



SAP Web Intelligence RESTful web service SDK User Guide
■ SAP BusinessObjects Business Intelligence platform 4.1

2013-05-07

Copyright

© 2013 SAP AG or an SAP affiliate company. All rights reserved. No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice. Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors. National product specifications may vary. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty. SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries. Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

2013-05-07

Contents

Chapter 1	Introduction to Web Intelligence RESTful Web Service SDK.....	7
1.1	Document History.....	7
Chapter 2	Using the SAP Web Intelligence RESTful Web Service SDK.....	13
2.1	Before You Begin.....	13
2.2	HTTP Status Errors.....	13
2.3	To Activate Trace Log.....	15
2.4	Retrieving URL for Web Intelligence RESTful Web Service Request.....	15
Chapter 3	API Reference.....	17
3.1	Logon & Logoff to the BI platform.....	17
3.1.1	To log on to the BI platform.....	18
3.1.2	To log off the BI platform.....	21
3.2	Managing documents.....	22
3.2.1	Supporting localized documents on the BI Platform.....	23
3.2.2	Document: retrieving, copying, or creating.....	24
3.2.3	Document: getting the details, updating, or deleting.....	31
3.2.4	Document properties.....	36
3.2.5	Exporting documents.....	44
3.2.6	Managing styles, formats, fonts, skins, and charsets	50
3.2.7	Managing functions, operators, and variables.....	83
3.2.8	Managing alerters and tracking changes.....	100
3.2.9	Managing attachments and links.....	129
3.2.10	Adding a cache entry to a document.....	148
3.3	Managing the document lifecycle (LCM).....	150
3.3.1	Document state: managing the state of the document (Life Cycle Management).....	150
3.3.2	Snapshots: getting a list of snapshots or creating a snapshot for a document	154
3.3.3	Snapshots: restoring a document to a specific snapshot	158
3.4	Managing reports.....	160
3.4.1	Getting, moving, or creating, copying a report.....	161
3.4.2	Getting report details and deleting reports.....	169
3.4.3	Get the map of a report.....	174

3.4.4	Exporting reports.....	177
3.4.5	Getting the list of elements used in a report.....	188
3.4.6	Getting report element details, exporting a report element.....	190
3.4.7	Report structure: getting and updating the structure (specifications) of a report.....	197
3.4.8	Drilling on report data.....	211
3.5	Managing data providers.....	247
3.5.1	Getting a list of data providers or adding a data provider	247
3.5.2	Moving a data provider in a document	251
3.5.3	Getting the details of a data provider or updating or deleting a data provider	253
3.5.4	Changing the data provider.....	260
3.5.5	Flows: getting the flows count for a data provider	275
3.5.6	Flows: getting the details of a flow in CSV or XML format.....	276
3.5.7	Query specification: getting and updating the query specification	280
3.6	Managing BW connections and BEx queries.....	287
3.6.1	Getting the list of BW connections.....	287
3.6.2	Getting the details of a BW connection by ID.....	290
3.6.3	Browsing the details of a BW connection.....	292
3.6.4	Getting the outline of a BEx query.....	296
3.6.5	Getting the capabilities of a BEx query.....	299
3.7	Scheduling documents.....	301
3.7.1	About date and time formats.....	302
3.7.2	Getting the list of schedules for a document.....	303
3.7.3	Adding a schedule to a document.....	305
3.7.4	Getting the details of a schedule.....	323
3.7.5	Deleting or cancelling the document schedule.....	325
3.8	Refreshing documents.....	326
3.8.1	Getting the document refresh parameters before refreshing a document.....	327
3.8.2	Refreshing a document.....	332
3.8.3	Cancelling the refresh of a document.....	360
3.9	Managing universes.....	362
3.9.1	Getting the list of available universes.....	362
3.9.2	Getting the details of a universe.....	364
3.9.3	Getting the query capabilities of a universe.....	369
Chapter 4	Example workflow: Open, refresh, save a document, and export a report.....	375
Chapter 5	RESTful services quick reference section.....	377
5.1	Managing documents.....	377
5.2	Managing the document lifecycle (LCM)	382
5.3	Managing reports	383

5.4	Managing data providers	387
5.5	Managing BW connections and BEx queries.....	389
5.6	Scheduling and refreshing documents	390
5.7	Managing universes	391
Appendix A	More Information.....	393
Index		395

Introduction to Web Intelligence RESTful Web Service SDK

The Web Intelligence RESTful web service SDK is an API used for manipulating the following:

- manipulating Web Intelligence documents and reports
- retrieving data from a dataprovider
- retrieving a list of available universes and details of an universes
- scheduling documents

It cannot be used to edit/create SAP Web Intelligence documents.

The Web Intelligence RESTful web service SDK relies on the BI platform RESTful web services API for session management and repository access. Before starting with the Web Intelligence RESTful web service SDK, we strongly recommend that you to read the *Business Intelligence Platform RESTful Web Service Developer Guide* referenced below.

Related Topics

- <http://help.sap.com/bobip#section6>

1.1 Document History

The following table provides an overview of the most important document changes.

Version	Date	Description
SAP Web Intelligence RESTful web services SDK 4.0 Service Pack 5.	September, 2012	First release of this document.

Version	Date	Description
SAP Web Intelligence RESTful web services SDK 4.0 SP6 and 4.1	June, 2013	

Version	Date	Description
		<p>Browsing and Life Cycle Management</p> <ul style="list-style-type: none"> List all Web Intelligence documents in CMS repository Get Web Intelligence Document details Open / Save / Close Web Intelligence document Access and Export Document [xls(x), pdf, xml] Access and Export Report [xls(x), pdf, xml, html] Get Page of a report in listing or paginated mode [xls(x), pdf, xml, html] Access and Export Report Element [xls(x), pdf, xml, html] <p>Document Refresh</p> <ul style="list-style-type: none"> Refresh document with support of Text, Numeric, optional/mandatory prompts Prompting and resolution of multiple contexts Refresh document with support of Date and Date-Time List of Values management (included hierarchical) in prompt workflows Manage Prompt on SAP variable (including optional/mandatory, key dates) Nested prompts Support <p>Data provider</p> <ul style="list-style-type: none"> Get list of data providers of a Web Intelligence document Get details of a data provider of a Web Intelligence document Rename, sort or delete a data provider from a document Export data (data provider) as CSV or XML Purge document Get query statement generated by query on universes (SQL or MDX) Change data provider source workflow <p>UNV > UNV</p> <p>UNV > UNX</p> <p>UNV OLAP BW > BEX</p>

Version	Date	Description
		<p>UNIX > UNIX</p> <p>BEX > BEX</p> <p>Data source</p> <ul style="list-style-type: none"> • Get list of universes stored in the CMS • Get basic information about a universe • Enhanced detailed information about a universe • Browse list of BEx connections and BW BEx queries • Get details of a Bex query • Get data source query capabilities <p>Query authoring</p> <ul style="list-style-type: none"> • Get the query specification associated to a data provider • Update the query specification associated to a data provider <p>Document authoring</p> <ul style="list-style-type: none"> • Create a blank document • Add a report in a document and set its report specification • Delete a report from a document • Get/Create/Delete/Update Track Data Change • Get/Create/Delete/Update Alerters • Get/Create/Delete/Update Styles • Get/Create/Delete/Update Skins (background images) • Get/Create/Delete/Update Variable expression • Manage drill and create drill snapshot • Manage document state through storage token (snapshot) • Get custom format number from the document • Get from server default document configuration • Get font mappings, CSS styles, format numbers, default skins, supported operators <p>Scheduling Properties</p>

Version	Date	Description
		<ul style="list-style-type: none">• Main recurrences (Once, Hourly, Daily, Weekly, Monthly, ...)• Support of server group• Get details of a schedule• Add new schedule

Using the SAP Web Intelligence RESTful Web Service SDK

The SAP Web Intelligence RESTful web service SDK complies with the RESTful methodology. You can access the Web Intelligence RESTful web service SDK using any programming language that supports making HTTP requests.

The `cURL` tool has excellent support for HTTP requests, so samples (for Windows and UNIX) in this guide are provided using this tool. You can find more information about `cURL` at the link referenced below.

Related Topics

- <http://curl.haxx.se/>

2.1 Before You Begin

Before you begin, be sure that the WebApplicationContainerServer (WACS) server is started and enabled.

To start the server, launch the Central Management Console.

1. In the "Organize" part select **Servers**
2. Unfold the **Server Categories** folder and select **Core Services**.
3. Right-click MySIA.WebApplicationContainerServer->Start Server
4. Right-click MySIA.WebApplicationContainerServer->Enable Server

2.2 HTTP Status Errors

The table below lists the set of status codes that provide status for HTTP requests.

Table 2-1: HTTP standard error codes

Status code	Error	Description
200	Success	Successful request.
400	Bad request	The requested resource exists, but the request contains errors.
401	Failed to logon or invalid session	Logon failed. Check that the user name, password, and server name are correct.
403	Access denied	You do not have permission to operate on the requested resource. The current session may have expired. Log on to obtain a new session.
404	Service is not available	The requested service is not provided by the RESTful web service SDK.
405	Invalid request method	A request was made using a method that was not supported by the resource. For example, using a <code>PUT</code> request on a read-only resource.
406	Not acceptable	The requested resource cannot generate the content type specified by the <code>Accept</code> attribute of the request header.
408	BI platform server timeout	The server timed out waiting for the request.
415	Unsupported media type	The request contains a media type that the server or resource does not support.

Status code	Error	Description
500	RESTful web service internal error	An unclassified error occurred. See the response body for more information.
503	RESTful web service plugin not found	RESTful web services are not available. Verify that RESTful web services are configured correctly.

2.3 To Activate Trace Log

To get details of the returned error:

1. Launch the "Central Management Console."
2. Select "Servers"
3. Expand "Service Categories" and select "Core Services" folder
4. Right-click the WACS (MySIA.WebApplicationContainerServer) server to get properties.
5. Check **Show Error Stack** in the "RESTful Web Service information."

2.4 Retrieving URL for Web Intelligence RESTful Web Service Request

To use the Web Intelligence RESTful web service SDK you must know the protocol, server name, port number and path of the service that listens to RESTful web service requests.

By default, the URL is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

Note:

vx corresponds to the version. For this release it is v1.

API Reference

This chapter lists the set of URLs of Web Intelligence RESTful web services plus the URL used to log onto the BI platform.

URLs are organized by category:

- Logon and logoff to the BI platform
- Document management
- Document Life Cycle Management
- Report management
- Data Provider access
- Scheduling
- Managing BW connections and BEx queries
- Refresh document
- Universe Management

Note:

Refer to *BI platform RESTful Web Service Developer Guide* at the link below for more information on the platform services.

Related Topics

- <http://help.sap.com/bobip#section6>
- [Logon & Logoff to the BI platform](#)
- [Managing documents](#)
- [Managing the document lifecycle \(LCM\)](#)
- [Managing reports](#)
- [Managing data providers](#)
- [Scheduling documents](#)
- [Managing BW connections and BEx queries](#)
- [Refreshing documents](#)
- [Managing universes](#)

3.1 Logon & Logoff to the BI platform

Using the `GET` method, retrieve an XML template to log onto the BI platform with user name, password, and authentication mode, then use the `POST` method to log on and obtain a logon token.

Log off by invalidating the logon token.

These features are developed by the BI platform and fully described in the *Business Intelligence Platform RESTful Web Service Developer Guide* at the link referenced below.

The default URL to request the BI platform web services is the following:

```
http://<serverName>:6405/biprws/
```

Related Topics

- <http://help.sap.com/bobip#section6>
- [To log on to the BI platform](#)
- [To log off the BI platform](#)

3.1.1 To log on to the BI platform

This section explains how to logon to the BI platform.

For a more complete explanation, refer to *BI platform RESTful Web Service Developer Guide* at the link referenced at the end of the section.

To retrieve an XML template

The XML template that you retrieve is used for the body of the `POST` request described further on.

Request:

```
GET http://<serverName>:6405/>biprws/logon/long
```

Header	Value
Accept	application/xml or application/json

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server

Header	Value
Date	Date and time of response
Content-Type	Type of content in the response body
Content-Length	Length of content in the response body

Body
A template to use to populate the POST request

Example:**Windows**

```
curl -G http://<serverName>:6405/biprws/logon/long -H accept="application/xml"
```

UNIX

```
curl -G http://<serverName>:6405/biprws/logon/long -H "accept=application/xml"
```

Response

```
<attrs xmlns="http://www.sap.com/rws/bip">
  <attr name="userName" type="string">
  </attr>
  <attr name="password" type="string">
  </attr>
  <attr name="auth" type="string" possibilities="secEnterprise,secLDAP,secWinAD,secSAPR3">secEnterprise
  </attr>
</attrs>
```

To receive a logon token providing user name and password

Use the XML template retrieved by the GET request described above.

Request:

```
POST http://<serverName>:6405/biprws/logon/long
```

Header	Value
Content-Type	application/xml or application/json
Accept	application/xml or application/json

Body	Description
<p>XML</p> <pre><attrs> <attr name="userName" type="string"/></attr> <attr name="pass" type="string"></attr> <attr name="auth" type="string"></attr> </attrs></pre> <p>Json</p> <pre>{"userName":username,"password":password,"auth":se cEnterprise}</pre>	<p>To define the user name, password and type of authentication.</p> <p>Authentication can be: secEnterprise, se cLDAP, secWinAD or secSAPR3</p>

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server
X-SAP-LogonToken	<p>A logon token. The SAP logon is in quotation marks.</p> <p>X-SAP-LogonToken:"tokenValue"</p>
Date	Date and time of the response
Content-Type	Type of content in the response body

Body	Description
<pre><attr name="logonToken" type="string"></pre>	<p>Contains a copy of the logon token. The logon token must be converted from XML-encoded format to its original format before it can be used</p>

Example: Login

Note:

Credentials are specified in the `login.xml` file called by "@login.xml" in cURL command.

login.xml details.

```
<attrs>
  <attr name="userName" type="string">Administrator</attr>
  <attr name="password" type="string">Password</attr>
  <attr name="auth" type="string">secEnterprise</attr>
</attrs>
```

Windows

```
curl -i -X "POST" -H "content-type: application/xml" -H "accept: application/xml" -d "@login.xml"
http://<serverName>:6405/biprws/logon/long
```

Note:

Once you get the X-SAP-LogonToken, you can use it through a variable. Copy the token value, in command line type: Set tokenValue=<pasted_value> and use the variable in your cURL request as is: %tokenValue%

The following request exports the X-SAP-LogonToken to xsaplogontoken.txt file.

```
curl -i -X "POST" -H "content-type: application/xml" -H "accept: application/xml" -d "@login.xml"
http://<serverName>:6405/biprws/logon/long | find "X-SAP-LogonToken: " > xsaplogontoken.txt
for /f "tokens=1,2 delims= " %a in (xsaplogontoken.txt) do @echo %b > xsaplogontoken.txt
```

UNIX

Note:

The X-SAP-LogonToken is exported in the logtok variable. This variable is used by all other URLs.

```
export logtok=`curl -i -X "POST" -H "content-type: application/xml" -H "accept: application/xml" -d "@login.xml"
http://<serverName>:6405/biprws/logon/long | grep "X-SAP-LogonToken: " | awk '{print $2;}'`
```

Related Topics

- <http://help.sap.com/bobip#section6>

3.1.2 To log off the BI platform

To invalidate the logon token and log off the BI platform.

Request:

POST http://<serverName>:6405/biprws/logoff

Header	Value
Accept	application/xml
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server
Date	Date and time of response
Content-Type	Type of content in the response body
Content-Length	Length of content in the response body

Body
None

Example:**Windows**

```
curl -i -X "POST" -H "accept: application/xml" -H 'X-sap-logontoken:"<TokenValue>"' http://<server Name>:6405/biprws/logoff
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept: application/xml" -H "X-sap-logontoken:$logtok" http://<server Name>:6405/biprws/logoff
```

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 24 May 2012 14:41:20 GMT
Content-Type: text/xml
Content-Length: 0
```

Related Topics

- [To log on to the BI platform](#)

3.2 Managing documents

Below are the main operations available to manage Web Intelligence documents.

You can get information about:

- Configuration formats
- Custom formats
- Documents
- Font mappings
- Functions
- Operators
- Report skins

You can manage

- Alerters
- Attachments
- Change tracking
- Documents, including exporting documents
- Links
- Styles
- Stylesheets (CSS)

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

Note:

Management of auto save & auto recovery configuration is currently not supported.

Related Topics

- [Document: retrieving, copying, or creating](#)
- [Document: getting the details, updating, or deleting](#)
- [Exporting a document in paginated mode](#)

3.2.1 Supporting localized documents on the BI Platform

RESTful web services SDK supports BI platform localized documents. In order to use localized documents, include the following declarations in the header:

Additional optional header declarations	Value
Accept-Language	This header declaration defines the Product Locale (the UI Locale). languages declared in the localized or preferred viewing locales of the documents. For example, en_US, fr_FR
X-SAP-PVL	This header declaration defines the Preferred Viewing Locale, also known as the Content Locale.

The Raylight service will open one instance of the Web Intelligence document in memory for each Preferred Viewing Locale requested by the client.

Example: To get the contents of a document in a specific locale: fr-FR

Windows

```
curl -G -i -H "accept:application/xml" "X-SAP-PVL:fr-FR" -H X-SAP-LogonToken:"""%tokenValue%"""  
http://<serverName>:6405/biprws/raylight/v1/documents/12345
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" "X-SAP-PVL:fr-FR" -H 'X-SAP-LogonToken:"$logtok"' http://<server  
Name>:6405/biprws/raylight/v1/documents/12345
```

3.2.2 Document: retrieving, copying, or creating

Use this URL to:

- Get the document list from the CMS (GET <url>/documents).
- Copy a document (POST <url>/documents).
- Create an empty document (POST <url>/documents).

Getting the Web Intelligence document list from the CMS

This retrieves the list of document stored in the CMS. The documents are sorted by name. The list depends on user access rights. You can also specify the number of documents to return for the list and the first document to be used as the start document in the document list that you want to retrieve.

Request:

```
GET http://<serverName>:6405/biprws/raylight/vx/documents
```


Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
limit	Optional. Number of documents to return Default = 10 Minimum = 0 Maximum = 50
offset	Optional. Indicator for the beginning of the list Default 0 Minimum = 0

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

List of Web Intelligence documents sorted by name. The list depends on user access rights. For each document:

<id>: Document ID

<cuid>: Unique document ID

<name>: Document name

<folderId>: Folder ID

<state>: State of document used by Raylight. Values can be:

- **Unused** The document has not been loaded by Raylight
- **Original** The document has been loaded by Raylight but not modified
- **Modified** The user sent Raylight requests which altered the state of the document.

Example: To get one document from the CMS**Windows**

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents?limit=1
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i "accept:application/xml" -H "X-SAP-LogonToken: "$logtok" http://<serverName>:6405/biprws/raylight/v1/documents?limit=1
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<documents>
  <document>
    <id>6804</id>
    <cuid>AbrVz6E95lNMtjZk4KLzPuM</cuid>
    <name>BOF-ROLE_AA_ActiveHierOnCtry - Levels 02 and 03</name>
    <folderId>7611</folderId>
    <state>Original</state>
  </document>
  <document>
    <id>6861</id>
    <cuid>AUwfvPdEG3xOoINoNEzDquQ</cuid>
    <name>This is a sample for BOF-QRY_SIMPLE - Filter not in Result</name>
    <folderId>7611</folderId>
    <state>Unused</state>
  </document>
  <document>
    <id>6869</id>
    <cuid>AclwDFEOuvNHtovI4H.anSM</cuid>
    <name>BOF-QRY_SIMPLE - Filter on Country</name>
```

```

<folderId>48162</folderId>
<state>Original</state>
</document>
<document>
  <id>7048</id>
  <cuid>AUiVpegCKnFBh4AX8YxMT5A</cuid>
  <name>BOF-QRY_SIMPLE - Filter on Prompt not in Result</name>
  <folderId>7611</folderId>
  <state>Modified</state>
</document>
<document>
  <id>7125</id>
  <cuid>AWa48YHHT6FFiUMP.T5wtBs</cuid>
  <name>BOF-QRY_SIMPLE - Query Prompt not in Result</name>
  <folderId>7611</folderId>
  <state>Unused</state>
</document>
<document>
  <id>5121</id>
  <cuid>ARnGKBoLv7lDuK_UWMUli5Q</cuid>
  <name>Chart demo</name>
  <description>Shows how data can be visualized on different kinds of charts. And the features supported
on the charts. </description>
  <folderId>5127</folderId>
  <state>Unused</state>
</document>
</documents>

```

Copying a Web Intelligence document

You must provide the ID of the document to copy through the `sourceId` parameter `sourceId=<Web_Intelligence_document_id>`. You define the name of the document and folder in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; `copy.xml`.

Request:

POST `http://<serverName>:6405/biprws/raylight/vx/documents?sourceId=<Web_Intelligence_document_id>`

Header	Value
Accept	application/xml or application/json
Content-type	application/xml or application/json

Parameter	Description
sourceId	<p>Mandatory. Integer. The identifier of the Web Intelligence document to copy.</p> <p>Minimum = 1</p> <p>Maximum = highest document number in existing documents.</p>

Body (XML)	Description
<pre><document> <name> </name> <folderId> </folderId> </document></pre>	<p><name>: The name of the copied document in the BI launch pad folder</p> <p><folderId>: The BI launch pad folder where the document is copied. By default: the folder of the original document.</p>

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server
Date	Date and time of the response
Content Type	application/xml or application/json
Content Length	Length of content in the response body

Body

In case of success:

```
<success>
<message>Document has been created successfully.</message>
<id>5678</id>
<success>
```

Otherwise: <error> ... </error>

Example: To copy document ID 4990**Note:**

The name of the copied document is provided in the `copy.xml` file called by `-d "@copy.xml"` in cURL command. The copied document is assigned the ID 5875 and appears in the BI launch pad folder as: Copy of 4990.

`copy.xml` details

```
<document>
  <name>
    Copy of 4990
  </name>
</document>
```

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" -d "@copy.xml" http://<serverName>:6405/biprws/raylight/v1/documents?sourceId=4990
```

UNIX**Note:**

Uses the `logtok` variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok"" -d "@copy.xml" http://<serverName>:6405/biprws/raylight/v1/documents?sourceId=4990
```

Response

```
HTTP/1.1 200 OK
Server:Apache-Coyote/1.1
Date: Mon, 04 Jun 2012 11:59:02 GMT
Content-Type: application/xml
Content-Length: 204

<success>
<message>The resource of type 'document' with identifier '5875' has been successfully created.</message>
<id>5875</id>
</success>
```

Creating an empty Web Intelligence document

Use this command to create an empty document. The body of the request is described in the `.xml` file, for example; `newdocument.xml`, and is provided using `-d "@copy.xml"`.

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents

Header	Value
Accept	application/xml or application/json
Content-type	application/xml or application/json

Body (XML)	Description
<pre><document> <name>Name_of_new_document </name> <folderId>Id_of_folder</folderId> </document></pre>	<p><name>: The name of the empty document you create. If you do not assign a name, the system will automatically assign an ID.</p> <p><folderId>: The BI launch pad folder where the document is created. If you do not declare a folder, the default BI launch pad folder is used.</p>

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server
Date	Date and time of the response
Content Type	application/xml or application/json

Body

In case of success:

```
<success>
```

```
<message>The resource of type 'Document' with identifier '{documentId} '
has been successfully created.</message>
```

```
<id>{documentId}</id>
```

```
</success>
```

Otherwise: `<error> ... </error>`

Example: To create a new Web Intelligence document**Windows**

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to
kenValue%"" -d "@newdocument.xml" http://<serverName>:6405/biprws/raylight/v1/documents
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$log
tok" -d "@newdocument.xml" http://<serverName>:6405/biprws/raylight/v1/documents
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<success>
<message>The resource of type 'Document' with identifier '5022' has been successfully created.</message>
<id="5022"></id>
</success>
```

Related Topics

- [To log on to the BI platform](#)
- [Document: getting the details, updating, or deleting](#)

3.2.3 Document: getting the details, updating, or deleting

Use this URL to do the following:

- Get the details of a document (GET <url>/documents/{documentId})
- Update a document (close/save) (PUT <url>/documents/{documentId})
- Delete a document (DELETE <URL>/documents/{documentId})

The Web Intelligence document is referenced by its ID.

It also exports document in various formats.

Format can be:

- XML
- PDF
- Excel 2003
- Excel 2007

It is also used to manage document life cycle (change the state, close, save, discard changes of a Web Intelligence document) using the PUT method. This feature is documented in the Life Cycle Management section. See the link referenced at the end of the section.

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Retrieving the details of a given Web Intelligence document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
trackerDocumentId	Optional query parameter - it allows to provide an identifier of a reference document for the trackdata feature. It must be provided only when the document is in the 'Unused' state.

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Details of the document identified by:

<id>, <cuid>, <name>, <folderId>, <path>, The last update date and time and <scheduled> false or true if the document has been scheduled.

And document properties

<updated> : The date and the time of the last update

<createdBy>: The name of the document creator

<lastAuthor>: The name of the last person who modified the document

<size>: The size of the document

<refreshOnOpen>: "true" If the document is set to be refreshed at the open time, otherwise "false"

<state>: State of document used by Raylight.

Values can be: Unused, Original, and Modified. Unused means that the document has not been loaded by Raylight. Original means that the document has been loaded by the web service but not modified. Modified means that the user sent requests which altered the state of the document.

Example: To get the details of document ID 8022**Windows**

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<server Name>:6405/biprws/raylight/v1/documents/1458
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/1458
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<document>
  <id>4958</id>
  <cuid>AQtkbbSqN4N0j3ydf.Sw1lY</cuid>
  <name>Formatting Sample</name>
  <description>This is a sample document showing formatting capabilities.</description>
  <folderId>8246</folderId>
  <path>My Favorites/web Intelligence</path>
  <updated>2012-09-05T14:25:05.277+02:00</updated>
    <scheduled>true</scheduled>
    <state>Unused</state>
    <createdBy>Administrator</createdBy>
    <lastAuthor>User1</lastAuthor>
    <size>29410</size>
    <refreshOnOpen>false</refreshOnOpen>
</document>
```

Deleting a document

You can delete a document referenced by its ID.

Request:

```
DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}
```

Note:

{documentId}: The identifier of the Web Intelligence document to delete, can be retrieved from the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	The ID of the document to delete.

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body
In case of success, "The resource of type 'Document' with identifier {documentId} has been successfully removed". Otherwise <error> ... </error>

Example: To delete document ID 8022

Windows

```
curl -X "DELETE" -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/8022
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -X "DELETE" -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Document' with identifier '8022' has been successfully removed.</message>
  <id="5022"></id>
</success>
```

Related Topics

- [To log on to the BI platform](#)
- [Document: retrieving, copying, or creating](#)
- [Exporting a document in paginated mode](#)
- [Document state: managing the state of the document \(Life Cycle Management\)](#)

3.2.4 Document properties

Use these functions to list or edit the document properties that are visible in the "Document Summary". Certain settings are attributed automatically and cannot be set manually (for example, the last refresh time).

Related Topics

- [Getting the properties of a document](#)
- [Update the properties of a document](#)

3.2.4.1 Getting the properties of a document

Get the properties of a document referenced by its `documentId` parameter. (GET `<url>documents/{documentId}/properties`)

Note:

`{documentId}`: The identifier of the Web Intelligence document is retrieved in the document list by:
GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Retrieves the properties of a document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/properties`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
<code>documentId</code>	A valid ID identifier of a Web Intelligence document

Response

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Body (XML)

The properties of the document:

```
<property key="SI_CUID">ATuas0Mb.mxHpfoi7Xrwfo</property>
<property key="lastrefreshetime">1352999520</property>
<property key="lastrefreshduration">1</property>
<property key="lastsavedby">Administrator</property>
<property key="enablealternateusertoeditdoc">true</property>
<property key="creatorversion">14.0.6.894</property>
<property key="disablequerypanel">false</property>
<property key="lastrefreshdate">15 novembre 2012 18:12:00 GMT+01:00</property>
<property key="enhancedViewing">false</property>
<property key="current_doc_id">9939</property>
<property key="osversion">?</property>
<property key="modificationtime">1352999598</property>
<property key="stripquery">true</property>
<property key="modificationdate">15 novembre 2012 18:13:18 GMT+01:00</property>
<property key="name">doc_unv_drill_enable</property>
<property key="createdby">Administrator</property>
<property key="tdcactivated">false</property>
<property key="reportselected">1</property>
<property key="nameinrepo">doc_unv_drill_enable</property>
<property key="permanentregionalformatting">false</property>
<property key="reporterverversion">14.0.6.894</property>
<property key="repositorytype">C</property>
<property key="locale">en_US</property>
<property key="refreshonopen">false</property>
<property key="docrepoid">9939</property>
<property key="documenttype">WID</property>
```

Body (XML)

```

<property key="autosynchro">false</property>
<property key="hassamplingresults">false</property>
<property key="tdcmodeauto">true</property>
<property key="effectiverefreshonopen">false</property>
<property key="contentlocale">fr_FR</property>
<property key="mergeprompts">true</property>
<property key="documentsize">31070</property>
<property key="extendmergedimension">false</property>
<property key="autorefresh">false</property>
<property key="creationtime">1352999595</property>
<property key="creationdate">15 novembre 2012 18:13:15 GMT+01:00</property>
<property key="ispartiallyrefreshed">false</property>
<property key="nbqaawsconnection">0</property>
<property key="documentversion">14.0.6.894</property>

```

Example: To get the properties of a document**Note:**

Retrieve details of the report (ID 3) from the document (ID 7858)

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/properties
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/properties
```

Response

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 14:24:12 GMT
Content-Type: application/xml
Content-Length: 167

```

```

<properties>
  <property key="SI_CUID">ATuas0Mb.mxHpfi0i7Xrwfo</property>

```



```

<property key="lastrefreshedtime">1352999520</property>
<property key="lastrefreshduration">1</property>
<property key="lastsavedby">Administrator</property>
<property key="enablealternateusertoeditdoc">true</property>
<property key="creatorversion">14.0.6.894</property>
<property key="disablequerypanel">false</property>
<property key="lastrefreshdate">15 novembre 2012 18:12:00 GMT+01:00</property>
<property key="enhancedViewing">false</property>
<property key="current_doc_id">9939</property>
<property key="osversion">?</property>
<property key="modificationtime">1352999598</property>
<property key="stripquery">true</property>
<property key="modificationdate">15 novembre 2012 18:13:18 GMT+01:00</property>
<property key="name">doc_unv_drill_enable</property>
<property key="createdby">Administrator</property>
<property key="tdcactivated">false</property>
<property key="reportselected">1</property>
<property key="nameinrepo">doc_unv_drill_enable</property>
<property key="permanentregionalformatting">false</property>
<property key="reporterterversion">14.0.6.894</property>
<property key="repositorytype">C</property>
<property key="locale">en_US</property>
<property key="refreshonopen">false</property>
<property key="docrepoid">9939</property>
<property key="documenttype">WID</property>
<property key="autosynchro">false</property>
<property key="hassamplingresults">false</property>
<property key="tdcmodeauto">true</property>
<property key="effectiverefreshonopen">false</property>
<property key="contentlocale">fr_FR</property>
<property key="mergeprompts">true</property>
<property key="documentsize">31070</property>
<property key="extendmergedimension">false</property>
<property key="autorefresh">false</property>
<property key="creationtime">1352999595</property>
<property key="creationdate">15 novembre 2012 18:13:15 GMT+01:00</property>
<property key="ispartiallyrefreshed">false</property>
<property key="nbqaawsconnection">0</property>
<property key="documentversion">14.0.6.894</property>
</properties>

```

Related Topics

- [Update the properties of a document](#)

3.2.4.2 Update the properties of a document

Update the properties of a document referenced by its `documentId` parameter. (PUT `<url>documents/{documentId}/properties`)

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/properties`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Body
<p>The document properties that you want to update, in an XML file, for example, define a file called docprop.xml:</p> <pre><properties> <property key="refreshonopen">true</property> </properties></pre>

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>

<message>The resource of type 'properties' has been successfully updated.</message>

<id>{documentId}</id>

</success>
```

Example: To update the document properties for document ID 9939

You define the report properties in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example, properties.xml.

Request:

Windows

```
curl -i -X "PUT" -H "accept:application/xml" X-SAP-LogonToken:"""%tokenValue%"""" -d "@properties.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9939/properties
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" 'X-SAP-LogonToken:$logtok' -d "@properties.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9939/properties
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<success>The resource of type 'properties' has been successfully updated.
<id>9939</id>
</success>
```

Related Topics

- [Getting the properties of a document](#)

3.2.5 Exporting documents

You can export a document in two ways: the entire document, or in paginated mode. You use the two following urls:

- Export a document (GET -s <url>/documents/{documentId} [?parameters])
- Export a document in paginated mode (GET -s <url>/documents/{documentId}/pages)

The output format can be:

- XML
- PDF
- Excel 2003
- Excel 2007

Related Topics

- [Exporting an entire document](#)
- [Exporting a document in paginated mode](#)

3.2.5.1 Exporting an entire document

You can export documents in the following formats:

- XML
- PDF
- Excel 2003
- Excel 2007

Exporting a document

Note:

If HTML output is chosen, image links will be generated by Raylight: thus, the logon token must still be valid when HTML output is displayed (to be able to get images from the generated links)

Request:

```
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId} [?parameters]
```

Header	Value
Accept	text/xml or application/pdf or application/vnd.ms-excel (for Excel 2003) or application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007)
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	a valid ID identifier of a WebI document

Table 3-44: Export document: Optional parameter for all supported formats

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any) . Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats.

Table 3-45: Export Document: Optional Parameter for Excel 2003 and 2007

Parameter	Description
optimized	Boolean. Default: <code>false</code> the generated output is not optimized for calculations inside Excel. Otherwise, <code>true</code> .

Response:

Body

The exported file in the format defined by `accept:[text/xml]|[application/pdf]|[application/vnd.ms-excel]|[application/vnd.openxmlformats-officedocument.spreadsheetml.sheet]`. The document is saved as the `documentId` with the output extension type.

Example: To export a document in XML format**Note:**

Exports the XML result in the `xmlfile.xml`, with a resolution of 150 dpi for any generated charts.

Windows

```
curl -G -s -H "accept:text/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8022?dpi=150 > xmlfile.xml
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:text/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8022?dpi=150 > xmlfile.xml
```

Example: To export a document in PDF format

Exports the result with the file name `exportpdf.pdf`. The default resolution of 300 dpi for any generated charts is used.

Windows

```
curl -G -s -H "accept:application/pdf" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8022 > exportpdf.pdf
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/pdf" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8022 > exportpdf.pdf
```

Example: To export a document in Excel 2003 format

Exports the result with the file name `excel2003.xls` file. The output is optimized for calculations inside Excel, and any generated charts will have a resolution of 150 dpi.

Windows

```
curl -G -s -H "accept:application/vnd.ms-excel" -H X-SAP-LogonToken:"""%tokenValue%""" "http://<serverName>:6405/biprws/raylight/v1/documents/8022?optimized=true&dpi=150" > excel2003.xls
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/vnd.ms-excel" -H 'X-SAP-LogonToken:"$logtok"' "http://<serverName>:6405/biprws/raylight/v1/documents/8022?optimized=true&dpi=150" > excel2003.xls
```

Example: To export a document in Excel 2007 format

Exports the result in the `excel2007.xls` file. The output is optimized for calculations inside Excel.

Windows

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H 'X-SAP-LogonToken:""%tokenValue%""' "http://<serverName>:6405/biprws/raylight/v1/documents/8022?optimized=true" > excel2007.xlsx
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H 'X-SAP-LogonToken:"$logtok"' "http://<serverName>:6405/biprws/raylight/v1/documents/8022?optimized=true" > excel2007.xlsx
```

3.2.5.2 Exporting a document in paginated mode

Exports in paginated mode to various formats a Web Intelligence document referenced by its ID.

The output format can be:

- XML
- PDF
- Excel 2003
- Excel 2007

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Request:

```
GET -s http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/pages
```

Header	Value
Accept	<ul style="list-style-type: none"> text/xml application/pdf application/vnd.ms-excel (for Excel 2003) application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007)
X-SAP-LogonToken	The logon token value, in quotation marks

Table 3-48: Optional parameters for all supported formats

Parameter	Description
dpi	<p>Resolution in dots per inch (dpi) for generated charts (if any). Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats</p>
mode	normal or quickDisplay

Table 3-49: Optional Parameters mode=normal

Parameter	Description
orientation	<p>Page orientation. Used to force a specific page orientation. Values: portrait, landscape</p>
widthScaling	<p>Number of pages per report displaying in width. Default:0 means no constraint in width.</p>
heightScaling	<p>Number of pages per report displaying in height. Default: 0 means no constraint in height.</p>

Table 3-50: Export Document: Optional Parameter for Excel 2003 and 2007

Parameter	Description
optimized	Boolean. Default: <code>false</code> . Otherwise <code>true</code> , the generated output is then optimized for calculations inside Excel.

Response:

Body
The exported file in the format defined by <code>accept: [text/xml] [application/pdf] [application/vnd.ms-excel] [application/vnd.openxmlformats-officedocument.spreadsheetml.sheet]</code> . The document is saved as the <code>documentId</code> with the output extension type.

Example: Export document in XML (paginated mode)

Note:

Exports the XML result in `xmlfile.xml`

Windows

```
curl -G -s -H "accept:text/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages > xmlfile.xml
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:text/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages > xmlfile.xml
```

Example: Export document in PDF (paginated mode)

Note:

Exports the PDF result in `exportpdffile.pdf`

Windows

```
curl -G -s -H "accept:application/pdf" -H X-SAP-LogonToken:"""%tokenValue%""" "http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages?mode=quickDisplay" > exportpdffile.pdf
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/pdf" -H 'X-SAP-LogonToken:$logtok' "http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages?mode=quickDisplay" > exportpdf.pdf
```

Example: Export document in Excel 2003 format (paginated mode)

Exports the result in `excel2003.xls` file.

Windows

```
curl -G -s -H "accept:application/vnd.ms-excel" -H 'X-SAP-LogonToken: ""%tokenValue%""' http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages?mode=normal > excel2003.xls
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/vnd.ms-excel" -H 'X-SAP-LogonToken:$logtok' "http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages?mode=normal&orientation=landscape" > excel2003.xls
```

Example: Export document in Excel 2007 format (paginated mode)

Exports the result in `excel2007.xls` file.

Windows

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -s -H 'X-SAP-LogonToken: ""%tokenValue%""' "http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages?optimized=true" > excel2007.xlsx
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -s -H 'X-SAP-LogonToken:$logtok' http://<serverName>:6405/biprws/raylight/v1/documents/8022/pages?optimized=true > excel2007.xlsx
```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [To log on to the BI platform](#)

3.2.6 Managing styles, formats, fonts, skins, and charsets

You can manage the following visual aspects of your documents and reports:

Configuration formats

Custom formats

Font mappings

Report skins

Styles

CSS stylesheets

Charsets

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.1 Getting the list of declared configuration formats

Get a list of all declared formats on the Web Intelligence Server for the default locale. Returns the list of default formats and available formats.

The results depends on the Preferred Viewing Locale (PVL). If you need a specific locale setting, the locale must be passed in the header of the HTTP request.

Request

GET `http://<serverName>:6405/biprws/raylight/vx/configuration/formats`

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json

Header	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Body (XML)

Details of the available formats identified by:

`<format default="true" type="Number" sample="1 234,57">` The default format for numbers. Similarly, the default format for "Currency", "DateTime", "Boolean", "Date", and "Time".

Similarly, all other available formats are described in the following way: `<format type="DateTime" sample="tuesday 21 september 2004"> <template positive="dddd d mmmm yyyy"/> </format>`.

There can be several types of format available for each of "Number", "Currency", "DateTime", "Boolean", "Date", and "Time".

```
<formats>
  <format default="true" type="Number" sample="1 234,57">
    <template positive=""/>
  </format>
  <format default="true" type="Currency" sample="1 234,57 €; -1 234,57 €">
    <template positive="#,#0.00' €'"/>
  </format>
  <format default="true" type="DateTime" sample="21/09/2004 20:45:30">
    <template positive="dd '/' MM'/' yyyy HH ':' mm ':' ss"/>
  </format>
  <format default="true" type="Boolean" sample="vrai; faux">
    <template positive="BOOLEAN"/>
  </format>
  <format default="true" type="Date" sample="21/09/2004">
    <template positive="dd '/' MM'/' yyyy"/>
  </format>
  <format default="true" type="Time" sample="20:45:30">
    <template positive="HH ':' mm ':' ss"/>
  </format>
  <format type="Number" sample="1 234,57">
    <template positive="STANDARD"/>
  </format>
  <format type="Number" sample="1,234567E3">
    <template positive="SCIENTIFIC"/>
  </format>
  <format type="Number" sample="1235; (1235)">
    <template positive="0" negative="(0)"/>
  </format>
  <format type="Number" sample="1234,57">
    <template positive="0.00"/>
  </format>
  <format type="Number" sample="1 235">
    <template positive="#,#0"/>
  </format>
  <format type="Currency" sample="1 234,57 €; -1 234,57 €">
    <template positive="#,#0.00' €'"/>
  </format>
  <format type="Currency" sample="1 234,57 €; 1 234,57 €">
    <template positive="#,#0.00' €" negative="[Rouge]#,#0.00' €'"/>
  </format>
  <format type="DateTime" sample="mardi 21 septembre 2004">
    <template positive="dddd d mmmm yyyy"/>
  </format>
  <format type="DateTime" sample="21 septembre 2004">
    <template positive="d mmmm yyyy"/>
  </format>
  <format type="DateTime" sample="21/09/04">
    <template positive="dd '/' MM'/' yy"/>
  </format>
  <format type="DateTime" sample="20:45:30">
    <template positive="HH ':' mm ':' ss"/>
  </format>
  <format type="Boolean" sample="vrai; faux">
    <template positive="BOOLEAN"/>
  </format>
  <format type="Percent" sample="123 456,70%">
    <template positive="#,#0.00[%] '%'"/>
  </format>
</formats>
```

Body (XML)

```
</formats>
```

Example: To Get the formats and their default settings from the Web Intelligence server**Windows**

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/configuration/formats
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/configuration/formats
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 3055
```

```
<formats>
  <format default="true" type="Number" sample="1 234,57">
    <template positive=""/>
  </format>
  <format default="true" type="Currency" sample="1 234,57 €; -1 234,57 €">
    <template positive="#,#0.00' €'"/>
  </format>
  <format default="true" type="DateTime" sample="21/09/2004 20:45:30">
    <template positive="dd '/' 'MM' '/' 'yyyy HH ':' 'mm ':' 'ss'"/>
  </format>
  <format default="true" type="Boolean" sample="vrai; faux">
    <template positive="BOOLEAN"/>
  </format>
  <format default="true" type="Date" sample="21/09/2004">
    <template positive="dd '/' 'MM' '/' 'yyyy'"/>
  </format>
  <format default="true" type="Time" sample="20:45:30">
    <template positive="HH ':' 'mm ':' 'ss'"/>
  </format>
  <format type="Number" sample="1 234,57">
    <template positive="STANDARD"/>
  </format>
  <format type="Number" sample="1,234567E3">
    <template positive="SCIENTIFIC"/>
  </format>
  <format type="Number" sample="1235; (1235)">
    <template positive="0" negative="(0)"/>
  </format>
  <format type="Number" sample="1234,57">
    <template positive="0.00"/>
  </format>
  <format type="Number" sample="1 235">
    <template positive="#,#0"/>
  </format>
  <format type="Currency" sample="1 234,57 €; -1 234,57 €">
    <template positive="#,#0.00' €'"/>
  </format>
  <format type="Currency" sample="1 234,57 €; 1 234,57 €">
    <template positive="#,#0.00' €'" negative="[Rouge]#,#0.00' €'"/>
  </format>
  <format type="DateTime" sample="mardi 21 septembre 2004">
    <template positive="dddd d mmmm yyyy"/>
```

```

</format>
<format type="DateTime" sample="21 septembre 2004">
  <template positive="d mmm yy"/>
</format>
<format type="DateTime" sample="21/09/04">
  <template positive="dd/'MM'/'yy"/>
</format>
<format type="DateTime" sample="20:45:30">
  <template positive="HH': 'mm': 'ss"/>
</format>
<format type="Boolean" sample="vrai; faux">
  <template positive="BOOLEAN"/>
</format>
<format type="Percent" sample="123 456,70%">
  <template positive="#,##0.00[%] '%"/>
</format>
</formats>

```

Related Topics

- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.2 Getting the list of custom formats used for numbers in a document

Use this URL to get a list of the custom format numbers defined in a document.

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET
 http://<serverName>:6405/biprws/raylight/vx/documents

Getting a list of formats

Request:

GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/formats

Header	Value
Accept	application/xml or application/json

Header	Value
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Description of the custom formats used in the Web Intelligence document:</p> <p>: Document ID</p> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <formats> <format type="Custom" sample="1,235"> Name of format, sample number <template positive="#,##0" negative="[red]#,##0" zero="No value" undefined="NaN" > Number format and color where defined. In this example, negative numbers will be in red. </template> </format> </formats></pre>

Example: To retrieve the number formats for document ID 4326**Windows**

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/4326/formats
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/4326/formats
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<formats>
  <format type="Custom" sample="1,235">
    <template positive="#,#0" negative="[red]#,##0"
      zero="No value" undefined="NaN">
    </template>
  </format>
</formats>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.3 Getting all declared font mappings on a server

Get a list of all declared font mappings on a Web Intelligence server. There are three available platforms: HTML, Java, and Windows, with different font mappings. This call returns all mappings for all platforms.

Request

```
GET http://<serverName>:6405/biprws/raylight/vx/configuration/fontmappings
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Provides the list of all declared fontmappings on the Web Intelligence server.</p> <p><code><fontmappings></code> The tags enclosing the font mapping information</p> <p><code><mapping platform="HTML"> ...</mapping></code> Includes the fonts for the mapping platform "HTML", "JAVA" or "WINDOWS".</p> <p><code> ... </code> Encloses the returned font definition for a platform.</p> <p><code><serverName>Arial</serverName></code> The name of the font server</p> <p><code><platformName>Arial, Helvetica...</platformName></code> The available fonts. Fonts with more than one word as their name are enclosed in parentheses.</p>

Example: Getting the font mappings from the Web Intelligence server

The following example returns the font mappings for all three servers, plus one for the Japanese language server for the HTML platform.

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/configuration/fontmappings
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/configuration/fontmappings
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<fontmappings>
  <mapping platform="HTML">
    <font>
      <serverName>Arial</serverName>
      <platformName>Arial, Helvetica, 'Courier New', 'Times New Roman'</platformName>
    </font>
    <font>
      <serverName>BOJapan</serverName>
      <platformName>'MS Gothic', 'Arial Unicode MS'</platformName>
    </font>
  </mapping>

  <mapping platform="Java">
    <font>
      <serverName>Arial</serverName>
      <platformName>Arial, Helvetica, 'Courier New', 'Times New Roman'</platformName>
    </font>
  </mapping>

  <mapping platform="Windows">
    <font>
      <serverName>Arial</serverName>
      <platformName>Arial</platformName>
    </font>
    <font>
      <serverName>default</serverName>
      <platformName>Arial</platformName>
    </font>
  </mapping>
</fontmappings>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.4 Getting the list of all declared report skins

Get a list of all declared skins on a Web Intelligence Server. There are four available types of skin:

- Cell
- Block
- Section
- Report

This call returns all skins for all types.

Getting the details of the available skins

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/configuration/skins`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	<code>application/xml</code> or <code>application/json</code>
Content-Length	Length of content in the response body

Body (XML)

Details of the available skins identified by:

`<skins>` Contains the list of defined skins.

`<skin type="Cell">` The type of skin, identified by a unique name `<name>Skin_name</name>`. There are four types, there can be many skins of each type. The four types are:

- "Cell"
- "Block"
- "Section"
- "Report"

The background fill for the skin: a color, pattern, or a gif URL is defined as follows:

`<background>`

`<color rgb="#ffffff>`

`image src="boimg://skin_marble.gif" display="Tile"/>`

`</background>`

Example: To get the details of the available skins**Windows**

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/configuration/skins
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/configuration/skins
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<skins>
  <skin type="Cell">
    <name>Curve</name>
    <background> <color rgb="#ffffff"/> <image src="boimg://skin_marble.gif" display="Tile"/>
  </skin>
  <skin type="Cell">
    <name>Business Objects</name>
    <background> <color rgb="#ffffff"/> <image src="boimg://skin_bo.gif" display="Stretch"/>
  </skin>
```

```
<skin type="Block">
  <name>Business Objects</name>
  <background> <color rgb="#ffffff"/> <image src="boimg://skin_bo.gif" display="Stretch"/>
</background>
</skin>

<skin type="Section">
  <name>Business Objects</name>
  <background> <color rgb="#ffffff"/> <image src="boimg://skin_bo.gif" display="Stretch"/>
</background>
</skin>

<skin type="Section">
  <name>Dots</name>
  <background> <color rgb="#ffffff"/> <image src="boimg://skin_listing.gif" display="Tile"/>
</background>
</skin>

<skin type="Report">
  <name>Dots</name>
  <background> <color rgb="#ffffff"/> <image src="boimg://skin_listing.gif" display="Tile"/>
</background>
</skin>
</skins>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.5 Getting the list of styles in a document

Use this URL to:

- Get the list of all defined styles in a given document. (GET <url>/documents/{documentId}/styles).
- Add a new style to the document (POST <url>/documents/{documentId}/styles).

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Getting the defined styles of a Web Intelligence document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/styles`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
<code>documentId</code>	A valid ID identifier of a Web Intelligence document

Response:

Header	Value
Status Code	HTTP response code
Content-Type	<code>application/xml</code> or <code>application/json</code>
Content-Length	Length of content in the response body

Body (XML)

```

<styles>

<style>

<id>1</id> <background> <color rgb="#ffffff"/> </background>

<alignment horizontal="left" vertical="bottom"/>

</style>

<style>

<id>2</id> <parentId>1</parentId> <border> <top thickness="none"
rgb="#000000" style="none"/> <bottom thickness="thin" rgb="#000000"
style="plain"/> <left thickness="none" rgb="#000000" style="none"/> <right
thickness="none" rgb="#000000" style="none"/> </border> <font size="12"
face="Arial" italic="false" bold="true" strikethrough="false" under
line="false" rgb="#000000"/> <alignment horizontal="left" vertical="bot
tom"/>

</style>

<style>

<id>3</id> <parentId>1</parentId> <border> <top thickness="thin"
rgb="#cacad9" style="plain"/> <bottom thickness="thin" rgb="#cacad9"
style="plain"/> <left thickness="thin" rgb="#cacad9" style="plain"/> <right
thickness="thin" rgb="#cacad9" style="plain"/> </border> <background>
<color rgb="#5175b9"/> </background> <font size="9" face="Arial" ital
ic="false" bold="true" strikethrough="false" underline="false"
rgb="#ffffff"/> <alignment horizontal="left" vertical="bottom"/>

</style>

</styles>

```

Example: To get the defined styles of document ID 3422**Windows**

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken: ""%tokenValue"" http://<serverName>:6405/biprws/raylight/v1/documents/3422/styles
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:$logtok'
http://<serverName>:6405/biprws/raylight/v1/documents/3422/styles
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<styles>
  <style>
    <id>1</id>
    <background>
      <color rgb="#ffffff"/>
    </background>
    <alignment horizontal="Left" vertical="Bottom"/>
  </style>
  <style>
    <id>2</id>
    <parentId>1</parentId>
    <border>
      <top thickness="None" rgb="#000000" style="None"/>
      <bottom thickness="Thin" rgb="#000000" style="Plain"/>
      <left thickness="None" rgb="#000000" style="None"/>
      <right thickness="None" rgb="#000000" style="None"/>
    </border>
    <font size="12" face="Arial" italic="false" bold="true" strikethrough="false" underline="false"
rgb="#000000"/>
    <alignment horizontal="Left" vertical="Bottom"/>
  </style>
  <style>
    <id>3</id>
    <parentId>1</parentId>
    <border>
      <top thickness="Thin" rgb="#cacad9" style="Plain"/>
      <bottom thickness="Thin" rgb="#cacad9" style="Plain"/>
      <left thickness="Thin" rgb="#cacad9" style="Plain"/>
      <right thickness="Thin" rgb="#cacad9" style="Plain"/>
    </border>
    <background>
      <color rgb="#5175b9"/>
    </background>
    <font size="9" face="Arial" italic="false" bold="true" strikethrough="false" underline="false"
rgb="#ffffff"/>
    <alignment horizontal="Left" vertical="Bottom"/>
  </style>
</styles>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.6 Adding a new style to a document

Adding a new style to a Web Intelligence document

Define the new style in an xml file containing the style definition(s), for example backgroundstyle.xml.

Request:

POST `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/styles`

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Body (XML)
<p>An XML file containing the style definition(s), for example backgroundstyle.xml</p> <pre><style> <background width="100" height="50"/> </style></pre>

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>

<message>The resource of type 'Style' with identifier 'StyleId' has been
successfully created.</message>

<id>{StyleId}</id>

</success>
```

Example: To add a style to document ID 3422

Call the file backgroundstyle.xml to add the background style.

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""to
kenValue%" -d "@backgroundstyle.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/styles
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$log
tok" -d "@backgroundstyle.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/styles
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Style' with identifier '64' has been successfully created.</message>
  <id>64</id>
</success>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)

- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.7 Getting a style definition

Use this URL to:

- Get the detailed description of a style (GET <url>/documents/{documentId}/styles/{styleId})

Note:

{styleId}: The identifier of the Web Intelligence document's style retrieved in the document's styles list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/styles`

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Retrieves the details of a given Web Intelligence document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/styles/{styleId}`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
styleId	A valid ID identifier of the document style

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Details of the document styles identified by:</p> <pre> <style> <id>3</id> <parentId>58</parentId> <border> <top thickness="thin" rgb="#cacad9" style="plain"/> <bottom thickness="thin" rgb="#cacad9" style="plain"/> <left thickness="thin" rgb="#cacad9" style="plain"/> <right thickness="thin" rgb="#cacad9" style="plain"/> </border> <background> <color rgb="#5175b9"/> </background> <alignment horizontal="left" vertical="bottom"/> </style> </pre>

Example: To get the document style for document 5022

Windows

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H X-SAP-LogonToken:"""%tokenValue"""" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?styleId=cccTODO
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"'
http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?styleId=cccTODO
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<style>
  <id>3</id>
  <parentId>58</parentId>
  <border>
    <top thickness="thin" rgb="#cacad9" style="plain"/>
    <bottom thickness="thin" rgb="#cacad9" style="plain"/>
    <left thickness="thin" rgb="#cacad9" style="plain"/>
    <right thickness="thin" rgb="#cacad9" style="plain"/>
  </border>
  <background>
    <color rgb="#5175b9"/>
  </background>
  <font size="9" face="Arial" italic="false" bold="true" strikethrough="false" underline="false"
rgb="#ffffff"/>
  <alignment horizontal="left" vertical="bottom"/>
</style>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Deleting a document style](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.8 Deleting a document style

Use this URL to:

- Delete a style definition (`DELETE <url>/documents/{documentId}/styles/{styleId}`)

Deleting a document style

Request:

DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/styles/{styleId}

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
styleId	A valid ID identifier of the document style

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Style' with identifier '{StyleId} ' has been successfully deleted.</message> <id>{StyleId}</id> </success></pre>

Example: To delete the style L4 for document 5022**Windows**

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?L4
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken: "$logtok"" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?L4
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Style' with identifier 'L4' has been successfully removed.</message>
  <id>L4</id>
</success>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Updating a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.9 Updating a document style

Use this URL to:

- Update a style definition (PUT <url>/documents/{documentId}/styles/{styleId})

Updating a document style

Declare the styleId that you want the document to use.

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/styles/{styleId}`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
styleId	A valid ID identifier of the document style

Body (XML)

An xml file containing the definition of the document styles identified by:

```
<style>
<id>3</id>
<parentId>58</parentId>
<border>
<top thickness="thin" rgb="#cacad9" style="plain"/>
<bottom thickness="thin" rgb="#cacad9" style="plain"/>
<left thickness="thin" rgb="#cacad9" style="plain"/>
<right thickness="thin" rgb="#cacad9" style="plain"/>
</border>
<background>
<color rgb="#5175b9"/> </background>
<font size="9" face="Arial" italic="false" bold="true" strikethrough="false"
underline="false" rgb="#ffffff"/>
<alignment horizontal="left" vertical="bottom"/>
</style>
```

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>

<message>The resource of type 'Style' with identifier '{StyleId} ' has
been successfully updated.</message>

<id>{StyleId}</id>

</success>
```

Example: To update the style (style Id 24) for document 5022

Write an xml file containing the style definition, in gthis example, style3.xml:

```
<style>
  <id>3</id>
  <parentId>58</parentId>
  <border>
    <top thickness="thin" rgb="#cacad9" style="plain"/>
    <bottom thickness="thin" rgb="#cacad9" style="plain"/>
    <left thickness="thin" rgb="#cacad9" style="plain"/>
    <right thickness="thin" rgb="#cacad9" style="plain"/>
  </border>
  <background>
    <color rgb="#5175b9"/>
  </background>
  <font size="9" face="Arial" italic="false" bold="true" strikethrough="false" underline="false"
rgb="#ffffff"/>
  <alignment horizontal="left" vertical="bottom"/>
</style>
```

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to
kenValue%" -d "@style.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?24
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$logtok"
-d "@style.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?24
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Style' with identifier '24' has been successfully updated.</message>
  <id>24</id>
</success>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [CSS: getting or updating the CSS for a document](#)

3.2.6.10 CSS: getting or updating the CSS for a document

Use this URL to:

- Get the CSS of the document (GET <url>/documents/{documentId}/css)
- Update the CSS of the document (PUT <url>/documents/{documentId}/css)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Getting the CSS for a given document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/css`

Header	Value
Accept	text/css
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Response:

The XML or Json flow that contains the document CSS.

Header	Value
Status Code	HTTP response code
Content-Type	text/css
Content-Length	Length of content in the response body

Body (XML)
Details of the CSS for the document:

Example: To get the CSS for document ID 451

Windows

```
curl -G -i -H "accept:text/css" -H X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/451/css
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:text/css" 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/451/css
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

FORM CELL.ia-form-header { /* Settings for header cells in a form */
    background-fill:color;
    font-size:9pt;
    color:#1D7DB3;
    background-color:#f8fbfc;
    font-weight-bold:yes;
    never-alternate:yes;
}

CELL.ia-form-separator { /* Settings for cells separating two form instances */
    background-fill:none; /* Invisible cell making the separation */
    border-top-style:none;
    border-right-style:none;
    border-bottom-style:none;
    border-left-style:none;
    border-top-width:0;
    border-right-width:0;
```

```
border-bottom-width:0;  
border-left-width:0;  
}
```

Updating the CSS for a given document

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/css`

Header	Value
Accept	application/xml
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Body (XML)
The CSS file.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	text/css
Content-Length	Length of content in the response body

Body

In case of success:

```
<success>
```

```
<message>The resource of type 'CSS' has been successfully updated.</message>
```

```
<id>11416</id>
```

```
<success>
```

Otherwise: `<error> ... </error>`

Example: To update the CSS for document 5022**Windows**

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/css" -H "X-SAP-LogonToken:""%tokenValue%""
-d "@css2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/css
```

UNIX**Note:**

Uses the `logtok` variable set at login.

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/css" -H "X-SAP-LogonToken:""$logtok"
-d "@css2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/css
```

Request body described in file `css2.xml`.

```
FORM CELL.ia-form-header { /* Settings for header cells in a form */
    background-fill:color;
    font-size:9pt;
    color:#1D7DB3;
    background-color:#f8fbfc;
    font-weight-bold:yes;
    never-alternate:yes;
}

CELL.ia-form-separator { /* Settings for cells separating two form instances */
    background-fill:none; /* Invisible cell making the separation */
    border-top-style:none;
    border-right-style:none;
    border-bottom-style:none;
    border-left-style:none;
    border-top-width:0;
    border-right-width:0;
    border-bottom-width:0;
    border-left-width:0;
}
```


Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Document' with identifier '5022' has been successfully updated.</message>
  <id="5022"></id>
</success>
```

Related Topics

- [Getting the list of declared configuration formats](#)
- [Getting the list of custom formats used for numbers in a document](#)
- [Getting all declared font mappings on a server](#)
- [Getting the list of all declared report skins](#)
- [Getting the list of styles in a document](#)
- [Adding a new style to a document](#)
- [Getting a style definition](#)
- [Deleting a document style](#)
- [Updating a document style](#)

3.2.6.11 Getting the list of supported charsets

Get a list of all supported charsets on a Web Intelligence Server. Charset is used as a parameter when exporting a document as CSV format.

This call returns the list of supported charsets.

Getting the supported charsets

Request URL

GET http://<serverName>:6405/biprws/raylight/vx/configuration/charsets

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Response Body (XML)

Details of the supported charsets. Each charset is identified by:

<charset> Contains the description of a supported charset.

<name>EUC-KR</name> The charset name.

<description>Korean (EUC-KR)</description> The charset description

Example: To get the details of the supported charsets

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/configuration/charsets
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: '$logtok'" http://<serverName>:6405/biprws/raylight/v1/configuration/charsets
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<charsets>
  <charset>
    <name>EUC-KR</name>
    <description>Korean (EUC-KR)</description>
  </charset>
  <charset>
    <name>ISO-8859-8</name>
    <description>Hebrew (ISO-8859-8)</description>
  </charset>
  <charset>
    <name>ISO-8859-5</name>
    <description>Cyrillic (ISO-8859-5)</description>
  </charset>
</charsets>
```

```
</charset>
...
< charset>
  < name>UTF-8</name>
  < description>UTF-8</description>
</ charset>
< charset>
  < name>HZ-GB-2312</name>
  < description>Chinese Simplified (HZ-GB2312)</description>
</ charset>
< charset>
  < name>CNS-11643</name>
  < description>Chinese Traditional (EUC-TW)</description>
</ charset>
< charset>
  < name>CP437</name>
  < description>IBM Latin US (CP437)</description>
</ charset>
</ charsets>
```

3.2.7 Managing functions, operators, and variables

This section describes the different methods you can use for managing formula engine functions and operators, and managing variables.

- Get the list of available formula engine functions (GET <url>/configuration/functions)
- Get the list of available formula engine operators (GET <url>/configuration/operators)
- Get the content of a documents variables dictionary (GET <url>/documents/{documentId}/variables)
- Add a new expression to a documents variables dictionary (POST <url>/documents/{documentId}/variables)
- Get the definition of a variable from a documents variable dictionary (GET <url>/documents/{documentId}/variables/{variableId})
- Modify the definition of an variable from a documents variable dictionary (PUT <url>/documents/{documentId}/variables/{variableId})
- Delete a variable from a documents variable dictionary (DELETE <url>/documents/{documentId}/variables/{variableId})

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of available formula engine operators](#)
- [Getting the list of variables](#)
- [Adding a variable definition](#)
- [Getting the definition of a variable](#)
- [Modifying a variable definition](#)
- [Deleting a variable definition](#)

3.2.7.1 Getting the list of available formula engine functions

Gets all functions of the available formula engine. This can be used to create formulas in the Report Specification or define variables in the document dictionary.

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/configuration/functions`

Notes

Header	Value
Accept	To retrieve the details of a document: application/xml
X-SAP-LogonToken	The logon token value, in quotation marks

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or
Content-Length	Length of content in the response body

Body (XML)

Details of the query specification attached to a given data provider, described as follows:

```
<functions> Contains the list of the functions and their descriptions.
<function category="Logical" returnType="Boolean"> Function type
<id>EVEN</id> Identifier of the function
<name>Even</name>
<description></description> Description of the function
<syntax>bool Even(number)</syntax>
```

Example: To get all function descriptions

The example below shows how to get all the function descriptions, the example of returned body contains just two examples.

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<server
Name>:6405/biprws/raylight/v1/configuration/functions
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/ray
light/v1/configuration/functions
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<functions>
  <function category="Logical" returnType="Boolean">
    <id>EVEN</id>
    <name>Even</name>
    <description>Determines whether a number is even</description>
    <syntax>bool Even(number)</syntax>
  </function>
  <function category="Document" returnType="String">
    <id>DOCUMENTAUTHOR</id>
    <name>DocumentAuthor</name>
    <description>Returns the InfoView logon of the document creator</description>
    <syntax>string DocumentAuthor()</syntax>
  </function>
</functions>
```

Related Topics

- [Getting the list of available formula engine operators](#)
- [Getting the list of variables](#)
- [Adding a variable definition](#)
- [Getting the definition of a variable](#)
- [Modifying a variable definition](#)
- [Deleting a variable definition](#)

3.2.7.2 Getting the list of available formula engine operators

Gets all operators of the formula engine. This can be used to create formulas in the Report Specification or define variables in the document dictionary.

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/configuration/operators`

Notes

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Details of the document operators available from the formula engine. Each operator is described as follows:

```
<operators> Contains the list of operator descriptions.

<operator> The start of the description for an operator.

<id>LINEAR</id> The ID of the operator.

<name> Linear</name> The name of the operator.

<description>...</description> The description of the operator.

<syntax>Linear</syntax> The syntax of the operator.

</operator>

...

...

</operators>
```

Example: To get all operator descriptions

The example below shows how to get all the operator descriptions, the example of returned body contains just two examples.

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/configuration/operators
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: "$logtok"" http://<serverName>:6405/biprws/raylight/v1/configuration/operators
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<operators>
  <operator>
    <id>LINEAR</id>
    <name>Linear</name>
    <description>Tells the Interpolation function to use linear regression to supply missing measure val
```

```

ues</description>
  <syntax>Linear</syntax>
</operator>
<operator>
  <id>INLIST</id>
  <name>InList</name>
  <description>Determines if a value is in a list</description>
  <syntax>bool test_value Inlist(value_list)</syntax>
</operator>
</operators>

```

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of variables](#)
- [Adding a variable definition](#)
- [Getting the definition of a variable](#)
- [Modifying a variable definition](#)
- [Deleting a variable definition](#)

3.2.7.3 Getting the list of variables

Use this report to:

- Get the content of a documents variables dictionary (GET <url>/documents/{documentId}/variables)

Getting the content of a documents variables dictionary

Request:

GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/variables

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Description of the variables used by the document. Each variable is described as follows:</p> <p><variable dataType="Numeric" qualification="Measure"> The type of data (numeric, string, xxx, and the qualification (measure, dimension, ss)</p> <p><id>L6</id> The identifier of the variable</p> <p><name>Min Revenue</name> The name of the variable, as used in the document</p>

Example: Retrieves the definitions of the variables used by document ID 4326

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/4326/formats
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/4326/formats
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
```

```

Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<variables>
  <variable dataType="Numeric" qualification="Measure">
    <id>L6</id>
    <name>Min Revenue</name>
  </variable>
  <variable dataType="Numeric" qualification="Measure">
    <id>L7</id>
    <name>RevenueThreshold</name>
  </variable>
  <variable dataType="Numeric" qualification="Measure">
    <id>L8</id>
    <name>Threshold factor</name>
  </variable>
  <variable dataType="Numeric" qualification="Measure">
    <id>L9</id>
    <name>Threshold Max</name>
  </variable>
  <variable dataType="Numeric" qualification="Measure">
    <id>LA</id>
    <name>Threshold Min</name>
  </variable>
</variables>

```

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of available formula engine operators](#)
- [Adding a variable definition](#)
- [Getting the definition of a variable](#)
- [Modifying a variable definition](#)
- [Deleting a variable definition](#)

3.2.7.4 Adding a variable definition

Adding a new expression to the documents variables dictionary

You define the expression in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; variable2.xml.

Note:

- The formula must be valid.
- You can only create a measure, an attribute or a dimension.
- When you create an attribute, the associated dimension is mandatory.

Request:

```
POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/variables
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Body (XML)
<p>An XML file containing the definition of the variable to be used by the document. Each variable is described as follows:</p> <pre> <variable qualification="Measure"> The type of variable <name>new variable</name> The name of the variable <definition>=[RevenueThreshold]*[Threshold factor]</definition> the variable definition </variable> </pre>

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
```

```
<message>The resource of type 'Variable' with identifier '{LB}' has been  
successfully created.</message>
```

```
<id>{LB}</id>
```

```
</success>
```

Otherwise: <error> ... </error>

Example: Adding a new variable definition to document ID 4326**Windows**

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" -d "@variable2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/4326/variables
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "POST" -H "X-SAP-LogonToken:$logtok" -d "@variable2.xml" http://<serverName>:6405/biprws/ray
light/v1/documents/4326/variables
```

Response

```
HTTP/1.1 200 OK
Server:Apache-Coyote/1.1
Date: Mon, 04 Jun 2012 11:59:02 GMT
Content-Type: application/xml
Content-Length: 204

<success>
<message>The resource of type 'Variable' with identifier 'LB' has been successfully created.</message>
<id>LB</id>
</success>
```

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of available formula engine operators](#)
- [Getting the list of variables](#)
- [Getting the definition of a variable](#)
- [Modifying a variable definition](#)
- [Deleting a variable definition](#)

3.2.7.5 Getting the definition of a variable

Use this to:

- Get the definition of a variable from a documents variable dictionary (GET <url>/documents/{documentId}/variables/{variableId})

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{variableId}: The identifier of the Web Intelligence variable retrieved in the document's variable list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/documentId/variables`

Getting the definition of a variable used by a document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/variables/{variableId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.
variableId	Mandatory. The ID of the variable used by the declared document.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Definition of the variable, described as follows:</p> <pre><variable dataType="Numeric" qualification="Measure"> The type and qualification of the variable <id>L9</id> The identifier of the variable <name>Threshold Max</name> The name of the variable, as used in the document <description></description> Description of the variable (optional), can be used to describe how the variable is used. <formulaLanguageId>[Threshold Max]</formulaLanguageId> <definition>=[RevenueThreshold]*(1+[Threshold factor])</definition> The formula used by the variable. </variable></pre>

Example: To get the definition of variable 'L9'

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8022/variables/?variableId=L9
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: "$logtok"" http://<serverName>:6405/biprws/raylight/v1/documents/8022/variables/?variableId=L9
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<variable dataType="Numeric" qualification="Measure">
  <id>L9</id>
  <name>Threshold Max</name>
  <description></description>
  <formulaLanguageId>[Threshold Max]</formulaLanguageId>
  <definition>=[RevenueThreshold] * (1+[Threshold factor])</definition>
</variable>
```

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of available formula engine operators](#)
- [Getting the list of variables](#)
- [Adding a variable definition](#)
- [Modifying a variable definition](#)
- [Deleting a variable definition](#)

3.2.7.6 Modifying a variable definition

Modifying the definition of a variable used by a document

Use this to:

- Modify the definition of an variable from a documents variable dictionary (PUT `<url>/documents/{documentId}/variables/{variableId}`)

You can change the following aspects of a variable: the variable qualification, its name, the formula.

Note:

The modified formula must be valid.

Note:

When you change the definition of the variable, you must refresh the document to commit your change to the report.

Request:

```
PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/variables/{variableId}
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks.

Parameter	Description
documentId	Mandatory. The ID of the document.
variableId	Mandatory. The ID of the variable used by the declared document.

Body (XML)
<p>The new definition of the variable in an XML file, described as follows:</p> <pre><variable qualification="Measure"> <name>Updated variable</name> <definition>=[RevenueThreshold]*[Threshold factor]/10</definition> </variable></pre>

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>

<message>The resource of type 'Variable' with identifier '{variableId} '
has been successfully updated.</message>

<id>{VariableId}</id>

</success>
```

Example: To modify a variable

You define the modified variable in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; variable3.xml.

Body (XML) "variable3.xml"

The new definition of the variable in an XML file, described as follows:

```
<variable qualification="Measure">
  <name>Updated variable</name>
  <definition>=[RevenueThreshold]*[Threshold factor]/10</definition>
</variable>
```

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to
kenValue%"" -d "@variable3.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8022/variables/?vari
ableId=L9
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$logtok"
-d "@variable3.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8022/variables/?variableId=L9
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
```

```
<message>The resource of type 'Variable' with identifier 'L9' has been successfully updated.</message>
<id>L9</id>
</success>
```

Note:

You must refresh the document in order to see the changes.

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of available formula engine operators](#)
- [Getting the list of variables](#)
- [Adding a variable definition](#)
- [Getting the definition of a variable](#)
- [Deleting a variable definition](#)

3.2.7.7 Deleting a variable definition**Deleting variable from a documents variables dictionary**

Use this to:

- Delete a variable from a documents variable dictionary (DELETE <url>/documents/{documentId}/variables/{variableId})

You can delete a variable from the variables dictionary of a document. Ensure that the variable is no longer used by the queries in the document.

Request:

```
DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/variables/{variableId}
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Parameter	Description
variableId	Mandatory. The ID of the variable that you want to delete.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Variable' with identifier '{variableId} ' has been successfully deleted.</message> <id>{variableId}</id> </success></pre>

Example: To delete a variable (L9) from a variables dictionary

Windows

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/8022/variables/?variableId=L9
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok"" http://<serverName>:6405/biprws/raylight/v1/documents/8022/variables/?variableId=L9
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Variable' with identifier 'L9' has been successfully deleted.</message>
  <id>L9</id>
</success>
```

Related Topics

- [Getting the list of available formula engine functions](#)
- [Getting the list of available formula engine operators](#)
- [Getting the list of variables](#)
- [Adding a variable definition](#)
- [Getting the definition of a variable](#)
- [Modifying a variable definition](#)

3.2.8 Managing alerters and tracking changes

This section describes the different methods you can use for managing alerters and the track data changes functions.

Alters

You use the alerters to track the changes in selected data and to configure the display of changed data. Use the formula language to build your custom alerters for formatting data changes. You can include special calculations based on data changes. For example, you can include a calculation to show the difference between the previous value and the current value of a measure.

Use this URL to:

- Get the list of all alerters defined in a document (GET <url>/documents/{documentId}/alerters)
- Add a new alerter to a document (POST <url>/documents/{documentId}/alerters)
- Get the full description of the specified alerter (GET <url>/documents/{documentId}/alerters/{alerterId}).
- Update the description of a specified alerter (PUT <url>/documents/{documentId}/alerters/{alerterId}).
- Delete a specified alerter (DELETE <url>/documents/{documentId}/alerters/{alerterId})

Note:

`{alerterId}`: The identifier of the Web Intelligence document's alerter retrieved in the document's alerters list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters`

Note:

- `{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`
- You must provide Alerter action description when creating a new alerter. An empty alerter will result in an error.

Trackdata

When you track data changes, you select a particular data refresh as a reference point. This data is known as the reference data. When you activate data tracking, you see your data in relation to the reference data.

Formatting to track changes either automatically or manually. You can track the following types of data change:

- Inserted data
- Deleted data
- Changed data
- Increased values
- Decreased values

In automatic data tracking mode, you always compare the current data with the data before the last refresh. This is achieved by automatically setting the current data as the reference data just before each refresh. The reference data is always one refresh behind the current data. Automatic data tracking is appropriate for scheduled documents when you want to compare the current data with the data before the last refresh.

In manual data tracking mode, you select the reference data. You continue to use this data as a reference point until you update the reference point.

Use this to:

- Get the tracker information (GET `<url>/documents/{documentId}/tracker`)
- Create the tracker resource (POST `<url>/documents/{documentId}/tracker`)
- Update the track data information (PUT `<url>/documents/{documentId}/tracker`)
- Delete the tracker resource (DELETE `<url>/documents/{documentId}/tracker`)

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)

- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.1 Getting a list of alerters

Use this URL to:

- Get the list of all alerters defined in a document (GET <url>/documents/{documentId}/alerters)
- Add a new alerter to a document (POST <url>/documents/{documentId}/alerters)

Note:

{alerterId}: The identifier of the Web Intelligence document's alerter retrieved in the document's alerters list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters`

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Getting the list of all alerters defined in a specified document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>A list of the document alerters, each alerter is identified by:</p> <p><id>{alerterId}</id>: the ID of the alerter.</p> <p><name>{alerterName}</name>: the name of the alerter.</p> <p><description>...</description>: The description of the alerter (if present).</p>

Example: To get the list of alerters for document ID 1223

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<alerters>
  <alerter>
    <id>1</id>
    <name>Sales Revenue</name>
    <description>Test Raylight</description>
  </alerter>
  <alerter>
    <id>2</id>
    <name>Profit Margin</name>
    <description>Test Raylight2</description>
  </alerter>
</alerters>
```

Related Topics

- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.2 Getting the description of an alerter

Use this URL to:

- Get the full description of the specified alerter (GET <url>/document/{documentId}/alerters/{alerterId}).
- Update the description of a specified alerter (PUT <url>/document/{documentId}/alerters/{alerterId}).
- Delete a specified alerter (DELETE <url>/document/{documentId}/alerters/{alerterId})

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{alerterId}: The identifier of the Web Intelligence document alerter, retrieved in the document's alerters list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters`

Getting the description of an alerter

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters/{alerterId}`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
alerterId	A valid ID identifier of an alerter for the document

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Details of the document alerter identified by:</p> <pre> <id>{alerterId}</id>: the ID of the alerter. <name>{alerterName}</name>: the name of the alerter. <description>...</description>: The description of the alerter (if present). <rule>...</rule>: The definition of the rule <action>...</action>: The definition of the affect on the document formatting.</pre>

Example: To get the description of alerter 3 for document 8022**Windows**

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8022/alerters/?alerterId=3
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8022/alerters/?alerterId=3
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 720

<alerter>
  <id>1</id>
  <name>Sales Revenue</name>
  <description>Test Raylight</description>
  <rule>
    <conditions>
      <condition expressionId="DP0.D093" operator="Greater">
        <operand>2000000</operand>
      </condition>
    </conditions>
    <action>
      <data>
        <formula type="HyperLink">test</formula>
        <format type="Custom" sample="1á234,57">
          <template positive="STANDARD"/>
        </format>
      </data>
      <style>
        <background>
          <color rgb="#ffff00"/>
        </background>
        <font rgb="#ff0000"/>
      </style>
    </action>
  </rule>
</alerter>
```

Related Topics

- [Getting a list of alerters](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.3 Updating the description of an alerter

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters/{alerterId}`

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
<code>documentId</code>	A valid ID identifier of a Web Intelligence document
<code>alerterId</code>	A valid ID identifier of an alerter for the document

Body

The description of the alerter in an XML file, for example, define a file called alerter4.xml:

```
<alerter>
  <id>1</id>
  <name>Sales Revenue</name>
  <description>Test Raylight</description>
  <rule>
    <conditions>
      <condition expressionId="DP0.D093" operator="Greater">
        <operand>2000000</operand>
      </condition>
    </conditions>
    <action>
      <data>
        <formula type="HyperLink">test</formula>
        <format type="Custom" sample="1â234,57">
          <template positive="STANDARD"/>
        </format>
      </data>
      <style>
        <background>
          <color rgb="#ffff00"/>
        </background>
        <font rgb="#ff0000"/>
      </style>
    </action>
  </rule>
</alerter>
```

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>

<message>The resource of type 'Alerter' with identifier '{alerterId}' has
been successfully updated.</message>

<id>{alerterId}</id>

</success>
```

Example: To update the description of a document alerter (Document ID=5022 , alerter ID=2)**Windows**

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:" "%tokenValue%" -d "@alerter4.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/alerters/?alerterId=2
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:" "$logtok" -d "@alerter4.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/alerters/?alerterId=2
```

Body:

An xml file 'alerter4.xml', called in the PUT method.

```
<alerter>
  <name>My alerter updated</name>
  <description>This is my new alerter updated</description>
  <rule>
    <expression>="goodbye";</expression>
  </rule>
</alerter>
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Alerter' with identifier '2' has been successfully updated.</message>
  <id="2"></id>
</success>
```

Example: 2: XML body file update a complex alerter

The body xml file below defines a complex alerter.

Body:

An xml file 'complexalerter.xml', called in the PUT method. The "PUT" call will reference this xml file.

```
<alerter>
  <name>Sample 2</name>
  <description>Add a complexe alerter using Raylight</description>
  <rule>
    <conditions>
      <condition expressionId="DP0.DO93" operator="Greater">
        <operand>10</operand>
        <operand>20</operand>
      </condition>
    </conditions>
  </rule>
</alerter>
```

```

    <condition expressionId="DP0.D093" operator="Greater">
      <target expressionId="DP0.D093" />
    </condition>
  </conditions>
  <action>
    <data>
      <formula type="HyperLink">my hyperlink</formula>
      <format type="Custom" sample="1 234,57">
        <template positive="STANDARD"/>
      </format>
    </data>
    <style>
      <border>
        <top thickness="Medium" rgb="#800000" style="Plain"/>
        <bottom thickness="Medium" rgb="#800000" style="Plain"/>
        <left thickness="Medium" rgb="#800000" style="Plain"/>
        <right thickness="Medium" rgb="#800000" style="Plain"/>
      </border>
      <background width="0" height="0">
        <color rgb="#00ff00"/>
        <image src="bores://00002">
          <alignment horizontal="Left" vertical="Top"/>
        </image>
      </background>
      <font size="12" face="Arial" italic="false" bold="false" strikethrough="true" underline="true"
rgb="#ff6600"/>
      <alignment horizontal="Center" vertical="Center"/>
    </style>
  </action>
</rule>
<rule>
  <expression>=&quot;hello&quot;;</expression>
</rule>
</alerter>

```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.4 Adding an alerter to a document

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Body (XML)	Description
<pre><alerter> <name>My alerter</name> <description>This is my new alerter</description> <rule> <expression>="hel lo";</expression> <action> <style> <background> <color rgb="#ffff00"/> </background> </style> </action> </rule></pre>	<p>The body is defined in an XML file and called in the request. It describes the name of the alerter, an optional description, and the SQL expression.</p> <p>Note: You must provide Alerter action description when creating a new alerter.</p>
Empty body	<p>Note: To save a <code>Modified</code> document do not use any body in your request.</p>

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Alerter' with identifier '{alerterId}' has been successfully created.</message> <id>{alerterId}</id> </success></pre>

Example: To add a new alerter to document ID 1223**Windows**

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%"" -d "@newalerter.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "POST" -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@newalerter.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
<message>The resource of type 'Alerter' with identifier '2' has been successfully created.</message>
  <id>2</id>
</success>
```


Example: To add a complex alerter to document ID 1223**Body (XML) complexalerter.xml**

In case of success:

```
<name>ciomplexalerter</name>
<description>Adda complexe alerter using Raylight</description>
<rule>
  <conditions>
    <condition expressionId="DP0.DO93" operator="Greater">
      <operand>10</operand>
      <operand>20</operand>
    </condition>
    <condition expressionId="DP0.DO93" operator="Greater">
      <target expressionId="DP0.DO93" />
    </condition>
  </conditions>
  <action>
    <data>
      <formula type="HyperLink">my hyperlink</formula>
      <format type="Custom" sample="1 234,57">
        <template positive="STANDARD"/>
      </format>
    </data>
    <style>
      <border>
        <top thickness="Medium" rgb="#800000" style="Plain"/>
        <bottom thickness="Medium" rgb="#800000" style="Plain"/>
        <left thickness="Medium" rgb="#800000" style="Plain"/>
        <right thickness="Medium" rgb="#800000" style="Plain"/>
      </border>
      <background width="0" height="0">
        <color rgb="#00ff00"/>
        <image src="bores://00002">
          <alignment horizontal="Left" vertical="Top"/>
        </image>
      </background>
      <font size="12" face="Arial" italic="false" bold="false" strikethrough="true" underline="true"
rgb="#ff6600"/>
      <alignment horizontal="Center" vertical="Center"/>
    </style>
  </action>
</rule>
<rule>
  <expression>=&quot;hello&quot;;</expression>
</rule>
</alerter>
```

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" -d "@complexalerter.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -i -X "POST" -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@complexalerter.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
<message>The resource of type 'Alerter' with identifier '2' has been successfully created.</message>
  <id>2</id>
</success>
```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.5 Deleting an alerter

Deleting an alerter

Request:

```
DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/alerters/{alerterId}
```

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
alerterId	A valid ID identifier of an alerter for the document

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Alerter' with identifier '{alerterId}' has been successfully deleted.</message> <id>{alerterId}</id> </success></pre>

Example: To delete alerter ID=3 of document 5022**Windows**

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken: ""tokenValue"" " http://<serverName>:6405/biprws/raylight/v1/documents/5022/alerters/3
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken: $logtok" http://<serverName>:6405/biprws/raylight/v1/documents/5022/alerters/3
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Alerter' with identifier '3' has been successfully removed.</message>
  <id>3</id>
</success>
```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.6 Getting the tracker settings for a document

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET
`http://<serverName>:6405/biprws/raylight/vx/documents`

Getting the definition of the trackdata settings used by a document

Obtains the current trackdata definition for a document. Returns an error if the trackdata function is not activated on this document.

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/tracker`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Definition of the trackdata information, described as follows:

```
<tracker mode="Manual" referenceDate="2012-11-16T10:56:21.951+01:00">
<added active="true"> <background/> <style italic="false" bold="true" strikethrough="false" under-
line="false" rgb="#0000aa"/>
</added>
<changed active="true"> <background rgb="#aabbcc"/> <style italic="false" bold="true"
strikethrough="false" underline="false" rgb="#0000bb"/> </changed>
<removed active="true"> <background/> <style italic="false" bold="true" strikethrough="false" under-
line="false" rgb="#0000cc"/> </removed>
<increasing threshold="5.0" applyThreshold="false" active="true"> <background rgb="#33cc33"/>
<style italic="false" bold="true" strikethrough="false" underline="false"/> </increasing>
<decreasing threshold="8.0" applyThreshold="true" active="false"> <background rgb="#33cc33"/>
<style italic="true" bold="false" strikethrough="true" underline="true"/></decreasing></tracker>
```

Where :

mode: (type=enumeration,("Auto","Manual")) specifies the mode of Track Data changes

Auto: Display changes are based on comparison with last data refresh

Manual: Display changes are based on comparison with data refresh from user reference

Threshold : (type=double) specifies the display changes threshold in percent for numeric data

Message returned in case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Tracker' does not exist.</message>
</error>
```

Example: To get the trackdata definition for document ID 8022**Windows**

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<server
Name>:6405/biprws/raylight/v1/documents/8022/tracker
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:$logtok' http://<serverName>:6405/biprws/raylight/v1/documents/8022/tracker
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<tracker mode="Manual" referenceDate="2012-11-16T10:56:21.951+01:00">
  <added active="true">
    <background/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000aa"/>
  </added>
  <changed active="true">
    <background rgb="#aabbcc"/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000bb"/>
  </changed>
  <removed active="true">
    <background/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000cc"/>
  </removed>
  <increasing threshold="5.0" applyThreshold="false" active="true">
    <background rgb="#33cc33"/>
    <style italic="false" bold="true" strikethrough="false" underline="false"/>
  </increasing>
  <decreasing threshold="8.0" applyThreshold="true" active="false">
    <background rgb="#33cc33"/>
    <style italic="true" bold="false" strikethrough="true" underline="true"/>
  </decreasing>
</tracker>
```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.7 Updating the trackdata setting

Updating the definition of the trackdata settings for a document

Update a trackdata function on a document. It updates the style applied on alerters. It is also possible to enable/disable the visualization of a change type. If not provided the track data definition, the mode is set to "Manual" and the document is set as reference.

The request is optional. If not provided the trackdata will be activated with default settings or the previous settings if trackdata has been activated previously.

Throws an error if resource is not created.

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/tracker`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Body (XML)

Define the trackdata settings that you want to update from the following possible properties:

```
<tracker mode="Manual">
  <added active="true">
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000aa"/>
  </added>
  <changed active="true">
    <background rgb="#aabbcc"/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000bb"/>
  </changed>
  <removed active="true">
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000cc"/>
  </removed>
  <increasing threshold="5.0" applyThreshold="false" active="true">
    <background rgb="#33cc33"/>
    <style italic="false" bold="true" strikethrough="false" underline="false"/>
  </increasing>
  <decreasing threshold="8.0" applyThreshold="true" active="false">
    <background rgb="#33cc33"/>
    <style italic="true" bold="false" strikethrough="true" underline="true"/>
  </decreasing>
</tracker>
```

Response:

The XML or Json flow that contains the success message or an error message.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json

Header	Value
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
```

```
<message>The resource of type 'Tracker' has been successfully updated for
document.</message>
```

```
<id>{documentId}</id>
```

```
</success>
```

Example: To update a trackdata setting for a document

Write the xml file for the body of the request. In this example, the file is called tracker2.xml and is called by -d "@tracker2.xml" in the request.

Body file (XML) "tracker2.xml"

```
<tracker mode="Manual" referenceDate="2012-11-16T10:56:21.951+01:00">
  <added active="true">
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000aa"/>
  </added>
  <changed active="true">
    <background rgb="#aabbcc"/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000bb"/>
  </changed>
  <removed active="true">
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000cc"/>
  </removed>
  <increasing threshold="5.0" applyThreshold="false" active="true">
    <background rgb="#33cc33"/>
    <style italic="false" bold="true" strikethrough="false" underline="false"/>
  </increasing>
  <decreasing threshold="8.0" applyThreshold="true" active="false">
    <background rgb="#33cc33"/>
    <style italic="true" bold="false" strikethrough="true" underline="true"/>
  </decreasing>
</tracker>
```

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@tracker2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/8022/tracker
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:$logtok' -d "@tracker2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/8022/tracker
```

Response:

Body (XML)

```
<success>
  <message> The resource of type 'Tracker' has been
  successfully updated for the document.</message>
  <id>{documentId}</id>
</success>
```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Creating or enabling the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.8 Creating or enabling the trackdata setting

Creating the definition of the trackdata settings for a document

Activate the trackdata function on a document.

The request is optional. If not provided the trackdata will be activated with default settings or the previous settings if TDC has been activated earlier.

Throw an error if resource is not created.

Request:

```
POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/track
er
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Body (XML)
<p>Definition of the trackdata settings, the default settings are as follows:</p> <pre> <tracker mode="Manual" referenceDate="2012-11-16T10:56:21.951+01:00"> <added active="true"> <background/> <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000aa"/> </added> <changed active="true"> <background rgb="#aabbcc"/> <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000bb"/> </changed> <removed active="true"> <background/> <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000cc"/> </removed> <increasing threshold="5.0" applyThreshold="false" active="true"> <back ground rgb="#33cc33"/> <style italic="false" bold="true" strikethrough="false" underline="false"/> </increasing> <decreasing threshold="8.0" applyThreshold="true" active="false"> <back ground rgb="#33cc33"/> <style italic="true" bold="false" strikethrough="true" underline="true"/></decreasing></tracker> </pre> <p>Where :</p> <p>mode: (type=enumeration,("Auto","Manual")) specifies the mode of Track Data changes</p> <p>Auto: Display changes are based on comparison with last data refresh</p> <p>Manual: Display changes are based on comparison with data refresh from user reference</p> <p>Threshold : (type=double) specifies the display changes threshold in percent for numeric data</p>

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Tracker' has been successfully created for document.</message> <id>{TrackerId}</id> </success></pre>

Message returned in case of error: HTTP Response Code: 404
<pre><error> <error_code>401</error_code> <message>The resource of type 'Tracker' is already created.</message> </error></pre>

Example: To create a trackdata setting for a document

The track data definition is provided in an xml file which is called by the -d "@{filename}.xml" in the cURL command. This file is the body of the request. In this example, the file is called tracker1.xml and is called by -d "@tracker1.xml" in the request.

Body file (XML) "tracker1.xml"

```
<tracker mode="Manual" referenceDate="2012-11-16T10:56:21.951+01:00">
  <added active="true">
    <background/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000aa"/>
  </added>
  <changed active="true">
    <background rgb="#aabbcc"/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000bb"/>
  </changed>
  <removed active="true">
    <background/>
    <style italic="false" bold="true" strikethrough="false" underline="false" rgb="#0000cc"/>
  </removed>
  <increasing threshold="5.0" applyThreshold="false" active="true">
    <background rgb="#33cc33"/>
    <style italic="false" bold="true" strikethrough="false" underline="false"/>
  </increasing>
  <decreasing threshold="8.0" applyThreshold="true" active="false">
    <background rgb="#33cc33"/>
    <style italic="true" bold="false" strikethrough="true" underline="true"/>
  </decreasing>
</tracker>
```

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" -d "@tracker1.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/8022/tracker
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "POST" -H "accept:application/xml" -H 'X-SAP-LogonToken:$logtok' -d "@tracker1.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/8022/tracker
```

Response:**Body (XML)**

```
<success>
  <message> The resource of type 'Tracker' has been
  successfully created for the document.</message>
  <id>{TrackerId}</id>
</success>
```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Disabling the trackdata setting](#)

3.2.8.9 Disabling the trackdata setting**Disabling the trackdata setting for a document**

To disable the trackdata setting, you delete the trackdata function on the document.

Throw an error if resource is not deleted.

Request:

```
DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/tracker
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.

Response:

The XML or Json flow that contains the result of the request.

Header	Value
Status Code	HTTP response code

Header	Value
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
<message>The resource of type 'Tracker' has been successfully removed.</message>
<id>{variableId}</id>
</success>
```

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Tracker' does not exist.</message>
</error>
```

Example: To disable the trackdata setting for a document**Windows**

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to
kenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/5022/tracker
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$log
tok" http://<serverName>:6405/biprws/raylight/v1/documents/5022/tracker
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
```



```
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<success>
  <message>The resource of type 'Tracker' with identifier '5022' has been successfully removed.</message>
  <id="5022"></id>
</success>
```

Related Topics

- [Getting a list of alerters](#)
- [Getting the description of an alerter](#)
- [Updating the description of an alerter](#)
- [Adding an alerter to a document](#)
- [Deleting an alerter](#)
- [Getting the tracker settings for a document](#)
- [Updating the trackdata setting](#)
- [Creating or enabling the trackdata setting](#)

3.2.9 Managing attachments and links

This section describes the different methods you can use for managing document attachments and links.

- Get the list of attachments to a document (GET <url>/documents/{documentId}/attachments/)
- Add an attachment to a document (POST <url>/documents/{documentId}/attachments/)
- Get the content of a documents links dictionary (GET <url>/documents/{documentId}/links)
- Add a new link to the documents expressions dictionary (POST <url>/documents/{documentId}/links)
- Retrieve the description of a link of a document (GET <url>/documents/{documentId}/links/{linkId}).
- Modify a link of a document (PUT <url>/documents/{documentId}/links/{linkId}).
- Remove a link from the document (DELETE <url>/documents/{documentId}/links/{linkId}).

Related Topics

- [Getting the list of document attachments](#)
- [Adding an attachment to a document](#)
- [Getting the links for a document](#)
- [Adding a link to a document](#)
- [Getting the document link definition](#)
- [Modifying a link in a document](#)
- [Deleting a document link](#)

3.2.9.1 Getting the list of document attachments

Use this URL to:

- Get the list of attachments to a document (GET <url>/documents/{documentId}/attachments/)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Request URL

Getting the list of attachments of a given Web Intelligence document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/attachments`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Response:

Header	Value
Status Code	HTTP response code
Content-Type	<code>application/xml</code> or <code>application/json</code>
Content-Length	Length of content in the response body

Body (XML)

Details of the attachments of the document. Each attachment is identified by:

The attachment mime type <contentType>

Document name <name>

Document description (if one exists, otherwise the tag is empty) <description>

Example: To get the list of attachments to document ID 8022**Windows**

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%"tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8022/attachments
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022/attachments
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<attachments>
  <attachment>
    <name>barometer.png</name>
    <size>13229</size>
    <releasemode>auto</releasemode>
    <md5hashcode>97B3E3B2745595A2CBA42CA825CCD656</md5hashcode>
    <mimeType>image/png</mimeType>
  </attachment>
  <attachment>
    <name>statistics.jpg</name>
    <size>18239</size>
    <releasemode>auto</releasemode>
    <md5hashcode>97B3E3B2745595A2CBA42CA825CCD656</md5hashcode>
    <mimeType>image/jpeg</mimeType>
  </attachment>
</attachments>
```

Related Topics

- [Adding an attachment to a document](#)
- [Getting the links for a document](#)
- [Adding a link to a document](#)
- [Getting the document link definition](#)
- [Modifying a link in a document](#)
- [Deleting a document link](#)

3.2.9.2 Adding an attachment to a document

Use this URL to:

- Add an attachment to a document (POST <url>/documents/{documentId}/attachments/)

Adding an attachment to a Web Intelligence document

You define the attachment in the body which is defined in an .xml file (for example, for example, attachment.xml) saved in the current path (usually the same path as the Curl tool). The multipart body request allows at least two parts: the information about the attachment (`attachmentInfos`) and the attachment content (`attachmentContent`).

In case of image upload, the mime types accepted are `image/png`, `image/jpeg` and `image/gif`.

Request:

POST `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/attachments`

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Header	Value
Request type	multipart/form-data
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Body (XML)

```
<attachment>
<name>sales_report</name>
<mimeType>image/png</mimeType>
<size>123</size>
</attachment>
```

```
-----###boundary123456798
Content-Disposition: form-data; name=""attachmentInfos""
Content-Type: application/xml

<attachment>
  <name>logo_picture</name>
  <mimeType>image/png</mimeType> <!--Optional-->
  <size>123</size>
</attachment>

-----###boundary123456798
Content-Disposition: form-data; name=""attachmentContent""; filename=""logo_picture.png""
Content-Type: image/png

(Content of file)

-----###boundary123456798
```

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
<message>The attachment resource has been successfully created.</message>
```

Example: To add an attachment to document ID 3422

You define the name of the attachment in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; attachment.xml.

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:multipart/form-data" -H "X-SAP-LogonToken:" "%tokenValue%" -d "@attachment.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/attachments
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:multipart/form-data" -H "X-SAP-LogonToken:" "$logtok" -d "@attachment.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/attachments
```

Request body

```
<attachment>
  <mimeType>image/png</mimeType>
  <name> myResourceName </name>
  <description> myResourceDescription </description>
</attachment>
```

Response

```
HTTP/1.1 200 OK
Server:Apache-Coyote/1.1
Date: Mon, 04 Jun 2012 11:59:02 GMT
Content-Type: application/xml
Content-Length: 204

<success>
<message>The resource of type "Attachment" has been successfully created.</message>
</success>
```

Related Topics

- [Getting the list of document attachments](#)
- [Getting the links for a document](#)
- [Adding a link to a document](#)
- [Getting the document link definition](#)
- [Modifying a link in a document](#)
- [Deleting a document link](#)

3.2.9.3 Getting the links for a document

Use this URL to:

- Get the content of a documents links dictionary (GET <url>/documents/{documentId}/links)
- Add a new link to the documents expressions dictionary (POST <url>/documents/{documentId}/links)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Getting the contents of a documents links dictionary**Request:**

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/links`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Details of the document links identified by:

```
<links>
<link dataType="String" qualification="Dimension">
<id>LB</id>
<name>Category</name>
</link>
<link dataType="String" qualification="Dimension">
<id>L8</id>
<name>City</name>
</link>
<link dataType="String" qualification="Dimension">
<id>LA</id>
<name>Lines</name>
</link>
<link dataType="String" qualification="Dimension">
<id>L7</id>
<name>State</name>
</link>
<link dataType="String" qualification="Dimension">
<id>L9</id>
<name>Store name</name>
</link>
<link dataType="String" qualification="Dimension">
<id>L6</id>
<name>Year</name>
</link>
</links>
```


Example: To get the details of the links of document ID 8022

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8022/links
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022/links
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<links>
  <link dataType="String" qualification="Dimension">
    <id>LB</id>
    <name>Category</name>
  </link>
  <link dataType="String" qualification="Dimension">
    <id>L8</id>
    <name>City</name>
  </link>
  <link dataType="String" qualification="Dimension">
    <id>LA</id>
    <name>Lines</name>
  </link>
  <link dataType="String" qualification="Dimension">
    <id>L7</id>
    <name>State</name>
  </link>
  <link dataType="String" qualification="Dimension">
    <id>L9</id>
    <name>Store name</name>
  </link>
  <link dataType="String" qualification="Dimension">
    <id>L6</id>
    <name>Year</name>
  </link>
</links>
```

Related Topics

- [Getting the list of document attachments](#)
- [Adding an attachment to a document](#)
- [Adding a link to a document](#)
- [Getting the document link definition](#)
- [Modifying a link in a document](#)
- [Deleting a document link](#)

3.2.9.4 Adding a link to a document

Use this URL to:

- Add a new link to the documents expressions dictionary (POST <url>/documents/{documentId}/links)

Adding a link to a Web Intelligence document

You define the link in the body which is defined in an .xml file saved in the current path (usually the same path as the Curl tool). For example; link1.xml.

In order to create/update a link, the following conditions must be respected:

- You can only link dimensions, attributes and hierarchies.
- The expressions to link must have the same data type.
- Each expression that you link to must be from a different data provider.
- The expressions that you link to must not be already used in another link.
- You cannot link to expressions that have been stripped (no query stripping allowed).
- The expressions must be compatible.

About compatibility:

- All custom formulas and variables will be considered as compatible to any other expressions and then ignored.
- Measures are always compatible with everything.
- If dimension A is linked, then A is compatible with all expressions in the flow lists that contain the link.
- Two dimensions or attributes from different flow lists are incompatible. Except when a dimension is linked, as mentioned above.

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/links

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Body (XML)
<p>...Description of the link</p> <pre><link> <name>new link</name> <linkedExpressions> <linkedExpression id="DP0.DObc"/> <linkedExpression id="DP1.DObc"/> </linkedExpressions> </link></pre>

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Link' has been successfully created.</mes sage></pre>

Example: To add a new link to document ID 3422

You define the link in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; link1.xml.

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" -d "@link1.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/attachments
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok"" -d "@link1.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/attachments
```

Request body

```
<link>
  <name>new link</name>
  <linkedExpressions>
    <linkedExpression id="DP0.DObc"/>
    <linkedExpression id="DP1.DObc"/>
  </linkedExpressions>
</link>
```

Response

```
HTTP/1.1 200 OK
Server:Apache-Coyote/1.1
Date: Mon, 04 Jun 2012 11:59:02 GMT
Content-Type: application/xml
Content-Length: 204

<success>
<message>The resource of type 'Link' with identifier 'L6' has been successfully created.</message>
</success>
```

Related Topics

- [Getting the list of document attachments](#)
- [Adding an attachment to a document](#)
- [Getting the links for a document](#)
- [Getting the document link definition](#)
- [Modifying a link in a document](#)
- [Deleting a document link](#)

3.2.9.5 Getting the document link definition

You can:

- Retrieve the description of a link of a document (GET <url>/documents/{documentId}/links/{linkId}).

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the document list by:
GET http://<serverName>:6405/biprws/raylight/vx/documents

Note:

{linkId}: The identifier of the Web Intelligence link is retrieved in the document's links list by: GET
http://<serverName>:6405/biprws/raylight/vx/documents/links

Getting the description of a link

Request:

GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/links/{linkId}

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
linkId	A valid ID identifier of a Web Intelligence document link

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

```

<link dataType="String" qualification="Dimension">
  <id>L6</id>
  <name>Year</name>
  <description>Year 2003 - 2006.</description>
  <dataSourceObjectId>DS0.DObc</dataSourceObjectId>
  <formulaLanguageId>[Year]</formulaLanguageId>
  <linkedExpressions>
    <linkedExpression id="DP0.DObc"/>
    <linkedExpression id="DP1.DObc"/>
  </linkedExpressions>
</link>

```

Example: To get the description of a link**Note:**

The document ID is 7738

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/7738/links/linkIdxxxx
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/7738/links/linkIdxxxx
```

Response

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 07 Jun 2012 08:23:37 GMT
Content-Type: application/xml
Content-Length: 217

<link dataType="String" qualification="Dimension">
  <id>L6</id>
  <name>Year</name>
  <description>Year 2003 - 2006.</description>
  <dataSourceObjectId>DS0.DObc</dataSourceObjectId>
  <formulaLanguageId>[Year]</formulaLanguageId>
  <linkedExpressions>
    <linkedExpression id="DP0.DObc"/>
    <linkedExpression id="DP1.DObc"/>
  </linkedExpressions>
</link>

```

```
</linkedExpressions>
</link>
```

Related Topics

- [Getting the list of document attachments](#)
- [Adding an attachment to a document](#)
- [Getting the links for a document](#)
- [Adding a link to a document](#)
- [Modifying a link in a document](#)
- [Deleting a document link](#)

3.2.9.6 Modifying a link in a document

You can:

- Modify a link of a document (PUT `<url>/documents/{documentId}/links/{linkId}`).

In order to modify a link, the following conditions must be respected:

- You can only link dimensions, attributes and hierarchies.
- The expressions to link must have the same data type.
- Each expression that you link to must be from a different data provider.
- The expressions that you link to must not be already used in another link.
- You cannot link to expressions that have been stripped (no query stripping allowed).
- The expressions must be compatible.

About compatibility:

- All custom formulas and variables will be considered as compatible to any other expressions and then ignored.
- Measures are always compatible with everything.
- If dimension A is linked, then A is compatible with all expressions in the flow lists that contain the link.
- Two dimensions or attributes from different flow lists are incompatible. Except when a dimension is linked, as mentioned above.

Modifying a link definition

Request:

```
PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/links/{linkId}
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
linkId	A valid ID identifier of a Web Intelligence document link

Body (XML)	Description
<pre><link> <name> new link</name> <description>A new link</description> <linkedExpressions> <linkedExpression id= "DP0.DObc" /> <linkedExpression id= "DP1.DObc" /> </linkedExpressions> </link></pre>	Mandatory.

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

```
<success> <message> Success. the resource of type 'Link' with identifier 'L9' has been successfully updated. </message> </success>
```

Example: To modify a link

You define thelink in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; link2.xml.

Request:

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%token Value%" -d "@link2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/links/L2
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$logtok" -d "@link2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/links/L2
```

Request body

```
<link dataType="String" qualification="Dimension">
  <id>L6</id>
  <name>Year</name>
  <description>Year 2003 - 2006.</description>
  <dataSourceObjectId>DS0.DObc</dataSourceObjectId>
  <formulaLanguageId>[Year]</formulaLanguageId>
  <linkedExpressions>
    <linkedExpression id="DP0.DObc"/>
    <linkedExpression id="DP1.DObc"/>
  </linkedExpressions>
</link>
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<success>
  <message>The resource of type 'Link' with identifier 'L2' has been successfully updated.</message>
  <id="67"></id>
</success>
```

Related Topics

- [Getting the list of document attachments](#)
- [Adding an attachment to a document](#)
- [Getting the links for a document](#)
- [Adding a link to a document](#)
- [Getting the document link definition](#)
- [Deleting a document link](#)

3.2.9.7 Deleting a document link

You can:

- Remove a link from the document (DELETE <url>/documents/{documentId}/links/{linkId}).

Deleting a link

You can delete a document referenced by its ID.

Request:

DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/links/{linkId}

Note:

{documentId}: The identifier of the Web Intelligence document to delete, can be retrieved from the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document
linkId	A valid ID identifier of a Web Intelligence document link

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body
In case of success, "The resource of type 'Link' with identifier {linkId} has been successfully removed". linkId Otherwise <error> ... </error>

Example: To delete link L4 of document ID 8022**Windows**

```
curl -X "DELETE" -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8022/links/L4
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -X "DELETE" -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022/links/L4
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Link' with identifier 'L4' has been successfully removed.</message>
  <id="L4"></id>
</success>
```

Related Topics

- [Getting the list of document attachments](#)
- [Adding an attachment to a document](#)
- [Getting the links for a document](#)

- [Adding a link to a document](#)
- [Getting the document link definition](#)
- [Modifying a link in a document](#)

3.2.10 Adding a cache entry to a document

Use this to add a cache entry to a document.

Add a new schedule to a document (POST <url>/documents/{documentId}/cache).

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the document list by:
GET http://<serverName>:6405/biprws/raylight/vx/documents.

Define the new style in an xml file containing the style definition(s), for example backgroundstyle.xml.

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/cache

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID identifier of a Web Intelligence document

Body (XML)

An XML file containing the cache entry definition(s), for example cache1.xml

```
<cache>
  <output mimeType="application/pdf">
    <locale value="fr_FR" />
    <locale value="de_DE" />
  </output>
  <output mimeType="application/pdf">
    <locale value="en" />
  </output>
</cache>
```

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
<message>The resource of type 'Cache' with identifier '{documentId}' has
been successfully updated.</message>
<id>{documentId}</id>
</success>
```

Example: To add a cache entry to document ID 3422

Call the file cache1.xml to add the background style.

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:" "%to
kenValue%" -d "@cache1.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/cache
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@cache1.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/cache
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Cache' has been successfully updated.</message>
  <id>3422</id>
</success>
```

3.3 Managing the document lifecycle (LCM)

The following are the main operations available to manage Web Intelligence life cycle documents. This feature is related to the state of a document which is retrieved by the following resource:

```
GET http://<serverName>:6405/biprws/raylight/vx/documents
```

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

3.3.1 Document state: managing the state of the document (Life Cycle Management)

This URL is used to manage Web Intelligence document life cycle. It is used to change the state of a document in order to manage memory in the Web Intelligence RESTful web service container. You can close a document referenced by its ID, or save it, and/or discard all changes previously made to it. You can move a document from a Modified or Original to Unused state in order to discard all changes made to the document and close the document.

The state of a document is provided by the GET method. The state can be: `Unused`, `Original`, or `Modified`. This state corresponds to the state of a document in the Raylight container.

- An `Unused` document is a document that has not been loaded in the Raylight container.

- An `Original` document is a document that has been loaded in the Raylight container but has not been modified. Its state can be changed to `Unused` to free the Raylight memory, closing the document.
- A `Modified` document is a document that has been loaded and modified through Raylight. Its state can be changed to `Unused`, it is closed to free the Raylight memory.

Note:

As soon as you open a document its status becomes `Original`.

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Request

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}`

Header	Value
Accept	application/xml or application/json

Header	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Body (XML)	Description																				
<pre><document> <state>Unused</state> </document></pre>	<p>Optional.</p> <p>Used to change the state of an <code>Original</code> or a <code>Modified</code> document.</p> <p>An <code>Original</code> document changed to <code>Unused</code> is closed.</p> <p>A <code>Modified</code> document changed to <code>Unused</code> is closed.</p> <p>Note:</p> <p>To save a <code>Modified</code> document do not use any body in your request.</p> <table><tr><th>Original document state</th><th>Current state = Unused</th><th>Current state = Original</th><th>Current state = Modified</th></tr><tr><td>Unused</td><td>Do nothing / resource not modified</td><td>Close doc / resource not modified</td><td>Close doc / resource updated</td></tr><tr><td>Original</td><td>N/A</td><td>N/A</td><td>N/A</td></tr><tr><td>Modified</td><td>N/A</td><td>N/A</td><td>N/A</td></tr><tr><td>Empty (new document)</td><td>Do nothing / resource not modified</td><td>Close doc / resource not modified</td><td>Save doc / resource updated</td></tr></table>	Original document state	Current state = Unused	Current state = Original	Current state = Modified	Unused	Do nothing / resource not modified	Close doc / resource not modified	Close doc / resource updated	Original	N/A	N/A	N/A	Modified	N/A	N/A	N/A	Empty (new document)	Do nothing / resource not modified	Close doc / resource not modified	Save doc / resource updated
Original document state	Current state = Unused	Current state = Original	Current state = Modified																		
Unused	Do nothing / resource not modified	Close doc / resource not modified	Close doc / resource updated																		
Original	N/A	N/A	N/A																		
Modified	N/A	N/A	N/A																		
Empty (new document)	Do nothing / resource not modified	Close doc / resource not modified	Save doc / resource updated																		
Empty body	<p>Important: a <code>Modified</code> document can be saved only with a request that has no request body or an empty body.</p> <p>A request on an <code>Original</code> document with no body or an empty body, close the <code>Original</code> document.</p>																				

Response:

Header	Value
Status Code	HTTP response code

Header	Value
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Message:</p> <pre><success> <message>The resource of type 'document' with identifier '9326' has been successfully updated.< /message> <id>9326</id></success></pre>

Example: To close an unmodified document

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:" "%to
kenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8009
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:" "$logtok"
http://<servername>:6405/biprws/raylight/v1/documents/8009
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 21 Aug 2012 08:35:08 GMT
Content-Type: application/xml
Content-Length: 196
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<success>
<message>The resource of type 'document' with identifier '8008' has not been modified.</message>
  <id>8008</id>
</success>
```

Example: To close an Original or a Modified document

Close and free the memory of the web service container.

Warning: modifications made on the Modified document are lost.

Note:

The Unused state is provided using the UnusedState.xml file called by -d "@UnusedState.xml" in cURL command.

UnusedState.xml details

```
<document>
  <state>Unused</state>
</document>
```

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -d "@UnusedState.xml" -H X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/12192
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -d "@UnusedState.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/12192
```

Example: To save a Modified document

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/12192
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/12192
```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [Getting the document refresh parameters before refreshing a document](#)
- [Document: getting the details, updating, or deleting](#)

3.3.2 Snapshots: getting a list of snapshots or creating a snapshot for a document

Use this to:

- Get the list of snapshots of a document (GET <url>/documents/{documentId}/snapshots)
- Create a snapshot of a document (POST <url>/documents/{documentId}/snapshots)

A snapshot is identified by a snapshot ID token. Document state management are managed in REBean by Storage Token. A document can have the following states:

- Open -> (State 1)
- Set Prompts -> (State 2)
- Refresh -> (State 3)
- Output-> (State 4)

These states are affected by each change of the document (not only in the case of a prompt workflow like above). This means for example

- Drill workflows
- Turn to
- Report filter manipulation
- Fold/Unfold ?

Storage token can be used in scheduling workflows.

You can customize one step and go back and forth between initial state and resulting state. This can be used as undo/redo or implement custom prompt dialog boxes

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

A document creator can create snapshot (corresponding to a wid with a storage token) and do the following:

Getting existing snapshot ID tokens

Use the following call

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/snapshots`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. A valid ID identifier of a Web Intelligence document

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server
Date	Date and time of the response
Content Type	application/xml or application/json

Body
<p>In case of success, the list of snapshot IDs for the document:</p> <pre><snapshots maxStackSize="10"><snapshot id="token1" /><snapshot id="token2" /> <snapshot id="token3" /></snapshots></pre> <pre><snapshots maxStackSize="10"> <snapshot id="token1"/> <snapshot id="token2"/> <snapshot id="token3"/> </snapshots></pre> <p>Otherwise: <error> ... </error></p>

Example: To get the ID tokens of document ID 8022

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%" http://<server Name>:6405/biprws/raylight/v1/documents/8022/snapshots
```

UNIX

Note:

Uses the logtok variable set at login

```
./curl.exe -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server Name>:6405/biprws/raylight/v1/documents/8022/snapshots
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
```

Content-Length: 355

```
<snapshots maxStackSize="10">
  <snapshot id="we00000000d447acb133a0" />
  <snapshot id="we00000000e5df6062ca2a" />
  <snapshot id="we00010000dc73921d7b5d" />
</snapshots>
```

To create a new storage token

The call returns the ID of the newly created storage token.

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/snapshots

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. A valid ID identifier of a Web Intelligence document

Response:

Header	Value
Status Code	HTTP response code
Server	Type of server
Date	Date and time of the response
Content Type	application/xml or application/json

Body

In case of success, the new snapshot ID for the document:

```
<snapshot id="tokenID"/>
```

Otherwise: <error> ... </error>

Example: To create a snapshot (and hence ID token) for document ID 8022**Windows**

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8022/snapshots
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:""$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022/snapshots
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<snapshots>
  <snapshot id="we00000000d447acb133a0" />
</snapshots>
```

3.3.3 Snapshots: restoring a document to a specific snapshot

Restore the document to the state corresponding to the given snapshot.

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{snapshotId}: The identifier of the Web Intelligence document snapshot retrieved in the document's snapshot list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/snapshots`

Request

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}?snapshotId={token}`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Body (XML)
Details of the document identified by: <documentId> The identifier of the document. <token> The identifier of the snapshot.

Response:

Header	Value
Status Code	HTTP response code
Content-Type	<code>application/xml</code> or <code>application/json</code>
Content-Length	Length of content in the response body

Body (XML)
In case of success: <success> <message>The resource of type 'Document' with identifier '{documentId}' has been successfully updated.</message> <id>{documentId}</id> </success>

Example: To restore document 5022 to snapshot ID we00000000e5df6062ca2a**Windows**

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to  
kenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/5022?snapshotId=we00000000e5df6062ca2a
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$logtok"  
http://<serverName>:6405/biprws/raylight/v1/documents/5022?snapshotId=we00000000e5df6062ca2a
```

Response

```
HTTP/1.1 200 OK  
Server: Apache-Coyote/1.1  
Date: Tue, 05 Jun 2012 08:10:36 GMT  
Content-Type: application/xml  
Content-Length: 355  
  
<success>  
  <message>The resource of type 'Document' with identifier '5022' has been successfully updated.</message>  
  <id="5022"></id>  
</success>
```

3.4 Managing reports

This section describes the main operations available on reports:

- Exporting reports
- Managing reports
- Managing report drillers
- Managing report elements
- Managing report structure

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

Restrictions

The following workflows are not delivered or supported:

- Retrieving & setting fold/unfold state of a document report
- Retrieving & setting collapse/expand state of a hierarchy displayed in a document report
- Management of Input Controls for a document report
- Text search in a document or report

Related Topics

- [Managing documents](#)
- [Getting, moving, or creating, copying a report](#)
- [Getting report details and deleting reports](#)

3.4.1 Getting, moving, or creating, copying a report

Use this to:

- Get all the reports of the document specified by the `documentId` URL's parameter. (GET `<url>/documents/{documentId}/reports`)
- Move a report in a document (PUT `<url>/documents/{documentId}/reports`)
- Create a Web Intelligence report in a given document specified in the URL parameter (POST `<url>/documents/{documentId}/reports`).

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`. See the link referenced below.

Getting a Web Intelligence report

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
<code>documentId</code>	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. <code>documentId</code> : (type=integer, minimum=1, maximum=highest document number in existing documents).

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
Provides per report: the id, the name, the reference, and the showDataChanges setting

Example: To get reports of document 12782

Note:

Retrieves reports of the document ID=12782.

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/12782/reports
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: '$logtok'" http://<serverName>:6405/biprws/raylight/v1/documents/12782/reports
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544
```

```
<reports>
  <report>
    <id>1</id>
    <name>Cross Tab View</name>
    <reference>1.RS</reference>
    <element>
      <name>2004</name>
      <reference>1.K.0</reference>
    </element>
    <element>
      <name>2005</name>
      <reference>1.K.1</reference>
    </element>
    <element>
      <name>2006</name>
      <reference>1.K.2</reference>
    </element>
  </report>
</reports>
```

```

<id>2</id>
<name>Simple charting view</name>
<reference>2.RS</reference>
<element>
  <name>2004</name>
  <reference>2.K.0</reference>
</element>
<element>
  <name>2005</name>
  <reference>2.K.1</reference>
</element>
<element>
  <name>2006</name>
  <reference>2.K.2</reference>
</element>
</report>
</reports>

```

Moving a report within a document

Move a report to a different position in the document.

Request:

PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports?fromId=<fromId>&toId=<toId>

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The ID of the document.
fromId	Mandatory. Integer. A valid identifier of a report to move
toId	Mandatory. Integer. A valid identifier of a report (position of the report after the move)

Response:

Header	Value
Status Code	HTTP response code.

Header	Value
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body
<p>In case of success:</p> <pre><success> <message>The reource of type 'Report' with identifier '1' has been success fully moved.</message> <id>2</id> <success> Otherwise: <error> ... </error></pre>

Example: To move a report from position 1 to position 3 in document 12782

Note:

Retrieves reports of the document ID=12782.

Windows

```
curl -PUT -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<server
Name>:6405/biprws/raylight/v1/documents/12782/reports?fromId=1&toId=3
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -PUT -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/ray
light/v1/documents/12782/reports?fromId=1&toId=3
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Report' with identifier '1' has been successfully moved.</message>
  <id>1</id>
</success>
```

Creating a Web Intelligence report

Creates a report in a given document specified in the URL parameter.

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory: A valid ID identifier of a Web Intelligence document

Body (XML)
<p>Optional:</p> <pre><report> <name>New report</name> The name of the report </report></pre>

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body

In case of success:

```
<success>
```

```
<message>The resource of type 'Report' with identifier '2' has been created successfully.</message>
```

```
<id>2</id>
```

```
<success>
```

Otherwise: `<error> ... </error>`

Example: To create a report for document 12782

Write an xml file describing the report, for example report2.xml:

```
<report>
  <name>New report</name>
</report>
```

Note:

Creates a new report in the document ID=12782.

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" -d "@report2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/12782/reports
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok"" -d "@report2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/12782/reports
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Report' with identifier '2' has been successfully created.</message>
  <id="2"></id>
</success>
```

Copying a Web Intelligence report

Copies a report in a given document specified in the URL parameter.

Request:

POST `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory: A valid ID identifier of a Web Intelligence document
fromId	Mandatory: a valid identifier of a report to copy.
keepDriller	Preserve the driller mode in the copied report (applicable only if the source report is in driller mode). Values: true/false, default is true

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body

In case of success:

```
<success>
```

```
<message>The resource of type 'Report' with identifier '2' has been successfully created.</message>
```

```
<id>2</id>
```

```
<success>
```

Otherwise: `<error> ... </error>`

Example: To copy a report for document 12782

Write an xml file describing the report, for example report2.xml:

Note:

Creates a new report in the document ID=12782.

Windows

```
curl -POST -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/14308/reports?fromId=1&keepDriller=false
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -POST -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/14308/reports?fromId=1&keepDriller=false
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<success>
  <message>The resource of type 'Report' with identifier '2' has been successfully created.</message>
  <id="2"></id>
</success>
```

Related Topics

- [Document: getting the details, updating, or deleting](#)

3.4.2 Getting report details and deleting reports

Use this to:

- Get the details of the report specified by the `reportId` parameter. (GET `<url>documents/{documentId}/reports/{reportId}`)
- Delete the report specified by the `reportId` parameter. (DELETE `<url>documents/{documentId}/reports/{reportId}`)

Note:

`{documentId}`: The identifier of the Web Intelligence document is retrieved in the document list by:
GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

`{reportId}`: The identifier of the report retrieved by:GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Retrieves details of a report

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
<code>documentId</code>	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. <code>documentId</code> : (type=integer, minimum=1, maximum=highest document number in existing documents).
<code>reportId</code>	Mandatory. Integer. The identifier of the report.

Response

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
Details of the report specified by reportId URL parameter: the id, the name, the reference and showDataChanges.

Example: To get the details of the report

Note:

Retrieve details of the report (ID 3) from the document (ID 7858)

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""%tokenValue%\"" http://<serverName>:6405/biprws/raylight/v1/documents/7858/reports/3
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/7858/reports/3
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 14:24:12 GMT
Content-Type: application/xml
Content-Length: 167
```

```
<report>
  <id>1</id>
  <name>Cross Tab View</name>
  <reference>1.RS</reference>
  <showDataChanges>false</showDataChanges>
  <element>
    <name>2004</name>
    <reference>1.K.0</reference>
  </element>
  <element>
    <name>2005</name>
    <reference>1.K.1</reference>
  </element>
  <element>
    <name>2006</name>
    <reference>1.K.2</reference>
  </element>
</report>
```

```

    </element>
  </report>

```

Deleting a specified report

Use this to delete a specified report in a document. You cannot undo this operation.

Request:

DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report.

Response

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

```
<success>The resource of type 'report' with identifier '9372' has been
successfully removed. <id>9372</id></success>
```

Example: To delete report 67 of document ID 9512**Windows**

```
curl -X "DELETE" -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" http://<server
Name>:6405/biprws/raylight/v1/documents/9512/reports/67
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -X "DELETE" -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server
Name>:6405/biprws/raylight/v1/documents/9512/reports/67
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Report' with identifier '67' has been successfully deleted.</message>
  <id="67"></id>
</success>
```

Update the properties of a specified report

Use this to delete a specified report in a document. You cannot undo this operation.

Request:

```
PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/re
ports/{reportId}
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory: A valid ID identifier of a Web Intelligence document

Parameter	Description
ReportId	Mandatory: A valid ID identifier of a Web Intelligence report

Request body (XML)
<pre><report> <showDataChanges>true</showDataChanges> showDataChanges: show("true","false") track- data on the report (tracker) </report></pre>

Response

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre><success>The resource of type 'report' with identifier '9372' has been successfully removed. <id>9372</id></success></pre>

Example: To update report 67 of document ID 9512

Windows

```
curl -PUT -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" -d @"update.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -PUT -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d @"update.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67
```

```
<report>
  <showDataChanges>true</showDataChanges>
</report>
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Report' with identifier '67' has been successfully deleted.</message>
  <id="67"></id>
</success>
```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [Document: getting the details, updating, or deleting](#)
- [Getting, moving, or creating, copying a report](#)
- [To log on to the BI platform](#)

3.4.3 Get the map of a report

Get the map of a report in a Web Intelligence document.

Request

```
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/map
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).
reference	Optional. A valid reference of a map node. For example: reference=1.G.1

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <map> <node reference="1.G.0" name="2004"/> <node reference="1.G.1" name="2005"/> <node reference="1.G.2" name="2006"/> </map></pre>

Example: To get the map for report 1 of document 18809

This example shows how to retrieve the full map references for a report. There are eight map nodes in the body of the response

Windows

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken: ""%tokenValue"" http://<serverName>:6405/biprws/raylight/v1/documents/18809/reports/1/map
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"'
http://<serverName>:6405/biprws/raylight/v1/documents/18809/reports/1/map
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 175
```

```
<map>
  <node reference="1.G.0" name="2004"/>
  <node reference="1.G.1" name="2005"/>
  <node reference="1.G.2" name="2006"/>
</map>
```

Example: To get the map for a node of a report

The following examples shows how to get the map reference for a specific node of a report. Once you have obtained the node references by using the previous example, you include the parameter `?reference=<node reference>`. Again, the document is ID 18809, and gteh report is ID 1.

Windows

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H X-SAP-LogonToken:""%tokenValue%"
http://<serverName>:6405/biprws/raylight/v1/documents/18809/reports/1/map?reference=1.G.1
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"'
http://<serverName>:6405/biprws/raylight/v1/documents/18809/reports/1/map?reference=1.G.1
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 175
```

```
<map>
  <node reference="1.5.1.0" name="California"/>
  <node reference="1.5.1.1" name="Colorado"/>
  <node reference="1.5.1.2" name="DC"/>
  <node reference="1.5.1.3" name="Florida"/>
  <node reference="1.5.1.4" name="Illinois"/>
  <node reference="1.5.1.5" name="Massachusetts"/>
  <node reference="1.5.1.6" name="New York"/>
  <node reference="1.5.1.7" name="Texas"/>
</map>
```


3.4.4 Exporting reports

You can export:

- An entire report as a single document
- A report in paginated mode (one file per report page)
- One page of a report.

You can export reports in the following formats:

- HTML
- MHTML (multipart HTML)
- XML
- PDF
- Excel 2003
- Excel 2007
- CSV

See the related links below for more information about exporting reports.

Related Topics

- [Exporting a report in listing mode \(not paginated\)](#)
- [Exporting a report in paginated mode](#)
- [Exporting a page of a report](#)

3.4.4.1 Exporting a report in listing mode (not paginated)

Exporting a report

This exports the report in the stated format. The document name and format are identified by `<documentname>.xxx`. Where xxx is the format:

- `<documentname>.htm` is a report exported in HTML format.
- `<documentname>.html` is a report exported in MHTML format.
- `<documentname>.xml` is a report exported in XML format.
- `<documentname>.pdf` is a report exported in PDF format.
- `<documentname>.xls` is a report exported in Excel 2003 format.
- `<documentname>.xlsx` is a report exported in Excel 2007 format.
- `<documentname>.csv` is a report exported in CSV format.

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}`

Header	Value
Accept	text/xml text/html or multi-part/related for MHTML, or application/pdf or application/vnd.ms-excel (for Excel 2003) or application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007) text/csv for CSV format
X-SAP-LogonToken	The logon token value, in quotation marks

Table 3-264: Export report: Optional parameter for all supported formats

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any) Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats

Table 3-265: Export Report: Optional Parameter for the HTML Output

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any) Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats
chartOutputFormat	Output format for generated chart. Values: jpeg, bmp, gif, png Default=png Note: in HTML format, image link is generated by the web service, so, the X-SAP-LogonToken must still be valid to get image from the generated link.

Table 3-266: Export Report: Optional Parameter for Excel 2003 and 2007

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any) Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats
optimized	Boolean. Default: false. Otherwise true the generated output is then optimized for calculations inside Excel.

Table 3-267: Export Report: Optional Parameters for CSV

Parameter	Description
textQualifier	(type=character, values= ' or ") a character used to surround each column value.

Parameter	Description
columnDelimiter	(type=string, values=, or ; or the special string Tab) a character put between columns.
charset	(type=string) a valid server charset. To get the list of available charsets, use the command: GET http://<serverName>:6405/biprws/raylight/vx/configuration/charsets

Result:

Header	Value
Status Code	HTTP response code
Content-Type	text/xml text/html or application/pdf or application/vnd.ms-excel (for Excel 2003) or application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007) text/csv for CSV output
Content-Length	Length of content in the response body

Body
The result of the exported report in specified format.

Example: To export a report in HTML format**Note:**

Exports the report in the `exportedreport.htm` file.

Windows

```
curl -G -s -H "accept:text/html" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67?chartOutputFormat=jpeg > exportedreport.htm
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:text/html" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67?chartOutputFormat=jpeg >exportedreport.xml
```

Related Topics

- [Getting the list of supported charsets](#)

3.4.4.2 Exporting a report in paginated mode

Exports in paginated mode to various formats a report of a Web Intelligence document. The first page of a Web Intelligence document begins at 0. The 404 error message "Not found" is returned after retrieving the last report page.

Format can be:

- XML
- PDF
- Excel 2003
- Excel 2007

Request:

```
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/pages
```

Note:

- `{documentId}`: The identifier of the Web Intelligence document is retrieved in the documents list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`
- `{reportId}`: The identifier of the report retrieved by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Header	Value
Accept	text/xml or application/pdf or application/vnd.ms-excel (for Excel 2003) or application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007)
X-SAP-LogonToken	The logon token value, in quotation marks.

Table 3-271: Optional parameter for all supported formats

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any). Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats
mode	normal or quickDisplay

Table 3-272: Optional Parameters mode=normal

Parameter	Description
orientation	Page orientation. Use to force a specific page orientation. Values: portrait, landscape
widthScaling	Number of pages per report displaying in width. Default:0 means no constraint in width.
heightScaling	Number of pages per report displaying in height. Default: 0 means no constraint in height.

Table 3-273: Optional Parameter for Excel 2003 and 2007

Parameter	Description
optimized	Boolean. Default: <code>false</code> . Otherwise <code>true</code> the generated output is then optimized for calculations inside Excel.

Example: To export a report in XML (paginated mode)**Windows**

```
curl -G -s -H "accept:text/xml" -H X-SAP-LogonToken:""<tokenValue>" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:text/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages
```

Example: To export a report in PDF (paginated mode)**Windows**

```
curl -G -s -H "accept:application/pdf" -H X-SAP-LogonToken:""<tokenValue>" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages?mode=normal&widthScaling=1
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/pdf" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages?mode=normal&widthScaling=1
```

Example: To export a report in Excel 2003 (paginated mode)**Windows**

```
curl -G -s -H "accept:application/vnd.ms-excel" -H "X-SAP-LogonToken:""<tokenValue>" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/vnd.ms-excel" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages
```

Example: To export a report in Excel 2007 (paginated mode)**Windows**

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:\""<tokenValue>\"" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/pages
```

UNIX

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:$logtok" http://$serverName:6405/biprws/raylight/v1/documents/9227/reports/1/pages
```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [Getting, moving, or creating, copying a report](#)
- [To log on to the BI platform](#)
- [Exporting a page of a report](#)

3.4.4.3 Exporting a page of a report

Exports a report page to various formats. The format can be:

- HTML
- MHTML (multi-part HTML)
- XML
- PDF
- Excel 2003
- Excel 2007

Request:

```
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/pages/{pageIndex}
```

Note:

- {documentId}: The identifier of the Web Intelligence document is retrieved in the documents list by: GET http://<serverName>:6405/biprws/raylight/vx/documents
- {reportId}: The identifier of the report retrieved by: GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports
- {pageIndex}: The page number. It begins to 0 and while the number of report pages is not known in advance, you have to use this url in a loop which returns the 404 error message "Not Found" after the last found page.

Header	Value
Accept	text/html or multi-part/related for MHTML, or text/xml or application/pdf or application/vnd.ms-excel (for Excel 2003) or application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007)
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report to export. reportId: (type=integer, minimum=1, maximum=highest report number in the document).
pageIndex	Mandatory. Integer. The identifier of the page of the report to export. pageIndex: (type=integer, minimum=1, maximum=highest page number in the report).

Table 3-276: Optional parameter for all supported formats

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any). Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats.

Parameter	Description
mode	normal or quickDisplay

Table 3-277: Optional Parameters mode=normal

Parameter	Description
orientation	Page orientation. Used to force a specific page orientation. Values: portrait, landscape
widthScaling	Number of pages per report displaying in width. Default: 0 means no constraint in width.
heightScaling	Number of pages per report displaying in height. Default: 0 means no constraint in height.

Table 3-278: Optional Parameter for the HTML Output

Parameter	Description
chartOutputFormat	Output format for generated chart (if any). Values: jpeg, bmp, gif, png Default=png Note: in HTML format, image link is generated by the web service, so, the X-SAP-LogonToken must still be valid to get image from the generated link.

Table 3-279: Optional Parameter for Excel 2003 and 2007

Parameter	Description
optimized	Boolean. Default: false. Otherwise true the generated output is then optimized for calculations inside Excel.

Example: To export the report page 1 in XML format**Note:**

Exports the first report page in the `page1.xml` file.

Windows

```
curl -H "accept:text/xml" -H "X-SAP-LogonToken:\""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/0"
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -H "accept:text/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/0"
```

Example: To export the report page 2 in PDF format**Note:**

Exports the report page 2 in the `page2.pdf` file.

Windows

```
curl -H "accept:application/pdf" -H "X-SAP-LogonToken:\""<tokenValue>%" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/1"
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -H "accept:application/pdf" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/1"
```

Example: To exports the report page 3 in Excel 2003 format**Note:**

Exports the report page 3 in the `page3.xls` file.

Windows

```
curl -H "accept:application/vnd.ms-excel" -H "X-SAP-LogonToken:\""<tokenValue>%" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/2"
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -H "accept:application/vnd.ms-excel" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/2"
```

Example: To export the report page 1 in Excel 2007**Note:**

Exports the first report page in the `page1.xlsx` file.

Windows

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken: ""<tokenValue>"" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/0
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/0
```

Related Topics

- [Exporting a report in paginated mode](#)
- [Document: retrieving, copying, or creating](#)
- [Getting, moving, or creating, copying a report](#)
- [To log on to the BI platform](#)

3.4.5 Getting the list of elements used in a report

This URL gets the elements of a specific report, referenced by its ID.

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`.

Note:

`{reportId}`: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

See the link referenced below.

Getting the elements of a Web Intelligence report**Request:**

```
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/elements
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Provides per element of the report: the id, the name, the reference of each element.</p> <p><code><elements> ... </element></code> Contains the list of the descriptions of each element in the report.</p> <p><code><element type="cell"></code> Identifies the element of the report. Elements can be:</p> <p><code><id>4</id></code> The identifier of the element (inter) each identifier is unique.</p> <p><code><name></name></code> The name of the element, if present.</p> <p><code><reference>UIREF:RID=1:BID=4</reference></code></p>

Example: To get the elements of report 2 of document ID 1224**Windows**

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/1224/reports/2
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/1224/reports/2
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<elements>
  <element type="cell">
    <id>4</id>
    <name></name>
    <reference>UIREF:RID=1:RID=4</reference>
  </element>
  <element type="vtable">
    <id>15</id>
    <name>Block 1</name>
    <reference>UIREF:RID=1:RID=15</reference>
  </element>
</elements>
```

3.4.6 Getting report element details, exporting a report element

Use this URL to:

- Get the details of a specific report element, referenced by its ID (GET <url>/documents/{documentId}/reports/{reportId}/elements/{elementId}).
- Export an element of a report (GET -s <url>/documents/{documentId}/reports/{reportId}/elements/{elementId})

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports

Note:

{elementId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/elements`

Getting the details of an element of a report**Request:**

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/elements/{elementId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document . documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type=integer, minimum=1, maximum=highest report number in the document).
elementId	Mandatory. Integer. The identifier of the element of the Web Intelligence report.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Definition of the report element, described as follows:

```
<element type="cell">
  <id>4</id>
  <name></name>
  <reference>UIREF:RID=1:BID=4</reference>
</element>
```

Example: To get the details of element 4 of report 2 of document ID 8022**Windows**

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8022/reports/2/elements/4
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022/reports/2/elements/4
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<elements>
  <element type="cell">
    <id>4</id>
    <name></name>
    <reference>UIREF:RID=1:BID=4</reference>
  </element>
  <element type="vtable">
    <id>15</id>
    <name>Block 1</name>
    <reference>UIREF:RID=1:BID=15</reference>
  </element>
</elements>
```

Exporting an element of a report

Exports a report element to various formats. The format can be:

- HTML
- MHTML
- XML
- PDF
- Excel 2003

- Excel 2007

Note:

If HTML output is chosen, image links will be generated by Raylight: thus, the logon token must still be valid when the HTML output is displayed (to be able to get images from the generated links).

Request:

```
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/elements/{elementId}
```

Note:

- {documentId}: The identifier of the Web Intelligence document is retrieved in the documents list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`
- {reportId}: The identifier of the report retrieved by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Header	Value
Accept	text/html or multipart/related (for MHTML) or text/xml or application/pdf or application/vnd.ms-excel (for Excel 2003) or application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (for Excel 2007)
X-SAP-LogonToken	The logon token value, in quotation marks

Table 3-289: Parameters

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report to export. reportId: (type=integer, minimum=1, maximum=highest report number in the document).

Parameter	Description
elementId	Mandatory. Integer. The identifier of the report element to export. reportId: (type=integer, minimum=1, maximum=highest element number in the report).
dataPath	Optional. String. Specify a data path, for example: <URL>/documents{documentId}/reports/{reportId}/elements/{elementId}?datapath=DP1.DObc:2004,DP1.DOa6:Austin.

Table 3-290: Optional parameter for all supported formats

Parameter	Description
dpi	Resolution in dots per inch (dpi) for generated charts (if any). Max:9600. Min:75 Default:300 for PDF format, 96 for all other formats.
mode	normal or quickDisplay

Table 3-291: Optional Parameters mode=normal

Parameter	Description
orientation	Page orientation. Used to force a specific page orientation. Values: portrait, landscape
widthScaling	Number of pages per report displaying in width. Default: 0 means no constraint in width.
heightScaling	Number of pages per report displaying in height. Default: 0 means no constraint in height.

Table 3-292: Optional Parameter for the HTML Output

Parameter	Description
chartOutputFormat	<p>Output format for generated chart (if any).</p> <p>Values: jpeg, bmp, gif, png</p> <p>Default=png</p> <p>Note: in HTML format, image link is generated by the web service, so, the X-SAP-LogonToken must still be valid to get image from the generated link.</p>

Table 3-293: Optional Parameter for Excel 2003 and 2007

Parameter	Description
optimized	<p>Boolean. Default: false. Otherwise true the generated output is then optimized for calculations inside Excel.</p>

Example: To export a report element in HTML**Windows**

```
curl -G -s -H "accept:text/html" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/ray
light/v1/documents/9227/reports/1/elements/4 > reportelement4.htm
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -s -H "accept:text/html" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/ray
light/v1/documents/9227/reports/1/elements/4
> reportelement4.htm
```

Example: To export a report element in XML**Windows**

```
curl -G -s -H "accept:text/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/ray
light/v1/documents/9227/reports/1/elements/4
> reportelement4.xml
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -s -H "accept:text/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4
> reportelement4.xml
```

Example: To export a report element in PDF**Windows**

```
curl -G -s -H "accept:application/pdf" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4
> reportelement4.pdf
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -s -H "accept:application/pdf" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4
> reportelement4.pdf
```

Example: To export a report element in Excel 2003**Windows**

```
curl -G -s -H "accept:application/vnd.ms-excel" -H "X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4?dpi=150&optimized=true > reportelement4.xls
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -s -H "accept:application/vnd.ms-excel" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4?dpi=150&optimized=true > reportelement4.xls
```

Example: To export a report element in Excel 2007**Windows**

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4 > reportelement4.xlsx
```

UNIX

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H 'X-SAP-LogonToken:"$logtok"' http://$serverName:6405/biprws/raylight/v1/documents/9227/reports/1/elements/4 > reportelement4.xlsx
```

3.4.7 Report structure: getting and updating the structure (specifications) of a report

Use this URL to:

- Get the report structure of the specified report (GET <url>/documents/{documentId}/reports/{reportId}/specification).
- Update the report structure of the specified report (PUT <url>/documents/{documentId}/reports/{reportId}/specification).

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by:
GET http://<serverName>:6405/biprws/raylight/vx/documents

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports

Getting the structure of a Web Intelligence report

The documents are sorted by name. The list depends on user access rights. You can specify the number of documents to return and the first document to be used as the start document in the document list.

Request:

GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/specification

Header	Value
Accept	text/xml
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.
reportId	Mandatory. The ID of the report.

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	text/xml
Content-Length	Length of content in the response body

Body (XML)
<p>The description of the report structure:</p> <pre> <REPORT rid="12" name="Report1"> <PAGE_HEADER bld="1" /> <PAGE_BODY bld="2"> <VTABLE y="100" x="100" bld="21" name="Table 1" > <ROWGROUP type="header" > <TR> <TDCELL bld="211" ><CONTENT>=nameof([YEAR])</CONTENT></TDCELL> <TDCELL bld="212" ><CONTENT>=nameof([INCOME])</CONTENT></TDCELL> </TR> </ROWGROUP> <ROWGROUP type="body" > <TR> <TDCELL bld="213" ><CONTENT>=[YEAR]</CONTENT></TDCELL> <TDCELL bld="214" ><CONTENT>=[INCOME]</CONTENT></TDCELL> </TR> </ROWGROUP> </VTABLE> </PAGE_BODY> <PAGE_FOOTER bld="3" /> </REPORT> </pre>

Example: To get the structure of a report**Request:****Windows**

```
curl -G -i -H "accept:text/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:text/xml" 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<REPORT rId="12" name="Report1">
  <PAGE_HEADER bId="1" />
  <PAGE_BODY bId="2">
    <VTABLE y="100" x="100" bId="21" name="Table 1" >
      <ROWGROUP type="header" >
        <TR>
          <TDCELL bId="211" ><CONTENT>=nameof ({ YEAR}) </CONTENT></TDCELL>
          <TDCELL bId="212" ><CONTENT>=nameof ({ INCOME}) </CONTENT></TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP type="body" >
        <TR>
          <TDCELL bId="213" ><CONTENT>= [YEAR] </CONTENT></TDCELL>
          <TDCELL bId="214" ><CONTENT>= [INCOME] </CONTENT></TDCELL>
        </TR>
      </ROWGROUP>
    </VTABLE>
  </PAGE_BODY>
  <PAGE_FOOTER bId="3" />
</REPORT>
```

Updating the report structure for a specified report**Request**

```
PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/specification
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
documentId	Mandatory. The ID of the document.
reportId	Mandatory. The ID of the report.

Body (XML)

```

<REPORT rId="12" name="Report1">
  <PAGE_HEADER bId="1" />
  <PAGE_BODY bId="2">
    <VTABLE y="100" x="100" bId="21" name="Table 1" >
      <ROWGROUP type="header" >
        <TR>
          <TDCELL bId="211" ><CONTENT>=nameof ([YEAR])</CONTENT></TDCELL>
          <TDCELL bId="212" ><CONTENT>=nameof ([INCOME])</CONTENT></TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP type="body" >
        <TR>
          <TDCELL bId="213" ><CONTENT>=[YEAR]</CONTENT></TDCELL>
          <TDCELL bId="214" ><CONTENT>=[INCOME]</CONTENT></TDCELL>
        </TR>
      </ROWGROUP>
    </VTABLE>
  </PAGE_BODY>
  <PAGE_FOOTER bId="3" />
</REPORT>

```

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	text/xml
Content-Length	Length of content in the response body

Body (XML)

```

<success><message>The resource of type 'report' with identifier '12' has
been successfully updated.</message> <id>12</id></success>

```


Example: 1: To add a simple cell to a report specifications for report ID 23

You define the report specifications in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; cell.xml.

Body (XMLfile)

```
<REPORT rId="12" name="My First Cell"> <PAGE_BODY>
  <CELL x="1000" y="1500">
    <CONTENT>"My First CELL"</CONTENT>
  </CELL>
</PAGE_BODY>
</REPORT>
```

Request:**Windows**

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" X-SAP-LogonToken:"""%tokenValue%"""
-d "@cell.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" 'X-SAP-LogonToken:"$logtok"' -d
"@cell.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<success><message>The resource of type 'Report' with identifier '23' has been successfully updated.</message>
<id>23</id>
</success>
```

Example: 2: To add a simple vertical table to a report ID 23

You define the report specifications in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; table.xml.

Body (XMLfile)

```
<REPORT name="Vertical Table">
  <PAGE_HEADER/>
  <PAGE_BODY>
    <VTABLE name="My Vertical Table" x="3037" y="4455">
      <ROWGROUP type="HEADER">
        <TR>
          <TDCELL>
            <CONTENT>=NameOf ([Country])</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=NameOf ([Revenue])</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP type="BODY">
        <TR>
          <TDCELL>
            <CONTENT>=[Country]</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=[Revenue]</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP type="FOOTER">
        <TR>
          <TDCELL>
            <CONTENT/>
          </TDCELL>
          <TDCELL>
            <CONTENT>=[Revenue]</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
    </VTABLE>
  </PAGE_BODY>
  <PAGE_FOOTER/>
</REPORT>
```

Request:

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" X-SAP-LogonToken:"""%tokenValue%"""  
-d "@table.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" 'X-SAP-LogonToken:"$logtok"' -d  
"@table.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<success><message>The resource of type 'Report' with identifier '23' has been successfully updated.</message>
<id>23</id>
</success>
```

Example: 3: To add a simple vertical table with break to a report ID 23

You define the report specifications in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; tablebreak.xml.

Body (XMLfile)

```
<REPORT name="Vertical Table And Break">
  <PAGE_HEADER/>
  <PAGE_BODY>
    <VTABLE name="Table And Break " x="3037" y="4455">
      <AXIS>
        <EXPRS>
          <AXIS_EXPR>=[Country]</AXIS_EXPR>
          <AXIS_EXPR>=[Resort]</AXIS_EXPR>
        </EXPRS>
        <BREAK bId="1" expr="=[Country]" addSort="yes" onePage="yes" newPage="yes" duplicate="center"/>
      </AXIS>
      <ROWGROUP breakId="1" type="HEADER">
        <TR>
          <TDCELL>
            <CONTENT>=NameOf ([Country])</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=NameOf ([Resort])</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=NameOf ([Revenue])</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP type="BODY">
        <TR>
          <TDCELL>
            <STYLE>
              <TEXTVALIGN value="center"/>
            </STYLE>
            <CONTENT>=[Country]</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=[Resort]</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=[Revenue]</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP breakId="1" type="FOOTER">
        <TR>
          <TDCELL>
            <CONTENT>=[Country]</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT/>
          </TDCELL>
          <TDCELL>
            <CONTENT/>
          </TDCELL>
        </TR>
      </ROWGROUP>
    </VTABLE>
  </PAGE_BODY>
  <PAGE_FOOTER/>
</REPORT>
```

Request:

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" X-SAP-LogonToken:"""%tokenValue%"""  
-d "@tablebreak.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" 'X-SAP-LogonToken:"$logtok"' -d  
"@tablebreak.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK  
Server: Apache-Coyote/1.1  
Date: Fri, 01 Jun 2012 09:49:52 GMT  
Content-Type: application/xml  
Content-Length: 217  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<success><message>The resource of type 'Report' with identifier '23' has been successfully updated.</message>  
<id>23</id>  
</success>
```

Example: 4: To add a simple vertical table with one filter on the report and one filter on the table to a report ID 23

You define the report specifications in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; tablefilter.xml.

Body (XMLfile)

Body (XMLfile)

```

<REPORT name="Filter">
  <DATA>
    <DATA_FILTER>
      <WHERE>
        <FILTER key="[Country]">
          <CONDITION operatorCondition="InList">
            <MEMBER>France</MEMBER>
          </CONDITION>
        </FILTER>
      </WHERE>
    </DATA_FILTER>
  </DATA>
  <PAGE_BODY>
    <VTABLE bId="17" name="Block 1" x="3749" y="3443" >
      <AXIS>
        <EXPRS>
          <AXIS_EXPR>=[Country]</AXIS_EXPR>
          <AXIS_EXPR>=[Resort]</AXIS_EXPR>
        </EXPRS>
      </AXIS>
      <DATA>
        <DATA_FILTER>
          <WHERE>
            <FILTER key="[Resort]">
              <CONDITION operatorCondition="InList">
                <MEMBER>French Riviera</MEMBER>
              </CONDITION>
            </FILTER>
          </WHERE>
        </DATA_FILTER>
      </DATA>
      <ROWGROUP type="HEADER">
        <TR height="567">
          <TDCELL>
            <CONTENT>=NameOf ([Country])</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=NameOf ([Resort])</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=NameOf ([Revenue])</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
      <ROWGROUP type="BODY">
        <TR height="567">
          <TDCELL>
            <CONTENT>=[Country]</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=[Resort]</CONTENT>
          </TDCELL>
          <TDCELL>
            <CONTENT>=[Revenue]</CONTENT>
          </TDCELL>
        </TR>
      </ROWGROUP>
    </VTABLE>
  </PAGE_BODY>
</REPORT>

```

Request:

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" X-SAP-LogonToken:"""%tokenValue%"""  
-d "@tablefilter.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" 'X-SAP-LogonToken:"$logtok"' -d  
"@tablefilter.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK  
Server: Apache-Coyote/1.1  
Date: Fri, 01 Jun 2012 09:49:52 GMT  
Content-Type: application/xml  
Content-Length: 217  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<success><message>The resource of type 'Report' with identifier '23' has been successfully updated.</message>  
<id>9372</id>  
</success>
```

Example: 5: To add a section (sort and filter) with a table and free cell to a report ID 23

You define the report specifications in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; section.xml.

Body (XMLfile)

```
<REPORT name="Section & Filter">
  <DATA/>
  <PAGE_BODY>
    <ROW height="6255"/>
    <SECTION>
      <DATA>
        <DATA_FILTER>
          <WHERE>
            <FILTER key="[Country]">
              <CONDITION operatorCondition="InList">
                <MEMBER>France</MEMBER>
              </CONDITION>
            </FILTER>
          </WHERE>
        </DATA_FILTER>
      </DATA>
      <AXIS>
        <SORTS>
          <SORT sign="desc" expr="[Country]"/>
        </SORTS>
        <AXIS_EXPR>[Country]</AXIS_EXPR>
      </AXIS>
      <SBODY bottomPadding="3638" bookmark="yes">
        <CELL class="ia-section-cell" x="600" y="500">
          <CONTENT>[Country]</CONTENT>
        </CELL>
        <VTABLE name="Block 1" x="3749" y="500">
          <DATA/>
          <ROWGROUP type="HEADER">
            <TR>
              <TDCELL>
                <CONTENT>NameOf ([Resort])</CONTENT>
              </TDCELL>
              <TDCELL>
                <CONTENT>NameOf ([Revenue])</CONTENT>
              </TDCELL>
            </TR>
          </ROWGROUP>
          <ROWGROUP type="BODY">
            <TR>
              <TDCELL>
                <CONTENT>[Resort]</CONTENT>
              </TDCELL>
              <TDCELL>
                <CONTENT>[Revenue]</CONTENT>
              </TDCELL>
            </TR>
          </ROWGROUP>
        </VTABLE>
      </SBODY>
    </SECTION>
    <ROW height="1500"/>
  </PAGE_BODY>
</REPORT>
```

Request:

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" X-SAP-LogonToken:"""%tokenValue%"""  
-d "@section.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" 'X-SAP-LogonToken:"$logtok"' -d  
"@section.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK  
Server: Apache-Coyote/1.1  
Date: Fri, 01 Jun 2012 09:49:52 GMT  
Content-Type: application/xml  
Content-Length: 217  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
  
<success><message>The resource of type 'Report' with identifier '23' has been successfully updated.</message>  
<id>23</id>  
</success>
```

Example: Add a free cell with hide conditional

Body (freecell.xml file)

```
<REPORT name="Cell with hide always and Cell with hide conditional">  
  <DATA/>  
  <PAGE_BODY>  
    <ROW height="2992" bottomPadding="0">  
      <CELL x="7416" y="432" hideAlways="yes" width="15048" height="1547">  
        <CONTENT>ALWAYS HIDE</CONTENT>  
      </CELL>  
      <CELL x="7416" y="2000" hideAlways="no" hideExpr="=Count([Country]) > 3" width="15048" height="1547">  
        <CONTENT>=Count([Country]) > 3</CONTENT>  
      </CELL>  
    </ROW>  
  </PAGE_BODY>  
</REPORT>
```

Request:

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" X-SAP-LogonToken:"""%tokenValue%"""  
-d "@freecell.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:text/xml" 'X-SAP-LogonToken:$logtok' -d
"@freecell.xml" http://<serverName>:6405/biprws/raylight/v1/documents/2334/reports/23/specification
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<success><message>The resource of type 'Report' with identifier '23' has been successfully updated.</message>

<id>23</id>
</success>
```

3.4.8 Drilling on report data

You use the drill function to analyze the results displayed in reports. Drilling on reports lets you look deeper into your data to discover the details behind a good or bad summary result displayed in tables, charts, or sections.

- Get the current drill information for a report (GET <url>/documents/{documentId}/reports/{reportId}/driller)
- Activate the query drill (POST <url>/documents/{documentId}/reports/{reportId}/driller)
- Update the drill output mode (PUT <url>/documents/{documentId}/reports/{reportId}/driller)
- Disable the drill (DELETE <url>/documents/{documentId}/reports/{reportId}/driller)
- Get the drill hierarchies (GET <url>/documents/{documentId}/reports/{reportId}/driller/hierarchies).
- Get the list of drill filters for a report (GET <url>/documents/{documentId}/reports/{reportId}/driller/filters)
- Get the details of a drill filter (GET <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId})
- Create a drill filter (POST <url>/documents/{documentId}/reports/{reportId}/driller/filters)
- Update a drill filter (PUT <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId})
- Remove a drill filter (DELETE <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId})
- Get the list of free drill elements (GET <url>/documents/{documentId}/reports/{reportId}/driller/drillelements)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Related Topics

- [Getting information about the query drill settings](#)
- [Enabling the query drill](#)
- [Changing the drill setting](#)
- [Disabling the query drill](#)
- [Getting information about the drill hierarchies](#)
- [Getting the list of drill filters for a report](#)
- [Getting the details of a drill filter](#)
- [Creating a drill filter for a report](#)
- [Updating a drill filter](#)
- [Removing a drill filter](#)
- [Getting the list of free drill elements](#)

3.4.8.1 Getting information about the drill hierarchies

Obtains the drill hierarchies. Returns an error, if the drill is not activated on this report.

GET `<url>/documents/{documentId}/reports/{reportId}/driller/hierarchies`

Obtains the current drill information.

Returns an error if the drill is not activated on the report.

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/hierarchies`

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json

Header	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type=integer, minimum=1, maximum=highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json

Header	Value
Content-Length	Length of content in the response body

Body (XML)

```
<hierarchies>
  <hierarchy>
    <id>DP0.DH1</id>
    <name>Resort Hierarchy</name>
    <dataProviderId>DP0</dataProviderId>
    <elements>
      <element qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
        <id>DP0.D039</id>
        <name>Country</name>
      </element>
      <element qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
        <id>DP0.D02</id>
        <name>Resort</name>
      </element>
      <element qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
        <id>DP0.D04</id>
        <name>Service Line</name>
      </element>
      <element qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
        <id>DP0.D05</id>
        <name>Service</name>
      </element>
    </elements>
  </hierarchy>
</hierarchies>
```

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>

<message>The resource of type 'Driller' does not exist.</message>
</error>
```

Example: To get the drill hierarchy information for report 1 of document ID 4567**Windows**

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H X-SAP-LogonToken:""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller/hierarchies
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"'
http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller/hierarchies
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 775

<hierarchies>
  <hierarchy>
    <id>DP0.DH1</id>
    <name>Resort Hierarchy</name>
    <dataProviderId>DP0</dataProviderId>
    <elements>
      <element>
        <id>DP0.DO39</id>
        <name>Country</name>
        <description></description>
        <filterValue></filterValue>
        <qualification>dimension</qualification>
        <inQuery>false</inQuery>
        <ambiguous>false</ambiguous>
        <inScope>true</inScope>
      </element>
      <element>
        <id>DP0.DO2</id>
        <name>Resort</name>
        <description></description>
        <filterValue></filterValue>
        <qualification>dimension</qualification>
        <inQuery>false</inQuery>
        <ambiguous>false</ambiguous>
        <inScope>true</inScope>
      </element>
      <element>
        <id>DP0.DO4</id>
        <name>Service Line</name>
        <description></description>
        <filterValue></filterValue>
        <qualification>dimension</qualification>
        <inQuery>false</inQuery>
        <ambiguous>false</ambiguous>
        <inScope>true</inScope>
      </element>
      <element>
        <id>DP0.DO5</id>
        <name>Service</name>
        <description></description>
        <filterValue></filterValue>
        <qualification>dimension</qualification>
        <inQuery>false</inQuery>
        <ambiguous>false</ambiguous>
        <inScope>true</inScope>
      </element>
    </elements>
  </hierarchy>
</hierarchies>
```

Related Topics

- [Getting information about the query drill settings](#)
- [Enabling the query drill](#)
- [Changing the drill setting](#)
- [Disabling the query drill](#)

3.4.8.2 Getting the list of drill filters for a report

Use this to:

- Get the list of drill filters for a report (GET <url>/documents/{documentId}/reports/{reportId}/driller/filters)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Getting the list of report drill filters

Obtains the list of drill filters for a report.

Returns an error if the drill is not activated on the report.

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/filters`

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).

Parameter	Description
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Description of the available filters for the report.</p> <pre><filters> <filter qualification="Dimension" inQuery="false" ambiguous="false" inScope="true"> <id>DP0.DO39</id> <value>France</value> </filter> </filters></pre>

In case of error: HTTP Response Code: 404
<pre><error> <error_code>400</error_code> <message>The resource of type 'Driller' does not exist.</message> </error></pre>

Example: To get the list of drill filters for report 1 of document ID 4567**Windows**

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H X-SAP-LogonToken:"""%tokenValue"""" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller/filters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller/filters
```

Response

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 375

<filters>
  <filter qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO39</id>
    <value>France</value>
  </filter>
</filters>

```

3.4.8.3 Getting the list of free drill elements

Obtains the free drill elements. Returns an error, if the drill is not activated on this report.

Use this to:

- Get the list of free drill elements for a report (GET <url>/documents/{documentId}/reports/{reportId}/driller/drillelements)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports

Getting the list of free drill elements

Obtains the list of free drill elements for a report.

Returns an error if the drill is not activated on the report.

Request URL

GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/drillelements

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Description of the available drill elements for the report.

drillelements: Contains the list of available drill elements.

drillelement **qualification=:** describes a drill element;

id: A unique identifier for the drill element.

name: The name of the drill element.

```
<drillelements>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO39</id>
    <name>Country</name>
  </drillelement>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO2</id>
    <name>Resort</name>
  </drillelement>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO4</id>
    <name>Service Line</name>
  </drillelement>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO5</id>
    <name>Service</name>
  </drillelement>
</drillelements>
```

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Driller' does not exist.</message>
</error>
```

Example: To get the list of available drill elements for report 1 of document ID 4567**Windows**

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller/drillelements
```

UNIX**Note:**

Uses the `logtok` variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok"" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller/drillelements
```

Response

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 375

<drillelements>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO39</id>
    <name>Country</name>
  </drillelement>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO2</id>
    <name>Resort</name>
  </drillelement>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO4</id>
    <name>Service Line</name>
  </drillelement>
  <drillelement qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
    <id>DP0.DO5</id>
    <name>Service</name>
  </drillelement>
</drillelements>

```

3.4.8.4 Getting information about the query drill settings

Use this to:

- Get current drill information for a report (GET <url>/documents/{documentId}/reports/{reportId}/driller)
- Activate the query drill (POST <url>/documents/{documentId}/reports/{reportId}/driller)
- Update the drill output mode (PUT <url>/documents/{documentId}/reports/{reportId}/driller)
- Delete the driller resource (this deactivates the drill) (DELETE <url>/documents/{documentId}/reports/{reportId}/driller)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports>

Getting information about the report driller settings

Obtains the current drill information.

Returns an error if the drill is not activated on the report.

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	<code>application/xml</code> or <code>application/json</code>
Content-Length	Length of content in the response body

Body (XML)

```
<driller>
  <output>true</output>
</driller>
```

Where: <output>=**boolean**, specifies the mode of drill (true = drill or false = drill light)

In case of error: HTTP Response Code: 404

```
<error>
  <error_code>400</error_code>
  <message>The resource of type 'Driller' does not exist.</message>
</error>
```

Example: To get the drill settings for report 1 of document ID 4567**Windows**

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 175

<driller>
  <output>true</output>
</driller>
```

Related Topics

- [Enabling the query drill](#)
- [Changing the drill setting](#)
- [Disabling the query drill](#)
- [Getting information about the drill hierarchies](#)

3.4.8.5 Changing the drill setting

Request URL

PUT `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type=integer, minimum=1, maximum=highest report number in the document).

Body XML
<pre><driller> <output>false</output> </driller></pre>

Reponse:

Header	Value
Status Code	HTTP response code

Header	Value
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

```
<success><message>The resource of type 'Driller' has been successfully
updated. </message>
<id>1</id>
</success>
```

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Driller' does not exist.</message>
<id>1</id>
/error>
```

Example: To change the drill setting

You define the driller state in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; drilelr.xml.

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%token
Value%"""" -d "@driller.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/reports/2/driller
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok"
-d "@driller.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/reports/2/driller
```

Request body

```
<driller>
  <output>false</output>
</driller>
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Driller' has been successfully updated.</message>
  <id>1</id>
</success>
```

Related Topics

- [Getting information about the query drill settings](#)
- [Enabling the query drill](#)
- [Disabling the query drill](#)
- [Getting information about the drill hierarchies](#)

3.4.8.6 Enabling the query drill

Activate the drill on a report. The request body is optional. If not provided, the drill will be activated with default settings (output set to true). Throws an error if resource is not created.

Request URL

POST `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller`

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type=integer, minimum=1, maximum=highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre><success> <message>The resource of type 'Driller' has been successfully created. </message> <id>2</id> </success></pre>

In case of error: HTTP Response Code: 400
<pre><error> <error_code>401</error_code> <message>The resource of type 'Driller' is already created.</message> </error></pre>

Example: To activate the drill for document ID 4567**Windows**

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/4567/reports/1/driller
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Driller' has been successfully created.</message>
<id>2</id>
</success>
```

Related Topics

- [Getting information about the query drill settings](#)
- [Changing the drill setting](#)
- [Disabling the query drill](#)
- [Getting information about the drill hierarchies](#)

3.4.8.7 Disabling the query drill

Disables the drill on the specified report. Throws an error if resource is not created.

Request URL

```
DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller
```

Notes

Header (typically, but not exclusively)	Value
Accept	To retrieve the details of a document: application/xml or application/json

Header (typically, but not exclusively)	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type=integer, minimum=1, maximum=highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre><success> <message>The resource of type 'Driller' has been successfully removed. </message> <id>2</id> </success></pre>

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Driller' does not exist.</message>
</error>
```

Example: To diasble the drill on a report**Windows**

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/8022/reports/1/driller
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8022/reports/1/driller
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Driller' has been successfully removed.</message>
</id>2</id>
</success>
```

Related Topics

- [Getting information about the query drill settings](#)
- [Enabling the query drill](#)
- [Changing the drill setting](#)
- [Getting information about the drill hierarchies](#)

3.4.8.8 Getting the details of a drill filter

Use this to:

- Get the details of a drill filter for a report (GET <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId})

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports>

Getting the details of a report drill filter

Obtains the details of the drill filter for a report..

Returns an error if the drill is not activated on the report.

Request URL

GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/filters/{filterId}>

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).
filterId	Mandatory. Integer. The identifier of the drill filter of the report. filterId: (type = integer, minimum = 1, maximum = highest filter number in the report).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Description of the drill filter.

```
<filter qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
  <id>DP0.D013</id>
  <lov hierarchical="false" partial="false" refreshable="false">
    <values>
      <value>Albertville</value>
      <value>Augsburg</value>
      <value>Belfast</value>
      .
      .
      <value>Washington D.C.</value>
      <value>Yokohama</value>
    </values>
    <columns mappingId="0">
      <column id="0" type="String">
        </column>
      </columns>
    </lov>
  </filter>
```

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Driller' does not exist.</message>
</error>
```

Example: To get the details of the drill filter DP0.D013 for report 1 of document ID 13343

Windows

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue"""" http://<serverName>:6405/biprws/raylight/v1/documents/13343/reports/1/driller/filters/DP0.D013
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -G -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:$logtok'
http://<serverName>:6405/biprws/raylight/v1/documents/13343/reports/1/driller/filters/DP0.DO13
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 375
```

```
<filter qualification="Dimension" inQuery="false" ambiguous="false" inScope="true">
  <id>DP0.DO13</id>
  <lov hierarchical="false" partial="false" refreshable="false">
    <values>
      <value>Albertville</value>
      <value>Augsburg</value>
      <value>Belfast</value>
      . . .
      <value>Washington D.C.</value>
      <value>Yokohama</value>
    </values>
    <columns mappingId="0">
      <column id="0" type="String">
        </column>
      </columns>
    </lov>
  </filter>
```

3.4.8.9 Creating a drill filter for a report

Use this to:

- Create a drill filter for a report (POST `<url>/documents/{documentId}/reports/{reportId}/driller/filters`)

Note:

`{documentId}`: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

`{reportId}`: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports`

Creating a report drill filter

Creates a drill filter for a report.

Returns an error if the drill is not activated on the report.

Request URL

POST `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/filters`

Header	Value
Accept	To retrieve the details of a document: <code>application/xml</code> or <code>application/json</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Body (XML)
<p>Write an xml file describing the description of the report drill filter and include the xml file as the body of the request. For example drillfilter1.xml.</p> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <filter> <id>DP0.DO39</id> <value>France</value> </filter></pre>

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	<code>application/xml</code> or <code>application/json</code>

Header	Value
Content-Length	Length of content in the response body

Body (XML)
<p>Message giving the result of the request.</p> <pre><success> <message>The resource of type 'DrillFilter' with identifier 'DP0.O39' has been successfully created.</message> <id>DP0.O39</id> </success></pre>

Example: To create a drill filter for report 1 of document 127

Note:

Use the xml file drillfilter2.xml for the body of the request.

drillfilter2.xml details:

```
<filter>
  <id>DP1.DO22</id>
  <value>France</value>
</filter>
```

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" -d "@drillfilter2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/127/reports/1/driller/filters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" -d "@drillfilter2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/127/reports/1/driller/filters
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'DrillFilter' with identifier 'DP1.DO22' has been successfully created.</message>
  <id>DP1.DO22</id>
</success>
```

3.4.8.10 Updating a drill filter

Use this to:

- Update the definition of a drill filter for a report (PUT <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId})

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports>

Updating a report drill filter

Updates the description of a drill filter for a report.

Returns an error if the drill is not activated on the report.

Request URL

PUT <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/filters/{filterId}>

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).

Parameter	Description
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).
filterId	Mandatory. The identifier of the drill filter of the report.

Body (XML)
Description of the updated drill filter. <pre><filter> <value>Germany</value> </filter></pre>

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

In case of success:
<pre><success> <message>The resource of type 'DrillFilter' with identifier 'DP2.011' has been successfully updated. </message> <id>DP2.011</id> </success></pre>

In case of error: HTTP Response Code: 404

```
<error>
<error_code>400</error_code>

<message>The resource of type 'Driller' does not exist.</message>

</error>
```

Example: To update drill filter DP2.011 of report 2 of document 223**Note:**

Write the new drill filter description in the `updatedrillfilter.xml` file called by `-d "@updatedrillfilter.xml"` in cURL command.

`updatedrillfilter.xml` details

```
<filter>
  <value>Germany</value>
</filter>
```

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -d "@updatedrillfilter.xml"
-H 'X-SAP-LogonToken:""%tokenValue%""' http://<serverName>:6405/biprws/raylight/v1/documents/223/re
ports/2/driller/filters/DP2.011
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -d "@updatedrillfilter.xml"
-H 'X-SAP-LogonToken:""%logtok%""' http://<serverName>:6405/biprws/raylight/v1/documents/223/re
ports/2/driller/filters/DP2.011
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Mon, 11 Jun 2012 08:38:03 GMT
Content-Type: application/xml
Content-Length: 204

<success>
  <message>The resource of type 'DrillFilter' with identifier 'DP2.011' has been successfully updated.</mes
  sage>
  <id>DP2.011</id>
</success>
```

3.4.8.11 Removing a drill filter

Use this to:

- Remove a drill filter for a report (DELETE <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId})

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports>

Removing a report drill filter

Deletes the drill filter for a report.

Returns an error if the drill is not activated on the report.

Request URL

DELETE <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/filters/{filterId}>

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).
filterId	Mandatory. The identifier of the drill filter of the report.

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
```

```
<message>The resource of type 'DrillFilter' with identifier 'DP1.034' has
been deleted successfully.</message>
```

```
<id>DP1.034</id>
```

```
<success>
```

Otherwise: <error> ... </error>

Example: To delete a drill filter DP5.012 from report 67 of document ID 9512

Windows

```
curl -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to
kenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67/driller/filters/DP5.012
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$logtok"
http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67/driller/filters/DP5.012
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<success>
<message>The resource of type 'DrillFilter' with identifier 'DP5.012' has been successfully deleted.</mes
sage>
```



```
<id>DP5.012</id>  
</success>
```

3.4.8.12 Performing a drill action

This method performs a drill action on a report. This throws an error if the resource is not created or if the parameters are not valid.

- Perform a drill action for a report (POST <url>/documents/{documentId}/reports/{reportId}/driller/instructions)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports>

Request URL

POST <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/instructions>

Header	Value
Accept	To retrieve the details of a document:application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).

Parameter	Description
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Body (XML)
<p>The drill instructions, in an XML file:</p> <pre><instruction type="Down"> Where type can be "Down", "Up", or "By". <elementId>11</elementId> Where elementId value is equal to the ID of the element on which you want to perform the drill action (ex : the ID of a vertical table). <from> <drillElement> <id>DP0.DO2</id> <filterValue>French Riviera</filterValue> </drillElement> </from> <to> <drillElement> <id>DP0.DO4</id> <hierarchyId>DP0.DH1</hierarchyId> </drillElement> </to> </instruction></pre>

Reponse:

Header	Value
Status Code	HTTP response code

Header	Value
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'DrillerInstruction' has been successfully created.</message> <id>1</id> </success></pre> <p>Otherwise: <error></p> <pre><error_code>400</error_code> <message>The resource of type 'Driller' does not exist.</message> </error></pre>

Example: To drill down

Windows

```
curl -POST -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%tokenValue%"" -d "@drilldown.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67/driller/instructions
```

UNIX

Note:

Uses the logtok variable set at login

```
curl POST -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok" -d "@drilldown.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67/driller/instructions
```

XML body file content (drilldown.xml):

```
<instruction type="Down">
  <elementId>11</elementId>
  <from>
    <drillElement>
      <id>DP0.DO2</id>
      <filterValue>French Riviera</filterValue>
```

```

    </drillElement>
  </from>
  <to>
    <drillElement>
      <id>DP0.D04</id>
      <hierarchyId>DP0.DH1</hierarchyId>
    </drillElement>
  </to>
</instruction>

```

Response

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'DrillerInstruction' has been successfully created.</message>
  <id>1</id>
</success>

```

3.4.8.13 Making a snapshot of a report in drill mode

This method makes a snapshot of a report in mode. This throws an error if the resource is not created or if the parameters are not valid.

- Perform a drill action for a report (POST <url>/documents/{documentId}/reports/{reportId}/driller/snapshot)

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{reportId}: The identifier of the Web Intelligence document report retrieved in the document's reports list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports>

Request URL

POST <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}/driller/snapshot>

Header	Value
Accept	To retrieve the details of a document: application/xml or application/json

Header	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document. documentId: (type = integer, minimum = 1, maximum = highest document number in existing documents).
reportId	Mandatory. Integer. The identifier of the report of the Web Intelligence report. reportId: (type = integer, minimum = 1, maximum = highest report number in the document).

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success>
<message>The resource of type 'Report' with identifier '{reportId}' has
been successfully created.</message>
<id>{reportId}</id>
</success>
```

Otherwise:

```
<error>
<error_code>400</error_code>
<message>The resource of type 'Driller' does not exist.</message>
</error>
```

Example: To make a snapshot of a report in drill mode**Note:**

Makes a snapshot of report 12 of document 7738.

Windows

```
curl "POST" -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""tokenValue\"" http://<server
Name>:6405/biprws/raylight/v1/documents/7738/reports/12/driller/snapshot
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl "POST" -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<server
Name>:6405/biprws/raylight/v1/documents/7738/reports/12/driller/snapshot
```

Response

```
<success>
  <message>Resource of type 'Report' with identifier '12' has been successfully created.</message>
  <id>12</id>
</success>
```

3.5 Managing data providers

A data provider is a query. This section describes the main operations available on the data provider:

- Managing data providers
- Managing flows
- Managing the query specification
- Getting the details of query mapping

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

Restrictions

The following workflows are not delivered or supported:

- Using Analysis Views as data source for a new data provider.
- Using a Custom Data Provider as data source for a new data provider.
- Changing data source when based on Analysis Views and Custom Data Provider.

Related Topics

- [Getting a list of data providers or adding a data provider](#)
- [Getting the details of a data provider or updating or deleting a data provider](#)
- [Flows: getting the flows count for a data provider](#)
- [Flows: getting the details of a flow in CSV or XML format](#)

3.5.1 Getting a list of data providers or adding a data provider

You can:

- Get the list of data providers of a document (GET <url>/documents/{documentId}/dataproviders).
- Add a data provider to the document (POST <url>/documents/{documentId}/dataproviders).
- Move a data provider from the document (PUT <url>/documents/{documentId}/dataproviders).

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the document list by:
GET http://<serverName>:6405/biprws/raylight/vx/documents

Getting the list of data providers of a document

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>For each data provider: id, name and the date of the last update.</p> <pre><dataproviders> <dataprovider> <id>DP1</id> <name>Query 1</name> </dataprovider> </dataproviders></pre>

Example: To get the list of dataproviders for a document

Note:

The document ID is 7738

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 07 Jun 2012 08:23:37 GMT
Content-Type: application/xml
Content-Length: 217
```

```
<dataproviders>
  <dataprovider>
    <id>DP0</id>
    <name>Query 1</name>
    <updated>2006-09-20Z</updated>
  </dataprovider>
  ...
</dataprovider>
</dataproviders>
```

Adding a data provider to a document

This URL allows to add a new data provider to a Web Intelligence document.

The data provider is defined by its name, unique within the WebI document, and the CUID of either a universe (UNV or UNX) or a connection specification for a direct access to data source.

Request:

POST http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders

Header	Value
Accept	application/xml or application/json

Header	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Body (XML)	Description
<pre><dataprovider> <name>Query1</name> <datasourceId>1234ABCD</datasourceId> </dataprovider></pre>	<p>Mandatory.</p> <p>Used to give the name and ID of the data source <code>name</code> is the name of the data source.</p> <p><code>datasourceId</code> is the reference ID of the data source.</p>

Body when referring to a BEx query (XML)	Description
<pre><dataprovider> <name>Query1</name> <datasourceId>11990;Z_BOBJ;AAQUERY_SAMPLE</datasourceId> </dataprovider></pre>	<p>Mandatory.</p> <p>Used to give the name and ID of the data source <code>name</code> is the name of the data source.</p> <p><code>datasourceId</code> is the reference ID of the BEx data source.</p>

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre><success> <message> Success. the resource of type 'Data Provider' with identifier 1234ABCD has been successfully updated. </message> </success></pre>

Example: To add a data provider to a document

You define the data provider add in the body which is defined in an .xml file saved in the current path (usually the same path as the Curl tool). For example; dp3.xml.

Request:

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%to  
kenValue%"" -d "@dp3.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/dataproviders
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "POST" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""$log  
tok" -d "@dp3.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/dataproviders
```

Response

```
HTTP/1.1 200 OK  
Server: Apache-Coyote/1.1  
Date: Tue, 05 Jun 2012 08:10:36 GMT  
Content-Type: application/xml  
Content-Length: 355
```

```
<success>  
  <message>The resource of type 'Data provider' with identifier 'DP3' has been successfully created.</message>  
  <id="67"></id>  
</success>
```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [To log on to the BI platform](#)

3.5.2 Moving a data provider in a document

Move an existing data provider in a Web Intelligence document.

Moves the data provider reference by the parameter `dataproviderId` within a Web Intelligence document specified by the parameter `documentId`.

Request:

```
PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dat  
aproviders?fromId=<DP_to_move>&toID=<pos_of_DP_aftermove>
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory: A valid ID identifier of a Web Intelligence document
fromId	Mandatory: The IDs of the data provider to move.
toId	Mandatory: The ID of the data provider (position of the data provider after the move).

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre>HTTP/1.1 200 OK Server: Apache-Coyote/1.1 Date: Tue, 05 Jun 2012 08:10:36 GMT Content-Type: application/xml Content-Length: 355 <success> <message>The resource of type 'Document' with identifier 'DP0' has been successfully updated.</message> <id>DP2</id> </success></pre>

Example: To move DP0 of document ID 8022 to position DP2**Windows**

```
curl -i -X "PUT" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8022/dataproviders?fromID=DP0&toId=DP2
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -X "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8022/dataproviders?fromID=DP0&toId=DP2
```

Response**Body (XML)**

```
<success>
  <message>Resources of type 'Data provider' with identifier 'DP0' has been successfully moved.</message>
  <id>DP2</id>
</success>
```

3.5.3 Getting the details of a data provider or updating or deleting a data provider

Use this URL to:

- Get the details of a data provider for a document (GET <url>/documents/{documentId}/dataproviders/{dataproviderId})
- Update or purge a data provider (PUT <url>/documents/{documentId}/dataproviders/{dataproviderId})
- Delete a data provider (DELETE <url>/documents/{documentId}/dataproviders/{dataproviderId})

Note:

- {documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET http://<serverName>:6405/biprws/raylight/vx/documents
- {dataproviderId}: The identifier of the dataprovider available for a document is retrieved by: GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders

Getting the details of a data provider

Gets details of a data provider for a Web Intelligence document specified by the URL parameter `documentId`. It is also used to update the data provider for the Web Intelligence document referenced by its ID.

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataproviderId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
<code>documentId</code>	Mandatory: A valid ID identifier of a Web Intelligence document
<code>dataproviderId</code>	Mandatory: A valid ID identifier of a data provider used in the Web Intelligence document.

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Details of the data provider:</p> <p><code>id</code>, <code>name</code>, the date and time of the last update, <code>ispartial</code>, the number of row and the objects dictionary. In <code>dictionary</code>, the object <code>id</code>, <code>type</code>, <code>name</code>, <code>description</code>, the data source object ID and the object qualification</p>

Example: To get data provider details**Note:**

Retrieves details of the dataprovider ID DP0 available for the document ID 7738

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""<TokenValue>\"" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dataprovider>
  <id>DP0</id>
  <name>Query 1</name>
  <dataSourceId>19689</dataSourceId>
  <updated>2012-07-31T15:44:25.000+02:00</updated>
  <duration>1</duration>
  <isPartial>false</isPartial>
  <rowCount>284</rowCount>
  <flowCount>1</flowCount>
  <dictionary>
    <expression qualification="Dimension" dataType="String">
      <id>DP0.DOa5</id>
      <name>Lines</name>
      <description>Product line. Each line contains a set of categories.</description>
      <dataSourceObjectId>DS0.DOa5</dataSourceObjectId>
    </expression>
    <expression qualification="Dimension" dataType="String">
      <id>DP0.D0da</id>
      <name>State</name>
      <description>State located.</description>
      <dataSourceObjectId>DS0.D0da</dataSourceObjectId>
    </expression>
    <expression qualification="Dimension" dataType="String">
      <id>DP0.D0bc</id>
      <name>Year</name>
      <description>Year 1999 - 2001.</description>
      <dataSourceObjectId>DS0.D0bc</dataSourceObjectId>
    </expression>
    <expression qualification="Measure" dataType="Numeric">
      <id>DP0.D093</id>
      <name>Sales revenue</name>
      <description>Sales revenue $ - $ revenue of SKU sold</description>
      <dataSourceObjectId>DS0.D093</dataSourceObjectId>
    </expression>
  </dictionary>
</dataprovider>
```

Updating a data provider

This method is used to purge a data provider.

Request:

```
PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataproviderId}
```

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. An integer. The ID of the document.
dataproviderId	Mandatory. An integer. The ID of the dataprovider.
purge	Optional. A boolean. Default: <code>false</code> . Indicates whether to purge the data provider or not.
purgeOptions	An optional string "prompts", to control the purge operation. If not set or empty, only the data provider will be purged.

Response

Body
<code><success> <message>The resource of type 'Data provider' with identifier 'DP2' has been successfully updated.</message> <id>DP2</id> <success></code>

Example: To purge a data provider, but not any prompts

This example updates (purges) DP0.

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "X-SAP-LogonToken:\""%tokenValue%" http://<server Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0?purge=true
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -X "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0?purge=true

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Data provider' with identifier 'DP0' has been successfully updated.</message>
  <id>DP0</id>
</success>
```

Example: To purge a data provider and prompts

This example updates (purges) DP0.

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H 'X-SAP-LogonToken:""%tokenValue%"' http://<server
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0?purge=true&purgeOptions=prompts
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -X "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0?purge=true&purgeOptions=prompts

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Data provider' with identifier 'DP0' has been successfully updated.</message>
  <id>DP0</id>
</success>
```

Example: To rename a data provider

This example refers to the xml file "rename.xml" to rename the data provider.

Body

```
<dataprovider>
  <name>My Renamed Data Provider</name>
</dataprovider>
```

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" -d "rename.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -X "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "rename.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355
```

```
<success>
  <message>The resource of type 'Data provider' with identifier 'DP0' has been successfully updated.</message>
  <id>DP0</id>
</success>
```

Removing a data provider from the list of data providers in a document

Request:

DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataproviderId}

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameters	Description
documentId	Integer. Mandatory. a valid identifier of a document
dataproviderId	Integer. Mandatory. a valid identifier of a data provider

Response:

Header	Value
Status Code	HTTP response code

Header	Value
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<pre><success> <message> The resource of type 'Data Provider' with identifier 1234ABCD has been successfully removed. </message> <id>1234ABCD</id> </success></pre>

Example: (XML format) Failed to remove the last data provider

Request

Windows

```
curl "DELETE" -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" http://<server Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl DELETE -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<server Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

Body (XML)
<pre><error> <error_code>101</error_code> <message>The resource of type 'Parameter' with identifier 'DP0' can not be removed.</message> </error></pre>

Example: (XML format) Remove the last data provider

Request

Windows

```
curl "DELETE" -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" http://<server Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl "DELETE" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 07 Jun 2012 08:23:37 GMT
Content-Type: application/xml
Content-Length: 217

<error>
  <error_code>101</error_code>
  <message>The resource of type 'Parameter' with identifier 'DP0' cannot be removed.</message>
</error>
```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [To log on to the BI platform](#)

3.5.4 Changing the data provider

Use the mappings URL to:

Get the mapping information: GET <URL>/documents/{documentId}/dataproviders/mappings

Change the data source: POST <URL>/documents/{documentId}/dataproviders/mappings

Note:

Changing data source when based on Analysis Views and Custom Data Provider.

Related Topics

- [Get the proposal mappings between a document's data providers and a given target data source](#)
- [Moving a data provider in a document](#)

3.5.4.1 Get the proposal mappings between a document's data providers and a given target data source

This URL gets the list of possible mappings.

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{originDataproviderIds}: The identifiers of the Web Intelligence document's data providers are retrieved in the document's data providers list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders`

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/mappings?originDataproviderIds={DP1Id},{DP2Id}&targetDataSourceId={DataSourceId}`

Note:

originDataproviderIds can be a list of data providers, each separated by a comma. If there is no originDataproviderIds specified, the request is run on all the existing data providers in the document.

Header	Value
Accept	To retrieve the details of a document: <code>text/xml</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID for the document.
originDataproviderIds	The IDs of the data providers to consider for the mapping, separated by a comma.
targetDataSourceId	A valid ID for a datasource to use for the proposal mapping.

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or
Content-Length	Length of content in the response body

Body (XML)
<p>Details of the data provider mappings, described as follows:</p> <pre><mappings> <content> <mapping status="Ok"/"Ambiguous"/"Not found">. The status can be "Ok" if the mapping fully matches, "Ambiguous" in case of uncertain mapping or "NotFound" if there is no possible match. The 'NotFound' status will remove from the document the corresponding id when doing the change- source. For each mapping, a source and corresponding (compatible) target: <source> <id>{sourceId}</id> The data source ID </source> <target> <id>{targetId}</id> The target ID </mapping> ... Other mappings ... </mappings></pre>

Example: To get the mapping for data provider DPO and datasource ID 9455

The example below shows how to get all the function descriptions, the example of returned body contains just two examples.

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDataProviderIds=DP0,DP1&targetDataSourceId=11414
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDataProviderIds=DP0,DP1&targetDataSourceId=11414
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 1800
```

```
<mappings>
  <content>
    <mapping status="Ok">
      <source>
        <id>DS0.D012</id>
      </source>
      <target>
        <id>DS1.D012</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D013</id>
      </source>
      <target>
        <id>DS1.D013</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D017</id>
      </source>
      <target>
        <id>DS1.D017</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D018</id>
      </source>
      <target>
        <id>DS1.D031</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D02</id>
      </source>
      <target>
        <id>DS1.D02</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D025</id>
      </source>
      <target>
        <id>DS1.D025</id>
      </target>
    </mapping>
  </content>
</mappings>
```

```

    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D04</id>
      </source>
      <target>
        <id>DS1.D04</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D06</id>
      </source>
      <target>
        <id>DS1.D06</id>
      </target>
    </mapping>
    <mapping status="Ok">
      <source>
        <id>DS0.D07</id>
      </source>
      <target>
        <id>DS1.D07</id>
      </target>
    </mapping>
    <mapping status="Ambiguous">
      <source>
        <id>DS0.D084</id>
      </source>
      <target>
        <id>DS1.D06</id>
      </target>
    </mapping>
  </content>
</mappings>

```

Example: 2: To get the mapping for a data provider DP0 when the target source is a BEx query

The example below shows how to get all the function descriptions when the target source is a BEx query ID= 10326;Z_BOBJ;AAQUERY_RESTRICT_KF.

Request:

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/10554/dataproviders/mappings?originDataProviderIds=DP0,DP1&targetDataSourceId=10326;Z_BOBJ;AAQUERY_RESTRICT_KF
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok"" http://<serverName>:6405/biprws/raylight/v1/documents/10554/dataproviders/mappings?originDataProviderIds=DP0,DP1&targetDataSourceId=10326;Z_BOBJ;AAQUERY_RESTRICT_KF
```

Response

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 1800

```



```

<mappings>
  <content>
    <mapping status="Ok">
      <source>
        <id>AZ_CITY</id>
      </source>
      <target>
        <id>AZ_CITY</id>
      </target>
    </mapping>
    <mapping status="Ambiguous">
      <source>
        <id>MA55OT46EO4K803Z77T003LNMU</id>
      </source>
      <target>
        <id>MD4NTN2159JU9ONUHTLMNBW3U</id>
      </target>
    </mapping>
  </content>
</mappings>

```

3.5.4.2 Updating the mappings of a document's data providers and a given target data source

Changes the data source using a given mapping. You can use the proposal mappings or you can declare custom mappings. For custom mappings you can change only the target id of a given mapping, but must keep all the source ids. If the target id is null, then the data source object will be removed.

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Note:

{originDataproviderIds}: The identifiers of the Web Intelligence document's data providers are retrieved in the document's data providers list by: GET <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders>

Request URL

POST <http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/mappings?originDataproviderIds={DP1Id},{DP2Id}&targetDataSourceId={DataSourceId}>

Header	Value
Accept	To retrieve the details of a document:text/xml

Header	Value
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	A valid ID for the document.
originDataproviderIds	The IDs of the data providers to consider for the mapping, separated by a comma.
targetDatasourceId	A valid ID for a datasource to use for the proposal mapping.

Body (XML)

Body (XML)

The changesource mapping, including parameters (responses to prompts) when necessary:

```
<mappings>
  <content>
    . . . [mappings]
  </content>
  <parameters>
    . . . [parameters]
  </parameters>
</mappings>
```

Example mapping:

```
<mappings>
  <content>
    <mapping>
      <source>
        <id>DS0.D084</id>
      </source>
      <target>
        <id>DS1.D06</id>
      </target>
    </mapping>
  </content>
```

Example parameters that follow after the mapping information:

```
<parameters>
  <parameter optional="false" type="context">
    <id>0</id>
    <name>Select a context</name>
    <answer constrained="true" type="Text">
      <values>
        <value id="2">Reservations</value>
      </values>
    </answer>
```

Body (XML)
<pre></parameter> </parameters> </mappings></pre>

Reponse:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml
Content-Length	Length of content in the response body

Body (XML)
<p>In case of success:</p> <pre><success> <message>The resource of type 'Document' with identifier 'documentId' has been successfully updated.</message> <id>{documentId}</id> </success></pre>

Example: To update the mapping for a data provider DPO and datasource ID 9455

The example below shows how to update the mappings for document ID 9455.

Body (XML) "mappings1.xml"

The changesource mapping, there are no parameters, and all sources are mapped to a target:

```
<mappings>
  <content>
    <mapping>
      <source>
        <id>DS0.DO12</id>
      </source>
      <target>
        <id>DS1.DO12</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>DS0.DO2</id>
      </source>
      <target>
        <id>DS1.DO2</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>DS0.DO6</id>
      </source>
      <target>
        <id>DS1.DO39</id>
      </target>
    </mapping>
  </content>
</mappings>
```

Request:**Windows**

```
curl -POST -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@mappings1.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -POST -i -H "accept:application/xml" -H "X-SAP-LogonToken:""$logtok" -d "@mappings1.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

Response:

```
<success>
  <message>The resource of type 'Document' with identifier '9455' has been successfully updated.</message>
  <id>9455</id>
</success>
```

Example: Mapping file where one source is not mapped to a target**Body (XML) "mappings2.xml"**

The changesource mapping, there are no parameters, and the last source (DS0.DO6) is not mapped to a target; it will be removed from the mapping:

```
<mappings>
  <content>
    <mapping>
      <source>
        <id>DS0.DO12</id>
      </source>
      <target>
        <id>DS1.DO12</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>DS0.DO2</id>
      </source>
      <target>
        <id>DS1.DO2</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>DS0.DO6</id>
      </source>
      <target/>
    </mapping>
  </content>
</mappings>
```

Request:**Windows**

```
curl -POST -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%" -d "@mappings2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -POST -i -H "accept:application/xml" -H 'X-SAP-LogonToken:$logtok' -d "@mappings2.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

Example: To declare mapping that includes parameters

When the document contains parameters, they will appear in the response as long as parameters are still needed (to be filled in the request body). When all the parameters have values, then the response will be successful.

Request Body (XML) "mappings3.xml"

The changesource mapping:

```
<mappings>
  <content>
    <mapping>
      <source>
        <id>_Vy_yePzREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Product].[Size]</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>_Vy_ye_zREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Employee].[Birth Date]</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>_Vy_yfPzREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Reseller].[Bank Name]</id>
      </target>
    </mapping>
  </content>
</mappings>
```

Request:

Windows

```
curl -POST -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%"" -d "@mappings3.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -POST -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@mappings3.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

Response

Parameters (responses to prompts are required), so the response to this request returns the mapping information and the parameter requests:

Response Body (XML)

The changesource mapping:

```
<mappings>
  <content>
    <mapping>
      <source>
        <id>_Vy_yePzREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Product].[Size]</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>_Vy_ye_zREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Employee].[Birth Date]</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>_Vy_yfPzREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Reseller].[Bank Name]</id>
      </target>
    </mapping>
  </content>
  <parameters>
    <parameter optional="false" type="prompt" dpId="DP1">
      <id>0</id>
      <technicalName>psEnter Age:</technicalName>
      <name>Enter Age:</name>
      <answer constrained="false" type="Text">
        <info cardinality="Single"/>
      </answer>
    </parameter>
  </parameters>
</mappings>
```

Resend the request with the parameter duly provided:

Request Body (XML) "mappings3.xml" with parameters added

The changesource mapping and parameter:

```
<mappings>
  <content>
    <mapping>
      <source>
        <id>_Vy_yePzREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Product].[Size]</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>_Vy_ye_zREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Employee].[Birth Date]</id>
      </target>
    </mapping>
    <mapping>
      <source>
        <id>_Vy_yfPzREeG4Q-3y4CsnNg</id>
      </source>
      <target>
        <id>A[Reseller].[Bank Name]</id>
      </target>
    </mapping>
  </content>
  <parameters>
    <parameter>
      <id>0</id>
      <answer>
        <values>
          <value id="1">25</value>
        </values>
      </answer>
    </parameter>
  </parameters>
</mappings>
```

Request:

Windows

```
curl -POST -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%" -d "@mappings3.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -POST -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@mappings3.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/9460/dataproviders/mappings?originDat
aproviderIds=DP0,DP1&targetDatasourceId=11414
```

Response:

```
<success>
  <message>The resource of type 'Document' with identifier '9460' has been successfully updated.</message>
  <id>9460</id>
</success>
```

3.5.5 Flows: getting the flows count for a data provider

Gets the flows count for a data provider.

Request:

GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataproviderId}/flows/count

- {documentId}: The identifier of the Web Intelligence document retrieved in the document list by:
GET http://<serverName>:6405/biprws/raylight/vx/documents
- {dataproviderId}: The identifier of the dataprovider available for a document is retrieved by:
GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders

Header	Value
Accept	text/plain
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
dataproviderId	Mandatory. Integer. The identifier of the dataprovider.

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	text/plain
Content-Length	Length of content in the response body

Body
The number of flows as an integer (from 1 to n).

Example: To get the flows count for a data provider

Note:

Retrieves the flows count for data provider ID DP0 available for the document ID 7738. Returns 1.

Windows

```
curl -G -i -H "accept:text/plain" -H X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0/flows/count
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:text/plain" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0/flows/count
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 07 Jun 2012 08:25:56 GMT
Content-Type: text/plain
Content-Length: 1

1
```

Related Topics

- [To log on to the BI platform](#)
- [Getting a list of data providers or adding a data provider](#)
- [Document: retrieving, copying, or creating](#)

3.5.6 Flows: getting the details of a flow in CSV or XML format

Gets in XML or CSV format, details on the data provider flow specified by the URL parameter `flowIndex`.

Note:

- `{documentId}`: The identifier of the Web Intelligence document is retrieved in the documents list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`.
- `{dataprovderId}`: The identifier of the dataprovder available for a document is retrieved by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders`
- `{flowIndex}`: The number of a dataprovder flow available for one document is retrieved by GET `http://<serverName>:6405/biprws/raylight/vx/documents/documentId/dataproviders/dataprovderId/flow/count`. The first index position is: 0.

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataprovderId}/flows/{flowIndex}`

Header	Value
Accept	text/xml or text/plain
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
dataprovderId	Mandatory. Integer. The identifier of the dataprovder.
flowId	Mandatory. Integer. The identifier of the flow.

Response:

Header	Value
Status Code	HTTP response code
Content-Type	text/xml or text/plain

Header	Value
Content-Length	Length of content in the response body

Body
<p>For one data provider retrieves the flow and the detail flow.</p> <p>In XML format, the index cells and their values by row.</p> <p>In CSV format, values only.</p>

Example: Retrieve values of a data provider flow in plain text format

Note:

Retrieves the first flow (flowIndex 0) of the data provider DP0 available for the document ID 7744.

Windows

```
curl -G -i -H "accept:text/plain" -H "X-SAP-LogonToken: ""tokenValue"" " http://<serverName>:6405/biprws/raylight/v1/documents/7744/dataproviders/DP0/flows/0
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:text/plain" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/7744/dataproviders/DP0/flows/0
```

Response (CSV format)

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 07 Jun 2012 12:10:05 GMT
Content-Type: text/plain
Content-Length: 1026

"Year";"State";"Sales revenue";"Margin"
"2001";"California";"1704210.8";"774893.4"
"2001";"Colorado";"448301.5";"203700.6"
"2001";"DC";"693210.5";"310356.2"
"2001";"Florida";"405985.1";"192479.3"
"2001";"Illinois";"737914.2";"348749.8"
"2002";"California";"2782679.5";"1076528"
"2002";"Colorado";"768389.5";"294482.6"
"2002";"DC";"1215158";"457230.6"
"2003";"California";"2992679";"1121488.5"
...
```

Response (XML format)

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Thu, 07 Jun 2012 12:10:05 GMT
Content-Type: text/plain
Content-Length: 1026

<DATA_PROVIDERS>
```

```

<DATA_PROVIDER>
<ROW>
<CELL INDEX="0">2006</CELL>
<CELL INDEX="1">12</CELL>
<CELL INDEX="2">Texas </CELL>
<CELL INDEX="3">Dallas</CELL>
<CELL INDEX="4">e-Fashion Dallas</CELL>
<CELL INDEX="5">43302.1</CELL>
<CELL INDEX="6">222</CELL>
<CELL INDEX="7">17829.4</CELL>
</ROW>
<ROW>
<CELL INDEX="0">2006</CELL>
<CELL INDEX="1">12</CELL>
<CELL INDEX="2">Texas </CELL>
<CELL INDEX="3">Houston</CELL>
<CELL INDEX="4">e-Fashion Houston</CELL>
<CELL INDEX="5">55454.6</CELL>
<CELL INDEX="6">258</CELL>
<CELL INDEX="7">24614.6</CELL>
</ROW>
<ROW>
<CELL INDEX="0">2006</CELL>
<CELL INDEX="1">12</CELL>
<CELL INDEX="2">Texas </CELL>
<CELL INDEX="3">Houston</CELL>
<CELL INDEX="4">e-Fashion Houston Leighton</CELL>
<CELL INDEX="5">77237.7</CELL>
<CELL INDEX="6">366</CELL>
<CELL INDEX="7">34926.7</CELL>
</ROW>
</DATA_PROVIDER>
</DATA_PROVIDERS>

"Year";"State";"Sales revenue";"Margin"
"2001";"California";"1704210.8";"774893.4"
"2001";"Colorado";"448301.5";"203700.6"
"2001";"DC";"693210.5";"310356.2"
"2001";"Florida";"405985.1";"192479.3"
"2001";"Illinois";"737914.2";"348749.8"
"2002";"California";"2782679.5";"1076528"
"2002";"Colorado";"768389.5";"294482.6"
"2002";"DC";"1215158";"457230.6"
"2003";"California";"2992679";"1121488.5"
...

```

Example: Retrieve values of a data provider flow in XML format

Note:

Retrieves the first flow (flowIndex 0) of the data provider DP0 available for the document ID 12575.

Windows

```
curl -G -H "accept:text/xml" -H X-SAP-LogonToken:""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/12575/dataproviders/DP0/flows/0
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -H "accept:text/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/12575/dataproviders/DP0/flows/0
```

Response

```

<DATA_PROVIDERS UID="20916|0|doc_basic|wid" FORCE_REFRESH="FALSE" DATA_PROVIDER="" FORMATTED="FALSE" REFRESH="FALSE" >
  <DATA_PROVIDER NAME="Query 1" SOURCE="eFashion-raylight" DATE="2012/07/31 13:44:25" DURATION="1" CUBE="0">

```

```

<COLUMN DSID="DS0.DObc" DPID="DP0.DObc" INDEX="0" TYPE="string">Year</COLUMN>
<COLUMN DSID="DS0.DOda" DPID="DP0.DOda" INDEX="1" TYPE="string">State</COLUMN>
<COLUMN DSID="DS0.DOa5" DPID="DP0.DOa5" INDEX="2" TYPE="string">Lines</COLUMN>
<COLUMN DSID="DS0.DO93" DPID="DP0.DO93" INDEX="3" TYPE="double">Sales revenue</COLUMN>
<ROW>
  <CELL INDEX="0">2004</CELL>
  <CELL INDEX="1">California</CELL>
  <CELL INDEX="2">Accessories</CELL>
  <CELL INDEX="3">489665.9</CELL>
</ROW>
<ROW>
  <CELL INDEX="0">2004</CELL>
  <CELL INDEX="1">California</CELL>
  <CELL INDEX="2">City Skirts</CELL>
  <CELL INDEX="3">11072.2</CELL>
</ROW>
<ROW>
  <CELL INDEX="0">2004</CELL>
  <CELL INDEX="1">California</CELL>
  <CELL INDEX="2">City Trousers</CELL>
  <CELL INDEX="3">10935.1</CELL>
</ROW>
<ROW>
  ...
</ROW>
<ROW>
  ...
</ROW>
<DATA_PROVIDER/>
<DATA_PROVIDERS/>

```

Related Topics

- [Document: retrieving, copying, or creating](#)
- [Getting a list of data providers or adding a data provider](#)
- [Flows: getting the flows count for a data provider](#)
- [To log on to the BI platform](#)

3.5.7 Query specification: getting and updating the query specification

Use this URL to:

- Return the query specification attached to a given data provider (GET `<url>/documents/{documentId}/dataprovers/{dataproversId}/specification`).
- Update the query specification attached to a given data provider PUT `<url>/documents/{documentId}/dataprovers/{dataproversId}/specification`).

Note:

{documentId}: The identifier of the Web Intelligence document retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Note:

{dataproversId}: The identifier of the Web Intelligence document's data provider retrieved in the document's data providers list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataprovers`

Getting the query specification for a data provider

Return the query specification attached to a data provider.

Request URL

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataproviderId}/specification`

Notes

Header	Value
Accept	text/xml
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
dataproviderId	Mandatory. Integer. The identifier of the data provider.

Response:

Header	Value
Status Code	HTTP response code
Content-Type	text/xml
Content-Length	Length of content in the response body

Body (XML)

Details of the document functions available from the formula engine. Each function is described as follows:

```
<queryspec:QuerySpec xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:queryspec="http://com.sap.sl.queryspec" dataProviderId="DP0">

<queryParameters>

<duplicatedRowsProperty activated="true" value="true"/> <maxRetrievalTimeIn
SecondsProperty value="300"/>

<maxRowsRetrievedProperty value="90000"/>

<removeEmptyRowsProperty activated="true" value="true"/>

<allowOtherUserToEditQueryProperty activated="true" value="true"/>

<resetContextOnRefreshProperty activated="true" value="true"/>

<stripQueryProperty/>

</queryParameters>

<queriesTree xsi:type="queryspec:QueryOperatorNode" queryOperator="Union">
<children xsi:type="queryspec:QueryDataNode">

<bOQuery name="Query" identifier="_1y8aENsVEeGswMB7H6mlQw">

<resultObjects identifier="DS0.DObc" name="Year"/>

<resultObjects identifier="DS0.DOda" name="State"/>

<resultObjects identifier="DS0.DOa5" name="Lines"/>

<resultObjects identifier="DS0.DO93" name="Sales revenue"/>

<conditionPart/>

</bOQuery>

</children>

</queriesTree>

<propertyBag key="DUPLICATED_ROWS_UNDEFINED_VALUE" value="-1"/>

<propertyBag key="RESET_CONTEXT_ON_REFRESH_UNDEFINED_VALUE" value="-1"/>

<propertyBag key="ALLOW_THE_USER_TO_EDIT_QUERY_UNDEFINED_VALUE" value="-
1"/>

</queryspec:QuerySpec>
```

Example: To get the query specification of data provider ID DP0 for document ID 7738**Note:**

The data provider ID is DP0, the document ID is 7738

Windows

```
curl -G -i -H "accept:text/xml" -H "X-SAP-LogonToken:""%"tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0/specification
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:text/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0/specification
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<queryspec:QuerySpec xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:queryspec="http://com.sap.sl.queryspec" dataProviderId="DP0">
  <queryParameters>
    <duplicatedRowsProperty activated="true" value="true"/>
    <maxRetrievalTimeInSecondsProperty value="300"/>
    <maxRowsRetrievedProperty value="90000"/>
    <removeEmptyRowsProperty activated="true" value="true"/>
    <allowOtherUserToEditQueryProperty activated="true" value="true"/>
    <resetContextOnRefreshProperty activated="true" value="true"/>
    <stripQueryProperty/>
  </queryParameters>
  <queriesTree xsi:type="queryspec:QueryOperatorNode" queryOperator="Union">
    <children xsi:type="queryspec:QueryDataNode">
      <bOQuery name="Query" identifier="_ly8aENsVEeGswMB7H6mlQw">
        <resultObjects identifier="DS0.D0bc" name="Year"/>
        <resultObjects identifier="DS0.D0da" name="State"/>
        <resultObjects identifier="DS0.D0a5" name="Lines"/>
        <resultObjects identifier="DS0.D093" name="Sales revenue"/>
        <conditionPart/>
      </bOQuery>
    </children>
  </queriesTree>
  <propertyBag key="DUPLICATED_ROWS_UNDEFINED_VALUE" value="-1"/>
  <propertyBag key="RESET_CONTEXT_ON_REFRESH_UNDEFINED_VALUE" value="-1"/>
  <propertyBag key="ALLOW_THE_USER_TO_EDIT_QUERY_UNDEFINED_VALUE" value="-1"/>
</queryspec:QuerySpec>
```

Updating the query specification for a data provider

Updates the query specification attached to a data provider.

Request PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/dataproviders/{dataProviderId}/specification

Header	Value
Accept	To retrieve the details of a document: <code>text/xml</code>
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
dataproviderId	Mandatory. Integer. The identifier of the dataprovider.

Body (XML)

Details of the updated properties for the data provider:

```
<queryParameters>
  <duplicatedRowsProperty activated="true" value="true"/>
  <maxRetrievalTimeInSecondsProperty value="300"/>
  <maxRowsRetrievedProperty value="90000"/>
  <removeEmptyRowsProperty activated="true" value="true"/>
  <allowOtherUserToEditQueryProperty activated="true" value="true"/>
  <resetContextOnRefreshProperty activated="true" value="true"/>
  <stripQueryProperty/>
</queryParameters>
<queriesTree xsi:type="queryspec:QueryOperatorNode" queryOperator="Union">
  <children xsi:type="queryspec:QueryDataNode">
    <bOQuery name="Query" identifier="_1y8aENsVEeGswMB7H6m1Qw">
      <resultObjects identifier="DS0.DObc" name="Year"/>
      <resultObjects identifier="DS0.DOda" name="State"/>
      <resultObjects identifier="DS0.DOa5" name="Lines"/>
      <resultObjects identifier="DS0.DO93" name="Sales revenue"/>
      <conditionPart/>
    </bOQuery>
  </children>
</queriesTree>
<propertyBag key="DUPLICATED_ROWS_UNDEFINED_VALUE" value="-1"/>
<propertyBag key="RESET_CONTEXT_ON_REFRESH_UNDEFINED_VALUE" value="-1"/>
<propertyBag key="ALLOW_THE_USER_TO_EDIT_QUERY_UNDEFINED_VALUE" value="-1"/></queryspec:QuerySpec>
```

Response:

Header	Value
Status Code	HTTP response code
Content-Type	text/xml
Content-Length	Length of content in the response body

Body (XML)

In case of success:

```
<success> <message>The resource of type 'Data provider' with identifier
'DP0' has been successfully updated.</message> <id>DP0</id></success>
```

Otherwise: <error> ... </error>

Example: To update the query specification for data provider ID DP0 for document ID 7738

Write the query specifications that you want to update in an xml file, for example; specifications.xml:

Body (XML) "specifications.xml"

```
<queryspec:QuerySpec xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:queryspec="http://com.sap.sl.queryspec" dataProviderId="DP0">
<queryParameters>
  <duplicatedRowsProperty activated="true" value="true"/>
  <maxRetrievalTimeInSecondsProperty value="300"/>
  <maxRowsRetrievedProperty value="90000"/>
  <removeEmptyRowsProperty activated="true" value="true"/>
  <allowOtherUserToEditQueryProperty activated="true" value="true"/>
  <resetContextOnRefreshProperty activated="true" value="true"/>
  <stripQueryProperty/>
</queryParameters>
<queriesTree xsi:type="queryspec:QueryOperatorNode" queryOperator="Union">
  <children xsi:type="queryspec:QueryDataNode">
    <bOQuery name="Query" identifier="_ly8aENsVEeGswMB7H6mlQw">
      <resultObjects identifier="DS0.DObc" name="Year"/>
      <resultObjects identifier="DS0.Doda" name="State"/>
      <resultObjects identifier="DS0.D0a5" name="Lines"/>
      <resultObjects identifier="DS0.D093" name="Sales revenue"/>
      <conditionPart/>
    </bOQuery>
  </children>
</queriesTree>
<propertyBag key="DUPLICATED_ROWS_UNDEFINED_VALUE" value="-1"/>
<propertyBag key="RESET_CONTEXT_ON_REFRESH_UNDEFINED_VALUE" value="-1"/>
<propertyBag key="ALLOW_THE_USER_TO_EDIT_QUERY_UNDEFINED_VALUE" value="-1"/>
</queryspec:QuerySpec>
```

Windows

```
curl -PUT -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@specifications.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0/specification
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -PUT -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" -d "@specifications.xml"
http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0/specification
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'Data provider' with identifier 'DP0' has been successfully updated.</message>
  <id="DP0"></id>
</success>
```

3.6 Managing BW connections and BEx queries

This section describes the main operations available for managing BW connections and BEx queries:

- Getting the list of BW connections GET <URL>/bwconnections
- Getting the details of a BW connection GET <URL>/bwconnections/{BwConnectionId}
- Browsing the details of a BW connection PUT <URL>/bwconnections/{BwConnectionId}
- Getting the outline of a BEx query PUT <URL>/bwconnections/{BwConnectionId}/outline
- Getting the capabilities of a BEx query PUT <URL>/bwconnections/{BwConnectionId}/capabilities

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

3.6.1 Getting the list of BW connections

Getting a the list of BW connections

This request returns the list of available Business Warehouse connections. You can only see the connections that you have authorization to see.

Request:

GET http://<serverName>:6405/biprws/raylight/vx/bwconnections

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
limit	Integer. The maximum number of connections to return. Default = 10
offset	Offset from the beginning of the list. Default value = 0

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
Provides the list of BW connections that you have the authorization to see/access.

Example: To get the list of BW connections**Note:**

Retrieves the list of available BW connections.

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:""%tokenValue%"%" http://<serverName>:6405/biprws/raylight/v1/bwconnections
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/bwconnections
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<bwconnections>
  <bwconnection type="Cube">
    <id>7052</id>
    <cuid>AdDDU67.DyxBkOgpzjDjN30</cuid>
    <name>Adventure Works MSAS2005</name>
    <folderId>7131</folderId>
  </bwconnection>
  <bwconnection type="Cube">
    <id>11540</id>
    <cuid>AbUAJD7zVpZFgy2jQQNMyI8</cuid>
    <name>AdventureWorks</name>
    <folderId>11484</folderId>
  </bwconnection>
  <bwconnection type="Cube">
    <id>11537</id>
    <cuid>AROnvrBXn1tOpsXj_jPPtYo</cuid>
    <name>ADW</name>
    <folderId>11484</folderId>
  </bwconnection>
  <bwconnection type="Query">
    <id>11487</id>
    <cuid>AZSaRdG.bLFAhGKZrHR_Evc</cuid>
    <name>bex_all_dim</name>
    <folderId>11484</folderId>
  </bwconnection>
  <bwconnection type="Query">
    <id>11489</id>
    <cuid>AXRu2fNiQphAtF4041J4OKg</cuid>
    <name>bex_simple</name>
    <folderId>11484</folderId>
  </bwconnection>
  <bwconnection type="Cube">
    <id>7249</id>
    <cuid>Adi_9N.gvR5MjotdV5akMkQ</cuid>
    <name>BICS_Cube</name>
    <folderId>4066</folderId>
  </bwconnection>
  <bwconnection type="Query">
    <id>7268</id>
    <cuid>AUeWmURZfzVKjnpwfTWWxvM</cuid>
    <name>BICS_Query</name>
    <folderId>4066</folderId>
  </bwconnection>
  <bwconnection type="System">
    <id>7039</id>
    <cuid>AY2UQVy3WRLioC8GBg5Vi0</cuid>
    <name>BICS_Server</name>
    <folderId>4066</folderId>
  </bwconnection>
</bwconnections>
```

3.6.2 Getting the details of a BW connection by ID

Getting the details of a BW connection

This request returns the details of a Business Warehouse connection.

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/bwconnections/{bwConnectionId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
bwConnectionId	Mandatory. The ID of the BW connection.

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Provides the details of a BW connection that you have the authorization to see/access.

```
<bwconnection type="Query">
  <id>11489</id>
  <cuid>AXRu2fNiQphAtF404lJ4OKg</cuid>
  <name>bex_simple</name>
  <folderId>11484</folderId>
  <bwnodes>
    <bwnode type="Query">
      <id>11489</id>
      <name>A simple Bex Query</name>
      <technicalName>QRY_SIMPLE</technicalName>
      <path>/QRY_SIMPLE</path>
    </bwnode>
  </bwnodes>
</bwconnection>
```

Example: To browse the details of a BW connection (of type query)**Note:**

Retrieves the list of available BW connections.

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" http://<server
Name>:6405/biprws/raylight/v1/bwconnections/{bwConnectionId}
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/ray
light/v1/bwconnections/{bwConnectionId}
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<bwconnection type="Query">
  <id>11489</id>
  <cuid>AXRu2fNiQphAtF404lJ4OKg</cuid>
  <name>bex_simple</name>
  <folderId>11484</folderId>
  <bwnodes>
    <bwnode type="Query">
      <id>11489</id>
      <name>A simple Bex Query</name>
      <technicalName>QRY_SIMPLE</technicalName>
      <path>/QRY_SIMPLE</path>
    </bwnode>
  </bwnodes>
</bwconnection>
```

Example: To browse the details of an incomplete BW connection (type System or Cube)**Note:**

Retrieves the list of available BW connections. For an incomplete BW connection, we have to browse the system/cube and find a bwnode of type Query (BEx query) which will complete the connection.

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/bwconnections/{bwConnectionId}
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: '$logtok'" http://<serverName>:6405/biprws/raylight/v1/bwconnections/{bwConnectionId}
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<bwconnection type="System">
  <id>11990</id>
  <cuid>Aaj0N_I.bSJElYDKk08sxTU</cuid>
  <name>raylight_BOF</name>
  <folderId>11484</folderId>
  <bwnodes>
    <bwnode type="Favorites">
      <name>Favorites</name>
      <technicalName>SystemFavoritesTopLevel</technicalName>
      <path>/SystemFavoritesTopLevel</path>
    </bwnode>
    <bwnode type="InfoArea">
      <name>InfoArea</name>
      <technicalName>SystemInfoareaTopLevel</technicalName>
      <path>/SystemInfoareaTopLevel</path>
    </bwnode>
  </bwnodes>
</bwconnection>
```

3.6.3 Browsing the details of a BW connection

Browsing the details of a BW connections

This request returns the details of a Business Warehouse connection. This URL allows you to browse the detail of an incomplete BW connection, referenced by its ID, and choose a BEx query which will complete the connection. Two browse methods are available: by path and find by pattern.

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/bwconnections/{BwConnectionId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
BwConnectionId	Mandatory. The ID of the BW connection.

Request body (XML)
<p>Define the path or search pattern.</p> <p>For a path:</p> <pre><bwnode> <path>[path/to/the/required/node]</path> </bwnode></pre> <p>For a search pattern:</p> <pre><bwnode> <pattern>[pattern to be used for the search]</pattern> </bwnode></pre>

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Provides the list of BW connections that you have the authorization to see/access.

Example: 1: To get the list of BEx queries inside a BW connection, browsing by path**Note:**

Retrieves the list of BEx queries inside a BW connection.

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" -d "@browsepath"
http://<serverName>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@browsepath" http://<server
Name>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}
```

Request body (XML) "browsepath.xml"

```
<bwnode>
  <path>/SystemInfoareaTopLevel/BUSINESSOBJECTS_QA/Z_BOBJ/TEST_DATE</path>
</bwnode>
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<bwconnection type="System">
  <id>11990</id>
  <cuid>Aaj0N_I.bsJElyDKk08sxTU</cuid>
  <name>raylight_BOF</name>
  <folderId>11484</folderId>
  <bwnodes>
    <bwnode type="Query">
      <id>11990;TEST_DATE</id>
      <name>test_data</name>
      <technicalName>TEST_DATE</technicalName>
      <mdxCompliant>true</mdxCompliant>
      <path>/SystemInfoareaTopLevel/BUSINESSOBJECTS_QA/Z_BOBJ/TEST_DATE</path>
    </bwnode>
  </bwnodes>
</bwconnection>
```

Example: 2: To get the list of BEx queries inside a BW connection, browsing by pattern**Note:**

Retrieves the list of BEx queries inside a BW connection matching a pattern.

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@browsepattern"
http://<serverName>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" -d "@browsepattern"
http://<serverName>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}
```

Request body (XML) "browsepattern.xml"

```
<bwnode>
  <pattern>*BOBJ*</pattern>
</bwnode>
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<bwconnection type="System">
  <id>11990</id>
  <cuid>AajON_I.bSJElYDKk08sxTU</cuid>
  <name>raylight_BOF</name>
  <folderId>11484</folderId>
  <bwnodes>
    <bwnode type="Query">
      <id>11990;Z_BOBJ;BOBJ_TEST_ROLE</id>
      <name>BOBJ_TEST_ROLE</name>
      <technicalName>BOBJ_TEST_ROLE</technicalName>
      <mdxCompliant>>false</mdxCompliant>
    </bwnode>
    <bwnode type="Query">
      <id>11990;Z_BOBJ;IE_ZBOBJ_COUNTRYVAR</id>
      <name>IE_ZBOBJ_COUNTRYVAR</name>
      <technicalName>IE_ZBOBJ_COUNTRYVAR</technicalName>
      <mdxCompliant>>true</mdxCompliant>
    </bwnode>
    <bwnode type="Query">
      <id>11990;T_BOBJ;ZCR_BOBJ_ALLF_MHIER_FCHAR_1</id>
      <name>ZCR_BOBJ_all_fields_mult_hier_free_chars</name>
      <technicalName>ZCR_BOBJ_ALLF_MHIER_FCHAR_1</technicalName>
      <mdxCompliant>>false</mdxCompliant>
    </bwnode>
    <bwnode type="Query">
      <id>11990;Z_BOBJ;Z_BOBJ_HB_SIMPLE</id>
      <name>Z_BOBJ_HB_simple</name>
      <technicalName>Z_BOBJ_HB_SIMPLE</technicalName>
      <mdxCompliant>>true</mdxCompliant>
    </bwnode>
  </bwnodes>
</bwconnection>
```

Note:

- A BEx query's ID is defined by the concatenation of the following, separated by ";" : the BW connection's ID and the technical name of the BEx query.
- Only MDX compliant BEx queries are usable in Web Intelligence workflows.

3.6.4 Getting the outline of a BEx query

Getting the outline of a BEx query

This request returns the details of a Business Warehouse connection.

Request:

PUT `http://<serverName>:6405/biprws/raylight/vx/bwconnections/{bwConnectionId}/outline`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
bwConnectionId	Mandatory. The ID of the BW connection.

Optional request body (XML)
<pre><bwnode> <id>The ID of the BEx query</id> </bwnode></pre>

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json

Header	Value
Content-Length	Length of content in the response body

Body (XML)

Provides the outline of a BEx query that you have the authorization to see/access.

```

<outline>
  <item type="Dimension">
    <name>Country</name>
    <description>Country</description>
    <item type="Hierarchy">
      <name>Country</name>
      <description>Country</description>
      <id>HZ_COUNTRY</id>
    </item>
    <item type="Hierarchy">
      <name>Country Hierarchy 01</name>
      <description>Country Hierarchy 01</description>
      <item type="Level">
        <name>Level 01</name>
        <description>Level 01</description>
        <id>LCOUNTRY_HIERARCHY_01|Z_COUNTRY.#1</id>
      </item>
      <item type="Level">
        <name>Level 02</name>
        <description>Level 02</description>
        <id>LCOUNTRY_HIERARCHY_01|Z_COUNTRY.#2</id>
      </item>
      <item type="Level">
        <name>Level 03</name>
        <description>Level 03</description>
        <id>LCOUNTRY_HIERARCHY_01|Z_COUNTRY.#3</id>
      </item>
      <id>HCOUNTRY_HIERARCHY_01|Z_COUNTRY</id>
    </item>
    . . .
  <item type="Measure">
    <name>Voyager Training 01 - Restricted KF</name>
    <description>Voyager Training 01 - Restricted KF</description>
    <item type="Attribute">
      <name>Voyager Training 01 - Restricted KF Currency</name>
      <id>MD4NUM119ATNWC7USQ4XYS0QY.Currency</id>
    </item>
    <item type="Attribute">
      <name>Voyager Training 01 - Restricted KF Formatted Value</name>
      <id>MD4NUM119ATNWC7USQ4XYS0QY.FormattedValue</id>
    </item>
    <id>MD4NUM119ATNWC7USQ4XYS0QY</id>
  </item>
</outline>

```

Example: To get the outline of a BEx query

Note:

Retrieves the list of available BW connections.

Request Body:**Request body (XML) bexqueryid.xml**

```
<bwnode>
  <id>The ID of the BEx query</id>
</bwnode>
```

```
<bwnode>
<id>11990;ROLE_ST_BEX5</id>
</bwnode>
```

Windows

```
curl "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:""%tokenValue%""' -d "@bexqueryid.xml"
http://<serverName>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}/outline
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@bexqueryid.xml"
http://<serverName>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}/outline
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<outline>
  <item type="Dimension">
    <name>Country</name>
    <description>Country</description>
    <item type="Hierarchy">
      <name>Country</name>
      <description>Country</description>
      <id>HZ_COUNTRY</id>
    </item>
    <item type="Hierarchy">
      <name>Country Hierarchy 01</name>
      <description>Country Hierarchy 01</description>
      <item type="Level">
        <name>Level 01</name>
        <description>Level 01</description>
        <id>LCOUNTRY_HIERARCHY_01|Z_COUNTRY.#1</id>
      </item>
      <item type="Level">
        <name>Level 02</name>
        <description>Level 02</description>
        <id>LCOUNTRY_HIERARCHY_01|Z_COUNTRY.#2</id>
      </item>
      <item type="Level">
        <name>Level 03</name>
        <description>Level 03</description>
        <id>LCOUNTRY_HIERARCHY_01|Z_COUNTRY.#3</id>
      </item>
      <id>HCOUNTRY_HIERARCHY_01|Z_COUNTRY</id>
    </item>
    . . .
```

```

<item type="Measure">
  <name>Voyager Training 01 - Restricted KF</name>
  <description>Voyager Training 01 - Restricted KF</description>
  <item type="Attribute">
    <name>Voyager Training 01 - Restricted KF Currency</name>
    <id>MD4NUM119ATNWC7USQ4XYSQA0QY.Currency</id>
  </item>
  <item type="Attribute">
    <name>Voyager Training 01 - Restricted KF Formatted Value</name>
    <id>MD4NUM119ATNWC7USQ4XYSQA0QY.FormattedValue</id>
  </item>
  <id>MD4NUM119ATNWC7USQ4XYSQA0QY</id>
</item>
</outline>

```

3.6.5 Getting the capabilities of a BEx query

Getting the capabilities of a BEx query

This request returns the capabilities of a BEx query.

Request:

PUT http://<serverName>:6405/biprws/raylight/vx/bwconnections/{bwConnectionId}/capabilities

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
bwConnectionId	Mandatory. The ID of the BW connection.

Optional request body (XML)

```

<bwnode>
  <id>11990;Z_BOBJ;ROLE_ST_BEX5</id>
</bwnode>

```

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

Provides the capabilities of a BEx query that you have the authorization to see/access.

```
<datasource:QueryCapability xmlns:datasource="http://com.sap.sl.datasource">
  <generalCapability customQueryScriptSupported="false" showHideScopeSupported="false"/>
  <dataProcessingCapability removeEmptyRowsAvailable="true" maxRetrievalTimeAvailable="false" queryStrip
pingAvailable="true"/>
  <filterCapability resultHierarchyInFilterSupported="false" constantComparisonSupported="true" hierarchy
ConstantOperandSupported="true">
    <supportedComparisonOperators>equal</supportedComparisonOperators>
    <supportedComparisonOperators>notEqual</supportedComparisonOperators>
    <supportedComparisonOperators>inList</supportedComparisonOperators>
    <supportedComparisonOperators>notInList</supportedComparisonOperators>
    <supportedComparisonOperators>between</supportedComparisonOperators>
    <supportedComparisonOperators>notBetween</supportedComparisonOperators>
    <supportedComparisonOperators>greater</supportedComparisonOperators>
    <supportedComparisonOperators>greaterOrEqual</supportedComparisonOperators>
    <supportedComparisonOperators>less</supportedComparisonOperators>
    <supportedComparisonOperators>lessOrEqual</supportedComparisonOperators>
    <supportedValueBasedHierarchyComparisonOperators>equal</supportedValueBasedHierarchyComparisonOperators>
    <supportedValueBasedHierarchyComparisonOperators>inList</supportedValueBasedHierarchyComparisonOperators>
    <supportedLevelBasedHierarchyComparisonOperators>equal</supportedLevelBasedHierarchyComparisonOperators>
    <supportedLevelBasedHierarchyComparisonOperators>inList</supportedLevelBasedHierarchyComparisonOperators>
    <supportedLogicalOperators>and</supportedLogicalOperators>
    <supportedObjects>attribute</supportedObjects>
    <supportedObjects>dimension</supportedObjects>
    <supportedObjects>hierarchy</supportedObjects>
    <supportedObjects>level</supportedObjects>
  </filterCapability>
  <resultObjectCapability useAttributeSeparatelyAvailable="true" buildLogicalSetWithMetadataFunctionsAvail
able="true" buildLogicalSetWithMemberFunctionsAvailable="true" memberSelectionPromptSupported="true"
supportsMemberSelectionCompletion="true"/>
</datasource:QueryCapability>
```

Example: To get the capabilities of a BEx query**Note:**

Retrieves the list of available BW connections.

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%"""" http://<server
Name>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}/capabilities
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/bwconnections/{BwConnectionId}/capabilities
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 12:06:42 GMT
Content-Type: application/xml
Content-Length: 544

<datasource:QueryCapability xmlns:datasource="http://com.sap.sl.datasource">
  <generalCapability customQueryScriptSupported="false" showHideScopeSupported="false"/>
  <dataProcessingCapability removeEmptyRowsAvailable="true" maxRetrievalTimeAvailable="false" queryStrip
pingAvailable="true"/>
  <filterCapability resultHierarchyInFilterSupported="false" constantComparisonSupported="true" hierarchy
ConstantOperandSupported="true">
    <supportedComparisonOperators>equal</supportedComparisonOperators>
    <supportedComparisonOperators>notEqual</supportedComparisonOperators>
    <supportedComparisonOperators>inList</supportedComparisonOperators>
    <supportedComparisonOperators>notInList</supportedComparisonOperators>
    <supportedComparisonOperators>between</supportedComparisonOperators>
    <supportedComparisonOperators>notBetween</supportedComparisonOperators>
    <supportedComparisonOperators>greater</supportedComparisonOperators>
    <supportedComparisonOperators>greaterOrEqual</supportedComparisonOperators>
    <supportedComparisonOperators>less</supportedComparisonOperators>
    <supportedComparisonOperators>lessOrEqual</supportedComparisonOperators>
    <supportedValueBasedHierarchyComparisonOperators>equal</supportedValueBasedHierarchyComparisonOperators>
    <supportedValueBasedHierarchyComparisonOperators>inList</supportedValueBasedHierarchyComparisonOperators>
    <supportedLevelBasedHierarchyComparisonOperators>equal</supportedLevelBasedHierarchyComparisonOperators>
    <supportedLevelBasedHierarchyComparisonOperators>inList</supportedLevelBasedHierarchyComparisonOperators>
    <supportedLogicalOperators>and</supportedLogicalOperators>
    <supportedObjects>attribute</supportedObjects>
    <supportedObjects>dimension</supportedObjects>
    <supportedObjects>hierarchy</supportedObjects>
    <supportedObjects>level</supportedObjects>
  </filterCapability>
  <resultObjectCapability useAttributeSeparatelyAvailable="true" buildLogicalSetWithMetadataFunctionsAvail
able="true" buildLogicalSetWithMemberFunctionsAvailable="true" memberSelectionPromptSupported="true" sup
portsMemberSelectionCompletion="true"/>
</datasource:QueryCapability>
```

3.7 Scheduling documents

This section describes the main operations to schedule Web Intelligence documents:

- How to retrieve the list of the existing Web Intelligence document schedules
- How to get the details of a schedule action
- How to add a new schedule
- How to delete an existing schedule

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

Related Topics

- [Getting the list of schedules for a document](#)
- [Adding a schedule to a document](#)
- [Getting the details of a schedule](#)
- [Deleting or cancelling the document schedule](#)

3.7.1 About date and time formats

Raylight supports the following date and time formats. The lexical representation of `dateTime` consists of finite-length sequences of characters of the following form:

`yyyy-mm-ddThh:mm:ss (zzzzzz)` where:

Character	Description
<code>yyyy</code>	is a four digit that represents the year
<code>-</code>	are separators between parts of the date portion
<code>first mm</code>	is a two-digit numeral that represents the month
<code>dd</code>	is a two-digit numeral that represents the day
<code>T</code>	is a separator indicating that time-of-day follows
<code>hh</code>	is a two-digit numeral that represents the hour
<code>:</code>	is a separator between parts of the time-of-day portion
<code>second mm</code>	is a two-digit numeral that represents the minute
<code>tt</code>	is a two-integer-digit numeral that represents the whole seconds
<code>zzzzzz</code> or <code>offset</code>	<code>zzzzzz</code> (if present) represents the timezone according to the W3C recommendation: http://www.w3.org/TR/xmlschema-2/#dateTime . If <code>zzzzzz</code> is not specified, the time specified is GMT. You can include an <code>offset</code> , for example, <code>-5:00</code> corresponds to Central Daylight Savings Time as well as Eastern Standard Time in the U.S.

Example:

`2002-10-10T12:00:00-05:00` (noon on 10 October 2002, Central Daylight Savings Time as well as Eastern Standard Time in the U.S.)

2002-10-10T12:00:00Z (equivalent date time with the timezone specified)

3.7.2 Getting the list of schedules for a document

This URL to:

- Get the list of existing schedules for a Web Intelligence document (GET `<url>/documents/{documentId}/schedules`).

For a more complete information on scheduling, refer to the *SAP Business Objects Business Intelligence Platform, BI Launch Pad User Guide* on the Help Portal. The link is referenced at the bottom of this section.

Note:

`{documentId}`: The identifier of the Web Intelligence document is retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`. You can know if a document has been scheduled by retrieving details on the document with: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}`. The `<scheduled>` element returned in the response body provides `true` if the document has been scheduled, otherwise `false`.

Getting a list of existing schedules

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/schedules`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json

Header	Value
Content-Length	Length of content in the response body

Body
<p>For each schedule, its id, name, format, and status.</p> <p>Status can be: Pending, Running, Paused, Completed or Failed.</p>

Example: To get the list of schedule instances for a document

Windows

```
curl -i -G -H "accept:application/xml" -H "X-SAP-logonToken: ""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/4986/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/4986/schedules
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: application/xml
Content-Length: 188

<schedules>
  <schedule>
    <id>28600</id>
    <name>instanceWebi2Inbox</name>
    <format>Webi</format>
    <status>Completed</status>
  </schedule>
  <schedule>
    <id>28609</id>
    <name>instancePDF</name>
    <format>Pdf</format>
    <status>Completed</status>
  </schedule>
  <schedule>
    <id>28651</id>
    <name>instanceWebi2Inbox</name>
    <format>Webi</format>
    <status>Completed</status>
  </schedule>
  <schedule>
    <id>28810</id>
    <name>instanceWebI</name>
    <format>Webi</format>
    <status>Completed</status>
  </schedule>
</schedules>
```


Related Topics

- [Adding a schedule to a document](#)
- [Getting the details of a schedule](#)
- [Deleting or cancelling the document schedule](#)

3.7.3 Adding a schedule to a document

This URL to:

- Add a new schedule to a document (POST <url>/documents/{documentId}/schedules).

For a more complete information on scheduling, refer to the *SAP Business Objects Business Intelligence Platform, BI Launch Pad User Guide* on the Help Portal. The link is referenced at the bottom of this section.

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the document list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`. You can know if a document has been scheduled by retrieving details on the document with: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}`. The <scheduled> element returned in the response body provides `true` if the document has been scheduled, otherwise `false`.

Adding a new schedule

You can add a new schedule for a Web Intelligence document.

The schedule can be sent in different formats to several destinations and set just for once, daily, hourly or monthly. You can also select a server group that the system uses to run the schedule. These settings are described in the body of the request.

Request:

POST `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/schedules`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

For the new schedule, specify in the request body:

- The name

- The destination. <inbox/> inbox of BI launch pad or the <mail/> address, <ftp/>, and <file system/>.
- The document format type. It can be: "webi", "pdf", "xls", "csv". By default, the format is "webi". "csv" type has specific options, see below.
- The recurrence can be: <once>, <now>, <hourly>, <daily>, <monthly>.
- The server group used to run the schedule.

These parameters are described for clarity, in the different tables below.

The following table describe the XML Request Body <destination/> part

Table 3-463: XML Request Body for <mail>

Body XML for <mail> destination
<pre> <schedule/> <name/> <destination/><keepInstanceInHistory/>. Boolean. Default = true. Keep the instance of the schedule document in history. For all possible destinations: <useSpecificName/>: The specific name of the schedule document at destination, otherwise the default name is used. Destination <mail> <mail> <from>somebody@company.com</from> <to>someone@company.com</to> <cc>somebody_in_copy@company.com</cc> <bcc>people_in_blind_copy@company.com</bcc> <subject>Web Intelligence Restful WS</subject> <message>Text to send</message> <addAttachment>true</addAttachment> </mail> </pre>

Table 3-464: XML Request Body for <inbox>

Body XML for <ftp> destination
<pre> <schedule/> <name>instanceToWebIntelligenceInbox </name> <format type="webi"/> <destination><inbox/></destination>.</pre>

Table 3-465: XML Request Body for <ftp>

Body XML for <ftp> destination
<pre> <schedule/> <name/> <destination/><keepInstanceInHistory/>. Boolean. Default = true. Keep the instance of the schedule document in history. For all possible destinations: <useSpecificName/>: The specific name of the schedule document at destination <useDefaultSettings/>. Boolean. Default = false. Destination <ftp> <ftp> <host>host_name </host> <port>port_numer</port> <username>user_name</username> <password>password</password> <account>account_name</account> <directory>directory_name</directory> </ftp></pre>

Table 3-466: XML Request Body for <filesystem>

Body XML for <filesystem> destination
<pre><schedule/> <name/> <destination/><keepInstanceInHistory/>. Boolean. Default = true. Keep the instance of the schedule document in history. For all possible destinations: <useSpecificName/>: If <useAutomaticName=false/> the specific name of the schedule document at destination <useDefaultSettings/>. Boolean. Default = false. Destination <filesystem> <filesystem> <username>user_name</username> <password>password</password> <directory>directory_name</directory> </filesystem></pre>

Table 3-467: XML Request Body for <format >

Body XML for <format> destination
<pre> <format type= "webi"/> or <format type= "pdf"/> or <format type= "xls"/> or <format type= "csv">...</csv> Specific for "csv" format type: <format type="csv"> <properties> <property key= "textQualifier" >'</property> This can be: ' or " <property key= "columnDelimiter" >,</property>. This can be: ,, ;, or Tab. <property key= "charset" >UTF-8</property>. A valid server charset. <property key= "onePerDataProvider" > false </property> </properties> </format> </pre>

Table 3-468: XML Request Body Recurrence

Body XML for Recurrence	
<pre> <schedule/> <destination/> <name/> <format type= /> </pre>	Schedule now

Body XML for Recurrence	
<pre> <once retriesAllowed="2" retryIntervalInSeconds= "60" > <startdate>2012-08-26T15:58:51.000+02:00</startdate> <enddate>2013-08-27T15:58:51.000+02:00</enddate> </once> </pre>	<p>Schedule just once.</p> <p>This schedule is launched at the start date only.</p>
<pre> <daily retriesAllowed="2" retryIntervalInSeconds= "60" > <startdate>2012-08-26T15:58:51.000+02:00</startdate> <enddate>2012-09-14T15:58:51.000+02:00</enddate> <dayinterval>1</dayinterval> </daily> </pre>	<p>Daily schedule</p> <ul style="list-style-type: none"> • retriesAllowed="2" retryIntervalInSeconds="60" (retry 2 times with 60 sec interval) • the date and the time to launch the schedule • the date and the time to stop the schedule • the day interval to repeat the schedule
<pre> <hourly retriesAllowed="2" retryIntervalInSeconds="60" > <startdate>2012-08-26T15:58:51.000+02:00</startdate> <enddate>2012-09-14T15:58:51.000+02:00</enddate> <hour>1</hour> <minute>30</minute> </hourly> </pre>	<p>Hourly schedule</p> <ul style="list-style-type: none"> • retriesAllowed="2" retryIntervalInSeconds="60" (retry 2 times with 60 sec interval) • the date and the time to launch the schedule • the date and the time to stop the schedule • the time interval to repeat the schedule.
<pre> <monthly retriesAllowed="2" retryIntervalInSeconds= "60" > <startdate>2012-08-26T15:58:51.000+02:00</startdate> <enddate>2013-09-14T15:58:51.000+02:00</enddate> <month>1</month> </monthly> </pre>	<p>Monthly schedule</p> <ul style="list-style-type: none"> • retriesAllowed="2" retryIntervalInSeconds="60" (retry 2 times with 60 sec interval) • the date and the time to launch the schedule • the date and the time to stop the schedule • the month interval to repeat the schedule

Table 3-469: XML Request Body Server Group

Body XML for Server Group	Description
<pre><serverGroup id="1236" required="false"/></pre>	<p>You can select the server group that the system uses to run a schedule.</p> <ul style="list-style-type: none">• <code>id</code>: an integer. The ID of the server group as defined in the Console Management System• <code>required</code>: a boolean. Default <code>false</code>. The schedule is then run preferably on the server that belongs to the specified server group. If the specified server is not available, then the object is processed on the next available server. Otherwise: <code>true</code>. <p>If <code>true</code> only the specified servers found within the selected server group is used. If all of the servers in the server group are unavailable, then the schedule is not processed.</p>

Table 3-470: XML Request Body for <CSV> format file

Body XML for <mail> destination
<pre> <Format type=CSV> <name/> <destination/> For all possible destinations: <useAutomaticName/>: Boolean. Default = true. The name of the schedule document. If true the name is the same as the one specified by <name/>. <useSpecificName/>: If <useAutomaticName=false/> the specific name of the schedule document at destination <keepInstanceInHistory/>. Boolean. Default = true. Keep the instance of the schedule docu- ment in history. <useDefaultSettings/>. Boolean. Default = false. Destination <mail> <mail> <from>somebody@company.com</from> <to>someone@company.com</to> <cc>somebody_in_copy@company.com</cc> <bcc>people_in_blind_copy@company.com</bcc> <subject>Web Intelligence Restful WS</subject> <message>Text to send</message> <addAttachment>true</addAttachment> </mail> </pre>

Table 3-471: XML Request Body for <CSV> output

Body XML for <mail> destination
<pre> <format type="csv"> <properties> <property key="textQualifier">"</property> <property key="columnDelimiter">,</property> <property key="charset">UTF-8</property> <property key="onePerDataProvider">>false</property> </properties> </format> </pre> <p>For all possible destinations:</p> <p><useAutomaticName/>: Boolean. Default = true. The name of the schedule document. If true the name is the same as the one specified by <name/>.</p> <p><useSpecificName/>: If <useAutomaticName=false/> the specific name of the schedule document at destination</p> <p><keepInstanceInHistory/>. Boolean. Default = true. Keep the instance of the schedule document in history.</p> <p><useDefaultSettings/>. Boolean. Default = false.</p> <p>Destination <mail></p> <pre> <mail> <from>somebody@company.com</from> <to>someone@company.com</to> <cc>somebody_in_copy@company.com</cc> <bcc>people_in_blind_copy@company.com</bcc> <subject>Web Intelligence Restful WS</subject> <message>Text to send</message> <addAttachment>>true</addAttachment> </mail> </pre>

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json.
Content-Length	Length of content in the response body

Body
<p>Return a message that provides schedule status. In case of success it provides the ID of the new schedule document.</p> <p>For more information on scheduling features, refer to the <i>BI launch pad User Guide</i> on the Help portal at the link referenced at the end of this section.</p>

Example: 1: Schedule once to BI Inbox**Note:**

Schedule the document ID 8002. Scheduling parameters are provided in the `scheduleOnce.xml` file called by `-d "@scheduleOnce.xml"` in cURL command. ID 9354 has been set to the scheduled document. This example shows how to schedule once, the result is sent to the inbox and uses `.`. If the schedule fails, there will be no retry.

This example shows how to add a new schedule to a destination with destination options:

- `keepInstanceInHistory`: Keep an instance in the history, specify the attribute with the value `true/false` in the tag `"destination"`. The default setting is `"true"`.
- `useSpecificName`: declare the specific name in the tag `"useSpecificName"`. Automatic name is used if the tag not specified.
- `fileExtension`: specify the attribute `"fileExtension"` with the value `true/false` in the tag `"useSpecificName"`. The default setting is `"true"`.

`scheduleOnce.xml` details

```
<schedule>
  <name>schedTest</name>
  <format type="webi"/>
  <destination keepInstanceInHistory="false">
    <useSpecificName fileExtension="false">-customTest</useSpecificName>
    <inbox/>
  </destination>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduleOnce.xml" -H "X-SAP-logonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduleOnce.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Mon, 11 Jun 2012 08:38:03 GMT
Content-Type: application/xml
Content-Length: 204

<success>
  <message>Schedule has been done.</message>
  <id>8002</id>
</success>
```

Example: 2: Schedule daily to BI inbox

Note:

Schedule the document ID 8002 daily, allow for two retries, each 60 seconds apart. Scheduling parameters are provided in the `scheduledDaily.xml` file called by `-d "@scheduleDaily.xml"` in cURL command. ID 9354 has been set to the scheduled document.

`scheduledDaily.xml` details

```
<schedule>
  <name>test_daily_inbox</name>
  <format type="webi"/>
  <destination><inbox/></destination>
  <daily retriesAllowed="2" retryIntervalInSeconds="60">
    <startdate>2012-08-26T14:00:00</startdate>
    <enddate>2012-09-14T16:00:00</enddate>
    <dayinterval>1</dayinterval>
  </daily>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledDaily.xml" -H "X-SAP-logonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledDaily.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 3: Schedule every 90 minutes to BI inbox**Note:**

Schedule the document ID 8002 every hour and a half, allow for two retries, each 60 seconds apart. Scheduling parameters are provided in the `scheduledHourly.xml` file called by `-d "@scheduledHourly.xml"` in cURL command. ID 9354 has been set to the scheduled document.

`scheduledHourly.xml` details

```
<schedule>
  <name>test_90min_inbox</name>
  <format type="webi"/>
  <destination><inbox/></destination>
  <hourly retriesAllowed="2" retryIntervalInSeconds="60">
    <startdate>2012-08-26T14:00:00</startdate>
    <enddate>2012-09-14T16:00:00</enddate>
    <hour>1</hour>
    <minute>30</minute>
  </hourly>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledHourly.xml"
-H X-SAP-logonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledHourly
Filesyst.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/sched
ules
```

Example: 4: Add a monthly schedule**Note:**

Schedule the document ID 8002. Scheduling parameters are provided in the `scheduledMonthly.xml` file called by `-d "@scheduledMonthlyFtp.xml"` in cURL command. ID 9354 has been set to the scheduled document.

`scheduledMonthly.xml` details

```
<schedule>
  <name>test_monthly_inbox</name>
  <format type="webi"/>
  <destination><inbox/></destination>
  <monthly retriesAllowed="2" retryIntervalInSeconds="30">
    <startdate>2012-08-26T14:00:00</startdate>
    <enddate>2012-09-14T16:00:00</enddate>
    <month>1</month>
  </monthly>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledMonthly.xml"
-H X-SAP-logonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the `logtok` variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledMonthly.xml.xml"
-H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 5: Add a new once schedule**Note:**

Schedule the document ID 8002. Scheduling parameters are provided in the `now.xml` file called by `-d "@now.xml"` in cURL command. ID 9354 has been set to the scheduled document.

`now.xml` details

```
<schedule>
  <name>test_once_inbox</name>
  <format type="webi"/>
  <destination><inbox/></destination>
  <monthly retriesAllowed="2" retryIntervalInSeconds="30">
    <startdate>2012-08-26T14:00:00</startdate>
    <enddate>2012-09-14T16:00:00</enddate>
    <month>1</month>
  </monthly>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@now.xml" -H X-SAP-lo
gonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@now.xml.xml" -H 'X-
SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 6: Add a new once schedule, with inbox destination for specific users**Note:**

This example shows how to add a new schedule with an inbox destination specifying four user ids (by default, the scheduled document is always send to the sender). The scheduled document can be sent as a shortcut by specifying the tag `sendAs` with the value `shortcut` (default value is `copy`). Schedule the document ID 8002. Scheduling parameters are provided in the `now.xml` file called by `-d "@now.xml"` in cURL command. ID 9354 has been set to the scheduled document.

`now.xml` details

```
<schedule>
  <name>test_once_inbox_users</name>
  <format type="webi"/>
  <destination>
    <inbox>
      <to>11,12,1,2
      <sendAs type="shortcut"/>
    </inbox>
  </destination>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@now.xml" -H X-SAP-logonToken:""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@now.xml.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 7: Add a new now schedule with email recipients

Note:

This example shows how to add a new hourly type schedule (every hour and a half between 26/08/2012 and 14/09/2012). If the schedule fails, there will be 2 retry with 60 seconds between each of them. Schedule the document ID 8002. Scheduling parameters are provided in the `scheduledHourlyMail.xml` file called by `-d "@scheduledHourlyMail.xml"` in cURL command. ID 9354 has been set to the scheduled document.

`scheduledHourlyMail.xml` details

```
<schedule>
  <name>instancePDFFormatToMultiEmailWithBccAndCcEmpty</name>
  <format type="pdf"/>
  <destination>
    <mail>
      <from>somebody@company.com</from>
      <to>person1@company.com;person2@company.com</to>
      <subject>Web Intelligence document</subject>
      <message>New version of pdf document for your attention.</message>
      <addAttachment>true</addAttachment>
    </mail>
  </destination>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledHourlyMail.xml" -H X-SAP-logonToken:""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledHourlyMail.xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 8: Schedule pdf document hourly by mail

Note:

This example shows how to add a new hourly type schedule (every hour between 26/08/2012 and 14/09/2012). If the schedule fails, there will be 2 retry with 60 seconds between each of them. Schedule the document ID 8002. Scheduling parameters are provided in the `scheduledHourly`

Mail.xml file called by -d "@scheduledHourlyMail.xml" in cURL command. ID 9354 has been set to the scheduled document.

scheduledHourlyMail.xml details

```
<schedule>
  <name>test_hourly_mail</name>
  <format type="pdf"/>
  <destination>
    <mail>
      <from>somebody@company.com</from>
      <to>person1@company.com;person2@company.com</to>
      <cc>somebody_in_copy@company.com</cc>
      <bcc>list_of_people_in_blind_copy.company.com</bcc>
      <subject>Web Intelligence Restful WS</subject>
      <message>Text to send</message>
      <addAttachment>true</addAttachment>
    </mail>
  </destination>
  <hourly retriesAllowed="2" retryIntervalInSeconds="30">
    <startdate>2012-08-26T14:00:00</startdate>
    <enddate>2012-09-14T16:00:00</enddate>
    <hour>1</hour>
    <minute>0</minute>
  </hourly>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledHourlyMail.xml"
-H "X-SAP-logonToken: ""<tokenValue>"" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduledHourlyMail.xml"
-H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 9: Add a new once schedule for a document that contains one prompt, with inbox destination

Note:

This example shows how to add a new schedule with an inbox destination (by default, the scheduled document is always send to the sender), since the document contains a prompt, the body xml file also provides the response to the prompt. This example shows how to add a new once (starting at 26/08/2012) schedule and how to fill a prompt text named "Enter State:" with value "Texas" on a DP with "DP0" as id.Schedule the document ID 8002. Scheduling parameters are provided in the now Prompt.xml file called by -d "@nowPrompt.xml" in cURL command. ID 9354 has been set to the scheduled document.

nowPrompt.xml details

```
<schedule>
  <name>test_monthly_ftp</name>
  <format type="webi"/>
  <destination><inbox/></destination>
  <once retriesAllowed="2" retryIntervalInSeconds="60">
    <startdate>2012-08-26T15:58:51.000+02:00</startdate>
    <enddate>2013-08-27T15:58:51.000+02:00</enddate>
  </once>
  <parameters>
    <parameter optional="false" type="prompt" dpId="DP0">
```

```
<id>0</id>
<technicalName>Enter State:</technicalName>
<answer constrained="false" type="text">
  <values>
    <value>Texas</value>
  </values>
</answer>
</parameter>
</parameters>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@nowPrompt.xml" -H X-SAP-logonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@nowPrompt.xml.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 10: Add a new now schedule with a filesystem as the destination

This example shows how to add a new now schedule with filesystem destination specifying a specific folder destination (the default folder is not used). The document is saved in the tmp directory of drive C.

filesystem.xml details

```
<schedule>
  <name>SystemDestinationAndSpecificFolder</name>
  <format type="pdf"/>
  <destination>
    <filesystem>
      <username>{name of user}</username>
      <password>{user's password}</password>
      <directory>c:/tmp</directory>
    </filesystem>
  </destination>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@filesystem.xml" -H X-SAP-logonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@filesystem.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```


Example: 11: Add a new now schedule with an FTP destination

This example shows how to add a new now schedule with ftp destination specifying a directory destination.

`sched_toFTPdestination.xml` details

```
<schedule>
  <name>nowschedTest_ftp</name>
  <format type="xls"/>
  <destination>
    <ftp>
      <host>vs0202.dhcp.pgdev.sap.corp</host>
      <port>21</port>
      <username>admin</username>
      <password>admin</password>
      <directory>\vs0202.dhcp.pgdev.sap.corp\FTP</directory>
    </ftp>
  </destination>
  <serverGroup id="10696" required="true"/>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@sched_toFTPdestination.xml" -H "X-SAP-logonToken: ""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@sched_toFTPdestination.xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 12: Add a new now schedule executed by servers in a specific group

This example shows how to add a new now schedule executed on servers in a specific server group in the CMS.

`sched_toservergrp.xml` details

```
<schedule>
  <name>scheduleInAServerGroup</name>
  <format type="webi"/>
  <serverGroup id="6839" required="true"/>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@sched_toservergrp.xml" -H "X-SAP-logonToken: ""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@sched_toservergrp.xml" -H 'X-SAP-LogonToken:$logtok' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: 13: Add a new now schedule executed by servers in a preferred server group

This example shows how to add a new now schedule executed on servers in a preferred server group

`sched_tospecificservergrp.xml` details

```
<schedule>
  <name>scheduleInAServerGroup</name>
  <format type="webi"/>
  <serverGroup id="6839"/>
</schedule>
```

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@sched_tospecificservergrp.xml" -H 'X-SAP-LogonToken:""%tokenValue%""' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@sched_tospecificservergrp.xml" -H 'X-SAP-LogonToken:$logtok' http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: Example 14: Add a new schedule with destination options

This example shows how to add a new schedule to the inbox with the following destination options:

- Keep an instance in the history: specify the attribut "keepInstanceInHistory" with the value true/false in the tag "destination". True is setted if attribut not specified.
- Use Specific Name: specify the specific name in the tag "useSpecificName". Automatic name is used if tag not specified.
- Add File Extension: specify the attribut "fileExtension" with the value true/false in the tag "useSpecificName". True is set if attribute not specified.

xml file details:

```
<schedule>
  <name>schedTest</name>
  <format type="webi"/>
  <destination keepInstanceInHistory="false">
    <useSpecificName fileExtension="false">-customTest</useSpecificName>
    <inbox/>
  </destination>
```

```
</schedule>
```

Related Topics

- [Getting the list of schedules for a document](#)
- [Getting the details of a schedule](#)
- [Deleting or cancelling the document schedule](#)

3.7.4 Getting the details of a schedule

Use this URL to:

- Get the details of the schedule specified by the URL parameter `scheduleId` (GET `<url>/documents/{documentId}/schedules/{scheduleId}`).
- Delete a schedule (DELETE `<url>/documents/{documentId}/schedules/{scheduleId}`).

Note:

- `{documentId}`: The identifier of the Web Intelligence document is retrieved in the documents list by: GET `http://<serverName>:6405/biprws/raylight/vx/documents`.
- `{scheduleId}`: The identifier of the schedule available for a document is retrieved by: GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/schedules`

Getting the details of a schedule

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/schedules/{scheduleId}`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body
Provides for the specified schedule: its id, name, format and status.

Example: To get the status of the schedule for a document

Note:

Retrieves schedule information of the scheduled instance ID 9439 based on the document 8023

Windows

```
curl -i -G -H "accept:application/xml" -H "X-SAP-LogonToken: ""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/8023/schedules/9439
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken: $logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8023/schedules/9439
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 15 Jun 2012 09:32:45 GMT
Content-Type: application/xml
Content-Length: 184
```

```
<schedule>
  <id>9439</id>
  <name>GC-now-schedule</name>
  <format type="webi"/>
  <status>Completed</status>
</schedule>
```

Related Topics

- [Getting the list of schedules for a document](#)
- [Adding a schedule to a document](#)
- [Deleting or cancelling the document schedule](#)

3.7.5 Deleting or cancelling the document schedule

Deleting a schedule

Request:

DELETE http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/schedules/{scheduleId}

Note:

- {documentId}: The identifier of the Web Intelligence document is retrieved in the documents list by: GET http://<serverName>:6405/biprws/raylight/vx/documents
- {scheduleId}: The identifier of the schedule available for a document is retrieved by: GET http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/schedules

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).
scheduleId	Mandatory. Integer. The identifier of the Web Intelligence document schedule.

Response:

Header	Value
Status Code	HTTP response code.
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body

Provides the status of the delete action.

message status and the id of the deleted schedule.

Example: To delete a scheduled document instance**Windows**

```
curl -i -X "DELETE" -H "accept:application/xml" -H "X-SAP-LogonToken:"<TokenValue>" http://<server Name>:6405/biprws/raylight/v1/documents/4986/schedules/9372
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -i -X "DELETE" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<server Name>:6405/biprws/raylight/v1/documents/4986/schedules/9372
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 15 Jun 2012 10:14:15 GMT
Content-Type: application/xml
Content-Length: 204

<success>
  <message>The resource of type 'schedule' with identifier '9372' has been successfully removed
  <id>9372</id>
</success>
```

Related Topics

- [Getting the list of schedules for a document](#)
- [Adding a schedule to a document](#)
- [Getting the details of a schedule](#)

3.8 Refreshing documents

This section describes the refresh document operations. When refreshing a document, contexts and prompts must be identified, then filled.

- Refresh document without parameters
- Refresh document with parameters
- Fill parameters (contexts and prompts)

Note:

Supported prompts for this version:

- prompt with text and long text data,
- prompt with numeric values
- prompt with dateTime values (single, multiple, and interval values)
- prompt with single value
- prompt with multiple values
- optional and non optional prompts

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

You can refresh a document by providing parameters you retrieve using the GET method.

You can also ask for the refresh without providing any parameters (no request body). In this case, if a context or a prompt needs to be filled, Web Intelligence Restful Web Service returns it. If no parameter has to be filled the document is refreshed.

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the documents list by:
GET `http://<serverName>:6405/biprws/raylight/vx/documents`

Related Topics

- [Getting the document refresh parameters before refreshing a document](#)
- [Refreshing a document](#)
- [Cancelling the refresh of a document](#)

3.8.1 Getting the document refresh parameters before refreshing a document

Use this URL to:

- Get the refresh parameters to be filled before running a refresh.
- Fill the refresh parameter (if needed) and run (refresh) the query of a document.

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the documents list by:
GET `http://<serverName>:6405/biprws/raylight/vx/documents`

When a parameter is marked as 'optional', that means this parameter is not mandatory. If you wish to update parameters to trigger a refresh, you can omit it. Cardinality gives information about how many values are expected. It can be 'single' (one value), 'multiple' (many values) or 'interval' (two values). The "parameter/answer/values" node allows client to provide answer values. Is is pre-filled automatically by Raylight with: previous value(s) if applicable, otherwise default value(s).

Getting the refresh parameters

Request

GET `http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters`

Header	Description
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value in quotation marks.

Parameter	Description
documentId	Mandatory. Integer. The identifier of the Web Intelligence document to retrieve. documentId: (type=integer, minimum=1, maximum=highest document number in existing documents).

Response

Header	Value
Status code	HTTP response code
Content-type	application/xml or application/json
Content-length	Length of the content in the response body.

Body
The set of parameters to be filled to refresh a document: context and/or prompt, ID, name and the answer type and values in case of pre-filled default values.

Example: To get the refresh parameters

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""%tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

Response**Note:**

If context

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Mon, 03 Sep 2012 14:10:42 GMT
Content-Type: application/xml
Content-Length: 689
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<parameters>
  <parameter type="context" optional="false">
    <id>0</id>
    <name>Select a context</name>
    <answer type="text" constrained="true">
      <info cardinality="single">
        <values>
          <!--Default values-->
          <value id="2">Reservations</value>
          <value id="1">Sales</value>
        </values>
        <previous>
          <!--Previous values-->
          <value id="2">Reservations</value>
        </previous>
        <values>
        </values>
      </info>
      <!--Values provided by client -->
      <values>
        <value id="2">Reservations</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

Note:

If no parameter

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Mon, 03 Sep 2012 14:21:05 GMT
Content-Type: application/xml
Content-Length: 70
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<parameters/>
```

Example: Context (XML format)

```
<parameters>
  <parameter optional="false" type="context">
    <id>0</id>
    <name>Select a context</name>
    <answer constrained="true" type="Text">
      <info cardinality="Single">
        <values>
          <!-- Default values -->
          <value id="2">Reservations</value>
          <value id="1">Sales</value>
        </values>
        <previous>
```

```

        <!-- Previous values -->
        <value id="2">Reservations</value>
      </previous>
    </info>
    <!-- Value(s) provided by client -->
    <values>
      <value id="2">Reservations</value>
    </values>
  </answer>
</parameter>
</parameters>

```

Example: Prompt of type 'dateTime' (XML format)

```

<parameters>
  <parameter optional="false" type="prompt">
    <id>0</id>
    <name>Enter Open Date:</name>
    <answer constrained="false" type="DateTime">
      <info cardinality="single">
        <previous>
          <value>1992-09-03T17:15:00.000+02:00</value>
        </previous>
      </info>
      <values>
        <value>1992-09-03T17:15:00.000+02:00</value>
      </values>
    </answer>
  </parameter>
</parameters>

```

Example: LOV details (XML format)

```

<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>Enter values for Customer:</technicalName>
    <name>Enter values for Customer:</name>
    <answer constrained="false" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="false" refreshable="true">
          <id>UNIVERSELOV_DS0.DOD</id>
          <intervals>
            <interval id="0">
              <value>Arai</value>
              <value>Okumura</value>
            </interval>
            <interval id="1">
              <value>Oneda</value>
              <value>Wilson</value>
            </interval>
          </intervals>
          <values>
            <value>Arai</value>
            <value>Baker</value>
            <value>Brendt</value>
            .
            .
            <value>Okumura</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="string">Customer </column>
          </columns>
        </lov>
      </info>
      <previous>
        <value>Arai</value>
        <value>Baker</value>
        <value>Brendt</value>
      </previous>
    </answer>
  </parameter>
</parameters>

```

```

        <value>Diemers</value>
        <value>Dupont</value>
        <value>Durnstein</value>
        <value>Edwards</value>
      </previous>
    <values>
      <value>Arai</value>
      <value>Baker</value>
      <value>Brendt</value>
      <value>Diemers</value>
      <value>Dupont</value>
      <value>Durnstein</value>
      <value>Edwards</value>
    </values>
  </answer>
</parameter>
. . .
</parameters>

```

Note:

<intervals/> are not included in the LOV details when there is only one interval.

Example: Multi-columns LOV details (XML format):

When a LOV accepts multi-columns values, all <value>'s are represented through the extended <cvalue> type, where each column of the value is given:

```

<cvalue>
  <column id="0">. . </column>
  <column id="1">. . </column>
  <column id="2">. . </column>
  . . .
  <column id="n">. . </column>
</cvalue>

```

The corresponding headers (column index, type and name) are given back through the <columns> type of the LOV information:

```

<columns mappingId="m">
  <column id="0" type=". . .">. . </column>
  <column id="1" type=". . .">. . </column>
  . . .
  <column id="n" type=". . .">. . </column>
</columns>

```

Where the mappingId shows the column index whose value needs to be given as answer (to refresh a WebI document), but also used as key for sorting.

```

<parameter optional="false" type="prompt" dpId="DP15">
  <id>6</id>
  <technicalName>Enter values for CustomLOV_3colonnes_AllDataType:</technicalName>
  <name>Enter values for CustomLOV_3colonnes_AllDataType:</name>
  <answer constrained="false" type="Text">
    <info cardinality="Single">
      <lov hierarchical="false" refreshable="true">
        <id>UNIVERSELOV_DS2.DOea</id>
        <intervals>
          <interval id="0">
            <cvalue>
              <column id="0">William</column>
              <column id="1">64</column>

```

```

        <column id="2">1995-12-19T02:00:00.000+01:00</column>
      </cvalue>
    <cvalue>
      <column id="0">Silke</column>
      <column id="1">63</column>
      <column id="2">1994-03-25T02:00:00.000+01:00</column>
    </cvalue>
  </interval>
  <interval id="1">
    .
    .
    .
  </interval>
</intervals>
<cvalues>
  <cvalue>
    <column id="0">Werner</column>
    <column id="1">42</column>
    <column id="2">1995-06-08T08:28:00.000+02:00</column>
  </cvalue>
  .
  .
  .
  <cvalue>
    <column id="0">Tony</column>
    <column id="1">55</column>
    <column id="2">1995-07-05T04:00:00.000+02:00</column>
  </cvalue>
</cvalues>
<columns mappingId="0">
  <column id="0" type="string">CustomLOV_3colonnes_AllDataType </column>
  <column id="1" type="double">Age </column>
  <column id="2" type="date_time">Invoice Date </column>
</columns>
</lov>
<previous>
  <value>Andre</value>
</previous>
</info>
<values>
  <value>Andre</value>
</values>
</answer>
</parameter>

```

Related Topics

- [To log on to the BI platform](#)
- [Document: retrieving, copying, or creating](#)

3.8.2 Refreshing a document

Use this URL to refresh a document.

Request

PUT <http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters>

Header	Value
Content-Type	application/xml or application/json
Accept	application/xml or application/json

Parameter	Value
documentId	Mandatory. Integer. The identifier of the Web Intelligence document.
context	Optional or Mandatory. Contexts are objects that are used in the query such as Reservations or Sales. Contexts can be constrained and can be single or multiple values.
prompt	Optional or mandatory. Text, numeric, dateTime
name	Name of the document
answer type	Text, numeric, or dateTime

Body	Description
<p>XML format:</p> <pre><parameters> <parameter> <id>0</id> <answer> <values> <value id="x"/> </values> </answer> </parameter> </parameters></pre> <p>Json format:</p> <pre>{ "parameters": { "parameter": { "id": 0, "answer": { "values": { "value": { "@id": "2" } } } } }</pre>	<p>Optional. In case of no context and no prompt, no body is required.</p> <p>For a context, this depends on the context type and ID number, returned in the body of the GET request, a value ID is returned in the <info/><values><value>.</p> <p>A prompt can be text, numeric, or dateTime and can be optional or mandatory.</p> <p>dateTime is of the format: yyyy-mm-ddThh:mm:ss.sss+HH:MM HH:MM is the time zone offset. Example: 1992-09-03T17:15:00.000+02:00.</p>

Response

Header	Value
Status code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body
Status on refresh.

Note:

- When a document does not need any parameters to be refreshed, do not send a request body.
- The server list required parameters in an iterative way if the user does not provide them.
- For documents which need parameter(s), if the user does not want send any, a parameter must be sent but without the answer block.

Related Topics

- [Getting the document refresh parameters before refreshing a document](#)
- [Cancelling the refresh of a document](#)

3.8.2.1 Example 1: Refreshing a document with parameters

Parameters are specified in `fillRefreshParameter.xml` called by `-d "@fillRefreshParameter.xml"` in cURL command

`fillRefreshParameter.xml` details

```
<parameters>
  <parameter>
    <id > 0 </ id>
    <answer>
      <values>
        <value id="x"/>value</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""%token
Value%"" -d "@fillRefreshParameter.xml" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/pa
rameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H '"X-SAP-LogonToken:$logtok"'
-d "@fillRefreshParameter.xml" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<success>
  <message>The resource of type 'document' with identifier 'xxx' has been successfully updated.</message>
  <id>xxx</id>
</success>
```

3.8.2.2 Example 2: Refreshing a document with a prompt of the type date time

The prompt cardinality can be 'single' (one value), 'multiple' (many values) or 'interval' (begin time and end time).

```
<?xml version=1.0" encoding="UTF-8" standalone="yes"?>
<parameters>
  <parameter optional="false" type="content">
    <id > 0 </ id>
    <name>Enter open Date </name>
    <answer constrained="true" type="text">
      <info cardinality="single">
        <values>
          <value>1992-09-03T17:15:00.000+02:00</value>
        </values>
      </info>
      <values>
        <value id="1">1992-09-03T17:15:00.000+02:00</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

3.8.2.3 Example 3: Document containing one parameter of type 'context' and one of type 'prompt'

Example: Refresh document with one parameter of type 'context' and one of type 'prompt' (XML format):

Since the RESTful service is stateless, the user must answer the first response before answering the subsequent returned prompt. If no context is provided, the document cannot be refreshed, the response contains the relevant information required.

Response body:

```
<?xml version=1.0" encoding="UTF-8" standalone="yes"?>
<parameters>
  <parameter optional="false" type="content">
    <id > 0 </ id>
    <name>Select a context</name>
    <answer constrained="true" type="text">
      <info cardinality="single">
        <values>
          <value id="2">Reservations</value>
          <value id="1">Sales</value>
        </values>
        <previous>
          <value id="1">Sales</value>
        </previous>
      </info>
      <values>
        <value id="1">Sales</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

The user provides the necessary context value as requested by the server in the request body (in an xml file).

Request body xml file content:

```
<parameters>
  <parameter optional="false" type="content">
    <id > 0 </ id>
    <answer type="text">
      <info cardinality="single">
        <values>
          <value id="1">Sales</value>
        </values>
      </answer>
    </parameter>
  </parameters>
```

Further input is required for the prompt. The response body indicates this:

```
<?xml version=1.0" encoding="UTF-8" standalone="yes"?>
<parameters>
  <parameter optional="false" type="content">
    <id > 1 </ id>
    <name>Enter a value for Country:</name>
    <answer constrained="false" type="text">
      <info cardinality="single">
        <previous>
          <value>France</value>
        </previous>
      </info>
      <values>
        <value>France</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

The user provides the necessary context value and prompt response as requested by the server:

Request body:

```

<parameters>
  <parameter optional="false" type="content">
    <id > 0 </ id>
    <answer type="text">
      <values>
        <value id="1">Sales</value>
      </values>
    </answer>
  </parameter>
</parameters>
<parameters>
  <parameter type="prompt">
    <id > 1 </ id>
    <answer type="text">
      <values>
        <value>France</value>
      </values>
    </answer>
  </parameter>
</parameters>

```

Response body:

```

<?xml version=1.0" encoding="UTF-8" standalone="yes"?>
<success>
  <message>The resource of type 'document' with identifier '23535' has been successfully updated.</message>
  <id>23535
</id>
</success>

```

3.8.2.4 Example 4: Specifying how LoV values are returned

A query may be given to specify how LOV values will be returned, and/or refreshed (if the LOV allows it).

Request

PUT http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters

Header	Value
Content-Type	application/xml or application/json
Accept	application/xml or application/json

Parameter	Value
documentId	Mandatory. Integer. The identifier of the Web Intelligence document.

Parameter	Value
context	Optional or Mandatory. Contexts are objects that are used in the query such as Reservations or Sales. Contexts can be constrained and can be single or multiple values.
prompt	Optional or mandatory. Text, numeric, dateTime
name	Name of the document
answer type	Text, numeric, or dateTime

Body	Description
<p>XML format:</p> <pre> <parameter> <id>1</id> <answer> <info> <lov> <query intervalId="2" intervalSize="6" refresh="true"> <sort order="Descending" /> <search>pattern</search> </query> </lov> </info> </answer> </parameter> </parameters> </pre>	<p>Optional. Where : .</p> <p>intervalId: (type=integer, optional) specifies which values interval should be returned. An error is returned if this index is out of range (depending of values count).</p> <p>intervalSize: (type=integer, optional) specifies how many (and which) values should be returned.</p> <p>refresh: (type=boolean, optional) refreshes the LOV values. An error is returned if the LOV does not allow refreshing.</p> <p>sort: (type=string, values="None" or "Ascending", default="Descending", optional)</p> <p>search pattern: (type=string, optional) the following wildcard characters may be used in the pattern string: "?" for 0 or 1 character, and "*" for 0 or n characters. For example, "M?Gregor" yields to the value McGregor, and "M*Gregor" to the values McGregor and MacGregor.</p>

Response

Header	Value
Status code	HTTP response code
Content-Type	application/xml or application/json

Header	Value
Content-Length	Length of content in the response body

Body
<pre> <parameters> <parameter optional="false" type="prompt" dpId="DP0"> <id>1</id> <technicalName>Select Max Age</technicalName> <name>Select Max Age</name> <answer constrained="false" type="Numeric"> <info cardinality="Single"> <lov hierarchical="false" partial="false" refreshable="true"> <id>UNIVERSELOV_DS0.DO58</id> <intervals> <interval id="0"> <value>74</value> <value>64</value> </interval> <interval id="1"> <value>63</value> <value>45</value> </interval> <interval id="2"> <value>42</value> <value>24</value> </interval> <interval id="3"> <value>22</value> <value>18</value> </interval> </intervals> <values> <value>42</value> <value>38</value> <value>36</value> <value>34</value> <value>29</value> <value>24</value> </values> <columns mappingId="0"> <column id="0" type="double">Age </column> </columns> </lov> <previous> <value>67</value> </previous> </info> <values> <value>67</value> </values> </answer> </parameter> . . . </parameters> </pre>

Note:

- Refreshing a Web Intelligence document with multi-columns parameters has to be done the same way as for single-column values, by giving only the cell(s) whose index matches the `mappingId`.

- `partial = "true"` in LOV information means that only a part of the LOV values can be returned because the LOV size is limited by server settings or the universe's query limit.

3.8.2.5 Example 5: A report with a hierarchical list of values

Hierarchical LoVs accept an extra `<path>` element in the `<query>` section to go down to each level of the hierarchy. You must repeat the query until you have reached the final leaf of the node. Consider the hierarchy below:

The refresh is performed as follows:

1. Send a first call without any `<path>` entry get the first level, as if there were no hierarchy, except for the `hierarchical="true"` tag. `final=false` means the value is a node of the hierarchy, whereas `final=true` or no tag at all means value is a leaf.
2. Send a second call with any of the known first-level values, this will get the corresponding second level values
3. Repeat the call, updating the query body with the required values for each level, until there are no more nodes, i.e. the user has reached a final leaf.
4. Refresh the document as usual, providing all required `<value>` and the mandatory `<path>`.

Example: Sending the first call

A first call without any `<path>` entry get the first level, as if there were no hierarchy, except for the `hierarchical="true"` tag. `final=false` means the value is a node of the hierarchy, whereas `final=true` or no tag at all means value is a leaf.

Windows

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:" "%to
kenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:" "$log
tok" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

The response returns the prompts at the first level in the hierarchy:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Tue, 05 Jun 2012 08:10:36 GMT
Content-Type: application/xml
Content-Length: 355

<parameters>
  <parameter optional="false" type="prompt" dpId="DPe">
    <id>0</id>
    <technicalName>Enter values for Customer in TreeViewLov:</technicalName>
    <name>Enter values for Customer in TreeViewLov:</name>
    <answer constrained="true" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="true" refreshable="true">
```

```

        <id>UNIVERSELOV_DS2.DObb</id>
        <intervals>
          <interval id="0">
            <value final="false">Australia</value>
            <value final="false">US</value>
          </interval>
        </intervals>
        <values>
          <value final="false">Australia</value>
          <value final="false">France</value>
          <value final="false">Germany</value>
          <value final="false">Holland</value>
          <value final="false">Japan</value>
          <value final="false">Madagascar</value>
          <value final="false">Middle East</value>
          <value final="false">Nepal</value>
          <value final="false">South Africa</value>
          <value final="false">UK</value>
          <value final="false">US</value>
        </values>
      </lov>
    </info>
  </answer>
</parameter>
</parameters>

```

Request body

```

<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value>France</value>
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>

```

Send the request again, this time referencing the request2.xml file:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:" "%to
kenValue%" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:" "$log
tok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

The response returns the prompts at the next level in the hierarchy:

```

<parameters>
  <parameter optional="false" type="prompt" dpId="DPe">
    <id>0</id>

```

```

<technicalName>Enter values for Customer in TreeViewLov:</technicalName>
<name>Enter values for Customer in TreeViewLov:</name>
<answer constrained="true" type="Text">
  <info cardinality="Multiple">
    <lov hierarchical="true" refreshable="true">
      <id>UNIVERSELOV_DS2.DObb</id>
      <intervals>
        <interval id="0">
          <value final="false">French Alps</value>
          <value final="false">Provence</value>
        </interval>
      </intervals>
      <values>
        <value final="false">French Alps</value>
        <value final="false">Normandy</value>
        <value final="false">Paris</value>
        <value final="false">Provence</value>
      </values>
    </lov>
  </info>
</answer>
</parameter>
</parameters>

```

Request body

```

<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value>France</value>
              <value>Provence</value>
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>

```

Send the request again, referencing the updated request2.xml file:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:" "%to
kenValue%" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:" "$log
tok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

The response returns the prompts at the next level in the hierarchy:

```

<parameters>
  <parameter optional="false" type="prompt" dpId="DPe">
    <id>0</id>

```

```

<technicalName>Enter values for Customer in TreeViewLov:</technicalName>
<name>Enter values for Customer in TreeViewLov:</name>
<answer constrained="true" type="Text">
  <info cardinality="Multiple">
    <lov hierarchical="true" refreshable="true">
      <id>UNIVERSELOV_DS2.DObb</id>
      <intervals>
        <interval id="0">
          <value final="false">Bordeaux</value>
          <value final="false">Nice</value>
        </interval>
      </intervals>
      <values>
        <value final="false">Bordeaux</value>
        <value final="false">Marseille</value>
        <value final="false">Nice</value>
      </values>
    </lov>
  </info>
</answer>
</parameter>
</parameters>

```

Request body

```

<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value>France</value>
              <value>Provence</value>
              <value>Bordeaux</value>
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>

```

Send the request again, referencing the updated request2.xml file:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""$logtok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

The response returns the prompts at the next level in the hierarchy:

```
<parameters>
```

```

<parameter optional="false" type="prompt" dpId="DPe">
  <id>0</id>
  <technicalName>Enter values for Customer in TreeViewLov:</technicalName>
  <name>Enter values for Customer in TreeViewLov:</name>
  <answer constrained="true" type="Text">
    <info cardinality="Multiple">
      <lov hierarchical="true" refreshable="true">
        <id>UNIVERSEL0V_DS2.D0bb</id>
        <updated>2012-12-11T15:59:15.000+01:00</updated>
        <intervals>
          <interval id="0">
            <value>Piaget</value>
            <value>Piaget</value>
          </interval>
        </intervals>
        <values>
          <value>Piaget</value>
        </values>
        <columns mappingId="0">
          <column id="0" type="string">Customer in TreeViewLov </column>
        </columns>
      </lov>
    </info>
  </answer>
</parameter>
</parameters>

```

The document can now be refreshed as usual, providing a <value> (<path> is not mandatory anymore as the values are now known).

Request body

```

<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <values>
        <value>Piaget</value>
      </values>
    </answer>
  </parameter>
</parameters>

```

Now all the information is in the body file, you can refresh the document using the PUT command and the request2.xml file. When the refresh is successful, you get the following response:

```

<success>
  <message>The resource of type 'Document' with identifier '9586' has been successfully updated.</message>
  <id>9586</id>
</success>

```


3.8.2.6 Example 6: Refreshing a report that has hierarchical mixed LoVs

The document contains hierarchical LoVs where each level of the hierarchy contains specific columns count and types.

First call: "GET"

The first call without any <path> entry gets the first hierarchical level, as if there were no hierarchy, except for the hierarchical="true" tag. final="false" means the value is a node of the hierarchy, whereas final=true or no tag at all means value is a leaf.

Windows

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%token
Value%""" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok"
http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

The response returns the prompts at the first level in the hierarchy:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>Saisir une ou plusieurs valeurs pour Invoice Date</technicalName>
    <name>Saisir une ou plusieurs valeurs pour Invoice Date</name>
    <answer constrained="false" type="DateTime">
      <info cardinality="Multiple">
        <lov hierarchical="true" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS0.D0119</id>
          <intervals>
            <interval id="0">
              <value final="false">FY1992</value>
              <value final="false">FY1995</value>
            </interval>
          </intervals>
          <values>
            <value final="false">FY1992</value>
            <value final="false">FY1993</value>
            <value final="false">FY1994</value>
            <value final="false">FY1995</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="String">Year </column>
          </columns>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

There is only one column for the 1st level, of type String.

Send second call giving known first level values

A second call with any of the known first-level values will get the corresponding second level values:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%token Value%"" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""$logtok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

Request body for request2.xml file

```
<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value>FY1993</value>
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

The response returns the prompts at the next level in the hierarchy:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>Saisir une ou plusieurs valeurs pour Invoice Date</technicalName>
    <name>Saisir une ou plusieurs valeurs pour Invoice Date</name>
    <answer constrained="false" type="DateTime">
      <info cardinality="Multiple">
        <lov hierarchical="true" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS0.D0119</id>
          <intervals>
            <interval id="0">
              <cvalue final="false">
                <column id="0">Q1</column>
                <column id="1">FY1993</column>
              </cvalue>
              <cvalue final="false">
                <column id="0">Q4</column>
                <column id="1">FY1993</column>
              </cvalue>
            </interval>
          </intervals>
          <cvalues>
            <cvalue final="false">
              <column id="0">Q1</column>
              <column id="1">FY1993</column>
            </cvalue>
            <cvalue final="false">
              <column id="0">Q2</column>
              <column id="1">FY1993</column>
            </cvalue>
          </cvalues>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

```

        </cvalue>
        <cvalue final="false">
          <column id="0">Q3</column>
          <column id="1">FY1993</column>
        </cvalue>
        <cvalue final="false">
          <column id="0">Q4</column>
          <column id="1">FY1993</column>
        </cvalue>
      </cvalues>
      <columns mappingId="0">
        <column id="0" type="String">Quarter </column>
        <column id="1" type="String">Year </column>
      </columns>
    </lov>
  </info>
</answer>
</parameter>
</parameters>

```

Here the second level gets two columns, both of type String, and where the selection one is the first (mappingId="0").

The calls are repeated

Until the last level of the hierarchy:

Request body:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""%token
Value%"" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""$logtok" '
-d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

Request body for request2.xml file

```

<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value>FY1993</value>
              <value>Q2</value>
              <value>06</value>
              <value>25</value>
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>

```

Response body:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>Saisir une ou plusieurs valeurs pour Invoice Date</technicalName>
    <name>Saisir une ou plusieurs valeurs pour Invoice Date</name>
    <answer constrained="false" type="DateTime">
      <info cardinality="Multiple">
        <lov hierarchical="true" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS0.D0119</id>
          <intervals>
            . . .
          </intervals>
          <cvalues>
            <cvalue>
              <column id="0">1993-06-19T02:00:00.000+02:00</column>
              <column id="1">06</column>
              <column id="2">FY1993</column>
            </cvalue>
            . . .
            <cvalue>
              <column id="0">1995-06-24T02:00:00.000+02:00</column>
              <column id="1">06</column>
              <column id="2">FY1995</column>
            </cvalue>
          </cvalues>
          <columns mappingId="0">
            <column id="0" type="DateTime">Invoice Date </column>
            <column id="1" type="String">Month </column>
            <column id="2" type="String">Year </column>
          </columns>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

There is no more `final="false"` tag: the last level of the hierarchy is reached, and it contains three columns where the selection one is the first (`mappingId="0"`), of type `DateTime`.

Finally, refresh the document

The document can now be refreshed as usual, providing a `<value>` (`<path>` is not mandatory any more as the values are now known).

Request body:**Windows**

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%token
Value%"""" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/9586/parameters
```

Request body for request2.xml file

```
<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <values>
        <value>1995-06-19T02:00:00.000+02:00</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

Response:

```
<success>
  <message>The resource of type 'Document' with identifier '9586' has been successfully updated.</message>
  <id>9586</id>
</success>
```

3.8.2.7 Example 7: A document containing hierarchical indexed Lists of Values (XML format)

In the case where LOV values are indexed (i.e. "<value id=...>...</value>"), only the id is mandatory.

A first call without any <path> entry gets the first level, as if there were no hierarchy (however, there is the hierarchical="true" tag).

Windows

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken: ""%token Value%"" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

The response returns the prompts at the first level in the hierarchy:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>customer Hierarchy Node variable mandatory</technicalName>
    <name>customer Hierarchy Node variable mandatory</name>
    <answer constrained="true" type="Text">
      <info cardinality="Single">
        <lov hierarchical="true" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS0.D048</id>
          <intervals>
            <interval id="0">
              <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[WORLD 0HIER_NODE]" final="false">
<column id="0">WORLD 0HIER_NODE</column>
<column id="1">WORLD</column>
              </cvalue>
              <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[REST_H 1HIER_REST]" final="false">
<column id="0">REST_H 1HIER_REST</column>
<column id="1">Not Assigned Country (s)</column>
              </cvalue>
            </interval>
          </intervals>
          <cvalues>
            <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[WORLD 0HIER_NODE]" final="false">
              <column id="0">WORLD 0HIER_NODE</column>
              <column id="1">WORLD</column>
            </cvalue>
            <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[REST_H 1HIER_REST]" final="false">
              <column id="0">REST_H 1HIER_REST</column>
              <column id="1">Not Assigned Country (s)</column>
            </cvalue>
          </cvalues>
          <columns mappingId="0">
            <column id="0" type="String">LovHierNodeL00 CountryBase </column>
            <column id="1" type="String">LovHierNodeL00 Country </column>
          </columns>
        </lov>
      </previous>
      <value id="[Z_COUNTRY COUNTRY_HIERAR
CHY_01].[00000000000000000000000000000013]">00000000000000000000000000000013</value>
    </previous>
  </info>
  <values>
    <value id="[Z_COUNTRY COUNTRY_HIERAR
CHY_01].[00000000000000000000000000000013]">00000000000000000000000000000013</value>
  </values>
</answer>
</parameter>
</parameters>
```

Send second call giving known first level values

A second call with any of the known first-level values will get the corresponding second level values:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%token
Value%"""" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H '"X-SAP-LogonToken:$logtok"'
-d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

Request body for request2.xml file

```
<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[WORLD OHIER_NODE]" />
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

The response returns the prompts at the next level in the hierarchy:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>customer Hierarchy Node variable mandatory</technicalName>
    <name>customer Hierarchy Node variable mandatory</name>
    <answer constrained="true" type="Text">
      <info cardinality="Single">
        <lov hierarchical="true" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS0.D048</id>
          <intervals>
            <interval id="0">
              <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[EUROPE OHIER_NODE]" final="false">
                <column id="0">EUROPE OHIER_NODE</column>
                <column id="1">EUROPE</column>
              </cvalue>
              <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[MIDDLE_EAST OHIER_NODE]" fi
                nal="false">
                <column id="0">MIDDLE EAST OHIER_NODE</column>
                <column id="1">MIDDLE_EAST</column>
              </cvalue>
            </interval>
          </intervals>
          <cvalues>
            <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[EUROPE OHIER_NODE]" final="false">
              <column id="0">EUROPE OHIER_NODE</column>
              <column id="1">EUROPE</column>
            </cvalue>
            <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[NORTH_AMERICA OHIER_NODE]" final="false">
              <column id="0">NORTH_AMERICA OHIER_NODE</column>
              <column id="1">NORTH_AMERICA</column>
            </cvalue>
            <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[ASIA_PAC OHIER_NODE]" final="false">
              <column id="0">ASIA_PAC OHIER_NODE</column>
              <column id="1">ASIA_PAC</column>
            </cvalue>
            <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[MIDDLE_EAST OHIER_NODE]" final="false">
              <column id="0">MIDDLE EAST OHIER_NODE</column>
              <column id="1">MIDDLE_EAST</column>
            </cvalue>
          </cvalues>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

```
</cvalues>
<columns mappingId="0">
    <column id="0" type="String">LovHierNodeL01 CountryBase </column>
    <column id="1" type="String">LovHierNodeL01 Country </column>
</columns>
</lov>
<previous>
    <value id="[Z_COUNTRY COUNTRY HIERAR
CHY_01].[000000000000000000000000000013]">00000000000000000000000013</value>
</previous>
</info>
<values>
    <value id="[Z_COUNTRY COUNTRY HIERAR
CHY_01].[000000000000000000000000000013]">00000000000000000000000013</value>
</values>
</answer>
</parameter>
</parameters>
```

Repeat for the second level

Request body:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H X-SAP-LogonToken:"""%token
Value%""" -d "@request2.xml" http://<serverName>:6405/biprws/ravlight/v1/documents/8816/parameters
```

UNIX

Note:

Uses the `logtok` variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

Request body for request2.xml file

```
<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <info>
        <lov>
          <query>
            <path>
              <value id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[WORLD 0HIER_NODE]" />
              <value id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[ASIA_PAC 0HIER_NODE]" />
            </path>
          </query>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

Response body:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP0">
    <id>0</id>
    <technicalName>customer Hierarchy Node variable mandatory</technicalName>
    <name>customer Hierarchy Node variable mandatory</name>
```



```

<answer constrained="true" type="Text">
  <info cardinality="Single">
    <lov hierarchical="true" partial="false" refreshable="true">
      <id>UNIVERSELOV_DS0.DO48</id>
      <intervals>
        <interval id="0">
          <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000003]">
<column id="0">00000000000000000000000000000000000000000003</column>
<column id="1">Australia</column>
          </cvalue>
          <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[000000000000000000000000063]">
<column id="0">000000000000000000000000000000000000000000063</column>
<column id="1">Thailand</column>
          </cvalue>
        </interval>
      </intervals>
      <cvalues>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[0000000000000000000000000003]">
          <column id="0">00000000000000000000000000000000000000000003</column>
          <column id="1">Australia</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[0000000000000000000000000006]">
          <column id="0">00000000000000000000000000000000000000000006</column>
          <column id="1">Bangladesh</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000015]">
          <column id="0">000000000000000000000000000000000000000000015</column>
          <column id="1">China</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000030]">
          <column id="0">000000000000000000000000000000000000000000030</column>
          <column id="1">Indonesia</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000035]">
          <column id="0">000000000000000000000000000000000000000000035</column>
          <column id="1">Japan</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000038]">
          <column id="0">000000000000000000000000000000000000000000038</column>
          <column id="1">Malaysia</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000041]">
          <column id="0">000000000000000000000000000000000000000000041</column>
          <column id="1">Nepal</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000050]">
          <column id="0">000000000000000000000000000000000000000000050</column>
          <column id="1">Philippines</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000058]">
          <column id="0">000000000000000000000000000000000000000000058</column>
          <column id="1">Singapore</column>
        </cvalue>
        <cvalue id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000063]">
          <column id="0">000000000000000000000000000000000000000000063</column>
          <column id="1">Thailand</column>
        </cvalue>
      </cvalues>
      <columns mappingId="0">
        <column id="0" type="String">LovHierNodeL02 CountryBase </column>
        <column id="1" type="String">LovHierNodeL02 Country </column>
      </columns>
    </lov>
    <previous>
      <value id="[Z_COUNTRY COUNTRY_HIERAR
CHY_01].[000000000000000000000000000000000000000000013]">000000000000000000000000000000000000000000013</value>
    </previous>
  </info>
  <values>
    <value id="[Z_COUNTRY COUNTRY_HIERAR
CHY_01].[000000000000000000000000000000000000000000013]">000000000000000000000000000000000000000000013</value>
  </values>
</answer>
</parameter>
</parameters>

```

There is no more `final="false"` tag: the last level of the hierarchy is reached.

Finally, refresh the document

The document can now be refreshed as usual, providing a <value> (<path> is not mandatory any more as the values are now known).

Request body:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%token Value%""" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok" -d "@request2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

Request body for request2.xml file

```
<parameters>
  <parameter>
    <id>0</id>
    <answer>
      <values>
        <value id="[Z_COUNTRY COUNTRY_HIERARCHY_01].[00000000000000000000000000000015]" />
      </values>
    </answer>
  </parameter>
</parameters>
```

Response:

```
<success>
  <message>The resource of type 'Document' with identifier '8816' has been successfully updated.</message>
  <id>8816</id>
</success>
```

3.8.2.8 Example 8: A document containing nested parameters in Lists of Values (XML format)

Nested parameters are also referred to as cascading parameters. In some cases, a list of values may be dependent on other parameters value(s). Raylight handles those cases the same way as previously, except that the dependent parameters are listed in the List of Value information XML block.

Step 1: Send the initial GET call without any request body:**Windows**

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken: ""%token
Value%"" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "GET" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:$logtok"
http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

The response returns the prompts at the first level in the hierarchy:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>1</id>
    <technicalName>Age from:</technicalName>
    <name>Age from:</name>
    <answer constrained="false" type="Numeric">
      <info cardinality="Single">
        <lov hierarchical="false" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS2.DO9</id>
          <intervals>
            <interval id="0">
              <value>18</value>
              <value>2600</value>
            </interval>
          </intervals>
          <values>
            <value>18</value>
            <value>19</value>
            . . .
            <value>2600</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="Numeric">Age </column>
          </columns>
        </lov>
      </info>
    </answer>
  </parameter>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>2</id>
    <technicalName>Date greater than</technicalName>
    <name>Date greater than</name>
    <answer constrained="false" type="DateTime">
      <info cardinality="Single">
        <lov hierarchical="false" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS2.DO17</id>
          <intervals>
            <interval id="0">
              <value>1992-02-29T22:45:00.000+01:00</value>
              <value>1992-11-11T03:55:00.000+01:00</value>
            </interval>
            . . .
            <interval id="9">
              <value>1995-10-13T01:00:00.000+01:00</value>
              <value>1995-12-29T01:00:00.000+01:00</value>
            </interval>
          </intervals>
          <values>
            <value>1992-02-29T22:45:00.000+01:00</value>
            <value>1992-09-03T17:15:00.000+02:00</value>
            . . .
            <value>1992-11-11T03:55:00.000+01:00</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="DateTime">Invoice Date </column>
          </columns>
        </lov>
      </info>
```

```

    </answer>
  </parameter>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>3</id>
    <technicalName>Region in list:</technicalName>
    <name>Region in list:</name>
    <answer constrained="false" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="false" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS2.DO12</id>
          <intervals>
            <interval id="0">
              <value>Arabia</value>
              <value>Western Cape</value>
            </interval>
          </intervals>
          <values>
            <value>Arabia</value>
            <value>Bavaria</value>
            . . .
            <value>Western Cape</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="String">Region </column>
          </columns>
        </lov>
      </info>
    </answer>
  </parameter>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>4</id>
    <technicalName>To:</technicalName>
    <name>To:</name>
    <answer constrained="false" type="Numeric">
      <info cardinality="Single">
        <lov hierarchical="false" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS2.DO9</id>
          <intervals>
            <interval id="0">
              <value>18</value>
              <value>2600</value>
            </interval>
          </intervals>
          <values>
            <value>18</value>
            <value>19</value>
            . . .
            <value>2600</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="Numeric">Age </column>
          </columns>
        </lov>
      </info>
    </answer>
  </parameter>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>0</id>
    <technicalName>Enter values for CustomLOV_3promptDependant:</technicalName>
    <name>Enter values for CustomLOV_3promptDependant:</name>
    <answer constrained="false" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="false" refreshable="true">
          <id>UNIVERSELOV_DS2.DO10c</id>
          <parameters>
            <id>1</id>
            <id>2</id>
            <id>3</id>
            <id>4</id>
          </parameters>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>

```

Note:

There is a `<parameters><id>1</id><id>2</id><id>3</id><id>4</id></parameters>` block nested inside the last parameter.

Step 2: Resolve the nested parameters to allow the main LOV to be filled in with values

Include the following request body in an xml file:

Request body

Request body for nestedparams.xml file

```
<parameters>
  <parameter>
    <id>1</id>
    <answer>
      <values>
        <value>1</value>
      </values>
    </answer>
  </parameter>
  <parameter>
    <id>2</id>
    <answer>
      <values>
        <value>1970-01-01T00:00:00.000+00:00</value>
      </values>
    </answer>
  </parameter>
  <parameter>
    <id>3</id>
    <answer>
      <values>
        <value>Arabia</value>
        <value>Bavaria</value>
        . . .
        <value>West Nepal</value>
        <value>Western Cape</value>
      </values>
    </answer>
  </parameter>
  <parameter>
    <id>4</id>
    <answer>
      <values>
        <value>150</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

A second call that includes the reference to the nestedparams.xml file will get the corresponding second level values:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""%token
Value%" -d "@nestedparams.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""$logtok" -d "@nestedparams.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

Response body:

```
<parameters>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>0</id>
    <technicalName>Enter values for CustomLOV_3promptDependant:</technicalName>
    <name>Enter values for CustomLOV_3promptDependant:</name>
    <answer constrained="false" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="false" partial="false" refreshable="true">
          <id>UNIVERSELOV_DS2.D010c</id>
          <intervals>
            <interval id="0">
              <value>AKOUTEY</value>
              <value>du Toit</value>
            </interval>
          </intervals>
          <values>
            <value>AKOUTEY</value>
            <value>Arai</value>
            <value>Baker</value>
            . . .
            <value>du Randt</value>
            <value>du Toit</value>
          </values>
          <columns mappingId="0">
            <column id="0" type="String">CustomLOV_3promptDependant </column>
          </columns>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>
```

When the list of nested parameters supplied is incomplete

Resolving only some of the nested parameters does not allow the main LOV to be filled in with values. In the body example below, only parameters 1 and 4 are provided. The response will request parameters 3 and 4 as can be seen in the response below.

Request body for newparams.xml file

```
<parameters>
  <parameter>
    <id>1</id>
    <answer>
      <values>
        <value>29</value>
      </values>
    </answer>
  </parameter>
  <parameter>
    <id>4</id>
    <answer>
      <values>
        <value>72</value>
      </values>
    </answer>
  </parameter>
</parameters>
```

A call that includes the reference to the newparams.xml file above) will get the response below:

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"""%token Value%"""" -d "@nestedparams.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok"" -d "@nestedparams.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8816/parameters
```

Response body:

```
<parameters>
  <parameter optional="false" type="prompt">
    <id>2</id>
    <technicalName>Date greater than</technicalName>
    <name>Date greater than</name>
    <answer constrained="false" type="DateTime">
      <info cardinality="Single">
        <lov hierarchical="false" refreshable="true">
          <id>UNIVERSELOV_DS2.D017</id>
        </lov>
      </info>
    </answer>
  </parameter>
  <parameter optional="false" type="prompt">
    <id>3</id>
    <technicalName>Region in list:</technicalName>
```

```

    <name>Region in list:</name>
    <answer constrained="false" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="false" refreshable="true">
          <id>UNIVERSELOV_DS2.D012</id>
        </lov>
      </info>
    </answer>
  </parameter>
  <parameter optional="false" type="prompt" dpId="DP11">
    <id>0</id>
    <technicalName>Enter values for CustomLOV_3promptDependant:</technicalName>
    <name>Enter values for CustomLOV_3promptDependant:</name>
    <answer constrained="false" type="Text">
      <info cardinality="Multiple">
        <lov hierarchical="false" refreshable="true">
          <id>UNIVERSELOV_DS2.D010c</id>
          <parameters>
            <id>2</id>
            <id>3</id>
          </parameters>
        </lov>
      </info>
    </answer>
  </parameter>
</parameters>

```

3.8.3 Cancelling the refresh of a document

Use the following URL to cancel the refresh of a document:

```
PUT <url>/documents/{documentId}/parameters/execution?cancel=<mode>
```

Cancelling a document refresh

You can cancel the refresh of a document that is being refreshed. If no execution is currently running, this will have no effect.

Note:

{documentId}: The identifier of the Web Intelligence document is retrieved in the documents list by:
GET <http://<serverName>:6405/biprws/raylight/vx/documents>

Request

```
PUT http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters/execution?cancel=<mode>
```

Header	Value
Content-Type	application/xml or application/json
Accept	application/xml or application/json

Parameter	Value
documentId	Mandatory. Integer. The identifier of the Web Intelligence document.
mode	<p>Mandatory. Defines how the cancel is managed: possible values are 'partial', 'restore' and 'purge'.</p> <ul style="list-style-type: none">• partial: When the cancel is performed, this displays the new values retrieved so far in the appropriate parts of the document. The rest of the document will display the values retrieved the last time the query was run.• restore: When the cancel is performed, this restores the values to the document that were retrieved the last time the query was run. The values displayed will not be the most up to date information available on the database. You can run the query later to return the up to date values from the database.• purge: When the cancel is performed, this displays the document empty of values. The structure and formatting of the document is retained. You can run the query later to return the up to date values from the database.

Response

Header	Value
Status code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body
Status on the cancel.

Example: To cancel a document refresh

How the cancel is managed is specified with the `mode` parameter.

Windows

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""%to  
kenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters/execution?can  
cel=partial
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "content-type:application/xml" -H "accept:application/xml" -H "X-SAP-LogonToken:""$log  
tok" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters/execution?cancel=partial
```

Response

```
HTTP/1.1 200 OK  
Server: Apache-Coyote/1.1  
Date: Tue, 05 Jun 2012 08:10:36 GMT  
Content-Type: application/xml  
Content-Length: 355  
  
<success>  
  <message>The resource of type 'document' with identifier 'xxx' has been successfully updated.</message>  
  <id>xxx</id>  
</success>
```

Related Topics

- [Getting the document refresh parameters before refreshing a document](#)
- [Refreshing a document](#)

3.9 Managing universes

This section describes the main operations available on Universes:

- Retrieves the universe list a user has access to (depending on user rights).
- Retrieves details on universe.
- Retrieves the query capabilities about a universe.

The default URL to request Web Intelligence RESTful web services is the following:

```
http://<serverName>:6405/biprws/raylight/vx
```

3.9.1 Getting the list of available universes

Gets the list of universes a user has access to depending on user rights.

Request:

GET `http://<serverName>:6405/biprws/raylight/vx/universes`

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Parameter	Description
limit	Optional. Default 10. Number of universes to return.
type	Optional. Universe type: unv, unx, all. Default: all.
offset	Optional. Default 0. Beginning of the universe list.

Response:

Header	Value
Status code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body.

Body
The list of universes for a user. Universe <code>id</code> , <code>cuid</code> , <code>name</code> , <code>type</code> and <code>folderId</code> in the CMS.

Example: To get the universe list**Note:**

Retrieves two .unx universes.

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"" http://<serverName>:6405/biprws/raylight/v1/universes?type=unx&limit=2
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/universes?type=unx&limit=2
```

```
<universes>
  <universe>
    <id>5588</id>
    <cuid>AWlhsUlpz7lDlnnFr3qMZZ8</cuid>
    <name>ADAPT (2)</name>
    <type>unv</type>
    <folderId>509</folderId>
  </universe>
  <universe>
    <id>5612</id>
    <cuid>AYCKrid6ngFGvrKlwVfZKj4</cuid>
    <name>ADAPT LiveTest</name>
    <type>unv</type>
    <folderId>509</folderId>
  </universe>
  <universe>
    <id>5580</id>
    <cuid>AXx7quD3gJh0tAr7gxQ5rI4</cuid>
    <name>ADAPT LiveTest_ODBC</name>
    <type>unv</type>
    <folderId>509</folderId>
  </universe>
  <universe>
    <id>5604</id>
    <cuid>AYG2Z0qbpPx0s8U8_c7nva0</cuid>
    <name>ADAPT LiveTest_OLEDB</name>
    <type>unv</type>
    <folderId>509</folderId>
  </universe>
  <universe>
    <id>6939</id>
    <cuid>AVWbGLO_Q5VAvY.EPw18i2w</cuid>
    <name>AdventureWorks.unx</name>
    <type>unx</type>
    <folderId>6870</folderId>
  </universe>
</universes>
```

Related Topics

- [To log on to the BI platform](#)

3.9.2 Getting the details of a universe

Gets the details of a universe referenced by its ID.

Note:

The detailed information of the universe describes only the default perspective.

Request:

GET http://<serverName>:6405/biprws/raylight/vx/universes/{universeId}

Note:

{universeId}: The identifier of the universe retrieved in the universe list by: GET
http://<serverName>:6405/biprws/raylight/vx/universes

Header	Value
Accept	application/xml or application/json
X-SAP-LogonToken	The logon token value, in quotation marks

Response:

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)
<p>Details of the universe identified by its:</p> <p><id>, <cuid>, <name>, <type>, <folderId>, <connected> the connection with the RDBMS is set and <outline> that described the universe elements.</p>

Example: Details of universe ID 9100**Windows**

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""<TokenValue>""" http://<serverName>:6405/biprws/raylight/v1/universes/9100
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:$logtok' http://<serverName>:6405/biprws/raylight/v1/universes/9100
```

Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<universe>
  <id>9100</id>
  <cuid>AftJgs7FPGNBmkdNDnoG8Aw</cuid>
  <name>unv2</name>
  <type>unv</type>
  <folderId>9089</folderId>
  <connected>true</connected>
  <outline>
    <folder>
      <name>City</name>
      <item type="BODimension">
        <name>City Id</name>
        <id>DS0.DO1</id>
      </item>
      <item type="BODimension">
        <name>City</name>
        <id>DS0.DO2</id>
      </item>
      <item type="BODimension">
        <name>Region Id</name>
        <id>DS0.DO3</id>
      </item>
    </folder>
    <folder>
      <name>Country</name>
      <item type="BODimension">
        <name>Country Id</name>
        <id>DS0.DO4</id>
      </item>
      <item type="BODimension">
        <name>Country</name>
        <id>DS0.DO5</id>
      </item>
    </folder>
    <folder>
      <name>Customer</name>
      <item type="BODimension">
        <name>Cust Id</name>
        <id>DS0.DO6</id>
      </item>
      <item type="BODimension">
        <name>First Name</name>
        <id>DS0.DO7</id>
      </item>
      <item type="BODimension">
        <name>Last Name</name>
        <id>DS0.DO8</id>
      </item>
      <item type="BODimension">
        <name>Age</name>
        <id>DS0.DO9</id>
      </item>
      <item type="BODimension">
        <name>Phone Number</name>
        <id>DS0.DOa</id>
      </item>
      <item type="BODimension">
        <name>Address</name>
        <id>DS0.DOb</id>
      </item>
      <item type="BODimension">
        <name>City Id</name>
        <id>DS0.DOc</id>
      </item>
    </folder>
  </outline>
</universe>
```

```

    <item type="BODimension">
      <name>Sales Id</name>
      <id>DS0.D0d</id>
    </item>
    <item type="BODimension">
      <name>Sponsor Id</name>
      <id>DS0.D0e</id>
    </item>
    <item type="BODimension">
      <name>Country Id</name>
      <id>DS0.D0f</id>
    </item>
    <item type="BODimension">
      <name>Nickname</name>
      <id>DS0.D010</id>
    </item>
    <item type="BODimension">
      <name>Birthdate</name>
      <id>DS0.D011</id>
    </item>
  </folder>
  <folder>
    <name>Defect</name>
    <item type="BODimension">
      <name>Item Id</name>
      <id>DS0.D012</id>
    </item>
    <item type="BODimension">
      <name>Inspection Time</name>
      <id>DS0.D013</id>
    </item>
    <item type="BODimension">
      <name>Defect Type Id</name>
      <id>DS0.D014</id>
    </item>
    <item type="BODimension">
      <name>Nb Of Units</name>
      <id>DS0.D015</id>
    </item>
    <item type="BODimension">
      <name>Nb Of Defects</name>
      <id>DS0.D016</id>
    </item>
  </folder>
  <folder>
    <name>Invoice Line</name>
    <item type="BODimension">
      <name>Inv Id</name>
      <id>DS0.D017</id>
    </item>
    <item type="BODimension">
      <name>Service Id</name>
      <id>DS0.D018</id>
    </item>
    <item type="BODimension">
      <name>Days</name>
      <id>DS0.D019</id>
    </item>
    <item type="BODimension">
      <name>Nb Guests</name>
      <id>DS0.D01a</id>
    </item>
  </folder>
  <folder>
    <name>Region</name>
    <item type="BODimension">
      <name>Region Id</name>
      <id>DS0.D01b</id>
    </item>
    <item type="BODimension">
      <name>Region</name>
      <id>DS0.D01c</id>
    </item>
    <item type="BODimension">
      <name>Country Id</name>
      <id>DS0.D01d</id>
    </item>
  </folder>
</folder>

```

```

    <name>Reject</name>
    <item type="BODimension">
      <name>Item Id</name>
      <id>DS0.D01e</id>
    </item>
    <item type="BODimension">
      <name>Inspection Time</name>
      <id>DS0.D01f</id>
    </item>
    <item type="BODimension">
      <name>Defect Type</name>
      <id>DS0.D020</id>
    </item>
    <item type="BODimension">
      <name>Nb Rejected</name>
      <id>DS0.D021</id>
    </item>
  </folder>
  <folder>
    <name>Inspection</name>
    <item type="BODimension">
      <name>Item Id</name>
      <id>DS0.D022</id>
    </item>
    <item type="BODimension">
      <name>Inspection Time</name>
      <id>DS0.D023</id>
    </item>
    <item type="BODimension">
      <name>Nb Inspected</name>
      <id>DS0.D024</id>
    </item>
  </folder>
  <folder>
    <name>Service Line</name>
    <item type="BODimension">
      <name>Sl Id</name>
      <id>DS0.D025</id>
    </item>
    <item type="BODimension">
      <name>Service Line</name>
      <id>DS0.D026</id>
    </item>
    <item type="BODimension">
      <name>Resort Id</name>
      <id>DS0.D027</id>
    </item>
  </folder>
  <folder>
    <name>Service</name>
    <item type="BODimension">
      <name>Service Id</name>
      <id>DS0.D028</id>
    </item>
    <item type="BODimension">
      <name>Service</name>
      <id>DS0.D029</id>
    </item>
    <item type="BODimension">
      <name>Sl Id</name>
      <id>DS0.D02a</id>
    </item>
    <item type="BODimension">
      <name>Price</name>
      <id>DS0.D02b</id>
    </item>
    <item type="BODimension">
      <name>Open Date</name>
      <id>DS0.D02c</id>
    </item>
    <item type="BODimension">
      <name>Last Sales Date By Service</name>
      <id>DS0.D02d</id>
    </item>
    <item type="BODimension">
      <name>Last Resa Date By Service</name>
      <id>DS0.D02e</id>
    </item>
  </folder>

```



```

<folder>
  <name>Sales</name>
  <item type="BODimension">
    <name>Inv Id</name>
    <id>DS0.DO2f</id>
  </item>
  <item type="BODimension">
    <name>Cust Id</name>
    <id>DS0.DO30</id>
  </item>
  <item type="BODimension">
    <name>Invoice Date</name>
    <id>DS0.DO31</id>
  </item>
</folder>
<folder>
  <name>Resort</name>
  <item type="BODimension">
    <name>Resort Id</name>
    <id>DS0.DO32</id>
  </item>
  <item type="BODimension">
    <name>Resort</name>
    <id>DS0.DO33</id>
  </item>
  <item type="BODimension">
    <name>Country Id</name>
    <id>DS0.DO34</id>
  </item>
</folder>
<folder>
  <name>unv2 Measures</name>
  <item type="Measure">
    <name>Number of Measuremt Value</name>
    <id>DS0.DO35</id>
  </item>
  <item type="Measure">
    <name>Min of Measuremt Value</name>
    <id>DS0.DO36</id>
  </item>
  <item type="Measure">
    <name>Max of Measuremt Value</name>
    <id>DS0.DO37</id>
  </item>
  <item type="Measure">
    <name>Number of Nb Inspected</name>
    <id>DS0.DO38</id>
  </item>
  <item type="Measure">
    <name>Number of Nb Rejected</name>
    <id>DS0.DO39</id>
  </item>
</folder>
</outline>
</universe>

```

Related Topics

- [Getting the list of available universes](#)
- [To log on to the BI platform](#)

3.9.3 Getting the query capabilities of a universe

Get the query capabilities of a universe.

Note:

{universeId}: The identifier of the universe retrieved in the universe list by: GET
http://<serverName>:6405/biprws/raylight/vx/universes

Request:

GET http://<serverName>:6405/biprws/raylight/vx/universes/{universeId}/capabilities

Header	Value
Accept	text/xml
X-SAP-LogonToken	Mandatory. The SAP logon token contained between quotation marks

Parameter	Description
universeId	Mandatory. Integer. The identifier of the Web Intelligence universe to retrieve. universeId: (type=integer, minimum=1, maximum=highest universe number in existing universes).

Response:

The XML or Json flow that contains the document list.

Header	Value
Status Code	HTTP response code
Content-Type	application/xml or application/json
Content-Length	Length of content in the response body

Body (XML)

The query capabilities of the Web Intelligence document. The list depends on user access rights. For each document:

General query capabilities

Data processing capabilities

Filter capabilities: sub queries supported, result hierarchy in filter, object comparison, constant comparison, query on query

List of supported comparison operators

List of query on query supported comparison operators

Example: Retrieves the capabilities of universe ID 2234 from the CMS**Windows**

```
curl -G -i -H "accept:text/xml" -H X-SAP-LogonToken:"""%tokenValue%"""" http://<serverName>:6405/biprws/raylight/v1/universe/2234/capabilities
```

UNIX**Note:**

Uses the logtok variable set at login

```
curl -G -i -H "accept:text/xml" 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/universe/2234/capabilities
```

Response

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Date: Fri, 01 Jun 2012 09:49:52 GMT
Content-Type: application/xml
Content-Length: 217
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<datasource:QueryCapability xmlns:datasource="http://com.sap.sl.datasource">
  <generalCapability combinedQueriesSupported="true" viewQueryScriptAvailable="true" maxValuesForInList="999"/>
  <dataProcessingCapability removeDuplicateRowsAvailable="true"/>
  <filterCapability subQueriesSupported="true" resultHierarchyInFilterSupported="false" objectComparisonSupported="true" constantComparisonSupported="true" queryOnQuerySupported="true">

    <supportedComparisonOperators>equal</supportedComparisonOperators>
    <supportedComparisonOperators>notEqual</supportedComparisonOperators>
    <supportedComparisonOperators>between</supportedComparisonOperators>
    <supportedComparisonOperators>notBetween</supportedComparisonOperators>
    <supportedComparisonOperators>like</supportedComparisonOperators>
    <supportedComparisonOperators>notLike</supportedComparisonOperators>
    <supportedComparisonOperators>exists</supportedComparisonOperators>
    <supportedComparisonOperators>greater</supportedComparisonOperators>
    <supportedComparisonOperators>greaterOrEqual</supportedComparisonOperators>
    <supportedComparisonOperators>less</supportedComparisonOperators>
    <supportedComparisonOperators>lessOrEqual</supportedComparisonOperators>
    <supportedComparisonOperators>inList</supportedComparisonOperators>
    <supportedComparisonOperators>notInList</supportedComparisonOperators>
    <supportedComparisonOperators>both</supportedComparisonOperators>
    <supportedComparisonOperators>except</supportedComparisonOperators>
    <supportedComparisonOperators>isNull</supportedComparisonOperators>
    <supportedComparisonOperators>notIsNull</supportedComparisonOperators>

  </filterCapability>
</datasource:QueryCapability>
```

```

<supportedLogicalOperators>or</supportedLogicalOperators>
<supportedLogicalOperators>and</supportedLogicalOperators>
<supportedObjects>attribute</supportedObjects>
<supportedObjects>dimension</supportedObjects>
<supportedObjects>measure</supportedObjects>

<queryOnQueryCapability>

<supportedCorrelationTypesByComparisonOperator>
  <value>Any</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="notEqual">
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="greater">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="greaterOrEqual">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="less">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="lessOrEqual">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="between"/>
<supportedCorrelationTypesByComparisonOperator key="notBetween"/>
<supportedCorrelationTypesByComparisonOperator key="inList">
  <value>None</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="notInList">
  <value>None</value></supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="isNull"/>
<supportedCorrelationTypesByComparisonOperator key="notIsNull"/>
<supportedCorrelationTypesByComparisonOperator key="like"/>
<supportedCorrelationTypesByComparisonOperator key="notLike"/>
<supportedCorrelationTypesByComparisonOperator key="both"/>
<supportedCorrelationTypesByComparisonOperator key="except"/>
<supportedCorrelationTypesByComparisonOperator key="exists"/>
<supportedComparisonOperatorsByCorrelationType>
  <value>inList</value>
  <value>notInList</value>
</supportedComparisonOperatorsByCorrelationType>
<supportedComparisonOperatorsByCorrelationType key="All">
  <value>greater</value>
  <value>greaterOrEqual</value>
  <value>less</value>
  <value>lessOrEqual</value>
  <value>notEqual</value>
</supportedComparisonOperatorsByCorrelationType>
<supportedComparisonOperatorsByCorrelationType key="Any">
  <value>greater</value>
  <value>greaterOrEqual</value>
  <value>less</value>
  <value>lessOrEqual</value>
  <value>equal</value>
</supportedComparisonOperatorsByCorrelationType>
<supportedComparisonOperatorsByCorrelationType key="Foreach"/>
</queryOnQueryCapability>

<subQueryCapability><supportedCorrelationTypesByComparisonOperator>
  <value>Any</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="notEqual">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="greater">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="greaterOrEqual">
  <value>Any</value>
  <value>All</value>

```

```

</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="less">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="lessOrEqual">
  <value>Any</value>
  <value>All</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="between"/>
<supportedCorrelationTypesByComparisonOperator key="notBetween"/>
<supportedCorrelationTypesByComparisonOperator key="inList">
  <value>Any</value>
</supportedCorrelationTypesByComparisonOperator>

<supportedCorrelationTypesByComparisonOperator key="notInList">
  <value>Any</value>
</supportedCorrelationTypesByComparisonOperator>
<supportedCorrelationTypesByComparisonOperator key="isNull"/>
<supportedCorrelationTypesByComparisonOperator key="notIsNull"/>
<supportedCorrelationTypesByComparisonOperator key="like"/>
<supportedCorrelationTypesByComparisonOperator key="notLike"/>
<supportedCorrelationTypesByComparisonOperator key="both"/>
<supportedCorrelationTypesByComparisonOperator key="except"/>
<supportedCorrelationTypesByComparisonOperator key="exists"/>
<supportedComparisonOperatorsByCorrelationType/> <supportedComparisonOperatorsByCorrelationType key="All">
  <value>greater</value>
  <value>greaterOrEqual</value>
  <value>less</value>
<value>lessOrEqual</value>
<value>equal</value>
</supportedComparisonOperatorsByCorrelationType>
<supportedComparisonOperatorsByCorrelationType key="Any">
  <value>greater</value>
  <value>greaterOrEqual</value>
  <value>less</value>
  <value>lessOrEqual</value> <value>equal</value>
  <value>notEqual</value>
  <value>inList</value>
  <value>notInList</value>
</supportedComparisonOperatorsByCorrelationType>
<supportedComparisonOperatorsByCorrelationType key="Foreach"/>
</subQueryCapability>
</filterCapability>
<resultObjectCapability useAttributeSeparatelyAvailable="true" sortObjectsAvailable="true" sortNonResultO
bjectsAvailable="true"/>
</datasource:QueryCapability>

```


Example workflow: Open, refresh, save a document, and export a report

The following example describes how to open a document, refresh it and answer prompts, then export a report in the document. You do the following:

- Logon
- Open a document from the CMS
- Refresh the document
- Export a report in the document in pdf format
- Save the document and close it
- Logoff

The second part of this example shows how you can retrieve the document and report IDs if required.

RESTful services quick reference section

This section contains the cheat sheets for the different areas of the Web Intelligence RESTful services API. Each sheet has a table of commands and includes examples of basic operations.

Related Topics

- [Managing documents](#)
- [Managing the document lifecycle \(LCM\)](#)
- [Managing reports](#)
- [Managing data providers](#)
- [Scheduling and refreshing documents](#)
- [Managing universes](#)

5.1 Managing documents

Below are the main commands, grouped by functional area. For more details, refer to the relevant sections of this User's Guide.

Table 5-1: Document Management

Action	Request
Alerters: Add a new alerter to a document	POST <url>/documents/{documentId}/alerters
Alerters: Delete an alerter	DELETE <url>/document/{documentId}/alerters/{alerterId}
Alerters: Get the full description of an alerter	GET <url>/document/{documentId}/alerters/{alerterId}
Alerters: Get the list of all alerters defined in a document	GET <url>/documents/{documentId}/alerters
Alerters: Update the description of an alerter	PUT <url>/document/{documentId}/alerters/{alerterId}

Action	Request
Attachments: Add an attachment to a document.	POST <url>/documents/{documentId}/attachments/
Attachments: Get the list of attachments to a document.	GET <url>/documents/{documentId}/attachments/
Cache: add a cache entry.	POST <url>/documents/{documentId}/cache
Copy a document.	POST <url>/documents
Create an empty document.	POST <url>/documents
CSS: Get the CSS of a document.	GET <url>/documents/{documentId}/css
CSS: Update the CSS of a document.	PUT <url>/documents/{documentId}/css
Custom format numbers: Get a list of the custom format numbers defined in a document	GET <url>/documents/{documentId}/formats
Delete a document.	DELETE <URL>/documents/{documentId}
Document details: Get the details of a document.	GET <url>/documents/{documentId}
Document list: Retrieve the document list from the CMS.	GET <url>/documents
Document properties: Update the document properties.	PUT <url>/documents/{documentId}/properties
Document list: Retrieve the document list from the CMS.	GET <url>/documents
Export a document.	GET <URL>/documents/{documentId}[?parameters] > <outputfilename>
Export a document in paginated mode.	GET -s <url>/documents/{documentId}/pages > filename.xxx xxx = file format: xml, pdf, xls, or xlsx. Depending on the output file format chosen, there are additional parameters you can define, such as dpi, optimization, width, and so on.
Formats: Get a list of all declared formats on the Web Intelligence Server for the default locale.	GET <url>/configuration/formats
Font mappings: Get a list of all declared font mappings on a Web Intelligence server.	GET <url>/configuration/fontmappings

Action	Request
Functions: Get all of the available functions of the formula engine	GET <url>/configuration/functions
Links: Add a new link to the documents expressions dictionary.	POST <url>/documents/{documentId}/links
Links: Get the content of a documents links dictionary.	GET <url>/documents/{documentId}/links
Links: Get the description of a link of a document.	GET <url>/documents/{documentId}/links/{linkId}
Links: Modify a link of a document.	PUT <url>/documents/{documentId}/links/{linkId}
Links: Remove a link from the document .	DELETE <url>/documents/{documentId}/links/{linkId}
Operators: Gets all of the available operators of the formula engine.	GET <url>/vx/configuration/operators
Skins: Get a list of all declared skins on a Web Intelligence Server.	GET <url>/configuration/skins
Styles: Get the list of all defined styles in a document.	GET <url>/documents/{documentId}/styles
Styles: Add a new style to a document	POST <url>/documents/{documentId}/styles
Styles: Delete a style definition.	DELETE <url>/documents/{documentId}/styles/{styleId}
Styles: Get the detailed description of a style.	GET <url>/documents/{documentId}/styles/{styleId}
Styles: Update a style definition.	PUT <url>/documents/{documentId}/styles/{styleId}
Track changes: Get the current trackdata definition for a document.	GET <url>/documents/{documentId}/tracker
Track changes: Enable the tracker resource.	POST <url>/documents/{documentId}/tracker
Track changes: Disable the tracker resource.	GET http://<serverName>:6405/biprws/raylight/vx/documents

Action	Request
Track changes: Change the track data information.	DELETE <url>/documents/{documentId}/tracker
Update a document (close/save).	PUT <url>/documents/{documentId}
Variables: Get the content of a documents variables dictionary.	GET <url>/documents/{documentId}/variables
Variables: Add a new expression to a documents variables dictionary.	POST <url>/documents/{documentId}/variables
Variables: Delete a variable from a documents variable dictionary	DELETE <url>/documents/{documentId}/variables/{variableId}
Variables: Get the definition of a variable from a documents variable dictionary.	GET <url>/documents/{documentId}/variables/{variableId}
Variables: Modify the definition of an variable from a documents variable dictionary.	PUT <url>/documents/{documentId}/variables/{variableId}

Example: To get the list of documents from the CMS

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents
```

Example: To add a new alerter to document ID 1223

Windows

```
curl -i -X "POST" -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" -d "@newalerter.xml" http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "POST" -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' -d "@newalerter.xml" http://<serverName>:6405/biprws/raylight/v1/documents/1223/alerters
```

Example: To modify a link

You define the link in the body which is defined in an .xml file saved in the current path (usually the same path as the cURL tool). For example; link2.xml.

Request:

Windows

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@link2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/links/L2
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl "PUT" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok" -d "@link2.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/links/L2
```

Example: To export a document in Excel 2007 format

Exports the result in the excel2007.xls file. The output is optimized for calculations inside Excel.

Windows

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:""%tokenValue%" -d "@copy.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8022?optimized=true > excel2007.xlsx
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -s -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H "X-SAP-LogonToken:"$logtok" -d "@copy.xml" http://<serverName>:6405/biprws/raylight/v1/documents/8022?optimized=true > excel2007.xlsx
```

Example: To delete the style for document 5022

Windows

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:""%tokenValue%" -d "@copy.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?xxxx
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -i -X "DELETE" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok" -d "@copy.xml" http://<serverName>:6405/biprws/raylight/v1/documents/5022/styles/?xxxx
```

5.2 Managing the document lifecycle (LCM)

Table 5-2: Document lifecycle management

Action	Request
Change the state of a document in order to manage memory.	PUT <url>/documents/{documentId}
Snapshots: Get the list of snapshots of a document.	GET <url>/documents/{documentId}/snapshots
Snapshots: Create a snapshot of a document.	POST <url>/documents/{documentId}/snapshots
Snapshots: Create a snapshot of a document.	POST <url>/documents/{documentId}/snapshots
Snapshots: Restore the document to the state corresponding to the given snapshot.	PUT <url>/documents/{documentId}?snapshotId={token}

Example: To close an unmodified document

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8009
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<servername>:6405/biprws/raylight/v1/documents/8009
```

Example: To get the ID tokens of document ID 8022

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/8022/snapshots
```

UNIX

Note:

Uses the logtok variable set at login

```
./curl.exe -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/8022/snapshots
```

Example: To restore document 5022 to snapshot ID we00000000e5df6062ca2a**Windows**

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%to
kenValue%"" http://<serverName>:6405/biprws/raylight/v1/documents/5022?snapshotId=we00000000e5df6062ca2a
```

UNIX**Note:**

Uses the logtok variable set at login.

```
curl -i -X "PUT" -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"$logtok"
http://<serverName>:6405/biprws/raylight/v1/documents/5022?snapshotId=we00000000e5df6062ca2a
```

5.3 Managing reports

Table 5-3: Report management

Action	Request
Copy a report in a document.	POST http://<server Name>:6405/biprws/raylight/vx/docu ments/{documentId}/reports
Create a report in a document.	POST <url>/documents/{documentId}/re ports
Delete a report	DELETE <url>documents/{documentId}/re ports/{reportId}
Drill: Change the drill output mode.	PUT <url>/documents/{documentId}/re ports/{reportId}/driller
Drill: Create a drill filter	POST <url>/documents/{documentId}/re ports/{reportId}/driller/filters
Drill: Create a snapshot of a report in drill mode	POST http://<server Name>:6405/biprws/raylight/vx/docu ments/{documentId}/reports/{repor tId}/driller/snapshot
Drill: Disable the drill.	DELETE <url>/documents/{documen tId}/reports/{reportId}/driller
Drill: Enable the query drill	POST <url>/documents/{documentId}/re ports/{reportId}/driller

Action	Request
Drill: Get information about the query drill	GET <url>/documents/{documentId}/reports/{reportId}/driller
Drill: Get information about the query drill hierarchies	GET <url>/documents/{documentId}/reports/{reportId}/driller/hierarchies
Drill: Get the list of drill filters for a report	GET <url>/documents/{documentId}/reports/{reportId}/driller/filters
Drill: Get the details of a drill filter	GET <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId}
Drill: Get the list of free drill elements	GET <url>/documents/{documentId}/reports/{reportId}/driller/drillelements
Drill: Removing a drill filter	DELETE <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId}
Drill: Updating a drill filter	PUT <url>/documents/{documentId}/reports/{reportId}/driller/filters/{filterId}
Drill: Performing a drill action	POST <url>/documents/{documentId}/reports/{reportId}/driller/instructions
Export the report in the stated format	<p>GET -s <url>/documents/{documentId}/reports/{reportId} > <reportName.fff></p> <p>fff is the format of the exported document: xml, pdf, htm, xls, or xlsx.</p>
Export the report in paginated mode to various formats	<p>GET -s <url>/documents/{documentId}/reports/{reportId}/pages > <reportName.fff></p> <p>fff is the format of the exported document: xml, pdf, xls, or xlsx.</p> <p>You can also set other parameters depending on the output format. Refer to the main topic for more details.</p>

Action	Request
Exports a report page to various formats.	<pre>GET -s <url>/documents/{documentId}/reports/{reportId}/pages/{pageIndex} > <reportName.fff></pre> <p>fff is the format of the exported document: xml, pdf, htm, xls, or xlsx.</p> <p>You can also set other parameters depending on the output format. Refer to the main topic for more details.</p>
Elements: Get the elements of a report.	<pre>GET <url>/documents/{documentId}/reports/{reportId}/elements</pre>
Elements: Get the details of a report element,.	<pre>GET <url>/documents/{documentId}/reports/{reportId}/elements/{elementId}</pre>
Elements: Export an element of a report to various formats.	<pre>GET -s <url>/documents/{documentId}/reports/{reportId}/elements/{elementId} > <reportelementn.fff></pre> <p>fff is the format of the exported document: xml, pdf, htm, xls, or xlsx.</p> <p>You can also set other parameters depending on the output format. Refer to the main topic for more details.</p>
Get all the reports of a document.	<pre>GET <url>/documents/{documentId}/reports</pre>
Get the details of a report	<pre>GET <url>documents/{documentId}/reports/{reportId}</pre>
Map: get the report map	<pre>GET <url>/documents/{documentId}/reports/{reportId}/map</pre>
Move a report within a document	<pre>POST <url>/documents/{documentId}/reports?fromId=<fromId>&toId=<toId></pre>
Structure: Get the report structure of the specified report	<pre>GET <url>/documents/{documentId}/reports/{reportId}/specification</pre>
Structure: Update the report structure of the specified report	<pre>PUT <url>/documents/{documentId}/reports/{reportId}/specification</pre>
Update the details of a report	<pre>PUT http://<serverName>:6405/biprws/raylight/vx/documents/{documentId}/reports/{reportId}</pre>

Example: To export the report page 1 in XML format**Note:**

Exports the first report page in the `page1.xml` file.

Windows

```
curl -G -s -H "accept:text/xml" -H 'X-SAP-LogonToken:""%tokenValue%""' http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/0 >page1.xml
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -s -H "accept:text/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/9646/reports/479/pages/0 >page1.xml
```

Example: To create a report for document 12782**Note:**

Retrieves reports of the document ID=12782.

Windows

```
curl -POST -i -H "accept:application/xml" -H 'X-SAP-LogonToken:""%tokenValue%""' http://<serverName>:6405/biprws/raylight/v1/documents/12782/reports
```

UNIX**Note:**

Uses the `logtok` variable set at login.

```
curl -POST -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/12782/reports
```

Example: To delete report 67 of document ID 9512**Windows**

```
curl -DELETE -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H 'X-SAP-LogonToken:""%tokenValue%""' http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl DELETE -H "accept:application/vnd.openxmlformats-officedocument.spreadsheetml.sheet" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/9512/reports/67
```

Example: To get the details of a report**Note:**

Retrieve details of the report (ID 3) from the document (ID 7858)

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:\""tokenValue\"" http://<serverName>:6405/biprws/raylight/v1/documents/7858/reports/3
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/7858/reports/3
```

5.4 Managing data providers

Table 5-4: Managing data providers

Action	Request
Add a data provider to a document	POST <url>/documents/{documentId}/dataproviders
Change the data source (update the data provider mapping)	POST <url>/documents/{documentId}/dataproviders/{dataprovderId}/mappings?originDataproviderIds={DP1Id},{DP2Id}&targetDatasourceId={DatasourceId}
Delete a data provider	DELETE <url>/documents/{documentId}/dataproviders/{dataprovderId}
Get details of a data provider for a document	GET <url>/documents/{documentId}/dataproviders/{dataprovderId}
Get the details on the data provider flow in XML or CSV format.	GET <url>/documents/{documentId}/dataproviders/{dataprovderId}/flows/{flowIndex}
Get the list of data providers of a document	GET <url>/documents/{documentId}/dataproviders
Get the list of possible data provider mappings	GET <url>/documents/{documentId}/dataproviders/{dataprovderId}/mappings?originDataproviderIds={DP1Id},{DP2Id}&targetDatasourceId={DatasourceId}

Action	Request
Get the number of flows available for a data provider.	GET <url>/dataproviders/{dataProviderId}/flows/count
Move a data provider in a document	PUT <url>/documents/{documentId}/dataproviders?fromId={DPn}&toId={DPx}
Remove a data provider from a document	DELETE <url>/documents/{documentId}/dataproviders
Update or purge a data provider.	PUT <url>/documents/{documentId}/dataproviders/{dataProviderId}
Update the query specification attached to a given data provider.	PUT <url>/documents/{documentId}/dataproviders/{dataProviderId}/specification

Example: To get the list of dataproviders for a document

Note:

The document ID is 7738

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%""" http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl -G -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/documents/7738/dataproviders
```

Example: To add a data provider

You define the data provider add in the body which is defined in an .xml file saved in the current path (usually the same path as the Curl tool). For example; dp3.xml.

Request:

Windows

```
curl "POST" -i -H "accept:application/xml" -H "content-type:application/xml" -H "X-SAP-LogonToken:"""%tokenValue%""" -d "@dp3.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/dataproviders
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl "POST" -i -H "accept:application/xml" -H "content-type:application/xml" -H 'X-SAP-LogonToken:"$logtok"'  
-d "@dp3.xml" http://<serverName>:6405/biprws/raylight/v1/documents/3422/dataproviders
```

Example: To update (purge) a data provider

This example updates (purges) DP0.

Windows

```
curl -i -X "PUT" -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<server  
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0?purge=true
```

UNIX

Note:

Uses the logtok variable set at login

```
curl -X "PUT" -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server  
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0?purge=true
```

Example: (XML format) Failed to remove the last data provider

Request

Windows

```
curl "DELETE" -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%""" http://<server  
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

UNIX

Note:

Uses the logtok variable set at login.

```
curl DELETE -i -H "accept:application/xml" -H 'X-SAP-LogonToken:"$logtok"' http://<server  
Name>:6405/biprws/raylight/v1/documents/7738/dataproviders/DP0
```

5.5 Managing BW connections and BEx queries

Table 5-5: Managing BW connections and BEx queries

Action	Request
Get the list of BW connections	GET http://<server Name>:6405/biprws/raylight/vx/bwconnections
Get the details of a BW connection	GET http://<server Name>:6405/biprws/raylight/vx/bwconnections/{bwConnectionId}
Browse the details of a BW connection	PUT http://<server Name>:6405/biprws/raylight/vx/bwconnections/{BwConnectionId}
Get the outline of a BEx query	PUT http://<server Name>:6405/biprws/raylight/vx/bwconnections/{bwConnectionId}/outline
Get the capabilities of a BEx query	PUT http://<server Name>:6405/biprws/raylight/vx/bwconnections/{bwConnectionId}/capabilities

5.6 Scheduling and refreshing documents

Table 5-6: Managing scheduling parameters

Action	Request
Get the list of existing schedules for a document.	GET <url>/documents/{documentId}/schedules
Add a new schedule for a document.	POST <url>/documents/{documentId}/schedules
Get the details of a schedule.	GET <URL>/documents/{documentId}/schedules/{scheduleId}
Delete a schedule.	DELETE <url>/documents/{documentId}/schedules/{scheduleId}

Table 5-7: Managing refresh parameters

Action	Request
Cancel the refresh of a document.	PUT <url>/documents/{documentId}/parameters/execution?cancel=<mode>
Get the refresh parameters to be filled before running a query.	GET <url>/documents/{documentId}/parameters
Fill the refresh parameter (if needed) and run (refresh) the query of a document.	GET <url>/documents

Example: To add a schedule to a document

To add a schedule to a document. The schedule is defined in as xml file called `scheduleNow.xml` and is referenced as follows: `-d "@scheduleNow.xml"`.

Windows

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduleNow.xml" -H "X-SAP-LogonToken:""%"tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -i -X "POST" -H "content-type:application/xml" -H "accept:application/xml" -d "@scheduleNow.xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/8002/schedules
```

Example: To get the refresh parameters

To get the refresh parameters for a document:

Windows

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:""%"tokenValue%" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

UNIX**Note:**

Uses the `logtok` variable set at login.

```
curl -G -i -H "accept:application/xml" -H "X-SAP-LogonToken:"$logtok" http://<serverName>:6405/biprws/raylight/v1/documents/{documentId}/parameters
```

5.7 Managing universes

Table 5-8: Managing universes

Action	Request
Gets the list of universes a user has access to depending on user rights.	GET <url>/universes
Gets the details of a universe.	GET <url>/universes/{universeId}
Get the query capabilities of a universe.	GET <url>/universes/{universeId}/capabilities

Example:

Get the query capabilities of a universe.

Windows

```
curl -G -i -H "accept:application/xml" -H X-SAP-LogonToken:"""%tokenValue%"" "http://<serverName>:6405/biprws/raylight/v1/universe/2234/capabilities"
```

UNIX**Note:**

Uses the `logtok` variable set at login

```
curl -G -i -H 'X-SAP-LogonToken:"$logtok"' http://<serverName>:6405/biprws/raylight/v1/universe/2234/capabilities
```


More Information

Information Resource	Location
SAP BusinessObjects product information	http://www.sap.com
SAP Help Portal	<p>Navigate to http://help.sap.com/businessobjects and on the "SAP BusinessObjects Overview" side panel click All Products.</p> <p>You can access the most up-to-date documentation covering all SAP BusinessObjects products and their deployment at the SAP Help Portal. You can download PDF versions or installable HTML libraries.</p> <p>Certain guides are stored on the SAP Service Marketplace and are not available from the SAP Help Portal. These guides are listed on the Help Portal accompanied by a link to the SAP Service Marketplace. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.</p>
SAP Service Marketplace	<p>http://service.sap.com/bosap-support > Documentation</p> <ul style="list-style-type: none"> • Installation guides: https://service.sap.com/bosap-instguides • Release notes: http://service.sap.com/releasenotes <p>The SAP Service Marketplace stores certain installation guides, upgrade and migration guides, deployment guides, release notes and Supported Platforms documents. Customers with a maintenance agreement have an authorized user ID to access this site. Contact your customer support representative to obtain an ID. If you are redirected to the SAP Service Marketplace from the SAP Help Portal, use the menu in the navigation pane on the left to locate the category containing the documentation you want to access.</p>
Docupedia	<p>https://cw.sdn.sap.com/cw/community/docupedia</p> <p>Docupedia provides additional documentation resources, a collaborative authoring environment, and an interactive feedback channel.</p>
Developer resources	<p>https://boc.sdn.sap.com/</p> <p>https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary</p>

Information Resource	Location
SAP BusinessObjects articles on the SAP Community Network	https://www.sdn.sap.com/irj/boc/businessobjects-articles These articles were formerly known as technical papers.
Notes	https://service.sap.com/notes These notes were formerly known as Knowledge Base articles.
Forums on the SAP Community Network	https://www.sdn.sap.com/irj/scn/forums
Training	http://www.sap.com/services/education From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style.
Online customer support	http://service.sap.com/bosap-support The SAP Support Portal contains information about Customer Support programs and services. It also has links to a wide range of technical information and downloads. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
Consulting	http://www.sap.com/services/bysubject/businessobjectsconsulting Consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in topics such as relational and multidimensional databases, connectivity, database design tools, and customized embedding technology.

Index

A

- add
 - alerter 100, 110
 - attachments 129, 132
 - data provider 247
 - document schedule 305
 - document style 51, 67
 - link 138
 - query 247
 - schedule 303, 305
 - variable 83, 90
- alerter
 - adding 100, 110
 - deleting 100, 104, 114
 - getting the description 100, 104
 - getting the list 100, 102
 - updating 107
- alerter description
 - getting 104
 - updating 100, 104
- attachments
 - adding 129, 132
 - getting list of 129, 130
- available functions
 - getting 84
- available universes
 - documents 362

B

- BEx queries 287
- browse
 - BW connections 287
- BW connections 287

C

- cache (document)
 - modifying 148
- change
 - drill 211, 224
 - track changes setting 100, 119
- change data source 251
- change source 261, 265
- charsets 81
 - getting 51
- cheat sheets
 - managing data providers 387
 - managing document lifecycle 382
 - managing documents 377

- cheat sheets (*continued*)
 - managing reports 383
 - managing universes 392
 - scheduling and refreshing
 - documents 390
- commands
 - document 23
 - report 160
- configuration formats
 - getting 51
- copy
 - document 24
- create
 - document 24
 - drill filter 233
 - report 161
 - snapshot 154
 - track changes setting 100, 123
- CSS
 - getting 77
 - updating 77
- custom formats (document) 51, 56

D

- data couorce
 - change 251
- data provider
 - changesource 261, 265
 - changing mappings 265
 - getting mappings 261
 - managing 247
- data provider specification
 - getting details 280
 - updating 280
- data providers
 - adding 247
 - deleting 253
 - getting details 253
 - getting list 247
 - management cheat sheet 387
 - moving 251
 - updating 253
- delete
 - alerter 100, 104, 114
 - data provider 253
 - document 32
 - document schedule 325
 - document style 69, 71
 - link details 129, 146
 - query 253

- delete (*continued*)
 - report 169
 - schedule 325
 - variable description 83, 98
- details of universe
 - getting 364
- disable
 - drill 211, 228
 - track changes setting 100, 127
- document
 - cancelling the refresh 360
 - commands 23
 - copying 24
 - creating 24
 - deleting 32
 - exporting 32, 44
 - exporting in paginated mode 47
 - getting details 32
 - getting font mappings 51, 58
 - getting list of custom formats 51, 56
 - life cycle management 150
 - management cheat sheet 377
 - managing 23
 - refreshing 327
 - restore 158
 - retrieving 24
 - scheduling 301, 326
 - versions 150
- document attachments
 - adding 129, 132
 - getting list of 129, 130
- document cache 148
- document configuration formats 51
- document CSS style
 - getting 77
 - updating 77
- document custom formats
 - getting list 51, 56
- document font mappings 51, 58
- document link
 - adding 138
 - deleting 129, 146
 - getting details 129, 141
 - getting list of 134
 - modifying 129, 143
- document properties 36
 - getting document properties 36
 - updating document properties 41
- document refresh
 - cancel 360

document refresh (*continued*)

getting 327

document schedule

adding 305

deleting 325

getting 303

getting details 323

document skins 51, 61

document snapshot

creating 154

getting list 154

document state

getting details 150

managing 150

document style

adding 51, 67

deleting 69, 71

getting a definition 69

updating 69, 73

document styles

getting 51, 63

drill

changing 211, 224

disabling 211, 228

enabling 211, 226

get hierarchy information 211, 212

get list of drill elements 218

get snapshot of a report 244

get state 211, 221

performing 241

drill filter

create 233

get details 230

get list 216

removing 239

updating 236

E

element details (report)

getting 190

elements (in a report)

getting 188

enable

drill 211

track changes setting 100, 123

enable drill 226

export

document 32, 44

document in paginated mode 47

page of a report 184

report 177

report (page of) 184

report element 190

report in paginated mode 181

F

filter

create drill filter 233

get drill filter details 230

removing a drill filter 239

update drill filter 236

flows

getting details 277

getting number 275

font mappings (document) 51, 58

functions

getting 84

G

get

alerter description 100, 104

alerter list 100, 102

attachments 129, 130

available functions 84

available operators 86

charsets 81

CSS style 77

data provider details 253

data provider list 247

data provider mappings 261

data provider specification details 280

details of universe 364

document configuration formats 51

document CSS style 77

document details 32

document properties 36

document refresh parameters 327

document schedule details 323

document schedules 303

document snapshots list 154

document style definition 69

document styles 51, 63

drill filter details 230

drill hierarchy information 211, 212

drill information 211, 221

elements 188

flow details 277

font mappings 51, 58

functions 84

link details 129, 141

links 134

list of alerters 100, 102

list of data providers 247

list of document snapshots 154

list of free drill elements 218

list of links 134

list of queries 247

list of report elements 188

get (*continued*)

list of snapshots 154

list of universes 362

number of flows 275

operators 86

properties of a document 36

query capabilities of universe 369

query details 253

query list 247

query mappings 261

query specification details 280

refresh parameters 327

report 161

report details 169

report drill filter list 216

report element details 190

report elements list 188

report map 174

report skins 51, 61

report specifications 197

schedule 303

schedule details 323

snapshot list 154

snapshots list 154

track changes setting 100, 116

universe details 364

universe list 362

universe query capabilities 369

variable description 83, 93

variables list 83, 88

L

lifecycle management 150

document 150

management cheat sheet 382

link

adding 138

deleting 129, 146

getting details 129, 141

getting list of 134

modifying 129, 143

list of elements (in a report) 188

list of universes 362

log off 21

log on 18

M

manage

data provider 247

documents 23

query 247

reports 160

universes 362

- map (report)
 - getting 174
- mappings (data provider) 261, 265
- methods
 - document 23, 247
 - report 160
- modify
 - document cache 148
 - link details 129, 143
 - variable description 83, 95
- move
 - data provider 251
 - query 251
 - report 161

O

- operators 86

P

- page
 - exporting a report page 184
- post
 - data provider mappings 265
 - query mappings 265
- properties
 - document properties 36
 - getting document properties 36
 - updating document properties 41
- put
 - document properties (update) 41

Q

- query
 - adding 247
 - changing mappings 265
 - deleting 253
 - getting details 253
 - getting list 247
 - getting mappings 261
 - managing 247
 - moving 251
 - updating 253
- query capabilities of universe
 - getting 369
- query mappings 261, 265
- query specification
 - getting details 280
 - updating 280

R

- refresh
 - cancel 360
 - document 327
 - document cheat sheet 390
 - getting parameters 327
- remove
 - drill filter 239
- report
 - commands 160
 - creating 161
 - deleting 169
 - exporting 177
 - exporting a page of 184
 - exporting in paginated mode 181
 - getting 161
 - getting details 169
 - managing 160
 - moving 161
- report drill filter
 - get list 216
- report element
 - exporting 190
- report element details
 - getting 190
- report elements
 - getting list 188
- report map
 - getting 174
- report snapshot
 - in drill mode 244
- report specifications
 - getting 197
 - updating 197
- reports
 - management cheat sheet 383
- restore
 - document 158
- retrieve
 - document 24

S

- schedule
 - adding 305
 - deleting 325
 - document 301, 326
 - document cheat sheet 390
 - getting 303
 - getting details 323
- skins
 - getting 51, 61
- snapshot
 - create 154
 - getting list 154

- snapshot (*continued*)
 - of a report in drill mode 244
- style (document CSS)
 - getting 77
- style (document)
 - getting 51, 63
- style definition (document)
 - getting 69

T

- track changes
 - disable 100, 127
- track changes setting
 - create 100, 123
 - enable 100, 123
 - get information 100, 116
 - update 100, 119

U

- universe
 - getting details of 364
 - list of available 362
 - management cheat sheet 392
 - managing 362
- universe query capabilities
 - getting details of 369
- universes
 - getting list of 362
- update
 - alerter description 100, 104, 107
 - CSS style 77
 - data provider 253
 - data provider specification 280
 - document cache 148
 - document CSS style 77
 - document properties 41
 - document style 69, 73
 - drill filter 236
 - properties of a document 41
 - query 253
 - query specification 280
 - report specifications 197
 - track changes setting 100, 119

V

- variable
 - deleting 83, 98
 - modifying 83, 95
- variable definition
 - getting 83, 93
- variable list
 - getting 83, 88

variable)
 adding 83, 90

version
 document 150