Configuring data collection: DataPower
SOA Appliance - BETA
Configuring data collection: DataPower SOA Appliance

This section describes the support for monitoring of service flows through an IBM® WebSphere® DataPower® SOA appliance, where the ITCAM for SOA data collector acts as a proxy between the services clients and servers.

The list of versions of the DataPower SOA appliance supported by ITCAM for SOA 7.2 Fix Pack 1 is available from the Software product compatibility reports website. For information about accessing reports from this website, see Required software.

The DataPower data collector can be integrated with ITCAM for Transactions. If configured, transaction events are sent to a Transaction Collector which stores and aggregates transaction data from multiple data collectors. For more information about configuring the interface to ITCAM for Transactions, see Integrating with ITCAM for Transactions.

IBM WebSphere DataPower SOA appliances are used for processing XML messages, and providing message transformation, services acceleration, and security functions. A DataPower appliance is typically used to improve the security and performance of services by offloading functions from the application server that is hosting the target service to a DataPower SOA appliance. Typical functions that are off-loaded include; authentication and authorization, XML schema validation, and services encryption and decryption.

ITCAM for SOA provides a DataPower data collector that operates as a proxy and monitors services flows through a DataPower SOA appliance, providing similar services management and availability information that ITCAM for SOA currently provides for application server runtime environments. This information is displayed in the Tivoli® Enterprise Portal using the usual predefined or user-defined workspaces and views.

DataPower supports two proxy types that can process SOA messages:

The Web Services Proxy
You configure a Web Services Proxy by importing one or more WSDL files and then telling the appliance where to direct those messages. Thus, the Web Services Proxy receives only SOAP messages.

The Multi-Protocol Gateway
The Multi-Protocol Gateway is more versatile than the Web Services Proxy. You can use it to process nearly any type of message, including SOAP, non-SOAP XML, text, or binary. For XML messages (including SOAP), XSL transforms are used to manipulate the message. For non-XML messages, similar transform actions can be built using IBM WebSphere Transformation Extender (for more information about this product, see http://www-306.ibm.com/software/integration/wdatastagetx/).

Upgrade your firmware
Before using the DataPower data collector proxy, you must upgrade the firmware on DataPower SOA appliances that you want to monitor, to include the necessary monitoring and data transformation capabilities.
Important:

- Upgrade your firmware to at least version 3.7.1 or later to monitor traffic through the Web Services Proxy.
- Upgrade your firmware to at least version 3.7.1 Fix Pack 4 or later to monitor traffic through a Multi-Protocol Gateway.

Consult your DataPower Appliance documentation for information about upgrading your firmware level.

The DataPower data collector as a proxy

Data collectors provided with ITCAM for SOA are usually installed directly into the application server runtime environment hosting the services being monitored. The DataPower SOA appliance, however, does not support the installation of additional software, such as a data collector. Unlike other application server runtime environments, the ITCAM for SOA data collector for the DataPower environment is installed on a separate computer system and uses a special communication mechanism that allows external software applications to receive data from its internal transaction log.

This communication mechanism is used to retrieve monitoring data about services requests flowing through one or more DataPower SOA appliances, and to convert the data into a format that ITCAM for SOA can process. In this way, the DataPower data collector acts as a proxy between the DataPower SOA appliances and the ITCAM for SOA monitoring agent.

The DataPower data collector can be installed on a dedicated computer system, or it can run on a computer that is also hosting data collectors for other application server runtime environments.

Important: IBM supports only one instance of the DataPower data collector running on any computer system. This one data collector instance, however, can monitor any number of domains on any number of appliances, subject to available resources.

When data collection is enabled for the DataPower environment, the data collector subscribes to each monitored DataPower SOA appliance and then polls the appliance for monitoring data at the specified interval. The data that is retrieved from the DataPower SOA appliance is written to metric log files in the format used by ITCAM for SOA. When this data is later displayed in the Tivoli Enterprise Portal, nodes are displayed in the Tivoli Enterprise Portal Navigator view that represent the DataPower SOA appliances that are being monitored. You can select workspaces under these nodes and view the services management data for the service requests flowing through the monitored DataPower SOA appliances.

The DataPower data collector can subscribe to multiple DataPower SOA appliances, and retrieve and manage data from multiple domains. This data can then be separated by DataPower domain or aggregated across multiple domains and appliances, depending on how you configure the data collector. The DataPower data collector uses a configuration file that contains information about which DataPower SOA appliances are being monitored and information needed to establish communication with each monitored appliance.
Planning for deployment

DataPower proxies are defined within *application domains*, and DataPower users can be restricted to access some or all domains. When configuring the DataPower data collector, you must understand how the domains and users are defined on the monitored DataPower SOA appliances, to ensure that the data collector uses valid authentication credentials. This refers to user IDs and passwords that have access to the DataPower domains containing the services proxies to be monitored. In addition, you must decide how you want to aggregate or separate the data collected from those domains for display in the Tivoli Enterprise Portal.

You can use DataPower SOA appliances in several typical configurations:

**Single appliance, single domain**

The data collector monitors a single DataPower SOA appliance, with all of the monitored resources that are defined in a single domain on the appliance.

**Single appliance, multiple domains**

The data collector monitors a single DataPower SOA appliance, but that appliance has monitored resources that are defined in more than one domain on the appliance.

**Multiple appliances with different configurations**

The data collector monitors multiple DataPower SOA appliances, and each appliance has a different configuration of resources to be monitored. Each appliance is configured for a particular job, with no intention of load-balancing or fail-over between appliances.

**Multiple appliances with identical configurations**

The data collector monitors multiple DataPower SOA appliances, and all of the appliances have an identical configuration of resources being monitored. All of the appliances are configured for the same job, taking advantage of load-balancing, and fail-over capabilities between appliances.

Given these typical configurations, the DataPower data collector provides a great deal of flexibility in defining how the collected monitoring data should be separated or aggregated, across a single appliance or multiple appliances, for display in the Tivoli Enterprise Portal. The following examples illustrate how data can be separated or aggregated for managing the data from various domains and appliances:

**Separation of data at the domain**

You can view the services management data for the resources in a single domain, separate from the data for resources in other domains.

**Aggregation of data across domains**

You can view the services management data for the resources in several domains (for example, all of the domains on a DataPower SOA appliance) in an aggregated form, with no regard for the domain in which individual resources are defined.

**Separation of data at the appliance**

You can view the services management data for resources on a single DataPower SOA appliance, separate from the data for resources on other appliances.

**Aggregation of data across appliances**

You can view the services management data for the resources on several DataPower SOA appliances (for example, all of the appliances in a
load-balancing cluster) in an aggregated form, with no regard for what activity occurs on each individual appliance.

By default, the DataPower data collector aggregates data for all of the monitored domains on a single DataPower SOA appliance (even if the domains are accessed using different credentials), and keeps the data from each DataPower SOA appliance separated. See “Enabling data collection” on page 19 for more information. However, operational flow data is collected only by individual domains. Refer to “Creating node names in Tivoli Enterprise Portal” on page 25 for additional details.

A single instance of the DataPower data collector can monitor any number of DataPower SOA appliances, limited only by the memory, CPU power, and other resources available to it.

**Important:** IBM supports running only a single instance of the data collector on any computer system.

**Aggregation**

For most ITCAM for SOA data collectors, aggregation of data is performed for each application server runtime environment. The nodes in the Tivoli Enterprise Portal Navigator view represent individual application server runtime environments that have their own individual data collectors. Because the DataPower data collector can monitor multiple DataPower SOA appliances and multiple domains within each appliance, this single application server runtime environment, single data collector model no longer applies.

Using the DataPower data collector, you assign names to groups of data, referred to as display groups. Each group of data is displayed as its own node in the Tivoli Enterprise Portal. Using these named display groups, you can configure the DataPower data collector to gather information from any domains on any DataPower SOA appliances for aggregation and display.

By carefully managing the way in which you name these display groups, the DataPower data collector can separate or aggregate the data in different ways, such as isolating data specific to an individual domain on a single DataPower SOA appliance, or aggregating data across several DataPower SOA appliances into a single display group. This simple display group naming mechanism gives you great flexibility in the separation and aggregation of the data that is displayed in the Tivoli Enterprise Portal. However, operational flow views are displayed by domain, not by display group. See “Creating node names in Tivoli Enterprise Portal” on page 25 for additional details.

**Deployment steps**

To deploy the DataPower data collector in your environment, complete the following general steps:

1. Configure your DataPower SOA appliances for monitoring (see “Configuring the DataPower SOA appliance for monitoring” on page 3 for details).
2. Enable the DataPower data collector (see “Enabling data collection” on page 19 for details).
3. Run the startDC script to start the data collector (see “Starting and stopping the data collector” on page 33 for details), or configure the data collector to run in the background and start the background task.
Unconfiguration steps
To unconfigure the DataPower Data Collector in your environment, complete the following general steps:
1. Run the `stopDC` script to stop the Data Collector (see “Starting and stopping the data collector” on page 33 for details) when the DataPower proxy is started as a background task, or run the `stop`, `quit`, or `exit` command from the console to initiate an orderly shutdown of the DataPower Data Collector.
2. Disable the DataPower Data Collector (see “Disabling data collection” on page 32 for details).

Configuring the DataPower SOA appliance for monitoring
Before a DataPower SOA appliance can be monitored by the DataPower data collector, configure the DataPower SOA appliance by completing these tasks, described in more detail in the sections that follow:
- Upgrade your DataPower firmware to the minimum supported version.
- Configure a user account on the DataPower SOA appliance for use with the DataPower data collector.
- Enable the XML Management Interface on the appliance.
- Check additional optional settings for each domain to be monitored.
- Enable the ITCAM for SOA transforms for the Web Services Proxy gateways and the Multi-Protocol gateways as needed.
- Configure the AAA policy for the Web Services Proxy gateways and the Multi-Protocol gateways if you plan to monitor Web service requesters by user ID.

Upgrading the DataPower firmware version
Before using the DataPower data collector proxy, you must upgrade the firmware on DataPower SOA appliances that you want to monitor, to include the necessary monitoring and data transformation capabilities. You must upgrade the firmware to version 3.7.1 or later to use the Web Services Proxy. To use the Multi-Protocol Gateway Proxy services on your DataPower appliance, you must upgrade the firmware to version 3.7.1 Fix Pack 4 or later.

Whenever you upgrade your firmware, verify that all of your configuration settings are set correctly.

Configuring a user account on the DataPower SOA appliance
The DataPower user ID used by the data collector must belong to a user group with the following permissions:
- `Read` permission on the Login XML-Mgmt Resource Type in the default domain.
- `Read` permission on the XML-mgmt Resource Type in each domain to be monitored using this user ID.
- `Read` permission on the (any) Resource Type in each domain to be monitored using this user ID.

See your DataPower WebGUI Guide or DataPower CLI Reference Guide for details on configuring user group permissions.
Configuring the XML Management Interface

The XML Management Interface on the appliance must be enabled using the DataPower administration console. To configure the XML Management interface, complete the following steps:


2. Complete the following steps to enable the XML Management interface.
   a. Log in to the administration console as the admin user for the default domain.
   b. Navigate to Objects > Management > XML Management Interface.
   c. Make note of the port number that is displayed. You must specify this port number later when you enable or disable data collection.
   d. In the Main tab, find the WS-Management Endpoint option and select the on check box.
   e. Click Apply to activate the changes and enable the WS-Management Endpoint.

3. Complete the following steps to configure the Web Services Agent for the default domain:
   a. Navigate to Services > Miscellaneous > Web Services Agent. For example:

   **Configure Web Services Management Agent**

   ![Configure Web Services Management Agent](image)

   Figure 1. Configure Web Services Management Agent page

   b. Set Administrative State to enabled.
   c. Set the set Buffering Mode option to discard or buffer. The default setting is discard.

   When Buffering Mode is set to buffer, the Web Services Agent buffers transaction information for the current domain when no registered ITCAM for SOA data collectors are running. Buffering reduces the loss of transaction information, but consumes more memory. Transaction records
are buffered until the configured size limits are reached. Buffering is a better choice when initially configuring data collection, when processing high volumes of data, or when there are multiple ITCAM for SOA subscribers. When **Buffering Mode** is set to **discard**, transaction information from the current domain is discarded when no registered ITCAM for SOA data collectors are running. Complications can occur if a new ITCAM for SOA subscriber replaces a former subscriber. High volumes of transactions might cause the Complete Records Count to reach the configured Maximum Record Size limit and transaction information to be discarded. Setting **Buffering Mode** to **discard** is suited to an environment where there is a single ITCAM for SOA subscriber, where the DataPower appliance is handling a low volume of transaction data, and where the ITCAM for SOA subscriber is collecting a low volume of metrics. For information about troubleshooting scenarios where transaction metrics are being discarded, see the *IBM Tivoli Composite Application Manager Troubleshooting Guide*.

d. Adjust the values for **Maximum Record Size** and **Maximum Memory Usage**, if necessary.

e. If you want the data collector to record message content in addition to summary metrics, change **Capture Mode** from **faults** to **all-messages**.

4. Configure the Web Services Agent for **all other domains** that are monitored by the DataPower data collector. For each domain, switch to the domain and complete step 3a on page 6 to step 3e.

### Configuring DataPower Processing Rules

In addition to upgrading your DataPower firmware, you might have to add XSL transforms to the action rules in each processing rule for each affected Web Services Proxy or Multi-Protocol Gateway Proxy object:

- If you are monitoring a WS-Proxy without topology support or transaction tracking API (TTAPI) support, no transforms are necessary.
- If you are monitoring topology or you configured a TTAPI in a WS-Proxy, the preloaded transforms must be added to your processing rules.
- If you are monitoring a Multi-Protocol gateway, certain transforms must be created.
- If you are monitoring topology or you configured a TTAPI on a Multi-Protocol gateway, more logic must be added either to those transforms or into additional transforms.

This section describes the procedures for configuring DataPower processing rules for Web Services Proxy gateways and DataPower Multi-Protocol gateways.

#### Configuring processing rules for DataPower Web Services Proxy gateways

This section describes the procedures for configuring DataPower processing rules for Web Services Proxy gateways.

For each Web Services Proxy object, you need to add these transforms, which are included in your DataPower firmware:

- soapreq.xsl
- soaprsp.xsl
- soaperror.xsl

Generally, the procedure to add these XSL transforms to the request and response paths involves these steps:
1. Open the Web Services Proxy Object.
2. Select the Policy tab.

![Diagram of policy tree]

Figure 2. Enabling ITCAM for SOA transforms in DataPower

3. If you currently do not have a processing rule defined, expand the node tree to display subnodes (proxy, wsdl, service, and others).
4. Select an Add Rule icon and continue to configure this new rule as if it already exists.
5. For an existing or newly added rule, complete the following steps:
   a. Add a Match action to the rule (the equal sign (=) icon), as needed, to match all messages, or to only match those you intend to monitor.
   b. Add a Transform action (the three-prong swirl icon) to add the ITCAM for SOA transforms that are preloaded in the firmware.
      - For a Request rule, use the store:///soapreq.xsl transform.
      - For a Response rule, use the store:///soaprsp.xsl transform.
      - For an Error rule, use the store:///soaperror.xsl transform.

   Important: For an Error rule, your Match action must match errors, not normal responses. This is an option when you configure your Match.
   c. Add a Results action to the rule (the arrow icon) to return the transformed message.
6. Apply and save your changes.

You can provide your own XSL style sheets to implement further configuration changes. Refer to the WebSphere DataPower documentation for more information about processing rules.

To use the requester identity monitoring function that is provided with ITCAM for SOA, you must complete one of the following configuration tasks:
- Configure an AAA policy for the WSP that you intend to monitor (see “Configuring the AAA policy” on page 18).
- Customize the requester identity using an XSL stylesheet (see “Configuring processing rules for DataPower gateways to monitor by requester identity” on page 17).
Configuring processing rules for DataPower Multi-Protocol Gateways

A DataPower multi-protocol gateway service handles SOA messages in a variety of transport protocols and message formats. It can process almost any incoming message and transform the message into a completely different format, if needed, before forwarding it to a server to perform additional business processing.

For example, you might use a multi-protocol gateway to provide an HTTP front end that receives a SOAP message, transforms it into a proprietary XML format, and places it on a queue for the Message Queue transport protocol. Unlike a Web Service Proxy, a multi-protocol gateway is not restricted to processing messages that are described by Web Services Description Language documents.

Because of its flexibility and its independence from Web Services Description Language document restrictions, configuring a multi-protocol gateway is different from configuring a web service proxy. The web service proxy knows the service port name and namespace, and the operation name and namespace of a SOAP message from information in the Web Services Description Language documents.

The multi-protocol gateway, however, cannot recognize this information because it has no Web Services Description Language documents. Therefore, it cannot pass this information along to the ITCAM for SOA data collector.

Instead, you must provide this information about service port name, service port namespace, operation name, and operation namespace, by writing the necessary logic into an XSLT processing rule in your DataPower multi-protocol gateway.

To record transaction data in the web service management (WSM) agent of each domain, your application developers must add additional logic to the XSLT processing rules. The ITCAM for SOA data collector polls the WSM agent at regular intervals to collect the data from the WSM agent.

For transactions to appear in ITCAM for SOA operational flow topology views, the application developers must write additional logic into the XSLT processing rules to track relationships among related web services.

The additional logic in your processing rules must perform the following steps:

1. Identify the message by setting message attributes.
2. Calculate the correlator for each request and response message in the transaction.
3. Record data about the request and response messages in the WSM agent at the end of each transaction.

Identifying the message: The wsm-agent-append() function sends data about the transaction to the WSM agent, from where the data is polled by ITCAM for SOA.

Before you invoke the wsm-agent-append() function, you must construct a processing variable containing a dpwsm:wsa-record nodeset and provide it as an argument to this method. This nodeset contains the various attributes that describe the message and the transaction.

Most of the elements in this nodeset can be set to their default values from the processing context. However, due to the absence of Web Services Description Language, the following elements must be set explicitly in your XSL:
**ws-operation**

The fully qualified operation namespace and operation name (for example, `{www.opnamespace.com}myOperation`).

**service-port**

The fully qualified service port namespace and service port name (for example, `{www.spnamespace.com}myServicePort`).

**is-one-way**

Set to `true()` if this operation is one-way, or `false()` if this operation is two-way.

**webservice**

Specifies one of the following numerical values that identify the format of the message being processed:

- The value 8 refers to a SOAP message.
- The values 9 refers to an XML message.
- The value 10 refers to a non-XML or binary message.

You should not set explicit values for any of the following elements, because they are calculated automatically by various processing tasks:

- `ws-correlator-sfid`
- `ws-client-socode`
- `ws-dp-socode`
- `ws-server-socode`
- `ws-client-hopcount`
- `ws-server-hopcount`
- `start-time`
- `duration-ms`
- `front-latency-ms`
- `back-latency-ms`
- `request-size`
- `response-size`

For more information about out the contents of this nodeset, see the `dp:wsm-agent-append()` function in the Metadata extension functions section of the DataPower Extensions Elements and Functions Catalog for the appliance. For a sample style sheet, see “Using sample style sheets to create transform actions” on page 13.

**Tracking relationships between web services using a correlator:**

For transactions to appear in ITCAM for SOA operational flow topology views, you must add additional logic to your processing rules to track relationships among related web services.

When the ITCAM for SOA data collector monitors a Web Service Proxy, the correlator is handled by the `soapreq.xsl` and `soaprsp.xsl` style sheets that are provided with the appliance firmware. When you use ITCAM for SOA to monitor multi-protocol gateways, however, you must invoke the logic necessary to calculate and propagate the correlator.

The logic extracts a string, calls a `dp:exter-correlator()` function, and attaches the result to the message. The `exter-correlator()` function uses a propriety algorithm
to calculate and propagate the correlator in transactions. Using this correlator, each of several concurrent messages are tracked independently and accurately, without errors from the heuristic algorithm.

**Handing the correlator in request messages:** When the gateway receives a request message from the client, it processes this message, and forwards it to the server for additional processing. Figure 3 displays the logic that must be added to the request processing logic to handle the correlator.

To correlate the transition of a request message through a multi-protocol gateway, add logic to the processing rules to complete these steps:

1. **Extract the incoming correlator.**
   - If your multi-protocol gateway receives SOAP messages from a web application server that is monitored by ITCAM for SOA, then you should include logic to receive an attached correlator string, if one is available.
   - The ITCAM for SOA data collector on the sending server adds a standard SOAP header named `{http://www.ibm.com/KD4Soap}KD4SoapHeaderV2` to each message that is sent. Your XSL logic must check for the presence of this header, and if it is present, extract the text node from within this element, similar to the following example:
     ```xml
       <xsl:when test='.//kd4:KD4SoapHeaderV2'>
         <xsl:value-of select='.//kd4:KD4SoapHeaderV2'/>
       </xsl:when>
       <xsl:otherwise>
         <xsl:text>NEW_CORRELATOR</xsl:text>
       </xsl:otherwise>
     </xsl:choose>
     ```
2. **Set the following service variables to identify the message:**
   - `var://service/wsm/operation` - The fully qualified operation namespace and operation name (for example, `{www.opNamespace.com}myOperation`).
   - `var://service/wsm/service-port` - The fully qualified service port namespace and service port name (for example, `{www.spNamespace.com}myServicePort`).
   - `var://service/soap-oneway-mep` - Set to `true()` if this operation is one-way, or `false()` otherwise.
var://service/wsm/service
Specifies one of the following numerical values that identify the format of the message being processed:
- The value 8 refers to a SOAP message.
- The values 9 refers to an XML message.
- The value 10 refers to a non-XML or binary message.

When you set these variables, allow the ws-operation, service-port, webservice, and is-one-way elements of the wsa-record nodeset to take their default values.

3. Calculate the correlator.
   Invoke the dp:exter-correlator() function. The first argument is the incoming correlator string. If you do not have an incoming correlator, use the literal string NEW_CORRELATOR instead. For a request message, specify 0 for the second argument.

4. Propagate the correlator.
   The return value from the dp:exter-correlator() function is the new correlator. Attach this string to the data collectors on the back-end servers.

   If you are sending SOAP messages, add XSTL logic to construct a {http://www.ibm.com/KD4Soap}KD4SoapHeaderV2 header and add it to the SOAP Header section of the outbound message.

   For sample style sheets, see “Using sample style sheets to create transform actions” on page 13.

Handling the correlator in a response message: When the gateway receives a response message from the server, it processes that message as needed, and forwards that response back to the client. Figure 4 displays the logic that must be added to the response processing logic to handle the correlator.

   ![Request processing logic](image)

   Request processing logic

   (1) Extract the correlator from response (rsp_corr)
   (2) Calculate the correlator
      
   
   rsp2 = dp:exter_correlator(rsp_corr, 1)
   (3) Attach new correlator (rsp2)

   Figure 4. Logic added to the response processing logic to handle the correlator

The process of calculating the correlator for a response message is similar to that of a request message. When processing a response message, add the following logic necessary to calculate and propagate the correlator:
1. Extract an incoming correlator, if one is present.
2. Calculate a new correlator.
   Pass in the newly extracted correlator, but use 1 instead of 0 for the second argument.
3. Attach the newly calculated correlator.
For a sample style sheets, see "Using sample style sheets to create transform actions."

Handling the correlator in an error response: The process of calculating the correlator for an error message is similar to that of a response message.

For an error processing rule, be careful to identify whether the error is on a request or a response, setting the direction according to the var://service/response-mode variable, as shown in the following example code:

```xml
<xsl:variable name="rmode">
  <xsl:value-of select="dp:variable('var://service/response-mode')"/>
</xsl:variable>

<xsl:variable name="newCorrelator">
  <xsl:choose>
    <xsl:when test="$rmode='2'">
      <xsl:value-of select="dp:exter-correlator($InboundCorrelator,'1')"/>
    </xsl:when>
    <xsl:otherwise>
      <xsl:value-of select="dp:exter-correlator($InboundCorrelator,'0')"/>
    </xsl:otherwise>
  </xsl:choose>
</xsl:variable>

<dp:set-variable name="var://context/kd4/responseCorrelatorOut" value="$newCorrelator"/>
```

The processing-rule context automatically populates the error details, and the response is identified as a fault to ITCAM for SOA. You must include the same XSTL logic on an error rule that you include on a response rule.

For more information about using dp:exter-correlator() function and for code examples, see the Metadata extension functions section of the DataPower Extensions Elements and Functions Catalog for the appliance. For a sample style sheet, see "Using sample style sheets to create transform actions."

Recording transaction data on the appliance: The wsm-agent-append() function records the various attributes of this transaction in the WSM agent. The data is available for ITCAM for SOA to collect.

Your processing rules must call the wsm-agent-append() function as close as possible to the end of the response processing rules and error processing rules.

Remember: You must call the exter-correlator() before you call the wsm-agent-append() function.

For more information about using the wsm-agent-append() function and for code examples, see the Metadata extension functions section of the DataPower Extensions Elements and Functions Catalog for the appliance. For a sample style sheet, see "Using sample style sheets to create transform actions."

Using sample style sheets to create transform actions: A sample set of style sheets is provided in SampleStyleSheets.zip on the ITCAM for SOA Service Management Connect wiki page. The style sheets are provided to assist in creating transform actions. See the wiki page for any restrictions that might apply to using the sample style sheets.
Figure 5 shows the XSLT logic that must be added to the processing rules to correlate messages across transaction data from multi-protocol gateways and to record the transaction data in the WSM agent for collection by ITCAM for SOA.

Use the sample style sheets to create and add logic to the request, response, and error messages to perform the following actions:

1. Extract the request correlator, if present.
   Use the style sheet `ExtractRequestCorrelator.xsl` as an example.

2. Set the service variables, if you plan to create a correlator to track request and response messages.
   Use the style sheet `SetMessageAttributes.xsl` as an example.

   **Important:**
   - The `wsa-record` noderset should take the default values from the service variables.
   - You must edit the `SetMessageAttributes.xsl` file to use the service and operation names that are appropriate for your application.

3. Calculate the correlator for the request message.
   Use the style sheet `CalculateRequestCorrelator.xsl` as an example.

4. Propagate the correlator to the server.
   Use the style sheet `AttachRequestCorrelator.xsl` as an example.

5. Extract the correlator from the response message.
   Use the style sheet `ExtractResponseCorrelator.xsl` as an example.

6. Calculate the correlator for the response message.
   Use the style sheet `CalculateResponseCorrelator.xsl` as an example.

7. Attach the new correlator to the response message.
   Use the style sheet `AttachResponseCorrelator.xsl` as an example.

8. Record transaction data in the web services management agent of the domain.
   Use the style sheet `SubmitMonitoringData.xsl` as an example.

To use the requester identity monitoring function that is provided with ITCAM for SOA, you must complete one of the following configuration tasks:

- Configure an AAA policy on each gateway that you intend to monitor
- Customize the requester identity using an XSL style sheet. You specify the requester identity using the `SubmitMonitoringData.xsl` style sheet in step 8.
For more information about tracking the requester identity, see “Configuring processing rules for DataPower gateways to monitor by requester identity” on page 17.

Enabling DataPower Multi-Protocol Gateways transforms: This section describes the procedures for adding the XSL transforms that you created for multi-protocol gateways (see “Using sample style sheets to create transform actions” on page 13) to the request, response, and error processing rules.

To add the XSL transforms to the request, response, and error paths, complete the following steps:

1. Open the multi-protocol gateway object.
2. Select a policy from the Multi-Protocol Gateway Policy drop-down list and click the edit button.
3. For the request processing rule, add the style sheets that you created to perform the following actions:
   a. Extract the request correlator.
   b. Set the service variables.
   c. Calculate the correlator for the request message.
   d. Propagate the correlator to the server.
   Click the transform actions (the three-prong swirl icon) button to add each style sheet.
   For more information about the sample style sheets to use to create each transform, see “Using sample style sheets to create transform actions” on page 13.
4. For the response and the error processing rules, add the style sheets that you created to complete the following actions:
   a. Extract the correlator from the response message.
   b. Calculate the correlator for the response message.
   c. Attach the new correlator to the response message.
   d. Record the transaction data in the WSM agent of the domain for collection by ITCAM for SOA.
   For more information about the sample style sheets to use to create each transform, see “Using sample style sheets to create transform actions” on page 13.
5. Verify that you created three processing rules (request, response, and error processing rules) and that you added a match action (the equal sign (=) icon), the four transform actions (the three-prong swirl icon), and a results action (the arrow icon) to each rule. For example:
6. Apply and save your changes.

Additional considerations: Consider the following requirements, limitations, and troubleshooting tips when configuring processing rules:

Service Port Name Type:
Because of known limitations, the Service Port Name Type attribute SPNAMETYPE is always set to 1 (WSDL Port Name), regardless of the values that are used in the var://service/wsm/service variable and the webservice element. Accurate values are necessary for correct calculation of various other identifiers within the context, however, so be sure to set them accurately.

Logging return codes:
Log the return code from dp:wsm-agent-append(). If there is an error in this function, or a problem with the set of data you collected for it to log, the return value from this function describes the error. To debug problems with this function, your XSL logic must make this return value available. If your return value says error - no record, this does not reflect an error in your XSL; it indicates that there are no subscriptions active for this domain at the moment. You can activate the ITCAM for SOA data collector or set the buffer mode setting for this domain to buffer to save metric records when the DataPower data collector is not running.

XSL variables and syntax:
Pay careful attention to references to XSL variables and DataPower context variables, and to use single quotation marks, and double quotation marks, correctly as needed.

Troubleshooting:
Other troubleshooting tools and techniques, including using a multistep probe and referring to the log files on both the DataPower appliance and on the ITCAM for SOA data collector, still useful in troubleshooting issues with monitoring the multi-protocol gateway service.
Configuring processing rules for DataPower gateways to monitor by requester identity

Before you use the requester identity monitoring function that is provided with ITCAM for SOA, you must configure the Web Services Proxy Gateway or the Multi-Protocol Gateway to track the requester identity using one of the following options:

- Configure an AAA policy on each gateway that you intend to monitor (see “Configuring the AAA policy” on page 18).
- Create an XSL style sheet to define a value for ws-client-id and add it after each response and error processing rule that you want to monitor (see “Customizing the requester identity”).

ITCAM for SOA can only return client ID requester identity information for the message when the gateway is configured to track requester identities.

The feature to perform service monitoring by requester identity in ITCAM for SOA is disabled by default. To enable service monitoring by requester identity, use the available Tivoli Enterprise Portal Take Action commands AddRequesterIdentity_610 and EnableReqIDMntr_610, and others, as described in the IBM Tivoli Composite Application Manager for SOA User's Guide.

Important: Fault monitoring by requester identity is always enabled for the DataPower data collector. For information about configuring the requester identity type for fault monitoring, see “Configuring the requester identity type for fault monitoring” on page 38.

Customizing the requester identity: You can define a value for Requester ID by customizing the DataPower variable ws-client-id using an XSL style sheet.

Sample style sheets are available from the ITCAM for SOA Service Management Connect wiki page.

To customize the requester identity, use the SubmitMonitoringData.xls style sheet for a Multi-Protocol Gateway and the CustomiseMonitoringData.xls style sheet for a Web Services Proxy Gateway.

In the style sheet, the following lines of code are used to determine the requester identity:

```
<dpwsm:ws-client-id default="yes" /> - <!-- Default uses internal calculation -->
<dpwsm:ws-clientid-extmthd default="yes" /> - <!-- Default uses internal calculation -->
```

Replace these lines in the style sheet with lines similar to the following lines:

```
<xsl:variable name="customName" select="dp:encode('ws-client-id','base-64')"/>
<dpwsm:ws-client-id><xsl:value-of select="$customName"/></dpwsm:ws-client-id>
<dpwsm:ws-clientid-extmthd>custom</dpwsm:ws-clientid-extmthd>
```

The first argument of dp:encode() is the value that you want to supply for the requester identity, for example, `dp:encode('my_requester_identity','base-64')`.

For both Multi-Protocol Gateways and Web Services Proxy Gateway, apply the style sheet to the response and error processing rules.
Configuring the AAA policy

If you plan to use the requester identity monitoring function that is provided with ITCAM for SOA to monitor by user identity, you can configure an AAA policy for the Web Services Proxy Gateway and Multi-Protocol Gateway on each processing rule that you intend to monitor.

If you currently have an AAA policy in place, no changes to that policy are required.

If no AAA policy exists on the processing rule, you can open a DataPower administrative console to add and configure the AAA policy. Refer to the procedures documented in Chapter 30 of the IBM WebSphere DataPower XML Security Gateway XS40 WebGUI Guide, version 3.7.3, for more information.

Important: Configure the processing rules to verify the AAA policy at the start of the processing rule before you extract the incoming correlator.

Extraction methods and priorities: Like most SOA environments, there are many ways to configure an AAA policy, and many different types of credentials are available. ITCAM for SOA selects the first available identity type from the prioritized list in Table 1.

Table 1. Extraction methods and priorities used by the DataPower DC to obtain the user identity.

<table>
<thead>
<tr>
<th>Client ID extraction method</th>
<th>Client ID type</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>http-basic-auth</td>
<td>User name</td>
<td>1</td>
</tr>
<tr>
<td>LTPA-auth</td>
<td>User ID</td>
<td>2</td>
</tr>
<tr>
<td>Saml-atr-name</td>
<td>User name</td>
<td>3</td>
</tr>
<tr>
<td>Saml-authen-name</td>
<td>User name</td>
<td>4</td>
</tr>
<tr>
<td>Wssec-username</td>
<td>User name</td>
<td>5</td>
</tr>
<tr>
<td>Custom</td>
<td>Result of XSL process</td>
<td>6</td>
</tr>
<tr>
<td>Client-IP-address</td>
<td>IP address</td>
<td>7</td>
</tr>
<tr>
<td>Client-ssl</td>
<td>Dn</td>
<td>8</td>
</tr>
<tr>
<td>Signer-dn</td>
<td>Dn</td>
<td>9</td>
</tr>
<tr>
<td>Wssec-binary-token</td>
<td>Token</td>
<td>10</td>
</tr>
<tr>
<td>Ws-secure-conversation</td>
<td>Context</td>
<td>11</td>
</tr>
<tr>
<td>Kerberos</td>
<td>Token</td>
<td>12</td>
</tr>
<tr>
<td>LTPA-token</td>
<td>Token</td>
<td>13</td>
</tr>
<tr>
<td>Ws-trust</td>
<td>Token</td>
<td>14</td>
</tr>
<tr>
<td>Cookie-token</td>
<td>Token</td>
<td>15</td>
</tr>
<tr>
<td>Xpath</td>
<td>Token</td>
<td>16</td>
</tr>
<tr>
<td>Saml-artifact</td>
<td>Artifact</td>
<td>17</td>
</tr>
</tbody>
</table>

All of the supported extraction methods are listed in the first column. The extraction methods are prioritized by the following rules:

- Sort the extraction methods by the precision of the corresponding user identity, from most specific to least specific:
  - Username → IP address → Dn → Token
Within the extraction methods that extract the user name and user ID, system-level authentication takes priority over message-level authentication:

(http-basic-auth, LTPA-auth, Saml-attr-name, Saml-authen-name) → (Wssec-username)

These priorities are only considered when multiple extraction methods are configured in the AAA policy. You cannot adjust these priorities.

Enabling data collection

This section describes how to configure the DataPower environment for data collection.

The DataPower configuration file

For the DataPower data collector, the Data Collector Configuration utility and the KD4configDC command manipulate the contents of a special DataPower configuration file by adding sections to the file when new DataPower monitoring is enabled, and removing sections from the file when monitoring is disabled. Each invocation of the Data Collector Configuration utility or the KD4configDC command adds, updates, or removes one section of the DataPower configuration file. Each section of the DataPower configuration file might be associated with its own data group, or it might be part of a larger data group to which other sections of the configuration file also belong.

The DataPower data collector uses this configuration file to identify the DataPower SOA appliances that are to be monitored and to specify all of the information that is needed to communicate with those appliances. Typical information that is stored for each connection includes host name and port, user ID and password, domains to monitor, and polling interval.

The configuration file is in the ITCAM4SOA_Home/KD4/config directory on Linux or UNIX systems and the ITCAM4SOA_Home/KD4/config directory on Windows systems and is called KD4.dpdcConfig.properties. This file is maintained separately from the existing KD4.dc.properties configuration file. This is a sample DataPower configuration file:

```
# Sample DataPower data collector configuration file
DataPower.count=3
#
DataPower.host.1=dpbox1
DataPower.port.1=5550
DataPower.path.1=/
DataPower.poll.1=60
DataPower.user.1=admin
DataPower.encpswd.1=#$%*&
DataPower.domainlist.1=default,testdom1
DataPower.displaygroup.1=dpbox1
DataPower.subExpire.1=15
DataPower.maxrecords.1=1000
#

DataPower.host.2=dpbox2
DataPower.port.2=5550
DataPower.path.2=/
DataPower.poll.2=30
DataPower.user.2=user1
DataPower.encpswd.2=&*%$#
DataPower.domainlist.2=userdom1,userdom2,userdom3
DataPower.displaygroup.2=user_doms
DataPower.subExpire.2=15
```

Configuring data collection: DataPower SOA Appliance 19
In the example, there are three sections in the configuration file. The properties in each of the three sections provide all of the information that is needed to establish and manage a single connection or session with each DataPower SOA appliance.

Change the information in this configuration file using either the Data Collector Configuration utility or the KD4configDC command. You can only modify the parameters that are set when you first run the Data Collector Configuration utility or when you first issued the KD4configDC command. To set additional parameters, you must manually add them to the configuration file.

Using various combinations of parameters in the Data Collector Configuration utility input pages or in the KD4configDC command, you can achieve different monitoring configurations to separate or aggregate data among domains and appliances. See "Considerations for enabling data collection for DataPower monitoring" on page 28 for more information.

Before you configure your DataPower environment for data collection, consult with your local systems management planners to understand which domains on which DataPower SOA appliances are to be monitored and how the data from these domains and appliances should be separated or aggregated for display in the Tivoli Enterprise Portal.

To set the DataPower.maxrecords property to an optimal value, it is useful to determine the number of transactions that are processed by each of the configured domains. The DataPower.maxrecords property must be set in line with the expected traffic levels of each configured domain. For more information about setting the transaction rate, see "Optimizing performance" on page 40.

Restriction: In an upgrade scenario, to set the maximum number of records for an existing display group, you must add the DataPower.maxrecords parameter manually to the section in the KD4.dpdcConfig.properties file that configures the display group.

Additional properties

Beginning with ITCAM for SOA version 7.2 Fix Pack 1, you can optionally add the following properties manually to the KD4.dpdcConfig.properties file:

**DataPower.dimension.domain**

Adds the DataPower domain attribute to the Service Inventory_610, the Service Inventory Requester Identity_610, and the Fault Log_610 tables. The domain attribute is added to the Application Server Cluster Name column in each table.

Because the Application Server Cluster Name is a dimensional field within the ITCAM for SOA Reports schema, when you run reports on the Tivoli Data Warehouse, you can display aggregated metric data by domains.
If the `DataPower.domainlist` parameter contains more than one domain, the `DataPower.dimension.domain` parameter is treated as being set to false.

**DataPower.alias**

Assigns an alias to the DataPower host. If this property is set, the alias is added to the Application Server Cell Name column in the Service Inventory_610, the Service Inventory Requester Identity_610, and the Fault Log_610 tables. Alternatively, you can use the `DataPower.displaygroup` property to create an alias for the DataPower host.

**DataPower.multihost.group**

The `DataPower.displaygroup` property defines the name to display for the DataPower appliance node in the Navigator view in the Tivoli Enterprise Portal. When the property is set, data from all DataPower appliances that are associated with the display group is aggregated for the display group. The data is available from the single node in the Navigator view.

The display group might be associated with different domains on the single DataPower appliance. In the Service Inventory_610, the Service Inventory Requester Identity_610, and the Fault Log_610 tables, the Application Server Node Name attribute displays the name of this appliance. Alternatively, the display group might be associated with multiple appliances. The Application Server Node Name attribute displays the name of only one of the appliances.

By setting the `DataPower.multihost.group` property to `true`, you indicate that there are multiple appliances associated with a display group. The Application Server Node Name attribute is not populated. By setting the property to `false`, you indicate that the display group is associated with a single DataPower appliance. The name of the DataPower appliance is added to the Application Server Node Name column. If the `DataPower.multihost.group` property is not specified in the configuration file, the property is set to `false`.

---

**Enabling data collection using the Data Collector Configuration utility**

To enable data collection using the Data Collector Configuration utility, complete the following steps:

1. **Run the Data Collector Configuration utility (see [Running the Data Collector Configuration utility](#)):**
   a. Select the DataPower SOA Appliance runtime environment.
   b. Select the **Configure > DataPower Instance** option.
c. Select the option to enable data collection.

For the remaining input parameters, refer to Table 2 on page 24 for a description of the information that is required, and see the additional information and examples of running the **KD4configDC** command to learn more about how to specify these parameters for your environment.

d. Specify the DataPower Host Name.

e. Specify the DataPower user ID.

f. Specify the DataPower password.

g. The default DataPower port number (5500) and the default polling interval (10 seconds) are provided. Accept these defaults or specify different values.

h. Verify that the DataPower Path is set to /.
Restriction: You cannot specify a path other than /. Changing its value has no effect.

i. Specify the DataPower Domain List.

Figure 9. Enabling data collection in a DataPower runtime environment

k. Wait for the configuration utility to complete the operation.
l. Exit the utility.

Important: When you deploy the DataPower data collector, you must start the data collector once you have enabled it to begin collecting data. For more information about starting the data collector, see "Starting and stopping the data collector" on page 33.

Enabling data collection using the KD4configDC command

The syntax for running the KD4configDC command for the DataPower environment is similar to the syntax for other supported ITCAM for SOA data collector environments. To run the KD4configDC command, first navigate to the following location, depending on your operating system:

Enter the KD4configDC command:

- For supported Windows operating systems:
  KD4configDC { -enable | -disable } -env 8 <env specific parameters>

- For supported AIX®, Solaris, HP-UX, and Linux operating systems:
  KD4configDC.sh { -enable | -disable } -env 8 <env specific parameters>

Specifying KD4configDC parameters

The env specific parameters defined for the DataPower invocation of the KD4configDC command are a series of key and value pairs that define the necessary properties for the affected section of the DataPower configuration file. These key and value pairs, which you can specify in any order on the command line, are shown in Table 2 on page 24.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Optional / Required</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>–host host name or IP address</td>
<td>Required</td>
<td></td>
<td>Defines the DataPower SOA appliance host name or IP address. This host name is used to establish a socket connection and is used as part of the Web address pointing to the DataPower SOA appliance. This value can be any length string, with no blank characters. See &quot;Creating node names in Tivoli Enterprise Portal&quot; on page 2 regarding possible truncation of this value in the node name.</td>
</tr>
<tr>
<td>–user user ID</td>
<td>Required</td>
<td></td>
<td>Defines the DataPower SOA appliance authentication user. This user must be a valid user for the DataPower SOA appliance defined by the –host parameter. See your DataPower documentation for information about creating and managing user IDs for DataPower SOA appliances. See &quot;Configuring a user account on the DataPower SOA appliance&quot; on page 5 for more information.</td>
</tr>
<tr>
<td>–pswd password</td>
<td>Optional</td>
<td>User is prompted, if necessary</td>
<td>Defines the DataPower SOA appliance authentication password, entered in clear text (not encoded). This password must be valid for the user defined in the –user parameter, and must be valid for the DataPower SOA appliance defined by the –host parameter. This password is automatically converted to an encoded (masked) form and is stored in the DataPower configuration file. See your DataPower documentation for information about creating and managing passwords for DataPower SOA appliances.</td>
</tr>
<tr>
<td>–port port number</td>
<td>Optional</td>
<td>5550</td>
<td>Defines the DataPower SOA appliance port number. This port number is used to establish a socket connection and is used as part of the Web address pointing to the DataPower SOA appliance. This value must be an integer from 0 to 65535. If this parameter is not specified, the default value is used.</td>
</tr>
<tr>
<td>–path path string</td>
<td>Required</td>
<td>/</td>
<td>Defines the DataPower SOA appliance path. This path is used as part of the Web address pointing to the DataPower SOA appliance. <strong>Restriction:</strong> You cannot specify a path other than / . Changing its value has no effect.</td>
</tr>
<tr>
<td>–poll polling interval</td>
<td>Optional</td>
<td>10 seconds</td>
<td>Defines the DataPower SOA appliance polling interval (in seconds). The data collector waits this amount of time between each poll of the DataPower SOA appliance. This must be an integer value, specified in seconds, between 1 and 300 (1 second to 5 minutes).</td>
</tr>
</tbody>
</table>
### Table 2. DataPower key and value pairs for the KD4configDC command (continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Optional / Required</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-maxrecords</td>
<td>Optional</td>
<td>15000</td>
<td>Defines the maximum number of records that the DataPower data collector can process from the DataPower SOA appliance per polling interval. This must be an integer value, between 1 and 30000.</td>
</tr>
<tr>
<td>-subexpire length of time the subscription is valid</td>
<td>Optional</td>
<td>15</td>
<td>Defines the length of time, in minutes, that the subscription of the DataPower data collector to the DataPower appliance remains valid. At the end of the subscription period, the DataPower data collector renewes its subscription to the DataPower appliance. This must be an integer value, specified in minutes, between 3 and 30.</td>
</tr>
<tr>
<td>-domainlist domainA, domainB, ...domainZ</td>
<td>Optional</td>
<td>No domainlist property is generated</td>
<td>Defines the DataPower SOA appliance domain list. This is a comma-separated list of domains to be monitored on the associated DataPower SOA appliance. Any domains in this list that are not authorized to the user defined by the -user parameter are not monitored. Each domain can be any string, with no blank characters. If you specify more than one domain name, separated by commas, the entire domain list must be enclosed in double quotation marks (for example, –domainlist &quot;domain1,domain2,domain3&quot;. See &quot;Considerations for enabling data collection for DataPower monitoring&quot; on page 28 for more information about using this domain list.</td>
</tr>
<tr>
<td>-displaygroup display group</td>
<td>Optional</td>
<td>No displaygroup property is generated</td>
<td>Defines the DataPower SOA appliance display name. The name can be any string, with no blank characters, up to 64 characters long. See &quot;Creating node names in Tivoli Enterprise Portal&quot; regarding possible truncation of this value in the node name. See &quot;Considerations for enabling data collection for DataPower monitoring&quot; on page 28 for more information about the use of this property.</td>
</tr>
</tbody>
</table>

**Important:** When you deploy the DataPower data collector, you must start the data collector once you have enabled it to begin collecting data. For more information about starting the data collector, see “Starting and stopping the data collector” on page 33.

**Creating node names in Tivoli Enterprise Portal**

The DataPower data collector host name is combined with the value of the -displaygroup parameter, separated by a hyphen, to form the node name that is seen in the Tivoli Enterprise Portal Navigator view.

For example, the KD4configDC command is run with the -displaygroup parameter specified with a value of dispName:
KD4configDC -enable -env 8 –host appHost –user user1 –displaygroup dispName

In this example, appHost is the host name of the DataPower SOA appliance. The DataPower data collector acts as a proxy on a different computer, and, in this example, proxy host name is dpdcHost. With the –displaygroup parameter specified in the KD4configDC command, the node name that is displayed in the Tivoli Enterprise Portal Navigator view is D4:dpdcHost-dispName.

If the –displaygroup parameter is not specified in the KD4configDC command, then the value of the –host parameter is used instead of –displaygroup to define the node name.

In the following example, the KD4configDC command is run without specifying the –displaygroup parameter:
KD4configDC -enable -env 8 –host appHost –user user1

The value of the –host parameter (appHost) is combined with the DataPower data collector host name dpdcHost to form the node name that is displayed in the Tivoli Enterprise Portal Navigator view. In this case the node name that is displayed in the Tivoli Enterprise Portal Navigator view is D4:dpdcHost-appHost.

Restriction: If the length of the node name exceeds 20 characters (including the hyphen), the node name is truncated from the end, meaning that the –displaygroup or –host parameter portion is truncated. You should be aware of this limitation of the Tivoli Enterprise Portal and choose values for the –displaygroup and –host parameters of appropriate length, to avoid truncation and possible confusion between similar names.

Examples of configuring for data collection
This section provides several examples of the kinds of monitoring that you can configure by running the KD4configDC command using various combinations of parameters.

The simplest form of the command: When you run the KD4configDC command to enable monitoring of a DataPower SOA appliance, this is the simplest format of the command:
KD4configDC -enable -env 8 –host host –user user

For example, with a host name of dpbox1 and a user name of admin, this is the following command:
KD4configDC -enable -env 8 –host dpbox1 –user admin

The DataPower configuration file is searched for an existing section that matches the specified host and user, and that has no domain list and no display group properties defined.
• If a matching section is found, you are prompted for a password, and the matching section is updated only by encoding the password and storing it in the configuration file for that section.
• If a matching section is not found in the configuration file, you are prompted for a password. A new section of the configuration file is created, with the specified values for the –host, –user and –pswd parameters that are stored in the configuration file for that section.

No values for –domainlist or –displaygroup parameters are defined, and other optional properties for the section are set to their default values.
Because no value for the –displaygroup parameter is specified, the host name dpbox1 is used as part of the node name that is displayed in the Tivoli Enterprise Portal (for example, D4:dpdcHost-dpbox1).

Specifying domain list or display group parameters: If the –domainlist or –displaygroup parameters are specified in the KD4configDC command, they become part of the matching criteria that determines whether the section exists or must be created as new. If a new section must be created, the values that are provided for the –domainlist and –displaygroup parameters are stored in the new section of the configuration file.

For example, if you have two domains, domainA and domainB that you are monitoring on a DataPower SOA appliance defined by a host named host1 and authorized to user userABC, run the KD4configDC command:

KD4configDC –enable –env 8 –host host1 –user userABC –domainlist "domainA, domainB"

The list of domains is surrounded by quotation marks, and there are no blank characters in the domain list.

As another example, suppose you want to create a node name to be displayed in the Tivoli Enterprise Portal Navigator view that contains the display group name of all_domains. This node name would be in the form of D4:dpdcHost-all_domains. Specify these parameters to run the KD4configDC command, with authorized user userABC:

KD4configDC –enable –env 8 –host host1 –user userABC –displaygroup all_domains

Changing existing values for the domain list or display group: To update an existing section of the configuration file by changing the value of either the –domainlist or –displaygroup parameter, you must first disable that section of the configuration file by running the KD4configDC command using the –disable parameter (see “Disabling data collection” on page 32), specifying the current values for the –domainlist or –displaygroup parameters.

After the section is disabled, run the KD4configDC command again, using the –enable parameter and specifying the desired values for the –domainlist and –displaygroup parameters.

Specifying the password parameter: The values specified by the –user and –pswd parameters are used to perform an HTTP basic authentication over the HTTPS session. The user and password combination must be a valid user ID and password defined using the administration interface of the DataPower SOA appliance. See the DataPower documentation for more information about creating authorized user names and passwords.

If the –pswd parameter is specified in the KD4configDC command, its value is immediately encoded and saved in the configuration file, and you are not prompted for the password. Any password that had previously been specified for an existing section (that matches the search according to the other parameters) is replaced by the new password value.

For example, if an existing section for host name H100 specified an authorized user of u2233, there would already be an encoded value of the valid password stored in the configuration file. To change the password value to x1y2z3, run the KD4configDC command:

KD4configDC –enable –env 8 –host H100 –user u2233 –pswd x1y2z3
Specifying optional parameters: If any of the remaining optional parameters, (
port or –poll) are specified as part of running the KD4configDC command, their values are stored in the configuration file in either the existing or newly created section. These values are not used as part of the matching criteria for locating existing sections of the configuration file.

The data collector uses the values in the –host and –port parameters to construct the URL in the form of https://host:port/. The URL is used as the target of the messages that are sent to the DataPower SOA appliance. Please confirm this update to correct.

Considerations for enabling data collection for DataPower monitoring

You can use variations of the parameters specified in the KD4configDC command to define properties that configure the monitoring of one or more DataPower SOA appliances. The following sections illustrate several possible configurations.

Multiple connections to the same DataPower SOA appliance: The DataPower configuration file supports multiple connections to the same DataPower SOA appliance by defining multiple sections of the configuration file with the same host and port properties. There are several reasons why you might want to do this, but the most obvious reason is to allow the data collector to use more than one set of authentication credentials.

For example, a DataPower SOA appliance might have three domains (dom1, dom2 and dom3) and two users (userA, with access only to dom1 and dom2, and userB with access only to dom3).

In this situation, neither userA nor userB can be used to access all of the domains on the appliance. To monitor all of the domains on the appliance, you must define two sections in the DataPower configuration file, one that specifies userA in one section and userB in the other section.

To define the two sections in the configuration file, run KD4configDC command twice, once for each user:

KD4configDC –enable –env 8 –host dpHost1 –user userA –domainlist "dom1,dom2"
KD4configDC –enable –env 8 –host dpHost1 –user userB –domainlist dom3

Restricting the list of domains being monitored: You can use the –domainlist parameter to restrict the list of domains being monitored for a user. For example, if userA has access to dom1 and dom2, but you are interested in monitoring only dom1, the –domainlist parameter can specify only dom1 to instruct the data collector not to monitor domain dom2:

KD4configDC –enable –env 8 –host dpHost1 –user userA –domainlist dom1

For each section in the DataPower configuration file, the domains that are monitored depend on the domains to which the user in that section has authorization, and the list of domains specified in the –domainlist parameter for that section. If this property is not specified, then all domains that are authorized to the user for that section are monitored.

Separating and aggregating data with the –displaygroup parameter: The –displaygroup parameter gives you substantial control over how the information from DataPower SOA appliances is separated or aggregated in the Tivoli Enterprise Portal views. Following are some examples of the ways that the –displaygroup parameter can be used to separate or aggregate data.
Separating data by host name or display group: Suppose that you have two DataPower SOA appliances (with different users and domains defined on each) and you want to view the information from these two appliances separately under two different nodes in the Tivoli Enterprise Portal Navigator view.

Because there are two DataPower SOA appliances, there must be at least two sections in the DataPower configuration file, one for each appliance hostname or IP address. Separation of the information from these two appliances can be done without using the –displaygroup parameter if the two sections of the configuration file have different values for the –host parameter. For example:

KD4configDC –enable –env 8 –host dpHost1 –user userA
KD4configDC –enable –env 8 –host dpHost2 –user userB

The node name that is displayed in the Tivoli Enterprise Portal is based on the value of the –host parameter if the –displaygroup parameter is not specified. As long as the two sections of the configuration file have different values for the –host parameter, ITCAM for SOA separates the data between the two appliances and displays it under two different nodes (for example, D4:dpdcHost-dpHost1 and D4:dpdcHost-dpHost2) in the Tivoli Enterprise Portal Navigator view.

Alternatively, you could use the –displaygroup parameter to specify a more meaningful string than the value of the –host parameter. As long as the values for –displaygroup for the two sections are different, the data is separated under different Tivoli Enterprise Portal Navigator nodes. For example:

KD4configDC –enable –env 8 –host dpHost1 –user userA –displaygroup Dom1
KD4configDC –enable –env 8 –host dpHost2 –user userB –displaygroup Dom2

Because the –displaygroup parameter is specified, the node name is created from those values and should be displayed in the Tivoli Enterprise Portal Navigator view as D4:dpdcHost-Dom1 and D4:dpdcHost-Dom2. However, note also that in this case, the length of the node name exceeds 20 characters, so each node name is truncated, as D4:dpdcHost-Dom1 and D4:dpdcHost-Dom2.

Separating data between domains by domain list: Suppose that you have a DataPower SOA appliance with a single user, userA, and two domains, dom1 and dom2.

You can configure the data collector to separate the information from these two domains under two different nodes in the Tivoli Enterprise Portal Navigator view by creating two sections in the DataPower configuration file. The first section specifies a –domainlist parameter of dom1 and a –displaygroup parameter of, for example, Dom1, and the second section specifies a –domainlist parameter of dom2 and a –displaygroup parameter of, for example, Dom2.

Example:

KD4configDC –enable –env 8 –host dpHost1 –user userA –domainlist dom1
–displaygroup Dom1
KD4configDC –enable –env 8 –host dpHost1 –user userA –domainlist dom2
–displaygroup Dom2

The different values for the –displaygroup parameter cause ITCAM for SOA to create two display groups that map to two different Tivoli Enterprise Portal Navigator view nodes: D4:dpdcHost-Dom1 and D4:dpdcHost-Dom2, and the –domainlist parameters restrict the data collected in each of those display groups to just the intended domain, even though userA has access to both domains.
Aggregating data across multiple domains: Suppose that you have a DataPower SOA appliance with a single user, userA, and two domains, dom1 and dom2. If the –displaygroup parameter is not specified in the section of the configuration file, the node name is created from the –host parameter, and the information for both dom1 and dom2 is aggregated under a single node (D4:dpdcHost-dpHost) in the Tivoli Enterprise Portal Navigator view.

Aggregating data across multiple domains under multiple nodes: To aggregate the data from domains domA1 and domB1 under one node name and the data from domA2 and domB2 under a different node name, configure four sections in the DataPower configuration file:

- user userA, domainlist domA1, displaygroup Group1
- user userB, domainlist domB1, displaygroup Group1
- user userA, domainlist domA2, displaygroup Group2
- user userB, domainlist domB2, displaygroup Group2

Example:
KD4configDC –enable –env 8 –host dpHost –user userA –domainlist "domA1" –displaygroup Group1
KD4configDC –enable –env 8 –host dpHost –user userB –domainlist "domB1" –displaygroup Group1
KD4configDC –enable –env 8 –host dpHost –user userA –domainlist "domA2" –displaygroup Group2
KD4configDC –enable –env 8 –host dpHost –user userB –domainlist "domB2" –displaygroup Group2

This causes ITCAM for SOA to aggregate the information from domA1 and domB1 under the Tivoli Enterprise Portal Navigator node D4:dpdcHost-Group1 and to aggregate the information from domA2 and domB2 under the Tivoli Enterprise Portal Navigator node D4:dpdcHost-Group2.

Aggregating data across all domains on a single host with multiple users: Suppose that you have a DataPower SOA appliance with two users (userA and userB) that each have access to different sets of domains (userA has access to domains named domA1 and domA2. The other user, userB, has access to domains named domB1 and domB2, respectively). To monitor all of these domains, the DataPower configuration file must have at least two sections, one with a –user parameter value of userA and one with a –user parameter value of userB, but each specifying the same value for the host name in the –host parameter.

Example:
KD4configDC –enable –env 8 –host dpHost –user userA
KD4configDC –enable –env 8 –host dpHost –user userB

If the –displaygroup parameter is not specified, the information from all four domains would be aggregated under the same Tivoli Enterprise Portal Navigator node, for example, D4:dpdcHost-dpHost.

Aggregating data for a cluster of appliances: You might have a collection of DataPower SOA appliances that are identically configured because they are used in a load-balancing or fail over situation. Because each appliance has its own host name or IP address, there must be at least one section in the configuration file for each appliance to inform the data collector how to communicate with each appliance.
If the –displaygroup parameter is not used, then the information for each of these appliances is separated into different nodes in the Tivoli Enterprise Portal Navigator view (for example, D4:dpdcHost-dpHost1), because the –host parameter is used by default.

To see how this cluster of appliances is performing as a whole, without regard to the performance of individual appliances, specify the same –displaygroup parameter (for example, Clust1) for every appliance. This causes the data collector to aggregate the data from all of these devices into a single display group, which is then displayed under a single Tivoli Enterprise Portal Navigator node (for example, D4:dpdcHost-Clust1).

Example:

```
KD4configDC –enable –env 8 –host dpHost1 –user userA –displaygroup Clust1
KD4configDC –enable –env 8 –host dpHost2 –user userA –displaygroup Clust1
```

When you specify the –displaygroup parameter for appliances in a cluster, the Application Server Node Name attribute in the Service Inventory_610, the Service Inventory Requester Identity_610, and the Fault Log_610 tables display the name of one of the appliances in the cluster. To set the attribute to empty, you can set the DataPower.multihost.group property in the DataPower configuration file to true. The DataPower.multihost.group property is introduced in ITCAM for SOA version 7.2 Fix Pack 1. For more information, see “The DataPower configuration file” on page 19.

Additional considerations: Take the following into account when considering whether to separate or aggregate data:

Combining configurations

You can mix and match any of the various configurations described in this section, which gives you a great deal of flexibility in defining how to aggregate or separate data in the Tivoli Enterprise Portal views. This flexibility might result in some DataPower domains being specified in more than one section. This is not considered an error.

Duplicated domains

You might want one section defined to allow data for a domain to be displayed on its own under one Tivoli Enterprise Portal Navigator node, and another section (or sections) defined to show the data for that domain aggregated with other domains, on the same or on a different machine. This is possible, but be aware that this results in the data for duplicated domains being retrieved multiple times by the data collector (once for each section in which it is configured). In this situation performance might be affected.

Aggregation defaults

By default, if no –domainlist parameter and no –displaygroup parameter are used at all in the DataPower configuration file, the DataPower data collector aggregates data for all of the monitored domains on a single DataPower SOA appliance (even if the domains are accessed using different credentials), and keeps the data from each DataPower SOA appliance separated.
Adding appliances to, or removing appliances from display groups

As you define your display groups, you should decide if the display group will always monitor only one appliance or multiple appliances, and once that display group is put into production, do not modify it by adding a second appliance, or by removing appliances to leave only one appliance monitored in the display group.

If your goal in this display group is to monitor a single domain on a single appliance, then after it is put into production, do not add a second appliance to that display group. If your goal is to monitor one domain across a set of appliances, then add at least the second appliance while you are still in development and test mode. After you put this display group into production, ensure that there are always at least two appliances monitored. You can increase from a cluster of two appliances to three or more as needed.

However, if you remove appliances from the display group so that there is only one remaining appliance, the node name attribute changes, and a new subnode identifier is displayed in the Tivoli Enterprise Portal Navigator Physical view. If you modify a display group that contains a single appliance by adding a second appliance, its subnode identifier will also change, and a new subnode with the same name is displayed on the Tivoli Enterprise Portal Navigator Physical view. You must then delete the offline subnode to remove the duplicate names in the view. Any historical data that you collected with the original subnode name is no longer associated with the new subnode name.

Disabling data collection

This section describes the procedures for disabling data collection in the DataPower environment.

To unconfigure the Data Power® data collector, you must stop the data collector before disabling it. For more information about stopping the data collector, see “Starting and stopping the data collector” on page 33.

Important: You do not have to stop the data collector first if you are making configuration changes.

Upgrading the data collector

If you are upgrading the data collector for the DataPower environment, do not use the –disable option of the KD4configDC script or the Data Collector Configuration utility described in this section. Instead, you should complete the following steps:

1. Stop the DataPower data collector using the stopDC command (see “Starting and stopping the data collector” on page 33 for more information.

2. Deregister the service or daemon, if applicable. See “Running the DataPower data collector as a Windows service or UNIX daemon” on page 34 for the procedures to stop a registered DataPower data collector and to deregister the service to remove it from the list of Windows services, if applicable.

See Upgrading to ITCAM for SOA version 7.2 for more information about upgrading your monitoring environment from a previous version.
Disabling data collection using the Data Collector Configuration utility

To disable data collection with the Data Collector Configuration utility, complete the following steps:

1. Run the Data Collector Configuration utility (see Running the Data Collector Configuration utility):
   a. Select the DataPower SOA Appliance runtime environment.
   b. Select the Configure DataPower Instance option.
   c. Select the option to disable data collection.
      For the remaining input parameters, refer to Table 2 on page 24 for a description of the information that is required, and see the additional information and examples of running the KD4configDC command to learn more about how to specify these parameters for your environment.
   d. Specify the DataPower Host Name.
   e. Specify the DataPower user ID.
   f. Specify the DataPower password.
   g. The default DataPower port number (5500) and the default polling interval (10 seconds) is provided. Accept these defaults or type over them to specify different values.
   h. Specify the DataPower Domain List.
   i. Optionally specify the DataPower Display Group.
   j. Wait for the configuration utility to complete the operation.
   k. Exit the utility.

Disabling data collection using the KD4configDC command

When you run the KD4configDC command to disable monitoring of a DataPower SOA appliance, use the simplest syntax for running the KD4configDC command:

KD4configDC –disable –env 8 –host host –user user

The DataPower configuration file is searched for an existing section that matches the specified host and user and that has no –domainlist parameter and no –displaygroup parameters defined. If a matching section is found, the section is removed from the file. If no matching section is found, an error message is issued and the file remains unchanged.

If the –domainlist or –displaygroup parameters are specified in the KD4configDC command, they become part of the matching criteria that determines whether the section exists and whether it is removed if found.

Restriction: If the –path parameter is specified, its value is ignored. Instead, the default value of / is used.

All other parameters (–port, –path, –poll, –maxrecords, –subexpire, or –password) are not required and are ignored, if specified on the command line.

Starting and stopping the data collector

Starting the data collector

This section provides the procedures that you can use to start and stop the data collector.
To start the DataPower data collector, open a command-line window, navigate to the `ITM_home\TMAITM6\KD4\bin` directory on Windows systems or the `ITM_home/platform/d4/KD4/bin` directory on Linux and UNIX systems, and run the `startDC` script. Examples:

```
startDC.bat
./startDC.sh
```

Optionally, you can specify the `-background` option to start the data collector proxy as a disconnected task. Examples:

```
startDC.bat -background
./startDC.sh -background
```

Once started, the data collector monitors the console for commands entered by the user.

If you start the data collector by running the `startDC` script in the foreground, you can stop the data collector only by running the `stop`, `quit`, or `exit` command from the command-line window. Any of these commands initiate an orderly shutdown of the DataPower data collector. The data collector waits for all communication sessions to end before the process terminates.

When in background mode or when you inadvertently exit the command-line window in foreground mode, to stop the data collector, navigate to the `ITM_home\TMAITM6\KD4\bin` directory on Windows systems or the `ITM_home/platform/d4/KD4/bin` directory on Linux and UNIX systems, and run the `stopDC` script. Examples:

```
stopDC.bat
./stopDC.sh
```

While the data collector is running, you can use the `KD4configDC` command to update the DataPower data collector configuration file and change which DataPower SOA appliances and domains are being monitored. You do not need to stop and restart the data collector to activate these changes. The running data collector detects the changed configuration file within 40 seconds and adjusts its monitoring to reflect the updated configuration.

### Running the DataPower data collector as a Windows service or UNIX daemon

ITCAM for SOA provides the capability to register and start the DataPower data collector as a service on supported Windows systems, or as a daemon on supported UNIX systems. Running the DataPower data collector in this way improves availability, because the data collector can be automatically restarted in the event of a system restart.

After installing ITCAM for SOA, the DataPower data collector is not automatically registered as a service or daemon, so you can still start the data collector using the `startDC` script (see “Starting and stopping the data collector” on page 33).

### Registering the DataPower data collector as a Windows service or UNIX daemon

To register the DataPower data collector as a Windows service or UNIX daemon, complete the following steps:
1. If the data collector is already started, stop the data collector (see “Starting and stopping the data collector” on page 33).

2. Register the DataPower data collector using either the Data Collector Configuration utility or the KD4configDC command.
   - Using the Data Collector Configuration utility:
     a. Start the Data Collector Configuration utility (ConfigDC) from the ITM_home\TMAITM6\KD4\bin directory on Windows systems or the ITM_home/platform/d4/KD4/bin directory on Linux and UNIX systems.
     b. Select the DataPower SOA Appliance runtime environment.
     c. Select the option Register or unregister DataPower as a service or daemon.
     d. Select the option Register DataPower as service or daemon
     e. The utility reminds you to stop the DataPower data collector before registering it. Click Next.
     f. Wait for the operation to complete.
     g. Exit the Data Collector Configuration utility.
   For more information about running the Data Collector Configuration utility, see Running the Data Collector Configuration utility.
   - Using the KD4configDC command on Windows systems, run this command:
     KD4configDC.bat –registerService –env 8
   On AIX, Linux, Solaris, or HP-UX systems, run this command:
     ./KD4configDC.sh –registerService –env 8
   For more information about running the KD4configDC command, see Running the KD4configDC script

On Windows systems, to verify that the DataPower data collector is registered as a service, complete the following steps:
1. From a command prompt, type services.msc to start the Windows Services Manager.
2. Locate the ITCAM for SOA WebSphere DataPower Data Collector service:

   Figure 10. Services Utility

3. Verify that the status of the ITCAM for SOA WebSphere DataPower Data Collector is set to Started.

Starting the registered DataPower data collector
After the DataPower data collector is registered as a service or daemon, it is set to start automatically when the system is rebooted. The data collector is not started immediately, however, in case the data collector is already running in console mode.

To start the DataPower data collector manually for Windows systems, use the Windows Service Manager, or run either of these commands from the ITM_home\TMAITM6\KD4\bin directory:
net start kd4dpdc
startDC.bat [-background]
To start the DataPower data collector manually on a Windows system using the Windows Service Manager, complete the following steps:

1. From a command prompt, type `services.msc` to start the Windows Services Manager.
2. Locate the **ITCAM for SOA WebSphere DataPower Data Collector** service.
3. Right-click the **ITCAM for SOA WebSphere DataPower Data Collector** service and click **Start**.

To start the DataPower data collector manually for Linux, AIX, Solaris, or HP-UX operating systems, you can run this command from the `ITM_home/platform/d4/KD4/bin` directory:

```
./startDC.sh [-background]
```

You can still start the DataPower data collector using the existing `startDC` script. On Windows systems, if the data collector is registered as a Windows service, however, the service starts instead of a console session.

**Stopping the registered DataPower data collector**

To stop the DataPower data collector manually on Windows systems, use the Windows Service Manager, or run either of these commands from the `ITM_home\TMAITM6\KD4\bin` directory:

```
net stop kd4dpdc
stopDC.bat [-background]
```

To stop the DataPower data collector manually on a Windows system using the Windows Service Manager, complete the following steps:

1. From a command prompt, type `services.msc` to start the Windows Services Manager.
2. Locate the **ITCAM for SOA WebSphere DataPower Data Collector** service.
3. Right-click the **ITCAM for SOA WebSphere DataPower Data Collector** service and click **Stop**.

To stop the DataPower data collector manually for Linux, AIX, Solaris, or HP-UX systems, run the following command from the `ITM_home/platform/d4/KD4/bin` directory:

```
./stopDC.sh [-background]
```

**Removing the DataPower data collector from the list of Windows services**

To remove the DataPower data collector from the list of Windows services, complete the following steps:

1. If the data collector is already running, stop it (see “Stopping the registered DataPower data collector”).
2. Remove the DataPower data collector from the list of Windows services using either the Data Collector Configuration utility or the `KD4configDC` command.
   - Using the Data Collector Configuration utility (see Running the Data Collector Configuration utility):
     a. Start the Data Collector Configuration utility (ConfigDC) from the `ITM_home\TMAITM6\KD4\bin` directory on Windows systems or the `ITM_home/platform/d4/KD4/bin` directory on Linux and UNIX systems.
     b. Select the DataPower SOA Appliance runtime environment.
c. Select the option Register or unregister DataPower as a service or daemon.
d. Select the option Unregister DataPower.
e. The utility reminds you to stop the DataPower data collector before you unregister it as a service. Click Next to continue.
f. Wait for the operation to complete.
g. Exit the Data Collector Configuration utility.

- Using the KD4configDC command (see Running the KD4configDC script), run the following command:
  KD4configDC.bat –deregisterService –env 8

On Windows systems, to verify that the DataPower data collector is removed from the list of registered service, complete the following steps:
1. From a command prompt, enter services.msc to start the Windows Services Manager.
2. Verify that the ITCAM for SOA WebSphere DataPower Data Collector service is no longer listed in the window.

**Deregister before uninstalling the product**

When you uninstall ITCAM for SOA, the process does not include removing the DataPower data collector from the list of Windows services. Because this process uses the Data Collector Configuration utility and the KD4configDC script, you should always deregister the DataPower data collector Windows service before you uninstall the product.

**Error handling**

Errors that occur while registering, starting, stopping, or deregistering the service are written to the Windows event log, and also to the Tivoli Monitoring reliability, availability, and serviceability (RAS) logs in the ITM_Home\logs directory.

The logfile name follows the Tivoli Monitoring logfile naming convention for ITCAM for SOA, host_name_d4_log_id.log (for example, omut3_d4_463742c2-01.log). Consult both the Windows event log and the Tivoli Monitoring RAS logs in case of service errors.

For more information about RAS logs, see the logs and data collection for troubleshooting section of the IBM Tivoli Monitoring: Troubleshooting Guide.

**Modifying the minimum and maximum JVM heap size**

Table 3 presents the recommended JVM heap size setting per transaction rate for the DataPower data collector.

<table>
<thead>
<tr>
<th>Transaction rate</th>
<th>Recommended heap size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1- 500 transactions per second</td>
<td>256 Mb</td>
</tr>
<tr>
<td>501-1000 transactions per second</td>
<td>768 Mb</td>
</tr>
<tr>
<td>1001-1500 transactions per second</td>
<td>1280 Mb</td>
</tr>
<tr>
<td>1501-2000 transactions per second</td>
<td>1792 Mb</td>
</tr>
</tbody>
</table>
To modify the minimum and maximum JVM heap size of the DataPower data collector, complete the following steps:

1. Stop the DataPower data collector:
   a. On Windows systems, go to the ITM_home\TMAITM6\KD4\bin directory and run stopDC.bat.
   b. On Linux and UNIX systems, go to the ITM_home/platform/d4/KD4/bin directory and run ./stopDC.sh.

2. Edit the startDC script to modify the JVM heap size.
   On a Windows system, complete the following steps:
   a. Edit the startDC.bat script in a text editor.
   b. Locate the line that starts with set JAVA_OPTS=.
      For example:
      ```
      set JAVA_OPTS=-Djava.protocol.handler.pkgs=com.ibm.net.ssl.www.protocol
      -Xms64m -Xmx256m -Xrs
      ```
   c. Modify the value of -Xms<size>m parameter to modify the minimum JVM heap size. Modify the value of -Xmx<size>m parameter to modify the maximum JVM heap size.
   d. Save the file.
   On a Linux or UNIX system, complete the following steps:
   a. Edit the ./startDC.sh script in a text editor.
   b. Locate the line that starts with JAVA_OPTS=
      For example: 
      ```
      JAVA_OPTS="-Djava.protocol.handler.pkgs=com.ibm.net.ssl.www.protocol
      -Xms64m -Xmx256m"
      ```
   c. Modify the value of -Xms<size>m parameter to modify the minimum JVM heap size. Modify the value of -Xmx<size>m parameter to modify the maximum JVM heap size.
   d. Save the file.

3. Start the DataPower data collector:
   a. On Windows systems, go to the ITM_home\TMAITM6\KD4\bin directory and run startDC.bat.
   b. On Linux and UNIX systems, go to the ITM_home/platform/d4/KD4/bin directory and run ./startDC.sh.

For information about resolving directory path variables, see Resolving directory path variables.

**Important:** Modifying the JVM heap size affects the performance of the DataPower data collector.

For more information about stopping and starting the DataPower data collector, see “Starting and stopping the data collector” on page 33.

---

**Configuring the requester identity type for fault monitoring**

You can configure the type of requester identity that is displayed in the Fault Details view and the Fault_Log_Table_610 table.

Beginning with ITCAM for SOA version 7.2 Fix Pack 1, a new property is added to the ITCAM for SOA KD4.dc.properties file. The kd4ира.fault.reqid.type
property specifies whether user ID or host IP address are displayed in the Requester Identity attribute in the Fault Details view and in the Fault_Log_Table_610 table.

To configure the type of requester identity to display, complete the following steps:

1. Navigate to the KD4.dc.properties file in the ITCAM4SOA_Home/KD4/config directory.
2. Edit the KD4.dc.properties file using a text editor.
3. Set the kd4ира.fault.reqid.type property to a value of 1, 2, or 3, where:
   - 1 User ID of the requester.
   - 2 Host IP address of the requester. The default value is 2.
   - 3 User ID of the requester. If user ID is not available, then use host IP address.
4. Save your changes in the KD4.dc.properties file.

If you enable service monitoring by requester identity, you might want to view the same requester identity value in the Fault Details view and in the Service Inventory Requester Identity table. To see the same value, you must set the fault and service monitoring requester identity types to the same value.

**Remember:** For service monitoring, you can set the requester identity type to either of the following values:
- User ID
- Host name or IP address

For more information about configuring requester identity for service monitoring, see “Workspaces for monitoring by requester identity” in the *IBM Tivoli Composite Application Manager for SOA User’s Guide*.

---

### Configuring the aggregation of metrics in the Service Inventory Requester Identity table

You can configure whether to aggregate metrics for service providers and requesters in the Services Inventory Requester Identity_610 table.

Beginning with ITCAM for SOA version 7.2 Fix Pack 1, a new property is added to the ITCAM for SOA KD4.dc.properties file. The kd4ира.reqid.requester.enabled property specifies whether to aggregate metrics for service providers and service requesters in the Services Inventory Requester Identity_610 table.

To configure the type of aggregation to implement, complete the following steps:

1. Navigate to the KD4.dc.properties file in the ITCAM4SOA_Home/KD4/config directory.
2. Edit the KD4.dc.properties file using a text editor.
3. Set the kd4ира.reqid.requester.enabled property to a value of 0 or 1, where:

   - 0 Aggregates metrics for service providers in the Services Inventory Requester Identity_610 table. The default value if the property is not set is 0.
   - 1 Aggregates metrics for service providers and service requesters in the Services Inventory Requester Identity_610 table.
4. Save your changes in the KD4.dc.properties file.

**Communicating between data collector and appliance**

Approximately every 10 minutes, the data collector queries the DataPower SOA appliance to see if the domain list has changed. If the domain list has changed, the data collector updates its information as needed and begins monitoring new domains defined on the Data Power appliance.

The data collector polls for data at a frequency that is defined by the –poll parameter (the default is 10 seconds).

Communication errors between the data collector and the DataPower SOA appliance that prevent the collection of data are logged to the console (the command-line session where the startDC script was executed) and to the operations log. If the data collector is running in background mode, errors are written to the operations log. Additional information concerning the communications between the data collector and DataPower SOA appliance can be obtained by setting the operations logging level for the data collector from error to info using the updateLogging Take Action command from the Tivoli Enterprise Portal.

**Optimizing performance**

Techniques for optimizing the performance of data collection and data processing are described in the following sections.

**Optimizing the performance of data processing**

The data collector is configured by default to monitor all services and operations, but with no logging of message content. This configuration optimizes the performance of processing data by not having to handle message content when it is not needed. When you modify any monitor controls to specify a content logging setting other than none, all of the full message content for all services and operations is retrieved from the appliance, and the data collector applies the configured monitor control rules (headers/body/full) by service and by operation.

See “Configuring the XML Management Interface” on page 6 for information about configuring the CaptureMode setting.

**Optimizing the performance of data collection**

The data collector is configured by default to poll the DataPower SOA appliance every 10 seconds for transaction records. By default, the data collector is configured to process a maximum of 15,000 transaction records during each polling interval. You can increase this value up to a maximum of 30,000 transaction records.

You can modify the polling interval and the number of records processed during the polling interval in line with the transaction volumes on your DataPower appliance. Modifying these properties provides the DataPower data collector with more control over the number of transaction records that it processes and helps to maximize the performance of the data collector.
In the KD4.dpdcConfig.properties file, the DataPower.poll property specifies the polling interval and the DataPower.maxrecords property specifies the maximum number of records to process per polling interval. Both properties are optional.

The combined values of the polling interval and the maximum number of records specifies a transaction rate for data collection. For example, to specify a transaction rate of 1,000 transaction records per second, set the DataPower.maxrecords property to 1000 and set the DataPower.poll property to 1.

You might observe record loss when you set the maximum number of records to retrieve to too low a value for your environment. On the DataPower appliance, if the number of records in the WSM buffer exceeds the maximum record size limit of the buffer, records might be lost. If you observe high bursts of traffic on your DataPower appliance, you might decide to increase the maximum record size limit of the buffer. For more information about modifying the maximum record size limit of the buffer, see “Configuring the XML Management Interface” on page 6.

Depending on your deployment, you might observe high CPU usage on the system where the data collector is installed during the processing of transaction data from the DataPower appliance when the maximum number of records is set to too high a value or when the polling interval is too frequent, or both.

To prevent high CPU usage and data loss from occurring, tune the polling interval and the maximum number of records for your DataPower deployment.

To modify the DataPower.poll and the DataPower.maxrecords properties, issue the KD4configDC command with the -maxrecords and -poll parameters. For example:

```
KD4configDC -enable -env 8 -host appHost -user user1 -poll 1 -maxrecords 1000
```

**Restriction:** In an upgrade scenario, to set the maximum number of records for an existing display group, you must add the DataPower.maxrecords property manually to section in the KD4.dpdcConfig.properties file that configures the display group.

For more information about issuing the KD4configDC command, see “Enabling data collection using the KD4configDC command” on page 23.

If you observe that CPU usage is high on the system where the data collector is running, you can also increase the maximum JVM heap size to improve the performance of the data collector. To modify the maximum JVM heap size, see “Modifying the minimum and maximum JVM heap size” on page 37.

---

**Limitations**

You can control the monitoring to limit the collection of data to specific services and operations, but you cannot use filter controls to reject messages of specific services or operations. To filter (reject) messages, use the Policy Rules and Actions of the DataPower Web Services Proxy configuration.

If the transforms are not activated, the data collector does not generate a meaningful correlator for messages flowing through the DataPower SOA appliance, so this data does not render properly in the topology, sequence diagram, and pattern views in the IBM Web Services Navigator. For more information, see “Configuring processing rules for DataPower Web Services Proxy gateways” on page 7.
The DataPower data collector creates metric, content, operation and trace log files, but because message filtering is not supported, it does not create any action log files.

Depending on your configuration, the data collector creates multiple metric and content log files to describe the existence of several different display groups. Each display group is represented as a separate node in the Tivoli Enterprise Portal Navigator view. The log files for each display group hold the data for all of the DataPower SOA appliances that are configured for that display group, which might span multiple DataPower domains and include data from multiple DataPower SOA appliances.

Before you attempt to delete an application domain from DataPower that is being monitored by the DataPower data collector, you must either run the KD4configDC command to disable monitoring of that application domain, or stop the data collector. Otherwise, when you attempt to delete the domain using the DataPower administration console, you receive an error and the domain cannot be deleted.

Troubleshooting

If you encounter a communication problem, time synchronization problems, or password problem, refer to the information in the following sections to help resolve the problem.

For more information about resolving problems with the DataPower data collector, see the IBM Tivoli Composite Application Manager for SOA Troubleshooting Guide.

Communication failures

If a communications failure occurs, the data collector attempts to initialize itself again and reestablish communication with the target DataPower SOA appliance after the next polling interval (the polling interval is specified by the –poll parameter). This process is repeated each polling interval until communication is successfully reestablished, or until the data collector is stopped. Setting the operations logging level to info gives you the best indication of status of the communications between the data collector and the appliance.

Synchronizing time between computer systems

For data to display properly in the Tivoli Enterprise Portal, the DataPower SOA appliance, the data collector system, and the Tivoli Enterprise Monitoring Server system must have their clocks synchronized within 5 minutes of each other, in terms of the UTC time they report.

Password problems

User IDs and passwords for DataPower SOA appliances are created and maintained at the appliance. See the documentation that is provided with your DataPower SOA appliance for information about creating and managing user IDs and passwords. When you have reset this information at the DataPower SOA appliance, use the KD4configDC command to update the DataPower configuration file with the latest user ID and password values for the appropriate section of the file.
Logging and Tracing

The data collector in the DataPower environment uses the same operation and trace log files as other environment types supported by ITCAM for SOA, but rather than creating separate operation and trace files for each display group, it creates a single operation file and a single trace file for all display groups.

The view in the Tivoli Enterprise Portal continues to operate as though you can configure logging and tracing for each display group, but the settings for all DataPower display groups are considered when the settings for the single log or trace writer instance are initialized, as follows:

- If any display group has trace set to ON, the DataPower data collector records tracing information for all display groups to the trace log.
- The DataPower data collector uses the lowest severity log setting (Info, Warning, or Error) of all display groups to determine what messages it records in the operations log. For example, if a display group has a log setting of Info, then all informational, warning, and error messages for all display groups are written to the operations log.

The DataPower data collector does not update the global configuration settings seen in the Tivoli Enterprise Portal to reflect the actual settings being used for the various display groups. As a result, the Tivoli Enterprise Portal might still show different log and trace settings for each display group, even though the data collector is using common settings for all display groups.

In addition to the operations log, the DataPower data collector issues messages to the console where the startDC script was run. All messages issued to the console are also recorded in the operations log. In background mode, information is written to the operations log. On Windows, errors might also be displayed in the Windows Application Event log.

(Optional) Verifying that the configuration of the data collector

After you have verified that the ITCAM for SOA agent has been installed and configured, you might want to verify that your DataPower data collector is configured correctly and monitoring data. To verify the configuration of your data collector, complete these additional tasks.

Verifying that the DataPower data collector is configured and started

Complete the following steps:

1. Verify that the DataPower data collector is started.
   When the data collector is started, an operations log is created. The operations log is in the ITCAM4SOA_Home/KD4/logs directory on Windows systems and in the ITCAM4SOA_Home/KD4/logs directory on Linux and UNIX systems. If there is no operations log, you must start the DataPower data collector (see “Starting and stopping the data collector” on page 33). If the data collector fails to start, check the operations log for failures. Communication failures or login failures are recorded in the operations log.

2. Verify that the DataPower data collector properties file, KD4.dpdcConfig.properties, exists.
The properties file is created when you enable data collection for one or more DataPower appliances using either the Data Collector Configuration utility or the KD4configDC command (see "Enabling data collection" on page 19).

The properties file is in the ITCAM4SOA_Home/KD4/config directory on Linux or UNIX systems and in the ITCAM4SOA_Home\KD4\config directory on Windows systems.

3. Verify that the content of the properties file matches the configuration changes that you made using the Data Collector Configuration utility or the KD4configDC command. Make sure that the connection to each DataPower appliance is configured correctly in this file:
   - Each section of the file represents a connection to a single DataPower SOA appliance. For each connection, the file stores the host name, port, user ID, and password that is needed to communicate with the device.
   - Each section of the file might specify the domains to monitor on the appliance.
   - Each section of the file might specify a display group that data from the appliance is displayed under in the Tivoli Enterprise Portal.

For example:

```
# Sample DataPower data collector configuration file
DataPower.count=1
#
DataPower.host.1=dpbox1
DataPower.port.1=5550
DataPower.path.1=/
DataPower.poll.1=60
DataPower.user.1=admin
DataPower.encpswd.1=#$%&
DataPower.domainlist.1=default,testdom1
DataPower.displaygroup.1=dpbox1
DataPower.subExpire.1=15
DataPower.maxrecords.1=1000
```

The property DataPower.count represents the number of DataPower appliances that are configured in the properties file. Verify that its value matches the number of sections in the file.

4. Verify that the user ID and passwords that are specified in the connection properties for each DataPower SOA appliance are correct.

To ensure that the password has not expired and that the user credentials have been used at least once to the log in to the appliance, log in to the DataPower WebGUI using the user ID and password configured in the properties file.

**Verifying that the DataPower appliance is storing transaction data**

Complete the following steps:

1. Confirm that the DataPower SOA appliance is configured to store data about business requests in the WSM agent that is associated with each domain:
   a. Confirm that Web Service Management option is selected for the DataPower XML Management Interface object.
   b. If you want to save metric records when the data collector is not started, from the WebGUI, navigate to Objects > Device Management > Web Services Management Agent. Verify that the WSM buffering mode option on the DataPower appliance is set to Buffer for each domain.

For more information about setting the Web Service Management option, see "Configuring the DataPower SOA appliance for monitoring" on page 5.
2. Verify that traffic is flowing through the DataPower device.

   From the WebGUI, navigate to Status > Web Service > Web Services Operation Metrics to verify that metrics are being generated. Traffic must be associated with the domains specified in the domain list in the properties file.

3. Verify that each WSM agent on the DataPower appliance is storing data about the request and response messages.

   To view the status of a WSM agent from the WSM Agent Status page, from the WebGUI, navigate to Status > Web Service > WSM Agent Status. The page displays the status of the WSM agent for each domain, for example:

   - Active Subscribers: 0
   - Records Seen: 1049
   - Records Lost: 0
   - Pending Records: 3
   - Complete Records: 87

   Where:

   **Records Seen**
   - When a transaction is processed for a domain by the DataPower Web Service Proxy, the Records Seen count increments by 1.

   **Records Lost**
   - The number of records that are discarded when the buffer is full.

   **Records Pending**
   - The number of requests that are waiting on a server response.

   **Complete Records**
   - The number of transactions that are ready for collection. This number reduces each time ITCAM for SOA collects data from the DataPower appliance.

   **Active Subscribers**
   - The Active Subscribers count increases from 0 to 1 when ITCAM for SOA collects data from the WSM. When ITCAM for SOA has collected the data, the number of complete records decreases to 0.

   Remember: When you reboot the appliance, the counters are reset.

**Verifying that metric log files are generated**

   Complete the following steps:

1. Verify that metric log files are generated for the DataPower appliance.

   Navigate to the $ITCAM4SOA_HOME\KD4\logs$ directory on Windows systems or the $ITCAM4SOA_HOME/KD4/logs$ directory on UNIX or Linux systems and verify that a metric file was created for the DataPower data collector in the format: $KD4.8.hostname.domain_name.display_group.metric.log$.

2. Verify that logs are being moved to the $ITCAM4SOA_HOME\KD4\logs\KD4.DCA.CACHE$ directory on Windows systems or the $ITCAM4SOA_HOME/KD4/logs/KD4.DCA.CACHE$ directory on Linux or UNIX systems approximately every 10 seconds after the contents of the file are read by the monitoring agent. If the files are not moved, the monitoring agent might not be running.

3. If you want to collect message content in addition to summary metrics, verify that the **CaptureMode** setting is set to all-messages for each domain that you are monitoring. To verify this setting, from the WebGUI, navigate to Services > Miscellaneous > Web Services Agent.
Verifying that web service proxy is configured correctly

Complete the following steps:

1. If you are monitoring traffic through a web service proxy, verify that traffic is being processed by the WS-proxy service on the Data Power SOA appliance.
   a. Log in to the administration console as admin in the default domain.
   b. Navigate to Status > Web Service > Web Service Operation Metrics. This page shows the number of times each operation has been invoked since the service was restarted. If this page does not show increasing counts, then this service is not the service that is being invoked.

2. Verify that performance summary metrics and topology data are displayed for the web service proxy in the Tivoli Enterprise Portal.

   If topology data is not displayed, the processing rules for the web service proxy might not be configured correctly. The correlator that is used to track topology data is not processed automatically by the web service proxy. You must add some transforms, which are included in your DataPower firmware, to the processing rules to handle the correlator.

   For more information about configuring processing rules for the web service proxy, see "Configuring processing rules for DataPower Web Services Proxy gateways" on page 7.

Verifying that the multi-protocol gateways are configured correctly

Verify that processing rules have been configured for monitoring traffic through the multi-protocol gateway. No monitoring of the multi-protocol gateways occurs until these processing rules have been added. Unlike the web service proxy, no default transforms are provided with the device. Instead, your application developers must add additional logic to the processing rules based on sample style sheets that are provided to correlate request and response messages and to add the data collected to the WSM for collection by ITCAM for SOA.

If you are monitoring a multi-protocol gateway, inspect the processing rules manually to confirm that the correlator is being calculated on the request and response messages, that the transaction data collected is sent to the WSM agent for ITCAM for SOA to collect, and that the necessary transforms have also been added for processing error message.

For more information about configuring processing rules for the multi-protocol gateway, see "Configuring processing rules for DataPower Multi-Protocol Gateways" on page 9.

Displaying the DataPower Console interface

You can select a DataPower SOA appliance that is displayed in a row of the Services Inventory attributes table in the Performance Summary workspace of the Tivoli Enterprise Portal, or select an instance of a DataPower mediation operation from one of the Operational Flow displays, and follow the procedure described in this section to display the DataPower Console user interface in a Tivoli Enterprise Portal view. You can then use this interface to configure the DataPower SOA appliance and the policies that the appliance applies to services traffic.

To display the DataPower Console user interface, complete the following steps:
1. From the Tivoli Enterprise Portal, navigate to the Performance Summary workspace and display the Services Inventory attributes table view.

2. Select a row in the table from the DataPower system referenced in the **Node Name** column, and click the link icon to select the DataPower Console option. The DataPower Console is displayed in a Tivoli Enterprise Portal view, similar to the example shown in Figure 11.

![DataPower Console](image)

Figure 11. The DataPower Console

**Considerations when displaying the DataPower Console view**

Additional limitations, restrictions, and considerations to keep in mind as you launch the DataPower Console user interface are presented in the following sections:

**Start only from a row that represents a DataPower environment**

You cannot start a DataPower Console session from a row in the Services Inventory attributes table that represents a non-DataPower application server runtime environment. DataPower SOA appliances are identified by a value of **DataPower** in the Application Server Environment column of the Services Inventory attributes table.

If you select a row that does not represent metrics data from a DataPower application server runtime environment, a browser session is started with the KD4LN0001E error message displayed.

If you select a row that does not represent a DataPower appliance, the link to the DataPower Console is not displayed.

**Cannot start from a row with aggregated DataPower metrics**

Because of how the data collector works and the way that the **Application Server Node Name** field is populated by the DataPower data collector, you cannot display...
the DataPower Console from a Services Inventory attributes table row that represents metrics aggregated from more than one physical DataPower SOA appliance.

In the case of aggregated metric data from more than one DataPower SOA appliance, the **Node Name** field is blank to prevent possibly displaying the DataPower Console for the wrong DataPower SOA appliance. If you select a row in the table that represents metrics aggregated from more than one physical DataPower SOA appliance, a browser session is started with error message KD4LN0002E displayed.

**The DataPower Console must be configured for Port 9090**
The link to the DataPower Console operates correctly only if the DataPower Console is configured for port 9090 (port 9090 is also the default port number). See the *DataPower WebGUI Guide* and the *DataPower CLI Reference Guide* for information about configuring the port.

**Logging in to DataPower**
If you have not previously logged in to the DataPower Console user interface in the current browser session, the web browser session opens with the login page. You need to log in with a valid user name and password to continue. If you have previously logged in, the session opens with the main page of the user interface.

See your DataPower documentation for more information about user names and password, and consult your local system administrator for assistance if needed.
Index

A
aggregation 4
across multiple domains 30
data collector, background mode 33
appliance
aggregating multiple, DataPower 2
cluster, DataPower 30
cluster of appliances 30
DataPower 2, 28
planning for 3
using display group 28
user account, configuring 5
user password 27
DataPower Console
considerations 47
display group
aggregation and separation 28
changing current value 27
DataPower 4
displaygroup
in node name, DataPower 25
domain
aggregating multiple, DataPower 2
application, DataPower 3
domain list
changing current value 27
restricting user access 28

B
background mode
DataPower data collector 33

C
cluster
DataPower SOA appliances 30
configuration file
DataPower 2, 19

D
data collector
communicating with appliance 40
configuring in DataPower environment 19
DataPower 1
DataPower, deploying 4
DataPower, starting 33
DataPower, stopping 33
DataPower, unconfiguring 5
disabling for DataPower 33
DataPower appliance, cluster 30
appliance, configuring 5
appliance, polling 2
configuration file 19
configuring environment for the data collector 19
Console user interface in TEP 46
data collector, background mode 33
data collector, starting and stopping 33
disabling data collector 33
firmware upgrade 1, 5
log in 48
multiple connections 28
proxy 2
style sheets 13

F
firmware
DataPower requirement 5
DataPower requirements 1

H
host
in node name, DataPower 25

K
KD4configDC
DataPower, optional parameters 28
for DataPower 23
parameters, DataPower 23
KD4LN0001E error message 47
KD4LN0002E error message 48

L
limitations
DataPower 41
logging
DataPower 43

M
Multi-Protocol Gateway
DataPower 1
multiple connections
DataPower 28

N
node name
in TEP, for DataPower 25

P
parameters
KD4configDC for DataPower, optional 28
password
DataPower, specifying 27
performance
DataPower 40
Performance Summary workspace 46
poll
DataPower SOA appliance 2
processing rules
configuring, for DataPower 7
proxy
DataPower 2

R
requester identity,
configuring 38, 39

S
separation
between domains by domain list 29
by display group 29
by host name 29
DataPower 28
using display group 28
Services Inventory attributes table 46
startDC 33, 40
starting DataPower data collector 4
stopDC 33
stopping DataPower data collector 5

T
Tivoli Enterprise Portal
display group, DataPower 4
node names, DataPower 25
tracing
DataPower 43
transaction log service
DataPower 2
troubleshooting 42, 43
appliance password 42
communication 42
synchronizing time 42

U
updateLogging Take Action 40

W
web services proxy
DataPower 1
XML Management Interface enabling, for DataPower 6