
AssignUserToProject	331
CheckPermission	332
CopyGroup	333
CopyUser	334
CreateGroup	336
CreateUser	337
CreateLDAPUser	338
CreateSSOUser	339
DeleteGroup	340
DeleteUser	341
GetGroupInfo	342
GetGroups	343
GetLoginSources	344
GetUserInfo	345
GetUsers	346
GrantPermissionToGroup	347
GrantPermissionToUser	348
InheritPermissionForUser	349
RevokePermissionFromGroup	350
RevokePermissionFromUser	351
UnassignCategoryFromGroup	352
UnassignGroupFromProject	353
UnassignUserFromGroup	354
UnassignUserFromProject	355
UpdateListAttributeValues	356
UpdateUser	358
UpdateUserAttributeValues	359
Complex Types	360
AdditionalUserInfo	361
Group	362
GroupInfo	363
GroupList	364
IdList	365
LoginSource	366
LoginSources	367
NameValue	368
NameValueList	369
Result	370
StringList	371
User	372
UserInfo	373

UserInfoExt	374
UserItemDef	375
UserItems	376
UserList	377
UserNameList	378
Appendix	379
Resource and Action Parameters	379

Chapter 3

RM Restful Services	381
General	382
Authentication for Rest Services	384
Methods	386
Reports	387
Report	390
Run	394
Containers	397
ContainerRequirements	400
Index	403

Preface

This document describes the Dimensions RM web services and restful services. Web services and restful services allow software developers to access functionality from custom programs.

Objective

The purpose of this guide is to describe the functionality and interfaces of Dimensions RM web services and restful services.

Audience

The audience for this manual is administrators and software developers who have extensive domain knowledge about Dimensions RM and programming languages such as Java or JavaScript.

Contacting Technical Support

Open Text provides technical support for all registered users of this product, including limited installation support for the first 30 days. If you need support contact us at:

<http://supportline.microfocus.com>

Language-specific technical support is available during local business hours. For all other hours, technical support is provided in English.

You can use the web page to:

- Report problems and ask questions.
- Obtain up-to-date technical support information, including that shared by our customers via the Web, automatic e-mail notification, newsgroups, and regional user groups.
- Access a knowledge base, which contains how-to information and allows you to search on keywords for technical bulletins.
- Download updates and fix releases for your Open Text products.

License and Copyright Information for Third-Party Software

For license and copyright information of third-party software included in this release, check the file `Third_Party_Licenses.txt`, which can be found in the Dimensions RM installation directory, e.g. `C:\Program Files\Open Text\Dimensions 24.4\RM`.

Chapter 1

RM Web Service

General	14
Methods	15
Complex Types	176
Simple Types	316

General

The Dimensions RM Web Service contains all non-administrative functions and types.

The default URL is *http://localhost:8080/rtmBrowser/WebServices*. Adjust protocol, server name and port to match your environment.

Methods

Name	Description
AddObjectToCollection	Add an object to a collection
AddSyncXrefLink	Adds a link to Sync_Xref table
AppendRMDocumentEx	Append an RM document to an existing document.
AppendRtmDocument	Append an RM document to an existing document.
ApproveChangeRequest	Approve proposed change for the object
AsyncAppendRtmDoc	Append an RM document to an existing document.
AsyncCreateRtmDoc	Create a new RM document .
AsyncQueryRtmDoc	Creates an XML representing RM Document structure.
AsyncReplaceRtmDoc	Replace an existing project rtm document.
ChangeRequest	Create a change request for the specified requirement
CollectionAssociationWithCMPProject	Associate or un-associate RM collection/baseline with CM project/stream.
CreateBaseline	Create a baseline with given requirements (possible to associate it to a CM baseline)
CreateCategory	Creates a new RM category
CreateChildDocument	Creates a new Child document of the specified Parent document
CreateCollection	Creates an empty collection
CreateObject	Creates a new requirement, based on given class name and requirement data
CreateObjectEx	Creates a new requirement, based on given class name and requirement data
CreateObjectWithAttachment	Creates a new requirement with attachment, based on given class id and requirement data
CreateProduct	Create a new Product
CreateRMDocumentEx	Create a new RM document
CreateRtmDocSnapshot	Creates document snapshot by document id
CreateRtmDocument	Creates a new document with given content
CreateTemplate	Creates a new project template for importing document using RM Import
DeleteBaseline	Deletes a baseline
DeleteObjects	Marks requirements as deleted
DeleteRtmDocument	Deletes document specified by document id
DeleteSyncXrefLink	Deletes a link from Sync_Xref table
DeleteTemplate	Deletes the project template specified by template id
DocumentAddChapter	Add chapter to the document

Name	Description
DocumentMoveObject	Moves object in the document
DocumentRemoveObject	Marks objects as removed from the Document
FindAndReplaceObject	Search requirement(s) based on attribute list and replaces those with specified requirement data
FindLinkedObjects	Returns links and requirements linked to specified requirement
FindModifiedObjects	Find objects that have been modified since the given time\date
FindObjects	Find objects that match the given search criteria.
FindObjectsInClasses	Find objects in classes that match the given search criteria.
FindObjectsInCollections	Find objects in the given collections
FindRtmDocument	Get a document specified by id
FindTemplate	Get a document template specified by id
getAsyncAppendRtmDocStatus	Get status from the async. commmand
getAsyncCreateRtmDocStatus	Get status from the async. commmand
getAsyncQueryRtmDocStatus	Get status from the async. commmand
getAsyncReplaceRtmDocStatus	Get status from the async. commmand
getAsyncStatus	Get status from the asynchronous commmand
GetAttributeDefInfo	Get information about attribute definition
GetAvailableTransitions	Get all available transitions for requirement defined in the workflow
GetCategoryByDisplayName	Get category id by category display name
GetCategoryById	Get category information by given category id
GetClassDefInfo	Get definition of specified class by class Id/name
GetCollectionById	Get a collection specified by collection id
GetCollectionByName	Get a collection specified by name
GetCollectionInfo	Get information about specified collection
GetObjectHierarchyInfo	Retrieves information about the hierarchy position for the specified requirement
GetObjectsInCollection	Get objects in the given collection
GetObjectsInCollections	Get objects in the given collections
GetObjectsInRMDocument	Get object in the given document
GetProjectInfo	Get information about project
GetRMDocSnapshotDiffUrl	Get the url of comparison specified document snapshots
GetRMDocSnapshots	Returns list of snapshots for specified document.
GetRtmDocSnapshotById	Retrieve document snapshot information by id.
GetRtmDocSnapshotByName	Retrieve document snapshot information by name.
GetSharedFile	Download a shared file for local use
GetTemplateFile	Get the template file of the template specified by template id
getTestAsyncStatus	Get status from the async. test commmand

Name	Description
GetUserAttributeDefInfo	Get user attribute definition with list of users filtered by category
IsSSOLoginEnabled	Returns information if SSO enabled or not
LinkObjects	Link multiple children to a parent object
LinkObjectsEx	Link multiple children to a parent object
LinkObjectsExById	Link multiple children to a parent object
ListAllAllowedCategories	Get a list of all allowed categories for the project.
ListCategories	Get a list of child categories for the given parent category.
ListCollections	Get a list of collection names in the project.
ListCollectionsEx	Get a list of collections in the project.
ListCollectionsEx2	Get a list of collections in the project.
ListObjectLinks	List secondary links for a specified requirement
ListRootCategories	Get a list of top-level categories in the project.
ListRtmDocuments	Get a list of rtm documents in the project.
ListRtmDocumentsEx	Get a list of rtm documents and snapshots in the project.
ListTemplates	Get a list of document templates in the project.
Login	Login into RTM - use returned sessionId for subsequent calls.
LoginByProjectId	Login into RTM - use returned sessionId for subsequent calls.
LoginEx	Login into RTM and return if it was admin login along with sessionId.
Logout	Logout of RTM.
ModifyObjectsByScript	Update/replace/approve/delete/undelete/transition requirements by script name/text
MoveObjectInHierarchy	Move requirement to the other position in the hierarchy
ProposeNewRequirement	Creates a requirement proposal
PutSharedFile	Upload a shared file for access by others
QueryAttributes	Return the list of defined attributes in a class
QueryAttributesEx	Return the list of defined attributes in a class
QueryAttributesEx2	Return the list of defined attributes in a class
QueryClasses	Return the list of defined classes in project
QueryClassesEx	Return the list of defined classes in project
QueryDatabases	Return the list of oracle databases
QueryLinkClasses	Return the list of linked classes information in project
QueryProjects	Return the list of defined projects in database
QueryProjectsEx	Return the list of defined projects in database
QueryRtmDocumentStructure	Creates an XML representing RM Document structure
QuerySystemInfo	Return the system information
ReadObject	Read object information with all the attribute values

Name	Description
ReadObjectAttachment	Read attachment of a requirement
ReadObjectEx	Read object information with all the attribute values
ReadObjects	Read object information with all the attribute values
ReadSyncXrefLink	Reads a link from Sync_Xref table
RejectChangeRequest	Service definition of function ns__RejectChangeRequest
RemoveObject	RemoveObject Removes a requirement from the database
RemoveObjectFromCollection	Service definition of function ns__RemoveObjectFromCollection
ReplaceObject	Replaces a RM requirement with specified requirement data
ReplaceObjectEx	Replaces a RM requirement with specified requirement data
ReplaceObjectWithAttachment	Replace a RM requirement with attachment, given a class id and requirement data
ResolveSuspicion	Clears suspect links
RTMDocumentAddChapter	Add the new chapter after the existing chapter at the same level.
RTMDocumentAddRequirement	Add requirement in the existing Chapter.
RTMDocumentExchangeRequirement	Exchange requirements inplace.
RTMDocumentRemoveRequirement	Marks requirements as removed from the Document.
runTestAsync	Run the async. test commmand
SendCMNotifications	Send notifications to CM
TransitionObject	Transition requirement to the specified state
UnlinkObjects	Unlink multiple children from a parent object
UnlinkObjectsEx	Unlink multiple children from a parent object
UnlinkObjectsExById	Unlink multiple children from a parent object
UpdateObject	Updates a RM requirement with specified requirement data
UpdateObjectAttachment	Updates a RM requirement attachment
UpdateObjectEx	Updates a RM requirement with specified requirement data
UpdateTemplate	Update the template name, filename, and description fields and the template file.
UpdateTemplateFile	Replace the template file of the template specified by template id
UploadImageForHTMLAttribute	Upload picture and file to the RM server

AddObjectToCollection

Add an object to a collection

Style

Document

Input (Literal)

The input of this method is the argument AddObjectToCollection having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
objectIdentifier	RequirementIdentifier	1..1	A combination of PUID/classname or objectID/classname that specifies a project requirement.
collectionName	string	1..1	The collection name

Output (Literal)

The output of this method is the argument addObjectToCollectionResponseResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	addObjectToCollectionResponse	1..1	Status of adding object to collection

Remarks

The AddObjectToCollection call provides a method for adding the requirement to the specified collection

Example of usage:

```
<urn:AddObjectToCollection>
  <sessionId>1443452989</sessionId>
  <objectIdentifier>
    <sClassName>Marketing_Requirements</sClassName>
    <nObjectId>38</nObjectId>
  </objectIdentifier>
  <collectionName>Scoping</collectionName>
</urn:AddObjectToCollection>
```

AddSyncXrefLink

Adds a link to Sync_Xref table

Style

Document

Input (Literal)

The input of this method is the argument AddSyncXrefLink having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
aXref	SyncXrefLink	1..1	A reference entry consists of a Dimensions RM class id, Dimensions RM PUID, foreign tool id and foreign item identifier.

Output (Literal)

The output of this method is the argument AddSyncXrefLinkResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	boolean	1..1	A flag indication whether the adding reference was successful or not

Remarks

The AddSyncXrefLink call provides a method for adding an association entry between Dimensions RM and the other (foreign) tool in SYNC_XREF table.

Example of usage:

```
<urn:AddSyncXrefLink>
  <sessionId>1444320707</sessionId>
  <aXref>
    <sForeignId>1</sForeignId>
    <nRMClassId>1</nRMClassId>
    <sRMPUID>MRKT_000003</sRMPUID>
    <sForeignToolName>TEST</sForeignToolName>
  </aXref>
</urn:AddSyncXrefLink>
```

AppendRMDocumentEx

Append a Dimension RM document to an existing document with possibility to use existing chapters and requirements.

Style

Document

Input (Literal)

The input of this method is the argument AppendRMDocumentEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
id	int	1..1	The document id
rtmDoc	RtmDocument	1..1	Content of the document

Output (Literal)

The output of this method is the argument AppendRMDocumentExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	AppendRMDocumentExResult	1..1	Result of appending document

Remarks

The AppendRMDocumentEx call provides a method for append an existing document with the specified document content and provide possibility to include existing requirements.

Note: If new sub-requirements specified along with a pre-existing parent requirement, the sub-requirements would be attached to the pre-existing requirement. Even if you specify a different location in the document for the parent requirement, its current location would be used instead of creating a copy of the requirement in the specified location.

Example of usage:

```
<urn:AppendRMDocumentEx>
  <sessionId>1444320707</sessionId>
  <id>12</id>
  <rtmDoc>
    <sContent><![CDATA[<?xml version="1.0" ?>
      <Document name="ZVBob3RvIFJlcXVpcmVtZW50cw=="
projectName="Uk1ERU1P" databaseName="Uk0=" version="1.1"
application="RM Web Services">
      <Chapter title="RG9jdW1bnRhdGlvb1BSZXZF1aXJlbWVudHM="
uid="185" iid="45">
      <Description/>
      <Requirement classid="1" className="Marketing_Requirements"
```



```
<Attribute attrid="31" typeid="5"
title="Text">QmFzZTY0IGlzIGegZ2VuZXJpYyB0ZXJtIGZvciBhIG51bWJlciBvZiBza
W1pbGFyIGVuY29kaW5nIHNjaGVtZXMgdGhhdCB1bmNvZGUgYmluYXJ5IGRhdGEgYnkgdHJ
lYXRpbmcgaXQgbnVtZXJpY2FsbHkgYW5kIHRyYW5zbGF0aw5nIGl0IGludG8gYSBiYXNlI
DY0IHJlcHJlc2VudGF0aw9uLiA=</Attribute>
ID">TVJLVF8wMDAwMjk=</Attribute>
</Requirement>
</Chapter>
</Chapter>
</Document>]]</sContent>
</rtmDoc>
</urn:AppendRtmDocument>
```


ApproveChangeRequest

Approve the proposed change request

Style

Document

Input (Literal)

The input of this method is the argument ApproveChangeRequest having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
className	string	1..1	Specifies the class name
requestId	long	1..1	Specifies the object id of proposed requirement

Output (Literal)

The output of this method is the argument ApproveChangeRequestResult having the structure defined by the following table.

Name	Type	Occurs	Description
nId	long	1..1	Object id of approved requirement
status	CommandStatus	1..1	Status of operation

Remarks

The ApproveChangeRequest call provides a method for approve proposed changed for requirement by given class name and object id of proposed change request.

Example of usage:

```
<urn:ApproveChangeRequest>
  <sessionId>1111111111</sessionId>
  <className>Marketing_Requirements</className>
  <requestId>57</requestId>
</urn:ApproveChangeRequest>
```

AsyncAppendRtmDoc

Append an rtm document to an existing document.

Style

Document

Input (Literal)

The input of this method is the argument AsyncAppendRtmDoc having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
id	int	1..1	
sContent	string	1..1	

Output (Literal)

The output of this method is the argument AsyncAppendRtmDocResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	AsyncAppendRtmDocResponse	1..1	

AsyncCreateRtmDoc

Create a new project rtm document.

Style

Document

Input (Literal)

The input of this method is the argument AsyncCreateRtmDoc having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
rtmDoc	RtmDocument	1..1	

Output (Literal)

The output of this method is the argument AsyncCreateRtmDocResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	AsyncCreateRtmDocResponse	1..1	

AsyncQueryRtmDoc

Creates an XML representing RM Document structure.

Style

Document

Input (Literal)

The input of this method is the argument AsyncQueryRtmDoc having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocId	int	1..1	
nChapterId	int	0..1	
bIncludeReq	boolean	0..1	

Output (Literal)

The output of this method is the argument getAsyncQueryRtmDocResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	getAsyncQueryRtmDocResponse	1..1	

AsyncReplaceRtmDoc

Replace an existing project rtm document.

Style

Document

Input (Literal)

The input of this method is the argument AsyncReplaceRtmDoc having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
id	int	1..1	
sContent	string	1..1	

Output (Literal)

The output of this method is the argument AsyncReplaceRtmDocResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	AsyncReplaceRtmDocResponse	1..1	

ChangeRequest

Create Change Request RM requirement with specified requirement data

Style

Document

Input (Literal)

The input of this method is the argument ChangeRequest having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	
anECPsInfo	ECPsObjectInfo	0..1	
inChangeReason	string	1..1	

Output (Literal)

The output of this method is the argument ChangeRequestResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

CollectionAssociationWithCMPProject

Associate or un-associate RM collection/baseline with CM project/stream

Style

Document

Input (Literal)

The input of this method is the argument CollectionAssociationWithCMPProject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sCollectionName	string	1..1	
projectCM	CMPProjectInfo	1..1	
bAssociate	boolean	0..1	

Output (Literal)

The output of this method is the argument CollectionAssociationWithCMPProjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
sResult	string	1..1	

Remarks

When bAssociate parameter is set to "true", then web service will work in Associate mode, if set to "false" then web service will work in Un-associate mode.

Example of usage:

```
<urn:CollectionAssociationWithCMPProject>
  <sessionId>1111111111</sessionId>
  <sCollectionName>test_col</sCollectionName>
  <projectCM>
    <sProductName>QLARIUS</sProductName>
    <sProjectName>JAVA_BRANCHA_PRJ</sProjectName>
  </projectCM>
  <bAssociate>true</bAssociate>
</urn:CollectionAssociationWithCMPProject>
```

CreateBaseline

Create a baseline with given requirements and associate it to a CM baseline

Style

Document

Input (Literal)

The input of this method is the argument CreateBaseline having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sName	string	1..1	
sCategoryFullName	string	0..1	
nCmSourceBaselineUid	int	1..1	
requirements	RequirementIdentifierList	1..1	

Output (Literal)

The output of this method is the argument CreateBaselineResult having the structure defined by the following table.

Name	Type	Occurs	Description
nId	int	1..1	
nStatus	int	1..1	
sUrl	string	1..1	

Remarks

Additional info:

Category name can be specified on baseline creation.

'sCategoryFullName' value can be specified as:

- a** category full name, separated by '/' or '\'
- b** category name(only in case when name is unique);

Example of usage:

```
<urn:CreateBaseline>
  <sessionId>1</sessionId>
  <sName>test baseline</sName>
  <sCategoryFullName>RMDEMO/Functional</sCategoryFullName>
  <requirements>
    <item>
      <sPUID>MRKT_000003</sPUID>
      <sClassName>Marketing_Requirements</sClassName>
    </item>
  </requirements>
</urn:CreateBaseline>
```



```
</item>
<item>
  <sPUID>MRKT_000004</sPUID>
  <sClassName>Marketing_Requirements</sClassName>
</item>
<item>
  <sPUID>MRKT_000021</sPUID>
  <sClassName>Marketing_Requirements</sClassName>
</item>
</requirements>
</urn:CreateBaseline>
```

CreateCategory

Creates a RM category, given a category name, description and parent category id.

Style

Document

Input (Literal)

The input of this method is the argument CreateCategory having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
CategoryReq	CategoryReq	1..1	

Output (Literal)

The output of this method is the argument createCategoryResponse having the structure defined by the following table.

Name	Type	Occurs	Description
item	Category	1..1	

CreateChildDocument

Create a new Child document of the specified Parent document.

Style

Document

Input (Literal)

The input of this method is the argument CreateChildDocument having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
rtmDoc	CreateChildDocumentData	1..1	Child document info

Output (Literal)

The output of this method is the argument CreateChildDocumentResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CreateRtmDocumentResult	1..1	Result with info about created document

Remarks

The CreateChildDocument call provides a method for creating new Child document based on specified Parent document.

Created Child doc will inherit all structure of the Parent doc.

Example of usage:

```
<urn:CreateChildDocument>
  <sessionId>1111111111</sessionId>
  <rtmDoc>
    <sName>Child doc</sName>
    <nMasterDocId>18</nMasterDocId>
  </rtmDoc>
</urn:CreateChildDocument>
```

CreateCollection

Creates an empty collections.

Style

Document

Input (Literal)

The input of this method is the argument CreateCollection having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
collectionName	CreateCollectionEx	1..1	
sCategoryFullName	string	0..1	

Output (Literal)

The output of this method is the argument CreateCollectionResponseResult having the structure defined by the following table.

Name	Type	Occurs	Description
nId	long	1..1	
sName	string	1..1	
sUrl	string	1..1	

Remarks

Additional info:

Category name can be specified on collection creation.

'sCategoryFullName' value can be specified as:

- a** category full name, separated by '/' or '\'
- b** category name(only in case when name is unique);

Example of usage:

```
<urn:CreateCollection>
  <sessionId>1111111111</sessionId>
  <collectionName>
    <sName>testCol21</sName>
  </collectionName>
  <sCategoryFullName>RMDEMO\Availability</sCategoryFullName>
</urn:CreateCollection>
```

CreateObject

Creates a new requirement, based on given class name and requirement data

Style

Document

Input (Literal)

The input of this method is the argument CreateObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
anObject	Requirement	1..1	A set of object attributes values

Output (Literal)

The output of this method is the argument CreateObjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	A set of object attributes values

Example of usage:

```
<urn:CreateObject>
  <sessionId>1478719333</sessionId>
  <anObject>
    <sClassName>Tests</sClassName>
    <sDescription>Object description</sDescription>
    <sTitle>Test name</sTitle>
    <aAttrList>
      <item>
        <sName>SUBJECT</sName>
        <sValue>Subject</sValue>
      </item>
      <item>
        <sName>URL</sName>
        <sValue>&lt;a href=&quot;https://www.opentext.com/
&quot; target=&quot;_blank&quot;&gt;Open Text&lt;/a&gt;&lt;a
href=&quot;https://www.opentext.com/en-us/products/dimensions-rm/
overview&quot; target=&quot;_blank&quot;&gt;Dimensions RM&lt;/a&gt;</
sValue>
      </item>
    </aAttrList>
  </anObject>
</urn:CreateObject>
```

CreateObjectEx

Creates a new requirement, based on given class name and requirement data

Style

Document

Input (Literal)

The input of this method is the argument CreateObjectEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
anObject	Requirement	1..1	A set of object attributes values
bDontGenerateAlfEvent	boolean	0..1	A flag indicating whether to generate an alf event

Output (Literal)

The output of this method is the argument CreateObjectExResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	A set of object attributes values

Example of usage:

```
<urn:CreateObjectEx>
  <sessionId>1478719333</sessionId>
  <anObject>
    <sClassName>Tests</sClassName>
    <sDescription>Object description</sDescription>
    <sTitle>Test name</sTitle>
    <aAttrList>
      <item>
        <sName>EXECUTION_STATUS</sName>
        <sValue>Failed</sValue>
      </item>
    </aAttrList>
  </anObject>
  <bDontGenerateAlfEvent>>false</bDontGenerateAlfEvent>
</urn:CreateObjectEx>
```

CreateObjectWithAttachment

Creates a new requirement with attachment, based on given class id and requirement data

Style

Document

Input (Literal)

The input of this method is the argument CreateObjectWithAttachment having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
anObject	Requirement	1..1	A set of object attributes values
sAttrName	string	0..1	File attachment attribute name
oFile	FileAttachment	0..1	File attachment details and contents

Output (Literal)

The output of this method is the argument CreateObjectWithAttachmentResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	A set of object attributes values

Example of usage:

```
<urn>CreateObjectWithAttachment>
  <sessionId>1479156038</sessionId>
  <anObject>
    <sClassName>Marketing_Requirements</sClassName>
    <aAttrList>
      <item>
        <sName>TITLE</sName>
        <sValue>Title</sValue>
      </item>
      <item>
        <sName>TEXT</sName>
        <sValue>Description of requirement</sValue>
      </item>
    </aAttrList>
  </anObject>
  <sAttrName>FILE_ATTACHMENT</sAttrName>
  <oFile>
    <sFullPath>c:\:examples</sFullPath>
```

```
<sFilename>doc</sFilename>  
  <encodedContents>VGvzdCBkb2N1bWVudA==</encodedContents>  
</oFile>  
</urn:CreateObjectWithAttachment>
```


CreateProduct

Create a new Product with the option to define if a corresponding category should be created

Style

Document

Input (Literal)

The input of this method is the document element CreateProduct having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
anObject	Requirement	1..1	A set of product attributes values
bCreateCategory	boolean	1..1	A flag indicating whether to create Category

Output (Literal)

The output of this method is the document element CreateObjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	A set of Product attributes values

Example of usage:

```
<urn:CreateProduct>
  <sessionId></sessionId>
  <anObject>
    <sTitle>General Product</sTitle>
    <aAttrList>
      <item>
        <sName>SHORT_NAME</sName>
        <sValue>PRD1</sValue>
      </item>
    </aAttrList>
  </anObject>
  <bCreateCategory>>false</bCreateCategory>
</urn:CreateProduct>
```



```

U1F1IjsgdHlwZT0idGV4dC9odG1sIgoKLS0tLS0tLS0tLS0tLS1Cb3VuZGFyeS0wMD1fUj
g2RmVRelZQc1N2ZVVHN2dTUXUKQ29udGVudC1UeXB10iB0ZXh0L2h0bWwKQ29udGVudC1U
cmFuc2Zlci1FbmNvZGluZzogYmFzZTY0CgpWR2hsSUdWUWFHOTBieUJ6ZVhOMFpXMGdjMm
hoYkd3Z2MzVndjRz15ZENCd2FHOTBiM01nZEdoaGRDQmhjbVVnUVdSM1lXNTAKYVhnZ0tE
TWdjMmw2WlhNcElHWnZjbTFoZENCdmNtbG5hVzVoYkhNZ2QybDBhQ0J1YjNRZ2JHVnpjeU
IwYUdGdU1ESTF0aUJqCmIyeHZjbk11SUV0dmJuTnBaR1Z5WVhScGIyNGdjMmh2ZFd4a0lH
SmxJR2RwZG1WdU1IUnZJSFJvWlNCM11Ya2dkR2hoZENCcwpZWEpuWlhJZ2NISnBib1J6SU
cxaGVTQmlaU0JoYzNOcGJXbHNZWfJswkNCaGJtUwdZU0J6YjJ4MWRHbHZiaUJ6YUdGc2JD
QmkKw1NCamFH0XpaVzRnYVc0Z2NHRnlkQ0JpWVhObFpDQjFjRz11SUh0MVkyZ2dabXhsZU
dsaWFXeHBkSGt1CgotLS0tLS0tLS0tLS0tLUJvdW5kYXJ5LTAWPV9SODZGZVF6VlBzU3Zl
VUc3Z1NRdS0tCg==</Attribute>
    <Attribute attrid="26" typeid="12" title="Rqmt ID">
TVJLVF8wMDAwMDI=</Attribute>
    </Requirement>
    </Chapter>
    </Document>]]</sContent>
</rtmDoc>
</urn:CreateRMDocumentEx>

```

The method allows to create document as a Parent document if provide `bIsMaster` as true in the input param of request.

```

<urn:CreateRMDocumentEx>
  <sessionId>1111111111</sessionId>
  <rtmDoc>
    <info>
      <sName>Test Specification Parent</sName>
    </info>
    <sContent><![CDATA[<?xml version="1.0" ?>
  <Document name="ZVBob3RvIFJlcXVpcmVtZW50cw=="
projectName="Uk1ERU1P" databaseName="Uk0=" version="1.1"
application="RM Web Services">
  <Chapter title="RGVzaWduIFJlcXVpcmVtZW50cw==" uid="169" iid="44">
    <Description/>
    <Requirement classid="1" className="Marketing_Requirements"
title="" category="5" uid="52" iid="22">
      <Attribute attrid="3011" typeid="1" title="Title">
U3VwG9ydCBZHZhbnRpeCBmb3JtYXRz</Attribute>
      <Attribute attrid="31" typeid="5" title="Text">
TU1NRS1WZXJzaW9u0iAxLjAKQ29udGVudC1UeXB10iBtdWx0aXBhcnQvcmlsYXRlZDsKIC
Bib3VuZGFyeT0iLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS1Cb3VuZGFyeS0wMD1fUj
g2RmVRelZQc1N2ZVVHN2dTUXUKQ29udGVudC1UeXB10iB0ZXh0L2h0bWwKQ29udGVudC1U
cmFuc2Zlci1FbmNvZGluZzogYmFzZTY0CgpWR2hsSUdWUWFHOTBieUJ6ZVhOMFpXMGdjMm

```

```
hoYkd3Z2MzVndjRz15ZENCd2FH0TBiM01nZEdoaGRDQmhjbVvNuvdSM11XNTAKYVhnZ0tE
TwdjMmw2WlhNcElHwnZjbTFoZENCdmNtbG5hVzVoYkhNZ2QybDBhQ0J1YjNRZ2JHVnpjeU
IwYUdGdU1ESTF0aUJqCmIyeHZjBk11SUV0dmJuTnBaR1Z5WVhScGIyNGdjMmh2ZFd4a01H
SmxJR2RwZG1WdU1IUnZJSFJvWlNCM11Ya2dkR2hoZENCcwpZWEpuWlhJZ2NISnBib1J6SU
cxaGVTQmlaU0JoYzN0cGJXbHNZWfJswkNCaGJtUwdZU0J6YjJ4MWRHbHZiaUJ6YUdGc2JD
QmkKwLNCamFH0XpaVzRnYVc0Z2NHRn1kQ0JpWVhObFpDQjFjRz11SUh0MVkyZ2dabXhsZU
dsawFXeHBKSGt1CgotLS0tLS0tLS0tLS0tLUJvdW5kYXJ5LTAwPV9S0DZGZVF6V1BzU3Z1
VUc3Z1NRdS0tCg==</Attribute>
  <Attribute attrid="26" typeid="12" title="Rqmt ID">
TVJLVF8wMDAwMDI=</Attribute>
  </Requirement>
</Chapter>
</Document>]]</sContent>
  <bIsMaster>>true</bIsMaster>
  </rtmDoc>
</urn:CreateRMDocumentEx>
```

Example of date attribute tag for creating requirements with filled date attribute:

```
<Attribute attrid="3389" typeid="3" title="date4">
MTQ4MTM0NjMwNQ==</Attribute>
```

where:

attrid - id of the date attribute

typeid - type id, for date attribute it always 3

title - attribute name, but can be specified as empty.

Inner text of the attribute tag (MTQ4MTM0NjMwNQ== in this example) must be the base64 encoded date which has to be in the same format as specified for this attribute in Class Definition.

CreateRtmDocSnapshot

Creates document snapshot by document id.

Style

Document

Input (Literal)

The input of this method is the argument CreateRtmDocSnapshot having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
snapshot	RtmDocSnapshot	1..1	A set of snapshot properties

Output (Literal)

The output of this method is the argument CreateRtmDocSnapshotResult having the structure defined by the following table.

Name	Type	Occurs	Description
snapshot	RtmDocSnapshotOut	1..1	A set of snapshot properties

When creating a new snapshot in the web service was added snapshot description and choice its type (MINOR|MAJOR|NONE).

- MINOR:** When you enter MINOR type it is means that the snapshot name will be consist of document name and two numbers separated by a point, regardless of which name was entered . First number is current amount major snapshots. Second number is current amount minor snapshots.
- MAJOR:** When you enter MAJOR type it is means that the snapshot name will be consist of document name and two numbers separated by a point, regardless of which name was entered. First number is current amount major snapshots. Second number always will be zero.
- NONE:** When you enter NONE type it is means that the snapshot name will be the same as entered name.

Example of usage:

```
<urn:CreateRtmDocSnapshot>
  <sessionId>1478719333</sessionId>
  <snapshot>
    <sName>SNAPSHOT</sName>
    <nDocumentId>12</nDocumentId>
    <sDescription>Provide some description there</sDescription>
    <sType>NONE</sType>
  </snapshot>
</urn:CreateRtmDocSnapshot>
```

CreateRtmDocument

Creates a new document with given content.

Style

Document

Input (Literal)

The input of this method is the argument CreateRtmDocument having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
rtmDoc	RtmDocumentWithCategory	1..1	A set of document properties

Output (Literal)

The output of this method is the argument CreateRtmDocumentResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CreateRtmDocumentResult	1..1	A set of document properties

Remarks

The CreateRtmDocument call provides a method for creating document with specified document content

Example of usage:

```
<urn:CreateRtmDocument>
  <sessionId>1111111111</sessionId>
  <rtmDoc>
    <info>
      <sName>Test Specification</sName>
    </info>
    <sContent><![CDATA[<?xml version="1.0" ?>
      <Document name="ZVBob3RvIFJlcXVpcmVtZW50cw=="
projectName="Uk1ERU1P" databaseName="Uk0=" version="1.1"
application="RM Web Services">
      <Chapter title="RGVzaWduIFJlcXVpcmVtZW50cw==" uid="169"
iid="44">
        <Description/>
        <Requirement classid="1" className="Marketing_Requirements"
title="" category="5" uid="52" iid="22">
          <Attribute attrid="3011" typeid="1"
title="Title">U3VwcG9ydCBBZHZhbnRpeCBmb3JtYXRz</Attribute>
          <Attribute attrid="31" typeid="5" title="Text">
TU1NRS1WZXJzaW9uOjAxLjAKQ29udGVudC1UeXB1OjBtdWx0aXBhcnQvcmlzZXI=
</Attribute>
        </Requirement>
      </Chapter>
    </sContent>
  </rtmDoc>
</urn:CreateRtmDocument>
```



```
g2RmVRe1ZQc1N2ZVVHN2dTUXUKQ29udGVudC1UeXB10iB0ZXh0L2h0bWwKQ29udGVudC1U
cmFuc2Z1ci1FbmNvZGluZzogYmFzZTY0CgpWR2hsSudWUWFH0TBieUJ6ZVh0MFpXMGdjMm
hoYkd3Z2MzVndjRz15ZENCd2FH0TBiM01nZEdoaGRDQmhjbVvVnUVdSM1lXNTAKYVhnZ0tE
TWdjMmw2WlhNcElHwnZjbTFoZENCdmNtbG5hVzVoYkhNZ2QybDBhQ0J1YjNRZ2JHVnpjeU
IwYUdGdU1ESTF0aUJqCmIyeHZjbk11SUV0dmJuTnBaR1Z5WVhScGIyNGdjMmh2ZFd4a0lH
SmxJR2RwZG1WdU1IUnZJSFJvWlNCM1lYa2dkR2hoZENCcwpZWepuWlhJZ2NISnBib1J6SU
cxaGVTQmlaU0JoYzN0cGJXbHNZWJswkNCaGJtUWdZU0J6YjJ4MWRHbHZiaUJ6YUdGc2JD
QmkKwLNCamFH0XpaVzRnYVc0Z2NHRnlkQ0JpWVhObFpDQjFjRz11SUh0MVkyZ2dabXhsZU
dsawFXeHBkSGt1CgotLS0tLS0tLS0tLS0tLUJvdW5kYXJ5LTAwPV9SODZGZVF6VlBzU3Zl
VUc3Z1NRdS0tCg==</Attribute>
    <Attribute attrid="26" typeid="12" title="Rqmt ID">
TVJLVF8wMDAwMDI=</Attribute>
    </Requirement>
    </Chapter>
    </Document>]]></sContent>
    <bIsMaster>true</bIsMaster>
  </rtmDoc>
</urn:CreateRtmDocument>
```

Example of date attribute tag for creating requirements with filled date attribute:

```
<Attribute attrid="3389" typeid="3" title="date4">
MTQ4MTM0NjMwNQ==</Attribute>
```

where:

attrid - id of the date attribute

typeid - type id, for date attribute it always 3

title - attribute name, but can be specified as empty.

Inner text of the attribute tag (MTQ4MTM0NjMwNQ== in this example) must be the base64 encoded date which has to be in the same format as specified for this attribute in Class Definition.

CreateTemplate

Creates a new project template for importing document using RM Import.

Style

Document

Input (Literal)

The input of this method is the argument CreateTemplate having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
docTemplate	Template	1..1	A set of template properties
encodedContent	base64Binary	0..	Content of template in xml format encoded to Base64Binary

Output (Literal)

The output of this method is the argument CreateTemplateResponse having the structure defined by the following table.

Name	Type	Occurs	Description
returnTemplate	Template	1..1	A set of template properties

Example of usage:

```
<urn:CreateTemplate>
  <sessionId>1478777334</sessionId>
  <docTemplate>
    <sName>TEM511</sName>
  </docTemplate>
<encodedContent>PD94bWwgdMvYc2lvcj0iMS4wIj8+DQoNCi08SW1wb3J0VGVCtGxhdG
UgZGVmYXVsdENsYXNzPSItMSIgdMvYc2lvcj0iMSIgaWQ9IjEiPg0KDQoNCi08R2VuZXJh
bFJ1bGUgaWQ9ImdlbmVvYyYwWiPg0KDQo8QXV0aG9yPnJtZGVtb2FkbWluPC9BdXR0b3I+DQ
oNCjxYXR1PjEwLzE4LzIwMDcgNjoxMj0MSBQTtwvRGF0ZT4NCg0KPERlc2NyaXB0aW9u
PlRoaxMgdGVtcGxhdGUgbGV0cyB5b3UgaW1wb3J0IGV29yZCBkb2N1bWVudCBpbmRvIE
RpbWVuc2lvcnMgUk0gZG12aWRlZCBpbjBjaGFwdGVycy4gUmVmZXIgdG8gdGh1IFNlcmVu
YSBEaW11bnNpb25zIFJNIEFkbWluaXN0cmF0b3Inc3IyBhdWlkZSBmb3IgbW9yZSBpbmZvcn
1hdGlvbiBhYm91dCBjcmVhdGluZyB2YXJpb3VzIHR5cGVzIG9mIHRlbXBsYXRlc3I4L0Rl
c2NyaXB0aW9uPg0KDQo8L0dlbmVvYyYwSdWxLPg0KDQoNCi08Q2hhcHRlc3I1bGUgaWQ9Im
NoYXB0ZXIiPg0KDQoNCi08SGVhZGluZ1J1bGUgaWQ9Imh1YWVpbmciIGhpbmQ9ImFsbCI+
DQoNCg0KLTxiZWZkaW5ncz4NCg0KPEh1YWRpbmci+U3RhbmRhc3IyZCBIZWFkaW5ncz
wvSGVhZGluZz4NCg0KPC9IZWFkaW5ncz4NCg0KPC9IZWFkaW5nUnVsZT4NCg0KPC9DaGFw
dGVyUnVsZT4NCg0KDQotPFJlcXVpcmVtZ50UnVsZSBpZD0icmVxdWlyZW11bnQiPg0KDQ
oNCi08TGF5b3V0UnVsZSBpZD0idGFibGVMYXlvdXQiIGlnbm9yZT0iZmFsc2UiPg0KDQoN
Ci08SGVhZGluZ1J1bGUgaWQ9Imh1YWVpbmciIGhpbmQ9ImFsbCI+DQoNCg0KLTxiZWZkaW
5ncz4NCg0KPEh1YWRpbmci+SGVhZGluZyAyPC9IZWFkaW5nPg0KDQo8SGVhZGluZz5IZWFk
```

```
aW5nIDM8L0h1YWRpbmc+DQoNCjxIZWFkaW5nPkhlYWRpbmcgNDwvSGVhZGluZz4NCg0KPE
h1YWRpbmc+SGVhZGluZyA1PC9IZWFkaW5nPg0KDQo8SGVhZGluZz5IZWFkaW5nIDY8L0h1
YWRpbmc+DQoNCjxIZWFkaW5nPkhlYWRpbmcgNzwwSGVhZGluZz4NCg0KPEh1YWRpbmc+SG
VhZGluZyA4PC9IZWFkaW5nPg0KDQo8SGVhZGluZz5IZWFkaW5nIDk8L0h1YWRpbmc+DQoN
CjwwSGVhZGluZz3M+DQoNCjwwSGVhZGluZz1J1bGU+DQoNCjwwTGZ5b3V0UnVsZT4NCg0KPC
9SZXF1aXJlbWVudFJ1bGU+DQoNCjwwSW1wb3J0VGltcGxhdGU+</encodedContent>
</urn:CreateTemplate>
```

DeleteBaseline

Deletes a baseline.

Style

Document

Input (Literal)

The input of this method is the argument DeleteBaseline having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
nBaselineId	int	1..1	Identifier of the baseline to be deleted

Output (Literal)

The output of this method is the argument DeleteBaselineResult having the structure defined by the following table.

Name	Type	Occurs	Description
nStatus	int	1..1	Status of operation (0 – success, 1 – failure)

Example of usage:

```
<urn:DeleteBaseline>  
  <sessionId>1478715622</sessionId>  
  <nBaselineId>11</nBaselineId>  
</urn:DeleteBaseline>
```

DeleteObjects

Marks requirements as deleted.

Style

Document

Input (Literal)

The input of this method is the argument DeleteObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Class name of objects to be deleted
aObjectPUIDs	StringList	1..1	Unique object identifiers of objects to be deleted

Output (Literal)

The output of this method is the argument DeleteObjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	boolean	1..1	A flag indicating whether the deletion was successful

Example of usage:

```
<urn:DeleteObjects>
  <sessionId>1478715622</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <aObjectPUIDs>
    <item>MRKT_000004</item>
    <item>MRKT_000007</item>
  </aObjectPUIDs>
</urn:DeleteObjects>
```

DeleteRtmDocument

Deletes document specified by document id.

Style

Document

Input (Literal)

The input of this method is the argument DeleteRtmDocument having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
id	int	1..1	Identifier of document to be deleted

Output (Literal)

The output of this method is the argument DeleteRtmDocumentResponse having the structure defined by the following table.

Name	Type	Occurs	Description
status	CommandStatus	1..1	Status of operation

Example of usage:

```
<urn>DeleteRtmDocument>  
  <sessionId>1478777334</sessionId>  
  <id>22</id>  
</urn>DeleteRtmDocument>
```

DeleteSyncXrefLink

Deletes a link from Sync_Xref table.

Style

Document

Input (Literal)

The input of this method is the argument DeleteSyncXrefLink having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
aXref	SyncXrefLink	1..1	

Output (Literal)

The output of this method is the argument DeleteSyncXrefLinkResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	boolean	1..1	

DeleteTemplate

Deletes the project template specified by template id.

Style

Document

Input (Literal)

The input of this method is the argument DeleteTemplate having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
id	int	1..1	Identifier of template to be deleted

Output (Literal)

The output of this method is the argument DeleteTemplateResponse having the structure defined by the following table.

Name	Type	Occurs	Description
returnTemplate	Template	1..1	A set of template properties

Example of usage:

```
<urn:DeleteTemplate>  
  <sessionId>1478777334</sessionId>  
  <id>25</id>  
</urn:DeleteTemplate>
```

DocumentAddChapter

Adds new chapter to the document.

Style

Document

Input (Literal)

The input of this method is the argument DocumentAddChapter having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
documentId	int	0..1	Specified document identifier
documentName	string	0..1	Specified document name
parentIID	int	0..1	Specifies internal id of parent object
chapterName	ns-chapterDetails	1..1	Name and description of chapter to be created
position	int	0..1	Position of added object (by default added at the end of parent, if less or equal 1 then add at the top of parent)

Output (Literal)

The output of this method is the argument DocumentAddChapterResults having the structure defined by the following table.

Name	Type	Occurs	Description
sResult	string	1..1	Result message
Chapter	DocumentObject	0..1	A set of chapter properties

Example of usage:

```
<urn:DocumentAddChapter>
  <sessionId>1478800111</sessionId>
  <documentId>23</documentId>
  <documentName>DOCUMENT</documentName>
  <parentIID>222</parentIID>
  <chapterName>
    <sName>chapter</sName>
  </chapterName>
</urn:DocumentAddChapter>
```


DocumentMoveObject

Moves chapter or requirement in the document.

Style

Document

Input (Literal)

The input of this method is the argument DocumentMoveObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	Specifies session id for connection
documentId	int	0..1	Specifies document id
documentName	string	0..1	Specifies document name
IID	int	1..1	Specifies internal id of the object
parentIID	int	1..1	Specifies internal id of parent object
position	int	0..1	Position of moved object (by default added at the end of parent, if less or equal 1 then add at the top of parent).

Output (Literal)

The output of this method is the argument DocumentMoveObjectResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	string	1..1	Result of moving object

Remarks

The DocumentMoveObject call provides a method to move objects in the document structure.

Document name or id can be used for providing details about document; iid of the objects could be taken from the QueryRtmDocumentStructure web service response.

Example of usage:

```
<urn:DocumentMoveObject>
  <sessionId>111111111</sessionId>
  <documentId>17</documentId>
  <IID>86</IID>
  <parentIID>82</parentIID>
  <position>1</position>
</urn:DocumentMoveObject>
```

DocumentRemoveObject

Marks objects as removed from the Document.

Style

Document

Input (Literal)

The input of this method is the argument DocumentRemoveObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	Specifies session id for connection
documentId	int	0..1	Specifies document id
documentName	string	0..1	Specifies document name
IID	int	1..1	Specifies internal id of the object

Output (Literal)

The output of this method is the argument DocumentRemoveObjectResponse having the structure defined by the following table.

Name	Type	Occurs	Description
sMessage	string	1..1	Result of removing object from document

Remarks

The DocumentRemoveObject call provides a method to remove objects in the document structure.

Document name or id could be used for providing details about document (if documentId and documentName parameters are set, documentId has higher priority.); iid of the object could be taken from the QueryRtmDocumentStructure web service response.

Example of usage:

```
<urn:DocumentRemoveObject>
  <sessionId>1111111111</sessionId>
  <documentId>17</documentId>
  <IID>86</IID>
</urn:DocumentRemoveObject>
```

FindAndReplaceObject

Searches requirement(s) based on attribute list and replaces those with specified requirement data

Style

Document

Input (Literal)

The input of this method is the argument FindAndReplaceObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	
aAttrList	NameValueList	0..1	
bCreateIfnotFound	boolean	0..1	

Output (Literal)

The output of this method is the argument FindAndReplaceObjectResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	ReplaceRequirementList	1..1	

FindLinkedObjects

Returns links and requirements linked to specified requirement

Style

Document

Input (Literal)

The input of this method is the argument FindLinkedObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Name of specified requirement class
nObjectId	int	1..1	Unique identifier of requirement
sStatus	string	0..1	Current status of requirement
aAttrList	StringList	0..1	A set of requirement attributes
bIncludeDeleted	boolean	0..1	A flag indicating whether to show requirements with deleted links

Output (Literal)

The output of this method is the argument LinkListEx having the structure defined by the following table.

Name	Type	Occurs	Description
sourceObjDetails	ObjectDetails	1..1	Source requirement details
relations	RelationList	0..1	List of linked requirements by each relationship with their details
nRelationType	int	1..1	Returns link direction 1 - primary 2 - secondary

Remarks

Additional info:

It is possible to return deleted links in the result list. If bIncludeDeleted value is set to true then all related requirements will be returned. Each returned item has status of link (current/deleted).

If bIncludeDeleted it set to false, or not specified, only current links will be returned.

Example of usage:

```
<urn:FindLinkedObjects>
  <sessionId>1436973992</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <nObjectId>1</nObjectId>
```

```
<bIncludeDeleted>true</bIncludeDeleted>  
</urn:FindLinkedObjects>
```

FindModifiedObjects

Find objects that have been modified since the given time\date

Style

Document

Input (Literal)

The input of this method is the argument FindModifiedObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Name of specified requirement class
nTimestamp	dateTime	1..1	Specified time and date
aObjectPUIDs	StringList	0..1	A set of requirements PUIDs, that can be used as search criteria
aAttrList	StringList	0..1	A set of requirement attributes, that can be used as search criteria

Output (Literal)

The output of this method is the argument FindModifiedObjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	RequirementList	1..1	A list of requirements (and its properties), that have been modified since specified time and date

Example of usage:

```
<urn:FindModifiedObjects>
  <sessionId>1479138561</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <nTimestamp>2001-09-30T01:54:23-07:00</nTimestamp>
</urn:FindModifiedObjects>
```

FindObject

Find objects that match the given search criteria.

Style

Document

Input (Literal)

The input of this method is the argument FindObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Name of class, used as search criteria (only requirement from specified class will be returned)
aKeywords	StringList	0..1	Keywords that can be used as search criteria
bSearchFullText	boolean	0..1	A flag indicating whether to search a keyword as a full text or not (partial match)
bIncludeSubCategories	boolean	0..1	A flag indicating whether to search within sub categories or not
sAdditionalConditions	string	0..1	Condition that can be used as search criteria
anExample	Requirement	0..1	A set of requirements attributes and its values that can be used as search criteria
aAttrList	StringList	0..1	Attributes to be shown in search results

Output (Literal)

The output of this method is the argument FindObjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementList	1..1	A list of requirements (that satisfy search criteria) and its details

Remarks

Additional info:

For looking for requirements in some categories, IN_CATEGORY attribute can be used.

To return requirements in subcategories set bIncludeSubCategories value as "true". If set bIncludeSubCategories value as "false" then requirements from specified category only will be returned.

Requirements could be filtered by condition, using sAdditionalConditions parameter.

Example of usage:

```
<urn:FindObject>
  <sessionId>1</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <bIncludeSubCategories>true</bIncludeSubCategories>
  <sAdditionalConditions>DELIVERY_PHASE IN ('Build1') and STATUS IN
('Replaced')</sAdditionalConditions>
  <anExample>
    <aAttrList>
      <item>
        <sName>IN_CATEGORY</sName>
        <sValue>Availability</sValue>
      </item>
      <item>
        <sName>IN_CATEGORY</sName>
        <sValue>Functional</sValue>
      </item>
    </aAttrList>
  </anExample>
  <aAttrList>
    <item>STATUS</item>
    <item>DELIVERY_PHASE</item>
    <item>IN_CATEGORY</item>
  </aAttrList>
</urn:FindObject>
```

Other example of "sAdditionalConditions" param value:

```
"STATUS IN ('Current','Replaced') and group not in ('User') and primary_in
MarketingRqmts_to_ProdRqmts"
```

As FindObjects uses doctool engine for filtering the full syntax from RM reports ("where" clause) is supported in this parameter.

sValue for IN_CATEGORY can be specified as:

- a** category id;
- b** category full name, separated by '/' or '\'
- c** category name(only in case when name is unique);

FindObjectInClasses

Find objects in classes that match the given search criteria.

Style

Document

Input (Literal)

The input of this method is the argument FindObjectsInClasses having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
aClassNamesList	StringList	0..1	Names of classes, used as search criteria (only requirement from specified classes will be returned)
aKeywords	StringList	0..1	Keywords that can be used as search criteria
bSearchFullText	boolean	0..1	A flag indicating whether to search a keyword as a full text or not (partial match)
bIncludeSubCategories	boolean	0..1	A flag indicating whether to search within sub categories or not
sAdditionalConditions	string	0..1	Condition that can be used as search criteria
anExample	Requirement	0..1	A set of requirements attributes and its values that can be used as search criteria
aAttrList	StringList	0..1	Attributes to be shown in search results

Output (Literal)

The output of this method is the argument FindObjectsInClassesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementList	1..1	A list of requirements (that satisfy search criteria) and its details

Remarks

Additional info:

For looking for requirements in some categories, IN_CATEGORY attribute can be used.

To return requirements in subcategories set bIncludeSubCategories value as "true". If set bIncludeSubCategories value as "false" then requirements from specified category only will be returned.

Example of usage:

```
<urn:FindObjectsInClasses>
  <sessionId>1</sessionId>
  <bIncludeSubCategories>true</bIncludeSubCategories>
  <anExample>
    <aAttrList>
      <item>
        <sName>IN_CATEGORY</sName>
        <sValue>1</sValue>
      </item>
    </aAttrList>
  </anExample>
  <aAttrList>
    <item>IN_CATEGORY</item>
  </aAttrList>
</urn:FindObjectsInClasses>
```

sValue for IN_CATEGORY can be:

- a** category id;
- b** category full name, separated by '/' or '\'
- c** category name(only in case when name is unique);

FindObjectInCollections

Find objects in the given collections.

Style

Document

Input (Literal)

The input of this method is the argument FindObjectsInCollections having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
collectionIds	IdList	1..1	Unique identifiers of collections, used as search criteria (only requirement from specified collections will be returned)
aSearchAttr	SearchRequirement	0..1	A set of requirements attributes and its values that can be used as search criteria
aKeywords	StringList	0..1	Keywords that can be used as search criteria
bSearchFullText	boolean	0..1	A flag indicating whether to search a keyword as a full text or not (partial match)
bDetagDescription	boolean	0..1	A flag indicating whether to detag description or not
aReturnAttrList	StringList	0..1	A set of attributes to be shown in search results

Output (Literal)

The output of this method is the argument FindObjectsInCollectionsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	RequirementFindList	1..1	A list of requirements from specified collection(s) and their attributes (specified in aReturnAttrList)

Remarks

Additional info:

RTM_KEYWORDs attributes now can be used in the output. Now it is split to 4 groups: Collections, Documents, Snapshots and Baselines.

Also next web services can use these attributes to be returned to display in results:

- FindModifiedObjects
- FindObjectsInClasses

- GetObjectsInCollection
- FindObjectsInCollections
- GetObjectsInCollections

Example of usage:

```
<urn:FindObjectsInCollections>
  <sessionId>1</sessionId>
  <collectionIds>
    <item>12</item>
  </collectionIds>
  <aSearchAttr/>
  <aKeywords/>
  <aReturnAttrList>
    <item>PUID</item>
    <item>RTM_KEYWORD</item>
    <item>RTM_DOCUMENTS</item>
    <item>RTM_COLLECTIONS</item>
    <item>RTM_BASELINES</item>
    <item>RTM_SNAPSHOTS</item>
    <item>RTM_LINKS_FROM</item>
    <item>RTM_LINKS_TO</item>
  </aReturnAttrList>
</urn:FindObjectsInCollections>
```

FindRtmDocument

Get a document specified by id.

Style

Document

Input (Literal)

The input of this method is the argument FindRtmDocument having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
id	int	1..1	Unique identifier of document

Output (Literal)

The output of this method is the argument FindRtmDocumentResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	RtmDocumentList	1..1	Document properties (Id, name and url to access document)

Example of usage:

```
<urn:FindRtmDocument>  
  <sessionId>1479216967</sessionId>  
  <id>12</id>  
</urn:FindRtmDocument>
```

FindTemplate

Get a document template specified by id.

Style

Document

Input (Literal)

The input of this method is the argument FindTemplate having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
id	int	1..1	Unique identifier of template

Output (Literal)

The output of this method is the argument FindTemplateResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	TemplateList	1..1	Template properties

Example of usage:

```
<urn:FindTemplate>  
  <sessionId>1479216967</sessionId>  
  <id>1</id>  
</urn:FindTemplate>
```

getAsyncAppendRtmDocStatus

Get status from the async. test commmand

Style

Document

Input (Literal)

The input of this method is the argument `getAsyncAppendRtmDocStatus` having the structure defined by the following table.

Name	Type	Occurs	Description
cmdToken	string	1..1	

Output (Literal)

The output of this method is the argument `getAsyncAppendRtmDocStatusResult` having the structure defined by the following table.

Name	Type	Occurs	Description
result	getAsyncAppendRtmDocStatusResponse	1..1	

getAsyncCreateRtmDocStatus

Get status from the async. test commmand

Style

Document

Input (Literal)

The input of this method is the argument getAsyncCreateRtmDocStatus having the structure defined by the following table.

Name	Type	Occurs	Description
cmdToken	string	1..1	

Output (Literal)

The output of this method is the argument getAsyncCreateRtmDocStatusResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	getAsyncAppendRtmDocStatusResponse	1..1	

getAsyncQueryRtmDocStatus

Service definition of function ns__getAsyncQueryRtmDocStatus

Style

Document

Input (Literal)

The input of this method is the argument getAsyncQueryRtmDocStatus having the structure defined by the following table.

Name	Type	Occurs	Description
cmdToken	string	1..1	

Output (Literal)

The output of this method is the argument getAsyncQueryRtmDocumentStatusResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	getAsyncQueryRtmDocumentStatusResponse	1..1	

getAsyncReplaceRtmDocStatus

Get status from the async. test commmand

Style

Document

Input (Literal)

The input of this method is the argument getAsyncReplaceRtmDocStatus having the structure defined by the following table.

Name	Type	Occurs	Description
cmdToken	string	1..1	

Output (Literal)

The output of this method is the argument getAsyncReplaceRtmDocStatusResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	getAsyncReplaceRtmDocStatusResponse	1..1	

getAsyncStatus

Get status from an asynchronous command

Style

Document

Input (Literal)

The input of this method is the argument `getAsyncStatus` having the structure defined by the following table.

Name	Type	Occurs	Description
cmdToken	string	1..1	

Output (Literal)

The output of this method is the argument `getAsyncStatusResponseResult` having the structure defined by the following table.

Name	Type	Occurs	Description
result	getAsyncStatusResponse	1..1	

GetAttributeDefInfo

Service definition of function ns__GetAttributeDefInfo

Style

Document

Input (Literal)

The input of this method is the argument GetAttributeDefInfo having the structure defined by the following table.

Name	Type	Occurs	Description
GetAttributeDefInfoRequest	GetAttributeDefInfoRequest	0..1	

Output (Literal)

The output of this method is the argument GetAttributeDefInfoReply of type GetAttributeDefInfoReply having the structure defined by the following table.

Name	Type	Occurs	Description
attributeDefInfo	AttributeDefInfo	1..1	

Remarks

The GetAttributeDefInfo call provides a method for getting information about attribute definition.

If list or user attribute is specified in input then it is possible to filter values for that attribute depending on Access Rights set for users in the specified category. In response will be list with only allowed users in the category.

Example of usage:

```
<urn:GetAttributeDefInfo>
  <urn:GetAttributeDefInfoRequest>
    <sessionId>1111111111</sessionId>
    <classSpecifier>
      <Type>0</Type>
      <Value>2</Value>
    </classSpecifier>
    <attributeDefSpecifier>
      <Type>1</Type>
      <Value>MANAGER</Value>
      <CategoryInfo>
        <DisplayName>RMDEMO</DisplayName>
      </CategoryInfo>
    </attributeDefSpecifier>
  </urn:GetAttributeDefInfoRequest>
</urn:GetAttributeDefInfo>
```

GetAvailableTransitions

Get all available transitions for requirement defined in the workflow.

Style

Document

Input (Literal)

The input of this method is the argument `GetAvailableTransitions` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
anObject	InRequirement	1..1	Requirement attributes and its values, used as search criteria

Output (Literal)

The output of this method is the argument `GetAvailableTransitionsResult` having the structure defined by the following table.

Name	Type	Occurs	Description
sPUIID	string	1..1	PUIID of specified requirement
sClassName	string	1..1	Name of requirement class
sDescription	string	0..1	Description of requirement
sCreatedBy	string	1..1	User, who created requirement
sModifiedBy	string	1..1	Last user, who modified requirement
nCreateTime	dateTime	1..1	Time of requirement creation
nModifiedTime	dateTime	1..1	Last time, when requirement was modified
sTitle	string	0..1	Title of requirement
nObjectId	int	1..1	Unique identifier of requirement
nObjectVersionId	int	1..1	Unique identifier of requirement version
sURL	string	1..1	Url, which can be used to access requirement in browser
sState	string	1..1	Workflow state of requirement
sOwner	string	1..1	Owner of workflow state
transitionList	Transitions	0..1	List of available transitions and its properties

Remarks

Example of usage:

```
<urn:GetAvailableTransitions>
  <sessionId>1111111111</sessionId>
  <anObject>
```

```
<sClassName>Component_Requirements</sClassName>  
<sPUID>COMP_000008</sPUID>  
</anObject>  
</urn:GetAvailableTransitions>
```

GetCategoryByDisplayName

Get category id by category display name.

Style

Document

Input (Literal)

The input of this method is the argument GetCategoryByDisplayName having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sDisplayName	string	1..1	Display name of category

Output (Literal)

The output of this method is the argument CategoryResult having the structure defined by the following table.

Name	Type	Occurs	Description
nId	long	1..1	Unique identifier of specified category

Remarks

Additional info:

'sDisplayName' value of category name can be specified as:

- a** category full name, separated by '/' or '\'
 - b** category name(only in case when name is unique);

Example of usage:

```
<urn:GetCategoryByDisplayName>
  <sessionId>1111111111</sessionId>
  <sDisplayName>RMDEMO/Availability</sDisplayName>
</urn:GetCategoryByDisplayName>
```

GetCategoryById

Get category information by given category id.

Style

Document

Input (Literal)

The input of this method is the argument GetCategoryById having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
categoryId	long	1..1	Unique identifier of category
parentLevel	long	0..1	Unique identifier of parent category

Output (Literal)

The output of this method is the argument CategoryEx having the structure defined by the following table.

Name	Type	Occurs	Description
nId	long	1..1	Unique identifier of specified category
nType	long	1..1	Type of specified category
sName	string	1..1	Name of specified category
sDescription	string	0..1	Description of specified category
sDisplayName	string	1..1	Display name of specified category
nParentId	long	1..1	Unique identifier of parent category

Remarks:

Returned 'nParentId' property shows the parent category of specified category. Since each category can have subcategory, same category can have few parent categories. Returned 'nParentId' value depends on specified 'parentLevel' value.

Example: There is category "C", which is a sub category of "B" category. And "B" category is subcategory of "A" category. So, category "C" have two parent categories "B" and "A".

For given example if to specify "categoryId" of "C" category, and specify 'parentLevel' as 0 – then returned 'nParentId' will show id of category "B". If specify 'parentLevel' as 1 – then returned 'nParentId' will show id of category "A".

So 'parentLevel' can have values from 0 to N, depending on number of parent categories of specified category. For any sub category, the highest parent category is root category. For specified root category 'nParentId' property have "-1" value, it means that there is not parent category of root category.

Example of usage:

```
<urn:GetCategoryById>
  <sessionId>1437063352</sessionId>
```



```
<categoryId>15</categoryId>  
<parentLevel>0</parentLevel>  
</urn:GetCategoryById>
```

Additional info:

'sDisplayName' value of category name in the output will be specified with '\' separator, e.g. 'RMDEMO\Availability\Cost'

GetClassDefInfo

Get definition of specified class by class Id/name

Style

Document

Input (Literal)

The input of this method is the argument GetClassDefInfo having the structure defined by the following table.

Name	Type	Occurs	Description
GetClassDefInfoRequest	GetClassDefInfoRequest	0..1	Class identification properties, used as search criteria

Output (Literal)

The output of this method is the argument GetClassDefInfoReply of type GetClassDefInfoReply having the structure defined by the following table.

Name	Type	Occurs	Description
classDefInfo	ClassDefInfo	1..1	A set of class properties

Example of usage:

```
<urn:GetClassDefInfoRequest>
  <sessionId>1479221671</sessionId>
  <classSpecifier>
    <Type>1</Type>
    <Value>Class_Name</Value>
  </classSpecifier>
</urn:GetClassDefInfoRequest>
```

GetCollectionById

Get a collection specified by collection id.

Style

Document

Input (Literal)

The input of this method is the argument GetCollectionById having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
collection-id	int	1..1	Unique identifier of collection

Output (Literal)

The output of this method is the argument GetCollectionByIdResult having the structure defined by the following table.

Name	Type	Occurs	Description
collection	CollectionEx	1..1	A set of collection properties

Example of usage:

```
<urn:GetCollectionById>  
  <sessionId>1479221671</sessionId>  
  <collection-id>1</collection-id>  
</urn:GetCollectionById>
```

GetCollectionByName

Get a collection specified by name

Style

Document

Input (Literal)

The input of this method is the argument GetCollectionByName having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
collection-name	string	1..1	Name of collection

Output (Literal)

The output of this method is the argument GetCollectionByNameResult having the structure defined by the following table.

Name	Type	Occurs	Description
collection	CollectionEx	1..1	A set of collection properties

Example of usage:

```
<urn:GetCollectionByName>  
  <sessionId>1479221671</sessionId>  
  <collection-name>Collection_Name</collection-name>  
</urn:GetCollectionByName>
```

GetCollectionInfo

Service definition of function ns__GetCollectionInfo

Style

Document

Input (Literal)

The input of this method is the argument GetCollectionInfo having the structure defined by the following table.

Name	Type	Occurs	Description
GetCollectionInfoRequest	GetCollectionInfoRequest	0..1	

Output (Literal)

The output of this method is the argument GetCollectionInfoReply of type GetCollectionInfoReply having the structure defined by the following table.

Name	Type	Occurs	Description
collectionInfo	CollectionInfo	1..1	

GetObjectHierarchyInfo

Retrieves information about the hierarchy position for the specified requirement.

Style

Document

Input (Literal)

The input of this method is the document element GetObjectHierarchyInfo having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
object	RequirementIdentifier	1..1	A set of object attributes values

Output (Literal)

The output of this method is the document element HierarchyInfoResponse having the structure defined by the following table.

Name	Type	Occurs	Description
parent	RequirementIdentifier	0..1	A set of parent object attributes values
order	int	1..1	Order of requirement in the scope of the parent

Example of usage:

```
<urn:GetObjectHierarchyInfo>
  <sessionId>1111111111</sessionId>
  <object>
    <sPUID>COMP_000006</sPUID>
    <sClassName>Component_Requirements</sClassName>
  </object>
</urn:GetObjectHierarchyInfo>
```

GetObjectsInCollection

Get objects in the given collection.

Style

Document

Input (Literal)

The input of this method is the argument `GetObjectsInCollection` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
collectionId	int	1..1	
bDetagDescription	boolean	0..1	
aAttrList	StringList	0..1	

Output (Literal)

The output of this method is the argument `GetObjectsInCollectionResponse` having the structure defined by the following table.

Name	Type	Occurs	Description
results	RequirementList	1..1	

GetObjectsInCollections

Get objects in the given collections.

Style

Document

Input (Literal)

The input of this method is the argument `GetObjectsInCollections` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
collectionIds	IdList	1..1	
bDetagDescription	boolean	0..1	
aAttrList	StringList	0..1	

Output (Literal)

The output of this method is the argument `GetObjectsInCollectionsResponse` having the structure defined by the following table.

Name	Type	Occurs	Description
results	RequirementList	1..1	

GetObjectsInRMDocument

Service definition of function ns__GetObjectsInRMDocument

Style

Document

Input (Literal)

The input of this method is the argument GetObjectsInRMDocument having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocId	int	1..1	
detagHtmlAttributes	boolean	0..1	
aAttrList	StringList	0..1	

Output (Literal)

The output of this method is the argument GetObjectsInRMDocumentResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	RequirementList	1..1	

GetProjectInfo

Service definition of function ns__GetProjectInfo

Style

Document

Input (Literal)

The input of this method is the argument GetProjectInfo having the structure defined by the following table.

Name	Type	Occurs	Description
GetProjectInfoRequest	GetProjectInfoRequest	0..1	

Output (Literal)

The output of this method is the argument GetProjectInfoReply of type GetProjectInfoReply having the structure defined by the following table.

Name	Type	Occurs	Description
projectInfo	ProjectInfo	1..1	

GetRMDocSnapshotDiffUrl

Service definition of function ns__GetRMDocSnapshotDiffUrl

Style

Document

Input (Literal)

The input of this method is the argument GetRMDocSnapshotDiffUrl having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
chngid	int	1..1	
baseid	int	1..1	

Output (Literal)

The output of this method is the argument GetRMDocSnapshotDiffUrlResult having the structure defined by the following table.

Name	Type	Occurs	Description
sUrl	string	1..1	

GetRMDocSnapshots

Returns list of snapshots for specified document.

Style

Document

Input (Literal)

The input of this method is the argument GetRMDocSnapshots having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
documentId	int	1..1	

Output (Literal)

The output of this method is the argument GetRMDocSnapshotsResult having the structure defined by the following table.

Name	Type	Occurs	Description
snapshots	RtmDocSnaphotOutList	1..1	

GetRtmDocSnapshotById

Retrieve document snapshot information by id. See ns__RtmDocSnapshot.

Style

Document

Input (Literal)

The input of this method is the argument GetRtmDocSnapshotById having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
id	int	1..1	

Output (Literal)

The output of this method is the argument GetRtmDocSnapshotByIdResult having the structure defined by the following table.

Name	Type	Occurs	Description
snapshot	RtmDocSnapshotOut	1..1	

GetRtmDocSnapshotByName

Retrieve document snapshot information by name. See ns__RtmDocSnapshot.

Style

Document

Input (Literal)

The input of this method is the argument GetRtmDocSnapshotByName having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
name	string	1..1	

Output (Literal)

The output of this method is the argument GetRtmDocSnapshotByNameResult having the structure defined by the following table.

Name	Type	Occurs	Description
snapshot	RtmDocSnapshotOut	1..1	

GetSharedFile

Download a shared file for local use

Style

Document

Input (Literal)

The input of this method is the argument GetSharedFile having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
iFileStorageId	int	1..1	
oFile	FileAttachment	1..1	

Output (Literal)

The output of this method is the argument FileAttachmentResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	FileAttachment	1..1	

GetTemplateFile

Get the template file of the template specified by template id

Style

Document

Input (Literal)

The input of this method is the argument GetTemplateFile having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
id	int	1..1	

Output (Literal)

The output of this method is the argument GetTemplateFileResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	base64Binary	1..1	

getTestAsyncStatus

Get status from the async. test command

Style

Document

Input (Literal)

The input of this method is the argument `getTestAsyncStatus` having the structure defined by the following table.

Name	Type	Occurs	Description
cmdToken	string	1..1	

Output (Literal)

The output of this method is the argument `getTestAsyncStatusResponseResult` having the structure defined by the following table.

Name	Type	Occurs	Description
result	testAsyncStatusResponse	1..1	

GetUserAttributeDefInfo

Get detailed information about user attribute and list of its values.

Style

Document

Input (Literal)

The input of this method is the argument GetUserAttributeDefInfo having the structure defined by the following table.

Name	Type	Occurs	Description
GetUserAttributeDefInfoRequest	GetUserAttributeDefInfoRequest	0..1	

Output (Literal)

The output of this method is the argument GetUserAttributeDefInfoReply of type GetUserAttributeDefInfoReply having the structure defined by the following table.

Name	Type	Occurs	Description
attributeDefInfo	UserAttributeDefInfo	1..1	

Remarks

The GetUserAttributeDefInfo call provides a method for get detailed information about user attribute and list of its values. In the output for users <DisplayName> value changes depending on the option selected in Project Settings.

Please note that user attribute values are filtered if category was specified.

Example of usage:

```
<urn:GetUserAttributeDefInfo>
  <urn:GetUserAttributeDefInfoRequest>
    <sessionId>1467923392</sessionId>
    <classSpecifier>
      <Type>1</Type>
      <Value>Product_Requirements</Value>
    </classSpecifier>
    <attributeDefSpecifier>
      <Type>1</Type>
      <Value>Manager</Value>
      <CategoryInfo>
        <CategoryId>3</CategoryId>
      </CategoryInfo>
    </attributeDefSpecifier>
  </urn:GetUserAttributeDefInfoRequest>
</urn:GetUserAttributeDefInfo>
```

IsSSOLoginEnabled

Service definition of function ns__IsSSOLoginEnabled

Style

Document

Input (Literal)

The input of this method is the argument IsSSOLoginEnabled having the structure defined by the following table.

Name	Type	Occurs	Description
sDatabase	string	1..1	

Output (Literal)

The output of this method is the argument IsSSOLoginEnabledResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	boolean	1..1	

LinkObjects

Link multiple children to a parent object

Style

Document

Input (Literal)

The input of this method is the argument LinkObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sParentClassName	string	1..1	
sParentPUID	string	1..1	
sChildClassName	string	1..1	
aChildPUIDs	StringList	1..1	

Output (Literal)

The output of this method is the argument LinkObjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	string	1..1	

Remarks

Example of usage:

```
<urn:LinkObjects>
  <sessionId>11111111</sessionId>
  <sParentClassName>Marketing_Requirements</sParentClassName>
  <sParentPUID>MRKT_000020</sParentPUID>
  <sChildClassName>Product_Requirements</sChildClassName>
  <aChildPUIDs>
    <item>PROD_000023</item>
  </aChildPUIDs>
</urn:LinkObjects>
```

Additional Info:

It is also possible to undelete deleted requirement links using LinkObjects/ LinkObjectsEx/ LinkObjectsExById web services. Any links between requirements that were deleted now can be restored.

Usage is the same as on creating link.

LinkObjectsEx

Link multiple children to a parent object

Style

Document

Input (Literal)

The input of this method is the argument LinkObjectsEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nRelationId	int	0..1	
sRelationName	string	0..1	
sParentPUID	string	1..1	
aChildPUIDs	StringList	1..1	

Output (Literal)

The output of this method is the argument LinkObjectsExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	string	1..1	

Remarks

Example of usage:

```
<urn:LinkObjectsEx>
  <sessionId>1111111111</sessionId>
  <nRelationId>5</nRelationId>
  <sParentPUID>MRKT_000021</sParentPUID>
  <aChildPUIDs>
    <item>PROD_000023</item>
  </aChildPUIDs>
</urn:LinkObjectsEx>
```

Additional Info:

It is also possible to undelete deleted requirement links using LinkObjects/ LinkObjectsEx/ LinkObjectsExById web services. Any links between requirements that were deleted now can be restored.

Usage is the same as on creating link.

LinkObjectsExById

Link multiple children to a parent object

Style

Document

Input (Literal)

The input of this method is the argument LinkObjectsExById having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nRelationId	int	0..1	
sRelationName	string	0..1	
nParentId	int	1..1	
aChildIds	IdList	1..1	

Output (Literal)

The output of this method is the argument LinkResults having the structure defined by the following table.

Name	Type	Occurs	Description
sResult	string	1..1	
lLinkList	StringList	0..1	

Remarks

Example of usage:

```
<urn:LinkObjectsExById>
  <sessionId>11111111</sessionId>
  <nRelationId>5</nRelationId>
  <nParentId>32</nParentId>
  <aChildIds>
    <item>47</item>
  </aChildIds>
</urn:LinkObjectsExById>
```

Additional Info:

It is also possible to undelete deleted requirement links using LinkObjects/ LinkObjectsEx/ LinkObjectsExById web services. Any links between requirements that were deleted now can be restored.

Usage is the same as on creating link.

ListAllAllowedCategories

Get a list of all allowed categories for the project.

Style

Document

Input (Literal)

The input of this method is the argument ListAllAllowedCategories having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument ListAllAllowedCategoriesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CategoryList	1..1	

ListCategories

Get a list of child categories for the given parent category.

Style

Document

Input (Literal)

The input of this method is the argument ListCategories having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
categoryId	long	1..1	

Output (Literal)

The output of this method is the argument ListCategoriesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CategoryList	1..1	

ListCollections

Get a list of collection names in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListCollections having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument ListCollectionsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CollectionList	1..1	

ListCollectionsEx

Get a list of collections in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListCollectionsEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument ListCollectionsExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CollectionExList	1..1	

ListCollectionsEx2

Get a list of collections in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListCollectionsEx2 having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument ListCollectionsEx2Response having the structure defined by the following table.

Name	Type	Occurs	Description
result	CollectionEx2List	1..1	

ListObjectLinks

List secondary links for a specified requirement

Style

Document

Input (Literal)

The input of this method is the argument ListObjectLinks having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
aRequirement	Requirement	1..1	
bIncludeDeleted	boolean	0..1	

Output (Literal)

The output of this method is the argument ListObjectLinksResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	LinkList	1..1	

Remarks

Additional info:

- 1 It is possible to return deleted links in the result list. If bIncludeDeleted value is set to true then all related requirements will be returned. Each returned item has status of link (current/deleted).
If bIncludeDeleted is set to false, or not specified, only current links will be returned.
- 2 If bIncludeAllRequirementVersions is set to true all requirement versions will be returned. Each returned item has status of link (current/replaced).
If bIncludeAllRequirementVersions is set to false, or not specified, only current links will be returned.

Example of usage:

```
<urn:ListObjectLinks>
  <sessionId>1436973992</sessionId>
  <aRequirement>
    <sClassName>Marketing_Requirements</sClassName>
    <nObjectId>1</nObjectId>
  </aRequirement>
  <bIncludeDeleted>true</bIncludeDeleted>
</urn:ListObjectLinks>
```

ListRootCategories

Get a list of top-level categories in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListRootCategories having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument ListRootCategoriesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	CategoryList	1..1	

ListRtmDocuments

Get a list of rtm documents in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListRtmDocuments having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
filter	DocumentsFilter	0..1	

Output (Literal)

The output of this method is the argument ListRtmDocumentsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	RtmDocumentList	1..1	

Remarks

Additional info:

ListRtmDocuments WS also allows to filter documents by category.

Optional input parameters were added:

- sCategory
Can be specified by full path (separated with '\' or '/') or by category_id
- bIncludeSubCategories

Example of usage:

```
<urn>ListRtmDocuments>
  <sessionId>1</sessionId>
  <filter>
    <sCreatedBy>ephoto</sCreatedBy>
    <sCategory>RMDEMO\Functional</sCategory>
    <bIncludeSubCategories>true</bIncludeSubCategories>
  </filter>
</urn>ListRtmDocuments>
```

It is also possible to filter list of documents to show Parent and Child documents or not, depending on bIncludeMasterDocuments and bIncludeChildDocuments values.

Example of usage:

```
<urn>ListRtmDocuments>
```

```
<sessionId>1</sessionId>
<filter>
  <sCreatedBy>ephoto</sCreatedBy>
  <bIncludeMasterDocuments>true</bIncludeMasterDocuments>
  <bIncludeChildDocuments>false</bIncludeChildDocuments>
</filter>
</urn:ListRtmDocuments>
```

ListRtmDocumentsEx

Get a list of rtm documents and snapshots in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListRtmDocumentsEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
filter	DocumentsFilter	0..1	Settings for querying documents
bIncludeSnapshots	boolean	1..1	Defines if the result should include snapshots along with the documents.

Output (Literal)

The output of this method is the argument ListRtmDocumentsExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	RtmDocumentListEx	1..1	

Remarks

Additional info:

ListRtmDocumentsEx WS also allows to filter documents by category:

- sCategory
Can be specified by full path (separated with '\' or '/') or by category_id
- bIncludeSubCategories

Example of usage:

```
<urn:ListRtmDocuments>
  <sessionId>1</sessionId>
  <filter>
    <sCreatedBy>ephoto</sCreatedBy>
    <sCategory>RMDemo\Functional</sCategory>
    <bIncludeSubCategories>true</bIncludeSubCategories>
  </filter>
</urn:ListRtmDocuments>
```

It is also possible to filter list of documents to show Parent and Child documents or not, depending on bIncludeMasterDocuments and bIncludeChildDocuments values.

Example of usage:


```
<urn:ListRtmDocuments>  
  <sessionId>1</sessionId>  
  <filter>  
    <sCreatedBy>ephoto</sCreatedBy>  
    <bIncludeMasterDocuments>true</bIncludeMasterDocuments>  
    <bIncludeChildDocuments>>false</bIncludeChildDocuments>  
  </filter>  
</urn:ListRtmDocuments>
```

ListTemplates

Get a list of document templates in the project.

Style

Document

Input (Literal)

The input of this method is the argument ListTemplates having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument ListTemplatesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	TemplateList	1..1	

Login

Login into RTM - use returned sessionId for subsequent calls.

Style

Document

Input (Literal)

The input of this method is the argument Login having the structure defined by the following table.

Name	Type	Occurs	Description
sUserId	string	0..1	
sPassword	string	0..1	
sProjectName	string	0..1	
sDatabase	string	1..1	

Output (Literal)

The output of this method is the argument LoginResult having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Remarks

- It is possible to login using Internal RM user or SSO user credentials.
- The session will expire as defined by the administrator for the instance you logged in to.

Example of usage:

```
<urn:Login>
<!--Internal RM user name or SSO user name-->
  <sUserId>epphoto</sUserId>
  <!--Internal RM user password or SSO user password-->
  <sPassword>rtm</sPassword>
  <sProjectName>RMDEMO</sProjectName>
  <sDatabase>RM</sDatabase>
</urn:Login>
```

Possible to use RM Browser session in any RM web service instead of sessionId value.

Required values:

- content of "authenticatedxxxxxx" cookie
- name of the database to log in

To use it specify for sessionId value:

"database name": "authenticatedxxxxxx cookie value" (without double quotes)

Example of usage:

```
<urn:ListRtmDocuments>
  <sessionId>RM:2EE9D7815C921FCB3E8F1FB17F8C9D83F5EC103B4B055E2AA6290
0F74F95E250B4C63A18A362554EE703EA2BD90F4C4771F8BAC71D6B8520CCCDF2FE64D
68A1269CC8741AF586AE10EFD7A5A905D5F71CAC63BAFBBDD4FD33408356A970669ADD
5D4F37CEEEEA89865930ECF8ADED5C26B2C6E62A6561592</sessionId>
</urn:ListRtmDocuments>
```

Logging in with SSO Token (e.g. from Azure AD) in Header

NOTE To get the SSO token, call the Rest service function GetSSOToken (<https://host/rtmBrowser/RestServices/Admin/GetSSOToken>).

The token value used is assigned to the "token" attribute in the JSON response. When using SAML2 format on the Login call, it must be decoded (from Base64 format). For JWT Login calls, the token can be used in Base64 encoding.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:rtmService">
  <soapenv:Header>
    TOKEN_VALUE
  </soapenv:Header>
  <soapenv:Body>
    <urn:Login>
      <sProjectName>RMDemo</sProjectName>
      <sDatabase>RM</sDatabase>
    </urn:Login>
  </soapenv:Body>
</soapenv:Envelope>
```

Logging in with SSO Token (e.g. from Azure AD) in Password Field

NOTE To get the SSO token, call the Rest service function GetSSOToken (<https://host/rtmBrowser/RestServices/Admin/GetSSOToken>).

The token value used is assigned to the "token" attribute in the JSON response. When using SAML2 format on the Login call, it must be decoded (from Base64 format). For JWT Login calls, the token can be used in Base64 encoding.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:rtmService">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:Login>
      <sPassword>TOKEN_VALUE</sPassword>
      <sProjectName>RMDemo</sProjectName>
      <sDatabase>RM</sDatabase>
    </urn:Login>
  </soapenv:Body>
</soapenv:Envelope>
```


LoginByProjectId

Login into RTM - use returned sessionId for subsequent calls.

Style

Document

Input (Literal)

The input of this method is the argument LoginByProjectId having the structure defined by the following table.

Name	Type	Occurs	Description
sUserId	string	0..1	
sPassword	string	0..1	
nProjectId	int	1..1	
sDatabase	string	1..1	

Output (Literal)

The output of this method is the argument LoginByProjectIdResult having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Remarks

- It is possible to login using Internal RM user or SSO user credentials.
- The session will expire as defined by the administrator for the instance you logged in to.

Example of usage:

```
<urn:LoginByProjectId>  
  <!--Internal RM user name or SSO user name-->  
  <sUserId>ephoto</sUserId>  
  <!--Internal RM user password or SSO user password-->  
  <sPassword>rtm</sPassword>  
  <nProjectId>1</nProjectId>  
  <sDatabase>RM</sDatabase>  
</urn:LoginByProjectId>
```

Logging in with SSO Token (e.g. from Azure AD) in Header



NOTE To get the SSO token, call the Rest service function GetSSOToken (<https://host/rtmBrowser/RestServices/Admin/GetSSOToken>).

The token value used is assigned to the "token" attribute in the JSON response. When using SAML2 format on the Login call, it must be decoded (from Base64 format). For JWT Login calls, the token can be used in Base64 encoding.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:rtmService">
  <soapenv:Header>
    TOKEN_VALUE
  </soapenv:Header>
  <soapenv:Body>
    <urn:LoginByProjectId>
      <nProjectId>1</nProjectId>
      <sDatabase>RM</sDatabase>
    </urn:LoginByProjectId>
  </soapenv:Body>
</soapenv:Envelope>
```

Logging in with SSO Token (e.g. from Azure AD) in Password Field



NOTE To get the SSO token, call the Rest service function GetSSOToken (<https://host/rtmBrowser/RestServices/Admin/GetSSOToken>).

The token value used is assigned to the "token" attribute in the JSON response. When using SAML2 format on the Login call, it must be decoded (from Base64 format). For JWT Login calls, the token can be used in Base64 encoding.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:rtmService">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:LoginByProjectId>
      <sPassword>TOKEN_VALUE</sPassword>
      <nProjectId>1</nProjectId>
      <sDatabase>RM</sDatabase>
    </urn:LoginByProjectId>
  </soapenv:Body>
</soapenv:Envelope>
```

LoginEx

Login into RTM and return if it was admin login along with sessionId.

Style

Document

Input (Literal)

The input of this method is the argument LoginEx having the structure defined by the following table.

Name	Type	Occurs	Description
sUserId	string	0..1	
sPassword	string	0..1	
sProjectName	string	0..1	
sDatabase	string	1..1	

Output (Literal)

The output of this method is the argument LoginExResult having the structure defined by the following table.

Name	Type	Occurs	Description
info	LoginExResultInfo	1..1	

Remarks

- It is possible to login using Internal RM user or SSO user credentials.
- The session will expire as defined by the administrator for the instance you logged in to.

Example of usage:

```
<urn:LoginEx>
  <!--Internal RM user name or SSO user name-->
  <sUserId>ephoto</sUserId>
  <!--Internal RM user password or SSO user password-->
  <sPassword>rtm</sPassword>
  <sProjectName>RMDEMO</sProjectName>
  <sDatabase>RM</sDatabase>
</urn:LoginEx>
```

Logging in with SSO Token (e.g. from Azure AD) in Header



NOTE To get the SSO token, call the Rest service function GetSSOToken (<https://host/rtmBrowser/RestServices/Admin/GetSSOToken>).

The token value used is assigned to the "token" attribute in the JSON response. When using SAML2 format on the Login call, it must be decoded (from Base64 format). For JWT Login calls, the token can be used in Base64 encoding.


```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/
envelope/" xmlns:urn="urn:rtmService">
  <soapenv:Header>
    TOKEN_VALUE
  </soapenv:Header>
  <soapenv:Body>
    <urn:LoginEx>
      <sProjectName>RMDEMO</sProjectName>
      <sDatabase>RM</sDatabase>
    </urn:LoginEx>
  </soapenv:Body>
</soapenv:Envelope>
```

Logging in with SSO Token (e.g. from Azure AD) in Password Field



NOTE To get the SSO token, call the Rest service function GetSSOToken (<https://host/rtmBrowser/RestServices/Admin/GetSSOToken>).

The token value used is assigned to the "token" attribute in the JSON response. When using SAML2 format on the Login call, it must be decoded (from Base64 format). For JWT Login calls, the token can be used in Base64 encoding.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/
envelope/" xmlns:urn="urn:rtmService">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:LoginEx>
      <sPassword>TOKEN_VALUE</sPassword>
      <sProjectName>RMDEMO</sProjectName>
      <sDatabase>RM</sDatabase>
    </urn:LoginEx>
  </soapenv:Body>
</soapenv:Envelope>
```

Logout

Logout of RTM.

Style

Document

Input (Literal)

The input of this method is the argument Logout having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument LogoutResult having the structure defined by the following table.

Name	Type	Occurs	Description
value	int	1..1	

ModifyObjectsByScript

Mass Replace, Update, Approve, Reject, Delete, Remove, Undelete, Transition requirements by script text or report name. (Reports with run-time parameters are also supported)

Style

Document

Input (Literal)

The input of this method is the argument `ModifyObjectsByScript` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sAction	string	1..1	
aAttributes	NameValueList	1..1	
anScript	Script	1..1	
anTransition	InTransition	0..1	
bDontGenerateAlfEvent	boolean	0..1	

Output (Literal)

The output of this method is the argument `ModifyObjectsResponse` having the structure defined by the following table.

Name	Type	Occurs	Description
aSucceedObjects	ObjectsList	1..1	
aFailedObjects	ObjectsList	1..1	

Remarks

Additional Info:

New attributes values section is used for replace, update actions. Either it used for Reject action to set up `CHANGE_REASON` attribute.

Note that for reject Action of requirements `CHANGE_REASON` is mandatory.

For locating requirements for mass update next possibilities:

- script name
- script text

Using action 'transition' possible to transition several requirements to some state.

Example of usage:

1 Update requirements by script text:

```
<urn:ModifyObjectsByScript>
  <sessionId>1</sessionId>
```

```
<sAction>Update</sAction>
<aAttributes>
  <item>
    <sName>PRIORITY</sName>
    <sValue>1</sValue>
  </item>
  <item>
    <sName>TT_ITEM_STATE</sName>
    <sValue>New</sValue>
  </item>
</aAttributes>
<anScript>
  <sScriptText>select object_id from Marketing_Requirements
where STATUS IN ('Current') and group in ('ePhoto Requirements')
</sScriptText>
</anScript>
</urn:ModifyObjectsByScript>
```

2 Remove requirements by script text

```
<urn:ModifyObjectsByScript>
  <sessionId>1</sessionId>
  <sAction>Remove</sAction>
  <anScript>
    <sScriptText>select object_id from Marketing_Requirements
where STATUS IN ('Current') and DELIVERY_PHASE in ('TBD')
</sScriptText>
  </anScript>
</urn:ModifyObjectsByScript>
```

3 Reject requirements by scriptname (also with run-time parameters):

```
<urn:ModifyObjectsByScript>
  <sessionId>1</sessionId>
  <sAction>Reject</sAction>
  <aAttributes>
    <item>
      <sName>CHANGE_REASON</sName>
      <sValue>Some reason text...</sValue>
    </item>
  </aAttributes>
  <anScript>
    <anSavedReport>
      <sCategoryName>RMDEMO</sCategoryName>
      <sReportName>All changed requirements from document
(run time)</sReportName>
    </anSavedReport>
  </anScript>
  <anRunTimeParameters>
    <item>
      <sName>RTP__GROUP_1</sName>
      <sValue>ePhoto Requirements</sValue>
    </item>
  </anRunTimeParameters>
```

```
</urn:ModifyObjectsByScript>  
4 Transition requirements by script:  
<urn:ModifyObjectsByScript>  
  <sessionId>1436979823</sessionId>  
  <sAction>transition</sAction>  
  <anScript>  
    <sScriptText>select object_id from Product_Requirements  
where STATUS IN ('Current') </sScriptText>  
  </anScript>  
  <anTransition>  
    <sTransitionName>Approve</sTransitionName>  
  </anTransition>  
</urn:ModifyObjectsByScript>
```

MoveObjectInHierarchy

Move requirement to the other position in the hierarchy.

Style

Document

Input (Literal)

The input of this method is the document element MoveObjectInHierarchy having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
object	RequirementIdentifier	1..1	A set of object attributes values
position	HierarchyInfo	1..1	Position and optionally new parent

Output (Literal)

The output of this method is the document element MoveObjectInHierarchyResponse having the structure defined by the following table.

Name	Type	Occurs	Description
sMessage	string	1..1	

Example of usage:

```
<urn:MoveObjectInHierarchy>
  <sessionId>1559933409</sessionId>
  <object>
    <!--Optional:-->
    <sPUID>COMP_000039</sPUID>
    <sClassName>Component_Requirements</sClassName>
  </object>
  <position>
    <order>2</order>
    <!--Optional:-->
    <parent>
      <sPUID>COMP_000012</sPUID>
      <sClassName>Component_Requirements</sClassName>
    </parent>
  </position>
</urn:MoveObjectInHierarchy>
```

ProposeNewRequirement

Creates a requirement proposal.

Style

Document

Input (Literal)

The input of this method is the argument `ProposeNewRequirement` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sClassName	string	1..1	
aAttrList	NameValueList	1..1	
anECPsInfo	ECPsObjectInfo	0..1	
inChangeReason	string	1..1	

Output (Literal)

The output of this method is the argument `ProposeNewRequirementResult` having the structure defined by the following table.

Name	Type	Occurs	Description
oProposeNewRequirementResult	ProposeNewRequirementResult	1..1	

Example of usage:

```
<urn:ProposeNewRequirement>
  <sessionId>1418399921</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <aAttrList>
    <!--Zero or more repetitions:-->
    <item>
      <sName>Title</sName>
      <!--Optional:-->
      <sValue>new MRKT 15 - title</sValue>
    </item>
    <item>
      <sName>Text</sName>
      <!--Optional:-->
      <sValue>new MRKT 15 - text</sValue>
    </item>
  </aAttrList>
  <!--Optional:-->
  <anECPsInfo>
```

```
<iECPsClassId>18</iECPsClassId>  
<iECPsObjectId>1</iECPsObjectId>  
</anECPsInfo>  
<!--Optional:-->  
<inChangeReason>vvvvvv</inChangeReason>  
</urn:ProposeNewRequirement>
```


PutSharedFile

Upload a shared file for access by others

Style

Document

Input (Literal)

The input of this method is the argument PutSharedFile having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
oFile	FileAttachment	1..1	

Output (Literal)

The output of this method is the argument PutSharedFileResponse having the structure defined by the following table.

Name	Type	Occurs	Description
iFileStorageId	int	1..1	

QueryAttributes

Return the list of attributes defined in a class.

Style

Document

Input (Literal)

The input of this method is the argument QueryAttributes having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sClassName	string	1..1	
CategoryInfo	Category Info	0..1	Category identification properties, used as search criteria

Output (Literal)

The output of this method is the argument QueryAttributesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	AttributeDefList	1..1	

Remarks

The QueryAttributes call provides a method for getting attributes list from the specified class.

Attribute properties such as mandatory, default values, etc. will be also returned in the output.

There is also a possibility to filter user attributes values to have only users/groups that are assigned to the specified category.

Example of usage:

- 1 Get all attributes from the specified class:

```
<urn:QueryAttributes>
  <sessionId>111111111</sessionId>
  <sClassName>Product_Requirements</sClassName>
</urn:QueryAttributes>
```

- 2 Get all attributes of the specified class but filter possible values of user attributes by category:

```
<urn:QueryAttributes>
  <sessionId>111111111</sessionId>
  <sClassName>Product_Requirements</sClassName>
  <CategoryInfo>
    <CategoryId>-1</CategoryId>
  </CategoryInfo>
</urn:QueryAttributes>
```

```
    <DisplayName>RMDEMO\Availability</DisplayName>  
  </CategoryInfo>  
</urn:QueryAttributes>
```

QueryAttributesEx

Return the list of attributes defined in a class.

Style

Document

Input (Literal)

The input of this method is the argument QueryAttributesEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sClassName	string	1..1	
CategoryInfo	CategoryInfo	0..1	Category identification properties, used as search criteria

Output (Literal)

The output of this method is the argument QueryAttributesExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	AttributeDefExList	1..1	

Remarks

The QueryAttributesEx call provides a method for getting attributes list from the specified class.

All attribute properties such as mandatory, default values, etc returned in the output.

There is also a possibility to filter user attributes values to have only users/groups assigned to the specified category.

Example of usage:

- 1 Get all attributes from the specified class:

```
<urn:QueryAttributesEx>
  <sessionId>1111111111</sessionId>
  <sClassName>Product_Requirements</sClassName>
</urn:QueryAttributesEx>
```

- 2 Get all attribute of the specified class but filter possible values of user attributes by category:

```
<urn:QueryAttributesEx>
  <sessionId>1111111111</sessionId>
  <sClassName>Product_Requirements</sClassName>
  <CategoryInfo>
    <CategoryId>17</CategoryId>
  </CategoryInfo>
</urn:QueryAttributesEx>
```

QueryAttributesEx2

Return the list of attributes defined in a class.

Style

Document

Input (Literal)

The input of this method is the argument QueryAttributesEx2 having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Name of requirement class
CategoryInfo	CategoryInfo	0..1	Category identification properties, used as search criteria

Output (Literal)

The output of this method is the argument QueryAttributesEx2Response having the structure defined by the following table.

Name	Type	Occurs	Description
result	AttributeDefEx2List	1..1	List of attribute properties

Remarks

QueryAttributesEx2 call provides a method for getting attributes list from the specified class.

All attribute properties such as mandatory, default values, etc returned in the output.

There is also a possibility to filter user attributes values to have only users/groups assigned to the specified category.

Service also returns information about list attribute values, whether value is default and deleted.

Example of usage:

- 1 Get all attributes from the specified class:

```
<urn:QueryAttributesEx2>
  <sessionId>1111111111</sessionId>
  <sClassName>Product_Requirements</sClassName>
</urn:QueryAttributesEx2>
```

- 2 Get all attribute of the specified class but filter possible values of user attributes by category:

```
<urn:QueryAttributesEx2>
  <sessionId>1111111111</sessionId>
  <sClassName>Product_Requirements</sClassName>
  <CategoryInfo>
    <CategoryId>17</CategoryId>
```

```
</CategoryInfo>  
</urn:QueryAttributesEx>
```

QueryClasses

Return the list of defined classes in project.

Style

Document

Input (Literal)

The input of this method is the argument QueryClasses having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument QueryClassesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	ClassDefList	1..1	

QueryClassesEx

Return the list of defined classes in project.

Style

Document

Input (Literal)

The input of this method is the argument QueryClassesEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument QueryClassesExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	ClassDefExList	1..1	

QueryDatabases

Return the list of oracle databases.

Style

Document

Input (Literal)

The input of this method is the argument QueryDatabases that is always empty.

Output (Literal)

The output of this method is the argument QueryDatabasesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	DatabaseList	1..1	

QueryLinkClasses

Return the list of linked classes information in project.

Style

Document

Input (Literal)

The input of this method is the argument QueryLinkClasses having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nClassId	int	1..1	

Output (Literal)

The output of this method is the argument QueryLinkClassesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	LinkClassDefList	1..1	

QueryProjects

Return the list of defined projects in database.

Style

Document

Input (Literal)

The input of this method is the argument QueryProjects having the structure defined by the following table.

Name	Type	Occurs	Description
sUserId	string	0..1	
sPassword	string	0..1	
sDatabase	string	1..1	

Output (Literal)

The output of this method is the argument QueryProjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	ProjectList	1..1	

QueryProjectsEx

Return the list of defined projects in database.

Style

Document

Input (Literal)

The input of this method is the argument QueryProjectsEx having the structure defined by the following table.

Name	Type	Occurs	Description
sUserId	string	0..1	
sPassword	string	0..1	
sDatabase	string	1..1	

Output (Literal)

The output of this method is the argument QueryProjectsExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	ProjectExList	1..1	

QueryRtmDocumentStructure

Creates an XML representing RM Document structure.

Style

Document

Input (Literal)

The input of this method is the argument QueryRtmDocumentStructure having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocId	int	1..1	
nChapterId	int	0..1	
bIncludeReq	boolean	0..1	

Output (Literal)

The output of this method is the argument RtmDocumentOut having the structure defined by the following table.

Name	Type	Occurs	Description
info	RtmDocumentProperties	1..1	
sContent	string	1..1	

Remarks

Example of usage:

```
<urn:QueryRtmDocumentStructure>
  <sessionId>111111111</sessionId>
  <nDocId>12</nDocId>
  <nChapterId>169</nChapterId>
  <bIncludeReq>true</bIncludeReq>
</urn:QueryRtmDocumentStructure>
```

Example of output:

```
<ns:RtmDocumentOut>
  <info>
    <nId>12</nId>
    <sName>ePhoto Requirements</sName>
  </info>
  <sContent><![CDATA[<?xml version="1.0" ?>
    <Document name="ZVBob3RvIFJlcXVpcmVtZW50cw==" projectName="Uk1ERU1P"
    databaseName="U1RN" version="1.1" application="RM Web Services">
    <Chapter title="ZVBob3RvIFJlcXVpcmVtZW50cw==" uid="1806" iid="1"
    puid="1">
```

```
<Description>TU1N...LS0K</Description>
<!-- ... -->
<Chapter title="UmVxdWlyZW1lbnRz" uid="20" iid="4" puid="4">
<Description/>
<Chapter title="R2VuZXJhbCBSZXF1aXJlbWVudHM=" uid="26" iid="5"
puid="5">
<Description/>
<!-- ... -->
<Chapter title="RnVuY3Rpb25hbCBSZXF1aXJlbWVudHM=" uid="158" iid="43"
puid="7">
<Description/>
<Requirement classid="1" className="Marketing_Requirements"
title="" category="7" uid="118" iid="21">
  <Attribute attrid="3011" typeid="1" title="Title">
RVBob3RvIHdpbGwgYmUgYW4gb25saW5lIHBob3RvIGFsYnVt</Attribute>
  <Attribute attrid="31" typeid="6" title="Text">
TU1N...LQo=</Attribute>
  <Attribute attrid="26" typeid="12" title="Rqmt ID">
TVJL...MDE=</Attribute>
  <!-- ... -->
</Requirement>
<!-- ... -->
</Chapter>
<!-- ... -->
</Chapter>
<!-- ... -->
</Chapter>
<!-- ... -->
</Chapter>
<!-- ... -->
<Chapter title="VGFiGUg2YgRmlndXJlcw==" uid="1594" iid="341"
puid="40">
<Description>TU1N...qLS0K</Description>
</Chapter>
</Chapter>
</Document>]]</sContent>
</ns:RtmDocumentOut>
```

Additional info:

In the output of web service, each chapter and each requirement has iid attribute which means internal id of object in the document.

iid value can be used in next web services which work with document structure:

- [DocumentAddChapter](#)
- [DocumentMoveObject](#)
- [DocumentRemoveObject](#)

QuerySystemInfo

Return the system information.

Style

Document

Input (Literal)

The input of this method is the argument QuerySystemInfo having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument SystemInfoResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	SystemInfo	1..1	

ReadObject

Read object information with all the attribute values

Style

Document

Input (Literal)

The input of this method is the argument ReadObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
aRequirement	RequirementIdentifier	1..1	

Output (Literal)

The output of this method is the argument ReadObjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

ReadObjectAttachment

Read attachment of a requirement

Style

Document

Input (Literal)

The input of this method is the argument ReadObjectAttachment having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
aRequirement	RequirementIdentifier	1..1	A combination of PUID/classname or objectID/classname that specifies a project requirement.
sAttrName	string	1..1	File attachment attribute name
nFileIndex	number	0..1	The index for the file to retrieve from the file attachment attribute. Specify nFileIndex only if the file attachment attribute has Allow multiple selections enabled.

Output (Literal)

The output of this method is the argument ReadObjectAttachmentResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	FileAttachment	1..1	

Examples of usage:

Reading from a simple file attachment attribute by PUID

```
<urn:ReadObjectAttachment>
  <sessionId>443452989</sessionId>
  <aRequirement>
    <sPUID>MC_000001</sPUID>
    <sClassName>MyClass</sClassName>
  </aRequirement>
  <sAttrName>ATTACHMENT</sAttrName>
</urn:ReadObjectAttachment>
```

Reading from a simple file attachment attribute by Object ID

```
<urn:ReadObjectAttachment>
  <sessionId>443452989</sessionId>
  <aRequirement>
    <sClassName>MyClass</sClassName>
    <nObjectId>423</nObjectId>
  </aRequirement>
  <sAttrName>ATTACHMENT</sAttrName>
</urn:ReadObjectAttachment>
```

Reading from a multi file attachment attribute by PUID

```
<urn:ReadObjectAttachment>
  <sessionId>443452989</sessionId>
  <aRequirement>
    <sPUID>MRKT_000003</sPUID>
    <sClassName>Marketing_Requirements</sClassName>
  </aRequirement>
  <sAttrName>FILE_ATTACHMENT</sAttrName>
  <nFileIndex>0</nFileIndex>
</urn:ReadObjectAttachment>
```

Reading from a multi file attachment attribute by object ID

```
<urn:ReadObjectAttachment>
  <sessionId>443452989</sessionId>
  <aRequirement>
    <sPUID>MRKT_000003</sPUID>
    <sClassName>Marketing_Requirements</sClassName>
    <nObjectId>58</nObjectId>
  </aRequirement>
  <sAttrName>FILE_ATTACHMENT</sAttrName>
  <nFileIndex>0</nFileIndex>
</urn:ReadObjectAttachment>
```

ReadObjectEx

Read object information with all the attribute values

Style

Document

Input (Literal)

The input of this method is the argument ReadObjectEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
aRequirement	RequirementIdentifier	1..1	
detagHtmlAttributes	boolean	0..1	

Output (Literal)

The output of this method is the argument ReadObjectExResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

ReadObjects

Read object information with all the attribute values

Style

Document

Input (Literal)

The input of this method is the argument ReadObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
requirements	RequirementIdentifierList	1..1	

Output (Literal)

The output of this method is the argument ReadObjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementList	1..1	

ReadSyncXrefLink

Reads a link from Sync_Xref table

Style

Document

Input (Literal)

The input of this method is the argument ReadSyncXrefLink having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
aXref	SyncXrefLink	1..1	

Output (Literal)

The output of this method is the argument ReadSyncXrefLinkResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	SyncXrefLinkList	1..1	

RejectChangeRequest

Service definition of function ns__RejectChangeRequest

Style

Document

Input (Literal)

The input of this method is the argument RejectChangeRequest having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
className	string	1..1	
requestId	long	1..1	
reason	string	1..1	

Output (Literal)

The output of this method is the argument RejectChangeRequestResult having the structure defined by the following table.

Name	Type	Occurs	Description
status	CommandStatus	1..1	

RemoveObject

RemoveObject Removes a requirement from the database

Style

Document

Input (Literal)

The input of this method is the argument RemoveObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
objectDef	ObjectDef	1..1	

Output (Literal)

The output of this method is the argument RemoveObjectResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	boolean	1..1	

RemoveObjectFromCollection

Service definition of function ns__RemoveObjectFromCollection

Style

Document

Input (Literal)

The input of this method is the argument RemoveObjectFromCollection having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
objectIdentifier	RequirementIdentifier	1..1	
collectionName	string	1..1	

Output (Literal)

The output of this method is the argument removeObjectFromCollectionResponseResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	removeObjectFromCollectionResponse	1..1	

ReplaceObject

Replaces a RM requirement with specified requirement data

Style

Document

Input (Literal)

The input of this method is the argument ReplaceObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	

Output (Literal)

The output of this method is the argument ReplaceObjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

ReplaceObjectEx

Replaces a RM requirement with specified requirement data

Style

Document

Input (Literal)

The input of this method is the argument ReplaceObjectEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	
bDontGenerateAlfEvent	boolean	0..1	

Output (Literal)

The output of this method is the argument ReplaceObjectExResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

ReplaceObjectWithAttachment

Replace a RM requirement with attachment, given a class id and requirement data

Style

Document

Input (Literal)

The input of this method is the argument `ReplaceObjectWithAttachment` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	
sAttrName	string	1..1	
oFile	FileAttachment	0..1	
bDontGenerateAlfEvent	boolean	0..1	

Output (Literal)

The output of this method is the argument `ReplaceObjectWithAttachmentResult` having the structure defined by the following table.

Name	Type	Occurs	Description
result	ReplaceRequirementOut	1..1	

ResolveSuspicion

Clears suspect links.

Style

Document

Input (Literal)

The input of this method is the argument ResolveSuspicion having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sClassName	string	1..1	
nObjectId	int	1..1	
bPrimaryLinks	boolean	0..1	If true , all outgoing links will be cleared from suspect. If false , outgoing links will still be suspect. Default is true
bSecondaryLinks	boolean	0..1	If true , all incoming links will be cleared from suspect. If false , incoming links will still be suspect. Default is true

RTMDocumentAddChapter

The New Chapter Shall be added after the Existing Chapter at the same level.

Style

Document

Input (Literal)

The input of this method is the argument RTMDocumentAddChapter having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocument-id	int	1..1	
nExistChapter-id	int	1..1	
chapterName	ns-chapterDetails	1..1	

Output (Literal)

The output of this method is the argument RTMDocumentAddChapterResponse having the structure defined by the following table.

Name	Type	Occurs	Description
sMessage	string	1..1	

RTMDocumentAddRequirement

The Requirement Shall be added in the Existing Chapter.

Style

Document

Input (Literal)

The input of this method is the argument RTMDocumentAddRequirement having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocument-id	int	1..1	
nChapter-id	int	0..1	
sClassName	string	0..1	
nObject-id	int	0..1	
anObject	Requirement	0..1	

Output (Literal)

The output of this method is the argument RTMDocumentAddRequirementResponse having the structure defined by the following table.

Name	Type	Occurs	Description
sMessage	string	1..1	

RTMDocumentExchangeRequirement

Exchange requirements inplace.

Style

Document

Input (Literal)

The input of this method is the argument RTMDocumentExchangeRequirement having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocument-id	int	1..1	
nChapter-id	int	1..1	
sClassName	string	1..1	
nOldObject-id	int	1..1	
nNewObject-id	int	1..1	
ECPObject	ECPObjInfo	0..1	

Output (Literal)

The output of this method is the argument RTMDocumentExchangeResponse having the structure defined by the following table.

Name	Type	Occurs	Description
sMessage	string	1..1	

RTMDocumentRemoveRequirement

Marks requirements as removed from the Document.

Style

Document

Input (Literal)

The input of this method is the argument RTMDocumentRemoveRequirement having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nDocument-id	int	1..1	
nChapter-id	int	1..1	
sClassName	string	1..1	
nObject-id	int	1..1	

Output (Literal)

The output of this method is the argument RTMDocumentRequirementResponse having the structure defined by the following table.

Name	Type	Occurs	Description
sMessage	string	1..1	

runTestAsync

Run the async. test commmand

Style

Document

Input (Literal)

The input of this method is the argument runTestAsync having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
param-1	runTestAsyncInputData	1..1	

Output (Literal)

The output of this method is the argument runTestAsyncResponseResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	runTestAsyncResponse	1..1	

SendCMNotifications

Send notifications to CM

Style

Document

Input (Literal)

The input of this method is the argument SendCMNotifications having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
CMUserId	string	1..1	
CMPassword	string	1..1	

Output (Literal)

The output of this method is the argument CMNotificationsResult having the structure defined by the following table.

Name	Type	Occurs	Description
nValue	int	1..1	

TransitionObject

Transition requirement to the specified state

Style

Document

Input (Literal)

The input of this method is the argument TransitionObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	TransitObject	1..1	

Output (Literal)

The output of this method is the argument TransitionObjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
sPUIID	string	1..1	
sClassName	string	1..1	
sDescription	string	0..1	
sCreatedBy	string	1..1	
sModifiedBy	string	1..1	
nCreateTime	dateTime	1..1	
nModifiedTime	dateTime	1..1	
sTitle	string	0..1	
nObjectId	int	1..1	
nObjectVersionId	int	1..1	
sURL	string	1..1	
sState	string	1..1	
sOwner	string	1..1	

Remarks

Example of usage:

```
<urn:TransitionObject>
  <sessionId>1111111111</sessionId>
  <anObject>
    <sClassName>Component_Requirements</sClassName>
    <nObjectId>20</nObjectId>
    <sTransitionName>DELEGATE</sTransitionName>
    <aAttrList>
```

```
        <item>
            <sName>DEVELOPER</sName>
            <sValue>MICHAEL</sValue>
        </item>
    </aAttrList>
</anObject>
</urn:TransitionObject>
```

UnlinkObjects

Unlink multiple children from a parent object

Style

Document

Input (Literal)

The input of this method is the argument UnlinkObjects having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sParentClassName	string	1..1	
sParentPUIID	string	1..1	
sChildClassName	string	1..1	
aChildPUIIDs	StringList	1..1	
bOnlyDelete	boolean	0..1	

Output (Literal)

The output of this method is the argument UnlinkObjectsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	string	1..1	

Remarks

Example of usage:

```
<urn:UnlinkObjects>
  <sessionId>11111111</sessionId>
  <sParentClassName>Marketing_Requirements</sParentClassName>
  <sParentPUIID>MRKT_000020</sParentPUIID>
  <sChildClassName>Product_Requirements</sChildClassName>
  <aChildPUIIDs>
    <item>PROD_000023</item>
  </aChildPUIIDs>
  <bOnlyDelete>>false</bOnlyDelete>
</urn:UnlinkObjects>
```

Additional Info:

It is possible not only remove requirement links but also delete requirement links in UnLinkObjects, UnLinkObjectsEx and UnLinkObjectsExById web services. If bOnlyDelete input parameter specified as true, then link will be deleted. If false or not specified then link will be removed at all.

UnlinkObjectsEx

Unlink multiple children from a parent object

Style

Document

Input (Literal)

The input of this method is the argument UnlinkObjectsEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nRelationId	int	0..1	
sRelationName	string	0..1	
sParentPUID	string	1..1	
aChildPUIIDs	StringList	1..1	
bOnlyDelete	boolean	0..1	

Output (Literal)

The output of this method is the argument UnlinkObjectsExResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	string	1..1	

Remarks

Example of usage:

```
<urn:UnlinkObjectsEx>
  <sessionId>1111111111</sessionId>
  <nRelationId>5</nRelationId>
  <sParentPUID>MRKT_000021</sParentPUID>
  <aChildPUIIDs>
    <item>PROD_000023</item>
  </aChildPUIIDs>
  <bOnlyDelete>>false</bOnlyDelete>
</urn:UnlinkObjectsEx>
```

Additional Info:

It is possible not only remove requirement links but also delete requirement links in UnLinkObjects, UnLinkObjectsEx and UnLinkObjectsExById web services. If bOnlyDelete input parameter specified as true, then link will be deleted. If false or not specified then link will be removed at all.

UnlinkObjectsExById

Unlink multiple children from a parent object

Style

Document

Input (Literal)

The input of this method is the argument UnlinkObjectsExById having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
nRelationId	int	0..1	
sRelationName	string	0..1	
nParentId	int	1..1	
aChildIds	IdList	1..1	
bOnlyDelete	boolean	0..1	

Output (Literal)

The output of this method is the argument UnLinkResults having the structure defined by the following table.

Name	Type	Occurs	Description
sResult	string	1..1	
lLinkList	StringList	0..1	

Remarks

Example of usage:

```
<urn:UnlinkObjectsExById>
  <sessionId>11111111</sessionId>
  <nRelationId>5</nRelationId>
  <nParentId>55</nParentId>
  <aChildIds>
    <item>32</item>
  </aChildIds>
  <bOnlyDelete>>false</bOnlyDelete>
</urn:UnlinkObjectsExById>
```

Additional Info:

It is possible not only remove requirement links but also delete requirement links in UnLinkObjects, UnLinkObjectsEx and UnLinkObjectsExById web services. If bOnlyDelete input parameter specified as true, then link will be deleted. If false or not specified then link will be removed at all.

UpdateObject

Updates a RM requirement with specified requirement data

Style

Document

Input (Literal)

The input of this method is the argument UpdateObject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	

Output (Literal)

The output of this method is the argument UpdateObjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

UpdateObjectAttachment

Updates a RM requirement attachment

Style

Document

Input (Literal)

The input of this method is the argument UpdateObjectAttachment having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Name of requirement class
sPUID	string	1..1	PUID of specified requirement
sAttrName	string	1..1	File attachment attribute name
nFileIndex	integer	0..1	The index at which the file will be added to the file attachment attribute. Specify nFileIndex only if the file attachment attribute has the Allow multiple selections option enabled. To add an attachment at the last position, set nFileIndex higher than the last index.
oFile	FileAttachment	0..1	The file attachment to upload. To delete an attachment, clear sFullPath and sFileName.

Output (Literal)

The output of this method is the argument UpdateObjectAttachmentResponse having the structure defined by the following table.

Name	Type	Occurs	Description
results	string	1..1	

Example of usage:

Uploading a file attachment to a simple file attachment attribute

```
<urn:UpdateObjectAttachment>
  <sessionId>443452989</sessionId>
  <sClassName>MyClass</sClassName>
  <sPUID>MC_000001</sPUID>
  <sAttrName>ATTACHMENT</sAttrName>
  <oFile>
    <sFullPath>C:\temp\Design.docx</sFullPath>
    <sFilename>Design.docx</sFilename>
  </oFile>
</urn:UpdateObjectAttachment>
```

Uploading a file attachment to a multi file attachment attribute

```
<urn:UpdateObjectAttachment>
  <sessionId>443452989</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <sPUID>MRKT_000003</sPUID>
  <sAttrName>FILE_ATTACHMENT</sAttrName>
  <nFileIndex>0</nFileIndex>
  <oFile>
    <sFullPath>C:\temp\Design.docx</sFullPath>
    <sFilename>Design.docx</sFilename>
  </oFile>
</urn:UpdateObjectAttachment>
```

UpdateObjectEx

Updates a RM requirement with specified requirement data

Style

Document

Input (Literal)

The input of this method is the argument UpdateObjectEx having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
anObject	Requirement	1..1	
bDontGenerateAlfEvent	boolean	0..1	

Output (Literal)

The output of this method is the argument UpdateObjectExResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	RequirementOut	1..1	

UpdateTemplate

Update the template name, filename, and description fields and the template file.

Style

Document

Input (Literal)

The input of this method is the argument UpdateTemplate having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
docTemplate	Template	1..1	
encodedContent	base64Binary	0..1	

Output (Literal)

The output of this method is the argument UpdateTemplateResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	int	1..1	

UpdateTemplateFile

Replace the template file of the template specified by template id

Style

Document

Input (Literal)

The input of this method is the argument UpdateTemplateFile having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
id	int	1..1	
encodedContent	base64Binary	0..1	

Output (Literal)

The output of this method is the argument UpdateTemplateFileResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	int	1..1	

UploadImageForHTMLAttribute

Upload any file to the RM server

Style

Document

Input (Literal)

The input of this method is the argument UploadImageForHTMLAttribute having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
files	ImageList	1..1	

Output (Literal)

The output of this method is the argument UploadImageResponse having the structure defined by the following table.

Name	Type	Occurs	Description
aSucceed	ImageUploadResultList	1..1	
aFailed	ImageUploadResultList	1..1	

Remarks

Example of usage:

```
<urn:UploadImageForHTMLAttribute>
  <sessionId>1406221868</sessionId>
  <files>
    <item>
      <!--test2.gif is a local file on RM application server-->
      <sFullPath>c:\Temp\_a\test2.gif</sFullPath>
      <sFilename>test2.gif</sFilename>
    </item>
    <item>
      <sFilename>test1.png</sFilename>
  </files>
</UploadImageForHTMLAttribute>
<encodedContents>iVBORw0KGgoAAAANSUUhEUgAAADwAAAA1CAIAAACskR/tAAAAAXNSR
0IARs4c6QAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAAHxSURBVGH
D7ZQxcIwEEU5Fwf iPJyBI6TgABYBloouDTMumKGgIJL2S1ptstgSlsYEvfmTSXYV7bMte
/V4Q7p0K7p0K7p0K7p0Kz5AerX7rhHsPpLJ0mJGpWDYBMA1xda1g61PGZEWO7YJZuv802m
xlwkasyJGmKChkyGNagXEIBM0FBYhbcia1aVfIGvWh0nvD0l3tRv26MTW+nhHyXE6Xn4vJ
nwdQVWhWPq+/Uq6Nl/Xk23dNqFyuLnFBKtjZQR1H1QViqXJwN+w82DXkMpwXdvfL+5n1KP
bvzm4lcnFWGhECKoKpdJk6WfjPNCfrrU+3tyj8FeFK7nu3QnZnKkasf/0gqpCoTQ7nSHwo
5bRcldy2Q626H+nQ4UiJ92qjnR41RD2uINrsGePJT1UjLiVC6oKZdL8huH18k+caSXnxC3
2h0S8hYYwhYKqQpE0zfY3j046vm5cC8tc2HHnjyWAZT6oKhRjP7NxpvmW/zyHb9xf15fils
WgqlAiTZZxNrvx8Rxb8C33K9W30EAjQlBVKHwRZydrVpd+gaxZXfoFsmYtQl0MMkFDIU0a
```

```
gt58iP1N0NB5Jm0Q27UJZussThqDnzIibRCbVg1GjjEuTYjdawSTJjBVmhBjZgm2ziFPei
F06VZ06VZ06VZ06Va8ofTj8Q06Ch0Ds8230QAAAABJRU5ErkJggg==
</encodedContents>
  </item>
</files>
</urn:UploadImageForHTMLAttribute>
```

Complex Types

Name	Description
addObjectToCollectionResponse	
AppendRMDocumentExResult	
AppendRtmDocumentResult	
AsyncAppendRtmDocResponse	
AsyncCreateRtmDocResponse	
AsyncReplaceRtmDocResponse	
AttributeDef	
AttributeDefEx	
AttributeDefEx2	
AttributeDefExList	
AttributeDefEx2List	
AttributeDefInfo	
AttributeDefList	
AttributeDefSpecifier	
AttributeFormatDef	
Category	
CategoryInfo	
CategoryAssign	
CategoryList	
CategoryReq	
ClassDef	
ClassDefEx	
ClassDefExList	
ClassDefInfo	
ClassDefList	
ClassSpecifier	
CmdStatus	
CMPProjectInfo	
CollectionEx	
CollectionEx2	
CollectionEx2List	
CollectionExList	
CollectionInfo	

Name	Description
CollectionList	
CollectionSpecifier	
CollectionTransferRules	
CollectionTransferRulesEx	
CommandStatus	
CreateCollectionEx	
CreateChildDocumentData	
CreateRMDocumentExResult	
CreateRtmDocumentResult	
DatabaseList	
DocumentObject	
DocumentsFilter	
ECPObjInfo	
ECPObjInfo	
FileAttachment	
FindRequirementOut	
FormatDef	
getAsyncAppendRtmDocStatusResponse	
getAsyncCreateRtmDocStatusResponse	
getAsyncQueryRtmDocResponse	
getAsyncQueryRtmDocumentStatusResponse	
getAsyncReplaceRtmDocStatusResponse	
getAsyncStatusResponse	
GetAttributeDefInfoReply	
GetAttributeDefInfoRequest	
GetClassDefInfoReply	
GetClassDefInfoRequest	
GetCollectionInfoReply	
GetCollectionInfoRequest	
GetProjectInfoReply	
GetProjectInfoRequest	
GetUserAttributeDefInfoReply	
GetUserAttributeDefInfoRequest	
HierarchyInfo	
IdList	
ImageList	

Name	Description
ImageUploadResult	
ImageUploadResultList	
InRequirement	
InTransition	
Link	
LinkClassDef	
LinkClassDefList	
LinkList	
ListItemDef	
ListItems	
LoginExResultInfo	
NameValue	
NameValueList	
ns-chapterDetails	
Object	
ObjectDef	
ObjectDetails	
ObjectDetailsList	
ObjectsList	
ProjectEx	
ProjectExList	
ProjectInfo	
ProjectList	
ProjectSpecifier	
ProposeNewRequirementResult	
RelationList	
RelationWithObjects	
removeObjectFromCollectionResponse	
ReplaceRequirementList	
ReplaceRequirementOut	
Report	
Requirement	
RequirementFindList	
RequirementIdentifier	
RequirementIdentifierList	
RequirementOut	

Name	Description
Result	
RtmDocSnapshot	
RtmDocSnapshotOut	
RtmDocSnapshotOutList	
RtmDocumentProperties	
RtmDocumentPropertiesEx	
RtmDocumentWithCategory	
RtmSnapshotListEx	
RtmSnapshotPropertiesEx	
runTestAsyncInputData	
runTestAsyncResponse	
runTestAsyncResponseData	
Script	
SearchRequirement	
StatusMessage	
StatusMessageList	
StringList	
SyncXrefLink	
SyncXrefLinkList	
SystemInfo	
Template	
TemplateList	
testAsyncStatusResponse	
Transition	
Transitions	
TransitObject	
UserAttributeDefInfo	
UserValue	
UserValueList	

addObjectToCollectionResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
status	cmdStatusCode	1..1	

AppendRMDocumentExResult

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CommandStatus	1..1	

AppendRtmDocumentResult

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CommandStatus	1..1	

AsyncAppendRtmDocResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCmdToken	string	1..1	
info	RtmDocumentProperties	1..1	
status	CmdStatus	1..1	

AsyncCreateRtmDocResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCmdToken	string	1..1	
info	RtmDocumentProperties	1..1	
status	CmdStatus	1..1	

AsyncReplaceRtmDocResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCmdToken	string	1..1	
info	RtmDocumentProperties	1..1	
status	CmdStatus	1..1	

AttributeDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nTypeId	int	1..1	
sTypeName	string	1..1	
sDisplayName	string	1..1	
sDefaultValue	string	0..1	
bMandatory	boolean	1..1	
bEditable	boolean	1..1	
bUnique	boolean	1..1	
bVisible	boolean	1..1	
bMultiple	boolean	1..1	
aFormat	FormatDef	1..1	
aValues	StringList	0..1	

AttributeDefEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nTypeId	int	1..1	
sTypeName	string	1..1	
sDisplayName	string	1..1	
sDefaultValue	string	0..1	
bMandatory	boolean	1..1	
bEditable	boolean	1..1	
bUnique	boolean	1..1	
bVisible	boolean	1..1	
bMultiple	boolean	1..1	
bImplicit	boolean	1..1	
bHtml	boolean	0..1	
aFormat	FormatDef	1..1	
aValues	StringList	0..1	


AttributeDefEx2

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
nId	int	1..1	Attribute identifier
sName	string	1..1	Attribute name
nTypeId	int	1..1	Attribute type identifier
sTypeName	string	1..1	Attribute type name
sDisplayName	string	1..1	Attribute display name
sDefaultValue	string	0..1	Attribute default value
bMandatory	boolean	1..1	Flag, indicating if attribute is mandatory
bEditable	boolean	1..1	Flag, indicating if attribute is editable
bUnique	boolean	1..1	Flag, indicating if attribute values must be unique
bVisable	boolean	1..1	Flag, indicating if attribute is visible in RM Browser
bMultiple	boolean	1..1	Flag, indicating if attribute can have multiple values
bImplicit	boolean	1..1	Flag, indicating if attribute is implicit
bHtml	boolean	0..1	Flag, indicating if attribute is HTML enabled
aFormat	FormatDef	1..1	Attribute format properties
aValues	ListItems	0..1	List of attribute values

AttributeDefExList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	AttributeDefEx	0..*	

AttributeDefEx2List

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	AttributeDefEx2	0..*	Attribute properties


AttributeDefInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nTypeId	int	1..1	
sTypeName	string	1..1	
sDisplayName	string	1..1	
sDefaultValue	string	1..1	
bMandatory	boolean	1..1	
bEditable	boolean	1..1	
bUnique	boolean	1..1	
bVisible	boolean	1..1	
bMultiple	boolean	1..1	
bImplicit	boolean	1..1	
bHtml	boolean	1..1	
aFormat	AttributeFormatDef	1..1	
aValues	StringList	0..1	

AttributeDefList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	AttributeDef	0..*	

AttributeDefSpecifier

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
Type	IdNameEnumeration	1..1	Possible values: 0 - then attribute will be specified by id 1 - then attribute will be specified by name
Value	IdNameUnion	1..1	id or attribute name(depending on previous setting)
CategoryInfo	CategoryInfo	0..1	Category information for filtering values

AttributeFormatDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nMaxLen	int	1..1	
nDisplayLen	int	1..1	
sCase	string	1..1	
sTemplate	string	1..1	
sMinValue	string	1..1	
sMaxValue	string	1..1	

Category

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	long	1..1	
nType	long	1..1	
sName	string	1..1	
sDescription	string	0..1	
sDisplayName	string	1..1	

Remarks

Additional info:

'sDisplayName' value of category name in the output will be specified with '\' separator, e.g. 'RMDEMO\Availability\Cost'

CategoryInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
CategoryId	long	0..1	Category id, default value = -1, means that no filter set by category id
DisplayName	string	0..1	Category name

CategoryAssign

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nCategoryId	long	0..1	
sDisplayName	string	0..1	

CategoryList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Category	0..*	

CategoryReq

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	
sDescription	string	0..1	
nParentCategoryId	long	1..1	
bInheritAccessRightsFromParentCategory	bool	0..1	

ClassDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	

ClassDefEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nDescriptionAttrId	int	1..1	
nTitleAttrId	int	1..1	
nType	int	1..1	

ClassDefExList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ClassDefEx	0..*	

ClassDefInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nDescriptionAttrId	int	1..1	
nTitleAttrId	int	1..1	
nType	int	1..1	

ClassDefList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ClassDef	0..*	

ClassSpecifier

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
Type	IdNameEnumeration	1..1	Possible values: 0 - then class will be specified by ID 1 - then class will be specified by name
Value	IdNameUnion	1..1	ID or class name (depending on previous setting)

CmdStatus

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
status	int	1..1	
msgs	StatusMessageList	0..1	

CMProjectInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sProductName	string	1..1	
sProjectName	string	1..1	

CollectionEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sType	string	1..1	
sName	string	1..1	
bDocument	boolean	1..1	
bBaseline	boolean	1..1	
sURL	string	1..1	

CollectionEx2

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sType	string	1..1	
sName	string	1..1	
bDocument	boolean	1..1	
bBaseline	boolean	1..1	
transferRules	CollectionTransferRules	1..1	

CollectionEx2List

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	CollectionEx2	0..*	

CollectionExList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	CollectionEx	0..*	

CollectionInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sType	string	1..1	
sName	string	1..1	
bDocument	boolean	1..1	
bBaseline	boolean	1..1	
transferRules	CollectionTransferRules	1..1	

CollectionList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	string	0..*	

CollectionSpecifier

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
Type	IdNameEnumeration	1..1	
Value	IdNameUnion	1..1	

CollectionTransferRules

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
bTransferToChild	boolean	1..1	
bDeleteFromParent	boolean	1..1	
bTransferBackToParent	boolean	1..1	

CollectionTransferRulesEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
bTransferToChild	boolean	0..1	
bDeleteFromParent	boolean	0..1	
bTransferBackToParent	boolean	0..1	
bObjectsAddOrRemove	boolean	0..1	

CommandStatus

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nStatusCode	int	1..1	
sMessage	string	0..1	

CreateCollectionEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	
sDescription	string	0..1	
transferRules	CollectionTransferRules	0..1	

CreateChildDocumentData

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	
nMasterDocId	int	0..1	
sMasterDocName	string	0..1	
categoryInfo	CategoryAssign	0..1	

CreateRMDocumentExResult

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CommandStatus	1..1	

CreateRtmDocumentResult

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CommandStatus	1..1	

DatabaseList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	string	0..*	

DocumentObject

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
iid	int	1..1	
objectId	int	1..1	
classId	int	1..1	

DocumentsFilter

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCreatedBy	string	0..1	
sModifiedBy	string	0..1	
sOwner	string	0..1	
nCreateTime	dateTime	0..1	
nCreateTimeAfter	dateTime	0..1	
nCreateTimeBefore	dateTime	0..1	
nModifiedTime	dateTime	0..1	
nModifiedTimeAfter	dateTime	0..1	
nModifiedTimeBefore	dateTime	0..1	
bIsBeforeAndAfterIncludeSpecifiedTime	boolean	0..1	
sCategory	string	0..1	
bIncludeSubCategories	boolean	0..1	
bIncludeMasterDocuments	boolean	0..1	
bIncludeChildDocuments	boolean	0..1	

ECPObjInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
classId	int	0..1	
className	string	0..1	
PUID	string	0..1	
objectId	int	0..1	


ECPsObjectInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
iECPsClassId	int	1..1	
iECPsObjectId	int	1..1	

FileAttachment

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sFullPath	string	0..1	
sFilename	string	1..1	
nModifiedTime	dateTime	0..1	
nChecksum	long	0..1	
encodedContents	base64Binary	0..1	

FindRequirementOut

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
aAttrList	NameValueList	1..1	

FormatDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nMaxLen	int	1..1	
nDisplayLen	int	1..1	
sCase	string	0..1	
sTemplate	string	0..1	
sMinValue	string	0..1	
sMaxValue	string	0..1	

getAsyncAppendRtmDocStatusResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CmdStatus	1..1	

getAsyncCreateRtmDocStatusResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CmdStatus	1..1	

getAsyncQueryRtmDocResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCmdToken	string	1..1	
rtmDoc	RtmDocument	1..1	
status	CmdStatus	1..1	

getAsyncQueryRtmDocumentStatusResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
rtmDoc	RtmDocument	1..1	
status	CmdStatus	1..1	

getAsyncReplaceRtmDocStatusResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
status	CmdStatus	1..1	

getAsyncStatusResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
status	CmdStatus	1..1	

GetAttributeDefInfoReply

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
attributeDefInfo	AttributeDefInfo	1..1	

GetAttributeDefInfoRequest

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sessionId	string	1..1	
classSpecifier	ClassSpecifier	1..1	
attributeDefSpecifier	AttributeDefSpecifier	1..1	

GetClassDefInfoReply

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
classDefInfo	ClassDefInfo	1..1	

GetClassDefInfoRequest

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sessionId	string	1..1	
classSpecifier	ClassSpecifier	1..1	


GetCollectionInfoReply

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
collectionInfo	CollectionInfo	1..1	

GetCollectionInfoRequest

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sessionId	string	1..1	
collectionSpecifier	CollectionSpecifier	1..1	

GetProjectInfoReply

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
projectInfo	ProjectInfo	1..1	

GetProjectInfoRequest

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
projectSpecifier	ProjectSpecifier	1..1	
sUserId	string	1..1	
sPassword	string	1..1	
sDatabase	string	1..1	

GetUserAttributeDefInfoReply

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
attributeDefInfo	UserAttributeDefInfo	1..1	

GetUserAttributeDefInfoRequest

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sessionId	string	1..1	
classSpecifier	ClassSpecifier	1..1	
attributeDefSpecifier	AttributeDefSpecifier	1..1	

HierarchyInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
parent	RequirementIdentifier	0..1	A set of object attributes values of parent requirement
order	int	1..1	Order of the object in scope of parent

IdList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	int	0..*	

ImageList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	FileAttachment	0..*	

ImageUploadResult

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sFilename	string	0..1	
result	string	0..1	

ImageUploadResultList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ImageUploadResult	0..*	

InRequirement

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sClassName	string	1..1	
nObjectId	long	0..1	
sPUID	string	0..1	

InTransition

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nTransitionId	long	0..1	
sTransitionName	string	0..1	

Link

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sPrimaryClassName	string	1..1	
nPrimaryObjectId	int	1..1	
sPrimaryPUID	string	1..1	
sSecondaryClassName	string	1..1	
nSecondaryObjectId	int	1..1	
sSecondaryPUID	string	1..1	
sRelationName	string	1..1	
nRelationId	int	1..1	
sRelationStatus	string	0..1	

LinkClassDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
sRelationName	string	1..1	

LinkClassDefList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	LinkClassDef	0..*	

LinkedList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Link	0..*	

ListItemDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	Attribute value name
bDefault	boolean	0..1	Flag, indicating if value is default
bDeleted	boolean	0..1	Flag, indicating if value is deleted

ListItems

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ListItemDef	0..*	List of attribute value properties

LoginExResultInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sessionId	string	1..1	
bIsAdmin	boolean	1..1	

NameValue

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	Attribute name
sValue	string	0..1	<p>The value the attribute can receive.</p> <p>URL attribute: The value must be an escaped HTML link. If the attribute supports several URLs, specify all URLs with this value.</p> <p><u>Example:</u> <pre>Open Text</pre> translates to <pre>&lt;a href=&quot;https://www.opentext.com/ &quot; target=&quot;_blank&quot;&gt;Open Text&lt;/a&gt;</pre> </p>

NameValueList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	NameValue	0..*	

ns-chapterDetails

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	
sDescription	string	0..1	

Object

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sClassName	string	1..1	
nObjectId	long	1..1	
sReason	string	0..1	

ObjectDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sClassName	string	1..1	
nObjectId	int	1..1	

ObjectDetails

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nObjectId	int	1..1	
sPUID	string	1..1	
sRelationStatus	string	0..1	
ObjAttrs	NameValueList	0..1	

ObjectDetailsList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ObjectDetails	0..*	

ObjectsList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Object	0..*	

ProjectEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
sDescription	string	0..1	
bLockStatus	boolean	1..1	
sVersion	string	0..1	
bAssociatedCM	boolean	1..1	

ProjectExList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ProjectEx	0..*	

ProjectInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
sDescription	string	1..1	
bLockStatus	boolean	1..1	
sVersion	string	1..1	
bAssociatedCM	boolean	1..1	

ProjectList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	string	0..*	

ProjectSpecifier

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
Type	IdNameEnumeration	1..1	
Value	IdNameUnion	1..1	

ProposeNewRequirementResult

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
result	RequirementOut	1..1	

Example:

```
<ns:ProposeNewRequirementResult>
  <result>
    <sPUID>MRKT_000055</sPUID>
    <sClassName>Marketing_Requirements</sClassName>
    <sDescription>new MRKT 15 - text</sDescription>
    <sCreatedBy>EPHOTO</sCreatedBy>
    <sModifiedBy>EPHOTO</sModifiedBy>
    <nCreateTime>2014-12-12T14:47:38+02:00</nCreateTime>
    <nModifiedTime>2014-12-12T14:48:09+02:00</nModifiedTime>
    <sTitle>new MRKT 15 - title</sTitle>
    <nObjectId>91</nObjectId>
    <nObjectVersionId>1</nObjectVersionId>
    <aAttrList/>
  </result>
</ns:ProposeNewRequirementResult>
```

RelationList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RelationWithObjects	0..*	

RelationWithObjects

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nRelationType	int	1..1	
nRelationId	int	1..1	
sRelationName	string	1..1	
ObjList	ObjectDetailsList	0..1	

removeObjectFromCollectionResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
status	cmdStatusCode	1..1	

ReplaceRequirementList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	ReplaceRequirementOut	0..*	

ReplaceRequirementOut

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
obj	RequirementOut	1..1	
eStatus	reqStatusCode	1..1	

Report

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCategoryName	string	1..1	
sReportName	string	1..1	

Requirement

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sPUID	string	0..1	
sClassName	string	0..1	
sDescription	string	0..1	
sCreatedBy	string	0..1	
sModifiedBy	string	0..1	
nCreateTime	dateTime	0..1	
nModifiedTime	dateTime	0..1	
sTitle	string	0..1	
nObjectId	int	0..1	
aAttrList	NameValueList	0..1	

RequirementFindList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	FindRequirementOut	0..*	

RequirementIdentifier

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sPUID	string	0..1	
sClassName	string	1..1	
nObjectId	int	0..1	

RequirementIdentifierList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RequirementIdentifier	0..*	

RequirementList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RequirementOut	0..*	

RequirementOut

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sPUIID	string	1..1	
sClassName	string	1..1	
sDescription	string	0..1	
sCreatedBy	string	1..1	
sModifiedBy	string	1..1	
nCreateTime	dateTime	1..1	
nModifiedTime	dateTime	1..1	
sTitle	string	1..1	
nObjectId	int	1..1	
nObjectVersionId	int	1..1	
sURL	string	0..1	
aAttrList	NameValueList	0..1	

Result

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nStatusCode	int	1..1	
sMessage	string	0..1	

RtmDocSnapshot

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	Snapshot name
nDocumentId	int	1..1	Document Id
sDescription	string	0..1	Snapshot description
sType	string	0..1	Snapshot type: MINOR, MAJOR, NONE If no value provided, then NONE is used

RtmDocSnapshotOut

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nDocumentId	int	1..1	
sUrl	string	1..1	

RtmDocSnaphotOutList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RtmDocSnapshotOut	0..*	

RtmDocument

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
sContent	string	0..1	

RtmDocumentList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RtmDocumentProperties	0..*	

RtmDocumentListEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RtmDocumentPropertiesEx	0..*	

RtmDocumentProperties

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	0..1	
sName	string	1..1	
sURL	string	0..1	

RtmDocumentPropertiesEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
bIsChild	boolean	1..1	If true, the document is a child document.
bIsMaster	boolean	1..1	If true, the document is a parent document.
nId	int	0..1	The ID of the document
nParentId	int	1..1	The ID of the parent document (if document is a child document), or 0
nTimeCreated	dateTime	0..1	The date and time the document was created
nTimeModified	dateTime	0..1	The date and time the document was last modified
sCategory	string	0..1	The full path of the category where the document is located
sCreatedBy	string	0..1	The user who created the document
sDescription	string	0..1	The description of the document
sModifiedBy	string	0..1	The last user who modified the document
sName	string	1..1	The name of the document
snapshots	RtmSnapshotListEx	0..1	The snapshots associated with the document
sOwner	string	0..1	The owner of the workflow state
sState	string	0..1	The workflow state of the document
sURL	string	0..1	The URL under which the document can be accessed in a web browser

RtmDocumentWithCategory

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
info	RtmDocumentProperties	1..1	
sContent	string	0..1	
categoryInfo	CategoryAssign	0..1	
bIsMaster	boolean	0..1	

RtmSnapshotListEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	RtmSnapshotPropertiesEx	0..*	

RtmSnapshotPropertiesEx

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	0..1	The ID of the snapshot
nTimeCreated	dateTime	0..1	The date and time the document was created
sCreatedBy	string	0..1	The user who created the document
sDescription	string	0..1	The description of the document
sName	string	1..1	The name of the document
sURL	string	0..1	The URL under which the document can be accessed in a web browser

runTestAsyncInputData

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
reflectInt	int	1..1	
sReflectString	string	0..1	
logInterval	int	1..1	
timeSpan	int	1..1	
createFailure	int	1..1	

runTestAsyncResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sCmdToken	string	0..1	
data	runTestAsyncResponseData	1..1	
status	CmdStatus	1..1	

runTestAsyncResponseData

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
reflectInt	int	1..1	
sReflectString	string	0..1	
times56	int	1..1	

Script

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sScriptText	string	0..1	
anSavedReport	Report	0..1	
anRunTimeParameters	NameValueList	0..1	

SearchRequirement

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sTitle	string	0..1	
sDescription	string	0..1	
nCreateTime	dateTime	0..1	
nModifiedTime	dateTime	0..1	
aAttrList	NameValueList	0..1	

StatusMessage

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
msgId	int	1..1	
level	int	1..1	
parms	StringList	0..1	

StatusMessageList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	StatusMessage	0..*	

StringList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	string	0..*	

SyncXrefLink

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sForeignId	string	0..1	
nRMClassId	int	0..1	
sRMPUID	string	0..1	
sForeignToolName	string	1..1	

SyncXrefLinkList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	SyncXrefLink	0..*	


SystemInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
sDBCharSet	string	0..1	
sClientCharSet	string	0..1	
sRMMajorVersion	string	0..1	
sRMMinorVersion	string	0..1	
sRMAdminMajor	string	0..1	
sRMAdminMinor	string	0..1	
sOperatingSystem	string	0..1	
sOperatingSystemVersion	string	0..1	
sDatabase	string	0..1	
sDatabaseVersion	string	0..1	

Template

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	0..1	
sName	string	1..1	
sFileName	string	0..1	
sDescription	string	0..1	
sCreatedBy	string	0..1	
sModifiedBy	string	0..1	
nLastModified	dateTime	0..1	

TemplateList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Template	0..*	

testAsyncStatusResponse

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
data	runTestAsyncResponseData	1..1	
status	CmdStatus	1..1	

Transition

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	long	1..1	
sName	string	1..1	
sDescription	string	1..1	
sToState	string	1..1	

Transitions

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Transition	0..*	


TransitObject

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
sClassName	string	1..1	
nObjectId	long	0..1	
sPUID	string	0..1	
nTransitionId	long	0..1	
sTransitionName	string	0..1	
aAttrList	NameValueList	0..1	

UserAttributeDefInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nId	int	1..1	
sName	string	1..1	
nTypeId	int	1..1	
sTypeName	string	1..1	
sDisplayName	string	1..1	
aDefaultValues	UserValueList	1..1	
bMandatory	boolean	1..1	
bEditable	boolean	1..1	
bUnique	boolean	1..1	
bVisable	boolean	1..1	
bMultiple	boolean	1..1	
bImplicit	boolean	1..1	
bHtml	boolean	1..1	
aFormat	AttributeFormatDef	1..1	
aValues	UserValueList	1..1	


UserValue

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
 SEQUENCE		1..1	
nId	integer	1..1	
sName	string	1..1	
sDisplayName	string	1..1	
sFirstName	string	0..1	
sLastName	string	0..1	
sPhone	string	0..1	
sFax	string	0..1	
sEmail	string	0..1	
sPager	string	0..1	
sLocation	string	0..1	
sMobilePhone	string	0..1	
sType	string	1..1	

UserValueList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	UserValue	0..*	

Simple Types

Name	Description
cmdStatusCode	
IdNameEnumeration	
IdNameUnion	
reqStatusCode	
statusCode	

cmdStatusCode

Derived By

Restricting string

Enumeration

Value	Description
SUCCESS	
FAILURE	
COLLECTION-NOT-FOUND	

IdNameEnumeration

Derived By

Restricting string

Enumeration

Value	Description
ID	
NAME	

IdNameUnion

Derived By

Restricting string

reqStatusCode

Derived By

Restricting string

Enumeration

Value	Description
eCREATED	
eREPLACED	
eUPDATED	
eNOCHANGE	

statusCode

Derived By

Restricting string

Enumeration

Value	Description
STARTING	
ACTIVE	
FINISHED	
ERR	
NOT-FOUND	

Chapter 224.4 (12.12.1)

RM Admin Web Service

General	324
Methods	325
Complex Types	360
Appendix	379

General

The Dimensions RM Web Service contains all administrative functions and types.

The default URL is *http://localhost:8080/rtmBrowser/WebServices*. Adjust protocol, server name and port to match your environment.

Methods

Name	Description
AssignCategoryToGroup	Assign category access to the group
AssignGroupToProject	Assign group to the project
AssignUsersInGroupPerCategory	Assign/revoke users access in group per category
AssignUserToGroup	Assign user to the group
AssignUserToProject	Assign user to the project
CheckPermission	Checks permission for the specified action
CopyGroup	Copy group with all project assignments and permissions
CopyUser	Copy user with group, project assignments and permissions
CreateGroup	Create a new group
CreateUser	Create a new user
CreateLDAPUser	Create a new LDAP user
CreateSSOUser	Create a new SSO user
DeleteGroup	Delete a group
DeleteUser	Delete a user
GetGroupInfo	Get information about group
GetGroups	Get list of groups
GetLoginSources	Get available login sources on the RM server
GetUserInfo	Get information about user
GetUsers	Get list of all users
GrantPermissionToGroup	Grant permission to group for some action
GrantPermissionToUser	Grant permission to user for some action
InheritPermissionForUser	Inherit permission to user for some action
RevokePermissionFromGroup	Revoke permission from group for some action
RevokePermissionFromUser	Revoke permission from user for some action
UnassignCategoryFromGroup	Un-assign category from the group
UnassignGroupFromProject	Un-assign group from the project
UnassignUserFromGroup	Un-assign user from the group
UnassignUserFromProject	Un-assign user from the project
UpdateListAttributeValues	Change list attribute values in attribute definition
UpdateUser	Update user information and account properties
UpdateUserAttributeValues	Allows to setup default values for the user attributes

AssignCategoryToGroup

Assign category access to the group

Style

Document

Input (Literal)

The input of this method is the argument AssignCategoryToGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	
sCategoryName	string	1..1	
bIncludeSubcategories	boolean	0..1	

Output (Literal)

The output of this method is the argument AssignCategoryToGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:AssignCategoryToGroup>
  <sessionId>1</sessionId>
  <sGroupName>read</sGroupName>
  <sProjectName>RMDEMO</sProjectName>
  <sCategoryName>RMDEMO/Functional</sCategoryName>
  <bIncludeSubcategories>true</bIncludeSubcategories>
</urn:AssignCategoryToGroup>
```

AssignGroupToProject

Assign group to the project

Style

Document

Input (Literal)

The input of this method is the argument AssignGroupToProject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	
bForceUserAssignment	boolean	0..1	

Output (Literal)

The output of this method is the argument AssignGroupToProjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:AssignGroupToProject>  
  <sessionId>1</sessionId>  
  <sGroupName>test1</sGroupName>  
  <sProjectName>RMDEMO</sProjectName>  
</urn:AssignGroupToProject>
```

AssignUsersInGroupPerCategory

Assign/revoke users access in group per category

Style

Document

Input (Literal)

The input of this method is the argument AssignUsersInGroupPerCategory having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sProjectName	string	1..1	
sCategoryName	string	1..1	
sGroupName	string	1..1	
excludeUsers	UserNameList	0..1	
includeUsers	UserNameList	0..1	

Output (Literal)

The output of this method is the argument AssignUsersInGroupPerCategoryResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:AssignUsersInGroupPerCategory>
  <sessionId>1433504615</sessionId>
  <sProjectName>RMDemo</sProjectName>
  <sCategoryName>RMDemo\123\test\some category</sCategoryName>
  <sGroupName>WRITE</sGroupName>
  <excludeUsers>
    <item>support</item>
    <item>test</item>
  </excludeUsers>
  <includeUsers>
    <item>ephoto</item>
    <item>sales</item>
  </includeUsers>
</urn:AssignUsersInGroupPerCategory>
```

Users in excludeUsers section will be unassign in WRITE group in specified category.

Users in includeUsers section will be assigned to WRITE group in specified category.

If Inherit access rights from parent category turned on, after applying changes it will be unchecked.

Additional Info:

'sCategoryName value of category name can be specified as:

- a** category full name, separated by '/' or '\'
- b** category name(only in case when name is unique);

AssignUserToGroup

Assign user to the group

Style

Document

Input (Literal)

The input of this method is the argument AssignUserToGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sGroupName	string	1..1	

Output (Literal)

The output of this method is the argument AssignUserToGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:AssignUserToGroup>
  <sessionId>1</sessionId>
  <sUserName>ephoto</sUserName>
  <sGroupName>read</sGroupName>
</urn:AssignUserToGroup>
```

AssignUserToProject

Assign user to the project

Style

Document

Input (Literal)

The input of this method is the argument AssignUserToProject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sUserName	string	1..1	User name
sGroupName	string	1..1	Group name
sProjectName	string	1..1	Project name
bDontAutoAssign	boolean	0..1	Default is "false". If the value is "false", the web service automatically assigns the specified user to the group categories, otherwise the user will not be assigned.

Output (Literal)

The output of this method is the argument AssignUserToProjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

- 1 Assign user to project in the group with automatical assignment to the group categories

```
<urn:AssignUserToProject>
  <sessionId>session id</sessionId>
  <sUserName>user1</sUserName>
  <sGroupName>Group1</sGroupName>
  <sProjectName>RMDEMO</sProjectName>
  <bDontAutoAssign>>false</bDontAutoAssign>
</urn:AssignUserToProject>
```

- 2 Assign user to project in the group without assigning to the group categories

```
<urn:AssignUserToProject>
  <sessionId>session id</sessionId>
  <sUserName>user2</sUserName>
  <sGroupName>Group1</sGroupName>
  <sProjectName>RMDEMO</sProjectName>
  <bDontAutoAssign>>true</bDontAutoAssign>
</urn:AssignUserToProject>
```

CheckPermission

Checks permission for the specified action.

Style

Document

Input (Literal)

The input of this method is the argument CheckPermission having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sResource	string	1..1	Specifies the permission set. For details, see chapter "Resource and Action Parameters" on page 379 .
sAction	string	1..1	Specifies the permission to check. For details, see chapter "Resource and Action Parameters" on page 379 .
sObject	string	0..1	Specifies the class name or relationship name.
sCategory	string	0..1	Specifies the category. Can be specified by full path (separated with '\' or '/') or by category ID.

Output (Literal)

The output of this method is the argument CheckPermissionResponse having the structure defined by the following table.

Name	Type	Occurs	Description
bAllowed	boolean	1..1	<ul style="list-style-type: none"> ▪ True: permission is granted ▪ False: permission is denied

Remarks

Example of usage:

```
<urn:CheckPermission>
  <sessionId>1111111111</sessionId>
  <sResource>Classes</sResource>
  <sAction>Read</sAction>
  <sObject>Marketing_Requirements</sObject>
  <sCategory>RMDEMO\Support\Maintenance</sCategory>
</urn:CheckPermission>
```

CopyGroup

Copy group with all project assignments and permissions

Style

Document

Input (Literal)

The input of this method is the argument CopyGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sGroupName	string	1..1	Name of new group
sSrcGroupName	string	1..1	Name of source group
anGroupInfo	GroupInfo	0..1	Additional group information

Output (Literal)

The output of this method is the argument CopyGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

The CopyGroup call provides a method for copying existing group with all project assignments and permissions.

Example of usage:

1 Create copy of group:

```
<urn:CopyGroup>
  <sessionId>1472146937</sessionId>
  <sGroupName>REPORTERS</sGroupName>
  <sSrcGroupName>icDemo Users</sSrcGroupName>
</urn:CopyGroup>
```

2 Create copy of group and provide description:

```
<urn:CopyGroup>
  <sessionId>1472146937</sessionId>
  <sGroupName>REPORTERS</sGroupName>
  <sSrcGroupName>icDemo Users</sSrcGroupName>
  <anGroupInfo>
    <sDescription>Reporters group</sDescription>
  </anGroupInfo>
</urn:CopyGroup>
```

CopyUser

Copy user with group, project assignments and permissions

Style

Document

Input (Literal)

The input of this method is the argument CopyUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sUserName	string	1..1	New user name
sSrcUserName	string	1..1	Source user name
sPassword	string	0..1	Password
anUserInfo	UserInfo	0..1	Additional user info

Output (Literal)

The output of this method is the argument CopyUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

The CopyUser call provides a method for copying existing user with all group, category, project assignments and permissions.

Example of usage:

- 1 Create Full copy of user (including all information - firstname, lastname, phone, etc.)

```
<urn:CopyUser>
  <sessionId>1472146937</sessionId>
  <sUserName>Joe1</sUserName>
  <sSrcUserName>joe</sSrcUserName>
  <sPassword>123456</sPassword>
  <anUserInfo>
  </anUserInfo>
</urn:CopyUser>
```

- 2 Create copy of user and override some information from source user:

```
<urn:CopyUser>
  <sessionId>1472146937</sessionId>
  <sUserName>Joe1</sUserName>
  <sSrcUserName>joe</sSrcUserName>
  <sPassword>123456</sPassword>
  <anUserInfo>
    <sFirstName>John</sFirstName>
  </anUserInfo>
</urn:CopyUser>
```

```
<sLastName>Pettor</sLastName>  
<sPhone>3558832224</sPhone>  
<sEmail>jpettor@mail.com</sEmail>  
</anUserInfo>  
</urn:CopyUser>
```

CreateGroup

Create a new group

Style

Document

Input (Literal)

The input of this method is the argument CreateGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
anGroupInfo	GroupInfo	0..1	

Output (Literal)

The output of this method is the argument CreateGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:CreateGroup>
  <sessionId>1</sessionId>
  <sGroupName>read</sGroupName>
  <anGroupInfo>
    <sDescription>Some description</sDescription>
  </anGroupInfo>
</urn:CreateGroup>
```


CreateUser

Create a new user

Style

Document

Input (Literal)

The input of this method is the argument CreateUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sPassword	string	0..1	
anUserInfo	UserInfo	0..1	

Output (Literal)

The output of this method is the argument CreateUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:CreateUser>
  <sessionId>1</sessionId>
  <sUserName>Teddy</sUserName>
  <sPassword>nice</sPassword>
  <anUserInfo>
    <bPwdNeverExpires>true</bPwdNeverExpires>
  </anUserInfo>
</urn:CreateUser>
```

CreateLDAPUser

Create a new LDAP user

Style

Document

Input (Literal)

The input of this method is the argument CreateLDAPUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sUserName	string	1..1	User name
anUserInfo	UserInfoExt	0..1	User information

Output (Literal)

The output of this method is the argument CreateLDAPUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	Result of LDAP user creation

Remarks

The CreateLDAPUser call provides a method for creation new LDAP user.

Created users will have LDAP type in RM(as it is not differentiated SSO or LDAP).

For such user it is not possible to change password settings, except (bAccountDisabled).

Example of usage:

```
<urn:CreateLDAPUser>
<sessionId>11111111</sessionId>
  <sUserName>Peter</sUserName>
  <anUserInfo>
    <sFirstName>Peter</sFirstName>
    <sLastName>Oneil</sLastName>
  </anUserInfo>
</urn: CreateLDAPUser>
```

CreateSSOUser

Create a new SSO user

Style

Document

Input (Literal)

The input of this method is the argument CreateSSOUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sUserName	string	1..1	User name
anUserInfo	UserInfoExt	0..1	User information

Output (Literal)

The output of this method is the argument CreateSSOUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	Result of SSO user creation

Remarks

The CreateSSOUser call provides a method for creation new SSO user.

Created users will have LDAP type in RM(as it is not differentiated SSO or LDAP).

For such user it is not possible to change password settings, except (bAccountDisabled).

Example of usage:

```
<urn:CreateSSOUser>
  <sessionId>111111111</sessionId>
  <sUserName>John</sUserName>
  <anUserInfo>
    <sFirstName>John</sFirstName>
    <sLastName>Harrison</sLastName>
  </anUserInfo>
</urn: CreateSSOUser>
```

DeleteGroup

Delete a group

Style

Document

Input (Literal)

The input of this method is the argument DeleteGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	

Output (Literal)

The output of this method is the argument DeleteGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:DeleteGroup>  
  <sessionId>1</sessionId>  
  <sGroupName>TestGroup</sGroupName>  
</urn:DeleteGroup>
```

DeleteUser

Delete a user

Style

Document

Input (Literal)

The input of this method is the argument DeleteUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	

Output (Literal)

The output of this method is the argument DeleteUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:DeleteUser>  
  <sessionId>1</sessionId>  
  <sUserName>testUser</sUserName>  
</urn:DeleteUser>
```

GetGroupInfo

Get information about group

Style

Document

Input (Literal)

The input of this method is the argument GetGroupInfo having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	

Output (Literal)

The output of this method is the argument GroupDetailInfo having the structure defined by the following table.

Name	Type	Occurs	Description
sGroupName	string	0..1	
anGroupInfo	GroupInfo	1..1	
aProjects	StringList	1..1	
aUsers	StringList	1..1	

Remarks

Example of usage:

```
<urn:GetGroupInfo>  
  <sessionId>1</sessionId>  
  <sGroupName>Administrators</sGroupName>  
</urn:GetGroupInfo>
```

GetGroups

Get list of groups

Style

Document

Input (Literal)

The input of this method is the argument GetGroups having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument GetGroupsResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	GroupList	1..1	

Remarks

Example of usage:

```
<urn:GetGroups>  
  <sessionId>1</sessionId>  
</urn:GetGroups>
```

GetLoginSources

Get available login sources on the RM server

Style

Document

Input (Literal)

The input of this method is the argument GetLoginSources having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	

Output (Literal)

The output of this method is the argument GetLoginSourcesResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	LoginSources	1..1	

Remarks

The GetLoginSources call provides a method to get login sources that are currently available on the RM server.

Example of usage:

```
<urn:GetLoginSources>  
  <sessionId>1464698712</sessionId>  
</urn:GetLoginSources>
```


GetUserInfo

Get information about user

Style

Document

Input (Literal)

The input of this method is the argument GetUserInfo having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
bIncludeAdditionalInfo	boolean	0..1	
bIncludeGroups	boolean	0..1	
bIncludeProjects	boolean	0..1	

Output (Literal)

The output of this method is the argument GetUserInfoResult having the structure defined by the following table.

Name	Type	Occurs	Description
sUserName	string	0..1	
sUserLoginType	string	0..1	
sFirstName	string	0..1	
sLastName	string	0..1	
sEmail	string	0..1	
bAccountDisabled	boolean	1..1	
anAdditionalUserInfo	AdditionalUserInfo	0..1	
aGroups	StringList	0..1	
aProjects	NameValueList	0..1	

Remarks

Example of usage:

```
<urn:GetUserInfo>
  <sessionId>1</sessionId>
  <sUserName>bill</sUserName>
  <bIncludeAdditionalInfo>>false</bIncludeAdditionalInfo>
  <bIncludeProjects>>true</bIncludeProjects>
</urn:GetUserInfo>
```

GetUsers

Get list of all users

Style

Document

Input (Literal)

The input of this method is the argument GetUsers having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	0..1	
sProjectName	string	0..1	
bIncludeAdditionalInfo	boolean	0..1	

Output (Literal)

The output of this method is the argument GetUsersResponse having the structure defined by the following table.

Name	Type	Occurs	Description
result	UserList	1..1	

Remarks

Example of usage:

```
<urn:GetUsers>
  <sessionId>1</sessionId>
  <!-- optional, if set then will be returned all users from this group
-->
  <sGroupName></sGroupName>
  <!-- optional, is set then will be returned all users assigned to
this project -->
  <sProjectName></sProjectName>
  <bIncludeAdditionalInfo>>false</bIncludeAdditionalInfo>
</urn:GetUsers>
```

GrantPermissionToGroup

Grant permission to group for some action

Style

Document

Input (Literal)

The input of this method is the argument GrantPermissionToGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	
sResource	string	1..1	
sAction	string	1..1	
sObject	string	0..1	

Output (Literal)

The output of this method is the argument GrantPermissionToGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:GrantPermissionToGroup>
  <sessionId>1</sessionId>
  <sGroupName>read</sGroupName>
  <sProjectName>RMDemo</sProjectName>
  <sResource>Categories</sResource>
  <sAction>FullAccess</sAction>
</urn:GrantPermissionToGroup>
```

GrantPermissionToUser

Grant permission to user for some action

Style

Document

Input (Literal)

The input of this method is the argument GrantPermissionToUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sProjectName	string	1..1	
sResource	string	1..1	
sAction	string	1..1	
sObject	string	0..1	

Output (Literal)

The output of this method is the argument GrantPermissionToUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:GrantPermissionToUser>
  <sessionId>1</sessionId>
  <sUserName>nata</sUserName>
  <sProjectName>RMDEMO</sProjectName>
  <sResource>Classes</sResource>
  <sAction>Create</sAction>
  <sObject>Marketing_Requirements</sObject>
</urn:GrantPermissionToUser>
```

InheritPermissionForUser

Inherit permission to user for some action

Style

Document

Input (Literal)

The input of this method is the argument InheritPermissionForUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sProjectName	string	1..1	
sResource	string	1..1	
sAction	string	1..1	
sObject	string	0..1	

Output (Literal)

The output of this method is the argument InheritPermissionForUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:InheritPermissionForUser>
  <sessionId>1</sessionId>
  <sUserName>ephoto</sUserName>
  <sProjectName>RMDEMO</sProjectName>
  <sResource>Attributes</sResource>
  <sAction>Update</sAction>
</urn:InheritPermissionForUser>
```

RevokePermissionFromGroup

Revoke permission from group for some action

Style

Document

Input (Literal)

The input of this method is the argument RevokePermissionFromGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	
sResource	string	1..1	
sAction	string	1..1	
sObject	string	0..1	

Output (Literal)

The output of this method is the argument RevokePermissionFromGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:RevokePermissionFromGroup>
  <sessionId>1</sessionId>
  <sGroupName>read</sGroupName>
  <sProjectName>RMDemo</sProjectName>
  <sResource>Categories</sResource>
  <sAction>FullAccess</sAction>
</urn:RevokePermissionFromGroup>
```

RevokePermissionFromUser

Revoke permission from user for some action

Style

Document

Input (Literal)

The input of this method is the argument RevokePermissionFromUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sProjectName	string	1..1	
sResource	string	1..1	
sAction	string	1..1	
sObject	string	0..1	

Output (Literal)

The output of this method is the argument RevokePermissionFromUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:RevokePermissionFromUser>
  <sessionId>1</sessionId>
  <sUserName>nata</sUserName>
  <sProjectName>RMDEMO</sProjectName>
  <sResource>Classes</sResource>
  <sAction>Create</sAction>
  <sObject>Marketing_Requirements</sObject>
</urn:RevokePermissionFromUser>
```

UnassignCategoryFromGroup

Un-assign category from the group

Style

Document

Input (Literal)

The input of this method is the argument UnassignCategoryFromGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	
sCategoryName	string	1..1	
bIncludeSubcategories	boolean	0..1	

Output (Literal)

The output of this method is the argument UnassignCategoryFromGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:UnassignCategoryFromGroup>
  <sessionId>1</sessionId>
  <sGroupName>read</sGroupName>
  <sProjectName>RMDemo</sProjectName>
  <sCategoryName>Functional</sCategoryName>
  <bIncludeSubcategories>true</bIncludeSubcategories>
</urn:UnassignCategoryFromGroup>
```


UnassignGroupFromProject

Un-assign group from the project

Style

Document

Input (Literal)

The input of this method is the argument `UnassignGroupFromProject` having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	

Output (Literal)

The output of this method is the argument `UnassignGroupFromProjectResult` having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:UnassignGroupFromProject>  
  <sessionId>1</sessionId>  
  <sGroupName>read</sGroupName>  
  <sProjectName>RMDemo</sProjectName>  
</urn:UnassignGroupFromProject>
```

UnassignUserFromGroup

Un-assign user from the group

Style

Document

Input (Literal)

The input of this method is the argument UnassignUserFromGroup having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sGroupName	string	1..1	

Output (Literal)

The output of this method is the argument UnassignUserFromGroupResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:UnassignUserFromGroup>  
  <sessionId>1</sessionId>  
  <sUserName>qa</sUserName>  
  <sGroupName>group</sGroupName>  
</urn:UnassignUserFromGroup>
```

UnassignUserFromProject

Un-assign user from the project

Style

Document

Input (Literal)

The input of this method is the argument UnassignUserFromProject having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	
sUserName	string	1..1	
sGroupName	string	1..1	
sProjectName	string	1..1	

Output (Literal)

The output of this method is the argument UnassignUserFromProjectResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:UnassignUserFromProject>  
  <sessionId>1</sessionId>  
  <sUserName>qa</sUserName>  
  <sGroupName>TEST</sGroupName>  
  <sProjectName>RMDEMO</sProjectName>  
</urn:UnassignUserFromProject>
```

UpdateListAttributeValues

Change list attribute values in attribute definition

Style

Document

Input (Literal)

The input of this method is the argument UpdateListAttributeValues having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Name of class where list attribute is defined
sAttributeName	string	1..1	List attribute name
aValues	ListItems	1..1	List of attribute values

Output (Literal)

The output of this method is the argument UpdateListAttributeValues having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	Result of updating list values

Remarks

The UpdateListAttributeValues call provides a method to change defined list attribute values to the new ones, or change properties of existing values.

After request list values will be ordered as specified in the input.

It is possible to set default attribute value(s), mark value as retired, remove and restore value.

In order to retire (delete) value, it is needed to put 'true' in <bDeleted> parameter.

In order to restore the retired value, it is needed to put 'false' in <bDeleted> parameter.

If some existing value is not specified in request, then it will be removed.

Example of usage:

```
<urn:UpdateListAttributeValues>
  <sessionId>1489425621</sessionId>
  <sClassName>Product_Requirements</sClassName>
  <sAttributeName>PRIORITY</sAttributeName>
  <aValues>
    <item>
      <sName>High</sName>
      <!--Optional:-->
      <bDefault>>false</bDefault>
    </item>
  </aValues>
</urn:UpdateListAttributeValues>
```

```
        <bDeleted>>false</bDeleted>
    </item>
    <item>
        <sName>Low</sName>
        <!--Optional:-->
        <bDefault>>true</bDefault>
        <bDeleted>>false</bDeleted>
    </item>
</aValues>
</urn:UpdateListAttributeValues>
```

UpdateUser

Update user information and account properties

Style

Document

Input (Literal)

The input of this method is the argument UpdateUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sUserName	string	1..1	Internal RM user or SSO user name
sPassword	string	0..1	New password
sLoginType	string	0..1	Possible values: RM, SSO SSO is also used for LDAP users.
sNewUserName	string	0..1	New user name
anUserInfo	UserInfo	0..1	

Output (Literal)

The output of this method is the argument UpdateUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Remarks

Example of usage:

```
<urn:UpdateUser>
  <sessionId>1</sessionId>
  <sUserName>admin</sUserName>
  <anUserInfo>
    <sPhone>123543324</sPhone>
    <sEmail>test@email.com</sEmail>
  </anUserInfo>
</urn:UpdateUser>
```

Changing the User Name:

```
<urn:UpdateUser>
  <sessionId></sessionId>
  <sUserName>DEV_MANAGER</sUserName>
  <sNewUserName>NEW_MANAGER</sNewUserName>
</urn:UpdateUser>
```

UpdateUserAttributeValues

Allows to setup default values for the user attributes.

Style

Document

Input (Literal)

The input of this method is the argument UpdateUser having the structure defined by the following table.

Name	Type	Occurs	Description
sessionId	string	1..1	The current session identifier
sClassName	string	1..1	Class name
sAttributeName	string	0..1	Attribute name
aValues	UserItems	0..1	Default values

Output (Literal)

The output of this method is the argument UpdateUserResult having the structure defined by the following table.

Name	Type	Occurs	Description
result	Result	1..1	

Example of usage:

```
<urn:UpdateUserAttributeValues>
  <sessionId>1568710553</sessionId>
  <sClassName>Marketing_Requirements</sClassName>
  <sAttributeName>USER_TEST</sAttributeName>
  <aValues>
    <item>
      <sName>joe</sName>
      <!-- Possible values for nItemType: 1 - groups, 2 - user, 3 - team-->
      <nItemType>2</nItemType>
      <bDefault>>true</bDefault>
    </item>
    <item>
      <sName>icDemo Users</sName>
      <!-- Possible values for nItemType: 1 - groups, 2 - user, 3 - team-->
      <nItemType>1</nItemType>
      <bDefault>>true</bDefault>
    </item>
  </aValues>
</urn:UpdateUserAttributeValues>
```

Complex Types

Name	Description
AdditionalUserInfo	
Group	
GroupInfo	
GroupList	
IdList	
LoginSource	
LoginSources	
NameValue	
NameValueList	
Result	
StringList	
User	
UserInfo	
UserInfoExt	
UserItemDef	
UserItems	
UserList	
UserNameList	

AdditionalUserInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sPhone	string	0..1	
sFax	string	0..1	
sPager	string	0..1	
sLocation	string	0..1	
sMobilePhone	string	0..1	
bChangePwdOnLogin	boolean	1..1	
bCanNotChangePwd	boolean	1..1	
bPwdNeverExpires	boolean	1..1	

Group

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sGroupName	string	0..1	
aProjects	StringList	1..1	

GroupInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sDescription	string	0..1	

GroupList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Group	0..*	

IdList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	int	0..*	

LoginSource

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	0..1	

LoginSources

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	LoginSource	0..*	

NameValue

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	
sValue	string	0..1	

NameValueList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	NameValue	0..*	

Result

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
nStatusCode	int	1..1	
sMessage	string	0..1	

StringList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	string	0..*	

User

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sUserName	string	0..1	
sUserLoginType	string	0..1	
sFirstName	string	0..1	
sLastName	string	0..1	
sEmail	string	0..1	
bAccountDisabled	boolean	1..1	
anAdditionalUserInfo	AdditionalUserInfo	0..1	

UserInfo

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sFirstName	string	0..1	
sLastName	string	0..1	
sPhone	string	0..1	
sFax	string	0..1	
sEmail	string	0..1	
sPager	string	0..1	
sLocation	string	0..1	
sMobilePhone	string	0..1	
bChangePwdOnLogin	boolean	0..1	
bCanNotChangePwd	boolean	0..1	
bPwdNeverExpires	boolean	0..1	
bAccountDisabled	boolean	0..1	

UserInfoExt

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sFirstName	string	0..1	
sLastName	string	0..1	
sPhone	string	0..1	
sFax	string	0..1	
sEmail	string	0..1	
sPager	string	0..1	
sLocation	string	0..1	
sMobilePhone	string	0..1	
bAccountDisabled	boolean	0..1	

UserItemDef

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sName	string	1..1	User/group/team name
sType	string	1..1	1 groups 2 user 3 team
bDefault	boolean	0..1	<ul style="list-style-type: none"> ▪ true: Value is default ▪ false: Value is not default

UserItems

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	UserItemDef	0..*	

UserList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	User	0..*	

UserNameList

Derived By

Restricting anyType

Content Model

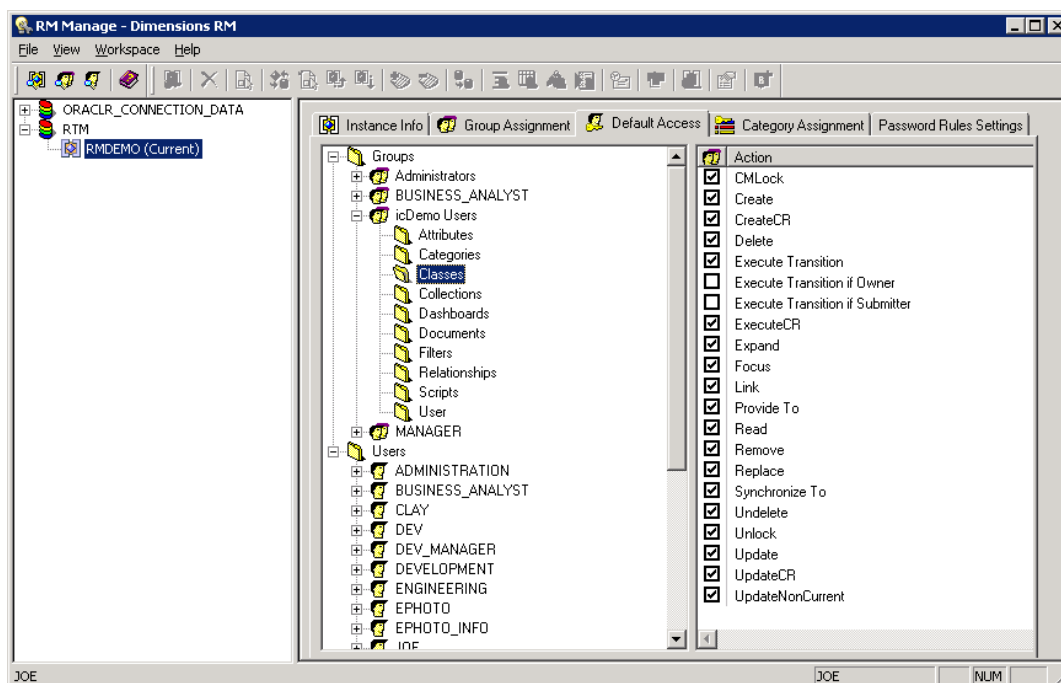
Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	string	0..*	

Appendix

Resource and Action Parameters

Several permission related functions require to specify a resource (usually sResource) and action (usually sAction) parameter. The values for action are only valid in combination with the related resource value. The easiest way to retrieve the resource and action combinations is by checking the **Default Access** options for a group or user in RM Manage.



Resource is identical with the folder name. In the image above, the resource **Class** is selected.

Action is an option name from the list of options.

For the resource **Classes**, possible action values are e.g. Delete, ExecuteCR, Read.

Chapter 3

RM Restful Services

General	382
Authentication for Rest Services	384
Methods	386

General

General information about RM Restful Services:

- 1 URI: `http://<host>:<port>/rtmBrowser/RestServices/<ServiceName>/<Action>`
- 2 Support only JSON/JSONP format.
- 3 RM Restful services are cookies independent. (don't set/retrieve cookies)
- 4 Errors are returned through the HTTP status in response headers:
 - a **401 Unauthorized:** The client should retry the request with a suitable Authorization header.
 - b **403 Forbidden:** The server understood the request, but is refusing to fulfill it. Authorization was failed.
 - c **404 Not found:** The server has not found anything matching the Request-URI.
 - d **500 Internal Server Error:** The server encountered an unexpected condition which prevented it from fulfilling the request: can't connect to oracle, data not found, etc.



NOTE Detail information about error presented in `X-Error-Message` response header field.

The screenshot shows a web browser's developer console with the 'Response Headers' tab selected. The headers are as follows:

Header	Value
Access-Control-Allow-Cred...	true
Access-Control-Allow-Orig...	*
Connection	close
Content-Length	1189
Content-Type	text/html; charset=utf-8
Date	Wed, 15 May 2013 15:10:39 GMT
Server	Unknown Web Server/1.0
X-Error-Message	Attribute 'PUID2' is not in the object 'MARKETING_RE

Registry Keys to Specify Default Database and Instance

The following Registry Keys, located on the RM Server, provide a mechanism for the organization to set a default for both database and instance. This eliminates the need to include both when relying the default settings.

Registry Key	Description
DEFAULT_DATABASE	The name of the Default Database, e.g., DRM
DEFAULT_INSTANCE	The name of the Default Instance, e.g., ALM_DEMO

RestServices/Image?authenticated=DBB6979CEFD5B27ED2B0E&db=RM&p=RMDEMO&u=ephoto&image= I9_7ani1a.gif



NOTE If the user has not specified any type of authentication in the HTTP request described above, the server will return response `401 Unauthorized`, in which case the login dialog will be raised.



Methods

Name	Description
Reports	Get reports
Report	Get report info
Run	Run report
Containers	Get containers
ContainerRequirements	Get requirements from the container

Reports

Get list of reports. Results could be filtered by report type, category and class id (class_id only for 'filter' reports)

Action

http://hostname:port/rtmBrowser/RestServices/Reports?

Style

HTTP GET

Input

The inputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
type	int	0..1	Report type: 0 - all 1 - filter 2 - script 3 - trace
category_id	int	0..1	Category id
depth	int	0..1	Depth search by category.
class_id	int	0..1	Class id (only for filter report).

Output (JSON)

The outputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
data	Complex Type	1..1	List of reports
message	string	1..1	
oiddata	object	1..1	
success	boolean	1..1	
total	int	1..1	

Complex Type: data

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Complex Type	0..*	

Complex Type: item**Content Model**

Contains elements as defined in the following table:

Component	Type	Occurs	Description
		1..1	
categoryId	int	1..1	Category id
description	string	1..1	Category name
id	int	1..1	Report id
name	string	1..1	Report name

Remarks**Example of usage:**

<http://host:8080/rtmBrowser/RestServices/Reports?db=RM&p=RMDEMO&u=ephoto&pwd=rtm&type=1>

Example of output:

```
{
  "data": [
    {
      "categoryId": 0,
      "classId": 1,
      "className": "Marketing_Requirements",
      "description": "This filter retrieves all current Marketing
Requirements assigned to a 'Build1' delivery phase.",
      "hasRuntimeParameters": false,
      "id": 183,
      "name": "Build 1 Marketing Rqmts",
      "type": "filter"
    },
    {
      "categoryId": 0,
      "classId": 1,
      "className": "Marketing_Requirements",
      "description": "This filter retrieves all current Marketing
Requirements.",
      "hasRuntimeParameters": false,
      "id": 181,
      "name": "Current Marketing Rqmts",
      "type": "filter"
    }
  ],
  "message": ""
}
```

```
"oiddata": null,  
"success": true,  
"total": 2  
}
```

Report

Get information about the specified report

Action

http://hostname:port/rtmBrowser/RestServices/Report?

Style

HTTP GET

Input

The inputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
id	int	0..1	Report id
name	string	0..1	Report name
category_id	int	0..1	Category id
category_name	string	0..1	Category name
rtval	int	0..1	If set to 1 – service will return additional information for the runtime parameters: possible values for the List Attribute, Categories, Containers.

Output (JSON)


The outputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
data	Complex Type	1..1	Report info
message	string	1..1	
oiddata	object	1..1	
success	boolean	1..1	
total	int	1..1	

Complex Type: data

Content Model

Contains elements as defined in the following table:


Component	Type	Occurs	Description
 SEQUENCE		0..1	
category_id	int	1..1	Category id
category_name	string	1..1	Category name
classes	Complex Type	1..1	Classes used in report
column_model	Complex Type	1..1	Attributes

Component	Type	Occurs	Description
description	string	1..1	Report description
id	int	1..1	Report Id
name	string	1..1	Report name
owner	string	1..1	Owner of the report
report_type	string	1..1	Report type
runtime_parameters	Complex Type	1..1	Runtime parameters
script	string	1..1	Script text

Complex Type: classes

Content Model


Contains elements as defined in the following table:

Component	Type	Occurs	Description
 SEQUENCE		1..*	
name	string	1..1	Class name
xref	string	0..1	Relationship name

Complex Type: column_model

Content Model


Contains elements as defined in the following table:

Component	Type	Occurs	Description
 SEQUENCE		0..*	
column_name	string	1..1	Attribute name
display_name	string	1..1	Attribute display name

Complex Type: runtime_parameters

Content Model

Contains elements as defined in the following table:

Component	Type	Occurs	Description
 SEQUENCE		0..*	
attr_name	string	1..1	Attribute name
attr_type	int	1..1	Attribute type
multiple	boolean	0..1	Is multi selection
name	string	1..1	Name

Component	Type	Occurs	Description
prompt	string	1..1	Prompt for value
values	StringList	0..1	Allowed values for the attribute

Remarks**Example of usage:**

[http://localhost:8080/rtmBrowser/RestServices/Report?name=All Current Component Requirements&db=RM&p=RMDEMO&u=ephoto&pwd=rtm](http://localhost:8080/rtmBrowser/RestServices/Report?name=All%20Current%20Component%20Requirements&db=RM&p=RMDEMO&u=ephoto&pwd=rtm)

Example of output:

```
{
  "data": {
    "category_id": 0,
    "category_name": "RMDEMO",
    "classes": [
      {
        "name": "Component_Requirements"
      }
    ],
    "column_model": [
      {
        "column_name": "PUID",
        "display_name": "Rqmt ID"
      },
      {
        "column_name": "DOCUMENT_ID",
        "display_name": "Document ID"
      },
      {
        "column_name": "PARAGRAPH_ID",
        "display_name": "Paragraph ID"
      },
      {
        "column_name": "TEXT",
        "display_name": "Text"
      }
    ],
    "description": "This script reports all the current Component Requirements",
    "id": 1143,
    "name": "All Current Component Requirements",
    "owner": "Ephoto",
    "report_type": "script",
    "runtime_parameters": [ ],
    "script": "select <Rqmt ID>PUID <Document ID>DOCUMENT_ID"
  }
}
```

```
<Paragraph ID>PARAGRAPH_ID <Text>TEXT from Component_Requirements where
STATUS = 'Current' order by @PUID"
  },
  "message": "",
  "oiddata": null,
  "success": true,
  "total": 0
}
```

Run

Run report by given report id or report name.

Action

http://hostname:port/rtmBrowser/RestServices/Report/Run?

Style

HTTP GET

Input

The inputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
id	int	0..1	Report id
name	string	0..1	Report name
category_id	int	0..1	Category id
category_name	string	0..1	Category name
page	int	0..1	Current page
rows	int	0..1	Count of rows on the page
sidx	string	0..1	Column for sorting
sord	string	0..1	Type of sorting: desc or asc
skipheaders	string	0..1	Skip headers for XREF query

Output (JSON)

The outputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
page	int	1..1	current page
records	int		records in page
rows	Complex Type		
total			total pages

Complex Type: rows

Content Model

Contains elements as defined in the following table:

Component	Type	Occurs	Description
SEQUENCE		1..1	
item	Complex Type	0..*	

Complex Type: item**Content Model**

Contains elements as defined in the following table:

Component	Type	Occurs	Description
SEQUENCE		1..1	
PUID		1..1	
TITLE	string	1..1	
id		1..1	row id
c		1..1	class id
o		1..1	object id

Remarks**Example of usage:**

```
http://localhost:8080/rtmBrowser/RestServices/Report/Run?name=SomeReport&db=RM&p=RMDEMO&u=epphoto&pwd=rtm
```

Example of output:

```
{
  "page" : 1,    -- current page
  "records" : 2, --records in page
  "rows" : [
    {
      "PUID" : "MRKT_000002",
      "TITLE" : "Support Advantix formats",
      "id" : 1,  --row id
      "c" : 1,  --class id
      "o" : 31  --object id
    },
    {
      "PUID" : "MRKT_000003",
      "TITLE" : "Runs on standard home PC",
      "id" : 2,  --row id
      "c" : 1,  --class id
      "o" : 32  --object id
    }
  ],
  "total" : 5  --total pages
}
```

}



NOTE Columns "c" and "o" returns always for all rows. They can be used for create URL to open requirement:

```
http://.../rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&c=1&o=32
```

Reports with runtime parameters

To run report with runtime parameters all values must be specified in the request: `http://.../RestServices/Report/Run?name=test&db=RM&p=RMDEMO&RTP__STATUS_2=Current&RTP__TITLE_1=demo`

Names of runtime parameters (RTP__STATUS_1, RTP__TITLE_2) can be obtained from report info REST Service: `http://.../RestServices/Report?name=test&db=RM&p=RMDEMO`

```
"runtime_parameters" : [
  {
    "attr_name" : "TITLE",
    "name" : "RTP__TITLE_1",
    "prompt" : "Enter Title"
  },
  {
    "attr_name" : "STATUS",
    "name" : "RTP__STATUS_2",
    "prompt" : "Enter Current Status"
  }
]
```



NOTE If one of the runtime parameters will not specified in request server will return an error.

The screenshot shows a browser's developer console with the following details:

- Console:** GET http://ua6237-9090/rtmBrowser/RestServices/Report...se&nd=1368630628656&rows=4&page=1&sidx=&sord=asc 500 Internal Server Error 10.29s
- Response Headers:**
 - Access-Control-Allow-Cred...: true
 - Access-Control-Allow-Orig...: *
 - Connection: close
 - Content-Length: 1189
 - Content-Type: text/html; charset=utf-8
 - Date: Wed, 15 May 2013 16:10:39 GMT
 - Server: Unknown Web Server/1.0
- X-Error-Message:** Attribute 'PUIID2' is not in the object 'MARKETING_REQUIREMENTS'

Additional info:

Using in SBM rest grids. Only class reports (by only 1 class) or multi-class scripts but with same column structure are supported in SBM rest grids. Multi-class script reports with different column structure, relationship reports, traceability reports are not supported, due to unavailability to present different column structure for different classes in SBM rest grid.

Containers

Get list of containers by specified type. Results could be filtered by container name.

Action

http://hostname:port/rtmBrowser/RestServices/Containers?

Style

HTTP GET

Input

The inputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
type	int	1..1	Container type Possible values: ALL - containers B - baselines C - collections D - documents S - snapshots
documentId	int	0..1	Document id if "type" parameter is set to S
designMode	boolean	0..1	possible value – true If that parameter is set (designMode=true) the service returns an example data
filter	string	0..1	Search by container name

Output (JSON)

The outputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
data	Complex Type	1..1	Containers info
message	string	1..1	
oiddata	object	1..1	
success	boolean	1..1	
total	int	1..1	

Complex Type: data

Content Model

Contains elements as defined in the following table:

Component	Type	Occurs	Description
SEQUENCE		0..*	
id	int	1..1	Container id

Component	Type	Occurs	Description
name	string	1..1	Container name
type	string	1..1	Container type B - baseline C - collection D - document S - snapshot
url	string	1..1	Direct url to the container

Remarks

"filter" is an optional parameter. A search string can be provided in that parameter. The search is case insensitive and searches by container name.

Example: find all containers which have AA or BB string in the name - filter=AA%20BB

Example of usage:

```
http://localhost:8080/rtmBrowser/RestServices/
Containers?sessionId=123456&type=all
```

Example of output:

```
{
  "data": [
    {
      "id": 11,
      "name": "Baseline of Engineering Hot List",
      "type": "B",
      "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=collection&db=RM&proj=RMDEMO&collectionId=11"
    },
    {
      "id": 20,
      "name": "Def document",
      "type": "D",
      "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=doc&db=RM&proj=RMDEMO&docID=20"
    },
    {
      "id": 19,
      "name": "ECP controlled",
      "type": "D",
      "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=doc&db=RM&proj=RMDEMO&docID=19"
    },
    {
      "id": 5,
      "name": "Engineering Hot List",
```

```
        "type": "C",
        "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=collection&db=RM&proj=RMDEMO&collectionId=5"
    }
],
"message": "",
"oiddata": null,
"success": true,
"total": 4
}
```

ContainerRequirements

Get requirements in the given container. Requirements could be filtered by specified criteria.

Action

`http://hostname:port/rtmBrowser/RestServices/ContainerRequirements?`

Style

HTTP GET

Input

The inputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
id	int	1..1	Container id
designMode	boolean	0..1	possible value - true If that parameter is set (designMode=true) the service returns example data
filter	string	0..1	Filter for the requirements list
searchFullText	boolean	0..1	

Output (JSON)

The outputs of this method are the arguments defined by the following table.

Name	Type	Occurs	Description
data	Complex Type	1..1	Requirements
message	string	1..1	
oiddata	object	1..1	
success	boolean	1..1	
total	int	1..1	

Complex Type: data

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		0..*	
category	string	1..1	Category
className	string	1..1	Class name
class_id	int	1..1	Class id
object_id	int	1..1	Object id
puid	string	1..1	PUID

Component	Type	Occurs	Description
status	string	1..1	Status
title	string	1..1	Title
url	string	1..1	url
version	int	1..1	Requirement version

Remarks

"filter" is optional parameter. A search string can be provided in that parameter. The search is case insensitive and searches for requirements - by PUID or TITLE or DESCRIPTION.

Optional "searchFullText" input parameter is available in ContainerRequirements REST service.

If it's set to true filter applies to Alphanumeric and Text attributes, as in the QuickSearch.

Example: find all requirements which have CC or DD string in the PUID or TITLE or DESCRIPTION - filter=CC%20DD

Example of usage:

```
http://localhost:8080/rtmBrowser/RestServices/
ContainerRequirements?sessionId=123456& containerId=12
```

Example of output:

```
{
  "data": [
    {
      "category": "RMDEMO",
      "className": "Marketing_Requirements",
      "class_id": 1,
      "object_id": 4,
      "puid": "MRKT_000004",
      "status": "Current",
      "title": "Annotate photos with text",
      "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&class=Marketing_Requirements
&puid=MRKT_000004&ver=1",
      "version": 1
    },
    {
      "category": "RMDEMO/Support/Training/Documentation",
      "className": "Marketing_Requirements",
      "class_id": 1,
      "object_id": 7,
      "puid": "MRKT_000007",
      "status": "Current",
      "title": "On-line help provided",
```

```
    "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&class=Marketing_Requirements
&puid=MRKT_000007&ver=1",
    "version": 1
  },
  {
    "category": "RMDEMO",
    "className": "Marketing_Requirements",
    "class_id": 1,
    "object_id": 26,
    "puid": "MRKT_000024",
    "status": "Current",
    "title": "Stored photo slideshows",
    "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&class=Marketing_Requirements
&puid=MRKT_000024&ver=1",
    "version": 1
  },
  {
    "category": "RMDEMO/Functional/Design and Construction",
    "className": "Marketing_Requirements",
    "class_id": 1,
    "object_id": 35,
    "puid": "MRKT_000031",
    "status": "Current",
    "title": "System comes in different colors",
    "url": "http://localhost:8080/rtmBrowser/cgi-bin/
rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&class=Marketing_Requirements
&puid=MRKT_000031&ver=1",
    "version": 1
  }
],
"message": "",
"oiddata": null,
"success": true,
"total": 4
}
```

Index

- B**
- baseline
 - functions
 - CreateBaseline 32
 - DeleteBaseline 51
 - types
 - RequirementIdentifierList 32
- C**
- change request
 - functions
 - ApproveChangeRequest 25
 - ChangeRequest 30
 - ProposeNewRequirement 127
 - RejectChangeRequest 150
 - types
 - CommandStatus 25, 150
 - ECPsObjectInfo 30, 127
 - NameValueList 127
 - ProposeNewRequirementResult 127
 - RequirementOut 30
 - cm
 - functions
 - CollectionAssociationWithCMPProject 31
 - SendCMNotifications 162
 - types
 - CMProjectInf 31
 - collection
 - functions
 - AddObjectToCollection 19
 - CollectionAssociationWithCMPProject 31
 - CreateCollection 36
 - FindObjectsInCollections 67
 - GetCollectionById 83
 - GetCollectionByName 84
 - GetCollectionInfo 85
 - GetObjectsInCollection 87
 - GetObjectsInCollections 88
 - ListCollections 105
 - ListCollectionsEx 106
 - ListCollectionsEx2 107
 - RemoveObjectFromCollection 152
 - types
 - addObjectToCollectionResponse 19
 - CMProjectInf 31
 - CollectionEx 83, 84
 - CollectionEx2List 107
 - CollectionExList 106
 - CollectionInfo 85
 - CollectionList 105
 - CreateCollectionEx 36
 - GetCollectionInfoRequest 85
 - IdList 67, 88
 - removeObjectFromCollectionResponse 152
 - RequirementFindList 67
 - RequirementIdentifier 152
 - RequirementIdentifier 19
 - RequirementList 87, 88
 - SearchRequirement 67
 - StringList 67, 87, 88
 - contacting technical support 11
- D**
- document
 - functions
 - AppendRMDocumentEx 21
 - AppendRtmDocument 23
 - AsyncAppendRtmDoc 26
 - AsyncCreateRtmDoc 27
 - AsyncQueryRtmDoc 28
 - AsyncReplaceRtmDoc 29
 - CreateChildDocument 35
 - CreateRMDocumentEx 42
 - CreateRtmDocument 46
 - DeleteRtmDocument 53
 - DocumentAddChapter 56
 - DocumentMoveObject 57
 - DocumentRemoveObject 58
 - FindRtmDocument 69
 - getAsyncAppendRtmDocStatus 71
 - getAsyncCreateRtmDocStatus 72
 - getAsyncQueryRtmDocStatus 73
 - getAsyncReplaceRtmDocStatus 74
 - ListRtmDocuments 110
 - QueryRtmDocumentStructure 141
 - RTMDocumentAddChapter 157
 - RTMDocumentAddRequirement 158
 - RTMDocumentExchangeRequirement 159
 - RTMDocumentRemoveRequirement 160
 - types
 - AppendRMDocumentExResult 21
 - AppendRtmDocumentResult 23

- AsyncCreateRtmDocResponse 27
- AsyncReplaceRtmDocResponse 29
- CommandStatus 53
- CreateChildDocumentData 35
- CreateRMDocumentExResult 42
- CreateRtmDocumentResult 35, 46
- DocumentObject 56
- DocumentsFilter 110
- ECPsObjectInfo 159
- getAsyncAppendRtmDocStatusResponse 71
- getAsyncCreateRtmDocStatusResponse 72
- getAsyncQueryRtmDocResponse 28
- getAsyncQueryRtmDocumentStatusResponse 73
- getAsyncReplaceRtmDocStatusResponse 74
- ns-chapterDetails 56, 157
- Requirement 158
- RtmDocument 21, 27
- RtmDocumentList 69, 110
- RtmDocumentProperties 141
- RtmDocumentWithCategory 42, 46

S

- snapshot
 - functions
 - CreateRtmDocSnapshot 45
 - GetRMDocSnapshotDiffUrl 91
 - GetRMDocSnapshots 92
 - GetRtmDocSnapshotById 93
 - GetRtmDocSnapshotByName 94
 - types
 - RtmDocSnaphotOutList 92
 - RtmDocSnapshot 45
 - RtmDocSnapshotOut 45, 93, 94

T

- technical support
 - contacting 11