



## Configuring Performance Monitoring

---

Performance Monitor is a carrier Ethernet software feature which provides:

- Monitoring delay measurements
- Monitoring loss measurements
- Monitoring ECE
- Monitoring EVC
- Storing the monitored data
- Transferring the monitored data

For information on configuring EVC, see [How to Configure Ethernet Virtual Circuit](#).

For information on configuring MEP, see [Creating MEP Configuration](#).

- [Restrictions for Configuring Performance Monitoring, page 1](#)
- [ITU-T Y.1731 Performance Monitoring in a Service Provider Network, page 1](#)
- [How to Configure Performance Monitoring, page 3](#)
- [Verifying Performance Monitoring, page 43](#)

## Restrictions for Configuring Performance Monitoring

- Performance monitoring is not enabled by default for loss measurement (LM), delay measurement (DM), EVC, and ECE.

## ITU-T Y.1731 Performance Monitoring in a Service Provider Network

ITU-T Y.1731 performance monitoring provides standard-based Ethernet performance monitoring that encompasses the measurement of Ethernet frame delay, frame-delay variation, and throughput as outlined in the ITU-T Y.1731 specification and interpreted by the Metro Ethernet Forum (MEF). Service providers offer

service level agreements (SLAs) that describe the level of performance customers can expect for services. This document describes the Ethernet performance management aspect of SLAs.

## Frame Delay and Frame-Delay Variation

The Frame Delay parameter can be used for on-demand OAM measurements of frame delay and frame-delay variation. When a maintenance end point (MEP) is enabled to generate frames with frame-delay measurement (ETH-DM) information, it periodically sends frames with ETH-DM information to its peer MEP in the same maintenance entity. Peer MEPs perform frame-delay and frame-delay variation measurements through this periodic exchange during the diagnostic interval.

An MEP requires the following specific configuration information to support ETH-DM:

- MEG level—MEG level at which the MEP exists
- Priority
- Drop eligibility—marked drop ineligible
- Transmission rate
- Total interval of ETH-DM
- MEF10 frame-delay variation algorithm

A MEP transmits frames with ETH-DM information using the TxTimeStampf information element. TxTimeStampf is the time stamp for when the ETH-DM frame was sent. A receiving MEP can compare the TxTimeStampf value with the RxTimeef value, which is the time the ETH-DM frame was received, and calculate one-way delay using the formula  $frame\ delay = RxTimeef - TxTimeStampf$ .

One-way frame-delay measurement (IDM) requires that clocks at both the transmitting MEP and the receiving MEPs are synchronized. Measuring frame-delay variation does not require clock synchronization and the variation can be measured using IDM or a frame-delay measurement message (DMM) and a frame-delay measurement reply (DMR) frame combination.

If it is not practical to have clocks synchronized, only two-way frame-delay measurements can be made. In this case, the MEP transmits a frame containing ETH-DM request information and the TxTimeStampf element, and the receiving MEP responds with a frame containing ETH-DM reply information and the TxTimeStampf value copied from the ETH-DM request information.

Two-way frame delay is calculated as  $frame\ delay = RxTimeb - TxTimeStampf$ , where RxTimeb is the time that the frame with ETH-DM reply information was received. Two-way frame delay and variation can be measured using only DMM and DMR frames.

To allow more precise two-way frame-delay measurement, the MEP replying to a frame with ETH-DM request information can also include two additional time stamps in the ETH-DM reply information:

- RxTimeStampf—Time stamp of the time at which the frame with ETH-DM request information was received.
- TxTimeStampb—Time stamp of the time at which the transmitting frame with ETH-DM reply information was sent.

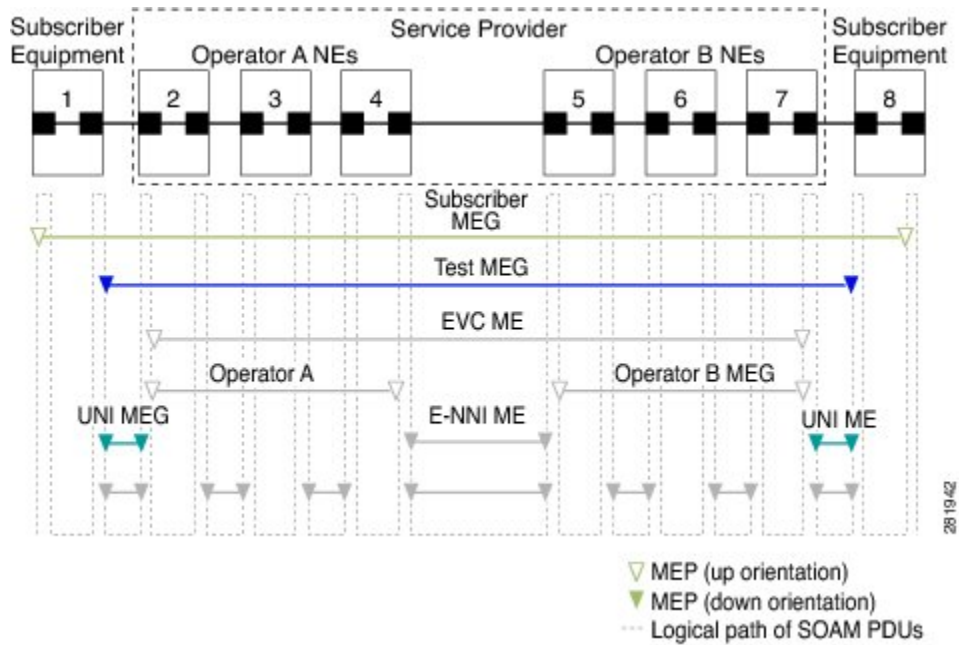


**Note** The frame-delay and frame-delay variation measurement processes are aborted when faults related to continuity and availability occur or when known network topology changes occur.

An MIP is transparent to the frames with ETH-DM information; therefore, an MIP does not require information to support the ETH-DM function.

The figure below shows a functional overview of a typical network in which Y.1731 performance monitoring is used.

**Figure 1: Y.1731 Performance Monitoring**



# How to Configure Performance Monitoring

## Provisioning the Controller to Configure Performance Monitoring

### DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<b>controller nid</b> <i>1/NID_ID</i>  <b>Example:</b> Switch(config)# controller nid 1/1	Enters the controller configuration mode.
Step 3	<b>OperationsMepPortType</b>  <b>Example:</b> Switch(config-controller)# <b>OperationsMepPortType</b>	Enters performance monitoring provisioning (PM) mode.
Step 4	<b>OperationsMepPortType</b> { <b>clearMepStats</b>   <b>default</b>   <b>exit</b>   <b>no</b>   <b>setAis</b>   <b>setDM</b>   <b>setLM</b>   <b>setLck</b>   <b>setLinkTrace</b>   <b>setLoopBack</b>   <b>setTst</b>   <b>showAis</b>   <b>showDM</b>   <b>showLM</b>   <b>showLck</b>   <b>showLinkTrace</b>   <b>showLoopBack</b>   <b>showTst</b>   <b>updateDM</b>   <b>updateTst</b> }  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# ? OperationsMepPortType sub-mode commands: clearMepStats Clear mep statistics request default Set a command to its defaults exit Exit from OperationsMepPortType sub configuration mode  no Negate a command or set its defaults setAis Enable or Disable AIS request setDM Enable or Disable delay measurement request setLM Enable or Disable loss measurement request setLck Enable or Disable lock signal request setLinkTrace Enable or Disable linktrace request setLoopBack Enable/Disable loopback setTst Enable or Disable test signal request showAis Show AIS configuration request showDM Show delay measurement statistics request showLM Show LM statistics request showLck Show lock signal current configuration request showLinkTrace Show linktrace state and current configuration request  showLoopBack Show loopback state and current configuration request  showTst Show test signal statistics and current configuration request updateDM Update DM parameters request updateTst Update Tst signal request	Displays the supported configurations for performance monitoring.
Step 5	<b>exit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# <b>exit</b>	Exits the performance monitoring provisioning mode.

### Configuration Example

The following example shows the supported PM configuration:

```
Switch(config-controller-OperationsMepPortType)# ?
OperationsMepPortType sub-mode commands:
  clearMepStats Clear mep statistics request
  default       Set a command to its defaults
  exit          Exit from OperationsMepPortType sub configuration mode
```

no	Negate a command or set its defaults
setAis	Enable or Disable AIS request
setDM	Enable or Disable delay measurement request
setLM	Enable or Disable loss measurement request
setLck	Enable or Disable lock signal request
setLinkTrace	Enable or Disable linktrace request
setLoopBack	Enable/Disable loopback
setTst	Enable or Disable test signal request
showAis	Show AIS configuration request
showDM	Show delay measurement statistics request
showLM	Show LM statistics request
showLck	Show lock signal current configuration request
showLinkTrace	Show linktrace state and current configuration request
showLoopBack	Show loopback state and current configuration request
showTst	Show test signal statistics and current configuration request
updateDM	Update DM parameters request
updateTst	Update Tst signal request

## Configuring Performance Monitoring with Default Configuration

You can set the default performance monitoring configurations on the controller.

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>default {clearMepStats   setAis   setDM   setLM   setLck   setLinkTrace   setLoopBack   setTst   showAis   showDM   showLM   showLck   showLinkTrace   showLoopBack   showTst   updateDM   updateTst   exit}</pre> <p><b>Example:</b> Switch(config-controller-OperationsMepPortType)# default ?</p> <pre>clearMepStats Clear mep statistics request exit Exit from OpearationsMepPortType sub configuration mode setAis Enable or Disable AIS request setDM Enable or Disable delay measurement request setLM Enable or Disable loss measurement request setLck Enable or Disable lock signal request  setLinkTrace Enable or Disable linktrace request setLoopBack Enable/Disable loopback setTst Enable or Disable test signal request  showAis Show AIS configuration request showDM Show delay measurement statistics request showLM Show LM statistics request showLck Show lock signal current configuration request showLinkTrace Show linktrace state and current</pre>	<p>Sets the default configuration.</p> <ul style="list-style-type: none"> <li>• <b>clearMepStats</b>—Clears MEP statistics.</li> <li>• <b>setAis</b>—Enables or disables AIS.</li> <li>• <b>setDM</b>—Enables or disables delay measurement.</li> <li>• <b>setLM</b>—Enables or disables loss measurement.</li> <li>• <b>setLck</b>—Enables or disables lock signals</li> <li>• <b>setLinkTrace</b>—Enables or disables link traces.</li> <li>• <b>setLoopBack</b>—Enables or disables loopback</li> <li>• <b>setTst</b>—Enables or disables the test signal.</li> <li>• <b>showAis</b>—Displays AIS configuration request.</li> <li>• <b>showDM</b>—Displays delay measurement statistics.</li> <li>• <b>showLM</b>—Displays loss measurement statistics.</li> <li>• <b>showLck</b>—Displays current configured loss signals.</li> <li>• <b>showLinkTrace</b>—Displays current configured link trace state.</li> </ul>

	Command or Action	Purpose
	<pre> configuration request showLoopBack Show loopback state and current configuration request showTst Show test signal statistics and current configuration request updateDM Update DM parameters request updateTst Update Tst signal request </pre>	<ul style="list-style-type: none"> <li>• <b>showLoopBack</b>—Displays current configured loopback state.</li> <li>• <b>showTst</b>—Displays current configured test signals statistics.</li> <li>• <b>updateDM</b>—Updates the delay measurement parameters.</li> <li>• <b>updateTst</b>—Updates test signal parameters.</li> <li>• <b>exit</b>—Exits from OperationsMepPortType configuration mode.</li> </ul>
<b>Step 2</b>	<p><b>exit</b></p> <p><b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>exit</b></p>	Exits the performance monitoring provisioning mode.

## Configuring Alarm Information Signal (AIS) on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<pre> setAis {commit   flush   aisConfig   review} </pre> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType) # setAis ?   aisConfig Enable or Disable AIS request   commit    commit setAis   flush     flush all setAis commands from qu   review    review setAis commands </pre>	<p>Configures alarm information signal (AIS).</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>aisConfig</b>—Enables or disables the AIS configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
<b>Step 2</b>	<pre> setAis aisConfig {aisaction {enable {frameRate {fr1m   fr1s}   disable}   protect {enable   disable} }   mepInstanceinstance_no} </pre>	<p>Configures AIS.</p> <ul style="list-style-type: none"> <li>• <b>aisaction</b>—Enables or disables AIS on the controller.</li> </ul>

	Command or Action	Purpose
	<p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setAis aisConfig aisaction enable frameRate fr1m Switch(config-controller-OperationsMepPortType) # setAis aisConfig aisaction enable frameRate fr1s Switch(config-controller-OperationsMepPortType) # setAis aisConfig aisaction enable protect enable Switch(config-controller-OperationsMepPortType) # setAis aisConfig mepInstance 20</pre>	<ul style="list-style-type: none"> <li>• <b>enable</b>—Enables lock signal configuration.</li> <li>• <b>disable</b>—Disables lock signal configuration.</li> <li>• <b>frameRate</b>—Indicates the frame rate.</li> <li>• <b>fr1m</b>—Indicates the frame rate is 1 f/m.</li> <li>• <b>fr1s</b>—Indicates the frame rate is 1 f/s.</li> <li>• <b>mepinstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<p><b>setAis review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setAis review</pre> <p>Commands in queue:</p> <pre>setAis aisConfig aisAction enable protect enable setAis aisConfig aisAction enable frameRate fr1s setAis aisConfig aisAction enable frameRate fr1m setAis aisConfig mepInstance 20</pre>	Displays the AIS configuration on the controller.
<b>Step 4</b>	<p><b>setAiscommit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setAis commit</pre>	Sends the AIS configuration to the NID.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # exit</pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to configure AIS on the controller:

```
Switch(config-controller-OperationsMepPortType) # setAis aisConfig aisaction enable frameRate
fr1m
Switch(config-controller-OperationsMepPortType) # setAis aisConfig aisaction enable frameRate
fr1s
Switch(config-controller-OperationsMepPortType) # setAis aisConfig aisaction enable protect
enable
Switch(config-controller-OperationsMepPortType) # setAis review
Switch(config-controller-OperationsMepPortType) # setAis aisConfig mepInstance 20

Commands in queue:
setAis aisConfig aisAction enable protect enable
setAis aisConfig aisAction enable frameRate fr1s
setAis aisConfig aisAction enable frameRate fr1m
setAis aisConfig mepInstance 20
```

```

Commands in queue:
    Commands in queue:
    setAis aisConfig mepInstance 2

Switch(config-controller-OperationsMepPortType)# setAis commit
SetAis_Output.mepResponse = 34537474

SetAis Commit Success!!!
Switch(config-controller-OperationsMepPortType)#end

```

## Configuring Delay Measurement (DM) on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><b>setDM</b> {commit   flush   dmConfig   review}</p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType)# setDM ? commit      commit setDM dmConfig    Enable or Disable delay measurement request  flush       flush all setDM commands from queue review      review setDM commands </pre>	<p>Configures delay measurement.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>dmConfig</b>—Enables or disables the delay measurement configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
<b>Step 2</b>	<p><b>setDM dmConfig</b> {dmaction {enable {calculation {flow   rdtrp}   cast {mutli   uni mepId mep_Id}   interval interval_no   lastN delay_calc   mode {oneway   twoway}   priority priority_no }   disable }   mepInstance instance_no }</p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable calculation flow Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable calculation rdtrp Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable cast multi Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable cast uni mepId 23 Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable interval 20 Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable lastN 200 Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable mode oneway Switch(config-controller-OperationsMepPortType)# setDM dmConfig dmaction enable mode twoway </pre>	<p>Sets DM parameters</p> <ul style="list-style-type: none"> <li>• <b>dmaction</b>—Enables or disables DM on the controller.</li> <li>• <b>enable</b>—Enables delay measurement configuration.</li> <li>• <b>disable</b>—Disables delay measurement configuration.</li> <li>• <b>calculation</b>—Calculates delay.</li> <li>• <b>flow</b>—Two-way delay is calculated as round trip symmetrical flow. Far end resistance time is subtracted.</li> <li>• <b>rdtrp</b>—Two-way delay is calculated as round trip delay. Far end resistance time is <i>not</i> subtracted.</li> <li>• <b>cast</b>—Specifies transmission mode.</li> <li>• <b>multi</b>—Specifies OAM protocol data units (PDU) transmission with multicast MAC.</li> </ul>



	Command or Action	Purpose
	<pre>Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable priority 3 Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable calculation rdtrp Switch(config-controller-OperationsMepPortType) # setDM dmConfig mepInstance 1</pre>	<ul style="list-style-type: none"> <li>• <b>uni</b>—Specifies OAM protocol data units (PDU) transmission with unicast MAC. The MAC is procured from the peer MEP MAC database.</li> <li>• <b>mepId</b> <i>mep_id</i>—Specifies Peer MEP ID for unicast MAC.</li> <li>• <b>interval</b> <i>interval_no</i>—Specifies the interval time between the PDU transmission in ms. The valid range is from 0 to 65535. The minimum value is 10ms.</li> <li>• <b>lastN</b>—Specifies the latest N delays for calculation. The valid range is from 10 to 2000.</li> <li>• <b>mode</b>—Specifies the mode of delay measurement.</li> <li>• <b>oneway</b>—Specifies mode on IDM PDU measurement.</li> <li>• <b>twoway</b>—Specifies mode on DMM or DMR PDU.</li> <li>• <b>priority</b> <i>priority_no</i>—Priority in case of tagged OAM. In the EVC domain this is the COS-ID. The valid range is from 0 to 7.</li> <li>• <b>mepinstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<p><b>Step 3</b></p>	<p><b>setDM review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setDM review</pre> <p>Commands in queue:</p> <pre>setDM dmConfig dmAction enable mode oneWay setDM dmConfig dmAction enable interval 10 setDM dmConfig dmAction enable lastN 20 setDM dmConfig dmAction enable calculation flow setDM dmConfig dmAction enable calculation rdtrp setDM dmConfig dmAction enable priority 2 setDM dmConfig dmAction enable calculation flow setDM dmConfig dmAction enable calculation rdtrp setDM dmConfig dmAction enable cast multi setDM dmConfig dmAction enable cast uni mepId 23 setDM dmConfig dmAction enable interval 20 setDM dmConfig dmAction enable priority 3 setDM dmConfig mepInstance 2</pre>	<p>Displays the DM configuration on the controller.</p>

	Command or Action	Purpose
Step 4	<b>setDMcommit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>setDM commit</b>  SetDM Commit Success!!!	Sends the DM configuration to the NID.
Step 5	<b>exit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>exit</b>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to configure DM on the controller:

```
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable calculation
flow
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable calculation
rdtrp
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable cast multi
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable cast uni
mepId 23
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable interval
20
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable lastN 200
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable mode oneway
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable mode twoway
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable priority 3
Switch(config-controller-OperationsMepPortType) # setDM dmConfig dmaction enable calculation
rdtrp
Switch(config-controller-OperationsMepPortType) # setDM dmConfig mepInstance 1
Switch(config-controller-OperationsMepPortType) # setDM review
```

Commands in queue:

```
setDM dmConfig dmAction enable mode oneWay
setDM dmConfig dmAction enable interval 10
setDM dmConfig dmAction enable lastN 20
setDM dmConfig dmAction enable calculation flow
setDM dmConfig dmAction enable calculation rdtrp
setDM dmConfig dmAction enable priority 2
setDM dmConfig dmAction enable calculation flow
setDM dmConfig dmAction enable calculation rdtrp
setDM dmConfig dmAction enable cast multi
setDM dmConfig dmAction enable cast uni mepId 23
setDM dmConfig dmAction enable interval 20
setDM dmConfig dmAction enable priority 3
setDM dmConfig mepInstance 2
```

```
Switch(config-controller-OperationsMepPortType) # setDM commit
DM.dmConfig.mepInstance = 119
DM.dmConfig.dmAction.t = 1
DM.dmConfig.dmAction.u.enable.priority = 4
DM.dmConfig.dmAction.u.enable.cast.t = 2
DM.dmConfig.dmAction.u.enable.cast.u.multi = 'any <b z="1">test</b> element'
DM.dmConfig.dmAction.u.enable.mode.t = 2
DM.dmConfig.dmAction.u.enable.mode.u.twoWay = 'any <b z="1">test</b> element'
DM.dmConfig.dmAction.u.enable.calculation.t = 2
```

```

DM.dmConfig.dmAction.u.enable.calculation.u.flow = 'any <b z="1">test</b> elemen
t'
DM.dmConfig.dmAction.u.enable.interval = 42689
DM.dmConfig.dmAction.u.enable.lastN = 1573
DM.dmConfig.mepInstance = 119
DM.dmConfig.dmAction.t = 1
DM.dmConfig.dmAction.u.enable.priority = 0
DM.dmConfig.dmAction.u.enable.cast.t = 1
DM.dmConfig.dmAction.u.enable.cast.u.uni.mepId = 23
DM.dmConfig.dmAction.u.enable.mode.t = 1
DM.dmConfig.dmAction.u.enable.mode.u.oneWay = 'one-way'
DM.dmConfig.dmAction.u.enable.calculation.t = 1
DM.dmConfig.dmAction.u.enable.calculation.u.rdtrp = 'rdtrp'
DM.dmConfig.dmAction.u.enable.interval = 42689
DM.dmConfig.dmAction.u.enable.lastN = 1573
SetDM_Output.mepResponse = 0

Switch(config-controller-OperationsMepPortType) #end
    
```

## Configuring Loss Measurement (LM) on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>setLM {commit   flush   lmConfig   review}</b></p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType) # setLM ?   commit      commit setLM   lmConfig    Enable or Disable loss measurement   request   flush       flush all setLM commands from queue   review      review setLM commands                     </pre>	<p>Configures loss measurement (LM).</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>lmConfig</b>—Enables or disables the loss measurement configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>setLM LmConfig { lmacction {enable {cast {multi   uni}   flr frame_interval   frameRate {fr10s   fr1m   fr1s   fr6h   fr6m}   mode {dual   single}   priority priority_no }   disable }   mepInstance instance_no }</b></p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmacction enable cast multi Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmacction enable cast uni Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmacction enable flr Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmacction frameRate fr10s Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmacction enable mode dual                     </pre>	<p>Sets LM parameters.</p> <ul style="list-style-type: none"> <li>• <b>lmacction</b>—Enables or disables LM on the controller.</li> <li>• <b>enable</b>—Enables loss measurement configuration.</li> <li>• <b>disable</b>—Disables loss measurement configuration.</li> <li>• <b>cast</b>—Specifies transmission mode.</li> <li>• <b>multi</b>—Specifies OAM protocol data units (PDU) transmission with multicast MAC.</li> <li>• <b>uni</b>—Specifies OAM protocol data units (PDU) transmission with unicast MAC. The MAC is procured from the peer MEP MAC database.</li> </ul>

	Command or Action	Purpose
	<pre>Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction enable priority 4 Switch(config-controller-OperationsMepPortType) # setLM lmConfig mepInstance 1</pre>	<ul style="list-style-type: none"> <li>• <b>flr</b> <i>frame_interval</i>—Specifies the frame loss ratio interval time. The valid range is from 0 to 99.</li> <li>• <b>frameRate</b>—Specifies the LM frame rate. <ul style="list-style-type: none"> <li>◦ <b>fr10s</b> —Specifies the frame rate as 10 f/s.</li> <li>◦ <b>fr1m</b> —Specifies the frame rate as 1 f/min.</li> <li>◦ <b>fr1s</b> —Specifies the frame rate as 1 f/s.</li> <li>◦ <b>fr6h</b> —Specifies the frame rate as 6 f/hour.</li> <li>◦ <b>fr6m</b> —Specifies the frame rate as 6 f/min.</li> </ul> </li> <li>• <b>mode</b>—Specifies the mode of delay measurement.</li> <li>• <b>dual</b>—Specifies dual LM mode on CCM PDU.</li> <li>• <b>single</b>—Specifies single LM mode on LMM or LMR PDU.</li> <li>• <b>priority</b> <i>priority_no</i>—Priority in case of tagged OAM. In the EVC domain this is the COS-ID. The valid range is from 0 to 7.</li> <li>• <b>mepinstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<pre>setLM review</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setLM review</pre> <p>Commands in queue:</p> <pre>setLM lmConfig lmAction enable cast multi setLM lmConfig lmAction enable cast uni setLM lmConfig lmAction enable flr 2 setLM lmConfig lmAction enable frameRate fr10s setLM lmConfig lmAction enable mode dual setLM lmConfig lmAction enable priority 4</pre>	Displays the LM configuration on the controller.
<b>Step 4</b>	<pre>setLM commit</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setLM commit SetLM Commit Success!!!</pre>	Sends the LM configuration to the NID.
<b>Step 5</b>	<pre>exit</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # exit</pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to configure LM on the controller:

```
Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction enable cast multi
Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction enable cast uni
Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction enable flr
Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction frameRate fr10s
Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction enable mode dual
Switch(config-controller-OperationsMepPortType) # setLM lmConfig lmaction enable priority 4
Switch(config-controller-OperationsMepPortType) # setLM lmConfig mepInstance 1
Switch(config-controller-OperationsMepPortType) # setLM review
```

Commands in queue:

```
setLM lmConfig lmAction enable cast multi
setLM lmConfig lmAction enable cast uni
setLM lmConfig lmAction enable flr 2
setLM lmConfig lmAction enable frameRate fr10s
setLM lmConfig lmAction enable mode dual
setLM lmConfig lmAction enable priority 4
```

```
Switch(config-controller-OperationsMepPortType) # setLM commit
SetLM_Output.mepResponse = 0 SetLM-Output.mepResponse = 0
SetLM Commit Success!!!
```

```
Switch(config-controller-OperationsMepPortType) #end
```

## Configuring Lock Signal on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>setLck</b> {commit   flush   lckConfig   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setLck ? commit      commit setLck flush      flush all setLck commands from queue lckConfig  Enable or Disable lock signal request review     review setLck commands</pre>	<p>Configures lock signal.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>lckConfig</b>—Enables or disables the lock signal configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>setLck lckConfig</b> {lckaction {enable frameRate {fr1m   fr1s}   disable}   mepInstance <i>instance_no</i>}</p>	<p>Sets lock signal parameters.</p> <ul style="list-style-type: none"> <li>• <b>lckaction</b>—Enables or disables lock signal on the controller.</li> </ul>

	Command or Action	Purpose
	<p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# setlck lckConfig lckaction enable frameRate fr1m Switch(config-controller-OperationsMepPortType)# setlck lckConfig lckaction mepInstance 1</pre>	<ul style="list-style-type: none"> <li>• <b>enable</b>—Enables lock signal configuration.</li> <li>• <b>frameRate</b>—Configures the frame rate. <ul style="list-style-type: none"> <li>◦ <b>fr1m</b>—Specifies frame rate as 1 f/m.</li> <li>◦ <b>fr1s</b>—Specifies frame rate as 1 f/s.</li> </ul> </li> <li>• <b>disable</b>—Disables lock signal configuration.</li> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<p><b>setlck review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# setlck review</pre> <p>Commands in queue:</p> <pre>setLck lckConfig lckAction enable frameRate fr1m setLck lckConfig mepInstance 1</pre>	Displays the lock signal configuration on the controller.
<b>Step 4</b>	<p><b>setlckcommit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# setlck commit</pre>	Sends the lock signal configuration to the NID.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# exit</pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to configure lock signal on the controller:

```
Switch(config-controller-OperationsMepPortType)# setlck lckConfig lckaction enable frameRate
fr1m
Switch(config-controller-OperationsMepPortType)# setlck lckConfig lckaction mepInstance 1
Switch(config-controller-OperationsMepPortType)# setlck review

Commands in queue:
setLck lckConfig lckAction enable frameRate fr1m
setLck lckConfig mepInstance 1

Switch(config-controller-OperationsMepPortType)# setlck commit
SetLck_Output.mepResponse = 0
SetLck Commit Success!!!
Switch(config-controller-OperationsMepPortType)#end
```

# Configuring LoopBack on the Controller

## Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>setLoopBack</b> {commit   flush   loopBackConfig   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setLoopBack ?   commit          commit setLoopBack   flush           flush all setLoopBack commands from   queue   loopBackConfig Enable/Disable loopback   review          review setLoopBack commands</pre>	<p>Configures loopback.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>loopBackConfig</b>—Enables or disables the loopback configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>setLoopBack loopBackConfig</b> {lbAction {enable {cast {multi   uni {destination {macAddress <i>target_MAC</i>   mepid <i>mep_id</i>}}}}   count <i>count_no</i>   dei {enable   disable}   interval <i>interval</i>   priority <i>priority_no</i>   size <i>bytes</i>   disable}   mepInstance <i>instance_no</i>}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable cast multi Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable cast uni destination macAddress mac1 Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable cast unidestination mepId 3 Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable count 345 Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable dei enable Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable interval 20 Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable priority 7 Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable size 1400 Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig mepInstance 125</pre>	<p>Sets loopback parameters.</p> <ul style="list-style-type: none"> <li>• <b>lbaction</b>—Enables or disables loop back on the controller.</li> <li>• <b>enable</b>—Enables loop back trace configuration.</li> <li>• <b>cast</b>—Specifies the type of loop back configuration.</li> <li>• <b>multi</b>—Specifies OAM protocol data units (PDU) transmission with multicast MAC.</li> <li>• <b>uni</b>—Specifies OAM protocol data units (PDU) transmission with unicast MAC. The MAC is procured from the peer MEP MAC database.</li> <li>• <b>destination</b>—Specifies the target peer MEP.</li> <li>• <b>macAddress <i>target_MAC</i></b>—Specifies the MAC address for LT in MEP.</li> <li>• <b>mepId <i>mep_id</i></b>—Specifies Peer MEP ID. The valid range is from 0 to 8191.</li> <li>• <b>dei</b>—Specifies drop eligible indicator for tagged OAM.</li> <li>• <b>enable</b>—Enables drop eligible indicator configuration.</li> <li>• <b>disable</b>—Disables drop eligible indicator configuration.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>count</b> <i>count_no</i>—Specifies the number of loop back PDU sent in a single loop test .</li> <li>• <b>interval</b> <i>interval_no</i>—Specifies the interval time between the PDU transmission in ms. The valid range is from 0 to 65535. The minimum value is 10ms.</li> <li>• <b>priority</b> <i>priority_no</i>—Specifies the priority for tagged OAM. In EVC domain, it indicates the COS-ID. The valid range is from 0 to 7.</li> <li>• <b>size</b> <i>frames</i>—Specifies the number of bytes. The valid range is from 1 to 1400.</li> <li>• <b>disable</b>—Disables loop back configuration.</li> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<b>setloopback review</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# <b>setloopback review</b> Commands in queue: setLoopBack loopBackConfig lbAction enable cast multi setLoopBack loopBackConfig lbAction enable cast uni destination macAddress mac1 setLoopBack loopBackConfig lbAction enable cast uni destination mepId 3 setLoopBack loopBackConfig lbAction enable count 345 setLoopBack loopBackConfig lbAction enable dei enable setLoopBack loopBackConfig lbAction enable interval 20 setLoopBack loopBackConfig lbAction enable priority 7 setLoopBack loopBackConfig lbAction enable size 1400 setLoopBack loopBackConfig mepInstance 125	Displays the loop back configuration on the controller.
<b>Step 4</b>	<b>setlckcommit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# <b>setloopback commit</b>	Sends the loop back configuration to the NID.
<b>Step 5</b>	<b>exit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# <b>exit</b>	Exits the performance monitoring provisioning mode.



### Configuration Example

The example shows how to configure loop back on the controller:

```
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
cast multi
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
cast uni destination macAddress mac1
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
cast unidestination mepId 3
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
count 345
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
dei enable
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
interval 20
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
priority 7
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig lbaction enable
size 1400
Switch(config-controller-OperationsMepPortType) # setLoopBack loopbackConfig mepInstance 125
Switch(config-controller-OperationsMepPortType) # setsetloopback review

Commands in queue:
    setLoopBack loopBackConfig lbAction enable cast multi
    setLoopBack loopBackConfig lbAction enable cast uni destination macAddre
ss mac1
    setLoopBack loopBackConfig lbAction enable cast uni destination mepId 3
    setLoopBack loopBackConfig lbAction enable cast uni destination mepId 3
    setLoopBack loopBackConfig lbAction enable count 345
    setLoopBack loopBackConfig lbAction enable dei enable
    setLoopBack loopBackConfig lbAction enable interval 20
    setLoopBack loopBackConfig lbAction enable priority 7
    setLoopBack loopBackConfig lbAction enable size 1400

Switch(config-controller-OperationsMepPortType) # setsetloopback commit
SetLoopBack_Output.mepResponse = 34275330

SetLoopBack Commit Success!!!
Switch(config-controller-OperationsMepPortType) #end
```

## Configuring Link Trace on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring, on page 3](#).

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>setLinkTrace</b> {commit   flush   linkTrace   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setLinkTrace ? commit      commit setLinkTrace flush      flush all setLinkTrace commands from</pre>	<p>Configures link trace.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>linkTrace</b>—Enables or disables the link trace configuration.</li> </ul>

	Command or Action	Purpose
	<pre>queue linkTrace Enable or Disable linktrace request review review setLinkTrace commands</pre>	<ul style="list-style-type: none"> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
<b>Step 2</b>	<p><b>setLinkTrace linkTrace {lTAction {enable {destination {macAddress target_MAC   mepId mep_id}   priority priority_no   ttl ttl_time disable}}   mepInstance instance_no}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable destination macAddress mac1 Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable destination mepId 3 Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable priority 2 Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable ttl 3</pre>	<p>Sets link trace parameters.</p> <ul style="list-style-type: none"> <li>• <b>lTaction</b>—Enables or disables link trace on the controller.</li> <li>• <b>enable</b>—Enables link trace configuration.</li> <li>• <b>destination</b>—Specifies the target peer MEP. <ul style="list-style-type: none"> <li>◦ <b>macAddress target_MAC</b>—Specifies the link trace MAC address for LT in MEP.</li> <li>◦ <b>mepId mep_id</b>—Specifies Peer MEP ID for link trace. The valid range is from 0 to 8191.</li> </ul> </li> <li>• <b>priority priority_no</b>—Specifies the priority for tagged OAM. In EVC domain, it indicates the COS-ID. The valid range is from 0 to 7.</li> <li>• <b>ttl ttl_time</b>—Specifies the time to live. The valid range is from 1 to 999.</li> <li>• <b>disable</b>—Disables link trace signal configuration.</li> <li>• <b>mepInstance instance_no</b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<p><b>setLinkTrace review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setlinkTrace review</pre>	<p>Displays the link trace configuration on the controller.</p>
<b>Step 4</b>	<p><b>setlckcommit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # setlinkTrace commit</pre>	<p>Sends the link trace configuration to the NID.</p>
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # exit</pre>	<p>Exits the performance monitoring provisioning mode.</p>

### Configuration Example

The example shows how to configure link trace on the controller:

```
Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable destination macAddress mac1
Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable destination mepId 3
Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable priority 2
Switch(config-controller-OperationsMepPortType) # setlinkTrace linkTrace ltkaction enable ttl 3
Switch(config-controller-OperationsMepPortType) # setlinkTrace review

Switch(config-controller-OperationsMepPortType) # setlinkTrace commit
SetLinkTrace_Output.mepResponse = 34340866

SetLinkTrace Commit Success!!!
Switch(config-controller-OperationsMepPortType) #end
```

## Configuring Test Signal on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring, on page 3](#).

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>setTst {commit   flush   tstConfig   review}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>setTst ?</b> commit      commit setTst flush       flush all setTst commands from queue review      review setTst commands tstConfig   Enable or Disable test signal request</pre>	<p>Configures test signal.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>tstConfig</b>—Enables or disables the test signal configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>setTst tstConfig {dei {enable   disable}   mepId mep_Id   mepInstance mep_instance   pattern {allOne   allZero   oneZero}   priority priority_no   rate bit_rate   sequence {enable   disable}   size frames}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>setTst tstConfig dei enable</b> Switch(config-controller-OperationsMepPortType) # <b>setTst tstConfig mepid 2</b> Switch(config-controller-OperationsMepPortType) # <b>setTst tstConfig mepinstance 2</b> Switch(config-controller-OperationsMepPortType) #</pre>	<p>Sets test signal parameters.</p> <ul style="list-style-type: none"> <li>• <b>dei</b>—Specifies drop eligible indicator for tagged OAM.</li> <li>• <b>enable</b>—Enables drop eligible indicator configuration.</li> <li>• <b>disable</b>—Disables drop eligible indicator configuration.</li> <li>• <b>mepId mep_instance</b>—Specifies Peer MEP ID. The valid range is from 0 to 8191.</li> <li>• <b>mepInstance instance_no</b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>

	Command or Action	Purpose
	<pre> setTst tstConfig pattern allOne Switch(config-controller-OperationsMepPortType) # setTst tstConfig pattern allZero Switch(config-controller-OperationsMepPortType) # setTst tstConfig sequence enable Switch(config-controller-OperationsMepPortType) # setTst tstConfig rate 400 Switch(config-controller-OperationsMepPortType) # setTst tstConfig size 45 </pre>	<ul style="list-style-type: none"> <li>• <b>priority</b> <i>priority_no</i>—Specifies the priority for tagged OAM. In EVC domain, it indicates the COS-ID. The valid range is from 0 to 7.</li> <li>• <b>pattern</b>—Specifies the sequence number is test PDU.</li> <li>• <b>allOne</b>—Specifies the test pattern to all one.</li> <li>• <b>allZero</b>—Specifies the test pattern to all zero.</li> <li>• <b>oneZero</b>—Specifies the test pattern to one zero.</li> <li>• <b>rate</b> <i>bit_rate</i>—Specifies the test frame transmission rate in Megabits per second. The valid range is from 1 to 1518.</li> <li>• <b>sequence</b>—Enables or disables the sequence in test PDU.</li> <li>• <b>size</b> <i>frames</i>—Specifies the test pattern size in frames. The valid range is from 1 to 1518.</li> </ul>
<b>Step 3</b>	<p><b>setTst review</b></p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType) # setTst review Commands in queue:   setTst tstConfig dei enable   setTst tstConfig mepId 2   setTst tstConfig mepInstance 2   setTst tstConfig pattern allOne   setTst tstConfig pattern allZero   setTst tstConfig priority 5   setTst tstConfig pattern allZero   setTst tstConfig rate 400   setTst tstConfig size 45 </pre>	Displays the test signal configuration on the controller.
<b>Step 4</b>	<p><b>setTstcommit</b></p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType) # setTst commit </pre>	Sends the link trace configuration to the NID.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType) # exit </pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to configure test signal on the controller:

```
Switch(config-controller-OperationsMepPortType) # setTst tstConfig dei enable
Switch(config-controller-OperationsMepPortType) # setTst tstConfig mepid 2
Switch(config-controller-OperationsMepPortType) # setTst tstConfig mepinstance 2
Switch(config-controller-OperationsMepPortType) # setTst tstConfig pattern allOne
Switch(config-controller-OperationsMepPortType) # setTst tstConfig pattern allZero
Switch(config-controller-OperationsMepPortType) # setTst tstConfig rate 400
Switch(config-controller-OperationsMepPortType) # setTst tstConfig sequence enable
Switch(config-controller-OperationsMepPortType) # setTst tstConfig size 45

Switch(config-controller-OperationsMepPortType) # setTst review
Commands in queue:
    setTst tstConfig dei enable
    setTst tstConfig mepId 2
    setTst tstConfig mepInstance 2
    setTst tstConfig pattern allOne
    setTst tstConfig pattern allZero
    setTst tstConfig priority 5
    setTst tstConfig pattern allZero
    setTst tstConfig rate 400
    setTst tstConfig size 45

Switch(config-controller-OperationsMepPortType) # setTst commit
SetTst_Output.mepResponse = 34471938
SetTst Commit Success!!!
Switch(config-controller-OperationsMepPortType) #end
```

## Viewing Alarm Information Signal (AIS) on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>showAis</b> {commit   flush   mepRequest   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showAis ? commit      commit showAis flush      flush all showAis commands from queue mepRequest  Show AIS configuration request review      review showAis commands</pre>	<p>Displays alarm information signal configuration.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the alarm configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>showAis mepRequest</b> {all   mepInstance <i>instance_no</i>}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) #</pre>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays AIS configuration for all MEPs on the controller.</li> <li>• <b>mepInstance <i>instance_no</i></b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>

	Command or Action	Purpose
	<pre>showAis mepRequest all Switch(config-controller-OperationsMepPortType) # showAis mepRequest mepInstance 120</pre>	
<b>Step 3</b>	<pre>showAis review</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showAis review</pre> <p>Commands in queue:</p> <pre>showAis mepRequest all showAis mepRequest mepInstance 120</pre>	Displays the configuration on the controller.
<b>Step 4</b>	<pre>showAis commit</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showAis commit</pre>	Sends the configuration to the NID.
<b>Step 5</b>	<pre>exit</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # exit</pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to display the AIS on the controller:

```
Switch(config-controller-OperationsMepPortType) # showAis mepRequest all
Switch(config-controller-OperationsMepPortType) # showAis mepRequest mepInstance 120
Switch(config-controller-OperationsMepPortType) # showAis review

Commands in queue:
showAis mepRequest all
showAis mepRequest mepInstance 120

Switch(config-controller-OperationsMepPortType) # showAis commit
Ais_Output.aisInfo.mepInst[0].config.mepInstance = 100
ShowAis_Output.aisInfo.mepInst[0].config.frameRate.t = 2
ShowAis_Output.aisInfo.mepInst[0].config.frameRate.u.frls = 'frls'
ShowAis_Output.aisInfo.mepInst[0].config.protect.t = 1
ShowAis_Output.aisInfo.mepInst[0].config.protect.u.enable = 'enable'

showAis Commit Success!!!
Switch(config-controller-OperationsMepPortType) # end
```

## Viewing Delay Measurement (DM) Statistics on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>showDM</b> {<b>commit</b>   <b>flush</b>   <b>mepRequest</b>   <b>review</b>}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showDM</b> ? commit          commit showDM flush           flush all showDM commands from queue mepRequest     Show delay measurement statistics request  review         review showDM commands</pre>	<p>Displays delay measurement (DM).</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>showDM mepRequest</b> {<b>all</b>   <b>mepInstance</b> <i>instance_no</i>}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showDM</b> <b>mepRequest all</b> Switch(config-controller-OperationsMepPortType) # <b>showDM</b> <b>mepRequest mepInstance 100</b></pre>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays DM configuration for all MEPs on the controller.</li> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
Step 3	<p><b>showDM review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showDM</b> <b>review</b> Commands in queue: showDM mepRequest all showDM mepRequest all showDM mepRequest mepInstance 100</pre>	<p>Displays the configuration on the controller.</p>
Step 4	<p><b>showDM commit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showDM</b> <b>commit</b></pre>	<p>Sends the configuration to the NID.</p>
Step 5	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>exit</b></pre>	<p>Exits the performance monitoring provisioning mode.</p>

## Configuration Example

The example shows how to display the delay measurement statistics on the controller:

```
Switch(config-controller-OperationsMepPortType) # showDM mepRequest all
Switch(config-controller-OperationsMepPortType) # showDM mepRequest mepInstance 100
Switch(config-controller-OperationsMepPortType) # showDM review
Commands in queue:
showDM mepRequest all
showDM mepRequest all
showDM mepRequest mepInstance 100
```

```

Switch(config-controller-OperationsMepPortType)# showDM commit
ShowDM_Output.dmInfo.mepInst[0].config.mepInstance = 98
ShowDM_Output.dmInfo.mepInst[0].config.priority = 7
ShowDM_Output.dmInfo.mepInst[0].config.cast.t = 1
ShowDM_Output.dmInfo.mepInst[0].config.cast.u.uni.mepId = 106
ShowDM_Output.dmInfo.mepInst[0].config.mode.t = 2
ShowDM_Output.dmInfo.mepInst[0].config.mode.u.twoWay = 'two-way'
ShowDM_Output.dmInfo.mepInst[0].config.txMode.t = 1
ShowDM_Output.dmInfo.mepInst[0].config.txMode.u.standardize = 'standardize'
ShowDM_Output.dmInfo.mepInst[0].config.calculation.t = 1
ShowDM_Output.dmInfo.mepInst[0].config.calculation.u.rdtrp = 'rdtrp'
ShowDM_Output.dmInfo.mepInst[0].config.interval = 10
ShowDM_Output.dmInfo.mepInst[0].config.lastN = 10
ShowDM_Output.dmInfo.mepInst[0].config.unit.t = 2
ShowDM_Output.dmInfo.mepInst[0].config.unit.u.us = 'micro seconds'
ShowDM_Output.dmInfo.mepInst[0].config.synchronized.t = 2
ShowDM_Output.dmInfo.mepInst[0].config.synchronized.u.disable = 'Disable'
ShowDM_Output.dmInfo.mepInst[0].config.overflowReset.t = 1
ShowDM_Output.dmInfo.mepInst[0].config.overflowReset.u.keep = 'keep'
ShowDM_Output.dmInfo.mepInst[0].state.mepInstance = 106
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.tx = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.rxTimeout = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.rx = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.rxError = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.avgTotal = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.avgLastN = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.min = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.max = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.avgVariationTotal = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.avgVariationLastN = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.minVar = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.maxVar = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.F_to_N.overflow = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.tx = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.rxTimeout = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.rx = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.rxError = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.avgTotal = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.avgLastN = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.min = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.max = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.avgVariationTotal = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.avgVariationLastN = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.minVar = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.maxVar = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.oneWay.N_to_F.overflow = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.tx = 793
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.rxTimeout = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.rx = 793
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.rxError = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.avgTotal = 17
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.avgLastN = 17
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.min = 17
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.max = 18
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.avgVariationTotal = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.avgVariationLastN = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.minVar = 0
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.maxVar = 1
ShowDM_Output.dmInfo.mepInst[0].state.mode.twoWay.overflow = 0
  showDM Commit Success!!!

Switch(config-controller-OperationsMepPortType)# end

```



## Viewing Loss Measurement (LM) Statistics on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><b>showLM</b> {commit   flush   mepRequest   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# showLM ?   commit      commit showLM   flush       flush all showLM commands from queue   mepRequest  Show LM statistics request   review      review showLM commands</pre>	<p>Displays loss measurement configuration.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
<b>Step 2</b>	<p><b>showLM mepRequest</b> {all   mepInstance <i>instance_no</i>}</p> <p><b>Example:</b></p> <pre>Switch(config-controllerOperationsMepPortType)# showLM mepRequest all Switch(config-controller-OperationsMepPortType)# showLM mepRequest mepInstance 100</pre>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays LM statistics for all MEPs on the controller.</li> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<p><b>showLM review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# showLM review Commands in queue:   showLM mepRequest all   showLM mepRequest mepInstance 100</pre>	<p>Displays the configuration on the controller.</p>
<b>Step 4</b>	<p><b>showLM commit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# showLM commit</pre>	<p>Sends the configuration to the NID.</p>
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# exit</pre>	<p>Exits the performance monitoring provisioning mode.</p>

### Configuration Example

The example shows how to display the loss measurement statistics on the controller:

```
Switch(config-controller-OperationsMepPortType)# showLM mepRequest all
Switch(config-controller-OperationsMepPortType)# showLM mepRequest mepInstance 100
Switch(config-controller-OperationsMepPortType)# showLM review

CCommands in queue:
    showLM mepRequest all
    showLM mepRequest mepInstance 100

Switch(config-controller-OperationsMepPortType)# showLM commit
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.mepInstance = 98
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.priority = 7
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.cast.t = 2
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.cast.u.multi = 'multi'
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.mode.t = 2
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.mode.u.single = 'single'
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.frameRate.t = 3
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.frameRate.u.frls = 'frls'
ShowLM_Output.lossMeasurentInfo.mepInst[0].config.flr = 5
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.mepInstance = 98
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.tx = 137
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.rx = 137
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.nearCount = 0
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.farCount = 1105217
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.nearRatio = 0
ShowLM_Output.lossMeasurentInfo.mepInst[0].state.farRatio = 94
    showLM Commit Success!!!

Switch(config-controller-OperationsMepPortType)# exit
```

## Viewing Lock Signal on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>showlck</b> {commit   flush   mepRequest   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# showAis ?   commit      commit showLck   flush       flush all showLck commands from queue   mepRequest  Show lock signal current configuration   request   review      review showLck commands</pre>	<p>Displays lock signal information.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>showlck mepRequest</b> {all   mepInstance <i>instance_no</i>}</p>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays lock signal configuration for all MEPs on the controller.</li> </ul>

	Command or Action	Purpose
	<p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showlck mepRequest all Switch(config-controller-OperationsMepPortType) # showlck mepRequest mepInstance 20</pre>	<ul style="list-style-type: none"> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
<b>Step 3</b>	<p><b>showlck review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showlck review Commands in queue:   showLck mepRequest all   showLck mepRequest mepInstance 20</pre>	Displays the configuration on the controller.
<b>Step 4</b>	<p><b>setlck commit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showlck commit</pre>	Sends the configuration to the NID.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # exit</pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to display the lock signal on the controller:

```
Switch(config-controller-OperationsMepPortType) # showlck mepRequest all
Switch(config-controller-OperationsMepPortType) # showlck mepRequest mepInstance 20
Switch(config-controller-OperationsMepPortType) # showlck review

Commands in queue:
  showLck mepRequest all
  showLck mepRequest mepInstance 20

Switch(config-controller-OperationsMepPortType) # showlck commit
SetLck_Output.mepResponse = 0

SetLck Commit Success!!!
Switch(config-controller-OperationsMepPortType) # end
```

## Viewing Loopback State on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>showLoopBack</b> {<b>commit</b>   <b>flush</b>   <b>mepRequest</b>   <b>review</b>}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showLoopBack</b> ?   commit      commit showLoopBack   flush       flush all showLoopBack commands from   queue   mepRequest  Show loopback state and current   configuration request   review      review showLoopBack commands</pre>	<p>Display loopback information.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>showLoopBack mepRequest</b> {<b>all</b>   <b>mepInstance</b> <i>instance_no</i>}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showLoopBack mepRequest all</b> Switch(config-controller-OperationsMepPortType) # <b>showLoopBack mepRequest mepInstance 30</b></pre>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays loopback configuration for all MEPs on the controller.</li> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
Step 3	<p><b>showLoopBack review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showLoopBack review</b></pre> <p>Commands in queue:</p> <pre>showLoopBack mepRequest all showLoopBack mepRequest mepInstance 30</pre>	<p>Displays the configuration on the controller.</p>
Step 4	<p><b>showLoopBack commit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>showLoopBack commit</b></pre>	<p>Sends the configuration to the NID.</p>
Step 5	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>exit</b></pre>	<p>Exits the performance monitoring provisioning mode.</p>

## Configuration Example

The example shows how to display the loop back state on the controller:

```
Switch(config-controller-OperationsMepPortType) # showLoopBack mepRequest all
Switch(config-controller-OperationsMepPortType) # showLoopBack mepRequest mepInstance 30
Switch(config-controller-OperationsMepPortType) # showLoopBack review
```

```
Commands in queue:
  showLoopBack mepRequest all
  showLoopBack mepRequest mepInstance 30
```

```

Switch(config-controller-OperationsMepPortType)# showLoopBack commit
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.mepInstance = 100
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.dei.t = 2
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.dei.u.disable = 'DEI Disable'
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.priority = 7
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.cast.t = 2
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.cast.u.multi = 'MULTI'
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.count = 10
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.size = 70
ShowLoopBack_Output.loopbackInfo.mepInst[0].config.interval = 1
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.mepInstance = 32
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.transactionId = 11
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.txLBM.upper = 0
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.txLBM.lower = 10
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.reply[0].rcvMac = '00-3A-99-FD-47-2F'
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.reply[0].received.upper = 0
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.reply[0].received.lower = 10
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.reply[0].outOfOrder.upper = 0
ShowLoopBack_Output.loopbackInfo.mepInst[0].state.reply[0].outOfOrder.lower = 0

Switch(config-controller-OperationsMepPortType)# exit

```

## Viewing Link Trace State on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>showLinkTrace</b> {commit   flush   mepRequest   review}</p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType)# showLinkTrace ?   commit      commit showLinkTrace   flush       flush all showLinkTrace commands from               queue   mepRequest  Show linktrace state and current               configuration request   review      review showLinkTrace commands </pre>	<p>Displays link trace configuration.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>showLinkTrace mepRequest</b> {all   mepInstance <i>instance_no</i>}</p> <p><b>Example:</b></p> <pre> Switch(config-controller-OperationsMepPortType)# showLinkTrace mepRequest all Switch(config-controller-OperationsMepPortType)# showLinkTrace mepRequest mepInstance 40 </pre>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays link trace state configuration for all MEPs on the controller.</li> <li>• <b>mepInstance <i>instance_no</i></b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>

	Command or Action	Purpose
Step 3	<b>showLinkTrace review</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>showLinkTrace review</b>  Commands in queue: showLinkTrace mepRequest all showLinkTrace mepRequest mepInstance 120	Displays the configuration on the controller.
Step 4	<b>showLinkTrace commit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>showLinkTrace commit</b>	Sends the configuration to the NID.
Step 5	<b>exit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>exit</b>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to display the link trace state on the controller:

```
Switch(config-controller-OperationsMepPortType) # showLinkTrace mepRequest all
Switch(config-controller-OperationsMepPortType) # showLinkTrace mepRequest mepInstance 40
Switch(config-controller-OperationsMepPortType) # showLinkTrace review

Commands in queue:
  showLinkTrace mepRequest all
  showLinkTrace mepRequest mepInstance 40

Switch(config-controller-OperationsMepPortType) # showLinkTrace commit
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].config.mepInstance = 100
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].config.priority = 7
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].config.mepId = 101
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].config.macAddress = '00-00-00-00-00-00'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].config.ttl = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].transactionId = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].ttl = 0
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].mode.t = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].mode.u.MEP = 'MEP'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].direction.t = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].direction.u.DOWN = 'DOWN'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].forwarded.t = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].forwarded.u.NO = 'Not
forwarded'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].relay = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].lastMac = '00-3A-99-FD-4A-53'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[0].reply[0].nextMac = '00-3A-99-FD-47-2F'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].transactionId = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].ttl = 0
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].mode.t = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].mode.u.MEP = 'MEP'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].direction.t = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].direction.u.DOWN = 'DOWN'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].forwarded.t = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].forwarded.u.NO = 'Not
forwarded'
```

```
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].relay = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].lastMac = '00-3A-99-FD-4A-53'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[1].reply[0].nextMac = '00-3A-99-FD-47-2F'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].transactionId = 3
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].ttl = 0
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].mode.t = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].mode.u.MEP = 'MEP'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].direction.t = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].direction.u.DOWN = 'DOWN'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].forwarded.t = 2
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].forwarded.u.NO = 'Not
forwarded'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].relay = 1
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].lastMac = '00-3A-99-FD-4A-53'
ShowLinkTrace_Output.linkTraceInfo.mepInst[0].state[2].reply[0].nextMac = '00-3A-99-FD-47-2F'

showLinkTrace Commit Success!!!

Switch(config-controller-OperationsMepPortType) # end
```

## Viewing Test Signal Statistics on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>showTst {commit   flush   mepRequest   review}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showTst ?   commit      commit showTst   flush       flush all showTst commands from queue   mepRequest  Show test signal statistics and current                 configuration request   review      review showTst commands</pre>	<p>Displays test signal statistics.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>mepRequest</b>—Displays the configuration.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>showTst mepRequest {all   mepInstance instance_no}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showTst mepRequest all Switch(config-controller-OperationsMepPortType) # showTst mepRequest mepInstance 50</pre>	<ul style="list-style-type: none"> <li>• <b>all</b>—Displays test signal statistics configuration for all MEPs on the controller.</li> <li>• <b>mepInstance instance_no</b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> </ul>
Step 3	<p><b>showTst review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # showTst review</pre> <p>Commands in queue:</p>	<p>Displays the configuration on the controller.</p>

	Command or Action	Purpose
	<pre>showTst mepRequest all showTst mepRequest mepInstance 50</pre>	
<b>Step 4</b>	<p><b>showTstcommit</b></p> <p><b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>showTst commit</b></p>	Sends the configuration to the NID.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>exit</b></p>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to display the test signal statistics on the controller:

```
Switch(config-controller-OperationsMepPortType) # showTst mepRequest all
Switch(config-controller-OperationsMepPortType) # showTst mepRequest mepInstance 120
Switch(config-controller-OperationsMepPortType) # showTst review
```

```
Commands in queue:
  showTst mepRequest all
  showTst mepRequest mepInstance 50
```

```
Switch(config-controller-OperationsMepPortType) # showTst commit
ShowTst_Output.tstInfo.mepInst[0].config.mepInstance = 100
ShowTst_Output.tstInfo.mepInst[0].config.dei.t = 2
ShowTst_Output.tstInfo.mepInst[0].config.dei.u.disable = 'Disable'
ShowTst_Output.tstInfo.mepInst[0].config.priority = 7
ShowTst_Output.tstInfo.mepInst[0].config.mepId = 101
ShowTst_Output.tstInfo.mepInst[0].config.rate = 1000
ShowTst_Output.tstInfo.mepInst[0].config.size = 64
ShowTst_Output.tstInfo.mepInst[0].config.pattern.t = 1
ShowTst_Output.tstInfo.mepInst[0].config.pattern.u.allZero = 'all-zero'
ShowTst_Output.tstInfo.mepInst[0].config.sequence.t = 1
ShowTst_Output.tstInfo.mepInst[0].config.sequence.u.enable = 'Enable'
ShowTst_Output.tstInfo.mepInst[0].config.Tx.t = 1
ShowTst_Output.tstInfo.mepInst[0].config.Tx.u.enable = 'Enable'
ShowTst_Output.tstInfo.mepInst[0].config.Rx.t = 1
ShowTst_Output.tstInfo.mepInst[0].config.Rx.u.enable = 'Enable'
ShowTst_Output.tstInfo.mepInst[0].state.txFrameCount.upper = 0
ShowTst_Output.tstInfo.mepInst[0].state.txFrameCount.lower = 241803
ShowTst_Output.tstInfo.mepInst[0].state.rxFrameCount.upper = 0
ShowTst_Output.tstInfo.mepInst[0].state.rxFrameCount.lower = 0
ShowTst_Output.tstInfo.mepInst[0].state.rxRate = 0
ShowTst_Output.tstInfo.mepInst[0].state.testTime = 162
  showTst Commit Success!!!
```

```
Switch(config-controller-OperationsMepPortType) # end
```



# Updating Delay Measurement (DM) on the Controller

## Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>updateDM</b> {commit   flush   updateDmConfig   review}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# updateDM ? commit          commit updateDM flush           flush all updateDM commands from queue  review         review updateDM commands updateDmConfig Update DM parameters request</pre>	<p>Updates delay measurement (DM).</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>updateDmConfig</b>—Updates the delay measurement parameters.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>updateDM updateDmConfig</b> {mepInstance <i>instance_no</i>   update {overflowReset {keep   reset}   synchronized {enable   disable}   txMode {proprietary   standardize}   unit {ns   us}}</p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig mepInstance 100 Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update overflowReset keep Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update overflowReset reset Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update synchronized enable Switchconfig-controller-OperationsMepPortType)# updateDM updateDmConfig update overflowReset keep Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update txMode proprietary Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update txMode standardize Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update unit ns Switch(config-controller-OperationsMepPortType)# updateDM updateDmConfig update unit us</pre>	<p>Updates DM parameters.</p> <ul style="list-style-type: none"> <li>• <b>mepInstance</b> <i>instance_no</i>—Indicates the MEP instance. The valid values are from 1 to 128.</li> <li>• <b>update</b>—Updates DM parameters for all MEPs on the controller.</li> <li>• <b>overflowReset</b>—Reset all delay Measurement results on total delay counters.</li> <li>• <b>keep</b>—Retains all delay Measurement results.</li> <li>• <b>reset</b>—Resets all delay Measurement results.</li> <li>• <b>synchronized</b>—Synchronizes near end and far end time intervals.</li> <li>• <b>enable</b>—Enables synchronization of near and far end time interval.</li> <li>• <b>disable</b>—Disables synchronization of near and far end time interval.</li> <li>• <b>txMode</b>—Sets the Tx mode.</li> <li>• <b>proprietary</b>—Sets the proprietary delay measurement parameters .</li> <li>• <b>standardize</b>—Sets the Y.1731 standards to transmit 1DM/DMR delay measurement parameters.</li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>unit</b>—Sets the delay in units.</li> <li>• <b>ns</b>—Specifies nanoseconds.</li> <li>• <b>us</b>—Sets microseconds.</li> </ul>
<b>Step 3</b>	<b>updateDM review</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>updateDM review</b> Commands in queue: updateDM updateDmConfig mepInstance 1 updateDM updateDmConfig update overflowReset keep  reset updateDM updateDmConfig update overflowReset updateDM updateDmConfig update synchronized enable updateDM updateDmConfig update txMode proprietary updateDM updateDmConfig update txMode standardize updateDM updateDmConfig update txMode standardize  updateDM updateDmConfig update unit ns updateDM updateDmConfig update unit us	Displays the configuration on the controller.
<b>Step 4</b>	<b>updateDM commit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>updateDM commit</b>	Sends the configuration to the NID.
<b>Step 5</b>	<b>exit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType) # <b>exit</b>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to update the delay measurement parameters on the controller:

```
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig mepInstance 100
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update overflowReset
keep
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update overflowReset
reset
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update synchronized
enable
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update overflowReset
keep
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update txMode
proprietary
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update txMode
standardize
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update unit ns
Switch(config-controller-OperationsMepPortType) # updateDM updateDmConfig update unit us
```

```
Switch(config-controller-OperationsMepPortType) # updateDM review
Commands in queue:
    updateDM updateDmConfig mepInstance 1
    updateDM updateDmConfig update overflowReset keep
    updateDM updateDmConfig update overflowReset reset
    updateDM updateDmConfig update synchronized enable
    updateDM updateDmConfig update txMode proprietary
    updateDM updateDmConfig update txMode standardize
    updateDM updateDmConfig update txMode standardize
    updateDM updateDmConfig update unit ns
    updateDM updateDmConfig update unit us

Switch(config-controller-OperationsMepPortType) # updateDM commit
Switch(config-controller-OperationsMepPortType) # end
```

## Updating Test Signal Parameters on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><b>updateTst {commit   flush   updateTstConfig   review}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>updateTst ?</b>     commit                commit updateTst     flush                 flush all updateTst commands from queue     review                review updateTst commands     updateTstConfig      Update Tst signal request</pre>	<p>Updates the test signal parameters.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> <li>• <b>updateTstConfig</b>—Updates the test signal parameters.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
Step 2	<p><b>updateTst updateTstConfig {mepInstance instance_no   update {Rx   Tx} {enable   disable}}</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>updateTst updatetstConfig mepInstance 100</b> Switch(config-controller-OperationsMepPortType) # <b>updateTst updatetstConfig update Rx enable</b> Switch(config-controller-OperationsMepPortType) # <b>updateTst updatetstConfig update Tx enable</b></pre>	<ul style="list-style-type: none"> <li>• <b>mepInstance instance_no</b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> <li>• <b>update</b>—Updates DM parameters for all MEPs.</li> <li>• <b>Rx</b>—Sets the Rx mode.</li> <li>• <b>Tx</b>—Sets the Tx mode.</li> <li>• <b>enable</b>—Enables the mode.</li> <li>• <b>disable</b>—Disables the mode.</li> </ul>
Step 3	<p><b>updateTst review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # <b>updateTst review</b></pre>	<p>Displays the configuration on the controller.</p>

	Command or Action	Purpose
	<pre>Commands in queue: updateTst updateTstConfig mepInstance 2 updateTst updateTstConfig update Rx enable updateTst updateTstConfig update Tx enable updateTst updateTstConfig update Tx enable</pre>	
<b>Step 4</b>	<p><b>updateTst commit</b></p> <p><b>Example:</b>  Switch(config-controller-OperationsMepPortType)#  <b>updateTst commit</b></p>	Sends the configuration to the NID.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b>  Switch(config-controller-OperationsMepPortType)# <b>exit</b></p>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to update the test signal parameters on the controller:

```
Switch(config-controller-OperationsMepPortType)# updateTst updatetstConfig mepInstance 100
Switch(config-controller-OperationsMepPortType)# updateTst updatetstConfig update Rx enable
Switch(config-controller-OperationsMepPortType)# updateTst updatetstConfig update Tx enable

Switch(config-controller-OperationsMepPortType)# updateTst review
Commands in queue:
updateTst updateTstConfig mepInstance 2
updateTst updateTstConfig update Rx enable
updateTst updateTstConfig update Tx enable
updateTst updateTstConfig update Tx enable

Switch(config-controller-OperationsMepPortType)# updateTst commit
Switch(config-controller-OperationsMepPortType)# end
```

## Clearing MEP Statistics on the Controller

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><b>clearMepStats {commit   flush   clearStats   review}</b></p> <p><b>Example:</b>  Switch(config-controller-OperationsMepPortType)#</p>	<p>Clears the MEP statistics.</p> <ul style="list-style-type: none"> <li>• <b>commit</b>—Sends the configuration to NID.</li> <li>• <b>flush</b>—Flushes all configuration from the queue.</li> </ul>

	Command or Action	Purpose
	<pre>clearMepStats ?   clearStats  Clear mep statistics request   commit      commit clearMepStats   flush       flush all clearMepStats commands from   queue   review      review clearMepStats commands</pre>	<ul style="list-style-type: none"> <li>• <b>clearStats</b>—Clears the MEP statistics.</li> <li>• <b>review</b>—Displays the configuration on the controller.</li> </ul>
<b>Step 2</b>	<pre>clearMepStats clearStats {mepInstance <i>instance_no</i>   StatsType {DM   LM   TST}}</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # clearMepStats clearStats mepInstance 25 Switch(config-controller-OperationsMepPortType) # clearMepStats clearStats statstype DM</pre>	<ul style="list-style-type: none"> <li>• <b>mepInstance <i>instance_no</i></b>—Indicates the MEP instance. The valid values are from 1 to 128.</li> <li>• <b>Statstype</b>—Indicates the protocol type.</li> <li>• <b>DM</b>—Specifies the delay measurement statistics.</li> <li>• <b>LM</b>—Specifies the loss measurement statistics.</li> <li>• <b>TST</b>—Specifies the test signal statistics.</li> </ul>
<b>Step 3</b>	<pre>clearMepStats review</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # clearMepStats review</pre> <p>Commands in queue:</p> <pre>clearMepStats clearStats mepInstance 23</pre>	Displays the configuration on the controller.
<b>Step 4</b>	<pre>clearMepStats commit</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # clearMepStats commit</pre>	Sends the configuration to the NID.
<b>Step 5</b>	<pre>exit</pre> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType) # exit</pre>	Exits the performance monitoring provisioning mode.

### Configuration Example

The example shows how to clear the MEP statistics on the controller:

```
Switch(config-controller-OperationsMepPortType) # clearMepStats clearStats mepInstance 25
Switch(config-controller-OperationsMepPortType) # clearMepStats clearStats statstype DM
Switch(config-controller-OperationsMepPortType) # clearMepStats review

Commands in queue:
clearMepStats clearStats mepInstance 23

Switch(config-controller-OperationsMepPortType) # clearMepStats commit
Switch(config-controller-OperationsMepPortType) #end
```

## Negating Performance Monitoring Configuration and Restoring Defaults

### Before You Begin

- Perform the steps to provision performance monitoring on the controller. See [Provisioning the Controller to Configure Performance Monitoring](#), on page 3.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><b>no ?</b></p> <p><b>Example:</b> Switch(config-controller-OperationsMepPortType)# <b>no ?</b></p> <pre> clearMepStats  Clear mep statistics request exit           Exit from OperationsMepPortType sub configuration mode setAis         Enable or Disable AIS request setDM          Enable or Disable delay measurement request setLM          Enable or Disable loss measurement request setLck         Enable or Disable lock signal request setLinkTrace   Enable or Disable linktrace request setLoopBack    Enable/Disable loopback setTst         Enable or Disable test signal request showAis        Show AIS configuration request showDM         Show delay measurement statistics request showLM         Show LM statistics request showLck        Show lock signal current configuration request showLinkTrace  Show linktrace state and current configuration request showLoopBack   Show loopback state and current configuration request showTst        Show test signal statistics and current configuration request updateDM       Update DM parameters request updateTst      Update Tst signal request </pre>	Negates the commands and sets the default configuration.
<b>Step 2</b>	<p><b>exit</b></p> <p><b>Example:</b> Switch(config-controller-OperationsMepPortType)# <b>exit</b></p>	Exits the performance monitoring provisioning mode.

# Setting Performance Monitoring Parameters

## SUMMARY STEPS

1. **configure terminal**
2. **controller nid *1/NID\_ID***
3. **ProvisionMepPortType**
4. **setPerformanceMonitoring perform-mon { interval { dm { disable | dm\_value } | evc { disable | evc\_value } | lm { disable | lm\_value } } | session { dm { disable | enable } | evc { disable | enable } | lm { disable | enable } } | storage { dm { disable | enable } | evc { disable | enable } | lm { disable | enable } | dm\_binning { disable | enable } } | transfer { fixed\_offset { disable | value } | hour { disable | value } | incomplete { disable | enable } | minute { disable | minute } | mode { all | disable | fixed | new } | random\_offset { disable | random\_offset } | status { disable | enable } | url { disable | enable } } }**
5. **setPerformanceMonitoring review**
6. **setPerformanceMonitoring commit**
7. **exit**

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>controller nid <i>1/NID_ID</i></b>  <b>Example:</b> Switch(config)# controller nid 1/1	Enters the controller configuration mode.
Step 3	<b>ProvisionMepPortType</b>  <b>Example:</b> Switch(config-controller)# ProvisionMepPortType	Enters the OperationsMepPortType mode and enables provisioning of the MEP.
Step 4	<b>setPerformanceMonitoring perform-mon { interval { dm { disable   dm_value }   evc { disable   evc_value }   lm { disable   lm_value } }   session { dm { disable   enable }   evc { disable   enable }   lm { disable   enable } }   storage { dm { disable   enable }   evc { disable   enable }   lm { disable   enable }   dm_binning { disable   enable } }   transfer { fixed_offset { disable   value }   hour { disable   value }   incomplete { disable   enable }   minute { disable   minute }   mode { all   disable   fixed   new }   random_offset { disable   random_offset }   status { disable   enable }   url { disable   enable } } }</b>  <b>Example:</b>	Specify the performance monitoring parameters. <ul style="list-style-type: none"> <li>• <b>interval</b>—Specifies measurement interval. <ul style="list-style-type: none"> <li>◦ <b>dm</b> —Specifies delay measurement interval in minute or disable it. The delay management ranges from 1-60 minute.</li> </ul> </li> </ul>

Command or Action	Purpose
<pre>Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer status enable  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer hour value 4  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer minute minute 30  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer fixed-offset value 11  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer random-offset random-offset 200  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer incomplete enable  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer mode fixed 24  Switch(config-controller-OperationsMepPortType)#setPerformanceMonitoring perform-mon transfer url url tftp://202.153.144.25/tftpboot/praveen</pre>	<ul style="list-style-type: none"> <li>◦ <b>evc</b> —Specifies EVC in minute or disable it. The EVC ranges from 1-60 minute.</li> <li>◦ <b>lm</b>—Specifies loss measurement in minute or disable it. The loss measurement ranges from 1-60 minute.</li> <li>• <b>session</b>—Specifies the session. <ul style="list-style-type: none"> <li>◦ <b>dm</b> —Disable or enable intervals from previous incomplete transfers.</li> <li>◦ <b>evc</b> —Disable or enable intervals from previous incomplete transfers.</li> <li>◦ <b>lm</b>—Disable or enable intervals from previous incomplete transfers.</li> </ul> </li> <li>• <b>storage</b>—Specifies storage. <ul style="list-style-type: none"> <li>◦ <b>dm</b> —Disable or enable intervals from previous incomplete transfers.</li> <li>◦ <b>evc</b> —Disable or enable intervals from previous incomplete transfers.</li> <li>◦ <b>lm</b>—Disable or enable intervals from previous incomplete transfers.</li> <li>◦ <b>dm_binning</b>—Disable or enable intervals from previous incomplete transfers.</li> </ul> </li> <li>• <b>transfer</b>—Enable transfer mode. <ul style="list-style-type: none"> <li>◦ <b>fixed-offset</b> —Specifies a scheduled offset value in minute or disable it. Allowed range is 1-15 minute.</li> </ul> </li> </ul>



	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>◦ <b>hour</b> —Specifies scheduled hour to transfer or disable it. Allowed range is 0-23 type hour.</li> <li>◦ <b>incomplete</b>—Disable or enable intervals from previous incomplete transfers.</li> <li>◦ <b>minute</b>—Specifies scheduled minute to transfer or disable it. Allowed range is 0, 15, 30 or 45.</li> <li>◦ <b>mode</b>—Specifies interval mode.                             <ul style="list-style-type: none"> <li>◦ <b>all</b>—Specifies all available interval .</li> <li>◦ <b>disable</b>—Disable interval.</li> <li>◦ <b>fixed</b>—Specifies fixed interval. Allowed range is 1-96.</li> <li>◦ <b>new</b>—Specifies new interval since last transfer .</li> </ul> </li> <li>◦ <b>random_offset</b>—Disable or enable random offset value. Allowed range is 0-900 seconds.</li> <li>◦ <b>status</b>—Disable or enable the status.</li> <li>◦ <b>url</b>—Disable or enable server url.</li> </ul>
<p><b>Step 5</b></p>	<p><b>setPerformanceMonitoring review</b></p> <p><b>Example:</b>                      Switch(config-controller-OperationsMepPortType) # setPerformanceMonitoring review</p>	<p>Displays the performance monitoring configuration.</p>

	Command or Action	Purpose
<b>Step 6</b>	<b>setPerformanceMonitoring commit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# setPerformanceMonitoring commit	Sends the performance monitoring configuration to the Cisco ME 1200 NID.
<b>Step 7</b>	<b>exit</b>  <b>Example:</b> Switch(config-controller-OperationsMepPortType)# exit Switch(config-controller)#	Exits to the controller configuration mode.

## Viewing Performance Monitoring Parameters

### SUMMARY STEPS

1. **configure terminal**
2. **controller nid 1/NID\_ID**
3. **ProvisionMepPortType**
4. **getPerformanceMonitoring getPerfomanceMonitorParameters**
5. **getPerformanceMonitoring review**
6. **g?etPerformanceMonitoring commit**
7. **exit**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
<b>Step 2</b>	<b>controller nid 1/NID_ID</b>  <b>Example:</b> Switch(config)# controller nid 1/1	Enters the controller configuration mode.
<b>Step 3</b>	<b>ProvisionMepPortType</b>  <b>Example:</b> Switch(config-controller)# OperationsMepPortType	Enters the OperationsMepPortType mode and enables provisioning of the MEP.

	Command or Action	Purpose
Step 4	<p><b>getPerformanceMonitoring getPerformanceMonitorParameters</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)#getPerformanceMonitoring getPerfomanceMonitorParameters</pre>	Retrieve Performance Monitor parameters
Step 5	<p><b>getPerformanceMonitoring review</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# getPerformanceMonitoring review</pre>	Displays the performance monitoring configuration.
Step 6	<p><b>g?etPerformanceMonitoring commit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# getPerformanceMonitoring commit</pre>	Sends the performance monitoring configuration to the Cisco ME 1200 NID.
Step 7	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>Switch(config-controller-OperationsMepPortType)# exit Switch(config-controller)#</pre>	Exits to the controller configuration mode.

## Verifying Performance Monitoring

Use the **show perf-mon** commands to verify the Performance Monitoring status on the controller.

**show perf-mon {current | interval-id unit instance instance\_id | interval-info | id unit} {feature {dm | ece | evc | lm}}**

- This command displays the current delay measurement status. The following is a sample output from the command:  
Switch# **show perf-mon current feature dm**
- This command displays the current loss measurement status. The following is a sample output from the command:  
Switch# **show perf-mon current feature lm**
- This command displays the delay measurement status for interval-id. The following is a sample output from the command:  
Switch# **show perf-mon interval-id id 3 instance 4 feature dm**

- This command displays the loss measurement status for interval-info. The following is a sample output from the command:

```
Switch# show perf-mon interval-info 5 feature lm
```