



Dimensions RM 12.6.1 Web Service and Rest Service Reference

Table of Contents

rtmService	3
rtmService Web Service.....	4
Methods: rtmService.....	5
Complex Types: rtmService.....	139
Simple Types: rtmService.....	271
rtmAdminService Web Service.....	283
Methods: rtmAdminService.....	284
Complex Types: rtmAdminService	314
RM Restful Services	331
RM Restful Services.....	332
Authentication for Rest Services.....	333
Methods: RestServices.....	334
General.....	348

rtmService

Sources

Name	Description
rtmService	gSOAP 2.7.12 generated service definition
rtmAdminService	gSOAP 2.7.12 generated service definition

rtmService Web Service

Description

gSOAP 2.7.12 generated service definition

See Also

- [Methods](#)
- [Complex Types](#)
- [Simple Types](#)

Methods: rtmService

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
CollectionAssociationWithCMProject	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 Master 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
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.

Name	Description
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
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
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.
ListTemplates	Get a list of document templates in the project.

Name	Description
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
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
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

Name	Description
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

Method: AddObjectToCollection

Description

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>
```

Method: AddSyncXrefLink

Description

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>
```

Method: AppendRMDocumentEx

Description

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="RG9jdW1lbnRhdGlvbiBSZXF1aXJlbWVudHM=" uid="185" iid="45">
          <Description/>
          <Requirement classid="1" className="Marketing_Requirements" title="" category="14" uid="60" iid="25">
            <Attribute attrid="3011" typeid="1" title="Title">T24tbGluZSBoZWxwIHByb3ZpZGVk</Attribute>
            <Attribute attrid="31" typeid="5"
            title="Text">TUIjNS1WZjZjZW50c2VudC1UeXBIOiBtdWx0aXBhcncVcmVsYXRIZDsKICBib3VudGFy
            eT0iLS0tLS0tLS0tLS0tLS0tQm91bmRhcncjMDA9X1BGR1dlUXpWUHTdmVVRzdU1F1ljsgdHwZT0idGV4dC9odG1slgo
            KLS0tLS0tLS0tLS0tLS1Cb3VudGFyS0wMD1fUEZHV2VRelZQc1N2ZVVHN2dTUXUKQ29udGVudC1UeXBIOiB0Z2h
            0L2h0bWwKQ29udGVudC1UcmFuc2Zlci1FbmNvZGlueGluZzogYmFzZTY0CgpWR2hsSUdWUWFHOTBieUJ6ZVhOMFpX
            MGdjMmhoYkd3Z2JXRnJaU0lwYUdVZ2RYTmxJRzltSUc5dUxXeHBibVVnYUdWc2NDQnoKYNlCMGFHRjBJRzV2SUd
            SdlkzVnRaVzUwVWVhScGlyNGdiM1JvWihJ22RHaGhiaUJoSUNKSFpYUjBhVzVUSUZOMFIYSjBaV1FpCkIHZDFvV1Js
            SUc1bFpXUnpJSFJ2SUdKbEltTjFjSEJzYVdWdWw0a0x0PT0Kc0tLS0tLS0tLS0tLS0tQm91bmRhcncjMDA9X1BGR1dlUXpW
            UHTdmVVRzdU1F1LS0K</Attribute>
            <Attribute attrid="26" typeid="12" title="Rqmt ID">TVJLVF8wMDAwMDc</Attribute>
          </Requirement>
        </Chapter>
      </Document>]]></sContent>
    </rtmDoc>
  </urn:AppendRMDocumentEx>

```

Method: AppendRtmDocument

Description

Append an RM document to an existing document.

Style

Document

Input (Literal)

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

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

Remarks

The AppendRtmDocument call provides a method for append an existing document with the specified document content.

Example of usage:

```
<urn:AppendRtmDocument>
  <sessionId>11111111</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="ZVBob3RvIFJlcXVpcmVtZW50cw==" uid="374">
          <Description/>
          <Chapter title="VXNpbmVtZW50cw==" uid="374">
            <Description/>
            <Requirement classid="1" className="Marketing_Requirements" title="" categoryid="6" uid="31">
              <Attribute attrid="3011" typeid="1" title="Title">U3lzdGVtIHJlc3BvbnNIIHRpbWVzMG==</Attribute>
              <Attribute attrid="31" typeid="5"
              title="Text">QmFzZTY0IGlzIGVtZW50cw==</Attribute>
              <Attribute attrid="26" typeid="12" title="Rqmt ID">TVJLVF8wMDAwMjk=</Attribute>
            </Requirement>
          </Chapter>
        </Chapter>
      </Document>]]></sContent>
    </rtmDoc>
  </urn:AppendRtmDocument>
```

Method: ApproveChangeRequest

Description

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>
```

Method: AsyncAppendRtmDoc

Description

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	

Method: AsyncCreateRtmDoc

Description

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	

Method: AsyncQueryRtmDoc

Description

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	

Method: AsyncReplaceRtmDoc

Description

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	

Method: ChangeRequest

Description

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	

Method: CollectionAssociationWithCMProject

Description

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

Style

Document

Input (Literal)

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

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

Output (Literal)

The output of this method is the argument CollectionAssociationWithCMProjectResult 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:CollectionAssociationWithCMProject>
  <sessionId>1111111111</sessionId>
  <sCollectionName>test_col</sCollectionName>
  <projectCM>
    <sProductName>QLARIUS</sProductName>
    <sProjectName>JAVA_BRANCHA_PRJ</sProjectName>
  </projectCM>
  <bAssociate>true</bAssociate>
</urn:CollectionAssociationWithCMProject>
```

Method: CreateBaseline

Description

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

Method: CreateCategory

Description

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	

Method: CreateChildDocument

Description

Create a new Child document of the specified Master 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 Master document.

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

Example of usage:

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

Method: CreateCollection

Description

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>testCol2</sName>
  </collectionName>
  <sCategoryFullName>RMDemo\Availability</sCategoryFullName>
</urn:CreateCollection>
```

Method: CreateObject

Description

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>
    </aAttrList>
  </anObject>
</urn:CreateObject>
```


Method: CreateObjectEx

Description

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>
```

Method: CreateObjectWithAttachment

Description

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>
```

Method: CreateProduct

Description

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>
```


Method: CreateRtmDocSnapshot

Description

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

Example of usage:

```
<urn:CreateRtmDocSnapshot>
  <sessionId>1478719333</sessionId>
  <snapshot>
    <sName>SNAPSHOT</sName>
    <nDocumentId>12</nDocumentId>
  </snapshot>
</urn:CreateRtmDocSnapshot>
```

Method: CreateRtmDocument

Description

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="RGVzaWdulFJlcXVpcmVtZW50cw==" 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">U3VvcG9ydCBBZHhbnRpeCBmb3JtYXRz</Attribute>
<Attribute attrid="31" typeid="5"
title="Text">TUINRS1WZXJzaW9uOiAxLjAKQ29udGVudC1UeXBIOiBtdWx0aXBhcnQvcmVsYXRIZDsKI
CBib3VuZGFyeT0iLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS1Cb3VuZGFyeS0wMD1fUjg2RmVRelZQc1N2ZVVHN2d
TUXUKQ29udGVudC1UeXBIOiB0ZXh0L2h0bWwKQ29udGVudC1UcmFuc2Zlci1FbmNvZGluZzogaFmFzZ
TY0CgpWR2hsSUdWUWFHOTBieUJ6ZVhOMFpXMGdjMmhoYkd3Z2MzVndjRzI5ZENCd2FHOTBiM01n
ZEdoaGRDQmhjbVVnUVdSMlIXNTAKYVhnZ0tETWdjMmhwZWIhNcEiHWnZjbTFoZENCdmNtbG5hVzVoY
khNZ2QybDBhQ0J1YjNRZ2JHVnpjeUlwYUdGdUIESTFOaUJqCmlyeHZjbnk1SUVOdmJuTnBaR1Z5WVh
ScGlyNGdjMmhwZ2Fmd4a0IHSmxJR2RwZG1WdUIlUnZJSFJvWlNlcm1iY2dkR2hoZENCcwpzWEpuWlhJZ
2NiSnBibJ6SUcxaGVTQmlaU0JoYzNOcGJXbHNZWFJsWkNCAgJtUWdZU0J6YjJ4MWRHbHZiaUJ6YU
dGc2JDQmkKWlNCamFHOXpaVzRnYVc0Z2NHrnlkQ0JpWVhObFpDQjFjRzI1SUhOMVkyZ2dabXhsZU
dsaWFXeHBkSGt1CgotLS0tLS0tLS0tLS0tLUJvdW5kYXJ5J5LTAwPV9SODZGZVF6ViBzU3ZiVUc3ZlNRd
S0tCg==</Attribute>
<Attribute attrid="26" typeid="12" title="Rqmt ID">TVJLVF8wMDAwMDI=</Attribute>
</Requirement>
</Chapter>
</Document>]]></sContent>
  </rtmDoc>
</urn:CreateRtmDocument>
```


Method: CreateTemplate

Description

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..1	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

Exapmple of usage:

```

<urn:CreateTemplate>
  <sessionId>1478777334</sessionId>
  <docTemplate>
    <sName>TEM511</sName>
  </docTemplate>
  <encodedContent>PD94bWwgdMvYc2lrbj0iMS4wlj8+DQoNCi08SW1wb3J0VGVtcGxhdGUgZGVmYX
  VsdENsYXNzPSltMSltdmVyc2lrbj0iMSlgaWQ9ljlEiPgoKDQoNCi08R2VuZXJhbFJ1bGUgaWQ9Imdlb
  mVyYWwiPgoKDQo8QXV0aG9yPnJtZGVtb2FkbWluPC9BdXR0b3I+DQoNCjxEYXRIPjEwLzE4LzlwM
  DcgNjoxMjo0MSBQTtwvRGF0ZT4NCg0KPERlc2NyaXB0aW9uPmVudCBpbmRvIERpbWVuc2lbnMgUk0gZGI2aWRIZCBpbiBja
  b3UgaW1wb3J0IGEGv29yZCBkb2N1bWVudCBpbmRvIERpbWVuc2lbnMgUk0gZGI2aWRIZCBpbiBja
  GFwdGVycy4gUmVmZXl0ZG8gdGhlIFNlcmVuYSBEaW1lbnNpb25zIFJNIEFkbWluaXN0cmF0b3IncycB
  HdWlkZSBmb3lgbW9yZSBpbmZvcmlhdGlvbiBhYm91dCBjcmVhdGluZyB2YXJpb3VzIHR5cGVzIG9m
  IHRlbXBsYXRlcy48L0Rlc2NyaXB0aW9uPgoKDQo8L0dlbmVyYWxSdWxIPgoKDQoNCi08Q2hhcHRlcl
  J1bGUgaWQ9ImNoYXB0ZXliPgoKDQoNCi08SGVhZGluz1J1bGUgaWQ9ImhIYWRpbmcilGhpbmQ9I
  mFsbCI+DQoNCg0KLTxlZWFkaW5ncz4NCg0KPEhIYWRpbmc+U3RhbmRhcmQgV29yZCBIZWFkaW
  5nczwwSGVhZGluz24NCg0KPC9IZWFkaW5ncz4NCg0KPC9IZWFkaW5nUnVsZT4NCg0KPC9DaGF
  wdGVyUnVsZT4NCg0KDQotPFJlcXVpcmVtZW50UnVsZSBpZD0icmVxdWlyZW1lbnQiPgoKDQoNCi0
  8TGF5b3V0UnVsZSBpZD0idGFibGVMYXlvdXQilGlnbm9yZT0iZmFsc2UiPgoKDQoNCi08SGVhZGluz
  1J1bGUgaWQ9ImhIYWRpbmcilGhpbmQ9ImFsbCI+DQoNCg0KLTxlZWFkaW5ncz4NCg0KPEhIYWRp
  bmc+SGVhZGluz25yPC9IZWFkaW5nPg0KDQo8SGVhZGluz25IZWFkaW5nIDM8L0hIYWRpbmc+DQ
  oNCjxlZWFkaW5nPkhlYWRpbmcgNDwvSGVhZGluz24NCg0KPEhIYWRpbmc+SGVhZGluz25A1PC9IZ
  WFkaW5nPg0KDQo8SGVhZGluz25IZWFkaW5nIDY8L0hIYWRpbmc+DQoNCjxlZWFkaW5nPkhlYWR
  pbmcgNzwwSGVhZGluz24NCg0KPEhIYWRpbmc+SGVhZGluz25A4PC9IZWFkaW5nPg0KDQo8SGVh
  ZGluz25IZWFkaW5nIDk8L0hIYWRpbmc+DQoNCjwvSGVhZGluz3M+DQoNCjwvSGVhZGluz1J1bGU
  +DQoNCjwvTGF5b3V0UnVsZT4NCg0KPC9SZXF1aXJlWVudFj1bGU+DQoNCjwvSW1wb3J0VGVtc
  GxhdGU+</encodedContent>
</urn:CreateTemplate>

```

Method: DeleteBaseline

Description

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>
```

Method: DeleteObjects

Description

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>
```

Method: DeleteRtmDocument

Description

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>
```

Method: DeleteSyncXrefLink

Description

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	

Method: DeleteTemplate

Description

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>
```

Method: DocumentAddChapter

Description

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>
```

Method: DocumentMoveObject

Description

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>1111111111</sessionId>
  <documentId>17</documentId>
  <IID>86</IID>
  <parentIID>82</parentIID>
  <position>1</position>
</urn:DocumentMoveObject>
```


Method: DocumentRemoveObject

Description

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>
```

Method: FindAndReplaceObject

Description

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	

Method: FindLinkedObjects

Description

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

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>
```

Method: FindModifiedObjects

Description

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 criateria

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>
```

Method: FindObjects

Description

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:FindObjects>
  <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>
</urn:FindObjects>
```

```
</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);

Method: FindObjectsInClasses

Description

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);

Method: FindObjectsInCollections

Description

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:FindObjectInCollections>
```

Method: FindRtmDocument

Description

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>
```

Method: FindTemplate

Description

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>
```

Method: getAsyncAppendRtmDocStatus

Description

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	

Method: getAsyncCreateRtmDocStatus

Description

Get status from the async. test command

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	getAsyncCreateRtmDocStatusResponse	1..1	

Method: getAsyncQueryRtmDocStatus

Description

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	

Method: getAsyncReplaceRtmDocStatus

Description

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	

Method: getAsyncStatus

Description

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	

Method: GetAttributeDefInfo

Description

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>
```

Method: GetAvailableTransitions

Description

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>
    <sPUIID>COMP_000008</sPUIID>
  </anObject>
</urn:GetAvailableTransitions>
```

Method: GetCategoryByDisplayName

Description

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>
```

Method: GetCategoryById

Description

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'

Method: GetClassDefInfo

Description

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>
```

Method: GetCollectionByld

Description

Get a collection specified by collection id.

Style

Document

Input (Literal)

The input of this method is the argument GetCollectionByld 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 GetCollectionByldResult 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:GetCollectionByld>  
  <sessionId>1479221671</sessionId>  
  <collection-id>1</collection-id>  
</urn:GetCollectionByld>
```

Method: GetCollectionByName

Description

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>
```


Method: GetCollectionInfo

Description

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	

Method: GetObjectsInCollection

Description

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	

Method: GetObjectsInCollections

Description

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	

Method: GetObjectsInRMDocument

Description

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	

Method: GetProjectInfo

Description

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	

Method: GetRMDocSnapshotDiffUrl

Description

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	

Method: GetRMDocSnapshots

Description

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	

Method: GetRtmDocSnapshotById

Description

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	

Method: GetRtmDocSnapshotByName

Description

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	

Method: GetSharedFile

Description

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	

Method: GetTemplateFile

Description

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	

Method: `getTestAsyncStatus`

Description

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	

Method: GetUserAttributeDefInfo

Description

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>
```

Method: IsSSOLoginEnabled

Description

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	

Method: LinkObjects

Description

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/ LinkObjectsExByld web services. Any links between requirements that were deleted now can be restored. Usage is the same as on creating link.

Method: LinkObjectsEx

Description

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.

Method: LinkObjectsExByld

Description

Link multiple children to a parent object

Style

Document

Input (Literal)

The input of this method is the argument LinkObjectsExByld 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:LinkObjectsExByld>
  <sessionId>111111111</sessionId>
  <nRelationId>5</nRelationId>
  <nParentId>32</nParentId>
  <aChildIds>
    <item>47</item>
  </aChildIds>
</urn:LinkObjectsExByld>
```

Additional Info:

It is also possible to undelete deleted requirement links using LinkObjects/ LinkObjectsEx/ LinkObjectsExByld web services. Any links between requirements that were deleted now can be restored. Usage is the same as on creating link.

Method: ListAllAllowedCategories

Description

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	

Method: ListCategories

Description

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	

Method: ListCollections

Description

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	

Method: ListCollectionsEx

Description

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	

Method: ListCollectionsEx2

Description

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	

Method: ListObjectLinks

Description

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:

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

Method: ListRootCategories

Description

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	

Method: ListRtmDocuments

Description

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
- bIncludeSubCategories,

Category can be specified by full path (separated with '\' or '/') or by category_id.

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 Master 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>
```

Method: ListTemplates

Description

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	

Method: Login

Description

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

Possible to login using Internal RM user or SSO user credentials.

Example of usage:

```
<urn:Login>
  <sUserId>ephoto</sUserId> <!--Internal RM user name or SSO user name-->
  <sPassword>rtm</sPassword><!--Internal RM user password or SSO user password-->
  <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:2EE9D7815C921FCB3E8F1FB17F8C9D83F5EC103B4B055E2AA62900F74F95E25
  0B4C63A18A362554EE703EA2BD90F4C4771F8BAC71D6B8520CCCDF2FE64D68A1269CC8741A
  F586AE10EFD7A5A905D5F71CAC63BAFBBDD4FD33408356A970669ADD5D4F37CEEEEA89865
  930ECF8ADED5C26B2C6E62A6561592</sessionId>
</urn:ListRtmDocuments>
```

Method: LoginByProjectId

Description

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

Possible to login using Internal RM user or SSO user credentials.

Example of usage:

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

Method: LoginEx

Description

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

Possible to login using Internal RM user or SSO user credentials.

Example of usage:

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

Method: Logout

Description

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	

Method: ModifyObjectsByScript

Description

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>

```


Method: ProposeNewRequirement

Description

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>vvvvv</inChangeReason>
</urn:ProposeNewRequirement>
```

Method: PutSharedFile

Description

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	

Method: QueryAttributes

Description

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

Method: QueryAttributesEx

Description

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>
```

Method: QueryAttributesEx2

Description

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:QueryAttributesEx>
```

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>
```

Method: QueryClasses

Description

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	

Method: QueryClassesEx

Description

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	

Method: QueryDatabases

Description

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	

Method: QueryLinkClasses

Description

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	

Method: QueryProjects

Description

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	

Method: QueryProjectsEx

Description

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	


```

xJR2RwZG1WdUIIUnZJSFJvWINCM1IYa2dkR2hoZENCcwpZWEpuWlhJZ2NISnBibIJ6SUcxaG
VTQmlaU0JoYzNOcGJXbHNZWFJsWkNCaGJtUWdZU0J6YjJ4MWRHbHZiaUJ6YUdGc2JDQm
kKWINCamFHOXpaVzRnYVc0Z2NHRnlkQ0JpWVhObFpDQjFjRzl1SUhOMVkyZ2dabXhsZUdsa
WFXeHBkSGt1CgotLS0tLS0tLS0tLS0tLUJvdW5kYXJ5LTAwPV9SODZGZVF6VIBzU3ZIVUc3Z1
NRdS0tCg==</Attribute>
<Attribute attrid="26" typeid="12" title="Rqmt ID">TVJLVF8wMDAwMDI=</Attribute>
</Requirement>
</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

```

Method: QuerySystemInfo

Description

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	

Method: ReadObject

Description

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	

Method: ReadObjectAttachment

Description

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	
aRequirement	RequirementIdentifier	1..1	
sAttrName	string	1..1	

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	

Method: ReadObjectEx

Description

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	

Method: ReadObjects

Description

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	

Method: ReadSyncXrefLink

Description

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	

Method: RejectChangeRequest

Description

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	

Method: RemoveObject

Description

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	

Method: RemoveObjectFromCollection

Description

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	

Method: ReplaceObject

Description

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	

Method: ReplaceObjectEx

Description

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	

Method: ResolveSuspicion

Description

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

Method: ReplaceObjectWithAttachment

Description

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	RequirementOut	1..1	

Method: RTMDocumentAddChapter

Description

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	

Method: RTMDocumentAddRequirement

Description

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	

Method: RTMDocumentExchangeRequirement

Description

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	

Method: RTMDocumentRemoveRequirement

Description

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	

Method: runTestAsync

Description

Run the async. test command

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	

Method: SendCMNotifications

Description

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	

Method: TransitionObject

Description

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
sPUID	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>111111111</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>
```

Method: UnlinkObjects

Description

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	
sParentPUID	string	1..1	
sChildClassName	string	1..1	
aChildPUIDs	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>
  <sParentPUID>MRKT_000020</sParentPUID>
  <sChildClassName>Product_Requirements</sChildClassName>
  <aChildPUIDs>
    <item>PROD_000023</item>
  </aChildPUIDs>
  <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.

Method: UnlinkObjectsEx

Description

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	
aChildPUIDs	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>
  <aChildPUIDs>
    <item>PROD_000023</item>
  </aChildPUIDs>
  <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.

Method: UnlinkObjectsExById

Description

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.

Method: UpdateObject

Description

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	

Method: UpdateObjectAttachment

Description

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	
sClassName	string	1..1	
sPUID	string	1..1	
sAttrName	string	1..1	
oFile	FileAttachment	0..1	

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	

Method: UpdateObjectEx

Description

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	

Method: UpdateTemplate

Description

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	

Method: UpdateTemplateFile

Description

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	

Method: UploadImageForHTMLAttribute

Description

Upload picture and 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>
      <encodedContents>iVBORw0KGgoAAAANSUheUgAAADwAAAA1CAIAAACSkR/tAAAAAXNSR0
      IARs4c6QAAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAAHxSURBVGh
      D7ZQxcslwEEU5FwfiPjyBI6TgAByBloouDTMumKGGjL2S1ptstgSlsYEVfmTSXYV7bMte/V4Q7p0
      K7p0K7p0K7p0Kz5AerX7rhHsPpIJ0mJGpWDYBMalxda1g6IPGZEW07YJZuv8O2mxlwkasyJGmK
      ChkyGNagXEIBM0FBYhbcia1aVfIGvWh0nvD0I3tRv26MTW+nhHyXE6Xn4vJnwdQVWhWPq+/Uq6
      NI/Xk23dNqFyuLnFBKtjZQR1H1QViqXJwN+w82DXkMpwXdvfL+5nIKPbvzm4IcnFWGhECKoKpdJk
      6WfjPNCfrrU+3tyj8FeFK7nu3QnZnKkasf/OgqpCoTQ7nSHwo5bRcldy2Q626H+nQ4UiJ92qjnR41R
      D2ulNrsGePJT1UjLiVC6oKZdL8huH18k+caSXnxC32h0S8hYYwhYKqQpE0zfY3jO46vm5cC8tc2H
      HnjyWAZT6oKhR.Jp7NxivpmW/zyHb9xfI5filsWgqIAiTZZxNrvx8Rxb8C33K9W30EAjQIBVKHwRZydr
      Vpd+gaxZXfoFsmYtQloMMkFDIUOagt58iP1N0NB5Jm0Q27UJZussThqDnzlibRCbVg1GjjEuTYjdaw
      STJjBVmhBjZgm2ziFPeiF06VZ06VZ06VZ06Va8ofTj8QO6Ch0Ds823OQAAAAABJRU5ErkJggg==
      </encodedContents>
    </item>
  </files>
</urn:UploadImageForHTMLAttribute>
```

Complex Types: rtmService

Complex Types

Name	Description
addObjectToCollectionResponse	
AppendRMDocumentExResult	
AppendRtmDocumentResult	
AsyncAppendRtmDocResponse	
AsyncCreateRtmDocResponse	
AsyncReplaceRtmDocResponse	
AttributeDef	
AttributeDefEx	
AttributeDefEx2	
AttributeDefExList	
AttributeDefEx2List	
AttributeDefInfo	
AttributeDefList	
AttributeDefSpecifier	
AttributeFormatDef	
Category	
CategoryAssign	
CategoryList	
CategoryReq	
ClassDef	
ClassDefEx	
ClassDefExList	
ClassDefInfo	
ClassDefList	
ClassSpecifier	
CmdStatus	
CollectionEx	
CollectionEx2	
CollectionEx2List	
CollectionExList	
CollectionInfo	
CollectionList	
CollectionSpecifier	
CollectionTransferRules	
CollectionTransferRulesEx	
CommandStatus	
CreateChildDocumentData	
CreateCollectionEx	
CreateRMDocumentExResult	
CreateRtmDocumentResult	

Name	Description
DatabaseList	
DocumentObject	
DocumentsFilter	
ECPObjInfo	
ECPsObjectInfo	
FileAttachment	
FindRequirementOut	
FormatDef	
getAsyncAppendRtmDocStatusResponse	
getAsyncCreateRtmDocStatusResponse	
getAsyncQueryRtmDocResponse	
getAsyncQueryRtmDocumentStatusResponse	
getAsyncReplaceRtmDocStatusResponse	
getAsyncStatusResponse	
GetAttributeDefInfoReply	
GetAttributeDefInfoRequest	
GetClassDefInfoReply	
GetClassDefInfoRequest	
GetCollectionInfoReply	
GetCollectionInfoRequest	
GetProjectInfoReply	
GetProjectInfoRequest	
GetUserAttributeDefInfoReply	
GetUserAttributeDefInfoRequest	
IdList	
ImageList	
ImageUploadResult	
ImageUploadResultList	
Link	
LinkClassDef	
LinkClassDefList	
LinkList	
ListItemDef	
ListItems	
LoginExResultInfo	
NameValue	
NameValueList	
ns-chapterDetails	
Object	
ObjectDef	
ObjectDetails	
ObjectDetailsList	
ObjectsList	

Name	Description
ProjectEx	
ProjectExList	
ProjectInfo	
ProjectList	
ProjectSpecifier	
ProposeNewRequirementResult	
RelationList	
RelationWithObjects	
removeObjectFromCollectionResponse	
ReplaceRequirementList	
ReplaceRequirementOut	
Report	
Requirement	
RequirementFindList	
RequirementIdentifier	
RequirementIdentifierList	
RequirementList	
RequirementOut	
Result	
RtmDocSnapshotOutList	
RtmDocSnapshot	
RtmDocSnapshotOut	
RtmDocument	
RtmDocumentList	
RtmDocumentProperties	
RtmDocumentWithCategory	
runTestAsyncInputData	
runTestAsyncResponse	
runTestAsyncResponseData	
Script	
SearchRequirement	
StatusMessage	
StatusMessageList	
StringList	
SyncXrefLink	
SyncXrefLinkList	
SystemInfo	
Template	
TemplateList	
testAsyncStatusResponse	
Transition	
Transitions	
TransitObject	

Name	Description
UserAttributeDefInfo	
UserValue	
UserValueList	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	


Complex Type: 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	


Complex Type: 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	

Complex Type: 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

Complex Type: 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..*	

Complex Type: 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

Complex Type: 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	
bVisable	boolean	1..1	
bMultiple	boolean	1..1	
bImplicit	boolean	1..1	
bHtml	boolean	1..1	
aFormat	AttributeFormatDef	1..1	
aValues	StringList	0..1	

Complex Type: 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..*	

Complex Type: 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 - than attribute will be specified by id 1 - than 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

Complex Type: 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	

Complex Type: 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'

Complex Type: 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

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	
Value	IdNameUnion	1..1	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	CollectionTransferRulesEx	0..1	

Complex Type: 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	

Complex Type: 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	CollectionTransferRulesEx	0..1	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	


Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	


Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: LinkList

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

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

Complex Type: 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

Complex Type: 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

Complex Type: 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	
blsAdmin	boolean	1..1	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	
sPUIID	string	1..1	
sRelationStatus	string	0..1	
ObjAttrs	NameValueList	0..1	

Complex Type: 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..*	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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>
```

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: Requirement

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

Component	Type	Occurs	Description
SEQUENCE		1..1	
sPUIID	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	

Complex Type: 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..*	

Complex Type: RequirementIdentifier

Derived By

Restricting anyType

Content Model

Contains elements as defined in the following table.

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

Complex Type: 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..*	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	
nDocumentId	int	1..1	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	
blsMaster	boolean	0..1	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	
sRMMinorVersion	string	0..1	
sRMMajorVersion	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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	
sPUIID	string	0..1	
nTransitionId	long	0..1	
sTransitionName	string	0..1	
aAttrList	NameValueList	0..1	

Complex Type: 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	
bVisible	boolean	1..1	
bMultiple	boolean	1..1	
bImplicit	boolean	1..1	
bHtml	boolean	1..1	
aFormat	AttributeFormatDef	1..1	
aValues	UserValueList	1..1	

Complex Type: 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	

Complex Type: 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: rtmService

Simple Types

Name	Description
cmdStatusCode	
IdNameEnumeration	
IdNameUnion	
reqStatusCode	
statusCode	

Simple Type: cmdStatusCode

Derived By

Restricting string

Enumeration

Value	Description
SUCCESS	
FAILURE	
COLLECTION-NOT-FOUND	

Simple Type: IdNameEnumeration

Derived By

Restricting string

Enumeration

Value	Description
ID	
NAME	

Simple Type: IdNameUnion

Derived By

Restricting string

Simple Type: reqStatusCode

Derived By

Restricting string

Enumeration

Value	Description
eCREATED	
eREPLACED	
eUPDATED	
eNOCHANGE	

Simple Type: statusCode

Derived By

Restricting string

Enumeration

Value	Description
STARTING	
ACTIVE	
FINISHED	
ERR	
NOT-FOUND	

rtmAdminService Web Service

Description

gSOAP 2.7.12 generated service definition

See Also

- [Methods](#)
- [Complex Types](#)

Methods: rtmAdminService

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
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	Creates a new LDAP user
CreateSSOUser	Creates 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

Method: AssignCategoryToGroup

Description

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>
```

Method: AssignGroupToProject

Description

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>
```

Method: AssignUsersInGroupPerCategory

Description

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:

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

Method: AssignUserToGroup

Description

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>epphoto</sUserName>  
  <sGroupName>read</sGroupName>  
</urn:AssignUserToGroup>
```


Method: AssignUserToProject

Description

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>
```

Method: CopyGroup

Description

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:

Create copy of group:

```
<urn:CopyGroup>
  <sessionId>1472146937</sessionId>
  <sGroupName>REPORTERS</sGroupName>
  <sSrcGroupName>icDemo Users</sSrcGroupName>
</urn:CopyGroup>
```

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>
```

Method: CopyUser

Description

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>
    <sLastName>Pettor</sLastName>
    <sPhone>3558832224</sPhone>
    <sEmail>jpettor@mail.com</sEmail>
  </anUserInfo>
</urn:CopyUser>
```

Method: CreateGroup

Description

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>
```

Method: CreateUser

Description

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>
```

Method: CreateLDAPUser

Description

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>
```

Method: CreateSSOUser

Description

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>11111111</sessionId>
  <sUserName>John</sUserName>
  <anUserInfo>
    <sFirstName>John</sFirstName>
    <sLastName>Harrison</sLastName>
  </anUserInfo>
</urn: CreateSSOUser>
```

Method: DeleteGroup

Description

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>
```


Method: DeleteUser

Description

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>
```

Method: GetGroupInfo

Description

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>
```

Method: GetGroups

Description

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>
```

Method: GetLoginSources

Description

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>
```

Method: GetUserInfo

Description

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>
```

Method: GetUsers

Description

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>
```

Method: GrantPermissionToGroup

Description

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>
```

Method: GrantPermissionToUser

Description

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>
```


Method: InheritPermissionForUser

Description

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>
```

Method: RevokePermissionFromGroup

Description

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>
```

Method: RevokePermissionFromUser

Description

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>
```

Method: UnassignCategoryFromGroup

Description

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>
```

Method: UnassignGroupFromProject

Description

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>
```

Method: UnassignUserFromGroup

Description

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>
```

Method: UnassignUserFromProject

Description

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>
```

Method: UpdateListAttributeValues

Description

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>
      <bDeleted>>false</bDeleted>
    </item>
    <item>
      <sName>Low</sName>
      <!--Optional:-->
      <bDefault>>true</bDefault>
      <bDeleted>>false</bDeleted>
    </item>
  </aValues>
</urn:UpdateListAttributeValues>
```


Method: UpdateUser

Description

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	
sUserName	string	1..1	
sPassword	string	0..1	
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>
```

Complex Types: rtmAdminService

Complex Types

Name	Description
AdditionalUserInfo	
Group	
GroupInfo	
GroupList	
IdList	
LoginSource	
LoginSources	
NameValue	
NameValueList	
Result	
StringList	
User	
UserInfo	
UserInfoExt	
UserList	
UserNameList	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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	

Complex Type: 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..*	

Complex Type: 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..*	

RM Restful Services

Sources

Name	Description
RM Restful Services	

RM Restful Services

See Also

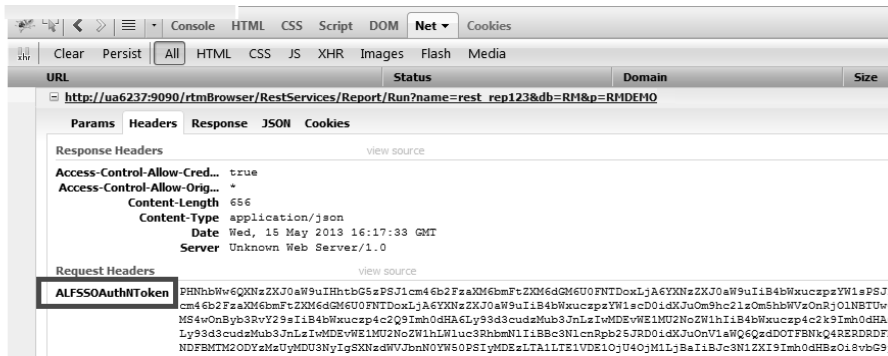
- [Authentication](#)
- [Methods](#)
- [General](#)

Authentication for Rest Services

Authentication is mandatory for using RM Rest services

Next authentication methods are allowed:

- 1) SSO Token in *ALFSSOAuthToken* header.



- 2) User name and Password
 - a) Basic access authentication (use the *Authorization* header)

The Authorization header is constructed as follows:

- Username and password are combined into a string "username:password"
- The resulting string literal is then encoded using Base64
- The authorization method and a space i.e. "Basic " is then put before the encoded string.

Example for ephoto:rtm: Authorization: Basic ZXBob3RvOnJ0bQ==

- b) User name and password in request parameters(also database and project should be provided):
RestServices/Report?name=myreport&u=ephoto&pwd=rtm&db=RM&p=RMDEMO

- 3) Session Id – can be retrieved from RM Login Web Service.
RestServices/Report?sessionId=1368617205&name=myreport&db=RM&p=RMDEMO
In this case db and project can be skipped

- 4) Authenticated encryption data (currently used only for get image request)
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: RestServices

Methods

Name	Description
Reports	Get reports
Report	Get report info
Run	Run report
Containers	Get containers
ContainerRequirements	Get requirements from the container

Method: Reports

Description: 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
category_name	string	0..1	Category name
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
 SEQUENCE		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=epphoto&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
}
```


Method: Report

Description: 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
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
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=epphoto&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
}

```

Method: Run

Description: 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=photo&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

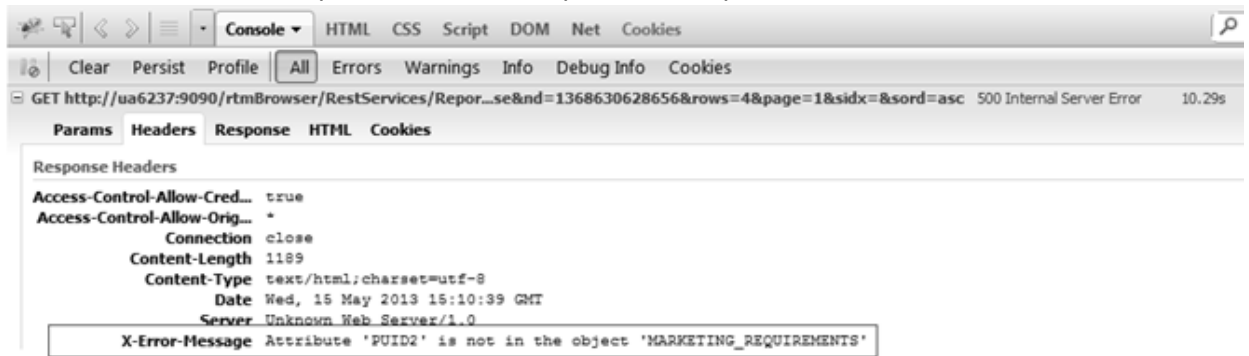
To run report with runtime parameters all values must be specified in the request:

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



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.

Method: Containers

Description: 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: <ul style="list-style-type: none"> • ALL - containers • D - documents • B - baselines • C - collections • 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
<small>REF</small> SEQUENCE		0..*	
id	int	1..1	Container id
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 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&collection
Id=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&collection
Id=5"
    }
  ],
  "message": "",
  "oiddata": null,
  "success": true,
  "total": 4
}
```


Method: ContainerRequirements

Description: 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
<small>REQ</small> 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
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://stl-qa-rm08:8080/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://stl-qa-rm08:8080/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://stl-qa-rm08:8080/rtmBrowser/cgi-bin/rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&class=Marketing_Requirements&puid=MRKT_000024&ver=1",
      "version": 1
    }
  ]
}
```

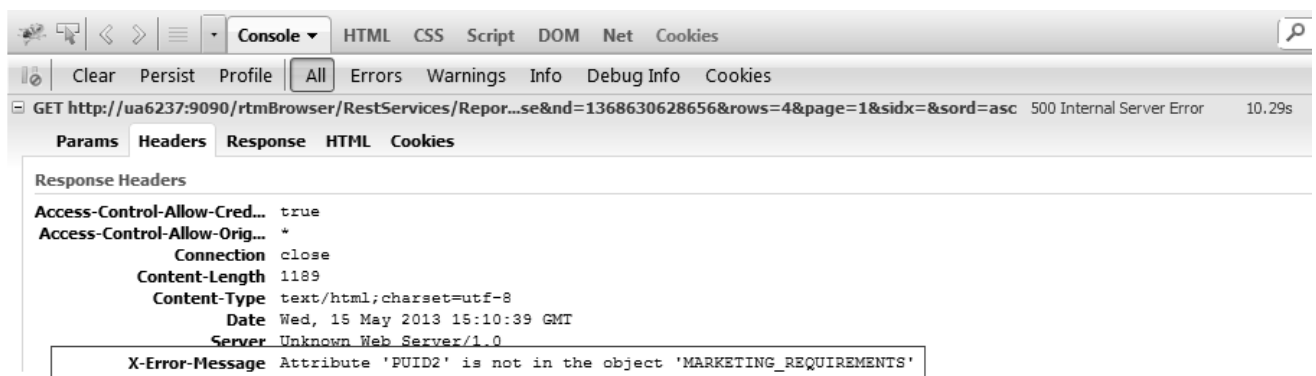
```
    },
    {
      "category": "RMDEMO/Functional/Design and Construction",
      "className": "Marketing_Requirements",
      "class_id": 1,
      "object_id": 35,
      "puid": "MRKT_000031",
      "status": "Current",
      "title": "System comes in different colors",
      "url":
"http://stl-qa-rm08:8080/rtmBrowser/cgi-bin/rtmBrowser.exe?goto=req&db=RM&proj=RMDEMO&class=Marketing_Requirements&puid=MRKT_000031&ver=1",
      "version": 1
    }
  ],
  "message": "",
  "oiddata": null,
  "success": true,
  "total": 4
}
```

General

General info about RM Restful Services:

- 1) URI: `http://<host>:<port>/rtmBrowser/RestServices/<ServiceName>/<Action>`
- 2) Support only JSON/JSONP format.
- 3) 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.*



- 4) RM Restful services are cookies independent. (don't set/retrieve cookies)