Bharat Bhushan

25th September, WebSphere User Group

WebSphere DataPower Release 5.0.0 SLM

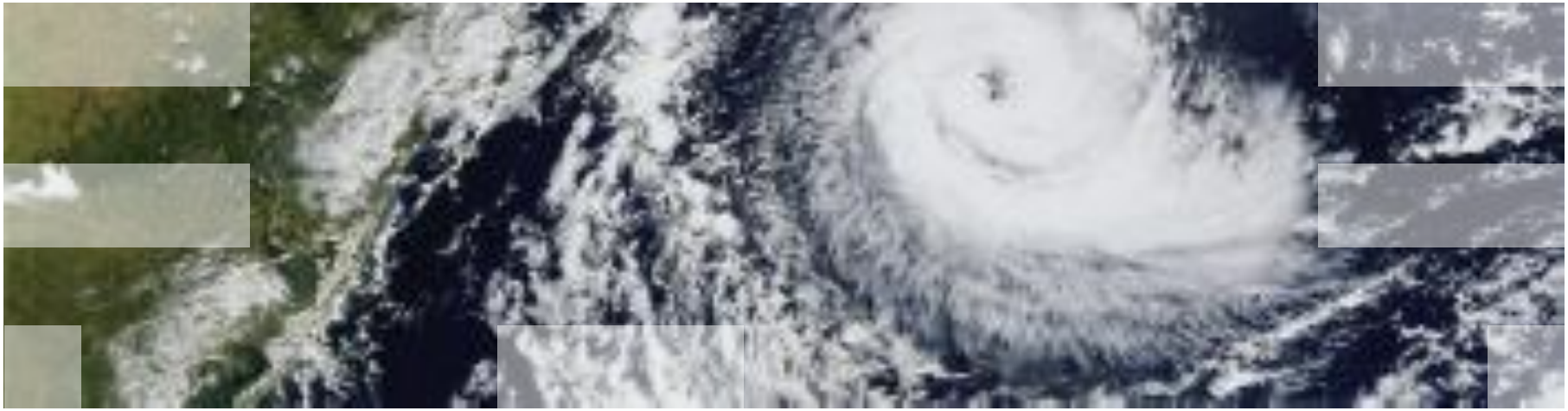


Table of contents

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

What It Is

A new configuration option for SLM peering that uses IP multicast packets as its means of communications.

Why It Is

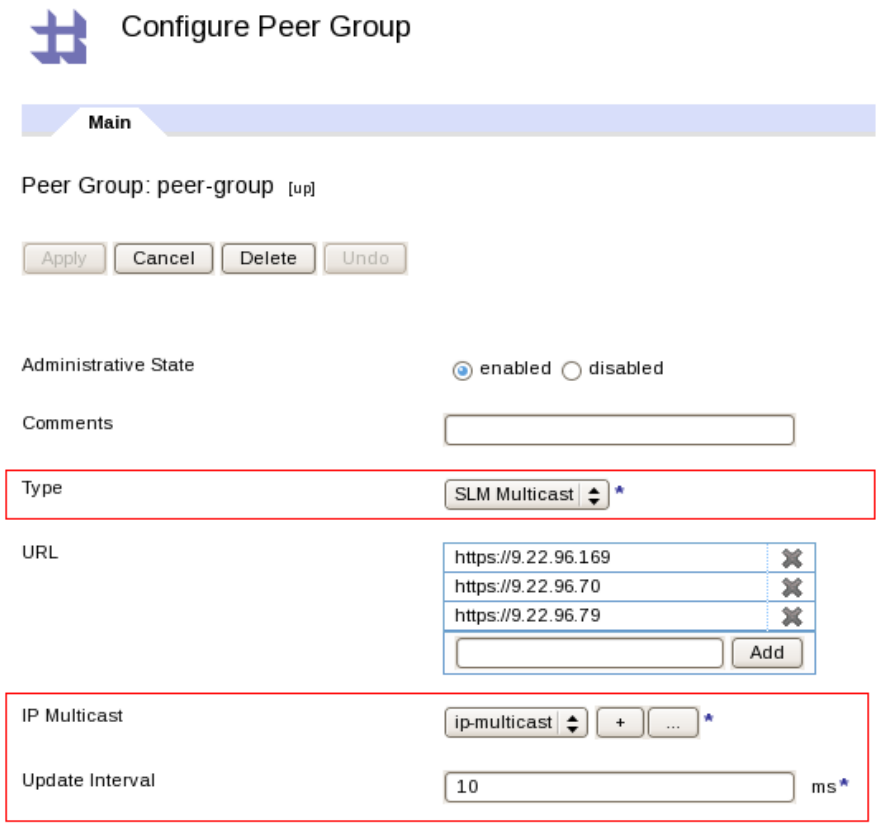
The current Unicast implementation of SLM peering has a minimum 1 second interval for updates between the peers. This does not always allow for accurate SLM enforcement between the peers within the group. Multicast peering allows update intervals as low as 1ms, which provides more accurate peering enforcement and more efficient handling of high incoming data rates.

Who Wants It

Customers who require very accurate SLM enforcement when using SLM peering, or use SLM peering with high incoming data rates.

How to Use It

Selecting “SLM Multicast” as the ***Type*** allows the ***IP Multicast*** object and ***Update Interval*** to be configured, as shown:



- **Type** - the type of peer group, either SLM Unicast or SLM Multicast

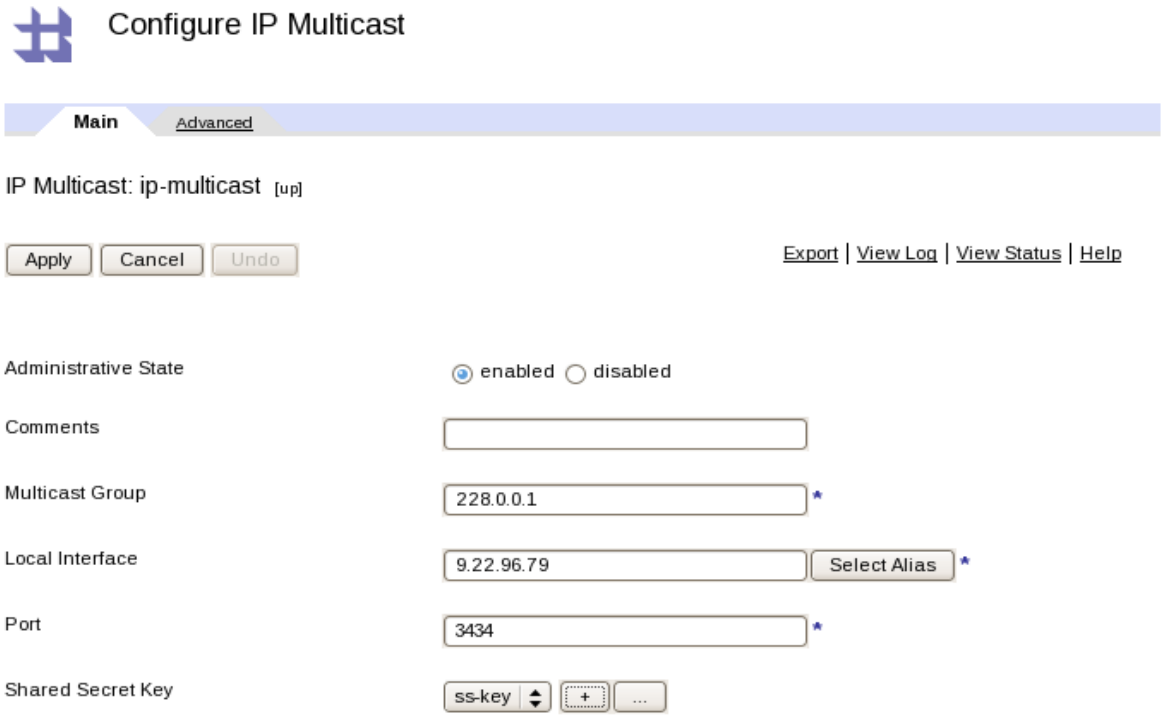
- **URL** - the URL of each peer. A port number is not required for multicast peers. The IP address of the local peer must match the Local Interface of the IP Multicast object.

- **IP Multicast** - an IP Multicast object that handles the underlying transmission of packets used for data exchange between the peers

- **Update Interval** - the time interval in milliseconds that data is transmitted between the multicast peers

How to Use It

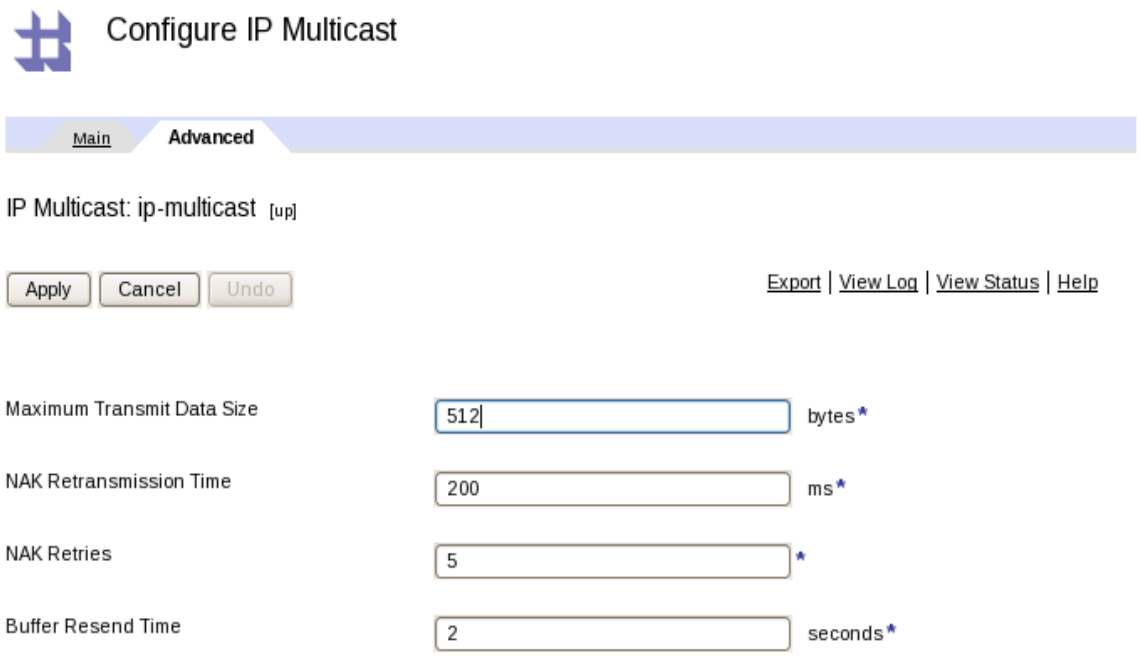
The main configuration options for the IP Multicast object are as shown:



- **Multicast Group** - the multicast IP group address
- **Local Interface** - the local address of the Ethernet interface used to transmit and receive multicast messages.
- **Port** - the IP port to use for transmitting and receiving multicast packets
- **Shared Secret Key** - the shared secret key to sign and verify multicast packets between peers (optional)

How to Use It

The Advanced configuration options for the IP Multicast object are as shown:



- **Maximum Transmit Data Size** - the maximum allowable size of a transmitted data block
- **NAK Retransmission Time** - the length of time to wait before a NAK is initially transmitted or is retransmitted
- **NAK Retries** - the number of times a missing NAK packet is retransmitted
- **Buffer Resend Time** - the number of seconds that a transmitted packet remains buffered

How to Use It

The status of the IP Multicast objects is as shown:

IP Multicast Status

Refresh Status

IP Multicast	Multicast Sender IP Address	Valid Packets Received	Invalid Packets Received	Packets Lost	NAKs Sent	NAKs Received
ip-multicast	9.22.96.70	59	0	0	0	0
ip-multicast	9.22.96.169	68	0	0	0	0

- **IP Multicast** - The name of the object transmitting and receiving multicast packets
- **Multicast Sender IP Address** - The IP address of the sender (remote peer)
- **Valid Packets Received** - The number of valid packets received at the local peer
- **Invalid Packets Received** - The number of invalid packets received (for example, duplicate packets)
- **Packets Lost** - The number of packets detected as lost
- **NAKs Sent** - The number of NAK packets sent to the multicast sender
- **NAKs Received** - The number of NAK packets received from the multicast sender

Known Limitations

The local interfaces used for the IP multicast traffic between the peers must be in the same subnet. It is recommended that the IP multicast interfaces are connected on a dedicated subnet.

Troubleshooting

The IP multicast status provider will show whether the multicast packets successfully reach each peer in the peer group. The presence of sent/received NAKs or lost packets indicates either a network problem that should be resolved, or a poor configuration of the IP multicast objects. All peers within the peer group must have the identical SLM/peer/IP multicast configuration (except for the local interface of the IP multicast object).

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

What It Is

The capability has been added to allow multiple SLM actions to be configured within a single multistep processing policy. Prior to this change, only a singe SLM action within a multistep processing policy was supported.

Why It Is

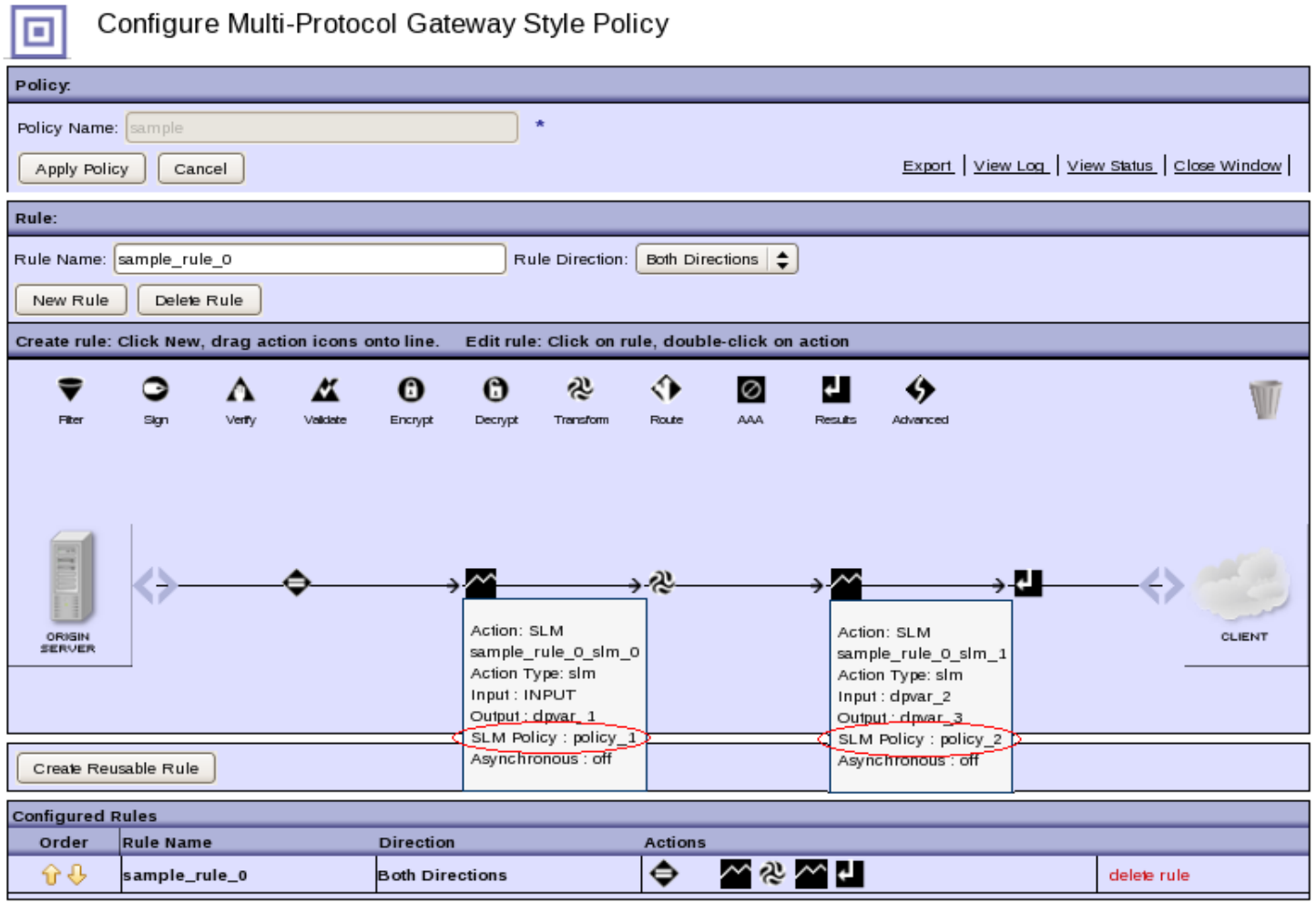
Prior to release 5.0.0, multiple SLM actions within a single processing rule were not supported. It was configurable but did not work properly, potentially leading to indefinite throttling once the threshold was reached. With this change, more than one SLM action within a single processing rule can be configured, and the underlying SLM policies will behave as expected. This is a common practice within governance policies where SLAs and SLDs are used.

Who Wants It

Customers who require different SLM actions to be enforced within different points of their multistep processing policies.

How to Use It

Note below circled in red where two different SLM actions are configured within the processing policy:



Known Limitations

If the same SLM policy is used more than once in the same multistep processing policy, then it will only be enforced once, at its initial location.

Troubleshooting

Multistep probes can be used to monitor the output of the SLM actions.

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

What It Is

SLM schedule objects can now be configured for date range limits, in addition to the existing time and day range limits.

Why It Is

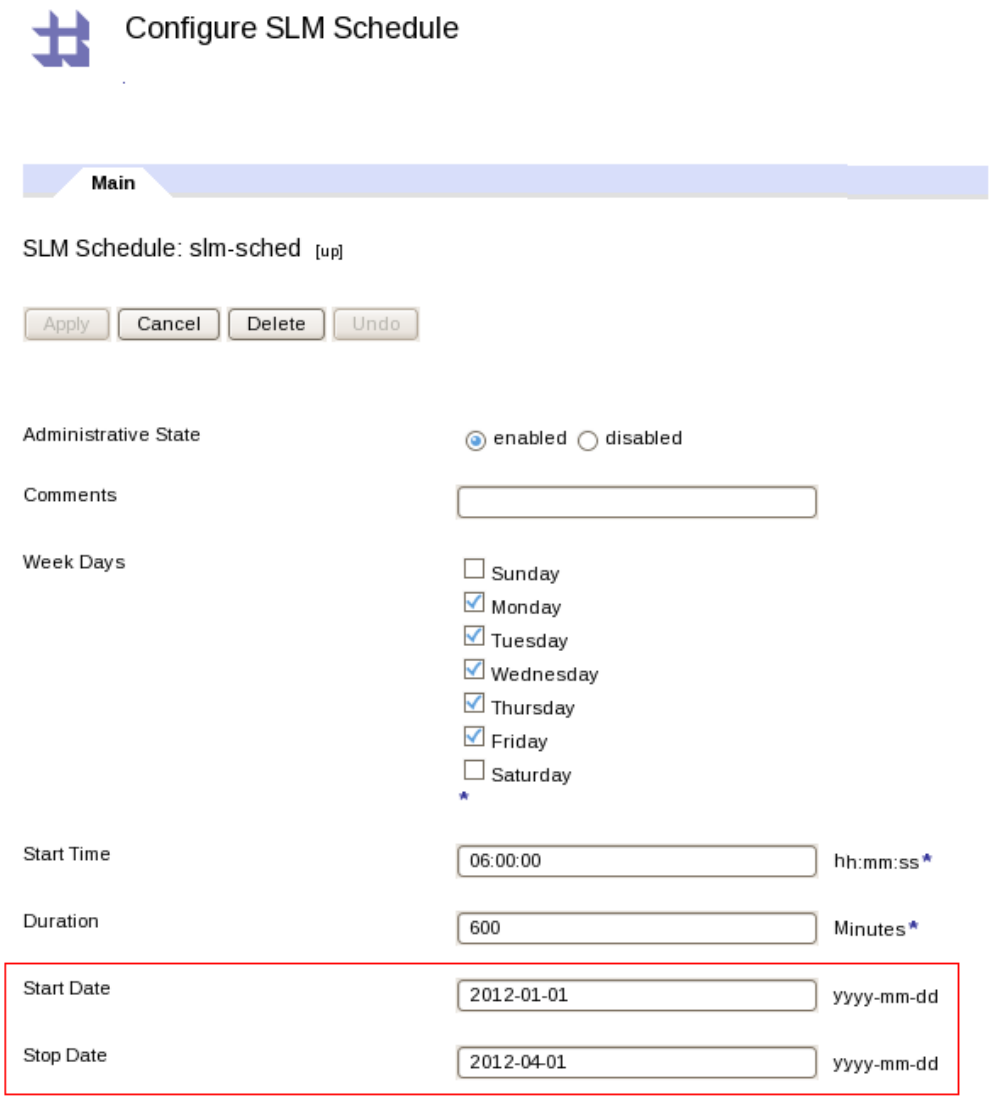
This capability provides additional flexibility in scheduling when an SLM policy will be enforced, to include date ranges. It also provides needed functionality to WSRR governance policies.

Who Wants It

Customers who require their SLM policies to be enforced during certain date ranges. For example, a special SLM policy might be in force for the “Black Friday” weekend.

How to Use It

The new fields are outlined in red below:



Known Limitations

Note that the ***Stop Date*** value is the first date that the SLM policy is NOT enforced.

For an SLM statement to be enforced, all of the day, time and date criteria must be met. The date range, as well and the existing day and time ranges, correspond to the local time zone configured for the appliance.

Troubleshooting

None.

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

What It Is

This feature improves the limitations on the number of unique SLM credential and resource requests.

Why It Is

Prior to this change, the number of unique SLM credential requests was set at 16K, and SLM resource requests to 1K. Once this number of credentials or resources was processed, all subsequent messages with new credentials or resources were rejected until the system was restarted. With this change, credential or resource values that are no longer active can be re-used for new values.

Who Wants It

Customers who use the “per extracted value” ***Match Type*** for their SLM credential or resource classes, and expect to encounter a large number of different credentials or resources.

How to Use It

No addition configuration is required to use this feature.

Known Limitations

The overall limit for **active** SLM credentials is still 16K, and 1K for resources.

Troubleshooting

If a message causes the SLM credential or resource limit to be reached, then an error message is logged.

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

What It Is

Output of the SLM multistep action written to the output context in the form of XML is being enhanced to include additional information about the SLM state. This additional information includes:

- The resource and credential class types
- The threshold interval length and type
- The threshold algorithm and type
- The threshold limit and current value

An XSD file representing the schema of the SLM XML output is also being published.

Why It Is

This additional information, particularly the current threshold value, can be dynamically parsed using XSLT from the SLM XML output and used within the processing policy to make better decisions with regard to message processing.

For example, a back-end timeout value can be dynamically modified based on how close the current threshold value is to its limit. With this information, a single SLM statement can be used for this purpose instead of having to configure a large number of SLM statements with “tiered” threshold limits. This makes message processing much more efficient.

Who Wants It

Customers who want fine-grained control of their processing policies based on the current SLM values.

How to Use It

The new fields for 5.0.0 are highlighted in red on the right-hand side.

Pre-5.0.0 SLM XML output

```
<SLMResults>
    <PolicyName>slm-policy</PolicyName>
    <ExecutionPolicy>terminate-at-first-reject</ExecutionPolicy>
    <Statement>
        <SLMId>1</SLMId>
        <UserString>stmt1</UserString>
        <Resource>
            <result>
                <match />
                <match-type>NoClassifier</match-type>
                <value>*</value>
            </result>
        </Resource>
        <Credential>
            <result>
                <match />
                <match-type>NoClassifier</match-type>
                <value>*</value>
            </result>
        </Credential>
        <CheckResult>accept</CheckResult>
        <StatementResult>accept</StatementResult>
    </Statement>
</SLMResults>
```

5.0.0 SLM XML output

```
<SLMResults>
    <PolicyName>slm-policy</PolicyName>
    <ExecutionPolicy>terminate-at-first-action</ExecutionPolicy>
    <Statement>
        <SLMId>1</SLMId>
        <UserString>stmt1</UserString>
        <Resource>
            <type />
            <result>
                <match />
                <match-type>NoClassifier</match-type>
                <value>*</value>
            </result>
        </Resource>
        <Credential>
            <type />
            <result>
                <match />
                <match-type>NoClassifier</match-type>
                <value>*</value>
            </result>
        </Credential>
        <ThreshIntervalLength>60</ThreshIntervalLength>
        <ThreshIntervalType>fixed</ThreshIntervalType>
        <ThreshAlgorithm>greater-than</ThreshAlgorithm>
        <ThresholdType>count-all</ThresholdType>
        <ThresholdLevel>10</ThresholdLevel>
        <ThresholdValue>1</ThresholdValue>
        <CheckResult>accept</CheckResult>
        <StatementResult>accept</StatementResult>
    </Statement>
</SLMResults>
```

Known Limitations

None.

Troubleshooting

This output can also be examined by enabling a probe for the output context of the SLM action.

SLM Multicast Peering

Multiple SLM Actions

Date Support for SLM Schedules

Improvement of SLM Resource and Credential Limits

SLM XML Output Enhancements

SLM Threshold Variability

What It Is

Two new extension functions that allow the SLM threshold level to be dynamically retrieved and set for concurrency-based SLM statements. Typically these functions would be used with a custom XSLT style sheet within an SLM credential or resource class to dynamically query and alter the threshold level for the transaction.

Why It Is

Currently, the threshold level is fixed for each SLM statement. If a customer wishes to have different threshold levels based on different conditions (e.g. part of the message content), then he/she must create a separate SLM statement for each threshold level. This change instead allows a single SLM statement to be used, making the message processing much more efficient and the configuration more maintainable.

Who Wants It

Customers who have complex SLM configurations. A single SLM statement can now dynamically set its threshold level based on a set of easily maintainable criteria (e.g. an XML file that is read in) matched against some property of the incoming message.

How to Use It

The extension function definitions are as follows:

slm-set-threshold-level()

Syntax

dp:slm-set-threshold-level(*name*, *ID*, *level*)

Parameters

name (`xs:string`) Specifies the name of the SLM policy.

ID (`xs:int`) Specifies the ID of the SLM statement.

level (`xs:unsignedInt`) Specifies the value to set the threshold level.

Results

An `xs:boolean` that specifies whether the threshold level is set. The function returns `TRUE` if the threshold level is set or returns `FALSE` if the threshold level is not set.

slm-get-threshold-level()

Syntax

dp:slm-get-threshold-level(*name*, *ID*)

Parameters

name (`xs:string`) Specifies the name of the SLM policy.

ID (`xs:int`) Specifies the ID of the SLM statement.

Results

An `xs:unsignedInt` that specifies the value of the threshold level.

Known Limitations

Setting the threshold level applies to the current transaction only, and only affects statements that are configured for concurrent connections or transactions. It does not affect statements configured for fixed or moving intervals.

Troubleshooting

None.

Questions