



Andrew White

Integrating WebSphere Service Registry and Repository with WebSphere DataPower

In this IBM® Redpapers™ publication, we discuss the built-in integration between IBM WebSphere Service Registry and Repository (WSRR) and IBM WebSphere® DataPower®, which allows users to create Web Service Proxies based on Web Services Description Language (WSDL) documents and attached Web Service Policy (WS-Policy) documents stored in WSRR.

We also describe how to extend the built-in integration between WSRR and WebSphere DataPower to allow dynamic endpoint selection of services based upon meta data stored in WSRR.

Product description

WebSphere DataPower provides the ability to understand and act upon application data as it traverses the network. While this application awareness is not, in itself, a new networking concept, XML has accelerated its appeal and complexity, that is, application awareness comes with many security, complexity, and performance challenges. As a result, a new genre of hardened software, hardware, and XML-centric appliances has arisen to bridge this gap.

These WebSphere DataPower appliances focus on providing consumability, performance, and hardened security. They can extend the Enterprise Service Bus (ESB) into the network and also provide an service-oriented architecture (SOA) gateway for business-to-business integration.

Value proposition

Integrating WebSphere Service Registry and Repository with WebSphere DataPower provides several benefits:

- ▶ Allows the creation of Web Service Proxy based upon a WSDL stored in WSRR instead of a static file. If the WSDL is updated in WSRR, then the Web Service Proxy synchronizing with that particular WSDL will also be updated the next time WebSphere DataPower synchronizes with WSRR.
- ▶ Using WSRR's policy attachment user interface, users can edit WSDL documents attached to WS-Policies and the results of this edit will be updated in WebSphere DataPower the next time synchronization occurs.
- ▶ By extending the built-in integration between WSRR and WebSphere DataPower, it is possible to configure a Web Service Proxy to dynamically choose an endpoint at run time based on meta data stored in WSRR.

Key integration features

The built-in WebSphere Service Registry and Repository subscription feature in WebSphere DataPower enables WebSphere DataPower to subscribe to a particular concept or WSDL. The subscription can be configured to either synchronize at a timed interval (poll) or when a user logs into WebSphere DataPower and manually runs the synchronization.

Web Services Proxies can then be created in WebSphere DataPower based upon a WebSphere Service Registry and Repository subscription. Any changes made to either the WSDL or to any attached Web Service Policies will be reflected in WebSphere DataPower the next time a synchronization takes place, as shown in Figure 1 on page 3.

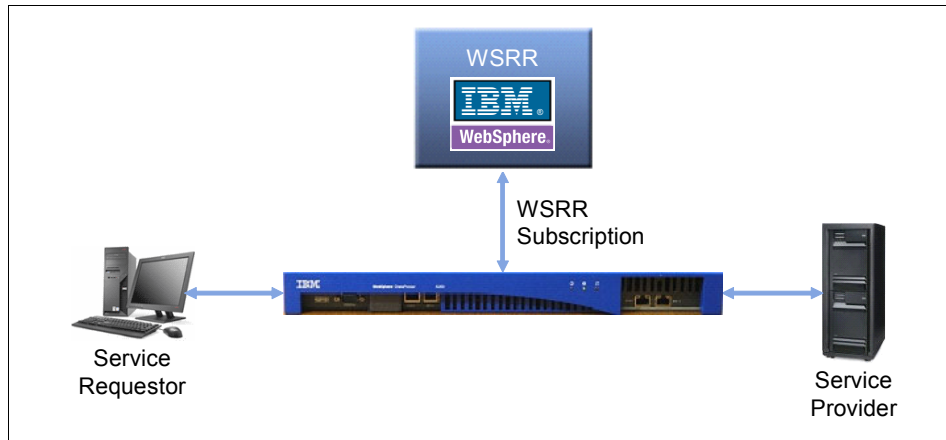


Figure 1 WebSphere DataPower subscription to WSRR

Creating Web Services Proxies based on WSRR subscriptions is covered in Chapter 4, “Using Web Service Proxy with WebSphere Registry and Repository”, in *IBM WebSphere DataPower SOA Appliances Part IV: Management and Governance*, REDP-4366.

WebSphere DataPower can be customized to allow dynamic service endpoint selection based upon meta data stored in WSRR, as shown in Figure 2.

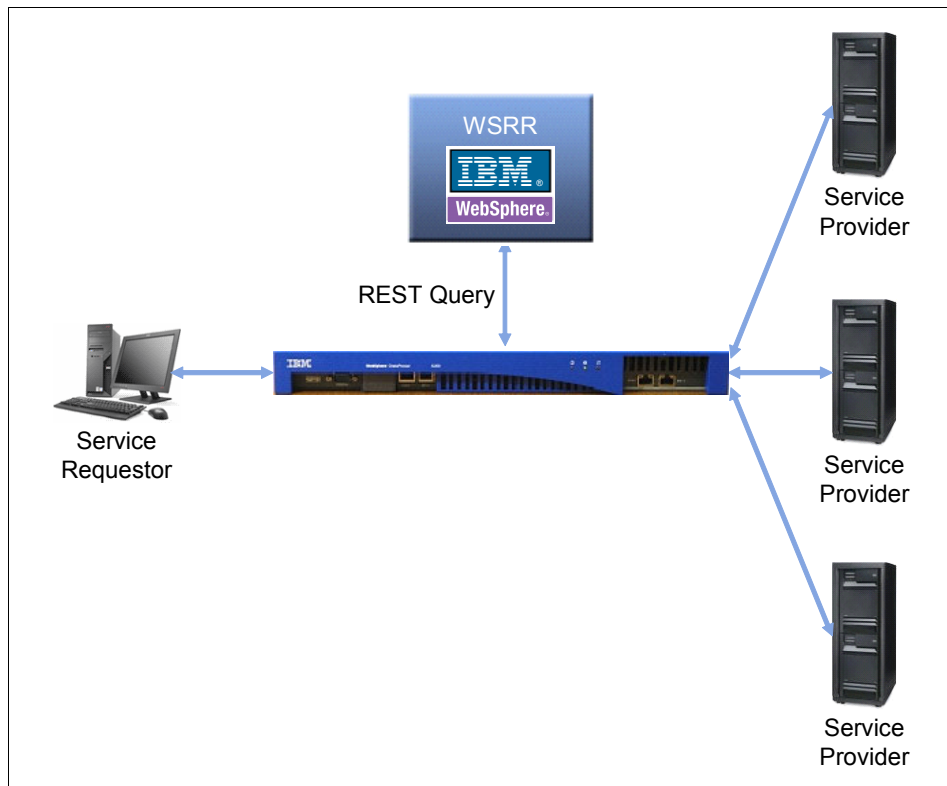


Figure 2 Dynamic endpoint selection using WSRR and WebSphere DataPower

Representational state transfer (REST) queries can be dynamically created at run time to query WSRR for either documents, meta data, or both, and then perform routing decisions made based on the results of those queries.

Dynamic endpoint selection scenario

This scenario describes how a fictitious company, JKHL Enterprises, uses WebSphere DataPower to dynamically retrieve a service endpoint stored in the organization's service repository hosted on WSRR.

Scenario overview

JKHL Enterprises currently has an Account Creation service running in their production environment. This service has recently been upgraded to run an additional verifyCreation operation, as shown in Figure 3. This minor upgrade of the service is backwards compatible with the first version.

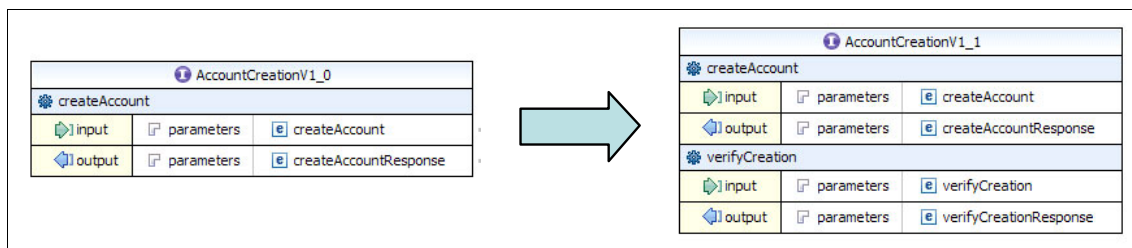


Figure 3 Upgrade to service interface

JKHL Enterprises uses WSRR as their service repository. To shield applications from changes to the Account Creation service, JKHL utilizes a XML firewall in WebSphere DataPower that will look up the *active* endpoint of the Account Creation service in WSRR.

Once Version 1_1 of the Account Creation service is deployed and available, the endpoint for Version 1_0 will be classified as offline. All service consumer requests for the Account Creation service will be directed to the endpoint for Version 1_1, enabling Version 1_0 to be deprecated with no impact on service consumers.

Note: In this minor upgrade scenario, there is no change to the name space of the service.

Integration issues

JKHL Enterprises uses dynamic endpoints for service calls in WebSphere DataPower.

In this endpoint lookup scenario, WSRR will be accessed to retrieve a single matching endpoint for the latest version of the createAccount operation of the AccountCreation service.

Uploading service definitions is explained in *Service Lifecycle Governance Using WebSphere Service Registry and Repository* and *Service Lifecycle Governance with IBM WebSphere Service Registry and Repository V6.3*, SG24-7793 and will not be described in this paper.

Solution overview

The endpoint lookup solution shown in Figure 4 shows how the WebSphere DataPower queries WSRR and selects the account creation service endpoint that is set to be online in WSRR.

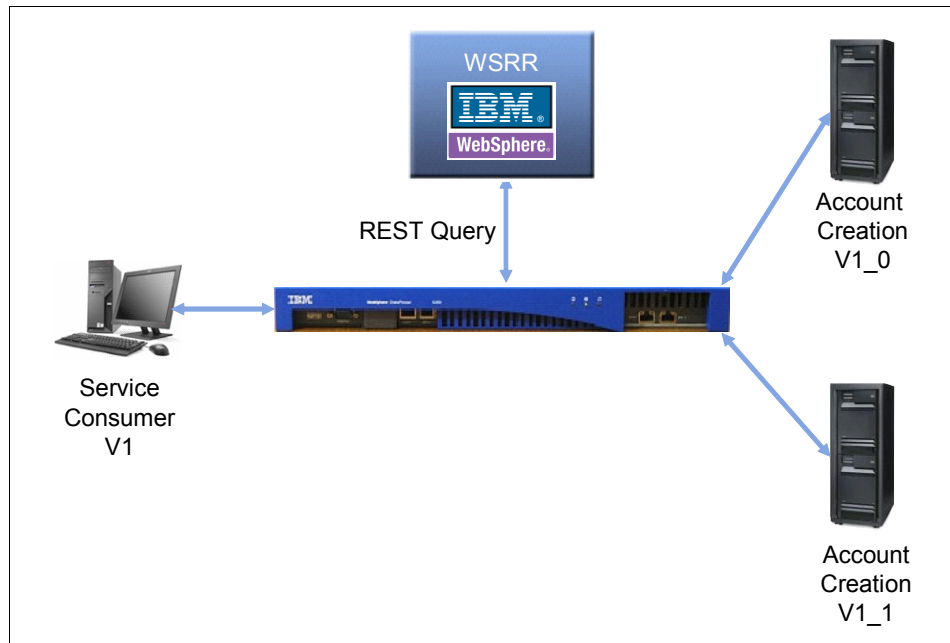


Figure 4 Scenario overview

Performing the integration

Note: In this scenario, the public/private key that WSRR uses is the same public/private key that the production endpoint uses and is the same key store DataPower uses for its HTTPS interface.

Perform the following steps to accomplish the integration:

1. In this scenario, security is enabled on WSRR and DataPower provides a HTTPS interface. We must obtain the WSRR public key certificate so that DataPower can communicate with WSRR over the HTTPS protocol, and obtain the public/private pair key store that DataPower will use for the HTTPS interface so that the interface will be accessible by consumers.
 - a. Log in to the WebSphere Application Server Integration Solution console.
 - b. Select **Security** → **SSL certificate and key management**.
 - c. Click **Key stores and certificates**.
 - d. Click **NodeDefaultKeyStore**.
 - e. Click **Personal certificates**.
 - f. Select the **Default** check box and click **Extract...**
 - g. In certificate file name field, enter /tmp/WSRRCert.der, select **Binary DER data**, and click **OK**.

Note: You *must* enter an absolute path in the certificate file name field.

- h. Copy the WSRRCert.der and key.p12 files to the local file system.

Note: The key.p12 file is located in
/WAS_HOME/profiles/PROFILE_NAME/config/cells/CELL_NAME/nodes/NODE_NAME/key.p12.

2. Create a Multi-Protocol Gateway.
 - a. Log in to the DataPower Web console.
 - b. Click **Multi-Protocol Gateway**.
 - c. Click **Add**.

- d. Enter AccountCreation in the Multi-Protocol Gateway Name field. Click **dynamic-backend**, select **default** in the XML Manager drop-down menu, and select **default** from Multi-Protocol Gateway Policy drop-down menu, as shown in Figure 5 on page 9.

Configure Multi-Protocol Gateway

General | Advanced | Stylesheet Params | Headers | Monitors | WS-Addressing | WS-ReliableMessaging

Apply | Cancel | Help

General Configuration

Multi-Protocol Gateway Name
 *

XML Manager
 + ... *

Summary

Multi-Protocol Gateway Policy
 + ... *

Type
 dynamic-backend
 static-backend *

URL Rewrite Policy
 + ...

Back side settings

With a dynamic proxy back end Multi-Protocol Gateway type, the back end server address and port are determined by a stylesheet in a policy action.

User Agent settings

Match	Property
Note: To edit the User Agent, please access via the XML Manager above.	

SSL Client Crypto Profile
 + ...

Response Type
 Non-XML
 Pass-Thru
 SOAP
 XML

Back attachment processing format
 Dynamic
 MIME
 DIME
 Detect

Back Side Timeout
 *

Stream Output to Back
 Buffer Messages
 Stream Messages

HTTP Version to Server
 HTTP 1.0
 HTTP 1.1

Propagate URI
 on off

Front side settings

Front Side Protocol

 Add + ... *

Request Type
 Non-XML
 Pass-Thru
 SOAP
 XML

Front attachment processing format
 Dynamic
 MIME
 DIME
 Detect

Front Side Timeout
 *

Stream Output to Front
 Buffer Messages
 Stream Messages

Figure 5 Creating the Multi-Protocol Gateway

- e. In the Front Side Protocol field, click +.
- f. Click **HTTPS (SSL) Front Side Handler**.
- g. In the new window that appears, enter HTTPSProdFSH in the Name field, as shown in Figure 6 on page 11. In the SSL Proxy field, click +.

Configure HTTPS (SSL) Front Side Handler

Main

HTTPS (SSL) Front Side Handler

[Help](#)

Name *

Admin State enabled disabled

Comments

Local IP Address *

Port Number *

HTTP Version to Client

Allowed Methods and Versions

- HTTP 1.0
- HTTP 1.1
- POST method
- GET method
- PUT method
- HEAD method
- OPTIONS
- TRACE method
- DELETE method
- URL with Query Strings
- URL with Fragment Identifiers
- URL with ..
- URL with cmd.exe

Persistent Connections on off

Compression on off

Maximum Allowed URL Length

Maximum Allowed Total Header Length

Maximum Number of HTTP Request Headers Allowed

Maximum Allowed Length of HTTP Header Name

Maximum Allowed Length of HTTP Header Value

Maximum Allowed Length of HTTP Query String

SSL Proxy *

Access Control List

Figure 6 Creating the Front Side Handler

- h. In the new window that appears, enter SSLProxyProfile in the Name field. Select **Reverse** from the SSL Direction drop-down menu and click + next to the Reverse (server) Crypto profile, as shown in Figure 7.

Configure SSL Proxy Profile

Main

SSL Proxy Profile

[Help](#)

Name *

Admin State enabled disabled

SSL Direction *

Reverse (Server) Crypto Profile + ... *

Server-side Session Caching on off

Server-side Session Cache Timeout seconds

Server-side Session Cache Size entries (x 1024)

Client Authentication Is Optional on off

Always Request Client Authentication on off

Figure 7 Configuring the SSL Proxy Profile

- i. In the new window that appears, enter ReverseCryptoProfile. Click + next to the Identification Credentials field, as shown in Figure 8.

Configure Crypto Profile

Main

Crypto Profile

[Help](#)

Name *

Admin State enabled disabled

Identification Credentials

Validation Credentials

Ciphers

Options

- Enable default settings
- Disable SSL version 2
- Disable SSL version 3
- Disable TLS version 1

* on off

Send Client CA List on off

Figure 8 Creating the Crypto Profile

- j. In the new window that appears, enter CryptoIDCreds in the Name field. Click + next to the Crypto Key field, as shown in Figure 9.

Configure Crypto Identification Credentials

Main

Crypto Identification Credentials

Apply Cancel Help

Name CryptoIDCreds *

Admin State enabled disabled

Crypto Key (none) + ... *

Certificate (none) + ... *

Intermediate CA Certificate (empty)

Figure 9 Creating the Crypto Identification Credentials

- k. In the new window that appears, click **Upload**, as shown in Figure 10 on page 15.

Configure Crypto Certificate

Main

Crypto Certificate

Apply Cancel Help

Name admin *

Admin State enabled disabled

File Name cert:/// *

Password

Password Alias on off

Ignore Expiration Dates on off

Figure 10 Creating the Crypto Certificate

- l. In the new window that appears, enter the location of the key .p12 file that you copied from step 1. Click **Upload**, click **Continue**, and the window will close.
- m. Enter the password to the key store in the Configure Crypto Key field (by default, this is WebAS). Click **Apply** and the window will close.
- n. In the Configure Crypto Identification Credentials window, click + next to the Certificate field.

- d. Click + next to the Multi-Protocol Gateway policy.
- e. In the new window that appears, enter AccountCreationPolicy, select **Client to Server** from the Rule Direction drop-down menu, and click **New Rule**, as shown in Figure 12.

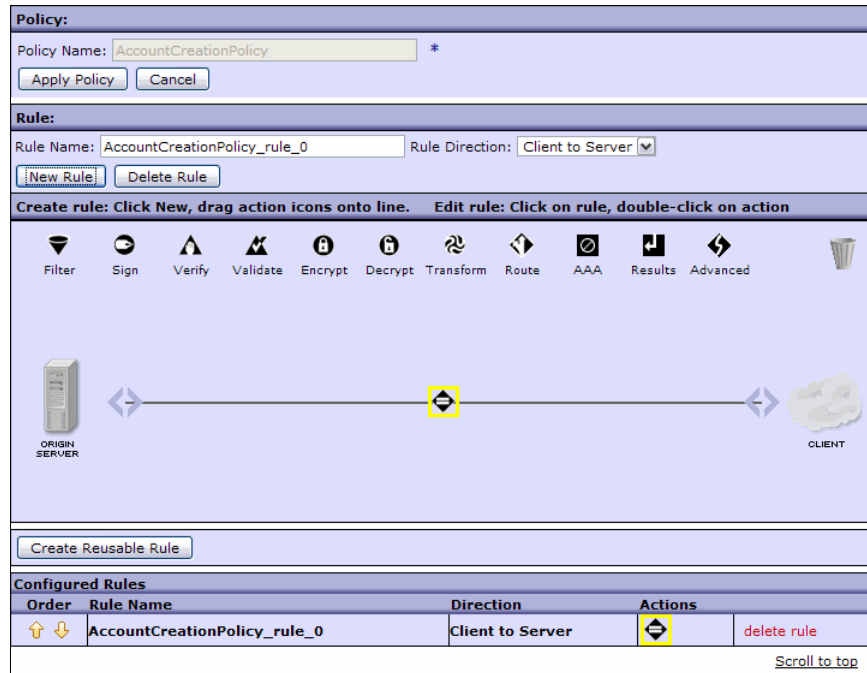


Figure 12 Creating a Multi-Protocol Gateway policy rule

- f. Double-click the **Match** action.
- g. In the new window that appears, click +. In the new window that appears, enter MatchRule in the Name field, click **Matching Rule**, and click **Add**. In the new window that appears, select **URL**, enter *, and click **Apply**. The window will close. In the Configure Matching Rule window, click **Apply** and the window will close. In the Configure Match Action window, click **Apply** and the window will close. Click **Done**.

h. Click **Advanced** and drag the icon onto the line, as shown in Figure 13.

Policy:
Policy Name: AccountCreationPolicy *
Apply Policy Cancel

Rule:
Rule Name: AccountCreationPolicy_rule_0 Rule Direction: Client to Server
New Rule Delete Rule

Create rule: Click New, drag action icons onto line. Edit rule: Click on rule, double-click on action

Filter Sign Verify Validate Encrypt Decrypt Transform Route AAA Results Advanced

ORIGIN SERVER CLIENT

Create Reusable Rule

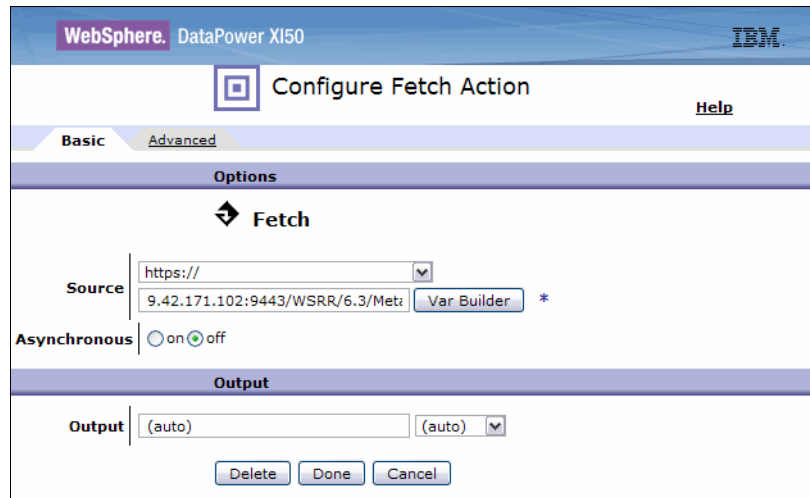
Configured Rules				
Order	Rule Name	Direction	Actions	
↑ ↓	AccountCreationPolicy_rule_0	Client to Server	Advanced	delete rule

Scroll to top

Figure 13 Inserting an advanced action into the rule

i. Double-click the **Advanced** icon previously dragged onto the line. In the new window that appears, select **Fetch** and click **Next**.

- j. Select **https://** from the Source drop-down menu and enter `https://9.42.171.102:9443/WSRR/6.3/Metadata/XML/PropertyQuery?query=/WSRR/GenericObject[@namespace=%27http://jkhle.itso.ibm.com/AccountCreationV1/service%27%20and%20exactlyClassifiedByAllOf%28%27http://www.ibm.com/xmlns/prod/serviceregistry/v6r3/ServiceModel%23SOAPServiceEndpoint%27,%27http://www.ibm.com/xmlns/prod/serviceregistry/lifecycle/v6r3/LifecycleDefinition%23online%27%29]&p1=name` in the text box, as shown in Figure 14. Click **Done** and the window will close.



The screenshot shows the 'Configure Fetch Action' dialog box in the WebSphere DataPower XI50 interface. The 'Source' field is set to 'https://' and the text box contains the URL '9.42.171.102:9443/WSRR/6.3/Met:'. The 'Asynchronous' option is set to 'off'. The 'Output' field is set to '(auto)'. Buttons for 'Delete', 'Done', and 'Cancel' are visible at the bottom.

Figure 14 Configuring the fetch action

Note: This URL is a REST action that queries WSRR to select SOAP Service Endpoints, which are classified as online and have the name space `http://jkhle.itso.ibm.com/AccountCreationV1/service`.

- k. Click the **Advanced** icon and drag it onto the line, as shown in Figure 15.

Policy:
 Policy Name: AccountCreationPolicy *
 Apply Policy Cancel

Rule:
 Rule Name: AccountCreationPolicy_rule_0 Rule Direction: Client to Server
 New Rule Delete Rule

Create rule: Click New, drag action icons onto line. Edit rule: Click on rule, double-click on action

Filter Sign Verify Validate Encrypt Decrypt Transform Route AAA Results Advanced

ORIGIN SERVER CLIENT

Create Reusable Rule

Configured Rules			
Order	Rule Name	Direction	Actions
↑ ↓	AccountCreationPolicy_rule_0	Client to Server	Filter Sign Verify Validate Encrypt Decrypt Transform Route AAA Results Advanced delete rule

[Scroll to top](#)

Figure 15 Adding another advanced action

- l. Double-click the **Advanced** icon previously dragged onto the line. In the new window that appears, select **Extract using Xpath** and click **Next**.

- m. Enter `/resources/resource/properties/property/@value ()` in the XPath field. Click **Var Builder**, enter JKHLE into the Context field, and enter Endpoint into the Variable Name field, as shown in Figure 16.

The screenshot shows the 'Configure Extract Using XPath Action' configuration page in WebSphere DataPower XI50. The 'Basic' tab is active. The 'Input' field is set to 'dpvar_1'. Under the 'Options' section, the 'Extract Using XPath' configuration is shown. The 'XPath' field contains the expression `/resources/resource/properties/pr`. Below this, the 'Variable Name' section is expanded, showing a 'Custom User Variable' table with the following data:

Variable Root	Context	Variable Name
var://context/	JKHLE	Endpoint

Below the table, there are sections for 'Extension Variables', 'Service Variables', and 'System Variables'. The 'Asynchronous' option is set to 'off'. The 'Output' field is set to '(auto)'. Buttons for 'Delete', 'Done', and 'Cancel' are at the bottom.

Figure 16 Extracting the previous REST query using an XPath expression

Note: This XPath query will give you the actual SOAP endpoint to which you route.

n. Click **Use Custom**, as shown in Figure 17.

The screenshot shows the 'Configure Extract Using XPath Action' configuration page in WebSphere DataPower XI50. The page is divided into sections: **Input**, **Options**, and **Output**. The **Input** section shows 'Input' set to 'dpvar_1'. The **Options** section is titled 'Extract Using XPath' and includes 'XPath' set to '/resources/resource/properties/prt', 'Variable Name' set to 'context/JKHLE/Endpoint', and 'Asynchronous' set to 'off'. The **Output** section shows 'Output' set to '(auto)'. Buttons for 'Delete', 'Done', and 'Cancel' are at the bottom.

Figure 17 Setting the result of the XPath expression to be a variable

- o. Click **Done** and the window will close. Click the **Route** icon and drag it onto the line, as shown in Figure 18.

Policy:
 Policy Name: AccountCreationPolicy *
 Apply Policy Cancel

Rule:
 Rule Name: AccountCreationPolicy_rule_0 Rule Direction: Client to Server
 New Rule Delete Rule

Create rule: Click New, drag action icons onto line. Edit rule: Click on rule, double-click on action

Filter Sign Verify Validate Encrypt Decrypt Transform Route AAA Results Advanced

ORIGIN SERVER → [Action: Match] → → → [Route] → CLIENT

Double click to configure.

Create Reusable Rule

Order	Rule Name	Direction	Actions	
↑ ↓	AccountCreationPolicy_rule_0	Client to Server	[Match] [Route] [Advanced]	delete rule

[Scroll to top](#)

Figure 18 Adding a route action

- p. Double-click the **Route** icon previously dragged onto the line. In the new window that appears, select **Use Variable to Select Destination**. Select **var://** and click **Var Builder**, enter JKHLE into the Context field, and enter Endpoint into the Variable Name field, as shown in Figure 19.

The screenshot shows the 'Configure Route (Using Variable) Action' dialog box in the WebSphere DataPower XI50 interface. The 'Options' tab is active, and the 'Route (Using Variable)' configuration is shown. Under 'Selection Method', the radio button for 'Use Variable to Select Destination' is selected. Below this, a dropdown menu shows 'var://' and a 'Var Builder' button is visible. The 'Destination' section contains a table with columns for 'Variable Root', 'Context', and 'Variable Name'. The 'Variable Root' is 'var://context/', 'Context' is 'JKHLE', and 'Variable Name' is 'Endpoint'. There is a 'Use Custom' button next to the 'Variable Name' field. Below the table are sections for 'Extension Variables', 'Service Variables', and 'System Variables'. The 'SSL Cred' is set to '(none)' with a '+' button and a 'Var Builder' button. The 'Asynchronous' option is set to 'off'. At the bottom are 'Delete', 'Done', and 'Cancel' buttons.

Figure 19 Routing using the variable

- q. Click **Use Custom**.
- r. Click **+**.

Note: This will create a SSL profile to encrypt the request to the SOAP Endpoint.

- s. In the window that appears, enter ForwardSSLProxy in the Name field, select **Forward** in the SSL Direction drop-down menu, and click **+**.
- t. In the new window that appears, enter ForwardCryptoProfile into the Name field and click **+** next to the Validation Credentials field.

- u. In the new window that appears, enter ForwardCryptoCreds in the Name field and click +.
 - v. In the new window that appears, enter ForwardCert in the Name field and click **Upload...**
 - w. Enter the location of the WSRRCert.der file (from step 1 on page 7), click **Upload**, click **Continue**, and the window will close.
 - x. In the Configure Crypto Certificate window, click **Apply** and the window will close.
 - y. In the Configure Crypto Validation Credentials window, click **Apply** and the window will close.
 - z. In the Configure Crypto Profile window, click **Apply** and the window will close.
 - aa. In the Configure SSL Proxy Profile window, click **Apply** and the window will close.
 - ab. In the Configure Route (Using Variable) Action window, click **Done** and the window will close.
 - ac. Click **Apply Policy** and click **Close Window**.
 - ad. In the Configure Multi-Protocol Gateway, click **Apply**.
4. DataPower will try and communicate with WSRR without using either the correct SSL certificate or the correct user information. The Multi-Protocol Gateway will have to be configured to communicate with the secure WSRR.
- a. Log in to the DataPower Web console.
 - b. Click **Multi-Protocol Gateway**.
 - c. Click **AccountCreation**.
 - d. Click + next to the XML Manager field.

e. Enter WSRRXMLManger in the Name field, as shown in Figure 20.

The screenshot shows the 'Configure XML Manager' interface. At the top, there is a navigation bar with tabs: 'Main', 'XML Parser', 'Document Cache', 'Extension Functions', and 'Document Cache Policy'. Below the navigation bar, the title 'XML Manager' is displayed. There are 'Apply' and 'Cancel' buttons on the left, and a 'Help' link on the right. The 'Name' field is filled with 'WSRRXMLManger' and has an asterisk next to it. Below this, there are several configuration options:

- Admin State:** Radio buttons for 'enabled' (selected) and 'disabled'.
- Comments:** An empty text input field.
- URL Refresh Policy:** A dropdown menu showing '(none)', followed by '+' and '...' buttons.
- Compile Options Policy:** A dropdown menu showing '(none)', followed by '+' and '...' buttons.
- XSL Cache Size:** A text input field containing '256', with 'stylesheets' written to its right.
- SHA1 Caching:** Radio buttons for 'on' (selected) and 'off'.
- Static Document Call:** Radio buttons for 'on' (selected) and 'off'.
- XSLT Expression Optimization:** Radio buttons for 'on' and 'off'.
- Load Balance Groups:** A dropdown menu showing '(empty)', followed by 'Add', '+', and '...' buttons.
- User Agent Configuration:** A dropdown menu showing 'default', followed by '+', '...', and an asterisk.

Figure 20 Creating a new XML manager

f. Click + next to the User Agent Configuration field.

- g. In the new window that appears, enter **WSRRConnection** in the Name field, as shown in Figure 21.

The screenshot shows a web-based configuration window titled "Configure User Agent". At the top, there is a navigation bar with tabs: "Main", "Proxy Policy", "SSL Proxy Profile Policy", "Basic-Auth Policy", and "Soap-Action Policy". The "Main" tab is selected. Below the navigation bar, the text "User Agent" is displayed. There are "Apply" and "Cancel" buttons on the left, and a "Help" link on the right. The main configuration area contains the following fields:

- Name:** A text input field containing "WSRRConnection" with an asterisk (*) to its right.
- Admin State:** Radio buttons for "enabled" (selected) and "disabled".
- Comments:** An empty text input field.
- HTTP Request-Header:** An empty text input field.
- Maximum Redirects:** A text input field containing the number "8".
- Timeout:** A text input field containing the number "300".

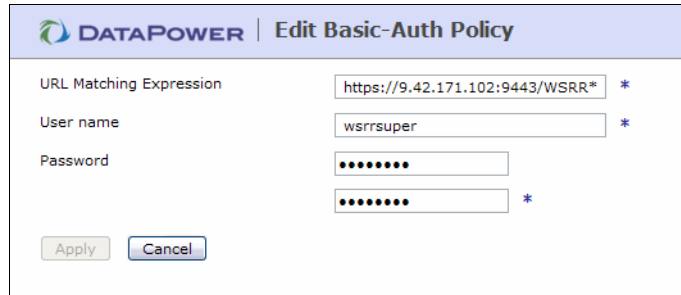
Figure 21 Creating a new user agent

- h. Click **SSL Proxy Profile Policy** and click **Add**.
- i. In the new window that appears, enter **https://9.42.171.102:9443/WSRR*** in the URL Matching Expression field and select **ForwardSSLProxy** in the SSL Proxy Profile drop-down menu.

Note: We can use the same SSL proxy profile for both the WSRR server and the production servers, if they share the same public/private key.

- j. Click **Apply** and the window will close.
- k. Click the **Basic-Auth Policy** tab.
- l. Click **Add**.

- m. In the new window that opens, enter `https://9.42.171.102:9443/WSRR*` in the URL Matching Expression field and the user name and password for communicating with WSRR, as shown in Figure 22.



The screenshot shows a web browser window titled "Edit Basic-Auth Policy" from the DataPower console. The form contains the following fields:

- URL Matching Expression:** `https://9.42.171.102:9443/WSRR*`
- User name:** `wsrrsuper`
- Password:** Two masked password fields (represented by dots).

At the bottom left, there are two buttons: "Apply" and "Cancel".

Figure 22 Configuring the Basic Auth Policy

- n. Click **Apply** and the window will close.
- o. In the Configure User Agent window, click **Apply** and the window will close.
- p. In the Configure XML Manager window, click **Apply** and the window will close.
- q. In the Configure Multi-Protocol Gateway window, click **Apply**.
5. Every time there is a request to DataPower, there will be a subsequent query to WSRR, which could have adverse performance implications. DataPower needs to be configured to cache the results of WSRR queries.
- Log in to the DataPower Web console.
 - Click **Multi-Protocol Gateway**.
 - Click **AccountCreation**.
 - In the XML Manager drop-down menu, click
 - Click **Document Cache Policy**.
 - Click **Add**.
 - Enter `https://9.42.171.102:9443/WSRR*` in the URL Match Expression field.
 - Click **Fixed** in the Policy Type drop-down menu.
 - Enter 900 in the TTL field.
 - Click **Apply** and the window will close.
 - In the Configure XML Manager window, click **Apply** and the window will close.

- I. In the Configure Multi-Protocol Gateway window, click **Apply**.

Note: This caching policy will cache the results of all queries to the WSRR REST interface.

The team that wrote this IBM Redpapers publication

This paper was produced at the International Technical Support Organization, Raleigh Center.

Andrew White is a software developer from the United Kingdom. He has 3 years of experience in middleware development at IBM. Andrew holds a degree in Computer Science from The University of Nottingham. His areas of expertise include WebSphere Application Server, WebSphere Service Registry and Repository (WSRR), WebSphere DataPower, WebSphere Process Server, Tivoli® Composite Application Manager for Service Oriented Architecture (ITCAM4SOA), Tivoli Security Policy Manager (TSPM), service-oriented Architecture (SOA), and Microsoft® .NET.

Thanks to the following people for their contributions to the paper:

- ▶ Martin Keen
- ▶ Nicole Hargrove
- ▶ Ian Heritage
- ▶ Prasad Imandi
- ▶ Wendy Neave
- ▶ Laura Olson
- ▶ Bhargav Perepa

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 give 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.

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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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.

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.

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.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

© Copyright International Business Machines Corporation 2009. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

This document REDP-4559-00 was created or updated on November 5, 2009.



Send us your comments in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:
ibm.com/redbooks
- ▶ Send your comments in an email to:
redbook@us.ibm.com
- ▶ Mail your comments to:
IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099, 2455 South Road
Poughkeepsie, NY 12601-5400 U.S.A.




Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

DataPower®
IBM®
Redpaper™

Redpapers™
Redbooks (logo) ®
Tivoli®

WebSphere®

The following terms are trademarks of other companies:

Microsoft, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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