

Focal Point[®]

RESTful API Reference Manual

7.42

Publication information

FPNR-7420-00 (November 2020)

Information in this publication is subject to change. Changes will be published in new editions or technical newsletters.

Documentation set

The documentation relating to this product includes:

- Focal Point APM Workspace Deployment Guide
- Focal Point APM Workspace User Guide
- Focal Point Business Rule Development and Usage Guide
- Focal Point Custom Chart Plugin Reference Manual
- Focal Point REST RDF API Reference Manual
- Focal Point RESTful API Reference Manual
- Focal Point Web Services API Reference Manual

Copyright notice

Focal Point® (the Programs and associated materials) is a proprietary product of UNICOM Systems, Inc. – a division of UNICOM Global. The Programs have been provided pursuant to License Agreement containing restrictions on their use. The programs and associated materials contain valuable trade secrets and proprietary information of UNICOM Systems, Inc. and are protected by United States Federal and non-United States copyright laws. The Programs and associated materials may not be reproduced, copied, changed, stored, disclosed to third parties, and distributed in any form or media (including but not limited to copies on magnetic media) without the express prior written permission of UNICOM Systems, Inc., UNICOM Plaza Suite 310, 15535 San Fernando Mission Blvd., Mission Hills, CA 91345 USA.

Focal Point®

© Copyright 1997-2020 All Rights Reserved. UNICOM Systems, Inc. – a division of UNICOM Global.

No part of this Program may be reproduced in any form or by electronic means, including the use of information storage and retrieval systems, without the express prior written consent and authorization of UNICOM Systems, Inc.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, without prior written permission from UNICOM Systems, Inc.

Disclaimer

We cannot guarantee freedom from, or assume any responsibility or liability for technical inaccuracies or typographical errors. The information herein is furnished for informational use only and should not be construed as a commitment by UNICOM Systems, Inc. – a division of UNICOM Global.

Trademarks

The following are trademarks or registered trademarks of UNICOM Systems, Inc. in the United States and/or other jurisdictions worldwide: Focal Point, UNICOM, UNICOM Systems.

Trademark acknowledgements

Divisions of UNICOM Global:
DETEC, iET Solutions, Macro 4, SoftLanding, UNICOM.

IBM:
IBM®, AIX®, CICS®, CICS/ESA®, CICS TS® CMAC®, Db2®, DFSMS/MVS®, Domino®, DOORS®, ESCON®, IBM MQ, IBM Z®, IMS™, iSeries®, Jazz™, Language Environment®, LE®, Lotus®, MQSeries®, Multi-Factor Authentication for z/OS®, MVS™, MVS/ESA®, OMEGAMON®, OS/390®, OS/400®, Power®, POWER®, pSeries®, RACF®, Rational®, Rational Team Concert®, RMF™, S/370®, S/390®, SMF®, System/390®, System i®, System p®, System z®, VisualAge®, VM/ESA®, VSE/ESA®, VTAM®, WebSphere®, z/OS®, z/VM®, z/VSE®, zSeries®, z Systems® and the IBM logo are trademarks or registered trademarks of IBM Corporation in the United States or other countries or both.

Microsoft:
Active Directory, Excel, Internet Explorer, Microsoft, Notepad, PowerPoint, Visio, Visual Basic, Windows, Windows 2000, Windows NT, Windows Server 2003, Windows Server 2007, Windows Vista, Windows XP, WordPad and/or other Microsoft products referenced are either trademarks or registered trademarks of Microsoft Corporation.

Adobe Systems Incorporated:
Adobe®, the Adobe logo, Acrobat® and Adobe Reader® are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Amazon.com, Inc.:
Amazon Web Services, the "Powered by AWS" logo and Amazon S3 are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries.

Apache Software Foundation:
Apache, Apache Commons, Apache Tomcat, Log4j, Lucene, Solr, Tomcat, Xalan and Xerces are trademarks of the Apache Software Foundation.

Apple Inc.:
AirPrint, App Store, iPad, iPhone and Safari are trademarks, registered trademarks or service marks of Apple Inc. registered in the United States and other countries.

Atlassian:
Atlassian and Jira are registered trademarks of Atlassian.

AXELOS Limited:
ITIL® is a registered trademark of AXELOS Limited, used under permission of AXELOS Limited.

babel:
A GitHub repository with contributions from Sebastian McKenzie and others and distributed under the MIT license.

BEA Systems, Inc.:
JRockit and WebLogic are registered trademarks of BEA Systems, Inc.

BCD Software, LLC:
ProGen Plus and WebSmart are trademarks or registered trademarks of BCD Software, LLC.

BMC Software Inc.:
Boole & Babbage, Data Packer, Optimizer and Super Optimizer are trademarks or registered trademarks of BMC Software, Inc., or its affiliates or subsidiaries (collectively, "BMC Software").

BSD:
PostgreSQL is distributed under the classic BSD license. (Portions Copyright © 1996-2006, PostgreSQL Global Development Group; Portions Copyright © 1994-1996 Regents of the University of California.)

CA, Inc.:
CA 2E, CA ACF2, CA Datacom, CA Endevor, CA IDMS, CA InterTest, CA NetMaster, CA Optimizer, CA Panexec, CA Panvalet, CA Ramis, CA Telon and CA Top Secret are registered trademarks of CA, Inc.

Candescent SoftBase LLC:
SoftBase is a registered trademark of Candescent SoftBase LLC.

Canonical Ltd:
Ubuntu is a registered trademark of Canonical Ltd.

Cappgemini:
IAF is a trademark of Cappgemini.

Chicago-Soft, Ltd.:
QuickRef is a trademark of Chicago-Soft, Ltd.

Cincom Systems, Inc.:
MANTIS is a registered trademark of Cincom Systems, Inc.

Cisco Systems, Inc.:
Cisco, Cisco Systems, Cisco Unity Express, the Cisco logo, the Cisco Systems logo and IOS are trademarks or registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

Computer Sciences Corporation:
Hogan and Hogan Umbrella are trademarks or registered trademarks of Computer Sciences Corporation.

Compuware Corporation:
Abend-AID and Compuware are trademarks or registered trademarks of Compuware Corporation.

Dell Inc.:
Dell and the Dell logo are trademarks of Dell Inc.

Emtex Limited:
Emtex and VIP are trademarks of Emtex Limited.

ExcelSystems Software Development Inc.:
WebSmart is a registered trademark of ExcelSystems Software Development Inc.

FasterXML, LLC:
Jackson is a FasterXML, LLC GitHub repository and is released under the terms of the Apache License 2.0.

Jean-loup Gailly and Mark Adler:
zlib is a registered trademark or trademark of Jean-loup Gailly and Mark Adler.

GNU General Public License:
Cygwin is free software released under the GNU General Public License.

Google Inc.:
Android™ platform, Google® and Google Chrome® are trademarks or registered trademarks of Google Inc.

Hewlett-Packard Development Company, L.P.:
HP and HP-UX are registered trademarks of Hewlett-Packard Development Company, L.P., and/or its subsidiaries.

hjson:
A GitHub repository with contributions from Christian Zangl and others and distributed under the MIT license.

Idera, Inc.:
LANSA is a registered trademark of Idera, Inc.

IDM Computer Solutions, Inc.:
UltraEdit is a trademark owned by IDM Computer Solutions, Inc. All rights reserved.

Infor (US), Inc.:
Lawson is a registered trademark of Infor (US), Inc.

Innovation Data Processing:
IAM is a registered trademark of Innovation Data Processing Corporation.

Hilton Janfield:
jquery.enhsplitter is a GitHub repository owned by Hilton Janfield and is released under the terms of the GNU Lesser General Public License.

Kofax, Inc.:
Kofax, the Kofax logo and Kofax Capture are the trademarks or registered trademarks of Kofax, Inc., in the United States and other countries.

Daniel Lovasko:
libdes © Copyright 2017-2018 Daniel Lovasko All rights reserved.

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Massachusetts Institute of Technology (MIT):
Kerberos is a trademark of the Massachusetts Institute of Technology (MIT).

Merrill Pty Ltd.:
MXG is a registered trademark of Merrill Pty Ltd.

Mozilla Foundation:
Firefox is a registered trademark of the Mozilla Foundation.

Mozilla Public License:
Expat is free software released under the Mozilla Public License.

OASIS:
BPEL is a trademark of OASIS.

Object Management Group:
BPMN™, Object Management Group®, OMG® and UML® are trademarks or registered trademarks of Object Management Group.

The Open Group:
ArchiMate, The Open Group, TOGAF and UNIX are registered trademarks of The Open Group in the United States and other countries.

Oracle Corporation:
EJB, Java, JDBC, JDK, JMX, JRE, JSP, JVM, OpenJDK, Solaris and SunOS are trademarks or registered trademarks of Oracle Corporation and/or its affiliates. Oracle is a registered trademark, and other Oracle product names, service names, slogans or logos are trademarks or registered trademarks of Oracle Corporation.

The pgAdmin Development Team:
pgAdmin is a trademark of The pgAdmin Development Team.

Pink Elephant Inc.:
Pink Elephant and its logo, PinkVERIFY, PinkSCAN, PinkATLAS, PinkSELECT and PinkREADY are either trademarks or registered trademarks of Pink Elephant Inc.

Python Software Foundation:
Python is a registered trademark of the Python Software Foundation in the United States.

QOS.ch:
SLF4J is from QOS.ch and its source code and binaries are distributed under the MIT license.

Red Hat, Inc.:
Red Hat, Red Hat Enterprise Linux, the Shadowman logo and JBoss are registered trademarks of Red Hat, Inc. in the United States and other countries.

SAP AG:
SAP, the SAP logo, the SAP Partner logo, Duet, SAP ArchiveLink, SAP Crystal Reports, SAP NetWeaver, SAP R/3 and SAPHIRE are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

SAS Institute Inc.:
SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries.

Secretary of State for Defence:
MODAF is a registered trademark of the Secretary of State for Defence.

Software AG:
Adabas and Natural are registered trademarks of Software AG. Software AG and all Software AG products are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc.

SPARC International, Inc.:
SPARC is a registered trademark of SPARC International, Inc. (Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.)

Splunk Inc.
Splunk®, Splunk>®, Listen to Your Data®, The Engine for Machine Data®, Hunk®, Splunk Cloud®, Splunk® Light, SPL™ and Splunk MINT™ are trademarks or registered trademarks of Splunk Inc. in the United States and other countries.

Standardware Inc.:
COPE is a trademark of Standardware Inc.

Sun Microsystems, Inc.:
Sun, Sun Microsystems, the Sun logo, MySQL and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the United States and other countries.

SUSE:
openSUSE, the openSUSE logo, SUSE, the SUSE logo, YAST, and the Chameleon mark are registered trademarks of SUSE or its subsidiaries or affiliates. AutoBuild, SUSE Studio, SUSECON, "We adapt. You succeed." are trademarks of SUSE or its subsidiaries or affiliates.

Syncsort Inc.:
Syncsort is a registered trademark of Syncsort Inc.

Simon Tatham:
PuTTY is copyright Simon Tatham.

TIBCO Software Inc.:
JasperReports and Jaspersoft are registered trademarks of TIBCO Software Inc. and/or its subsidiaries in the United States and/or other countries.

Linus Torvalds:
Linux is a registered trademark of Linus Torvalds.

Unicode, Inc.:
Unicode and the Unicode logo are registered trademarks of Unicode, Inc. in the United States and other countries.

VMware, Inc.:
spring is a registered trademark of VMware, Inc.

Wireshark Foundation:
Wireshark and the "fin" logo are registered trademarks of the Wireshark Foundation.

XEROX CORPORATION:
XEROX, The Document Company and the stylized X are trademarks of XEROX CORPORATION.

X.Org Foundation:
X Window System is a trademark of the X.Org Foundation.

Zachman International, Inc.:
Zachman®, Zachman Framework™ and Zachman International® are trademarks or registered trademarks of Zachman International, Inc.

Additional trademarks and registered trademarks are the property of their respective owners.

Contents

1 Focal Point RESTful API	6
Setup	6
Authentication	6
Character encoding	7
Resources overview	7
Links	7
Service documents	8
Element collection	9
Elements	19
Attributes	22

1 Focal Point RESTful API

The RESTful API for 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 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 Focal Point. The examples are in Java source files that can be compiled and run. The RESTful API examples are available in Focal Point install directory\apiexamples\RESTAPIExample.zip. An administrator having access to the application server on which 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 server name part of the resource URI can be configured in the Login or Balancer URL page.

To specify the server name for the resource URI, in Focal Point, from the User menu, select **Administration**. On the Administration page, click **Application > Login Page**. Under Login Page Settings, click the **Edit** icon for Login or Balancer URL, specify the URL, and click the **Save** icon.

Note Make sure that the host name or the server name does not change. A change in the host name can lead to broken links in Focal Point integrated systems that link to the Focal Point resources.

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 characters are encoded by the client correctly.

Character encoding

The XML that is retrieved from Focal Point uses UTF-8 character encoding.

For XML that is sent to 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 the XML is used. The encoding that is used must match any declared encoding. You can use UTF-8 encoding when you communicate with Focal Point and specify UTF-8 encoding in the HTTP header and XML prolog.

Resources overview

The REST XML resources in Focal Point are service, element collections, element and attributes. These are the resource types and example REST API URIs for the resources:

Resource type	Example REST API URI	Description	Supported REST operations
Service document	<code>http://focalpointserver/fp/resources/</code>	The service document has the high level resources for workspace, module, views that a user has access to.	GET
Element collection	<code>http://focalpointserver/fp/resources/workspaces/1/modules/1/elements/</code>	An element collection has the contents of a view or module.	GET POST
Element	<code>http://focalpointserver/fp/resources/workspaces/1/modules/1/elements/1</code>	Element has contents of attributes	GET
Attribute	<code>http://focalpointserver/fp/resources/workspaces/1/modules/1/elements/1/attributes/1</code>	Attribute	GET PUT POST DELETE

Links

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

Mandatory attribute

Link attributes	Description
href	The URI of the referenced resource.

Optional attributes

Link attributes	Description
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 has the high level resources for workspace, module, views that a user has access to. Only users with workspace administrator rights can access modules.

The resource URI for the service document is:

`http://focalpointserver/fp/resources/`

GET method

Accept header

`application/xml`

Response body

```
<fp:service
xsi:schemaLocation="http://schema.example.com/focalpoint/resources
http://focalpoint.example.com/fp/dtd/service.xsd"> <fp:workspace>
<fp:title>SmartCloud</fp:title> <fp:alias>625ca8ed-d4c6-40a0-b8a2-
ccc0319726a5</fp:alias> <fp:modules> <fp:collection addable="true">
<fp:title>Elements</fp:title> <fp:alias>0f615fdc-ed3d-4ce5-9eff-
4f9b625986e8</fp:alias> <fp:indexList
href="http://focalpoint.example.com/fp/resources/workspaces/2/modules/1/ele
ments/"> <fp:indexTree
href="http://focalpoint.example.com/fp/resources/workspaces/2/modules/1/ele
ments/?tree=true"> <fp:fullList
href="http://focalpoint.example.com/fp/resources/workspaces/2/modules/1/ele
ments/?includeAttributes=true"> <fp:fullTree
href="http://focalpoint.example.com/fp/resources/workspaces/2/modules/1/ele
ments/?tree=true&includeAttributes=true"> </fp:collection>
```

Service document description

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

Link	Visualization
indexList	A 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 are not displayed is included.
fullList	A 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 collection

An element collection has the contents of a view or module. The URI for an element collection is available in the service document and the general URI is:

```
http://focalpointserver/context/resources/workspaces/1/modules/1/elements/
```

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 these 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 operation. 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 in the folders. The folders 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.

The resource URI for the service document is:

```
http://focalpointserver/fp/resources/
```

GET method

Use this method to retrieve the collection of elements in a module. You can use the `View` parameter to filter the elements and attributes.

URL

```
https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/.xml?includeAttributes=true&view=14
```

Accept header

```
application/xml
```

Response body

```
<ns:elementCollection
xsi:schemaLocation="https://fpserver.com:9443/fp/namespace/workspaces/2/modules/1/views/14/elements
https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements.xsd?
view=14"> <ns:Elements count="1"> <ns:Element> <ns:attributes>
<ns:Element_Information/> <ns:ID></ns:ID> <ns:Title></ns:Title>
<ns:Description></ns:Description> <ns:type_is_a_folder>
</ns:type_is_a_folder> </ns:attributes> <ns:selfLink
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/element
```

```
s/15?view=14"/> <ns:alias>38bfbaaa-6af2-49ed-a6f4-d0cf39704a55</ns:alias>
</ns:Element></ns:Elements> <fp:link
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/element
s/?view=14" rel="alternate"/> <fp:link
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/element
s/?view=14&includeAttributes=true&tree=true" rel="alternate"/> <fp:link
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/element
s/?view=14&tree=true" rel="alternate"/></ns:elementCollection>
```

Supported parameters for the GET method

Parameter	Description
includeAttributes	Set this parameter to true to retrieve the attributes of the elements.
	Type: Boolean
view	Use this parameter to specify the view ID for retrieving the elements or attributes that are in the view.
	Type: Integer
filter	This must be used with the view parameter. Use this parameter to pass the filter ID for retrieving elements and attributes based on the filter criteria.
	Type: Integer
	Example
	<code>view=14&filter=1</code>
tree	Set this parameter to view the element collection in the tree structure
	Type: Boolean
modifiedSince	Use this parameter to retrieve the element that have changed since a specified date. The following formats are supported. Note: The user can also supply <code>modifiedsince</code> as a parameter.
	<code>yyyy-MM-dd 'T' HH:mm:ss</code>
	<code>yyyy-MM-dd 'T' HH:mm</code>
	<code>yyyy-MM-dd</code>
	<code>yyyy-MM-dd HH:mm:ss</code>
	<code>yyyy-MM-dd HH:mm</code>
	Type: Date or DateTime

Parameter	Description
	Example
	2013-04-19T15:16:38+05:30
	2009-01-22T12:00:00+01:00
metadata	Set the parameter to <code>schema</code> to retrieve the schema of the element collection's XML document
	Type: String
	Example
	metadata=schema
fields	Use this parameter to pass XPath expression to retrieve specific set of attributes or to filter elements based on specific values of attributes.
	Type: XPath expression
	Example: To retrieve a specific set of attributes
	<code>fields=elementCollection/Element/attributes/(ID Title Owner)</code>
	Example: To filter elements based on attribute values
	<code>fields=elementCollection/Element/attributes/(Title contains X OR Status is Approved)</code>
	<code>fields=elementCollection/Element/attributes/(Title = X Status is Approved)</code>
	<code>fields=elementCollection/Element/attributes/(Title is X AND Status = Approved)</code>
	<code>fields=elementCollection/Element/attributes/(Title contains X AND Cost < 100)</code>

Parameter	Description
	<p data-bbox="566 268 662 302">Notes</p> <ul data-bbox="566 336 1396 414" style="list-style-type: none"> <li data-bbox="566 336 1396 414">▪ Attribute names and the attribute values should be enclosed in single quotes if they contain the following special characters: <p data-bbox="630 448 662 481">(</p> <p data-bbox="630 515 662 548">)</p> <p data-bbox="630 582 662 616"><</p> <p data-bbox="630 649 662 683">></p> <p data-bbox="630 716 662 750"> </p> <p data-bbox="630 784 662 817">=</p> <p data-bbox="598 851 1428 929">If the names or values already contain single quotes (') then they need to be escaped with a backslash as: \'</p> <p data-bbox="598 963 726 996">Example:</p> <p data-bbox="598 1008 1460 1086"><code>fields=elementCollection/Element/attributes/('Comments(incl)' contains 'Shift>Technology')</code></p> <ul data-bbox="566 1108 1412 1187" style="list-style-type: none"> <li data-bbox="566 1108 1412 1187">▪ The following characters or character sequences can be used as operators: <p data-bbox="630 1220 662 1254">=</p> <p data-bbox="630 1288 662 1321"><</p> <p data-bbox="630 1355 662 1388">></p> <p data-bbox="630 1422 662 1456">is</p> <p data-bbox="630 1489 750 1523">contains</p> <ul data-bbox="566 1556 1444 1635" style="list-style-type: none"> <li data-bbox="566 1556 1444 1635">▪ Different filtering conditions can be connected using the following connectors (case sensitive): <p data-bbox="630 1668 702 1702">AND</p> <p data-bbox="630 1736 686 1769">OR</p> <p data-bbox="630 1803 829 1836"> (same as OR)</p> <ul data-bbox="566 1870 1460 2027" style="list-style-type: none"> <li data-bbox="566 1870 1460 2027">▪ If you are using the fields parameter for filtering the elements, avoid providing modifiedSince parameter. Instead use the fields parameter with a filtering condition specified on Last Changed Date attribute. <ul data-bbox="566 2060 1460 2240" style="list-style-type: none"> <li data-bbox="566 2060 1460 2240">▪ Only the following types of attributes are supported for specifying the filtering conditions: Text, Integer, Float, Date, Choice, Check Box, MultChoice, Parent Folder, Created Date, Created By, Last Changed Date, Last Changed By, Owner, Link, Linklist, Incoming Link, Unique Id

Parameter	Description
	<ul style="list-style-type: none"> ▪ For attributes like Choice and MultiChoice you can specify the name of the items as the filtering value. For Link, Linklist and Incoming Link you need to specify the id of the target element as the filtering condition. For Parent Folder, Owner, Created By and Last Modified by attributes you can either specify the name or id of the target element. ▪ To filter based on more than one attribute value, provide the filtering condition by comma-separating the individual values. <p>Example:</p> <pre>fields=elementCollection/Element/attributes/(Geography contains Asia,Europe)</pre> <p>This will filter the elements whose Geography contains both Asia and Europe (applies 'all' condition).</p> <p>For multivalued attributes like MultiChoice and Linklist you can prefix the filtering condition with <code>or</code>, to specify the 'any' condition while filtering.</p> <p>Example:</p> <pre>fields=elementCollection/Element/attributes/(Geography contains or,Asia,Europe)</pre> <p>This will filter the elements whose Geography contains either Asia or Europe (applies 'any' condition).</p> <ul style="list-style-type: none"> ▪ For parent folder attribute prefix the filtering condition with <code>r</code>, to set recursive rule. <p>Example:</p> <pre>fields=elementCollection/Element/attributes/(Parent Folder is r,System)</pre> <p>This will filter all elements whose parent folder is System or any of its subfolders.</p>
<p>paging, pageSize, pageNo</p>	<p>Use these parameters to retrieve the collection of elements that are paginated. When pagination is enabled the XML document shows an additional link to the next page.</p>
	<p>Types</p>
	<p>paging: Boolean</p>
	<p>pageSize: Integer</p>
	<p>Note: The user can also supply pagesize as a parameter.</p>

Parameter	Description
	<p>pageNo: Integer</p> <p>Note: The user can also supply pageno as a parameter.</p>
	<p>Examples</p> <pre><fp:link href="https://fpserver.com:9443/fp/resources/workspaces /2/modules/1/elements/?pageno=1&pageSize=2&paging=true& includeAttributes=true" rel="next" Page="0"/> paging=true&pageSize=2&pageno=1</pre>
includeHistoryOfAttributes, start, end	<p>Set the parameter to “true”, and then specify the start and the end date to retrieve the history of the attributes.</p> <ul style="list-style-type: none"> ▪ yyyy-MM-dd'T'HH:mm:ss ▪ yyyy-MM-dd'T'HH:mm ▪ yyyy-MM-dd ▪ yyyy-MM-dd HH:mm:ss ▪ yyyy-MM-dd HH:mm
	<p>Types</p> <p>includeHistoryOfAttributes: Boolean</p> <p>start: Date or Date Time</p> <p>end: Date or Date time</p>
	<p>Examples</p> <pre>includeHistoryOfAttributes=true&start=2011-03- 22T12:00:00+01:00&end=2011-06- 22T12:00:00+01:00 includeHistoryOfAttributes=true&start=2011-03- 22T12:00+GMT+01:00&end=2011-06-22T12:00+GMT+01:00</pre>
genericTimeGrid	<p>Set the parameter to true for retrieving the time grid attribute sheets in a generic format. This can be used for reporting using Rational Insight.</p> <p>Note: The user can also supply generictimegrid as a parameter.</p>
	<p>Type: Boolean</p>
	<p>Example</p> <pre>genericTimeGrid=true</pre>

Parameter	Description
optimize	Set the parameter to true to optimize the content. If this is set to true, the response will not include rich text and formula information of the attributes. Also alternate urls to the element will not be provided in the response.
	Type: Boolean
	Example
	<code>optimize=true</code>
includeLinkTable	Set the parameter to true to include attributes from the link targets in the response. The view should have a link table defined for the link/link list attributes. All visible attributes shown in the link table set up will be part of the link/linklist attribute response.
	Type: Boolean
	Example
	<code>includeLinkTable=true</code>
sortID	Use this parameter to sort the elements in the collection based on an attribute in the view. Provide the id of the attribute as the value for the parameter.
	Type: Integer
	Example
	<code>sortID=100</code>
descending	Use this parameter to sort the elements in the collection in descending order. If this parameter is not specified, sorting will be done in ascending order.
	Type: Boolean
	Example
	<code>descending=true</code>

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.

POST method

Use this method to create new element to a module. Non-administrators can use view parameters to create. This example is on using the POST method to create new elements. The user must have administrator permission for creating elements in the module.

Example on using the POST method to create elements

URL

`https://fpserver.com:9443/fp/resources/workspaces/3/modules/1/elements/.xml`

Accept header

`application/xml`

Request body

```
<ns:newElement
xmlns:ns="https://fpserver.com:9443/fp/namespace/workspaces/3/modules/1/elements" xmlns:fp="http://schema.ibm.com/focalpoint/resources" > <ns:Title>
<fp:textValue xmlns:fp="http://schema.ibm.com/focalpoint/resources">karthi
1</fp:textValue> </ns:Title> <ns:MealPreference
value="NonVeg"></ns:MealPreference> <ns:type_is_a_folder
value="true"></ns:type_is_a_folder> <ns:Comments> <ns:newTextListEntry>
<fp:textValue xmlns:fp="http://schema.ibm.com/focalpoint/resources">text
entry 1</fp:textValue> </ns:newTextListEntry> <ns:newTextListEntry>
<fp:textValue xmlns:fp="http://schema.ibm.com/focalpoint/resources">text
entry 2</fp:textValue> </ns:newTextListEntry> </ns:Comments>
<ns:RelatedReleases> <fp:linkChange
value="https://fpserver.com:9443/fp/resources/workspaces/3/modules/3/elements/40"></fp:linkChange> <fp:linkChange
value="https://fpserver.com:9443/fp/resources/workspaces/3/modules/3/elements/41"></fp:linkChange> </ns:RelatedReleases><ns:Parent_Folder
value="https://fpserver.com:9443/fp/resources/workspaces/3/modules/1/elements/61"></ns:Parent_Folder> </ns:newElement>
```

The root node `<ns:newElement>` is based on the module name. For example: if the module name is products, then the new element is product.

Supported parameters for the POST method

Parameter	Type
optimize	Set this parameter to true to optimize element creation.
Type: Boolean	

This example uses the POST method to create new elements using a view. The non-administrator can add elements to the module using the views.

Example on using the POST method to create elements using views1

URL	https://fpserver.com:9443/fp/resources/workspaces/3/modules/1/elements/.xml?view=200&optimize=true
Accept header	application/xml
Request body	<pre><ns:newElement xmlns:ns="https://fpserver.com:9443/fp/namespace/workspaces/3/modules/1/views/200/elements" xmlns:fp="http://schema.ibm.com/focalpoint/resources" > ... </ns:newElement></pre> <p>The root node (<ns:newElement>) is based on the module name. For example: if the module name is products, then the new element is product.</p>

Multipart POST: This example uses the POST method to update multiple attributes through one HTTP request. The content type for this request is `multipart/related`. You can only use the POST method only to add links, delete links of an element.

URL	https://fpserver.com:9443/fp/resources/workspaces/5/modules/1/elements/.xml
Accept header	application/xml
Content type	multipart/related
Request body	<pre><attributeChangeCollection xmlns:ns="https://schema.com/focalpoint/resources" > <linkListChange href="/14/attributes/224"> <ns:addLink value="https://fpserver.com:9443/fp/resources/workspaces/5/module s/3/elements/20"></ns:addLink> <ns:addLink value="https://fpserver.com:9443/fp/resources/workspaces/5/module s/3/elements/19"></ns:addLink> <ns:deleteLink value="https://fpserver.com:9443/fp/resources/workspaces/5/module s/3/elements/18"></ns:deleteLink> </linkListChange> <textListEntryChange href="/14/attributes/223/entries/" > <ns:textValue>one</ns:textValue> </textListEntryChange> </attributeChangeCollection ></pre>

PUT method

Multipart PUT: This example is on using the PUT method to update multiple attributes through one HTTP request. The content type for this request is `multipart/related`.

URL	http://fpserver.com:9443/fp/resources/workspaces/5/modules/1/elements.xml
Accept header	application/xml
Content type	multipart/related
Request body	<pre><attributeChangeCollection xmlns:ns="http://schema.com/focalpoint/resources"> <textAttributeChange href="./14/attributes/2"> <ns:textValue>new text value</ns:textValue> </textAttributeChange> <textAttributeChange href="./15/attributes/2"> <ns:textValue>new text value</ns:textValue> </textAttributeChange> <textListEntryChange href="./14/attributes/223/entries/212309123" > <ns:textValue>replaced value</ns:textValue> </textListEntryChange> </attributeChangeCollection></pre>

Elements

An element contains all the attribute values of the element.

GET method

Use this method to retrieve the element details. You can use the View parameter to filter the attributes.

URL	https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/001.xml
Accept header	application/xml

URL <https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/001.xml>

Response body

```
-<ns:Element
xsi:schemaLocation="https://fpserver.com:9443/fp/namespace/workspaces/2/modules/1/element
https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/element.xsd"> <ns:attributes> <ns:ID writable="false">
<ns:uniqueId value=""/> <ns:alias>799af0ed-6316-4116-8749-
ce2353e80ac5</ns:alias> </ns:ID> <ns>Title writable="true">
<ns:text></ns:text> <ns:alias>7d5c25cb-e2a2-4d94-8b03-
lcf603d2833b</ns:alias> <ns:editLink
href="https://fpserver.com:9443/fp/resources/workspaces/2/module
s/1/elements/1/attributes/2"/> </ns>Title> <ns>Description
writable="true"></ns>Description> <ns:type_is_a_folder wr
itable="true"></ns:type_is_a_folder>
<ns:check_boxes_are_selected_in_the_chart
writable="true"></ns:check_boxes_are_selected_in_the_chart>
<ns:Element_Information writable="false"/> <ns:Owner
writable="true"></ns:Owner> <ns:Creator
writable="false"></ns:Creator> <ns:Created_Date
writable="false"></ns:Created_Date> <ns>Last_Changed_By
writable="false"></ns>Last_Changed_By> <ns>Last_Changed_Date
writable="false"></ns>Last_Changed_Date> <ns:Parent_Folder
writable="true"></ns:Parent_Folder> <ns:Text_List
writable="true"></ns:Text_List> </ns:attributes><ns:selfLink
href="https://fpserver.com:9443/fp/resources/workspaces/2/module
s/1/elements/1"/> <ns:mSPLink
href="https://fpserver.com:9443/fp/resources/workspaces/2/module
s/1/elements/1?format=mspxml"/> <ns:htmlLink
href="https://fpserver.com:9443/fp/servlet/Login?go=2,1"/>
<ns:alias>7602dc18-d64c-4360-8098-de0b0a385bf8</ns:alias>
</ns:Element>
```

Parameters	Type	Description	Example
view	Integer	Use this parameter to specify the view ID for retrieving the attributes that are in the view.	
filter	Integer	This must be used with the view parameter. Use this parameter to pass the filter ID for retrieving attributes based on the filter criteria.	view=14&filter=1
metadata	String	Set the parameter to <code>schema</code> to retrieve the schema of the element's XML document	metadata=schema

Parameters	Type	Description	Example
fields	XPath expression	Use this parameter to pass XPath expression to retrieve specific set of attributes.	fields=attributes/(ID Title Owner)
includeHistoryOfAttributes	Boolean	Set the parameter to true and specify the start and the end date to retrieve the history of the attributes.	includeHistoryOfAttributes=true&start=2011-03-
start	Date or Date Time	yyyy-MM-dd'T'HH:mm:ss	22T12:00:00+01:00&end=2011-06-
end	Date or Date time	yyyy-MM-dd'T'HH:mm	22T12:00:00+01:00
		yyyy-MM-dd	includeHistoryOfAttributes=true&start=2011-03-
		yyyy-MM-dd HH:mm:ss	22T12:00+GMT+01:00&end=2011-06-
		yyyy-MM-dd HH:mm	22T12:00+GMT+01:00
genericTimeGrid	Boolean	Set the parameter to true for retrieving the time grid attribute sheets in a generic format. This can be used for reporting using Rational Insight. Note: The user can also supply generictimegrid as a parameter.	genericTimeGrid=true
optimize	Boolean	Set the parameter to true to optimize the content. If this is set to true, the response will not include rich text and formula information of the attributes. Also alternate urls to the element will not be provided in the response.	optimize=true
includeLinkTable	Boolean	Set the parameter to true to include attributes from the link targets in the response. The view should have a link table defined for the link/link list attributes. All visible attributes shown in the link table set up will be part of the link/linklist attribute response.	includeLinkTable=true

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.

Attributes

The representation of an attribute contains the value of the attribute. Certain attributes have multiple collection of values and each value has attribute URIs. For example: TextList and File. The representation of the single attribute value also includes the edit links for each text entry in the list. You can change an attribute value of an element by using PUT, POST, or DELETE:

Use PUT operation to update these attribute types: CheckBox, Choice, Date, Float, Integer, Link, LinkList, Matrix, Multichoice, Text, Time Grid, URL, and UniqueId.

For TextList, LinkList and File attributes, use POST operation to add a text entry, link entry or a file.

Use PUT operation to update a text entry, link entry or file, and use DELETE operation to remove a text entry or file.

For a Version attribute, use POST operation to increment to the next major version.

The XML format to update an attribute value is different from 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.

GET method

Use this method to retrieve the attribute details.

URL	https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/15/attributes/220.xml
Accept header	application/xml

URL <https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/15/attributes/220.xml>

Response body

```
<ns:Text_List writable="true"
xsi:schemaLocation="https://fpserver.com:9443/fp/namespace/workspaces/2/modules/1/attribute
https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/15/attributes/220/attribute.xsd"> <ns:textList>
<fp:textListEntry createdDate="2013-05-02T13:50:09+05:30" lastChangedDate="2013-05-02T13:50:09+05:30"> <fp:author
title="Admin"/><fp:lastChangedBy title="Admin"/>
<fp:textValue>Sample Text 1</fp:textValue>
<fp:formattedTextValue>Sample Text 1<BR></fp:formattedTextValue>
<fp:xhtmlTextValue><div>Sample Text 1</div></fp:xhtmlTextValue>
<fp:link
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/15/attributes/220/entries/1367482809131"
rel="edit"/> </fp:textListEntry> <fp:textListEntry
createdDate="2013-05-02T13:50:17+05:30" lastChangedDate="2013-05-02T13:50:17+05:30"> <fp:author
title="Admin"/><fp:lastChangedBy title="Admin"/>
<fp:textValue>Sample Text 2</fp:textValue>
<fp:formattedTextValue>Sample Text 2</fp:formattedTextValue>
<fp:xhtmlTextValue><div>Sample Text 2</div></fp:xhtmlTextValue>
<fp:link
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/15/attributes/220/entries/1367482817756"
rel="edit"/> </fp:textListEntry> </ns:textList>
<ns:alias>ed2ad85f-5b87-4704-b384-e041888439c0</ns:alias>
<ns:editLink
href="https://fpserver.com:9443/fp/resources/workspaces/2/modules/1/elements/15/attributes/220"/> </ns:Text_List>
```

Parameters	Type	Description	Example
view	Integer	Use this parameter to specify the view ID for retrieving the attributes that are in the view.	
filter	Integer	This must be used with the view parameter. Use this parameter to pass the filter ID for retrieving elements and attributes based on the filter criteria.	view=14&filter=1
metadata	String	Set the parameter to <code>schema</code> to retrieve the schema of the attribute's XML document	metadata=schema

Parameters	Type	Description	Example
includeHistoryOfAttributes	Boolean	Set the parameter to true and specify the start and the end date to retrieve the history of the attributes.	includeHistoryOfAttributes=true&start=2011-03-
start	Date or Date Time	yyyy-MM-dd'T'HH:mm:ss	22T12:00:00+01:00&end=2011-06-
end	Date or Date time	yyyy-MM-dd'T'HH:mm	22T12:00:00+01:00
		yyyy-MM-dd	includeHistoryOfAttributes=true&start=2011-03-
		yyyy-MM-dd HH:mm:ss	22T12:00+GMT+01:00&end=2011-06-
		yyyy-MM-dd HH:mm	22T12:00+GMT+01:00
generictimegrid	Boolean	Set the parameter to true for retrieving the time grid attribute sheets in a generic format. This can be used for reporting using Rational Insight.	generictimegrid=true

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.

POST method

Use this method to create new entry to the attribute collection. This example is on using the POST method to add new values to an attribute TextList collection. The user must have administrator permission for adding values to the attribute collection.

URL <http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/53/attributes/237.xml>

Accept header application/xml

Request body

```
<textListEntryChange
xmlns:ns="http://schema.ibm.com/focalpoint/resources">
<ns:textValue>one</ns:textValue> </textListEntryChange>
```

This example uses the POST method to add new values to a File attribute.

URL <http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/53/attributes/228.xml>

Accept header application/xml

URL <http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/53/attributes/228.xml>

Request body

```
<fileChange
xmlns:ns="http://schema.ibm.com/focalpoint/resources">
<ns:fileName>a.txt</ns:fileName>
<ns:contentType>plain/text</ns:contentType>
<ns:fileData>YWJj</ns:fileData> </fileChange>
```

This example uses the POST method to add new values to a LinkList attribute and deleting a link.

URL <http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/53/attributes/236.xml>

Accept header application/xml

Request body

```
<linkListChange
xmlns:ns="http://schema.ibm.com/focalpoint/resources">
<ns:addLink
value="http://fpserver.com:8091/fp/resources/workspaces/2/modules/13/elements/30"> </ns:addLink>
<ns:deleteLink
value="http://fpserver.com:8091/fp/resources/workspaces/2/modules/13/elements/29"></ns:deleteLink> </linkListChange>
```

This example uses the POST method to update the value of the Version Attribute to the next major version.

URL <http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/53/attributes/245.xml>

Accept header application/xml

Request body

```
<VersionAttribute nextMajorVersion="true"></VersionAttribute>
```

PUT methods

This example uses the PUT method to update the attribute value.

Attribute type Request body

Check Box, Lock

```
<checkBoxChange value="false"> </checkBoxChange>
```

Choice

```
<choiceAttributeChange value="Closed">
</choiceAttributeChange>
```

Date

```
<dateChange
xmlns:ns="http://schema.ibm.com/focalpoint/resources">
<ns:value>2011-01-01T00:00:00+05:30</ns:value> </dateChange>
```

Attribute type	Request body
Float	<pre><floatChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:value>201</ns:value> </floatChange></pre>
File	<p>URL example:</p> <pre>http://localhost:8080/fp/resources/workspaces/3/modules/13/elements/16/attributes/237/files/01.xml <fileChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:fileName>a.txt</ns:fileName> <ns:contentType>plain/text</ns:contentType> <ns:fileData>YWJj</ns:fileData> </fileChange></pre>
Link	<pre><linkValueChange value="http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/16"> </linkValueChange></pre>
Integer	<pre><integerChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:value>201</ns:value> </integerChange></pre>
List (Link)	<pre><linkChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:linkChange value="http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/72"> </ns:linkChange> <ns:linkChange value="http://fpserver.com:8091/fp/resources/workspaces/2/modules/1/elements/73"> </ns:linkChange> </linkChange></pre>
List (Text)	<p>URL example:</p> <pre>http://localhost:8080/fp/resources/workspaces/3/modules/13/elements/19/attributes/247/entries/1369815257240.xml <textListEntryChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:textValue>one</ns:textValue> </textListEntryChange></pre>
Matrix	<pre><matrixChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:row> <ns:cell> <ns:value>A</ns:value> </ns:cell> <ns:cell> <ns:value>B</ns:value> </ns:cell> </ns:row> <ns:row> <ns:cell> <ns:value>Row 1</ns:value> </ns:cell> <ns:cell> <ns:value>Simple text value</ns:value> </ns:cell> <ns:cell> <ns:expression>1 + 2</ns:expression> </ns:cell> </ns:row> </matrixChange></pre>
MultiChoice	<pre><multichoiceAttributeValues xmlns:ns="http://fpserver.com:8091/fp/namespace/workspaces/2/modules/1/attribute"> <ns:selected>Open</ns:selected> <ns:selected>One</ns:selected> </multichoiceAttributeValues></pre>

Attribute type	Request body
Text	<pre><textAttributeChange xmlns:ns="http://schema.ibm.com/focalpoint/resources"> <ns:textValue>new text value</ns:textValue> </textAttributeChange></pre>
Timegrid (Header)	<pre><timeGridChange xmlns:ns="http://fpserver.com:8091/fp/namespace/workspaces/2 /modules/1/attribute"> <ns:timeGridSetting sheetId="1" startDate="2011-12-31" endDate="2012-03-30"> </ns:timeGridSetting> </timeGridChange></pre>
TimeGrid (Data)	<pre><timeGridChange xmlns:ns="http://fpserver.com:8091/fp/namespace/workspaces/2 /modules/1/attribute"> <ns:cell sheetId="1" date="2011-12- 31" columnId="A" rowId="1"> <ns:value>23</ns:value> </ns:cell> <ns:cell sheetId="1" date="2011-12-31" columnId="A" rowId="2"> <ns:value>24</ns:value> </ns:cell> </timeGridChange></pre>
UniqueID	<pre><uniqueIdChange value="5671"> </uniqueIdChange></pre>
URL	<pre><urlChange value="www.ibm.com/developerworks"> </urlChange></pre>

DELETE method

This example uses the DELETE method to delete the attribute value.

Attribute type	Request body
File	<p>URL example:</p> <pre>http://localhost:8080/fp/resources/workspaces/3/modules/13/e lements/16/attributes/237/files/01.xml</pre> <p>To delete the 01 file, specify this URL and use the delete command.</p>
List (Text)	<p>URL example:</p> <pre>http://localhost:8080/fp/resources/workspaces/3/modules/13/e lements/19/attributes/247/entries/1369815257240.xml</pre> <p>To delete the entry from the list text attribute.</p>



www.unicomsi.com

We welcome feedback on our documentation. Please email us at:
tech.authors@unicomsi.com

www.unicomglobal.com