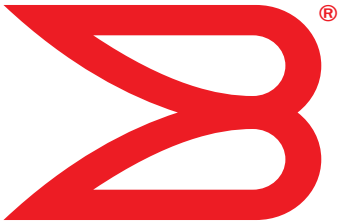
53-1003041-01
30 September, 2013



Multi-Service IronWare

YANG Configuration Guide

Supporting Multi-Service IronWare R05.6.00

BROCADE

Copyright © 2013 Brocade Communications Systems, Inc. All Rights Reserved.

ADX, AnyIO, Brocade, Brocade Assurance, the B-wing symbol, DCX, Fabric OS, ICX, MLX, MyBrocade, OpenScript, VCS, VDX, and Vyatta are registered trademarks, and HyperEdge, The Effortless Network, and The On-Demand Data Center are trademarks of Brocade Communications Systems, Inc., in the United States and/or in other countries. Other brands, products, or service names mentioned may be trademarks of their respective owners.

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. shall have no liability or responsibility to any person or entity with respect to any loss, cost, liability, or damages arising from the information contained in this book or the computer programs that accompany it.

The product described by this document may contain “open source” software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit *http://www.brocade.com/support/oscd*.

Brocade Communications Systems, Incorporated

Corporate and Latin American Headquarters
Brocade Communications Systems, Inc.
130 Holger Way
San Jose, CA 95134
Tel: 1-408-333-8000
Fax: 1-408-333-8101
E-mail: info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems China HK, Ltd.
No. 1 Guanghua Road
Chao Yang District
Units 2718 and 2818
Beijing 100020, China
Tel: +8610 6588 8888
Fax: +8610 6588 9999
E-mail: china-info@brocade.com

European Headquarters
Brocade Communications Switzerland Sàrl
Centre Swissair
Tour B - 4ème étage
29, Route de l'Aéroport
Case Postale 105
CH-1215 Genève 15
Switzerland
Tel: +41 22 799 5640
Fax: +41 22 799 5641
E-mail: emea-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems Co., Ltd. (Shenzhen WFOE)
Citic Plaza
No. 233 Tian He Road North
Unit 1308 – 13th Floor
Guangzhou, China
Tel: +8620 3891 2000
Fax: +8620 3891 2111
E-mail: china-info@brocade.com

Document History

Title	Publication number	Summary of changes	Date
Multi-Service IronWare YANG Configuration Guide	*53-1003041-01*	Release 05.5.00c document updated with enhancements in Release 05.6.00	30 September, 2013

Contents

About This Document

Audience v
Supported hardware and software v
Document conventions v
Text formatting v
Notes, cautions, and danger notices vi
Related publications vi
Getting technical help vii

Chapter 1 Overview

Introduction to YANG 1

Chapter 2 Common Definitions and Headers

In this chapter 3
Common definition and header elements 3
NetIron configuration file 7
NetIron state data 8
Device version state data 9
RPC delete operation example 16

Chapter 3 Interface Configuration

In this chapter 19
Overview 19
Interface configuration elements 19
Interface identifier 19
Alarm monitoring 20
Admin status choice 20
Gig default 21
Layer 2 switching 22
Drop precedence 22
Loop detection 23
Load interval 23
Source guard 24
Flow control 24
Speed duplex 24
IPv4 configuration 25
IPv6 configuration 25

Interface state elements 26
Full examples 32
<get> operation examples 42
<get-config> operation example 53

Chapter 4 MPLS Configuration

In this chapter 59
Overview 59
MPLS configuration elements 59
MPLS priority 60
MPLS traffic engineering 61
MPLS path 62
MPLS LSP configuration 63
MPLS RSVP configuration 67
Router MPLS configuration 67
MPLS state elements 68
Full examples 74
<get> operation examples 84
<get-config> operation example 88

Chapter 5 VLAN Configuration

In this chapter 91
Overview 91
VLAN configuration elements 91
VLAN identifier 91
VLAN name 92
Tagged ports 92
Untagged ports 92
Uplink ports 93
Router interface 93
Loop detection 93
VLAN accounting 93
VLAN CPU protection 94
VLAN state elements 94
Full examples 97

About This Document

Audience

This document is designed for system administrators with a working knowledge of Layer 2 and Layer 3 switching and routing.

Supported hardware and software

This guide describes the YANG models supported on the software release R05.6.00 of the NetIron devices.

The following hardware platforms support the YANG models described in this guide:

- Brocade NetIron CER
- Brocade NetIron CES
- Brocade NetIron XMR
- Brocade MLX series

Document conventions

This section describes text formatting conventions and important notice formats used in this document.

Text formatting

The narrative-text formatting conventions that are used are as follows:

bold text	Identifies command names
	Identifies the names of user-manipulated GUI elements
	Identifies keywords
	Identifies text to enter at the GUI or CLI
italic text	Provides emphasis
	Identifies variables
	Identifies document titles
`code` text	Identifies CLI output

For readability, command names in the narrative portions of this guide are presented in bold; for example, **show version**.

Notes, cautions, and danger notices

The following notices and statements are used in this manual. They are listed below in order of increasing severity of potential hazards.

NOTE
A note provides a tip, guidance or advice, emphasizes important information, or provides a reference to related information.

CAUTION

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER

A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Related publications

For the latest edition of these documents, which contain the most up-to-date information, see Documentation at *http://www.brocade.com/ethernetproducts*

- *Multi-Service IronWare Administration Configuration Guide*
- *Multi-Service IronWare Security Configuration Guide*
- *Multi-Service IronWare Switching Configuration Guide*
- *Multi-Service IronWare Routing Configuration Guide*
- *Multi-Service IronWare Traffic Management Configuration Guide*
- *Multi-Service IronWare Multicast Configuration Guide*
- *Multi-Service IronWare Multiprotocol Label Switch (MPLS) Configuration Guide*
- *Multi-Service IronWare Software Defined Networking (SDN) Configuration Guide*
- *Brocade MLX Series and NetIron Family YANG Configuration Guide*
- *Brocade MLX Series and NetIron XMR Series Diagnostic Reference*
- *Unified IP MIB Reference*
- *Multi-Service IronWare Software Upgrade Guide*
- *Brocade MLXe Series Installation Guide*
- *Brocade MLX Series and Brocade NetIron XMR Installation Guide*

- *Brocade NetIron CES 2000 Series and Brocade NetIron CER 2000 Series Hardware Installation Guide*

Getting technical help

To contact Technical Support, go to *http://www.brocade.com/services-support/index.page* for the latest e-mail and telephone contact information.

Chapter

1 Overview

Introduction to YANG

NOTE
Brocade maintains limited support only for this feature.

YANG is a data modeling language. The language is used to model the supported configuration and state data manipulated by the Network Configuration (NETCONF) protocol and NETCONF remote procedure calls (RPCs). For more information on NETCONF, refer to the *Multi-Service IronWare Administration Configuration Guide*.

Each block of YANG data is encapsulated as a module containing a header information, linkage statements, meta information, revision history, and module definitions. Modules can contain one or more submodules with the same structure. The following code example shows the structure of a header statement, along with linkage and meta information (which typically contains contact information and a high-level description of the module).

```
module common-defs
{
    namespace "http://brocade.com/ns/netconf/config/netiron-config/";
    prefix "common";

    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the common definitions. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }

}
```

NOTE
The YANG model does not perform data allocation or NETCONF protocol operations. Refer to the *Multi-Service IronWare Administration Configuration Guide* for more information on these topics. For the most recent YANG models, visit *http://my.brocade.com*.

Chapter

Common Definitions and Headers

2

In this chapter

- Common definition and header elements 3
- NetIron configuration file 7
- Device version state data 9
- RPC delete operation example 16

Common definition and header elements

The following YANG model contains the common type definitions and the common objects used by the other YANG models.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
module common-defs
{
    namespace "http://brocade.com/ns/netconf/config/netiron-config/";
    prefix "common";

    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the common definitions. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }

    typedef interface-all-types
    {
        type string
        {
            pattern
                 "ethernet\s+\d{1,2}/\d{1,2}|"+
                 "ethernet\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}|"+
                 "pos\s+\d{1,2}/\d{1,2}|"+
```

```
                "pos\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}|"+
                "ve\s+\d|"+
                "loopback\s+\d|"+
                "management\s+\d|"+
                "tunnel\s+\d";

                 length "4..100";
    }
    description
        "Single or range of interfaces "+
        "Format is etherenet SLOT/PORT or "+
        "etherent SLOT/PORT to SLOT/PORT "+
        "pos SLOT/PORT "+
        "pos SLOT/PORT to SLOT/PORT "+
        "ve ve_id "+
        "loopback loopback_id (1 to 64) "+
        "management mgmt_id (1 or 2) "+
        "tunnel runnel_id";
}

typedef interface-eth-pos-ve-multiple
{
    type string
    {
        pattern
             "ethernet\s+\d{1,2}/\d{1,2}|"+
             "ethernet\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}|"+
             "pos\s+\d{1,2}/\d{1,2}|"+
             "pos\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}|"+
             "ve\s+\d";

              length "4..100";
    }
    description
        "Single or range of interfaces "+
        "Format is etherenet SLOT/PORT or "+
        "etherent SLOT/PORT to SLOT/PORT "+
        "pos SLOT/PORT "+
        "pos SLOT/PORT to SLOT/PORT "+
        "ve ve_id";
}

typedef interface-eth-pos-ve-single
{
    type string
    {
        pattern
             "ethernet\s+\d{1,2}/\d{1,2}|"+
             "pos\s+\d{1,2}/\d{1,2}|"+
             "ve\s+\d";

              length "4..100";
    }
    description
        "Single or range of interfaces "+
        "Format is etherenet SLOT/PORT or "+
        "pos SLOT/PORT "+
        "ve ve_id";
}
```

```
typedef ethernet-intf-single
{
    type string
    {
        pattern "ethernet\s+\d{1,2}/\d{1,2}";
        length "11..20";
    }
    description "Single ethernet interface";
}

typedef ethernet-pos-intf-single
{
    type string
    {
        pattern
            "ethernet\s+\d{1,2}/\d{1,2}|"+
            "pos\s+\d{1,2}/\d{1,2}";
        length "11..20";
    }
    description "Single ethernet or pos interface";
}

typedef ve-intf-single
{
    type string
    {
        pattern "ve\s+\d";
        length "4..20";
    }
    description "VE interface";
}

typedef ethernet-intf-multiple
{
    type string
    {
        pattern
             "ethernet\s+\d{1,2}/\d{1,2}|"+
             "ethernet\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}";
        length "11..50";
    }
    description "Multiple ethernet interfaces";
}

typedef ethernet-pos-intf-multiple
{
    type string
    {
        pattern
             "ethernet\s+\d{1,2}/\d{1,2}|"+
             "ethernet\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}|"+
             "pos\s+\d{1,2}/\d{1,2}|"+
             "pos\s+\d{1,2}/\d{1,2}\s+to\d{1,2}/\d{1,2}";
        length "7..50";
    }
    description "Multiple ethernet or pos interfaces";
}

typedef vlan-identifier
{
```

```
    type uint32
    {
        range "1..4090";
    }
    description "VLAN Identifier";
}

typedef ipv4_address
{
    type string
    {
        length "7..15";
    }
    description "IPv4 Address";
}

typedef ipv6_address
{
    type string
    {
        length "3..39";
    }
    description "IPv6 Address";
}

typedef ipv4_address_mask
{
    type string
    {
        length "9..19";
    }
    description "IPv4 Address/mask";
}

typedef ipv6_address_mask
{
    type string
    {
        length "6..43";
    }
    description "IPv6 Address/mask";
}


typedef mac_address
{
    type string
    {
        length "1..21";
    }
    description "Ethernet MAC address";
}
typedef priority-value
{
    type uint32
    {
        range "0..7";
    }
    description "Priority Value";
}
```

```
    grouping priority-grouping
    {
        description "Priority";
        container priority
        {
            description "Priority";
            leaf value
            {
                type priority-value;
                description "Priority value";
            }
            leaf force
            {
                type empty;
                description "Priority force";
            }
        }

    }
}
```

NetIron configuration file

The following YANG model defines the header information, linkage statements, meta information, and the revision history for the NetIron configuration group.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
module netiron-config
{
    namespace "http://brocade.com/ns/netconf/config/netiron-config/";
    prefix "brcd";

    include common-defs;
    include vlan-config;
    include interface-config;
    include mpls-config;

    organization
    "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description
    "NetIron Config module. VERSION: ";

    revision 2011-04-20
    {
    description "Initial revision";
```

```
    }
    grouping netiron-config
    {
        description "NetIron configuration group";
        container netiron-config
        {
            description "NetIron configuration";
            uses vlan-config-grouping;
            uses interface-config-grouping;
            uses mpls-config-grouping;
        }
    }
}
```

NetIron state data

The following YANG model contains the state data for the NetIron device.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
module netiron-statedata
{
    namespace "http://brocade.com/ns/netconf/config/netiron-config/";
    prefix "statedata";

    include vlan-statedata;
    include interface-statedata;
    include mpls-statedata;

    organization
    "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description
    "NetIron device state data module. VERSION: ";

    revision 2011-04-20
    {
    description "Initial revision";
    }

    grouping netiron-statedata
    {
        description "NetIron device state data group";
        container netiron-statedata
        {
            description "NetIron device state data";
           config false;
            uses vlan-statedata-grouping;
```

```
            uses interface-statedata-grouping;
            uses mpls-statedata-grouping;
        }
    }
}
```

Device version state data

The following YANG model contains the state data for the device version.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule version-statedata
{
    belongs-to netiron-statedata
    {
        prefix "statedata";
    }

    import common-defs
    {
        prefix "common";
    }

    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the device version state data. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }

    grouping device-id-grouping
    {
        leaf serial
        {
            type string
            {
                length 1..20;
            }
            description "device serial number";
        }

        leaf part
        {
            type string
```

```
        {
            length 1..20;
        }
        description "device part number";
    }
}

grouping software-version-grouping
{
    leaf version
    {
        type string
        {
            length 1..63;
        }
        description "Image version.";
    }

    leaf compiled-on
    {
        type string
        {
            length 1..127;
        }
        description "The image compilation date and time.";
    }

    leaf label
    {
        type string
        {
            length 1..15;
        }
        description "The image label..";
    }

    leaf length
    {
        type uint32
        {
            range "0..4294967295";
        }
        description "The image length.";
    }

    leaf source
    {
        type string
        {
            length 1..63;
        }
        description "The boot source";
    }
}

grouping flash-grouping
{
    leaf flash-size
    {
        type uint32
```

```
        {
            range "0..4294967295";
        }
        description "The flash size.";
    }

    leaf flash-name
    {
        type string
        {
            length "1..31";
        }
        description "The flash name.";
    }
}

grouping license-grouping
{
    description "Software License Id";
    leaf software-packaging-type
    {
        type string
        {
            length 1..31;
        }
        description "Software Packaging Type";
    }
    leaf license-id
    {
        type string
        {
                    length 1..11;
        }
        description "device license identifier";
    }
}

grouping version-statedata-grouping
{
    description "device version state data";
    container version-statedata
    {
        leaf system
        {
            type string
            {
                length 1..127;
            }
            description "System model/product/chassis type";
        }

        leaf chassis-mode
        {
            type string
            {
                length 1..16;
            }
            description "Chassis mode (XMR/MLX)";
        }
```

```
container system-id
{
    description "System Identifier";
    uses device-id-grouping;
}

container factory-id
{
    description "System Factory Identifier";
    uses device-id-grouping;
}

container license
{
    description "Software License Id";
    uses license-grouping;
}

leaf cpld-version
{
    type uint8
    {
        range "0..255";
    }
    description "The CPLD Version.";
}

leaf micro-controller-version
{
    type uint8
    {
        range "0..255";
    }
    description "The Micro Controller Version.";
}

leaf pbif-version
{
    type uint8
    {
        range "0..255";
    }
    description "The PBIF Version (0 = not ready).";
}

container daughter-board-id
{
    description "Daughter Board";
    uses device-id-grouping;
}

list module
{
    key "module-name";
    description "The module name (CER has only 1 module).";
    leaf module-name
    {
        type string
        {
            length 1..256;
```

```
    }
    description "Module name";
}

leaf module-state
{
    type string
    {
        length 1..16;
    }
    description "Module state (if down)";
}

container module-id
{
    description "Module Identifier";
    uses device-id-grouping;
}

list fe
{
    key "fe-id";
    description "The FE";
    leaf fe-id
    {
        type uint8
        {
            range 1..16;
        }
        description "FE id";
    }
    leaf fe-type
    {
        type uint32
        {
            range 1..1048575;
        }
        description "FE type";
    }

    leaf fe-ver
    {
        type uint8
        {
            range 1..16;
        }
        description "FE version";
    }
}

container module-license
{
    description "Software License Id";
    uses license-grouping;
}

container software-version
{
    description "Software Versions";
    container boot
```

```
    {
        description "Boot Image Software Versions";
        uses software-version-grouping;
    }
    container monitor
    {
        description "Monitor Image Software Versions";
        uses software-version-grouping;
    }
    container ironware
    {
        description "Ironware Image Software Versions";
        uses software-version-grouping;
    }
}

leaf fpga-versions
{
    type uint8
    {
        range "0..255";
    }
    description "The FPGA Version.";
}

container mac-string-container
{
    description "MAC string container";
    leaf-list mac-string-list
    {
        description "The MAC string.";
        type string
        {
            length "1..15";
        }
    }
}

container processor
{
    leaf processor-speed
    {
        type uint32
        {
            range "0..4294967295";
        }
        description "The Processor Speed (MHz)";
    }

    leaf processor-type
    {
        type string
        {
            length "1..5";
        }
        description "Processor Type";
    }

    leaf processor-revision
    {
```

```
            type uint32
            {
                range "0..4294967295";
            }
            description "Processor revision";
        }

        leaf bus-speed
        {
            type uint32
            {
                range "0..4294967295";
            }
            description "The Bus Speed (MHz)";
        }
    }

    container boot-flash
    {
        description "Boot Flash";
        uses flash-grouping;
    }

    container code-flash
    {
        description "Code Flash";
        uses flash-grouping;
    }

    leaf dram-size
    {
        type uint32
        {
            range "0..4294967295";
        }
        description "The DRAM size.";
    }

    leaf sram-size
    {
        type uint32
        {
            range "0..4294967295";
        }
        description "The SRAM size.";
    }

    leaf bram-size
    {
        type uint32
        {
            range "0..4294967295";
        }
        description "The BRAM size.";
    }

    container dma-container
    {
        description "DMA information container.";
        list dma-list
```

```
                        {
                            key "ppcr-id";
                            description "The DMA information container.";
                            leaf ppcr-id
                            {
                                type uint32
                                {
                                    range "0..4294967295";
                                }
                                description "The PPCR Id.";
                            }
                            leaf cam-size
                            {
                                type uint32
                                {
                                    range "0..4294967295";
                                }
                                description "The CAM size (K entries)";
                            }
                            leaf pram-size
                            {
                                type uint32
                                {
                                    range "0..4294967295";
                                }
                                description "The PRAM size (K entries)";
                            }
                            leaf age-ram-size
                            {
                                type uint32
                                {
                                    range "0..4294967295";
                                }
                                description "The AGE RAM size (K entries)";
                            }
                        }
                    }

                    leaf system-uptime
                    {
                        type string
                        {
                            length "1..63";
                        }
                        description "System Uptime";
                    }
                }
            }
        }
    }
```

RPC delete operation example

The following example shows the RPC request and response for a delete operation.

```
<nc:rpc message-id="1" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/">
<nc:edit-config>
```

```
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<brcd:netiron-config>
<brcd:interface-config>
<brcd:interface nc:operation="delete">
<brcd:interface-id>ethernet 1/1</brcd:interface-id>
</brcd:interface>
</brcd:interface-config>
</brcd:netiron-config>
</nc:config>
</nc:edit-config>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="1">
<nc:ok></nc:ok>
</nc:rpc-reply>
]]>]]>
```

RPC delete operation example

Chapter

Interface Configuration 3

In this chapter

- Overview 19
- Interface configuration elements 19
- Interface state elements 26
- Full examples 32

Overview

This chapter describes the interface components that provide support for configuring the network interfaces, both link-level (including VLAN-tagged interfaces) and IP addresses (IPv4 and IPv6).

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

Interface configuration elements

Each element of the interface corresponds to the following:

- A YANG model description
- An XML description
- A CLI description

The following subsections compare a specific example for each interface configuration element.

Interface identifier

The interface identifier is a restricted integer type that behaves as a unique identifier for single or a group of interfaces. This element is an edit request for an Ethernet interface and a VE interface and disables each port.

TABLE 1 Interface identifier

YANG model	XML	CLI

leaf interface-id
{
type interface-all-types;
description "Single or group of interfaces";
}
	<*brcd:interface-id*>ethernet 1/1</*brcd:interface-id*>
	<*brcd:interface-id*>ve 2</*brcd:interface-id*>
		interface ethernet
		<*slot-number*>/<*port-number*>

Alarm monitoring

This data element indicates whether or not alarm monitoring is enabled. This data element only applies to 10 GbE interfaces previously configured in the WAN PHY mode.

TABLE 2 Alarm monitoring

YANG model	XML	CLI

leaf alarm-monitoring
{
type empty;
description "Enable alarm monitoring";
}
	<brcd:alarm-monitoring></brcd:alarm-monitoring>	[**no**] **alarm-monitoring**

Admin status choice

This data element is used to enable or disable the interface.

TABLE 3 Admin status choice

YANG model	XML	CLI
choice admin-status-choice { description "Enable/Disable the Interface";		
case enable {		

leaf enable
{
type empty;
description "Enable the Interface";
}
	<brcd:enable></brcd:enable>	**enable**

}
case disable
{		

leaf disable
{
type empty;
description "Disable the Interface";
}
	<brcd:disable></brcd:disable>	**disable**

} }		

Gig default

This data element indicates the Gigabit negotiation modes.

TABLE 4 Gig default

YANG model	XML	CLI

container gig-default
{
description "Global Gig port default options";
choice gig-default-option
{
description "Gig default type";
case auto-gig
{
leaf auto-gig
{
type empty;
description "Autonegotiation";
}
}
case neg-off
{
leaf neg-off
{
type empty;
description "Non-Autonegotiation";
}
}
case auto-full
{
leaf auto-full
{
type empty;
description " Full Autonegotiation";
}
}
case neg-full-auto
{
leaf neg-full-auto
{
type empty;
description "Autonegotiation first, "+
"if failed try non-autonegotiation";
}
}
}
}
	<brcd:gig-default>
	<brcd:auto-gig></brcd:auto-gig>
	</brcd:gig-default>
	
	<brcd:gig-default>
	<brcd:neg-off></brcd:neg-off>
	</brcd:gig-default>
	
	<brcd:gig-default>
	<brcd:auto-full></brcd:auto-full>
	</brcd:gig-default>
	
	<brcd:gig-default>
	<brcd:neg-full-auto></brcd:neg-full-auto>
	</brcd:gig-default>
		[no] **gig-default auto-gig** | **neg-off** | **auto-full** | **neg-full-auto**

Layer 2 switching

This data element indicates whether or not Layer 2 switching is enabled.

TABLE 5 Layer 2 switching

YANG model	XML	CLI

leaf route-only
{
type empty;
description "Enable Layer 2 switching";
}
	<*brcd:route-only*></*brcd:route-only*>	[**no**] **route-only**

Drop precedence

This data element indicates the Quality of Service (QoS) drop precedence value.

TABLE 6 Drop precedence

YANG model	XML	CLI

container drop-precedence
{
description "QOS drop precedence";
leaf value
{
type uint32
{
range "0..3";
}
description "Drop precedence value";
}
leaf force
{
type empty;
description "Drop precedence force";
}
}
	<*brcd:drop-precedence*>
	<*brcd:value*>0</*brcd:value*>
	</*brcd:drop-precedence*>
	
	<*brcd:drop-precedence*>
	<*brcd:force*></*brcd:force*>
	</*brcd:drop-precedence*>
		[**no**] **drop-precedence** <*dp-value*>
		
		[**no**] **drop-precedence force**

Loop detection

This data element indicates whether or not loop detection is enabled.

TABLE 7 Loop detection

YANG model	XML	CLI

container loop-detection
{
description "Shutdown this port if receiving "+
"packets originated from this port";
leaf vlan
{
type vlan-identifier;
description "Loop detection for this vlan";
}
leaf shutdown-disable
{
type empty;
description "Disable shutdown by "+
"loop detection";
}
leaf shutdown-sending-port
{
type empty;
description "Shutdown the sending port";
}
}
	<brcd:loop-detection>
	<brcd:vlan>2</brcd:vlan>
	</brcd:loop-detection>
	
	<brcd:loop-detection>
	<brcd:shutdown-disable></brcd:shutdown-disable>
	</brcd:loop-detection>
	
	<brcd:loop-detection>
	<brcd:shutdown-sending-port></brcd:shutdown-sending-port>
	</brcd:loop-detection>
		[**no**] **loop-detection** [**vlan** *<vlan_id>*]
		
		loop-detection shutdown-disable
		
		[**no**] **loop-detection shutdown-sending-port**

Load interval

This data element indicates the time of the load interval during which the average byte and packet rates are calculated. The time can be configured in multiples of 30 seconds in the range from 30 through 300 seconds. The default value of the load interval is 300 seconds.

TABLE 8 Load interval

YANG model	XML	CLI

leaf load-interval
{
type uint32
{
range "30..300";
}
units "Seconds";
description "Load interval, multiple of 30";
}
	*<brcd:load-interval>*30*</brcd:load-interval>*	[**no**] **load-interval** *<seconds>*

Source guard

This data element indicates whether or not the port is set as an IP source guarded port.

NOTE
This data element is supported only on Brocade MLX series and Brocade NetIron XMR platforms.

TABLE 9 Source guard

YANG model	XML	CLI

leaf source-guard
{
type empty;
description "Enable/ IP source guard";
}
	<brcd:source-guard></brcd:source-guard>	[no] **source-guard**

Flow control

This data element indicates the flow control status on full-duplex ports on a system.

TABLE 10 Flow control

YANG model	XML	CLI

leaf flow-control
{
type empty;
description "Enable 802.3 flow control "+
"on full duplex port";
}
	<brcd:flow-control></brcd:flow-control>	[no] **flow-control**

Speed duplex

This data element sets the port speed of an interface.

TABLE 11 Speed duplex

YANG model	XML	CLI

leaf speed-duplex
{
type interface_speed_duplex;
description "Speed duplex for this interface";
}
	*<brcd:speed-duplex>*auto*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*1000-master*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*1000-slave*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*1000-full*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*1000-half*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*100-full*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*100-half*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*10-full*</brcd:speed-duplex>*
	*<brcd:speed-duplex>*10-half*</brcd:speed-duplex>*
		[no] **speed-duplex** *<value>*

IPv4 configuration

This data element indicates the IPv4 address of an interface.

TABLE 12 IPv4 configuration

YANG model	XML	CLI

container ip
{
description "Interface IPv4 configuration";
leaf address
{
type ipv4_address_mask;
description "IPv4 address";
}
}
	<*brcd:ip*>
	<*brcd:address*>1.1.1.1/24</*brcd:address*>
	</*brcd:ip*>
		[**no**] **ip address** <*ip-addr*>/ <*ip-mask*>

IPv6 configuration

This data element indicates if IPv6 is enabled on this interface and the IPv6 address of the interface.

TABLE 13 IPv6 configuration

YANG model	XML	CLI

container ipv6
{
description "Interface IPv6 configuration";
leaf address
{
type ipv6_address_mask;
description "IPv6 address";
}
leaf enable
{
type empty;
description "Enable IPv6 on this interface";
}
}
	<*brcd:ipv6*>
	<*brcd:address*>::1/32</*brcd:address*>
	</*brcd:ipv6*>
	
	<*brcd:ipv6*>
	<*brcd:enable*></*brcd:enable*>
	</*brcd:ipv6*>
		[**no**] **ipv6 address** <*IPv6 address*>/<*IPV6 mask*>
		
		[**no**] **ipv6 enable**

Interface state elements

The following is the YANG model for the Interface state data elements.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule interface-statedata
{
    belongs-to netiron-statedata
    {
        prefix "statedata";
    }

    import common-defs
    {
        prefix "common";
    }

    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the device interface state data. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }
    grouping interface-statedata-grouping
    {
        description "device interface state data group";
        container interface-statedata
        {
            description "List of interface configured on the system";

            list interface
            {
                key interface-id;
                description "device interface state data";

                leaf interface-id
                {
                    type common:interface-all-types;
                    description "The interface identifier including chassis slot
and port number";
                }

                choice link-state
                {
                    description "The state of the link";
```

```
    case up
    {
        leaf up
        {
            type empty;
            description "The link is up";
        }
    }
    case down
    {
        leaf down
        {
            type empty;
            description "The link is down";
        }
    }
}

choice l2-state
{
    description "The L2 state for the port";

    case disabled
    {
        leaf disabled
        {
            type empty;
            description "The L2 state is disabled";
        }
    }
    case blocked
    {
        leaf blocked
        {
            type empty;
            description "The L2 state is blocked";
        }
    }
    case listening
    {
        leaf listening
        {
            type empty;
            description "The L2 state is listening";
        }
    }
    case learning
    {
        leaf learning
        {
            type empty;
            description "The L2 state is learning";
        }
    }
    case preforwarding
    {
        leaf preforwarding
        {
            type empty;
```

```
            description "The L2 state is preforwarding";
        }
    }
    case forward
    {
        leaf forward
        {
            type empty;
            description "The L2 state is forward";
        }
    }
    case none
    {
        leaf none
        {
            type empty;
            description "The L2 state is none";
        }
    }
    case lk-disable
    {
        leaf lk-disable
        {
            type empty;
            description "The L2 state is LK-DISABLE";
        }
    }
    case oam-disable
    {
        leaf oam-disable
        {
            type empty;
            description "The L2 state is OAM-DISABLE";
        }
    }
    case dot1x-blocked
    {
        leaf dot1x-blocked
        {
            type empty;
            description "The L2 state is DOT1X-BLOCKED";
        }
    }
    case lacp-blocked
    {
        leaf lacp-blocked
        {
            type empty;
            description "The L2 state is LACP-BLOCKED";
        }
    }
    case chdlc-down
    {
        leaf chdlc-down
        {
            type empty;
            description "The L2 state is cHDLC-DOWN";
        }
    }
    case ppp-down
```

```
    {
        leaf ppp-down
        {
            type empty;
            description "The L2 state is PPP-DOWN";
        }
  }
    case inactive
    {
        leaf inactive
        {
            type empty;
            description "The L2 state is inactive";
        }
    }
}

choice duplex
{
    description "The duplex mode of the interface";

    case half
    {
        leaf half
        {
            type empty;
            description "Half Duplex";
        }
    }
    case full
    {
        leaf full
        {
            type empty;
            description "Full Duplex";
        }
    }
    case none
    {
        leaf none
        {
            type empty;
            description "None";
        }
    }
}

leaf speed
{
    type string
    {
        length "1..6";
    }
    description "The bandwidth of the interface";
}

leaf trunk-id
{
    type string
    {
```

```
            length "1..6";
        }
        description "The trunk-id";
    }

    choice tag-mode
    {
        description "The tag mode of the interface";

        case yes
        {
            leaf yes
            {
                type empty;
                description "Yes";
            }
        }
        case no
        {
            leaf no
            {
                type empty;
                description "No";
            }
        }
        case na
        {
            leaf na
            {
                type empty;
                description "N/A";
            }
        }
    }

    choice priority-level
    {
        description "The priority level";

        case level0
        {
            leaf level0
            {
                type empty;
                description "Level 0";
            }
        }
        case level1
        {
            leaf level1
            {
                type empty;
                description "Level 1";
            }
        }
        case level2
        {
            leaf level2
            {
                type empty;
```

```
                description "Level 2";
            }
        }
        case level3
        {
            leaf level3
            {
                type empty;
                description "Level 3";
            }
        }
        case level4
        {
            leaf level4
            {
                type empty;
                description "Level 4";
            }
        }
        case level5
        {
            leaf level5
            {
                type empty;
                description "Level 5";
            }
        }
        case level6
        {
            leaf level6
            {
                type empty;
                description "Level 6";
            }
        }
        case level7
        {
            leaf level7
            {
                type empty;
                description "Level 7";
            }
        }
    }

    leaf mac-address
    {
        type common:mac_address;
        description "The MAC address of the inteface";
    }

    leaf name
    {
        type string
        {
            length "1..256";
        }
        description "The full port name of the interface";
    }
```

```
            }

        }

    }

}
```

Full examples

The following shows the complete YANG model for interface configuration, as well as example RPC requests and responses.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule interface-config
{
    belongs-to netiron-config
    {
        prefix brcd;
    }

    import common-defs
    {
        prefix "common";
    }


    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the Interface configuration. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }

    typedef interface_speed_duplex
    {
        type enumeration
        {
            enum auto
            {
                value "0";
                description "Autonegotiation";
            }
            enum 10-full
```

```
        {
            value "1";
            description "10M, full duplex";
        }
        enum 10-half
        {
            value "2";
            description "10M, half duplex";
        }
        enum 100-full
        {
            value "3";
            description "100M, full duplex";
        }
        enum 100-half
        {
            value "4";
            description "100M, half duplex";
        }
        enum 1000-full
        {
            value "5";
            description "1G, full duplex";
        }
        enum 1000-half
        {
            value "6";
            description "1G, half duplex";
        }
        enum 1000-master
        {
            value "7";
            description "1G, master";
        }
       enum 1000-slave
        {
            value "9";
            description "1G, slave. The 1000-slave speed duplex is"
                      + " applicable only for CES/CER devises";
        }
    }
    description "Speed duplex";
}


grouping interface-config-grouping
{
    description "Interface configuration group";
    container interface-config
    {
   description "Interface configuration";
        list interface
        {
            key interface-id;
            description "List of Interfaces configured "+
                        "on the system";
            leaf interface-id
            {
                type interface-all-types;
                description "Single or group of interfaces";
```

```
}

leaf alarm-monitoring
{
    type empty;
    description "Enable alarm monitoring";
}

choice admin-status-choice
{
    description "Enable/Tear down the LSP";

    case enable
    {
        leaf enable
        {
            type empty;
            description "Enable the LSP";
        }
    }
    case disable
    {
        leaf disable
        {
            type empty;
            description "Tear down the LSP";
        }
    }
}

container gig-default
{
    description "Global Gig port default options";
    choice gig-default-option
    {
        description "Gig default type";
        case auto-gig
        {
            leaf auto-gig
            {
                type empty;
                description "Autonegotiation";
            }
        }
        case neg-off
        {
            leaf neg-off
            {
                type empty;
                description "Non-Autonegotiation";
            }
        }
        case auto-full
        {
            leaf auto-full
            {
                type empty;
                description " Full Autonegotiation";
            }
        }
```

```
        case neg-full-auto
        {
            leaf neg-full-auto
            {
                type empty;
                description "Autonegotiation first, "+
  "if failed try non-autonegotiation";

            }
        }
    }

}

leaf route-only
{
    type empty;
    description "Enable Layer 2 switching";
}

container drop-precedence
{
    description "QOS drop precedence";
    leaf value
    {
        type uint32
        {
            range "0..3";
        }
        description "Drop precedence value";
    }
    leaf force
    {
        type empty;
        description "Drop precedence force";

    }
}

container loop-detection
{
    description "Shutdown this port if receiving "+
             "packets originated from this port";
    leaf vlan
    {
        type vlan-identifier;
        description "Loop detection for this vlan";
    }

    leaf shutdown-disable
    {
        type empty;
        description "Disable shutdown by "+
            "loop detection";
    }

    leaf shutdown-sending-port
    {
        type empty;
        description "Shutdown the sending port";
```

```
            }
        }

        leaf load-interval
        {
            type uint32
            {
                range "30..300";
            }
            units "Seconds";
            description "Load interval, multiple of 30";
        }

leaf source-guard
{
    type empty;
    description "Enable/ IP source guard";
}

        leaf flow-control
        {
            type empty;
            description "Enable 802.3 flow control "+
                      "on full duplex port";
        }

        leaf speed-duplex
        {
            type interface_speed_duplex;
            description "Speed duplex for this interface";
        }

        container ip
        {
            description "Interface IPv4 configuration";
            leaf address
            {
                type ipv4_address_mask;
                description "IPv4 address";
            }
        }

        container ipv6
        {
            description "Interface IPv6 configuration";
            leaf address
            {
                type ipv6_address_mask;
                description "IPv6 address";
            }
            leaf enable
            {
                type empty;
                description "Enable IPv6 on this interface";
            }
        }

        uses priority-grouping;

    }
```

```
          }

    }

}
```

The following example shows the RPC request and response of an <edit-config> operation.

```
<nc:rpc message-id="1" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/">
<nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<brcd:netiron-config>
<brcd:vlan-config>
<brcd:vlan>
<brcd:vlan-id>2</brcd:vlan-id>
<brcd:untagged>ethernet 1/6 to 1/8</brcd:untagged>
<brcd:router-interface> ve 2</brcd:router-interface>
</brcd:vlan>
</brcd:vlan-config>
<brcd:interface-config>
<brcd:interface>
<brcd:interface-id>ethernet 1/1</brcd:interface-id>
<brcd:enable></brcd:enable>
<brcd:ip>
<brcd:address>1.1.1.1/24</brcd:address>
</brcd:ip>
<brcd:ipv6>
<brcd:address>2000:0:0:1::1/64</brcd:address>
<brcd:enable></brcd:enable>
</brcd:ipv6>
<brcd:flow-control></brcd:flow-control>
<brcd:load-interval>30</brcd:load-interval>
<brcd:drop-precedence>
<brcd:value>0</brcd:value>
</brcd:drop-precedence>
<brcd:route-only></brcd:route-only>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/2</brcd:interface-id>
<brcd:enable></brcd:enable>
<brcd:ip>
<brcd:address>1.1.12.1/24</brcd:address>
</brcd:ip>
<brcd:ipv6>
<brcd:address>2000:0:0:12::1/64</brcd:address>
<brcd:enable></brcd:enable>
</brcd:ipv6>
<brcd:flow-control></brcd:flow-control>
<brcd:load-interval>30</brcd:load-interval>
<brcd:drop-precedence>
<brcd:value>0</brcd:value>
</brcd:drop-precedence>
<brcd:route-only></brcd:route-only>
</brcd:interface>
<brcd:interface>
```

```
<brcd:interface-id>ethernet 1/3</brcd:interface-id>
<brcd:disable></brcd:disable>
<brcd:ip>
<brcd:address>1.1.3.1/24</brcd:address>
</brcd:ip>
<brcd:ipv6>
<brcd:address>2000:0:0:3::1/64</brcd:address>
<brcd:enable></brcd:enable>
</brcd:ipv6>
<brcd:flow-control></brcd:flow-control>
<brcd:load-interval>30</brcd:load-interval>
<brcd:drop-precedence>
<brcd:value>0</brcd:value>
</brcd:drop-precedence>
<brcd:route-only></brcd:route-only>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/4</brcd:interface-id>
<brcd:enable></brcd:enable>
<brcd:ip>
<brcd:address>1.1.4.1/24</brcd:address>
</brcd:ip>
<brcd:ipv6>
<brcd:address>2000:0:0:4::1/64</brcd:address>
<brcd:enable></brcd:enable>
</brcd:ipv6>
<brcd:flow-control></brcd:flow-control>
<brcd:load-interval>30</brcd:load-interval>
<brcd:drop-precedence>
<brcd:value>3</brcd:value>
</brcd:drop-precedence>
<brcd:route-only></brcd:route-only>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/5</brcd:interface-id>
<brcd:enable></brcd:enable>
<brcd:ip>
<brcd:address>1.1.5.1/24</brcd:address>
</brcd:ip>
<brcd:ipv6>
<brcd:address>2000:0:0:5::1/64</brcd:address>
<brcd:enable></brcd:enable>
</brcd:ipv6>
<brcd:flow-control></brcd:flow-control>
<brcd:load-interval>30</brcd:load-interval>
<brcd:drop-precedence>
<brcd:value>0</brcd:value>
</brcd:drop-precedence>
<brcd:route-only></brcd:route-only>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/6</brcd:interface-id>
<brcd:enable></brcd:enable>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/7</brcd:interface-id>
<brcd:enable></brcd:enable>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/8</brcd:interface-id>
```

```
<brcd:enable></brcd:enable>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ve 2</brcd:interface-id>
<brcd:enable></brcd:enable>
<brcd:ip>
<brcd:address>1.1.2.1/24</brcd:address>
</brcd:ip>
<brcd:ipv6>
<brcd:address>2000:0:0:2::1/64</brcd:address>
<brcd:enable></brcd:enable>
</brcd:ipv6>
</brcd:interface>
</brcd:interface-config>
</brcd:netiron-config>
</nc:config>
</nc:edit-config>
</nc:rpc>
]]>]]>
```

The following example shows the complete CLI for an interface configuration.

```
Brocade# show run
Current configuration:
!
ver V5.2.0T183
!
!

no spanning-tree
!
!
vlan 1 name DEFAULT-VLAN
!
vlan 2
 untagged ethe 1/6 to 1/8
 router-interface ve 2
!
!
system-max vlan 4095
system-max virtual-interface 4095
!
aaa authentication enable default radius tacacs+ local
aaa authentication login default local radius tacacs+
aaa authentication login privilege-mode
aaa authorization exec default  tacacs+ radius
aaa accounting commands 0 default start-stop  radius
aaa accounting exec default start-stop  radius
aaa accounting system default start-stop  radius
radius-server host 10.20.99.135 auth-port 1812 acct-port 1813 default key 0 rxperf
tacacs-server host 10.20.99.135 auth-port 49 default key 2 $VUNTblVH
sntp server 10.20.99.135 4
sntp server 10.20.99.130 4
enable snmp config-radius
enable aaa console
console timeout 5
logging host 10.20.99.136
logging console
username brocade password .....
username nclocalnormal privilege 5 password .....
username nclocalsuper password .....
ip route 0.0.0.0/0 10.20.96.1
!
!
!
!
!
snmp-server
snmp-server community ..... ro
fdp run
!
netconf server
!
!
!
!
!
!
!
interface management 1
```

```
 ipv6 address 2001::10:20:99:187/64
 ip address 10.20.99.187/20
 enable
!
interface ethernet 1/1
 enable
 load-interval 30
 route-only
 ip address 1.1.1.1/24
 ipv6 address 2000:0:0:1::1/64
 ipv6 enable
!
interface ethernet 1/2
 enable
 load-interval 30
 route-only
 ip address 1.1.12.1/24
 ipv6 address 2000:0:0:12::1/64
 ipv6 enable
!
interface ethernet 1/3
 load-interval 30
 route-only
 ip address 1.1.3.1/24
 ipv6 address 2000:0:0:3::1/64
 ipv6 enable
!
interface ethernet 1/4
 enable
 load-interval 30
 route-only
 ip address 1.1.4.1/24
 ipv6 address 2000:0:0:4::1/64
 ipv6 enable
 drop-precedence 3
!
interface ethernet 1/5
 enable
 load-interval 30
 route-only
 ip address 1.1.5.1/24
 ipv6 address 2000:0:0:5::1/64
 ipv6 enable
!
interface ethernet 1/6
 enable
!
interface ethernet 1/7
 enable
!
interface ethernet 1/8
 enable
!
interface ve 2
 ip address 1.1.2.1/24
 ipv6 address 2000:0:0:2::1/64
 ipv6 enable
!
!
!
```

```
!
!
!
!
end
```

<get> operation examples

The following example shows the RPC request and response of a <get> operation with filter for interface statedata.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:get>
<nc:filter >
<brcd:netiron-statedata>
<brcd:interface-statedata/>
</brcd:netiron-statedata>
</nc:filter>
</nc:get>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:data>
<netiron-statedata xmlns="http://www.brocade.com/netconf/config/netiron-config/">
<brcd:interface-statedata>
<brcd:interface>
  <brcd:interface-id>management 1</brcd:interface-id>
  <brcd:link-state>
   <brcd:up></brcd:up>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:forward></brcd:forward>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:full></brcd:full>
  </brcd:duplex>
  <brcd:speed>1G</brcd:speed>
  <brcd:tag-mode>
   <brcd:yes></brcd:yes>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8180</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/1</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
```

```
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8181</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/2</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
 </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8182</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/3</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8183</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/4</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
```

```
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8184</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/5</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8185</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/6</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8186</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/7</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
```

```
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8187</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/8</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8188</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/9</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8189</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/10</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
```

```
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.818a</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/11</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.818b</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/12</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.818c</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/13</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
```

```
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.818d</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/14</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.818e</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/15</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.818f</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/16</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
```

```
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8190</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/17</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8191</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/18</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8192</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/19</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
```

```
  </brcd:duplex>
 <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8193</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/20</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8194</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/21</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8195</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/22</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
```

```
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8196</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/23</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8197</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/24</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
    </brcd:duplex>
    <brcd:speed></brcd:speed>
    <brcd:tag-mode>
     <brcd:no></brcd:no>
    </brcd:tag-mode>
    <brcd:priority-level>
     <brcd:level0></brcd:level0>
    </brcd:priority-level>
    <brcd:mac-address>748e.f811.8198</brcd:mac-address>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 2/1</brcd:interface-id>
    <brcd:link-state>
     <brcd:down></brcd:down>
    </brcd:link-state>
    <brcd:l2-state>
     <brcd:disabled></brcd:disabled>
    </brcd:l2-state>
    <brcd:duplex>
     <brcd:none></brcd:none>
```

```
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.81b1</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 2/2</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.81b2</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>loopback 1</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
  <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:yes></brcd:yes>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>N/A</brcd:mac-address>
</brcd:interface>
</brcd:interface-statedata>
</netiron-statedata>
</nc:data>
</nc:rpc-reply>
]]>]]>
```

The following example shows the RPC request and response of a <get> operation with filter for interface statedata and specific interfaces.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:get>
<nc:filter >
<brcd:netiron-statedata>
<brcd:interface-statedata>
<brcd:interface>
<brcd:interface-id>ethernet 1/1</brcd:interface-id>
</brcd:interface>
<brcd:interface>
<brcd:interface-id>ethernet 1/6</brcd:interface-id>
</brcd:interface>
</brcd:interface-statedata>
</brcd:netiron-statedata>
</nc:filter>
</nc:get>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:data>
<netiron-statedata xmlns="http://www.brocade.com/netconf/config/netiron-config/">
<brcd:interface-statedata>
<brcd:interface>
  <brcd:interface-id>ethernet 1/1</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8181</brcd:mac-address>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/6</brcd:interface-id>
  <brcd:link-state>
   <brcd:down></brcd:down>
  </brcd:link-state>
  <brcd:l2-state>
   <brcd:disabled></brcd:disabled>
  </brcd:l2-state>
  <brcd:duplex>
   <brcd:none></brcd:none>
  </brcd:duplex>
  <brcd:speed></brcd:speed>
```

```
  <brcd:tag-mode>
   <brcd:no></brcd:no>
  </brcd:tag-mode>
  <brcd:priority-level>
   <brcd:level0></brcd:level0>
  </brcd:priority-level>
  <brcd:mac-address>748e.f811.8186</brcd:mac-address>
</brcd:interface>
</brcd:interface-statedata>
</netiron-statedata>
</nc:data>
</nc:rpc-reply>
]]>]]>
```

<get-config> operation example

The following example shows the RPC request and response of a <get-config> operation.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="1">
<nc:get-config>
<nc:source>
<nc:running/>
</nc:source>
<nc:filter nc:type="subtree">
<brcd:netiron-config>
<brcd:interface-config/>
</brcd:netiron-config>
</nc:filter>
</nc:get-config>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="1">
<nc:data>
<brcd:netiron-config>
<brcd:interface-config>
<brcd:interface>
  <brcd:interface-id>management 1</brcd:interface-id>
  <brcd:enable></brcd:enable>
  <brcd:ip>
   <brcd:address>10.20.99.187/20</brcd:address>
  </brcd:ip>
  <brcd:ipv6>
   <brcd:address>2001::10:20:99:187/64</brcd:address>
  </brcd:ipv6>
  <brcd:priority>
  </brcd:priority>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 1/1</brcd:interface-id>
  <brcd:disable></brcd:disable>
  <brcd:loop-detection>
  </brcd:loop-detection>
  <brcd:flow-control></brcd:flow-control>
  <brcd:speed-duplex>auto</brcd:speed-duplex>
  <brcd:priority>
```

```
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/2</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/3</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/4</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/5</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/6</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/7</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
```

```
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/8</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/9</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/10</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/11</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/12</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/13</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
```

```
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/14</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/15</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/16</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/17</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/18</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/19</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
```

```
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/20</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/21</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/22</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/23</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 1/24</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:speed-duplex>auto</brcd:speed-duplex>
    <brcd:priority>
    </brcd:priority>
  </brcd:interface>
  <brcd:interface>
    <brcd:interface-id>ethernet 2/1</brcd:interface-id>
    <brcd:disable></brcd:disable>
    <brcd:loop-detection>
    </brcd:loop-detection>
    <brcd:flow-control></brcd:flow-control>
    <brcd:priority>
    </brcd:priority>
```

```
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>ethernet 2/2</brcd:interface-id>
  <brcd:disable></brcd:disable>
  <brcd:loop-detection>
  </brcd:loop-detection>
  <brcd:flow-control></brcd:flow-control>
  <brcd:priority>
  </brcd:priority>
</brcd:interface>
<brcd:interface>
  <brcd:interface-id>loopback 1</brcd:interface-id>
  <brcd:enable></brcd:enable>
  <brcd:ip>
   <brcd:address>172.13.32.1/32</brcd:address>
  </brcd:ip>
</brcd:interface>
</brcd:interface-config>
</brcd:netiron-config>
</nc:data>
</nc:rpc-reply>
]]>]]>
```

Chapter

4

MPLS Configuration

In this chapter

- Overview 59
- MPLS configuration elements 59
- MPLS state elements 68
- Full examples 74

Overview

This chapter describes how to configure Multiprotocol Label Switching (MPLS) using the YANG model, and illustrates the equivalent XML and CLI.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

MPLS configuration elements

Each element of the MPLS corresponds to the following:

- A YANG model description
- An XML description
- A CLI description

The following subsections compare a specific example for each MPLS configuration element.

MPLS priority

This data element specifies the setup priority and the hold priority.

TABLE 14 MPLS priority

YANG model	XML	CLI

grouping mpls-priority-grouping
{
description "MPLS Setup and hold priority group";
container mpls-priority
{
description "Setup and hold priority";
 	
leaf setup-priority
{
type uint32
{
range "0..7";
}
mandatory "true";
description "Setup priority";
}
 	
leaf hold-priority
{
type uint32
{
range "0..7";
}
mandatory "true";
description "Hold priority";
}
}
}
	<*brcd:mpls-priority*>
	<*brcd:setup-priority*>7</*brcd:setup-priority*>
	</*brcd:mpls-priority*>
	
	<*brcd:mpls-priority*>
	<*brcd:hold-priority*>7</*brcd:hold-priority*>
	</*brcd:mpls-priority*>
		[**no**] **priority** <*setup-priority*>
		
		[**no**] **priority** <*hold-priority*>

MPLS traffic engineering

The following data element specifies the traffic engineering parameters.

TABLE 15 MPLS traffic engineering

YANG model	XML	CLI

grouping mpls-traffic-eng-grouping
{
description "MPLS Traffic Engineering parameters group";
 	
container mpls-traffic-eng
{
description "MPLS Traffic Engineering parameters";
leaf max-burst
{
type uint32
{
range "0..2147483647";
}
units "Bytes";
description "Maximum burst";
}
	<*brcd:mpls-traffic-eng*>
	<*brcd:max-burst*>214748364</*brcd:max-burst*>
	</*brcd:mpls-traffic-eng*>
		[no] **traffic-eng max-burst** <*bytes*>

leaf max-rate
{
type uint32
{
range "0..2147483647";
}
units "Kilobits per second";
description "Maximum traffic rate";
}
	<*brcd:mpls-traffic-eng*>
	<*brcd:max-rate*>214748364</*brcd:max-rate*>
	</*brcd:mpls-traffic-eng*>
		[no] **traffic-eng max-rate** <*rate*>

leaf mean-rate
{
type uint32
{
range "0..2147483647";
}
units "Kilobits per seconds";
description "Mean traffic rate";
}
 	
}
 	
}
	<*brcd:mpls-traffic-eng*>
	<*brcd:mean-rate*>4500</*brcd:mean-rate*>
	</*brcd:mpls-traffic-eng*>
		[no] **traffic-eng mean-rate** <*rate*>

MPLS path

The following data element creates a path and inserts nodes into the path.

TABLE 16 MPLS path

YANG model	XML	CLI
grouping mpls-path-grouping { description "MPLS Explicit Path configuration group"; list path { key "name"; description "Explicit route path";		

leaf name
{
type mpls-path-type;
description "Path Name";
}
	<*brcd:path*>
	<*brcd:name*>example</*brcd:name*>
		[**no**] **path** <*path name*>

leaf-list strict
{
type ipv4_address;
description "Strict node";
}
	<*brcd:strict*>1.1.1.1</*brcd:strict*>	[**no**] **strict** <*ip address*>

leaf-list loose
{
type ipv4_address;
description "Loose node";
}
	<*brcd:loose*>2.2.2.2</*brcd:loose*>	[**no**] **loose** <*ip address*>

list insert-strict
{
key "addr before";
description "Insert Strict nodes";
leaf addr
{
type ipv4_address;
description "Strict node";
}
leaf before
{
type ipv4_address;
description "Insert before this node";
}
}
	<*brcd:insert-strict*>
	<*brcd:addr*>3.3.3.3</*brcd:addr*>
	<*brcd:before*>2.2.2.2</*brcd:before*>
	</*brcd:insert-strict*>
		[**no**] **insert strict** <*ip address*> **before** <*ip address*>

TABLE 16 MPLS path (Continued)

YANG model	XML	CLI

list insert-loose
{
key "addr before";
description "Insert Loose nodes";
leaf addr
{
type ipv4_address;
description "Loose node";
}
leaf before
{
type ipv4_address;
description "Insert before this node";
}
}
}
}
	<*brcd:insert-loose*>
	<*brcd:addr*>4.4.4.4</*brcd:addr*>
	<*brcd:before*>1.1.1.1</*brcd:before*>
	</*brcd:insert-loose*>
	</*brcd:path*>
		[**no**] **insert loose** <*ip address*>
		before <*ip address*>

MPLS LSP configuration

The following data element configures the Label Switch Path (LSP).

TABLE 17 MPLS LSP

YANG model	XML	CLI

grouping mpls-lsp-grouping
{
description "MPLS LSP configuration group";
list lsp
{
key "name";
description "MPLS Signaled label switched
"+"path configuration";
leaf name
{
type string
{
length "1..64";
}
description "LSP name";
}
leaf adaptive
{
type empty;
description "Enable LSP to be modified
on fly";
}
	<*brcd:lsp*>
	<*brcd:name*>example</*brcd:name*>
		[**no**] **lsp** <*name*>

TABLE 17 MPLS LSP (Continued)

YANG model	XML	CLI

leaf commit
{
type empty;
description "Apply the parameter modifications to LSP";
}
	<*brcd:commit*></*brcd:commit*>	[**no**] **commit**

choice admin-status-choice
{
description "Enable/Tear down the LSP";
 	
case enable
{
leaf enable
{
type empty;
description "Enable the LSP";
}
}
case disable
{
leaf disable
{
type empty;
description "Tear down the LSP";
}
}
}
	<*brcd:enable*></*brcd:enable*>
	
	<*brcd:disable*></*brcd:disable*>
		[**no**] **enable**
		
		[**no**] **disable**

leaf from
{
type ipv4_address;
description "Ingress router of the LSP";
}
	<*brcd:from*>5.5.5.5</*brcd:from*>	**from** <*ip address*>

leaf to
{
type ipv4_address;
description "Egress router of the LSP";
}
	<*brcd:to*>6.6.6.6</*brcd:to*>	**to** <*ip address*>

leaf hop-limit
{
type hop-limit-type;
description "Limit of hops the LSP can traverse";
}
	<*brcd:hop-limit*>20</*brcd:hop-limit*>	[**no**] **hop-limit** <*number*>

TABLE 17 MPLS LSP (Continued)

YANG model	XML	CLI

leaf ipmtu
{
type uint32;
{
range "576..9194";
}
description "IP Packet MTU size";
}
NOTE: The maximum value varies depending on the default max-frame-size configured on the device.
	<*brcd:ipmtu*>1526</*brcd:ipmtu*>	[**no**] **ipmtu** <*packet-size*>

leaf ldp-tunneling
{
type empty;
description " Enable LDP over RSVP tunneling";
}
	<*brcd:ldp-tunneling*></*brcd:ldp-tunneling*>	[**no**] **ldp-tunneling**

leaf metric
{
type uint32
{
range "1..65535";
}
description "LSP metric";
}
	<*brcd:metric*>600</*brcd:metric*>	[**no**] **metric** <*number*>

leaf primary-path
{
type mpls-path-type;
description "LSP Primary explicit path";
}
uses mpls-priority-grouping;
	<*brcd:primary-path*>example</*brcd:primary-path*>	[**no**] **primary-path** <*path name*>

leaf record
{
type empty;
description "Enable recording path routes";
}
	<*brcd:record*></*brcd:record*>	[**no**] **record**

leaf reoptimize-timer
{
type uint32
{
range "300..65535";
}
units "seconds";
description "Reoptimize timer value";
}
	<*brcd:reoptimize-timer*>3600</*brcd:reoptimize-timer*>	[**no**] **reoptimize_timer** <*seconds*>

TABLE 17 MPLS LSP (Continued)

YANG model	XML	CLI

leaf revert-timer
{
type uint32;
{
range "1..65535";
}
units "seconds";
description "LSP revert timer";
}
uses mpls-traffic-eng-grouping;
 	
list secondary-path
{
key "name";
description "Secondary Path configuration";
leaf name
{
type mpls-path-type;
description "Name of the Secondary Path";
}
}
}
}
	<*brcd:revert-timer*>2999</*brcd:revert-timer*>
	<*brcd:secondary-path*>
	<*brcd:name*>example2</*brcd:name*>
	</*brcd:secondary-path*>
	</*brcd:lsp*>
		[**no**] **revert-time** <*timer-value*>
		[**no**] **secondary-path** <*path name*>

MPLS RSVP configuration

The following data element configures the Resource ReSerVation Protocol (RSVP) parameters.

TABLE 18 MPLS RSVP configuration

YANG model	XML	CLI

grouping mpls-rsvp-grouping
{
description "MPLS RSVP configuration group";
container rsvp
{
description "MPLS RSVP configuration";
leaf refresh-interval
{
type uint32
{
range "1..360";
}
units "seconds";
description "Avg. interval between " + "refresh path and resv msgs";
}
	<*brcd:rsvp*>
	<*brcd:refresh-interval*>360</*brcd:refresh-interval*>
	</*brcd:rsvp*>
		[**no**] **refresh-interval** <*seconds*>

leaf refresh-multiple
{
type uint32
{
range "1..255";
}
description "Num of unresponded path " + "or resv before time out";
}
 	
}
}
	<*brcd:rsvp*>
	<*brcd:refresh-multiple*>255</*brcd:refresh-multiple*>
	</*brcd:rsvp*>
		[**no**] **refresh-multiple** <*intervals*>

Router MPLS configuration

The following data element contains the MPLS configuration.

TABLE 19 Router MPLS configuration

YANG model	XML	CLI

grouping mpls-config-grouping
{
description "Router MPLS configuration group";
container mpls-config
{
description "Router MPLS configuration";
uses mpls-path-grouping;
uses mpls-lsp-grouping;
uses mpls-rsvp-grouping;
}
}
	<brcd:mpls-config>
	</brcd:mpls-config>
		[**no**] **router mpls**

MPLS state elements

The following is the YANG model for the MPLS state data elements.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule mpls-statedata
{
    belongs-to netiron-statedata
    {
        prefix "statedata";
    }

    import common-defs
    {
        prefix "common";
    }

    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the device Router MPLS state data. VERSION:
";

    revision 2011-04-20
    {
        description "Initial revision";
    }

    grouping per-priority-bandwidth-grouping
    {
```

```
    description "Bandwidth(kbps) per priority (0-7)";
    list priority-bandwidth
    {
        key "priority";
        description "Bandwidth(kbps) for a priority";
        leaf priority
        {
            type common:priority-value;
            description "Priority value";
        }
        leaf bandwidth
        {
            type uint32
            {
                range "0..4294967295";
            }
            description "Bandwidth(kbps)";
        }
    }

}

grouping mpls-lsp-statedata-grouping
{
    description "Router MPLS LSP statedata grouping";
    container mpls-lsp-statedata
    {
        description "Router MPLS LSP statedata";
        list lsp
        {
            key "name";
            description "MPLS Signaled label switched "+
                 "path configuration";

            leaf name
            {
                type string
                {
                    length "1..64";
                }
                description "LSP name";
            }

            leaf to
            {
                type common:ipv4_address;
                description "Egress LER for the LSP";
            }


            choice admin-state
            {
                description "The administrative state of the LSP";

                case up
                {
                    leaf up
                    {
                        type empty;
                        description "Up";
```

```
        }
    }
    case down
    {
        leaf down
        {
            type empty;
            description "Down";
        }
    }
}

choice oper-state
{
    description "The operational state of the LSP";

    case up
    {
        leaf up
        {
            type empty;
            description "Up";
        }
    }
    case up-star
    {
        leaf up-star
        {
            type empty;
            description "Up*";
        }
    }
    case down
    {
        leaf down
        {
            type empty;
            description "Down";
        }
    }
}

leaf tunnel-intf
{
    type string
    {
        length "1..16";
    }
    description "The tunnel interface";
}

leaf up-dn-times
{
    type uint16
    {
        range "0..65535";
    }
    description "The number of times the operational state"+
    " of the LSP's primary path has transitioned from DOWN"+
    " to UP";
```

```
                    }

                    leaf retry-no
                    {
                        type uint16
                        {
                            range "0..65535";
                        }
                        description "The number of attempts the ingress LER has"+
                                    " made to connect to the egress LER";
                    }

                    leaf active-path
                    {
                        type string
                        {
                            length "1..64";
                        }
                        description "The path currently in use for this LSP.";
                    }

                }

            }

        }

        grouping mpls-interface-statedata-grouping
        {
            description "Router MPLS interface statedata grouping";
            container mpls-interface-statedata
            {
                description "Router MPLS interface state data";
                list interface
                {
                    key interface-id;
                    description "List of MPLS-enabled interfaces -- state data";

                    leaf interface-id
                    {
                        type common:interface-all-types;
                        description "The interface identifier including chassis slot
and port number";
                    }

                    choice admin-state
                    {
                        description "The administrative state of the interface";

                        case up
                        {
                            leaf up
                            {
                                type empty;
                                description "Up";
                            }
                        }
                        case down
                        {
                            leaf down
```

```
        {
            type empty;
            description "Down";
        }
    }
}

choice oper-state
{
    description "The operational state of the interface";

    case up
    {
        leaf up
        {
            type empty;
            description "Up";
        }
    }
    case down
    {
        leaf down
        {
            type empty;
            description "Down";
        }
    }
}

leaf maximum-bw
{
    type uint32
    {
        range "0..4294967295";
    }
    description "The maximum outbound bandwidth that can"+
    " be used on the interface.";
}

leaf maximum-reservable-bw
{
    type uint32
    {
        range "0..4294967295";
    }
    description "The maximum bandwidth that can be"+
    " reserved on the interface.";
}

leaf admin-groups-bitmap
{
    type uint32
    {
        range "0..4294967295";
    }
    description "The bitfield indicating the"+
    " administrative groups to which this interface belongs.";
}

container per-priority-reservable-bw
```

```
                {
                    description "The amount of bandwidth not yet reserved on"
                    +"the interface per priority (0 to 7)";
                    uses per-priority-bandwidth-grouping;
                }

                container per-priority-last-sent-reservable-bw
                {
                    description "The values in the Unreserved Bandwidth TLV"
                    +" sent in the most recent OSPF-TE LSA.";
                    uses per-priority-bandwidth-grouping;
                }

                container protecting-bypass-lsp
                {
                    description "The name of any bypass LSPs that are"
                    +"protecting this interface.";
                    leaf-list lsp
                    {
                        type string
                        {
                            length "1..64";
                        }
                        description "bypass LSP name";
                    }
                }

            }

        }

    grouping mpls-statedata-grouping
    {
        description "Router MPLS state data group";
        container mpls-statedata
        {
            description "Router MPLS state data";
            uses mpls-lsp-statedata-grouping;
            uses mpls-interface-statedata-grouping;

        }

    }

}
```

Full examples

The following shows the complete YANG model for MPLS configuration, as well as example RPC requests and responses.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule mpls-config
{
    belongs-to netiron-config
    {
        prefix brcd;
    }

    import common-defs
    {
        prefix "common";
    }


    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
             "URL:  www.brocade.com";

    description "This module defines the Router MPLS configuration. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }

    typedef hop-limit-type
    {
        type uint32
        {
            range "0..255";
        }
        description "Type for Limit of hops";
    }

    typedef hop-limit-type
    {
        type uint32
        {
            range "0..255";
        }
        description "Type for Limit of hops";
    }

    typedef mpls-path-type
```

```
{
    type string
    {
        length "1..64";
    }
    description "Path Name";
}

grouping mpls-priority-grouping
{
    description "MPLS Setup and hold priority group";
    container mpls-priority
    {
        description "Setup and hold priority";

        leaf setup-priority
        {
            type uint32
            {
                range "0..7";
            }
            mandatory "true";
            description "Setup priority";
        }

        leaf hold-priority
        {
            type uint32
            {
                range "0..7";
            }
            mandatory "true";
            description "Hold priority";
        }
    }

}

grouping mpls-traffic-eng-grouping
{
    description "MPLS Traffic Engineering parameters group";

    container mpls-traffic-eng
    {
        description "MPLS Traffic Engineering parameters";

        leaf max-burst
        {
            type uint32
            {
                range "0..2147483647";
            }
            units "Bytes";
            description "Maximum burst";
        }

        leaf max-rate
        {
            type uint32
            {
```

```
                    range "0..2147483647";
                }
                units "Kilobits per second";
                description "Maximum traffic rate";
            }

            leaf mean-rate
            {
                type uint32
                {
                    range "0..2147483647";
                }
                units "Kilobits per seconds";
                description "Mean traffic rate";
            }

        }

    }

    grouping mpls-path-grouping
    {
        description "MPLS Explicit Path configuration group";
        list path
        {
            key "name";
            description "Explicit route path";
            leaf name
            {
                type mpls-path-type;
                description "Path Name";
            }
            leaf-list strict
            {
                type ipv4_address;
                description "Strict node";
            }

            leaf-list loose
            {
                type ipv4_address;
                description "Loose node";
            }

            list insert-strict
            {
                key "addr before";
                description "Insert Strict nodes";
                leaf addr
       {
            type ipv4_address;
            description "Strict node";
       }
       leaf before
       {
            type ipv4_address;
           description "Insert before this node";
       }
            }
```

```
        list insert-loose
        {
            key "addr before";
            description "Insert Loose nodes";
            leaf addr
   {
       type ipv4_address;
       description "Loose node";
   }
   leaf before
   {
       type ipv4_address;
      description "Insert before this node";
   }
        }

    }

}

grouping mpls-lsp-grouping
{
    description "MPLS LSP configuration group";
    list lsp
    {
        key "name";
        description "MPLS Signaled label switched "+
                 "path configuration";

        leaf name
        {
            type string
            {
                length "1..64";
            }
            description "LSP name";
        }

        leaf adaptive
        {
            type empty;
            description "Enable LSP to be modified on fly";
        }

        leaf commit
        {
            type empty;
            description "Apply the parameter modifications to LSP";
        }

        choice admin-status-choice
        {
            description "Enable/Tear down the LSP";

            case enable
            {
                leaf enable
                {
                    type empty;
                    description "Enable the LSP";
```

```
        }
    }
    case disable
    {
        leaf disable
        {
            type empty;
            description "Tear down the LSP";
        }
    }
}

leaf from
{
    type ipv4_address;
    description "Ingress router of the LSP";
}

leaf to
{
    type ipv4_address;
    description "Egress router of the LSP";
}

leaf hop-limit
{
    type hop-limit-type;
    description "Limit of hops the LSP can traverse";
}

leaf ipmtu
{
    type uint32;
    description "IP Packet MTU size";
}

leaf ldp-tunneling
{
    type empty;
    description " Enable LDP over RSVP tunneling";
}

leaf metric
{
    type uint32
    {
        range "1..65535";
    }
    description "LSP metric";
}

leaf primary-path
{
    type mpls-path-type;
    description "LSP Primary explicit path";
}

uses mpls-priority-grouping;

leaf record
```

```
        {
            type empty;
            description "Enable recording path routes";
        }

        leaf reoptimize-timer
        {
            type uint32
            {
                range "300..65535";
            }
            units "seconds";
            description "Reoptimize timer value";
        }

        leaf revert-timer
        {
            type uint32;
            units "seconds";
            description "LSP revert timer";
        }

        uses mpls-traffic-eng-grouping;

        list secondary-path
        {
            key "name";
            description "Secondary Path configuration";
            leaf name
            {
                type mpls-path-type;
                description "Name of the Secondary Path";
            }
        }


    }

}

grouping mpls-rsvp-grouping
{
    description "MPLS RSVP configuration group";
    container rsvp
    {
        description "MPLS RSVP configuration";

        leaf refresh-interval
        {
            type uint32
            {
                range "1..360";
            }
            units "seconds";
            description "Avg. interval between " +
                    "refresh path and resv msgs";
        }

        leaf refresh-multiple
        {
```

```
                type uint32
                {
                   range "1..255";
                }
                description "Num of unresponded path " +
                       "or resv before time out";
            }

        }

    }

    grouping mpls-config-grouping
    {
        description "Router MPLS configuration group";
        container mpls-config
        {
            description "Router MPLS configuration";
            uses mpls-path-grouping;
            uses mpls-lsp-grouping;
            uses mpls-rsvp-grouping;

        }

    }

}
```

The following example shows the RPC request and response of an <edit-config> operation.

```
<nc:rpc message-id="1" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/">
<nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<brcd:netiron-config>
<brcd:mpls-config>
<brcd:path>
<brcd:name>example</brcd:name>
<brcd:strict>1.1.1.1</brcd:strict>
<brcd:loose>2.2.2.2</brcd:loose>
<brcd:insert-strict>
<brcd:addr>3.3.3.3</brcd:addr>
<brcd:before>2.2.2.2</brcd:before>
</brcd:insert-strict>
<brcd:insert-loose>
<brcd:addr>4.4.4.4</brcd:addr>
<brcd:before>1.1.1.1</brcd:before>
</brcd:insert-loose>
</brcd:path>
<brcd:path>
<brcd:name>example2</brcd:name>
<brcd:strict>10.10.10.10</brcd:strict>
</brcd:path>
<brcd:lsp>
<brcd:name>example</brcd:name>
<brcd:adaptive></brcd:adaptive>
```

```
<brcd:from>5.5.5.5</brcd:from>
<brcd:to>6.6.6.6</brcd:to>
<brcd:primary-path>example</brcd:primary-path>
<brcd:hop-limit>20</brcd:hop-limit>
<brcd:ipmtu>1526</brcd:ipmtu>
<brcd:ldp-tunneling></brcd:ldp-tunneling>
<brcd:metric>600</brcd:metric>
<brcd:record></brcd:record>
<brcd:reoptimize-timer>3600</brcd:reoptimize-timer>
<brcd:revert-timer>2999</brcd:revert-timer>
<brcd:secondary-path>
<brcd:name>example2</brcd:name>
</brcd:secondary-path>
<brcd:mpls-priority>
<brcd:hold-priority>6</brcd:hold-priority>
<brcd:setup-priority>6</brcd:setup-priority>
</brcd:mpls-priority>
<brcd:mpls-traffic-eng>
<brcd:max-burst>214748364</brcd:max-burst>
<brcd:max-rate>214748364</brcd:max-rate>
<brcd:mean-rate>4500</brcd:mean-rate>
</brcd:mpls-traffic-eng>
<brcd:enable></brcd:enable>
</brcd:lsp>
<brcd:rsvp>
<brcd:refresh-interval>360</brcd:refresh-interval>
<brcd:refresh-multiple>255</brcd:refresh-multiple>
</brcd:rsvp>
</brcd:mpls-config>
</brcd:netiron-config>
</nc:config>
</nc:edit-config>
</nc:rpc>
]]>]]>
```

The following example shows the complete CLI for an MLPS configuration.

```
Brocade# show run
Current configuration:
!
ver V5.2.0T183
!
!

no spanning-tree
!
!
vlan 1 name DEFAULT-VLAN
!
vlan 4089
 untagged ethe 2/1 to 2/2
!
!
system-max vlan 4095
system-max virtual-interface 4095
!
aaa authentication enable default radius tacacs+ local
aaa authentication login default local radius tacacs+
aaa authentication login privilege-mode
```

```
aaa authorization exec default  tacacs+ radius
aaa accounting commands 0 default start-stop  radius
aaa accounting exec default start-stop  radius
aaa accounting system default start-stop  radius
radius-server host 10.20.99.135 auth-port 1812 acct-port 1813 default key 0 rxperf
tacacs-server host 10.20.99.135 auth-port 49 default key 2 $VUNTblVH
sntp server 10.20.99.135 4
sntp server 10.20.99.130 4
enable snmp config-radius
enable aaa console
console timeout 5
logging host 10.20.99.136
logging console
username brocade password .....
username nclocalnormal privilege 5 password .....
username nclocalsuper password .....
ip route 0.0.0.0/0 10.20.96.1
!
!
!
!
!
snmp-server
snmp-server community ..... ro
fdp run
!
netconf server
!
router isis
 net 56.0000.0001.0209.9187.00
 is-type level-2
 address-family ipv4 unicast
  metric-style wide
 exit-address-family

 address-family ipv6 unicast
  no adjacency-check
 exit-address-family

!
!
!
!
!
!
interface loopback 1
 ip address 172.13.32.1/32
!
!
interface management 1
 ipv6 address 2001::10:20:99:187/64
 ip address 10.20.99.187/20
 enable
!
interface ethernet 1/3
 ip router isis
 ip address 1.1.3.1/24
 isis circuit-type level-2
 isis hello-interval 1
 isis metric 110
```

```
 isis point-to-point
!
interface ethernet 2/1
 enable
!
interface ethernet 2/2
 enable
!
!
!
router mpls
 policy
  traffic-eng isis level-2
  retry-time 300
  no rapid-retry


 rsvp
  refresh-interval 360
  refresh-multiple 255

 mpls-interface e1/3
  rsvp-reliable-messaging  rapid-retrans-interval 2000  rapid-retry-limit 5
  rsvp-refresh-reduction bundle-messages
  rsvp-refresh-reduction summary-refresh
  reservable-bandwidth percentage 80


 path example
  strict 1.1.1.1
  strict 3.3.3.3
  loose 4.4.4.4
  loose 2.2.2.2

 path example2
  strict 2.2.2.2
  strict 10.10.10.10


 lsp example
  to 6.6.6.6
  from 5.5.5.5
  primary example
  priority 6 6
  traffic-eng max-rate 214748364 mean-rate 4500 max-burst 214748364
  metric 600
  hop-limit 20
  adaptive
  secondary example2
  revert-timer 2999
  ldp-tunneling
  enable
```

```
!
!
!
!
!
end
```

<get> operation examples

The following example shows the RPC request and response of a <get> operation for a single LSP.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:get>
<nc:filter nc:type="subtree" >
<brcd:netiron-statedata>
<brcd:mpls-statedata>
<brcd:mpls-lsp-statedata>
<brcd:name>examplelsp1</brcd:name>
</brcd:mpls-lsp-statedata>
</brcd:mpls-statedata>
</brcd:netiron-statedata>
</nc:filter>
</nc:get>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:data>
<netiron-statedata xmlns="http://www.brocade.com/netconf/config/netiron-config/">
<brcd:mpls-statedata>
<brcd:mpls-lsp-statedata>
  <brcd:name>examplelsp1</brcd:name>
  <brcd:to>1.99.161.1</brcd:to>
  <brcd:admin-state>
   <brcd:up></brcd:up>
  </brcd:admin-state>
  <brcd:oper-state>
   <brcd:down></brcd:down>
  </brcd:oper-state>
  <brcd:tunnel-intf>tnl1</brcd:tunnel-intf>
  <brcd:up-dn-times>0</brcd:up-dn-times>
  <brcd:retry-no>14</brcd:retry-no>
</brcd:mpls-lsp-statedata>
</brcd:mpls-statedata>
</netiron-statedata>
</nc:data>
</nc:rpc-reply>
]]>]]>
```

The following example shows the RPC request and response of a <get> operation for a list of LSPs.

```
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="1">
<nc:ok></nc:ok>
</nc:rpc-reply>
```

```
]]>]]>
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:get>
<nc:filter nc:type="subtree" >
<brcd:netiron-statedata>
<brcd:mpls-statedata/>
</brcd:netiron-statedata>
</nc:filter>
</nc:get>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="25">
<nc:data>
<netiron-statedata xmlns="http://www.brocade.com/netconf/config/netiron-config/">
<brcd:mpls-statedata>
<brcd:mpls-lsp-statedata>
  <brcd:name>examplelsp1</brcd:name>
  <brcd:to>1.99.161.1</brcd:to>
  <brcd:admin-state>
   <brcd:up></brcd:up>
  </brcd:admin-state>
  <brcd:oper-state>
   <brcd:down></brcd:down>
  </brcd:oper-state>
  <brcd:tunnel-intf>tnl1</brcd:tunnel-intf>
  <brcd:up-dn-times>0</brcd:up-dn-times>
  <brcd:retry-no>13</brcd:retry-no>
</brcd:mpls-lsp-statedata>
<brcd:mpls-lsp-statedata>
  <brcd:name>examplelsp2</brcd:name>
  <brcd:to>1.99.145.1</brcd:to>
  <brcd:admin-state>
   <brcd:up></brcd:up>
  </brcd:admin-state>
  <brcd:oper-state>
   <brcd:down></brcd:down>
  </brcd:oper-state>
  <brcd:tunnel-intf>tnl2</brcd:tunnel-intf>
  <brcd:up-dn-times>0</brcd:up-dn-times>
  <brcd:retry-no>0</brcd:retry-no>
</brcd:mpls-lsp-statedata>
<brcd:mpls-interface-statedata>
  <brcd:interface-id>ve 100</brcd:interface-id>
  <brcd:admin-state>
   <brcd:down></brcd:down>
  </brcd:admin-state>
  <brcd:oper-state>
   <brcd:down></brcd:down>
  </brcd:oper-state>
  <brcd:maximum-bw>0</brcd:maximum-bw>
  <brcd:maximum-reservable-bw>0</brcd:maximum-reservable-bw>
  <brcd:admin-group-bitmap>0</brcd:admin-group-bitmap>
  <brcd:per-priority-reservable-bw>
   <brcd:priority-bandwidth>
    <brcd:priority>0</brcd:priority>
    <brcd:bandwidth>0</brcd:bandwidth>
```

```
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>1</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>2</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>3</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>4</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>5</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>6</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>7</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
   </brcd:per-priority-reservable-bw>
  <brcd:per-priority-last-sent-reservable-bw>
    <brcd:priority-bandwidth>
     <brcd:priority>0</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>1</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>2</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>3</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>4</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>5</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
    <brcd:priority-bandwidth>
     <brcd:priority>6</brcd:priority>
     <brcd:bandwidth>0</brcd:bandwidth>
    </brcd:priority-bandwidth>
   <brcd:priority-bandwidth>
```

```
    <brcd:priority>7</brcd:priority>
    <brcd:bandwidth>0</brcd:bandwidth>
   </brcd:priority-bandwidth>
  </brcd:per-priority-last-sent-reservable-bw>
  <brcd:protecting-bypass-lsp>
  </brcd:protecting-bypass-lsp>
</brcd:mpls-interface-statedata>
</brcd:mpls-statedata>
</netiron-statedata>
</nc:data>
</nc:rpc-reply>
]]>]]>
```

<get-config> operation example

The following example shows the RPC request and response of a <get-config> operation.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="1">
<nc:get-config>
<nc:source>
<nc:running/>
</nc:source>
<nc:filter nc:type="subtree">
<brcd:netiron-config>
<brcd:mpls-config/>
</brcd:netiron-config>
</nc:filter>
</nc:get-config>
</nc:rpc>
]]>]]>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/"
message-id="1">
<nc:data>
<brcd:netiron-config>
<brcd:mpls-config>
<brcd:path>
  <brcd:name>Pri_to_mlx4_2</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
  <brcd:strict>1.99.142.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Pri_to_mlx4_3</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Pri_to_mlx4_4</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Pri_to_mlx4_5</brcd:name>
  <brcd:strict>1.99.145.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Pri_to_mlx8_1</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
  <brcd:strict>1.99.141.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Pri_to_u40</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
  <brcd:strict>1.99.142.1</brcd:strict>
  <brcd:strict>1.99.30.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Pri_to_u41</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
  <brcd:strict>1.99.142.1</brcd:strict>
  <brcd:strict>1.99.40.1</brcd:strict>
</brcd:path>
<brcd:path>
```

```
  <brcd:name>Sec_to_mlx4_2</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
  <brcd:strict>1.99.141.1</brcd:strict>
  <brcd:strict>1.99.142.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Sec_to_mlx4_3</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
  <brcd:strict>1.99.143.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Sec_to_mlx4_4</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
  <brcd:strict>1.99.144.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Sec_to_mlx4_5</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
  <brcd:strict>1.99.145.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Sec_to_mlx8_1</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
  <brcd:strict>1.99.142.1</brcd:strict>
  <brcd:strict>1.99.141.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Sec_to_u40</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
  <brcd:strict>1.99.141.1</brcd:strict>
  <brcd:strict>1.99.30.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Sec_to_u41</brcd:name>
  <brcd:strict>1.99.143.1</brcd:strict>
 <brcd:strict>1.99.141.1</brcd:strict>
  <brcd:strict>1.99.40.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Ter_to_mlx4_2</brcd:name>
  <brcd:strict>1.99.40.1</brcd:strict>
  <brcd:strict>1.99.142.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Ter_to_mlx4_3</brcd:name>
  <brcd:strict>1.99.145.1</brcd:strict>
  <brcd:strict>1.99.143.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Ter_to_mlx4_4</brcd:name>
  <brcd:strict>1.99.145.1</brcd:strict>
  <brcd:strict>1.99.144.11</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Ter_to_mlx4_5</brcd:name>
  <brcd:strict>1.99.144.1</brcd:strict>
  <brcd:strict>1.99.145.1</brcd:strict>
</brcd:path>
<brcd:path>
  <brcd:name>Ter_to_mlx8_1</brcd:name>
```

```
    <brcd:strict>1.99.30.1</brcd:strict>
    <brcd:strict>1.99.141.1</brcd:strict>
  </brcd:path>
  <brcd:path>
    <brcd:name>Ter_to_u40</brcd:name>
    <brcd:strict>1.99.30.1</brcd:strict>
  </brcd:path>
  <brcd:path>
    <brcd:name>Ter_to_u41</brcd:name>
    <brcd:strict>1.99.40.1</brcd:strict>
  </brcd:path>
  <brcd:lsp>
    <brcd:name>scriptlsp1008</brcd:name>
    <brcd:adaptive></brcd:adaptive>
    <brcd:from>161.0.0.113</brcd:from>
    <brcd:to>81.0.0.113</brcd:to>
    <brcd:disable></brcd:disable>
    <brcd:hop-limit>10</brcd:hop-limit>
    <brcd:ipmtu>1526</brcd:ipmtu>
    <brcd:ldp-tunneling></brcd:ldp-tunneling>
    <brcd:metric>600</brcd:metric>
    <brcd:primary-path>Pri_to_mlx8_1</brcd:primary-path>
    <brcd:record></brcd:record>
    <brcd:reoptimize-timer>3600</brcd:reoptimize-timer>
    <brcd:revert-timer>43200</brcd:revert-timer>
    <brcd:mpls-traffic-eng>
     <brcd:max-burst>44736</brcd:max-burst>
     <brcd:max-rate>6312</brcd:max-rate>
     <brcd:mean-rate>1544</brcd:mean-rate>
    </brcd:mpls-traffic-eng>
    <brcd:secondary-path>
     <brcd:name>Ter_to_mlx8_1</brcd:name>
    </brcd:secondary-path>
    <brcd:secondary-path>
     <brcd:name>Sec_to_mlx8_1</brcd:name>
    </brcd:secondary-path>
  </brcd:lsp>
  </brcd:mpls-config>
  </brcd:netiron-config>
  </nc:data>
  </nc:rpc-reply>
  ]]>]]>
```

Chapter

VLAN Configuration 5

In this chapter

- Overview 91
- VLAN configuration elements 91
- VLAN state elements 94
- Full examples 97

Overview

This chapter describes how to create virtual local area network (VLAN) with tagged and untagged ports using the YANG model, and illustrates the equivalent XML and CLI commands.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

VLAN configuration elements

Each element of the VLAN corresponds to the following:

- A YANG model description
- An XML description
- A CLI description

The following subsections compare a specific example for each VLAN configuration element.

VLAN identifier

The VLAN identifier is a restricted integer type that behaves as a unique identifier.

TABLE 20 VLAN identifier

YANG model	XML	CLI

leaf vlan-id
{
type vlan-identifier;
description "VLAN ID";
}
	<*brcd:vlan*>
	<*brcd:vlan-id*>2</*brcd:vlan-id*>
	</*brcd:vlan*>
		vlan <*num*>

VLAN name

The VLAN name is a unique string that describes a specific VLAN.

TABLE 21 VLAN name

YANG model	XML	CLI

leaf vlan-name
{
type string
{
length "1..31";
}
description "VLAN Name";
}
	<*brcd:vlan-name*>example</*brcd:vlan-name*>	[**no**] **vlan** <*vlan-id*> **name** <*vlan-name*>

Tagged ports

This data element indicates that a specific port is a tagged member of the VLAN.

TABLE 22 VLAN tagged ports

YANG model	XML	CLI

leaf-list tagged
{
type ethernet-pos-intf-multiple;
description "Tagged Interfaces list";
}
	<*brcd:tagged*>ethernet 1/1 to 1/2</*brcd:tagged*>
	<*brcd:tagged*>ethernet 1/4</*brcd:tagged*>
		[**no**] **tagged ethernet** <*slot-number*>/<*port-number*> [**to** <*slot-number*>/<*port-number*> | **ethernet** <*slot-number*>/<*port-number*>]

Untagged ports

This data element indicates that a specific port is an untagged member of the VLAN.

TABLE 23 VLAN untagged ports

YANG model	XML	CLI

leaf-list untagged
{
type ethernet-pos-intf-multiple;
description "Untagged Interfaces list";
}
	<*brcd:untagged*>ethernet 1/1 to 1/2</*brcd:untagged*>
	<*brcd:untagged*>ethernet 1/4</*brcd:untagged*>
		[**no**] **untagged** <*slot-number*>/<*port-number*> [**to** <*slot-number*>/<*port-number*> | **ethernet** <*slot-number*>/<*port-number*>]

Uplink ports

This data element indicates which ports in the VLAN are connected to outside resources.

TABLE 24 Uplink ports

YANG model	XML	CLI

leaf-list uplink-switch
{
type ethernet-pos-intf-multiple;
description "List of Uplink ports to enable "+
"uplink switching on this VLAN.";
}
	<*brcd:uplink-switch*>ethernet 1/4</*brcd:uplink-switch*>	[**no**] **uplink-switch ethernet** <*port-number*> [**to** <*port-number*> | **ethernet** <*port-number*>]

Router interface

This data element is the name of the router interface.

TABLE 25 Router interface

YANG model	XML	CLI

leaf router-interface
{
type ve-intf-single;
description "router VE interface";
}
	<*brcd:router-interface*>ve 2</*brcd:router-interface*>	**router-interface ve** <*ve-number*>

Loop detection

This data element indicates whether or not loop detection is enabled.

TABLE 26 Loop detection

YANG model	XML	CLI

leaf loop-detection
{
type empty;
description "Enable loop detection on the VLAN";
}
	<*brcd:loop-detection*></*brcd:loop-detection*>	[**no**] **loop-detection**

VLAN accounting

This data element indicates whether or not VLAN accounting is enabled.

NOTE
This data element is supported only on Brocade MLX series and Brocade NetIron XMR devices.

TABLE 27 VLAN accounting

YANG model	XML	CLI

leaf vlan-accounting
{
type empty;
description "Enable VLAN accounting on the VLAN";
}
	<brcd:vlan-accounting></brcd:vlan-accounting>	[**no**] **vlan-accounting**

VLAN CPU protection

This data element indicates whether or not CPU protection is enabled.

TABLE 28 VLAN CPU protection

YANG model	XML	CLI

leaf vlan-cpu-protection
{
type empty;
description "Enables Vlan CPU protection for "+
"unknown unicast and multicast packets";
}
	<brcd:vlan-cpu-protection></brcd:vlan-cpu-protection>	[**no**] **vlan-cpu-protection**

VLAN state elements

The following is the YANG model for the VLAN state data elements.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule vlan-statedata
{
    belongs-to netiron-statedata
    {
        prefix "statedata";
    }

    import common-defs
    {
        prefix "common";
    }

    organization
        "Brocade Communications Inc.";

    contact
             "Technical Support Center"+
             "130 Holger Way,"+
             "San Jose, CA  95134"+
             "Email:  ipsupport@brocade.com"+
             "Phone: 1-800-752-8061"+
```

```
         "URL:  www.brocade.com";

description "This module defines the device VLAN state data. VERSION: ";

revision 2011-04-20
{
    description "Initial revision";

grouping vlan-statedata-grouping
{
    description "device VLAN state data group";
    container vlan-statedata
    {
        description "List of VLANs configured on the system";

        list vlan
        {
            key vlan-id;
            max-elements 4090;
            description "device VLAN state data";

            leaf vlan-id
            {
                type common:vlan-identifier;
                description "VLAN ID";
            }

            leaf vlan-name
            {
                type string
                {
                    length "1..31";
                }
                description "VLAN Name";
            }

            leaf topo-hw-idx
            {
                type uint32
                {
                    range "0..511";
                }
                description "Topology hardware index";
            }

            leaf topo-sw-idx
            {
                type uint32
                {
                    range "1..256";
                }
                description "Topology group id";
            }

            leaf topo-next-vlan
            {
                type common:vlan-identifier;
                description "Topology next vlan";
            }
```

```
                    list port
                    {
                        key port-id;
                        description "VLAN port list";

                        leaf port-id
                        {
                            type common:interface-all-types;
                            description "VLAN port";
                        }

                        leaf tag-mode
                        {
                            type string
                            {
                                pattern
                                    "(untagged)"+
                                    "|(tagged)"+
                                    "|(broken)";
                            }
                            description "Tag mode of port";
                        }

                        leaf state
                        {
                            type string
                            {
                                length "1..19";
                            }
                            description "Current state of the port.";
                        }
                    }

                    leaf bytes-received
                    {
                        type string
                        {
                            length "1..31";
                        }
                        description "Bytes received";
                    }
                }

                leaf-list untagged
                {
                    type common:ethernet-pos-intf-multiple;
                    description "All the ports that are configured as untagged ports
in all the VLANs on the device";
                }

                leaf-list tagged
                {
                    type common:ethernet-pos-intf-multiple;
                   description "All the ports that are configured as tagged ports in
all the VLANs on the device";
                }

                leaf-list dual-mode
                {
                    type common:ethernet-pos-intf-multiple;
```

```
                description "All the ports that are configured as dual-mode ports
in all the VLANs on the device";
            }

            leaf default-vlan-id
            {
                type common:vlan-identifier;
                description "Default VLAN identifier";
            }

            leaf control-vlan-id
            {
                type common:vlan-identifier;
                description "Control VLAN identifier";
            }

            leaf maximum-port-vlan-entries
            {
                type uint32
                {
                    range "1..512";
                }
                description "Maximum PORT-VLAN entries";
            }

        }

    }

}
```

Full examples

The following shows the complete YANG model for the VLAN configuration, as well as example RPC requests and responses.

NOTE
For the most recent YANG models, visit *http://my.brocade.com*.

```
submodule vlan-config
{
    belongs-to netiron-config
    {
        prefix brcd;
    }

    import common-defs
    {
        prefix "common";
    }


    organization
        "Brocade Communications Inc.";

    contact
```

```
          "Technical Support Center"+
          "130 Holger Way,"+
          "San Jose, CA  95134"+
          "Email:  ipsupport@brocade.com"+
          "Phone: 1-800-752-8061"+
          "URL:  www.brocade.com";

    description "This module defines the VLAN configuration. VERSION: ";

    revision 2011-04-20
    {
        description "Initial revision";
    }

    grouping vlan-config-grouping
    {
        description "VLAN configuration group";
        container vlan-config
        {
            description "List of VLANs configured on the system";

            list vlan
            {
                key vlan-id;
                max-elements 4090;
                description "VLAN configuration";

                leaf vlan-id
                {
                    type vlan-identifier;
                    description "VLAN ID";
                }

                leaf vlan-name
                {
                    type string
                    {
                        length "1..31";
                    }
                    description "VLAN Name";
                }

                leaf-list tagged
                {
                    type ethernet-pos-intf-multiple;
                    description "Tagged Interfaces list";
                }

                leaf-list untagged
                {
                    type ethernet-pos-intf-multiple;
                    description "Untagged Interfaces list";
                }

                leaf-list uplink-switch
                {
                    type ethernet-pos-intf-multiple;
                    description "List of Uplink ports to enable "+
                           "uplink switching on this VLAN.";
                }
```

```
                uses priority-grouping;

                leaf router-interface
                {
                    type ve-intf-single;
                    description "router VE inteface";
                }

                leaf loop-detection
                {
                    type empty;
                    description "Enable loop detection on the VLAN";
                }

                leaf vlan-accounting
                {
                    type empty;
                    description "Enable VLAN Accounting on the VLAN";
                }

                leaf vlan-cpu-protection
                {
                    type empty;
                    description "Enables Vlan CPU protection for "+
                        "unknown unicast and multicast packets";
                }

            }

            leaf default-vlan-id
        {
            type vlan-identifier;
            description "Default VLAN identifier";
        }

        }

    }

}
```

The following example shows the RPC request and response of an <edit-config> operation.

```
<nc:rpc message-id="1" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:brcd="http://brocade.com/ns/netconf/config/netiron-config/">
<nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<brcd:netiron-config>
<brcd:vlan-config>
<brcd:default-vlan>2020</brcd:default-vlan>
<brcd:vlan>
<brcd:vlan-id>20</brcd:vlan-id>
<brcd:vlan-name>Test20</brcd:vlan-name>
<brcd:tagged>ethernet 1/2 to 1/3</brcd:tagged>
<brcd:tagged>ethernet 1/6 to 1/7</brcd:tagged>
```

```
<brcd:tagged>ethernet 1/10</brcd:tagged>
<brcd:tagged>ethernet 1/12 to 1/17</brcd:tagged>
<brcd:untagged>ethernet 1/4 to 1/5</brcd:untagged>
<brcd:untagged>ethernet 1/8 to 1/9</brcd:untagged>
<brcd:untagged>ethernet 1/11</brcd:untagged>
<brcd:untagged>ethernet 1/18</brcd:untagged>
<brcd:uplink-switch>ethernet 1/12 to 1/16</brcd:uplink-switch>
<brcd:router-interface>ve 10</brcd:router-interface>
<brcd:vlan-cpu-protection></brcd:vlan-cpu-protection>
<brcd:loop-detection></brcd:loop-detection>
<brcd:vlan-accounting></brcd:vlan-accounting>
<brcd:priority>
<brcd:value>6</brcd:value>
<brcd:force></brcd:force>
</brcd:priority>
</brcd:vlan>
<brcd:vlan>
<brcd:vlan-id>35</brcd:vlan-id>
<brcd:vlan-name>Test35</brcd:vlan-name>
<brcd:tagged>ethernet 2/2 to 2/3</brcd:tagged>
<brcd:tagged>ethernet 2/6 to 2/7</brcd:tagged>
<brcd:tagged>ethernet 2/10</brcd:tagged>
<brcd:tagged>ethernet 2/12 to 2/17</brcd:tagged>
<brcd:untagged>ethernet 2/4 to 2/5</brcd:untagged>
<brcd:untagged>ethernet 2/8 to 2/9</brcd:untagged>
<brcd:untagged>ethernet 2/11</brcd:untagged>
<brcd:untagged>ethernet 2/18</brcd:untagged>
<brcd:uplink-switch>ethernet 2/12 to 2/16</brcd:uplink-switch>
<brcd:router-interface>ve 20</brcd:router-interface>
<brcd:vlan-cpu-protection></brcd:vlan-cpu-protection>
<brcd:loop-detection></brcd:loop-detection>
<brcd:vlan-accounting></brcd:vlan-accounting>
<brcd:priority>
<brcd:value>3</brcd:value>
<brcd:force></brcd:force>
</brcd:priority>
</brcd:vlan>
</brcd:vlan-config>
</brcd:netiron-config>
</nc:config>
</nc:edit-config>
</nc:rpc>
]]>]]>
```

The following example shows the complete CLI for a VLAN configuration.

```
default-vlan 2020
vlan 20 name Test20
tagged ethernet 1/2 to 1/3
tagged ethernet 1/6 to 1/7
tagged ethernet 1/10
tagged ethernet 1/12 to 1/17
untagged ethernet 1/4 to 1/5
untagged ethernet 1/8 to 1/9
untagged ethernet 1/11
untagged ethernet 1/18
uplink-switch ethernet 1/12 to 1/16
router-interface ve 10
loop-detection
```

```
vlan-accounting
vlan-cpu-protection
priority 6
priority force
vlan 35 name Test35
tagged ethernet 2/2 to 2/3
tagged ethernet 2/6 to 2/7
tagged ethernet 2/10
tagged ethernet 2/12 to 2/17
untagged ethernet 2/4 to 2/5
untagged ethernet 2/8 to 2/9
untagged ethernet 2/11
untagged ethernet 2/18
uplink-switch ethernet 2/12 to 2/16
router-interface ve 20
loop-detection
vlan-accounting
vlan-cpu-protection
priority 3
priority force
```

Full examples