



Note: Before using this information and the product it supports, read the information in the Notices section.

© Copyright IBM Corporation 1997, 2011.
U.S. Government Users Restricted Rights - Use, duplication, or disclosure restricted by GSA ADP
Schedule Contract with IBM Corp.

Table of contents

Rational Focal Point RESTful API.....	4
Setup.....	4
Authentication.....	4
Character encoding.....	4
Links.....	5
Service documents.....	6
Element collections.....	6
Changed elements.....	7
Elements.....	7
Attributes.....	8
Notices.....	9
Trademark acknowledgments.....	11

Rational Focal Point RESTful API

The RESTful API for IBM® Rational® Focal Point™ uses HTTP to send and retrieve XML documents. To understand and use the RESTful API, you must be familiar with the REST principles and with HTTP, XML, and the XML Schema.

The RESTful API for Rational Focal Point is a provisional API. It is not for production use and can change at any time.

You can refer to the REST API examples to learn how to use the APIs for Rational Focal Point. The examples are in Java source files that can be compiled and run.

The RESTful API examples are available in Rational Focal Point install directory\apiexamples\RESTAPIExample.zip. An administrator having access to the application server on which Rational Focal Point is installed can provide you with the example package. For more information about the source files and the process to compile the code, see the readme file in the example package.

Setup

The URIs that Rational Focal Point returns are absolute. To create absolute URIs, the Rational Focal Point server must have information about its own server name. To configure this setting, click **Application > Login Page > Login or Balancer URL**.

Authentication

The requests to the RESTful API must be authenticated by using HTTP basic authentication. Unless you use HTTPS, the user name and password are sent without encryption.

In HTTP basic authentication, character encoding is not specified for user names and passwords. User names and passwords can include ASCII characters only. You might be able to use ISO-8859-1 characters if the client correctly encodes the characters.

Character encoding

All XML that is retrieved from Rational Focal Point uses UTF-8 character encoding. For XML that is sent to Rational Focal Point, the charset of the Content-Type HTTP header is used, if present. Otherwise, the encoding that is specified in the XML prolog is used. If no prolog is present, the default for XML is used. The used encoding must match any declared encoding. To avoid confusion, use UTF-8 when you communicate with Rational Focal Point and specify UTF-8 in both the HTTP header and XML prolog.

Links

All XML representations of Rational Focal Point resources can link to other resources by using the XML element link. That link is similar to link found in the Atom format. For more information, see <http://www.atomenabled.org/developers/syndication/atom-format-spec.php#element.link>

Link attributes	Description
Mandatory	
href	The URI of the referenced resource
Optional	
rel	The rel attribute can have one of the following attributes: <ul style="list-style-type: none">• alternate: An alternate representation of the resource. Example: an HTML page• enclosure: A related resource that might be large and require special handling. Example: a binary file• related: A related resource• self: The resource itself• edit: A reference to a resource that can be edited, which might be the resource itself
title	A human-readable title of the link
type	The media (MIME) type of the resource
hreflang	The language of the resource
length	The size of the referenced resource in bytes

Service documents

The service document is the starting point of the RESTful API. The URI for the service document is `http://your Rational Focal Point main URL/context/resources/`. For example, <http://fp.example.com/fp/resources/>.

The service document lists the workspaces that the current user can access. For each workspace, the modules, display views, and add views that the user can access are listed.

Note: Only workspace administrators can access modules.

Each module and view contains an element collection. You can view the element collection in four ways in the service document:

<i>Link</i>	<i>Visualization</i>
indexList	A flat list of elements that includes only the title and URI of each element. If a sort attribute is defined, the elements are sorted accordingly.
indexTree	A tree structure of elements that includes only the title and URI of each element. The title, but not the URI of folders that not are displayed is included.
fullList	A flat list of elements that includes all attribute values for the attributes in the view or module. If a sort attribute is defined, the elements are sorted accordingly.
fullTree	A tree structure of elements that includes the attribute values for the attributes in the view or module.

For details and annotations of the XML format for the service document, see the XML Schema document. The URI for the schema is in the *schemaLocation* XML attribute of the service document.

Element collections

An element collection is the contents of a view or module. The URI for an element collection is stated in the service document.

The XML representation of the element collection contains a URI for each element in the element collection. Depending on the link that was chosen in the service document, the collection might include attribute values and might be structured as tree or a flat list. A single element contains all attribute values and contains more details than the collection.

The XML format for the element collection is defined by the XML Schema document in the *schemaLocation* XML attribute. The XML Schema is unique for each view and module, and dynamically reflects the attribute setup of the module or view. The schema might change in the

following situations:

- An attribute is added, removed, or changed
- A view definition is changed
- A module is renamed

If a collection is an add view or a module, you can add an element to the collection by using POST. The body of the request must contain an XML representation of the element. The XML format to use is in the XML Schema for the collection.

If the collection is an add view, the contents of the view are the folders that can be parents when a new element is added.

Note: RESTful APIs ignore the filters that are applied in views. All the elements of a view are displayed.

Note: The RESTful API ignores any active baselines. Elements and attribute values will be displayed as if no baseline was active

Changed elements

You can filter the element collection to display only elements that have changed since a specified date. To filter a collection, add the ModifiedSince query parameter to the URI of the element collection. The value for ModifiedSince is a datetime value. All elements that have attributes that log history and that have changed since the date that the ModifiedSince value indicates are included.

The format of the ModifiedSince parameter must be an [XML Schema dateTime](#) string or a string that can be parsed by using one of the following [Java SimpleDateFormat](#) patterns:

```
yyyy-MM-dd'T'HH:mm:ss z
yyyy-MM-dd'T'HH:mm z
yyyy-MM-dd
yyyy-MM-dd HH:mm:ss z
yyyy-MM-dd HH:mm z
```

Example:

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/?
ModifiedSince=2009-01-22T12:00:00%2B01:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/?
ModifiedSince=2009-01-22T12:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/?
ModifiedSince=2009-01-22T12:00+GMT%2B01:00
```

Elements

The representation of an element contains all attribute values for the element and all related links to other resources. An element is updated when individual attributes in the element change; the element cannot be updated as a whole. Each editable attribute has an **edit** link which holds the URI to be used when updating the attribute value. You can also use the URI to retrieve the attribute value, but that value is identical to the value in the representation of the element. The XML format is defined by the XML Schema document in the *schemaLocation* XML attribute. The XML Schema is unique for each view and module and dynamically reflects the attribute setup of the module or view. The schema might change in one of the following situations:

- An attribute is added, removed, or changed

- A view definition is changed
- A module is renamed

Note: RESTful APIs ignore the filters that are applied in views. All the attributes of a view are displayed.

Note: The RESTful API ignores any active baselines. Attribute values will be retrieved as if no baseline was active.

Attributes

The representation of an attribute contains the value of the attribute. The XML is the same as the XML included in the XML representation for an element, except in the case of a TextList attribute. In that case, the single attribute value representation also includes edit links for each text entry in the list.

The XML format is defined by the XML Schema document in the *schemaLocation* XML attribute. The XML format is similar for all attribute types, but some parts are unique to each attribute configuration. If the attribute configuration is changed, the XML Schema might change dynamically.

You can change an attribute value by using PUT, POST, or DELETE:

- Use PUT to update the following attribute types: CheckBox, Choice, Date, Float, Integer, Link, LinkList, Matrix, Multichoice, Text, Time Grid, URL, and UniqueId.
- For TextList and File attributes, use POST to add a text entry or a file, use PUT to update a text entry or file, and use DELETE to remove a text entry or file.
- For a Version attribute, use POST to increment to the next major version.

The XML format to update an attribute value is different than the XML representation of the value. For example, a Text attribute might have both a text value and an expression, but when the attribute is updated, it can be given either a new text value or an expression. When you update an attribute, use the XML that is described by the XML Schema document for the attribute.

Note: The RESTful API ignores any active baselines. Attribute values will be retrieved as if no baseline was active.

History of Attributes

The representation of an element and an attribute contains the history of changed attributes if user provides the following parameters along with the HTTP request.

Parameters	Description
includeHistoryOfAttributes	Boolean value. If True, then the history of the attributes are included in the Element or the Attribute representation
start	Date value. Provide the date value since the element or the Attribute history to be included
end	Date value. Provide the date value before the element or the Attribute history to be included

All elements that have attributes that log history and that have changed since the date that the start value indicates and before the end value are included.

The XML format is defined by the XML Schema document in the *schemaLocation* XML attribute. The XML Schema is unique for each view and module and dynamically reflects the attribute setup of the module or view.

The format of the start and end parameter must be an [XML Schema dateTime](#) string or a string that can be parsed by using one of the following [Java SimpleDateFormat](#) patterns:

```
yyyy-MM-dd'T'HH:mm:ss z
yyyy-MM-dd'T'HH:mm z
yyyy-MM-dd
yyyy-MM-dd HH:mm:ss z
yyyy-MM-dd HH:mm z
```

Example:

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/55?
includeHistoryOfAttributes=true&start=2011-03-22T12:00:00%2B01:00&end=2011-06-
22T12:00:00%2B01:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/55?
includeHistoryOfAttributes=true&start=2011-03-22T12:00&end==2011-06-22T12:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/55?
includeHistoryOfAttributes=true&start=2011-03-22T12:00+GMT%2B01:00&end=2011-
06-22T12:00+GMT%2B01:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/55/attribu
tes/212?includeHistoryOfAttributes=true&start=2011-03-
22T12:00:00%2B01:00&end=2011-06-22T12:00:00%2B01:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/55/attribu
tes/212?includeHistoryOfAttributes=true&start=2011-03-22T12:00&end=2011-06-
22T12:00
```

```
http://fp.example.com/fp/resources/workspaces/3/modules/31/elements/55/attribu
tes/212?includeHistoryOfAttributes=true&start=2011-03-22T12:00+GMT
%2B01:00&end=2011-06-22T12:00+GMT%2B01:00
```

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Intellectual Property Dept. for Rational Software
IBM Corporation
5 Technology Park Drive
Westford, MA 01886
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information in softcopy, the photographs and color illustrations may not appear.

Trademark acknowledgments

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International

Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at <http://www.ibm.com/legal/copytrade.shtml>.

Other company, product, or service names may be trademarks or service marks of others.