



# Cisco XML Operational Requests and Fault Management

A client application can send an extensible markup language (XML) request to get the router operational information using either a native data <Get> request along with the <Operational> tag, or the equivalent command-line interface (CLI) command. Although the CLI is more familiar to users, the advantage of using the <Get> request is that the response data is encoded in XML format instead of being only uninterpreted text enclosed within <CLI> tags.

This chapter contains the following sections:

- [Operational Get Requests, page 3-53](#)
- [Action Requests, page 3-54](#)

## Operational Get Requests

The content and format of operational <Get> requests are described in additional detail in [Chapter 4, “Cisco XML and Native Data Operations.”](#)

The following example shows a <Get> request to retrieve the global Border Gateway Protocol (BGP) process information. This request returns BGP process information similar to that displayed by the **show ip bgp process detail** CLI command.

### Sample XML Client Request to Get BGP Information

```
<?xml version="1.0" encoding="UTF-8"?>
<Request MajorVersion="1" MinorVersion="0">
  <Get>
    <Operational>
      <BGP MajorVersion="1" MinorVersion="0">
        <GlobalProcessInfo/>
      </BGP>
    </Operational>
  </Get>
</Request>
```

### Sample XML Response from the Router

```
<?xml version="1.0" encoding="UTF-8"?>
<Response MajorVersion="1" MinorVersion="0">
  <Get>
    <Operational>
      <BGP MajorVersion="1" MinorVersion="0">
        <GlobalProcessInfo>
```

```

<InStandaloneMode>true</InStandaloneMode>
<RouterID>10.1.0.1</RouterID>
<ConfiguredRouterID>33.67.205.171</ConfiguredRouterID>
<LocalAS>3</LocalAS>
<RestartCount>1</RestartCount>
.
.
.
    returned data
.
.
.
<IsGracefulRestart>true</IsGracefulRestart>
<RestartTime>180</RestartTime>
<StalePathTime>300</StalePathTime>
<RIBPurgeTimeout>300</RIBPurgeTimeout>
<UpdateDelay>55</UpdateDelay>
</GlobalProcessInfo>
</BGP>
</Operational>
</Get>
</Response>

```

## Action Requests

A client application can send a <Set> request along with the <Action> tag to trigger unique actions on the router. For example, an object may be set with an action request to inform the router to clear a particular counter or reset some functionality. Most often this operation involves setting the value of a Boolean object to “true”. The content and format of <Set> requests are described in additional detail in [Chapter 4, “Cisco XML and Native Data Operations.”](#)

The following example shows an action request to clear the BGP performance statistics information. This request is equivalent to the **clear ip bgp performance-statistics** CLI command.

### Sample XML Request to Clear BGP Performance Statistics Information

```

<?xml version="1.0" encoding="UTF-8"?>
<Request MajorVersion="1" MinorVersion="0">
  <Set>
    <Action>
      <BGP MajorVersion="0" MinorVersion="0">
        <ClearPerformanceStats>true</ClearPerformanceStats>
      </BGP>
    </Action>
  </Set>
</Request>

```

### Sample XML Response from the Router

```

<?xml version="1.0" encoding="UTF-8"?>
<Response MajorVersion="1" MinorVersion="0">
  <Set>
    <Action/>
  </Set>
</Response>

```

In addition, the following example is showing an action request to clear the peer drop information for all BGP neighbors. This request is equivalent to the CLI command **clear ip bgp peer-drops \***.

**Sample XML Request to Clear Peer Drop Information for All BGP Neighbors**

```
<?xml version="1.0" encoding="UTF-8"?>
<Request MajorVersion="1" MinorVersion="0">
  <Set>
    <Action>
      <BGP MajorVersion="0" MinorVersion="0">
        <ClearDrops>
          <All>true</All>
        </ClearDrops>
      </BGP>
    </Action>
  </Set>
</Request>
```

**Sample XML Response from the Router**

```
<?xml version="1.0" encoding="UTF-8"?>
<Response MajorVersion="1" MinorVersion="0">
  <Set>
    <Action/>
  </Set>
</Response>
```

## Cisco XML and Fault Management

When a client application successfully commits the target configuration to the router's running configuration, the configuration manager writes a single configuration change event to system message logging (syslog). As a result, a fault management event notification is written to the Alarm Channel (that is, the Common Object Request Broker Architecture [CORBA] event notification channel for alarms) and subsequently forwarded to any registered configuration agents.

### Configuration Change Notification

Table 3-1 provides event notification for configuration changes information.

**Table 3-1** Event Notifications for Configuration Changes

Event Notification	Description
userid	The name of the user who performed the commit operation.
timestamp	The date and time of the commit.
commit	The unique ID associated with the commit.

The following example shows a configuration change notification:

```
RP/0/RP0/CPU0:Sep 18 09:43:42.747 : %CLIENTLIBCFGMGR-6-CONFIG_CHANGE : A configuration
commit by user root occurred at 'Wed Sep 18 09:43:42 2004 '. The configuration changes are
saved on the router in file: 010208180943.0
```

Upon receiving the configuration change notification, a client application can then use the <Load> and <Get> operations to load and browse the changed configuration.

