Alcatel-Lucent 7450

ETHERNET SERVICE SWITCH | RELEASE 13.0.R1

VERSATILE SERVICE MODULE GUIDE

Alcatel·Lucent

Disclaimers

Alcatel-Lucent products are intended for commercial uses. Without the appropriate network design engineering, they must not be sold, licensed or otherwise distributed for use in any hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life-support machines, or weapons systems, in which the failure of products could lead directly to death, personal injury, or severe physical or environmental damage. The customer hereby agrees that the use, sale, license or other distribution of the products for any such application without the prior written consent of Alcatel-Lucent, shall be at the customer's sole risk. The customer hereby agrees to defend and hold Alcatel-Lucent harmless from any claims for loss, cost, damage, expense or liability that may arise out of or in connection with the use, sale, license or other distribution of the products in such applications.

This document may contain information regarding the use and installation of non-Alcatel-Lucent products. Please note that this information is provided as a courtesy to assist you. While Alcatel-Lucent tries to ensure that this information accurately reflects information provided by the supplier, please refer to the materials provided with any non-Alcatel-Lucent product and contact the supplier for confirmation. Alcatel-Lucent assumes no responsibility or liability for incorrect or incomplete information provided about non-Alcatel-Lucent products.

However, this does not constitute a representation or warranty. The warranties provided for Alcatel-Lucent products, if any, are set forth in contractual documentation entered into by Alcatel-Lucent and its customers.

This document was originally written in English. If there is any conflict or inconsistency between the English version and any other version of a document, the English version shall prevail.

Table of Contents

Table of Contents

List of Tables

List of Tables

List of Figures

Versatile Service Module

List of Figures

Preface

About This Guide

This guide describes Versatile Service Module (VSM) functionality provided by Alcatel-Lucent's family of routers and presents examples to configure and implement various protocols and services.

This guide is organized into functional chapters and provides concepts and descriptions of the implementation flow, as well as Command Line Interface (CLI) syntax and command usage.

Audience

This guide is intended for network administrators who are responsible for configuring the 7450 ESS routers. It is assumed that the network administrators have an understanding of networking principles and configurations. Concepts described in this guide include the following:

- Versatile Service Module (VSM) service parameters
- Service management

List of Technical Publications

The 7450 ESS documentation set is composed of the following guides:

- 7450 ESS Basic System Configuration Guide

  This guide describes basic system configurations and operations.

- 7450 ESS System Management Guide

  This guide describes system security and access configurations as well as event logging and accounting logs.

- 7450 ESS Interface Configuration Guide

  This guide describes card, Media Dependent Adapter (MDA) and port provisioning.

- 7450 ESS Router Configuration Guide

  This guide describes logical IP routing interfaces and associated attributes such as an IP address, as well as IP and MAC-based filtering, and VRRP and Cflowd.

- 7450 ESS Routing Protocols Guide

  This guide provides an overview of routing concepts and provides configuration examples for RIP, OSPF, IS-IS, BGP, and route policies.

- 7450 ESS MPLS Guide

  This guide describes how to configure Multiprotocol Label Switching (MPLS) and Label Distribution Protocol (LDP).

- 7450 ESS Services Overview Guide
  This guide describes how to configure service parameters such as service distribution points (SDPs), customer information, and user services.

- 7450 ESS Layer 2 Services and EVPN Guide
  This guide describes Virtual Leased Lines (VLL), Virtual Private LAN Service (VPLS), Provider Backbone Bridging (PBB), and Ethernet VPN (EVPN).

- 7450 ESS Layer 3 Services Guide
  This guide describes Internet Enhanced Services (IES) and Virtual Private Routed Network (VPRN) services.

- 7450 ESS Versatile Service Module Guide
  This guide describes how to configure service parameters for the Versatile Service Module (VSM).

- 7450 ESS OAM and Diagnostics Guide

  This guide describes how to configure features such as service mirroring and Operations, Administration and Management (OAM) tools.

- 7450 ESS Triple Play Guide

  This guide describes Triple Play services and support provided by the 7450 ESS and presents examples to configure and implement various protocols and services.

- 7450 ESS Quality of Service Guide

  This guide describes how to configure Quality of Service (QoS) policy management.
- Multi-Service Integrated Service Adapter Guide

  This guide describes services provided by integrated service adapters such as Application Assurance, ad insertion (ADI) and Network Address Translation (NAT).

Technical Support

If you purchased a service agreement for your 7450 ESS router and related products from a distributor or authorized reseller, contact the technical support staff for that distributor or reseller for assistance. If you purchased an Alcatel-Lucent service agreement, follow this link to contact an Alcatel-Lucent support representative and to access product manuals and documentation updates:

http://support.alcatel-lucent.com

Versatile Service Module

In This Chapter

This chapter provides information about configuring Versatile Service Module (VSM) parameters.

Topics in this chapter include:

VSM Overview

In many instances, it is desirable to process a stream of packets from one or more subscribers through multiple features that, for one reason or another, are mutually exclusive in the 7450 ESS forwarding planes. For example, multiple subscriber sites could be bridged together through a VPLS instance while requiring in-service high speed Internet access (IES). Functionality of this type can be handled several ways:

- Multiple System Solution on page 14
- Hybrid Service Solution on page 14
- Single System Multiple Interface Solution on page 15
- Full Feature Internal Service Cross Connect Solution on page 15

For the purpose exploring each of these solutions, the VPLS and IES service interconnection scenario is examined.

Multiple System Solution

The multiple system (meaning multiple boxes) solution splits the functionality between two distinct nodes. The first node performs the VPLS bridging functions while maintaining per site QoS and accounting functions. The second node connects to the first node as a destination in the VPLS service. This connection could be configured as a SAP to SAP or a pseudo-wire spoke connection.

Hybrid Service Solution

The hybrid solution merges the two services into a single, common service. This can be accomplished for our example service interconnect by either supporting a virtual IP interface in the context of a VPLS service or providing an IP-only solution that provides for multiple SAPs on a single IES IP interface.

The hybrid solution does not provide for separate accounting and QoS for packets forwarded (or routed) between the subscriber sites and the packets routed to next-hops outside the subscriber domain.

Single System Multiple Interface Solution

The single system solution retains the same SLA enforcement and accounting capabilities as the multiple system solution but with the advantage of only requiring a single chassis. This is accomplished by defining the VPLS and IES services on different physical interfaces of the same type. Both interfaces are defined as access types and use the same encapsulation type (i.e., Dot1q). The services are configured with the same encapsulation values and the physical interfaces are interconnected using an external jumper cable. To avoid single point of failure issues, Link Aggregation Groups (LAG) can be used to provide an N-to-1 redundancy mechanism (as well as adding more interconnect bandwidth).

Full Feature Internal Service Cross Connect Solution

The internal service cross connect solution provides similar functionality as the single system multiple interface solution while attempting to minimize the cost, density and provisioning issues inherent to the external port jumper method. The internal service cross connection feature uses new service provisioning objects and a new type of hardware adapter to manage internal service cross connections. The remainder of this document describes the internal service cross connection feature.

Functional Components

The internal service cross connection feature uses an adapter designed to fit within an IOM (Input Output Module) MDA (Media Dependant Adapter) slot. There are two types of adapters the VSM-CCA and the new VSM-CCA-XP which supports all of the features of the VSM-CCA but allows for a new higher capacity mode. One or more adapters are placed into a cross connect aggregation group (CCAG). To cross connect two services, each service is bound to the same cross connect aggregation group using the same cross connection identifier. This section introduces each object and gives a brief explanation of its function.

Service Cross Connect Adapter (CCA)

The VSM Cross Connect Adapter (CCA) is a type of MDA for 7450 ESS platforms designed to provide an egress to ingress forwarding plane interconnection. When a CCA is installed in an MDA slot, a set of virtual ports is available to the system providing the ability to extend packet processing through an extra set of egress and ingress forwarding paths that CCA interfaces.

Unlike external port connections which utilize two TX-RX paths, a CCA interconnects the egress forwarding path on the IOM directly to the ingress forwarding path. This eliminates the need for the physical port MAC, PHY, cable and other MDA-specific components producing a less costly and more reliable adapter. The complete 10G+ forwarding path is available allowing single conversations up to 10G.

Bandwidth is utilized more efficiently than with externally cabled ports. Typically, the offered load presented to each side of the cross-connect port-pair is asymmetric in nature. When physical ports are used to cross connect services, each service is egress bandwidth-limited to the link speed of the TX-RX path.

If one TX-RX path is under-utilized, egress services on the other path cannot make use of the available bandwidth. Since the CCA is forwarding all services over the same path, all the available bandwidth can be used.

The forwarding plane that the CCA interconnects maintains the complete egress and ingress features of the services it is interconnecting. This includes the ability to remap QoS, enforce policing and shaping, and provide ingress and egress accounting for each service.

Internal Service CCAG

VSM CCAs are placed in a CCAG. A CCAG provides a mechanism to aggregate multiple CCAs into a single forwarding group. The CCAG uses conversation hashing to dynamically distribute cross-connect traffic to the active CCAs in the aggregation group. In the event that an active CCA fails or is removed from the group, the conversation hashing function redistributes the traffic over the remaining active CCAs within the group.

The conversation hashing mechanism performed for a CCAG is identical to the hashing functions performed for Ethernet LAGs (Link Aggregation Groups).

Internal Service Cross Connect Identifier (CCID)

Services and IP interfaces are bound to a CCAG through a CCID (Cross Connect Identifier). When two services or a service and an IP interface are assigned the same CCID the CCAG attempts to provide a cross connection path between the objects. The CCID enables multiple pairs of cross connected services to share the same CCAG.

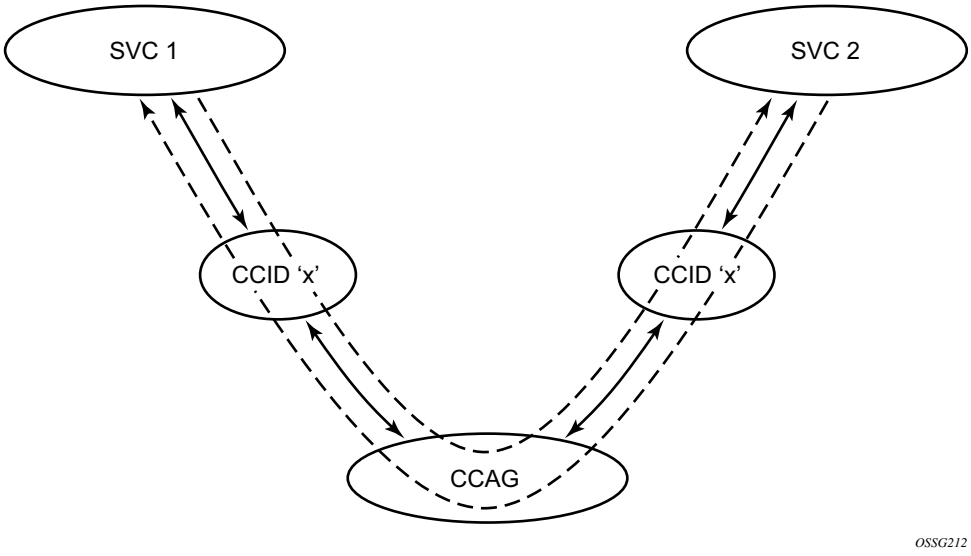

Figure 1: Internal Service Interconnection Using CCID

From a service perspective, a CCID is an object that not only binds two services together, but also provides the attachment point for the ingress and egress QoS, filtering, and accounting parameters. When considered in conjunction with the CCAG, it allows the actual cross connection path (through the CCAs) to be indirectly associated with the services using the CCAG and maintains a simplified provisioning model over port level cross connected services.

CCAG Bandwidth and Resiliency

A CCAG is an intermediate object between cross-connected objects (SAPs and network IP interfaces) and the CCAs. A CCAG is similar to a Link Aggregation Group (LAG) of Ethernet ports and uses the same underlying mechanisms to distribute conversations over multiple CCAs and converge when a CCA becomes active or inactive in the group.

When a CCAG is created, the system allocates six Ethernet LAGs for the virtual ports on the CCAs placed into the group. Each virtual port is placed into a respective LAG. For instance, each time a CCA is placed into the CCAG, virtual port 1 on that CCA is placed into the first LAG allocated to that CCAG. Virtual port 2 is placed into the second LAG on the CCAG. Virtual ports 3 through 6 are placed into their respective LAGs as well.

Using the set of LAGs provides a mechanism for conversation hashing or service mapping over all member CCAs in the CCAG. In the unlikely event that a CCA fails or is removed from the CCAG, the system will automatically modify the conversation hashing or service mapping on the CCAG to represent the available active CCAs.

CCAG LAG Attributes

Unlike a user-provisioned LAG, the internal LAGs do not use a primary member to control the typical port level configuration parameters. Instead, the parameters usually found at the port level are implemented directly on the CCAG internal LAG representative objects (**sap-sap**, **sap-net** and **net-sap**) for each path. These commands perform functions such as MTU definition and locally administering the MAC address.

The default unique MAC addresses used each internal LAG within the CCAG are automatically assigned from the chassis MAC pool. These MAC addresses are assigned from the pool based on an offset relative to the CCAG-ID. The same set of default MAC addresses are assigned each time a specific CCAG-ID is created.

Although a CCAG uses internal LAG mechanisms, the LACP protocol is not supported or required. LAG resources used for CCAG purposes are not exposed to the user.

CCAG Traffic Distribution

A CCAG uses both direct object mapping and conversation hashing to distribute traffic over multiple CCAs. To understand how each object type’s ingress traffic is distributed over the active CCAs in a CCAG, refer to the LAG and ECMP Hashing section of the 7450 ESS Interface Configuration Guide.

CCAG SAP QoS

When a SAP is created on a CCAG, the service queues defined by the ingress and egress QoS policy are created on each CCA member in the CCAG. Packets are forwarded to the egress queues based on the hashing or service mapping enforced by the LAG functions internal to the system. Packets are received on a CCA ingress queue based on which CCA handled the egress processing. Each ingress and egress hardware queue buffering and rate parameters are managed by the system based on one of two models governed by the state of the LAG QoS adaptation setting. The adaptation state also governs the application of hierarchical virtual schedulers associated with the SAP queues.

Link Level CCAG SAP QoS Adaptation

Link level QoS adaptation is set when the CCA access QoS adaptation flag is set to **link**. Link-level distribution informs the system that a service queue's buffering and rate parameters should be applied directly to each hardware queue representing the service queue. For example, when a service queue is configured with a rate equal to 10Mbps, each corresponding CCA hardware queue will be configured with a rate of 10Mbps. Given many flows conversation hashing to different CCAs, the maximum forwarded rate will be the 10Mbps multiplied by the number of active CCAs.

When a link-level adaptation service queue is a child to a parent virtual scheduler, the parent scheduler and the rest of the scheduler hierarchy is implemented per CCA. An instance of the scheduler policy is maintained per CCA.

When a CCAG SAP is a member of a Multi-Service Site (MSS), all SAPs in the MSS must be CCAG SAPs created on the same CCAG-ID.

Distributed CCAG SAP QoS Adaptation

Distributed QoS adaptation is set when the CCA access QoS adaptation flag is set to distribute. The distributed QoS parameter setting informs the system that a service queue's buffering and rate parameters should be distributed between the active CCAs in the CCAG. For example, when a service queue is configured with a rate equal to 10Mbps and two CCAs are active in the CCAG, each corresponding CCA hardware queue will be configured with a rate of 5Mbps (1/2 of the provisioned service queue parameters). Given many flows conversation hashing to different CCAs, the maximum forwarded rate will be limited to 10Mbps.

When a distributed adaptation service queue is a child to a parent virtual scheduler, the parent scheduler and the rest of the scheduler hierarchy is implemented on each IOM with an active member CCA from the CCAG. The scheduler parameters are divided amongst the IOMs with active CCAs based on the total number of active CCAs. If there are three active CCAs in the

CCAG, each CCA represents 1/3 of rate and CIR defined for each scheduler in the policy. If two of the active CCAs are on one IOM and one active CCA is on a second IOM, the first IOM would receive 2/3 of the rate and CIR for each scheduler and the second IOM would receive 1/3. The overall distribution is based on the following equation:

IOM Scheduler Rate = Policy Scheduler Rate * (Number Active CCAs on IOM / Total Active CCAs)

IOM Scheduler CIR = Policy Scheduler CIR * (Number Active CCAs on IOM / Total Active CCAs)

When a CCAG SAP is a member of a multi-service site, all SAPs in the multi-service site must be CCAG SAPs created on the same CCAG-ID.

VSM-CCA-XP

In addition to supporting all the features of the existing VSM-CCA, the new VSM-CCA-XP MDA offers a new hybrid mode for simplified provisioning and a higher capacity VSM when inserted on IOM3 cards. As with the VSM-CCA MDA the complete forwarding path bandwidth (in this case 25G) is available allowing single conversations up to 25G on a single MDA.

The uses cases for VSM-CCA-XP are nearly identical to the VSM-CCA. When configured as a VSM-CCA-XP port x/x1 and port x/x/2 are internally connected. Therefore configuration is very similar to a physical loop back port using Ethernet with dot1Q encapsulation. The use of hybrid port removes the requirement to configure net and sap parameters and simplifies provisioning. The use of the Ethernet VLAN Tag is used to connect the SAPs.

VSM-CCA-XP Exceptions:

- While LAG is available LACP is not allowed.
- Ethernet CFM is only available when Eth-Rings are configured on the VSM (Ethernet rings use Ethernet MEPS for Control)

The new VSM-CCA-XP can be configured as a VSM-CCA MDA to support CCA functions on IOM1, IOM2 and IOM3. On IOM3 the VSM-CCA MDA supports a loop back mode that uses LAG and 2 ports using Ethernet as the internal connection. The LAG feature also conversations hashing just as the original VSM-CCA. The hybrid port mode eliminates the need to specify network or access modes.

Sample Configuration for MDA. Normally when a VSM-CCA-XP MDA is inserted it may be configured as a VSM-CCA or a VSM-CCA-XP.

```
===============================================================================
MDA Summary
===============================================================================
Slot  Mda   Provisioned           Equipped              Admin     Operational
            Mda-type              Mda-type              State     State
-------------------------------------------------------------------------------
```

```
1     1     vsm-cca              vsm-cca-xp              up        up
      2     vsm-cca-xp           vsm-cca-xp              up        up
===============================================================================
    card 1
        mda 2
            mda-type vsm-cca-xp
            no shutdown
        exit
    exit
```

Sample VSM-CCM-XP Configuration for Ports:

```
    port 1/2/1
        ethernet
        exit
        no shutdown
    exit
    port 1/2/2
        ethernet
        exit
        no shutdown
    exit
```

Port and Ethernet QoS parameters may be configured as with physical port. The Ethernet on VSM-CCA-XP has a reduced set of features. For example dot1Q is the only supported encapsulation. LACP cannot be configured on LAGs using the port.

The ports may be used directly by service SAP in the case of a single loop back. If resiliency desired, or more capacity is needed, a LAG can be configured.

Sample Configuration for LAG on a single VSM-CCA-XP MDA:

```
    lag 1
        mode hybrid
        encap-type dot1q
        port 1/2/1 // VSM-CCA-XP
        no shutdown
    exit
    lag 2
        mode hybrid
        encap-type dot1q
        port 1/2/2 // VSM-CCA-XP
        no shutdown
    exit
```

The following is a sample for an VPLS service equivalent using the LAG port.

```
  vpls 121 customer 1 create
     stp
         shutdown
     exit
     sap lag-1:1001 create // Connect using VLAN Tag 1001
     exit
     no shutdown
     ...
 exit
```

The following is a sample for an IES service equivalent to the configuration.

```
 ies 122 customer 1 create
     interface "Loopback" create
         address 8.1.1.1/24
         sap lag-2:1001 create
             ingress
                qos 3
             exit
             egress
                 qos 1010
             exit
         exit
     exit
     ...
     no shutdown
exit
```

A VSM-CCA-XP may be configured as either a VSM-CCA MDA or a VSM-CCA-XP MDA. When configured as a VSM-CCA-XP it is not a member of a CCA Group (ref VSM-CCA-XP).

Configuration Process Overview

Figure 2 displays the basic components to configure card, router interface, and service CCAG components.

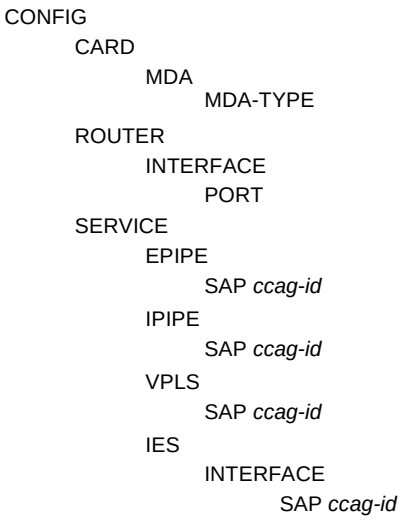

Figure 2: VSM/CCAG Configuration Components

Configuration Notes

The following information describes provisioning caveats:

- Services can only be provisioned on Ethernet SAPs.
- The cross connections supported are:
  - → IP to all Layer 2 SAPs
  - → SAP to SAP of all types with the exception of:
    - A cross connection within the same service.
    - An IES service to another IES service.

Configuring VSM and CCAG with CLI

This section provides information to configure cards, MDAs, and ports.

Topics in this section include:

Basic Configuration

The following fields require specific input (there are no defaults) to configure VSM:

- CCAG ID
- For a local service, two SAPs must be configured specifying the source and destination nodes and ports
- For a distributed service, one SAP and one SDP must be specified

The following example displays VSM defaults when a *ccag-id* is created.

```
A:ALA-48>config>vsm# info detail
#----------------------------------------
echo "Versatile Services Module Configuration"
#----------------------------------------
    vsm
        ccag 1 create
            no description
            cca-rate max
            access
                adapt-qos distribute
            exit
            path a
                weight 50
                rate max aggregate
                sap-sap
                    no mac
                    no mtu
                    egress
                        pool
                            resv-cbs default
                            slope-policy "default"
                        exit
                    exit
                    ingress
                        pool
                            resv-cbs default
                            slope-policy "default"
                        exit
                    exit
                exit
                sap-net
                    no mac
                    no mtu
                    egress
                        pool
                            resv-cbs default
                            slope-policy "default"
                        exit
                    exit
                    ingress
                        pool
                            resv-cbs default
                            slope-policy "default"
                        exit
                    exit
                exit
```

```
        net-sap
            no mac
            no mtu
            no accounting-policy
            no collect-stats
            queue-policy "default"
            egress
                pool
                    resv-cbs default
                    slope-policy "default"
                exit
            exit
        exit
    exit
    path b
        weight 50
        rate max aggregate
        sap-sap
            no mac
            no mtu
            egress
                pool
                    resv-cbs default
                    slope-policy "default"
                exit
            exit
            ingress
                pool
                    resv-cbs default
                    slope-policy "default"
                exit
            exit
        exit
        sap-net
            no mac
            no mtu
            egress
                pool
                    resv-cbs default
                    slope-policy "default"
                exit
            exit
            ingress
                pool
                    resv-cbs default
                    slope-policy "default"
                exit
            exit
        exit
        net-sap
            no mac
            no mtu
            no accounting-policy
            no collect-stats
            queue-policy "default"
            egress
                pool
                    resv-cbs default
                    slope-policy "default"
                exit
            exit
```

```
                exit
            exit
            exit
            no shutdown
        exit
    exit
-----------------------------------------------
A:ALA-48>config>vsm#
```

Common Configuration Tasks

This section provides a brief overview of the tasks that should be performed to configure VSM on an MDA, router, router interface, and services.

- Provision one or more CCA as MDAs in the system.
- Create VSM CCAGs on the system.
- Provision CCAG path bandwidth, path weighting, and overall bandwidth parameters.
- Provision member CCAs into a CCAG.
- Provision service SAPs using a CCAG, path, and CCID for cross connect purposes.
- Bind routed network IP interfaces to a CCAG, path, and CCID for cross connect purposes.

Configure VSM CCAG Components

Use the CLI syntax displayed below to configure the following entities:

- Provision VSM on an MDA on page 29
- Cross Connecting Network IP Interfaces on page 34
- Provision CCAG Parameters on page 31
- Configure Path Components on page 32
- Cross Connecting Services on page 35

Provision VSM on an MDA

Before a CCA module may be utilized in the system, the CCA must be provisioned into an MDA slot. The MDA provisioning command must be modified to support provisioning a CCA adaptor type. Up to 8 member CCAs can be configured per CCAG.

CLI Syntax: `config>card# mda` *`mda-number mda-type`* `{other-MDA-type|cca}`

The following example displays the command usage to provision CCA on an MDA:

Example:
```
config# card 10
    config>card# mda 1
    config>card>mda# mda-type vsm-cca
    config>card>mda# exit
    config>card#
```

The following example displays the configuration:

```
A:ALA-48>config>card# info
----------------------------------------------
        card-type iom-20g
        mda 1
            mda-type vsm-cca
        exit
        mda 2
            mda-type m20-1gb-tx
        exit
----------------------------------------------
A:ALA-48>config>card#
```

Provision CCAG Parameters

Once a CCA is provisioned into the system, it must be placed in a Cross Connect Aggregation Group (CCAG) to be used by cross connect objects. Besides CCA membership, the CCAG also supports bandwidth control parameters (see Configure Path Components on page 32) used to manipulate forwarding distribution between objects in the alpha and beta path groups and the aggregate rate allowed on the CCA.

Use the following CLI syntax to provision CCAG components.

CLI Syntax:
```
config>vsm#
            ccag ccag-id [create]
            cca-rate kilobits-per-second
            description description-string
            member-cca card-slot/mda-number
            path {a|b}
            no shutdown
```

The following example displays the command usage to provision CCAG components:

Example:
```
config>vsm# ccag 1
       config>vsm>ccag# description "VSM test"
       config>vsm>ccag# cca-rate 1000000
       config>vsm>ccag# member-cca 10/1
```

The following example displays the configuration:

```
A:ALA-48>config>vsm# info
----------------------------------------------
        ccag 1 create
            description "VSM test"
            cca-rate 1000000
            member-cca 10/1
        exit
...
----------------------------------------------
A:ALA-48>config>vsm#
```

Configure Path Components

Each CCA is divided into two distinct paths for bandwidth management purposes. One path is identified as alpha (a) and the other beta (b). The significance of each path for bandwidth distribution is dependent on the relative path weights each path is given in relationship to the other. A maximum path rate may also be defined allowing the provisioning of a maximum cap on the aggregate bandwidth allowed to the SAP or IP interface queues associated with the path.

Each path is separated into three other contexts; SAP-2-SAP (sap-sap), SAP-2-Net (sap-net) and Net-2-SAP (net-sap). Each path context allows for the definition of the features that are usually associated with physical ports on other MDAs in the system. These include buffer pool management, ingress network queue definitions and accounting policy control.

Use the following CLI syntax to provision path components.

- Net SAP
- SAP net
- SAP SAP

Use the following CLI syntax to provision CCAG path components.

CLI Syntax:

```
config>vsm>ccag#
            path {a|b}
               net-sap
                  accounting-policy policy-id
                  collect-stats
                  egress
                     pool
                        resv-cbs percent-or-default
                        slope-policy slope-policy-name
                  mac ieee-address
                  mtu mtu-bytes
                  queue-policy queue-policy-name
               rate kilo-bits-per-second [aggregate|cca]
               sap-net
                  egress
                     pool
                        resv-cbs percent-or-default
                        slope-policy slope-policy-name
                  ingress
                     pool
                        resv-cbs percent-or-default
                        slope-policy slope-policy-name
                  mac ieee-address
                  mtu mtu-bytes
               sap-sap
```

```
egress
   pool
      resv-cbs percent-or-default
      slope-policy slope-policy-name
ingress
   pool
      resv-cbs percent-or-default
      slope-policy slope-policy-name
mac ieee-address
mtu mtu-bytes
weight path-weight
```

The following displays a CCAG path configuration example:

```
A:ALA-48>config>vsm# info
----------------------------------------------
        ccag 1 create
            description "VSM test"
            member-cca 10/1
            path a
                weight 100
            exit
            path b
                weight 100
                rate 99999999
            exit
            no shutdown
        exit
...
----------------------------------------------
A:ALA-48>config>vsm#
```

Cross Connecting Network IP Interfaces

To support cross connection between services and network IP interfaces, the network interface port command has been augmented to allow the binding of the IP interface to a **ccag** *cc-id*. Similar to service CCAG SAPs, the network IP interface port binding command must reference the ccag-id, the CCA path (.a or .b) and the *cc-id* used by the service CCAG SAP on the other CCA path.

Use the following CLI syntax to configure CCAG a network IP interface.

CLI Syntax:
```
config# router [router-name]
            interface interface-name
               port ccag-ccag-id.{a|b}[.net-sap]:cc-id
               address {ip-address/mask |ip-address netmask} [broadcast
                  all-ones|host-ones]
               mac ieee-address
```

The following displays CCAG network IP interface configuration examples:

```
A:ALA-48>config>router# info
#------------------------------------------
echo "IP Configuration"
#------------------------------------------
...
        interface "ccanet"
            address 2.1.1.1/24
            port ccag-1.a.net-sap:200
            mac 00:00:00:00:00:ff
        exit
        interface "ccanet2"
            address 4.1.1.1/24
            port ccag-1.b.net-sap:300
            static-arp 4.1.1.2 00:00:00:00:00:aa
        exit
...
#------------------------------------------
A:ALA-48>config>router#
```

Cross Connecting Services

Services are provisioned onto a CCAG using a special CCAG SAP definition. CCAG SAPs must reference a *ccag-id*, a CCA path (a or b), a pairing type (sap-sap or sap-net) and a unique *cc-id*. The *ccag-id* identifies the group of CCAs that will be used for forwarding packets associated with the SAP. The path identifies the bandwidth control grouping used to manage CCA egress bandwidth. The pairing type helps the system identify which buffering resources will be used to manage egress queuing of packets. Finally, the *cc-id* is used to explicitly cross connect the SAP to another SAP or network IP interface configured with the same *cc-id*.

- Epipe on page 35
- VPLS on page 36
- IES on page 37

Epipe

CLI Syntax:
```
config>service#
            epipe service-id [customer customer-id]
               sap ccag-ccag-id.{a|b}[.sap-net|.sap-sap]:cc-id [create]
```

The following displays an Epipe SAP configuration referencing a *ccag-id*:

```
A:ALA-48>config>service# info
----------------------------------------
...
        epipe 103 customer 6 vpn 103 create
            sap 3/1/1.1.1 create
            exit
            sap ccag-1.a:100 create
            exit
            no shutdown
        exit
----------------------------------------
A:ALA-48>config>service#
```

VPLS **CLI Syntax:** config>service#
vpls *service-id* [customer *customer-id*]
sap ccag-*ccag-id*.{a|b}[.*sap-net*|.*sap-sap*]:*cc-id* [create]

The following displays a VPLS SAP configuration referencing a *ccag-id*:

```
A:ALA-48>config>service# info
----------------------------------------
...
        vpls 740 customer 1 vpn 740 create
            stp
                shutdown
            exit
            sap 1/1/19:1 create
            exit
            sap 1/1/19:2 create
                ingress
                    qos 3
                exit
            exit
            sap ccag-1.a:456 create
                ingress
                    qos 3
                exit
                egress
                    qos 1010
                exit
            exit
            no shutdown
        exit
...
----------------------------------------
A:ALA-48>config>service#
```

IES

CLI Syntax: config>service#
ies *service-id* [customer *customer-id*]
interface *ip-interface-name*
sap ccag-*ccag-id*.{a|b}[.*sap-net*|.*sap-sap*]:*cc-id* [create]

The following displays an IES SAP configuration referencing a *ccag-id*:

```
A:ALA-48>config>service# info
----------------------------------------
...
        ies 200 customer 1 create
            interface "ccaiesif" create
                address 8.1.1.1/24
                sap ccag-1.b:456 create
                    ingress
                        qos 3
                    exit
                    egress
                        qos 1010
                    exit
                exit
            exit
            no shutdown
        exit
...
----------------------------------------
A:ALA-48>config>service#
```

Service Management Tasks

This section discusses the following service management tasks:

Modifying or Deleting a VSM MDA

To change or delete a VSM MDA already provisioned for a specific slot, first you must shut down and remove all service SAP and router interface associations (page 39) to delete the VSM MDA from the configuration.

CLI Syntax:
```
config> card slot-number
          [no] mda mda-number
             [no] mda-type mda-type
             shutdown
```

Example:
```
config# card 10
    config>card# mda 1
    config>card>mda# mda-type vsm-cca
    config>card>mda# shutdown
    config>card>mda# exit
    config>card# no mda 1
```

The following example displays the configuration:

```
A:ALA-48>config>card# info
----------------------------------------------
        card-type iom-20g
        mda 2
            mda-type vsm-cca
        exit
----------------------------------------------
A:ALA-48>config>card#
```

Modifying CCAG Parameters on a Network IP Interface

CLI Syntax: config# router [*router-name*]
interface *interface-name*
shutdown
no port ccag-*ccag-id*.{a|b}[.net-sap]:*cc-id*

The following example displays the command usage:

Example:
```
config>router# interface ccanet
config>router>if# address 3.1.1.1/24
config>router>if# exit
```

```
A:ALA-48>config>router# info
----------------------------------------------
#----------------------------------------
echo "IP Configuration"
#----------------------------------------
...
        interface "ccanet"
            address 3.1.1.1/24
            port ccag-1.a.net-sap:200
            mac 00:00:00:00:00:ff
        exit
        interface "ccanet2"
            address 4.1.1.1/24
            port ccag-1.b.net-sap:300
            static-arp 4.1.1.2 00:00:00:00:00:aa
        exit
...
#----------------------------------------
A:ALA-48>config>router#
```

Modifying CCAG Parameters

CLI Syntax: config>vsm#
```
ccag ccag-id [create]
no ccag ccag-id [force]
   access
      adapt-qos {link|distribute|port-fair}
   cca-rate kilobits-per-second
   no cca-rate
   description description-string
   no description
   [no] member-cca card-slot/mda-number
   path {a|b}
   no shutdown
```

The following example displays the command usage to provision CCAG components:

Example:
```
config>vsm# ccag 1
config>vsm>ccag# access
config>vsm>ccag>access#
config>vsm>ccag>access# adapt-qos distribute
config>vsm>ccag>access# exit
config>vsm>ccag# member-cca 10/2
config>vsm>ccag# exit
```

The following example displays the configuration:

```
A:ALA-48>config>vsm# info
----------------------------------------------
        ccag 1 create
            description "VSM test"
            member-cca 10/1
            member-cca 10/2
            path a
                weight 100
            exit
            path b
                weight 100
                rate 99999999
            exit
            no shutdown
        exit

...
----------------------------------------------
A:ALA-48>config>vsm# ccag 1
```

Modifying Path Parameters

The following example displays the command usage to provision CCAG path parameters:

Example:
```
config>vsm# ccag 1
       config>vsm>ccag# path a
       config>vsm>ccag>path# no weight
       config>vsm>ccag>path# net-sap
       config>vsm>ccag>path>net-sap# queue-policy nq1
       config>vsm>ccag>path>net-sap# egress
       config>vsm>ccag>path>net-sap>egr# pool
       config>vsm>ccag>path>net-sap>egr>pool# slope-policy A
       config>vsm>ccag>path>net-sap>egr>pool# exit
       config>vsm>ccag>path>net-sap>egr# exit
       config>vsm>ccag>path>net-sap# exit
       config>vsm>ccag>path# exit
       config>vsm>ccag# path b
       config>vsm>ccag>path# no rate
       config>vsm>ccag>path# sap-sap
       config>vsm>ccag>path>sap-sap# egress
       config>vsm>ccag>path>sap-sap>egr# pool
       config>vsm>ccag>path>sap-sap>egr>pool#
       config>vsm>ccag>path>sap-sap>egr>pool# slope-policy B
       config>vsm>ccag>path>sap-sap>egr>pool# exit
       config>vsm>ccag>path>sap-sap>egr# exit
       config>vsm>ccag>path>sap-sap# exit
       config>vsm>ccag>path# exit
       config>vsm>ccag#
```

The following example displays the configuration:

```
A:ALA-48>config>vsm# info
----------------------------------------------

        ccag 1 create
            description "VSM test"
            member-cca 10/1
            member-cca 10/2
            path a
                net-sap
                    queue-policy "nq1"
                    egress
                        pool
                            slope-policy "A"
                        exit
                    exit
                exit
            exit
            path b
                weight 100
                sap-sap
                    egress
                        pool
```

```
                            slope-policy "B"
                        exit
                    exit
                exit
            exit
            no shutdown
        exit
...
-----------------------------------------------
A:ALA-48>config>vsm#
```

Modifying Service Parameters

Epipe

CLI Syntax:
```
config>service#
            epipe service-id
               sap ccag-ccag-id.{a|b}[.sap-net|.sap-sap]:cc-id
               no sap sap-id
               shutdown
```

The following service examples display the command usage to provision CCAG.

Example:
```
config>service# epipe 103
        config>service>epipe#  sap ccag-1.a:100
        config>service>epipe>sap# shutdown
        config>service>epipe>sap# exit
        config>service>epipe# no sap ccag-1.a:100
        config>service>epipe# sap ccag-1.b:200 create
        config>service>epipe>sap$ no shutdown
        config>service>epipe>sap$ exit
        config>service>epipe#
```

The following output displays the configuration:

```
A:ALA-48>config>service>epipe# info
----------------------------------------------
            sap 3/1/1.1.1 create
            exit
            sap ccag-1.b:200 create
            exit
            no shutdown
----------------------------------------------
A:ALA-48>config>service>epipe#
```

VPLS

CLI Syntax:
```
config>service#
    vpls service-id [customer customer-id]
        sap ccag-ccag-id.{a|b}[.sap-net|.sap-sap]:cc-id
        no sap sap-id
        shutdown
```

Example:
```
config>service>vpls# sap ccag-1.a:456
config>service>vpls>sap# shutdown
config>service>vpls>sap# exit
config>service>vpls# no sap ccag-1.a:456
config>service>vpls# sap ccag-1.b:100 create
config>service>vpls>sap$ no shutdown
config>service>vpls>sap$ exit
config>service>vpls# sap ccag-1.a:100
config>service>vpls>sap# ingress
config>service>vpls>sap>ingress# qos 3
config>service>vpls>sap>ingress# exit
config>service>vpls>sap# egress
config>service>vpls>sap>egress# qos 1010
config>service>vpls>sap>egress# exit
config>service>vpls>sap# exit
```

```
A:ALA-48>config>service>vpls# info
----------------------------------------------
            stp
                shutdown
            exit
            sap 1/1/19:1 create
            exit
            sap 1/1/19:2 create
                ingress
                    qos 3
                exit
            exit
            sap ccag-1.b:100 create
            exit
            no shutdown
----------------------------------------------
A:ALA-48>config>service>vpls#
```

IES

CLI Syntax:
```
config>service#
    ies service-id [customer customer-id]
       interface ip-interface-name
          sap ccag-ccag-id.{a|b}[.sap-net|.sap-sap]:cc-id
          no sap sap-id
          shutdown
```

Example:
```
config>service# ies 200
config>service>ies# interface "ccanet6"
config>service>ies>if# sap ccag-1.a:101 create
config>service>ies>if>sap# ingress
config>service>ies>if>sap>ingress# qos 3
config>service>ies>if>sap>ingress# exit
config>service>ies>if>sap# egress
config>service>ies>if>sap>egress# qos 1010
config>service>ies>if>sap>egress# exit
config>service>ies>if>sap# no shutdown
config>service>ies>if>sap# exit
config>service>ies>if#
```

The following output displays the configuration:

```
A:ALA-48>config>service>ies# info
----------------------------------------------
            interface "ccaiesif" create
                address 8.1.1.1/24
                sap ccag-1.b:456 create
                    ingress
                        qos 3
                    exit
                    egress
                        qos 1010
                    exit
                exit
            exit
            interface "ccanet6" create
                address 7.1.1.1/24
                sap ccag-1.a:101 create
                    ingress
                        qos 3
                    exit
                    egress
                        qos 1010
                    exit
                exit
            exit
            no shutdown
----------------------------------------------
A:ALA-48>config>service>ies#
```

VSM Command Reference

Command Hierarchies

VSM Configuration Commands

```
config
    — vsm
        — ccag ccag-id [create]
        — no ccag ccag-id [force]
            — access
                — adapt-qos {link | distribute | port-fair}
                — no adapt-qos
        — cca-rate kilobits-per-second
        — no cca-rate
        — description description-string
        — no description
        — [no] member-cca card-slot/mda-number
        — path {a | b}
            — net-sap
                — accounting-policy accounting-policy
                — no accounting-policy
                — [no] collect-stats
                — egress
                    — pool
                        — resv-cbs percentage-of-pool
                        — no resv-cbs
                        — slope-policy slope-policy-name
                        — no slope-policy
                — mac mac-address
                — no mac
                — mtu mtu-size
                — no mtu
                — queue-policy queue-policy-name
                — no queue-policy
            — rate kilobits-per-second [aggregate | cca]
            — no rate
            — sap-net
                — egress
                    — pool
                        — resv-cbs percentage-of-pool
                        — no resv-cbs
                        — slope-policy slope-policy-name
                        — no slope-policy
                — ingress
                    — pool
```

```
                                        — resv-cbs percentage-of-pool
                                        — no resv-cbs
                                        — slope-policy slope-policy-name
                                        — no slope-policy
                                — mac mac-address
                                — no mac
                                — mtu mtu-size
                                — no mtu
                        — sap-sap
                                — egress
                                        — pool
                                                — resv-cbs percentage-of-pool
                                                — no resv-cbs
                                                — slope-policy slope-policy-name
                                                — no slope-policy
                                — ingress
                                        — pool
                                                — resv-cbs percentage-of-pool
                                                — no resv-cbs
                                                — slope-policy slope-policy-name
                                                — no slope-policy
                                — mac mac-address
                                — no mac
                                — mtu mtu-size
                                — no mtu
                        — weight path-weight
                        — no weight
        — [no] shutdown
```

Related Commands

config card *slot-number*
- **mda** {**1** | **2**} **type** {**existing-mda-types** | **vsm**}
- [**no**] **mda** {**1** | **2**}

config router [*router-name*]
- [**no**] **interface** *ip-interface-name*
  - **port ccag-***ccag-id*.{**a** | **b**}[.*net-sap*]:*cc-id*
  - **no port**

config service
- **epipe** *service-id* [**customer** *customer-id*]
  - **sap ccag-***ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id* [**create**]
  - **no sap ccag-***ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id*

config service
- **vpls** *service-id* [**customer** *customer-id*]
  - **sap ccag-***ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id* [**create**]
  - **no sap ccag-***ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id*

ies *service-id* [**customer** *customer-id*]
- **interface** *ip-interface-name*
  - **sap ccag-***ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id* [**create**]
  - **no sap ccag-***ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id*

VSM Configuration Commands

Generic Commands

shutdown

Syntax [**no**] **shutdown**

Context config>vsm>ccag

Description This command controls the administrative state of the *ccag-id* the command is executed under. Upon creation, the default state of a CCAG is to be administratively up which corresponds to the no shutdown form of the command. If the CCAG must be forced to be operationally down, the **shutdown** command will place the CCAG into an administratively down state causing the operational state to also be down.

When a CCAG is shutdown, all SAPs associated with the CCAG will be operationally down. An operationally down SAP cannot be used for forwarding packets. If the SAP is part of the VPLS service, all MAC entries associated with the SAP will be removed from the VPLS FDB and the SAP will be removed from the flooding domain of the VPLS. If the SAP is part of an IES service, the associated IP interface will be set to an operationally down state. Network IP interfaces bound to a shutdown CCAG will be operationally down as well.

Executing the no shutdown command sets the CCAG to the default up administrative state. As long as at least one member CCA in the CCAG is active, all SAPs and network IP interfaces associated with the CCAG will be allowed to enter the operationally up state.

Default **no shutdown**

description

Syntax **description** *description-string*
no description

Context config>vsm

Description This command defines an informational string associated with the CCAG. The description string may be up to 80 characters long and contain only printable ASCII characters. Each time this command is successfully executed, any previous description string will be overwritten. If the command fails due to improper string definition, a previously successful description string will remain.

The **no** form of the command removes any current description string from the CCAG.

Default **None** (A description string must be explicitly defined)

Parameters *description-string* — Defines the string of printable ASCII characters, up to 80 characters that will be stored and displayed as a description for the *ccag-id* that the **description** command is executed under. The string must be entered in double quotation marks if the string contains spaces.

VSM CLI Tree Node Commands

vsm

Syntax [**no**] **vsm**

Context config

Description This command changes the current CLI context to the CCA nodal context. The CCA nodal context is where CCAGs are created and maintained.

The CCA nodal context always exists and cannot be removed.

ccag

Syntax **ccag** *ccag-id* [**create**]
no ccag *ccag-id* [**force**]

Context config>vsm

Description This command creates a Cross Connect Aggregation Group (CCAG). A CCAG represents a group of CCAs as a common forwarding entity. Objects requiring a CCA cross connect function are mapped to a CCAG, not the individual CCAs within the CCAG. The CCAG treats each active member CCA as a possible destination when forwarding packets between the cross connected objects mapped to the CCAG. The system uses both conversation hashing functions and direct service mappings to determine the load sharing distribution between the active CCAs. All packets for a given conversation flow through the same CCA to preserve packet order. Packet ordering may be momentarily affected during convergence events when CCAs are dynamically added or removed from the active list.

The CCAG context is used to manage the following functions per CCAG instance:

- Informational description of the CCAG
- Administrative state of the CCAG
- Alpha path bandwidth and weight parameters
- Beta path bandwidth and weight parameters
- CCA total bandwidth limit
- CCA membership in the CCAG

The **no** form of the command removes an existing *ccag-id* from the system. Once the specified *ccag-id* is removed from the system, it may not be referenced by any cross connect objects. If the **force** keyword is not specified, the **no ccag** *ccag-id* command will fail if the specified *ccag-id* has one or more *cc-ids* associated with it. In the event that the specified *ccag-id* does not exist, the **no ccag** *ccag-id* command will return to the current CLI context without any change to the system.

Default **None** (each CCAG context must be explicitly created to be used)

Parameters *ccag-id* — Identifies the CCAG instance that the system is creating or editing. Up to eight CCAGs may be created within the system. A *ccag-id* must be created on the system prior to creating cross connect object associations.

After a *ccag-id* is created, a CCAG SAP may be created with an association with the *ccag-id*. A CCAG SAP is identified by a concatenation of an existing *ccag-id* and a *cc-id*. The *cc-id* must match the *cc-id* of the other object the CCAG SAP is paired with on the *ccag-id*. The created *ccag-id* may also be associated with a network IP interface. A network IP interface is bound to the *ccag-id* through the port command in the config router interface ip-interface context and references the *ccag-id* and a *cc-id*. Again, the *cc-id* must match the other object the IP interface is paired with on the *ccag-id*.

Once created, the **ccag** *ccag-id* command may be executed to enter the *ccag-id* instance for the purpose of editing the CCAG parameters or operational state.

Values 1 through 8

create — The **create** keyword explicitly indicates that the specified *ccag-id* is being created. Handling the inclusion or exclusion state of the create keyword is dependent on the system environment variable create.

When the system environment variable create is enabled, the system requires the explicit use of the create keyword when creating objects such as a CCAG. If the keyword is not included and the *ccag-id* has not already been created, an error will occur and the CLI will remain at the current CLI context. This is designed to prevent the inadvertent creation of a CCAG instance in the event where the wrong *ccag-id* is specified during an attempt to edit an existing CCAG instance. If the create keyword is specified, the *ccag-id* will be created given the *ccag-id* is within the proper range for CCAG identifiers.

When the system environment variable create is disabled (using the no create command), the system will not require the create keyword when creating a CCAG instance. In the event that the ccag command is issued with a *ccag-id* that previously had not been created, that *ccag-id* will be considered available for cross connect associations and bindings.

Once a *ccag-id* has been created, the create keyword is ignored when a ccag command is executed with that *ccag-id*. The **ccag** *ccag-id* create command will only result in a CLI context change to the specified CCAG instance for a pre-existing *ccag-id.*

force — The **force** keyword removes the specified *ccag-id* regardless of the presence of one or more *cc-id*. If a SAP exists on the *ccag-id*, the force keyword will cause the SAP to be removed from the configuration. If a network IP interface is bound to the *ccag-id*, the interface will be silently unbound from the *ccag-id*. The force keyword is intended as a time saving feature, preventing the need to first remove all service and network associations with the *ccag-id*.
It is not required to first remove all CCAs from the CCAG prior to deleting the CCAG from the system. When the CCAG is removed, association with all member CCAs is automatically removed.

access

Syntax **access**

Context config>vsm>ccag *ccag-id*

Description This command changes the current CLI context to the CCAG access nodal context. The access nodal context contains the qos adaptation command used to control the SAP QoS distribution across the active member CCAs within the CCAG.

The CCAG access nodal context always exists and cannot be removed.

adapt-qos

Syntax **adapt-qos** {**link** | **distribute | port-fair**}
no adapt-qos

Context config>vsm>ccag *ccag-id*>access

Description This command controls how the CCAG SAP queue and virtual scheduler buffering and rate parameters are adapted over multiple active CCAs. Two adaptation modes are supported; link and distributed.

The **no** form of the command returns the CCAG access QoS adaptation rule to the default setting of distribute.

Default **distribute**

Parameters **link —** The **link** keyword is mutually exclusive with the **distribute** and **port-fair** keywords. When **link** is specified, the CCAG will create the SAP queues and virtual schedulers on each CCA with the actual parameters defined in the QoS and scheduler policies. This mode is useful when conversation hashing places all or most traffic over a single CCA.

distribute — The **distribute** keyword is mutually exclusive with the **link** and **port-fair** keywords. When **distribute** is specified, the CCAG SAP queues and schedulers on each CCA will receive a portion of the defined parameters in the QoS and scheduler policies. The portion is decided on an IOM basis with the ratio determined by the number of active CCA members on the IOM relative to the total number of active members within the CCAG. The following equation may be used to determine the actual ratio:

IOM-parameter-value = (IOM-active-CCA / total-active-CCA) * policy-parameter-value

port-fair — The **port-fair** keyword is mutually exclusive with the **link** and **distribute** keywords. When **port-fair** is specified, the CCAG SAP queues and schedulers on each CCA will receive a portion of the defined parameters in the QoS and scheduler policies. The portion is per-port basis and equals the value configured divided by the total number of active members within the CCAG. The following equation may be used to determine the actual ratio:

Per-port-parameter-value = (1 / total-active-CCA) * policy-parameter-value

cca-rate

Syntax **cca-rate** *kilobits-per-second*
no cca-rate

Context config cca>ccag *ccag-id*

Description This command defines a maximum forwarding rate for each CCA member within the CCAG. Support of setting a maximum CCA forwarding rate is provided to prevent overrunning the ingress forwarding plane when sub-line rate ingress features are enabled. The primary ingress feature requiring this support is dual ingress access queuing. When dual ingress queuing is enabled on cross connect SAPs, the CCA forwarding rate should be limited to a rate that prevents packet loss due to ingress forwarding congestion. The specified limit is applied to the aggregate alpha and beta path bandwidth.
The **no** form of the command removes CCA bandwidth rate limiting.

Parameters *kilobits-per-second* — Defines the maximum CCA rate in kilobits per second. The actual Kilobits per second rate is rounded up to the nearest 50Mbps increment.

Values 0 — 100000000, max

Default **max**

member-cca

Syntax [**no**] **member-cca** *card-slot/mda-number*

Context config>vsm>ccag *ccag-id*

Description This command adds and deletes provisioned CCAs from the CCAG. The only requirement to defining a CCA member is that the defined MDA position be provisioned as type cca. A CCA does not need to be populated in the defined MDA position prior to membership definition. A non-populated CCA member is considered inactive from a CCAG perspective. A populated CCA member will become active once it has been initialized by the system. A CCA member may be removed from the CCAG or depopulated from MDA slot at any time. At least one member CCA must be active on the CCAG for the CCAG to be placed in the operational state. Up to 8 member CCAs can be configured per CCAG.
The **no** form of the command removes a CCA member from the CCAG. If the CCA does not exist or is not currently a member of the CCAG, no error is returned. Once removed from the CCAG, all forwarding through the specified CCA stops.

Parameters *card-slot/mda-number* — Identifies the system MDA slot that is will be added as a member CCA for the CCAG. The specified MDA slot must have been pre-provisioned as type cca for the membership command to be successful.

card-slot — Defines the IOM slot the provisioned CCA is or will be populated. It is separated from the following mda-position portion of the parameter by a forward slash (/).

Values 1 through 10 (chassis type dependent)

mda-position — The mda-position portion of the parameter defines the MDA slot number on the IOM the CCA is or will be populated. It must be separated from the preceding card-slot portion of the parameter by a forward slash (/).

Values 1 or 2 (IOM type dependent)

VSM Path Commands

path

Syntax **path** {**a** | **b**}

Context config>vsm>ccag *ccag-id*

Description This command changes the current CLI context to the path nodal context. The CCA path nodal context is where each CCA path bandwidth, buffer and accounting parameters are maintained. The path context command must be specified with either the a or b keyword specifying the CCA path context to be entered.

Each CCA is divided into two distinct paths for bandwidth management purposes. One path is identified as alpha (a) and the other beta (b). The significance of each path for bandwidth distribution is dependent on the relative path weights each path is given in relationship to the other. A maximum path rate may also be defined allowing the provisioning of a maximum cap on the aggregate bandwidth allowed to the SAP or IP interface queues associated with the path. Each path is separated into three other contexts; SAP-2-SAP (sap-sap), SAP-2-Net (sap-net) and Net-2-SAP (net-sap). Each path context allows for the definition of the features that are usually associated with physical ports on other MDAs in the system. These include buffer pool management, ingress network queue definitions and accounting policy control.

The CCA path nodal contexts always exist and cannot be removed.

Parameters **a —** The **a** keyword is mutually exclusive to the **b** keyword and defines the CLI CCA path context to be the alpha path. Either the **a** or **b** path must be specified. If the **a** or **b** keyword is not present, the path command will fail without changing the current CLI context.

b — The **b** keyword is mutually exclusive to the **a** keyword and defines the CLI CCA path context to be the beta path. Either the **a** or **b** path must be specified. If the **a** or **b** keyword is not present, the path command will fail without changing the current CLI context.

rate

Syntax **rate** *kilobits-per-second* [**aggregate** | **cca**]
no rate

Context config>vsm>ccag *ccag-id*>path {a | b}

Description This command defines a specific bandwidth rate limitation for the alpha or beta paths on each member CCA in the CCAG. Use of the rate command is optional. When the rate command is not executed or the no rate command is executed, bandwidth allocated to the path is not limited to a specific rate.

Path limiting on a CCA prevents the aggregate bandwidth for the path from exceeding a certain rate. If the rate is exceeded, the CCA will backpressure all active egress queues sending on that path.

Access to the available bandwidth is dependent on the various parameters associated with each object egress queue.

The specified rate may be defined as an aggregate path rate for all CCAs in the CCAG or it may be defined as a per CCA path rate.

The **no** form of the command removes path rate limiting from all CCAs in the CCAG membership list for the path.

Default **None** (rate limiting the alpha path must be explicitly defined)

Parameters *kilobits-per-second* — Defines the path rate in kilobits per second. The **aggregate** and **cca** keywords specify how the defined rate is applied on a per CCA basis. The actual rate at each CCA is rounded up to the nearest 50Mbps.

Values 0 — 100000000, max

Default **max**

aggregate — The **aggregate** keyword is optional and mutually exclusive to the **cca** keyword. When **aggregate** is specified, the defined rate is divided among the CCAs in the CCAG member list based on the number of active CCAs. If three CCAs are active, the rate is divided by three and the result is applied to each active CCA. If a fourth CCA becomes active on the CCAG, the defined rate is then divided by four with the result applied to each CCA member on the CCAG. The actual rate at each CCA is implemented in 50Mbps increments. The system will adapt the specified rate to the best rate available per CCA.

Default When the kilobits-per-second parameter is specified, the default keyword is aggregate.

cca — The **cca** keyword is optional and mutually exclusive to the **aggregate** keyword. When **cca** is specified, the defined rate is applied to all CCAs in the CCAG member list. The actual rate at each CCA is implemented in 50Mbps increments. The system will adapt the specified rate to the best rate available per CCA.

weight

Syntax **weight** *path-weight*
no weight

Context config cca>ccag ccag-id>path {a | b}

Description This command defines a scheduling weight to the aggregate output of the alpha and beta paths. The specified weight is used to calculate a scheduling percentage for each path. The percentage for each path is based on:

Alpha scheduling percentage = alpha-path-weight / (alpha-path-weight + beta-path-weight)
Beta scheduling percentage = blue-path-weight / (alpha-path-weight + beta-path-weight)

Based on the above calculation, the sum of the alpha and beta scheduling percentage always equals 100 percent. When one path is not using all of its available scheduling bandwidth, the other path may use the remainder.

The **no** form of the command returns the path-weight for the path to the default value of 50.

Parameters *path-weight* — The path-weight parameter is required and is used by the system to determine the scheduling percentage for both paths. Changing the path-weight for one path affects both paths scheduling percentage. The resulting scheduling percentage changes are applied to all CCAs in the CCAG membership list.

Values 1 to 100

Default **50**

sap-sap

Syntax **sap-sap**

Context config cca>ccag ccag-id>path {a | b}

Description This command changes the current CLI context to the path SAP-SAP nodal context. This context contains the ingress and egress buffer pool configuration commands. The sap-sap>path context is associated with all SAPs defined on the CCAG path (alpha or beta depending on the path context) that cross connect to a SAP on the other path.

The CCA path SAP-SAP nodal context always exists and cannot be removed.

mac

Syntax [**no**] **mac** *mac-address*

Context config>vsm>ccag *ccag-id*>path {a | b}
config>vsm>ccag *ccag-id*>path {a | b}>sap-net
config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command overrides the default MAC address for the path's context.

The **no** form of the command returns the in-use MAC address for the path's context to the default MAC from the chassis MAC pool.

Parameters *mac-address* — Defines the IEEE MAC address that is to be associated with the path's context.

Values Any valid IEEE MAC source MAC address
(6 byte address expressed in hexadecimal notation with each byte separated by a dash (-)).

Default The path's default sap-sap MAC address is derived from the chassis MAC address pool.

mtu

Syntax **mtu** *mtu-size*
no mtu

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-sap
config>vsm>ccag *ccag-id*>path {a | b}>sap-net
config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command overrides the default port level MTU for the path's context.

The **no** form of the command returns the MTU for the path's sap-sap context to the default MTU.

Parameters *mtu-size* — Defines the Ethernet MTU that is to be associated with the path's context.

Default **1518** - sap-sap
1518 - sap-net
9212 - net-sap

Values 512 — 9212 bytes

egress

Syntax **egress**

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-sap
config>vsm>ccag *ccag-id*>path {a | b}>sap-net
config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command changes the current CLI context to the path's context. This context contains the egress buffer pool configuration commands.

The CCA path's egress nodal context always exists and cannot be removed.

pool

Syntax **pool**

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-sap>egress
config>vsm>ccag *ccag-id*>path {a | b}>sap-sap>ingress
config>vsm>ccag *ccag-id*>path {a | b}>sap-net>egress
config>vsm>ccag *ccag-id*>path {a | b}>sap-net>ingress
config>vsm>ccag *ccag-id*>path {a | b}>net-sap>egress

Description This command changes the current CLI context to the path's nodal context. This context contains the egress buffer pool configuration commands. The CCA path's egress or ingress pool nodal context always exists and cannot be removed.

resv-cbs

Syntax [**no**] **resv-cbs percentage-of-pool**

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-sap>egress>pool
config>vsm>ccag *ccag-id*>path {a | b}>sap-sap>ingress>pool
config>vsm>ccag *ccag-id*>path {a | b}>sap-net>egress>pool
config>vsm>ccag *ccag-id*>path {a | b}>sap-net>ingress>pool
config>vsm>ccag *ccag-id*>path {a | b}>net-sap>egress>pool

Description This command defines the percentage of the buffer pool that is considered reserved for the CBS buffer allocation for queues created in the path's pool context.

The **no** form of the command returns the reserved portion of the buffer pool to the default percentage.

Parameters *percentage-of-pool* — The percentage-of-pool parameter defines the percentage of the buffer pool that is not considered shared. The shared portion of the pool is used by queues that have crossed their CBS buffer threshold and is subject to the WRED slope functions. The reserved portion of the pool is used by queues that have not crossed their CBS threshold. The aggregate CBS on the queues associated with the pool may oversubscribe the resv-cbs percentage. If the reserved portion is oversubscribed and the in-use reserved buffers exceed the defined percentage, buffers are removed from the shared portion of the pool.

Values 1 to 100 (percent)

Default **30**

slope-policy

Syntax **slope-policy slope-policy-name**
no slope-policy

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-sap>egress>pool
config>vsm>ccag *ccag-id*>path {a | b}>sap-sap>ingress>pool
config>vsm>ccag *ccag-id*>path {a | b}>sap-net>egress>pool
config>vsm>ccag *ccag-id*>path {a | b}>sap-net>ingress>pool
config>vsm>ccag *ccag-id*>path {a | b}>net-sap>egress>pool

Description This command defines the slope policy used to mange the shared portion of the buffer pools WRED slopes. The commands in the policy control the administrative state of the slopes, the start and knee points of each slope and the time-average-factor for the weighted average buffer utilization calculation.

The **no** form of the command configures the default slope policy as the managing policy for the buffer pool.

Parameters *slope-policy-name* — Defines the name of the WRED slope policy used to manage the WRED slopes in the shared portion of the buffer pool.

Values Any existing slope policy name.

ingress

Syntax **ingress**

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-sap
config>vsm>ccag *ccag-id*>path {a | b}>sap-net

Description This command changes the current CLI context to the path's context. This context contains the ingress buffer pool configuration commands.
The CCA path's ingress nodal context always exists and cannot be removed.

sap-net

Syntax **sap-net**

Context config>vsm>ccag *ccag-id*>path {a | b}

Description This command changes the current CLI context to the path sap-net nodal context. This context contains the ingress and egress buffer pool configuration commands. The sap-net>path context is associated with all SAPs defined on the CCAG path (alpha or beta depending on the path context) that cross connect to a network IP interface on the other path.
The CCA path sap-net nodal context always exists and cannot be removed.

slope-policy

Syntax **slope-policy** *slope-policy-name*
no slope-policy

Context config>vsm>ccag *ccag-id*>path {a | b}>sap-net>ingress>pool

Description This command defines the slope policy used to mange the shared portion of the buffer pools WRED slopes. The commands in the policy control the administrative state of the slopes, the start and knee points of each slope and the time-average-factor for the weighted average buffer utilization calculation.

The **no** form of the command configures the default slope policy as the managing policy for the buffer pool.

Parameters *slope-policy-name* — The slope-policy-name parameter defines the name of the WRED slope policy used to manage the WRED slopes in the shared portion of the buffer pool.

Values Any existing slope policy name.

net-sap

Syntax **net-sap**

Context config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command changes the current CLI context to the path net-sap nodal context. The net-sap nodal context contains the network accounting and queue policies and the egress buffer pool configuration commands. The net-sap path context is associated with all network IP interfaces bound to the CCAG path (alpha or beta depending on the path context) that cross connects to a SAP on the other path. The CCA path net-sap nodal context always exists and cannot be removed.

mtu

Syntax **mtu** *mtu-size*
no mtu

Context config>vsm>ccag *ccag-id*>path {a | b}>

Description This command overrides the default port level MTU for the path's net-sap context.

The **no** form of the command returns the MTU for the path's net-sap context to the default MTU.

Parameters *mtu-size* — The mtu-size, in bytes, defines the Ethernet MTU that is to be associated with the path's net-sap context.

Default **1522**

accounting-policy

Syntax **accounting-policy** *accounting-policy*
no accounting-policy

Context config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command defines the network accounting policy that will be used to define which statistics will be collected when the **collect-stats** command is enabled in the path's net-sap context.

The **no** form of the command reverts the path's net-sap context statistics billing collection to the statistics defined in the default network accounting policy.

Parameters *accounting-policy* — The accounting-policy parameter is required and identifies which set of statistics will be collected for billing output.

Values Any existing network accounting policy in the system.

Default **The default network accounting policy**

collect-stats

Syntax [**no**] **collect-stats**

Context config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command enables collecting stats on the path's net-sap context. When enabled the statistics defined in the accounting-policy accounting-policy command will be collected according to the specifications in the policy.

The **no** form of the command disables network billing statistics collection on the net-sap context.

Default Network statistics are not collected by default on the net-sap context.

queue-policy

Syntax **queue-policy** *queue-policy-name*
no queue-policy

Context config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command defines the egress network queues used by IP interfaces bound to the path's net-sap context. The specified *queue-policy-name* defines the number of queues, the rate and buffering parameters for the queues and the forwarding class mappings to the queues.

The **no** form of the command reverts the path's net-sap network IP interface queues to the systems default queue policy.

Parameters *queue-policy-name* — Specifies which existing Queue Policy will define the queuing structure for network IP interfaces bound to the path's net-sap context.

Values Any existing queue policy on the system.

Default The default queue policy is used when another is not specified.

egress

Syntax **egress**

Context config>vsm>ccag *ccag-id*>path {a | b}>net-sap

Description This command changes the current CLI context to the path>net-sap>egress nodal context. This context contains the egress buffer pool configuration commands.

The CCA path net-sap egress nodal context always exists and cannot be removed.

pool

Syntax **pool**

Context config>vsm>ccag *ccag-id*>path {a | b}>net-sap>egress

Description This command changes the current CLI context to the path>net-sap>egress pool>nodal context. This context contains the egress buffer pool configuration commands.

The CCA path net-sap egress pool nodal context always exists and cannot be removed.

Related Commands

Refer to the 7450 ESS OS Interface Configuration Guide for more card, MDA, and port command information. Refer to the 7450 ESS OS Services Guide for details about configuring specific service parameters.

mda

Syntax **mda** *mda-slot*
no mda *mda-slot*

Context config>card

Description This command provisions an adaptor into an MDA position on an IOM slot. The provisioned MDA may or may not exist in the system at the time of provisioning. If the provisioned MDA does not currently exist in the specified MDA position number, it is considered to be a 'ghost' MDA. Ports and other resources on a ghost MDA may be configured once the MDA is provisioned. When a proper MDA matching the provisioned MDA type is inserted into the IOM MDA position, forwarding though the MDA based on configured services or network interface will be available once the MDA has been properly initialized.

A Versatile Service Module (VSM) is provisioned into the system in the same manner as all other adaptors using MDA slots. Once a VSM is provisioned, independent of it actually existing in the system on the specified slot and MDA position, the VSM may be defined as a member of a CCAG (Cross Connect Adaptor Group). A VSM inserted into the system prior to provisioning is not available for CCAG membership and will be treated as an unprovisioned MDA.

Once a VSM is provisioned and populated in the system, it cannot be used until it has been defined membership into a CCAG. When the CCAG membership has been defined for the VSM, the various internal resources of the VSM will be configured according to the CCAG bandwidth control parameters. This includes the alpha and beta path weights, the alpha and beta path maximum rates and the aggregate alpha and beta maximum rate. A VSM-CCA-XP may be configured as either a VSM-CCA MDA or a VSM-CCA-XP MDA. When configured as a VSM-CCA-XP it is not a member of a CCA Group (ref VSM-CCA-XP).

The **no** form of the command unprovisions an MDA from the system. For a VSM to be unprovisioned, the VSM must not be a member of a CCAG. If the VSM is a member of a CCAG, the **no cca** *slot-number/mda-number* command must be used in the CCAG member-list context. Once a CCA is unprovisioned from the system; it cannot be made a member of a CCAG until it has been reprovisioned.

Default **None** (An MDA position number must be explicitly specified.)

Parameters *mda-slot* — Defines the position on the card slot-number the CCA will be populated into. On the iom-20g IOM module, two MDA positions are available. Future IOMs will support a different number of MDA positions.

Values 1 or 2

port

Syntax **port ccag**-*ccag-id*.{**a** | **b**}[.*net-sa*p]:*cc-id*
no port

Context config>router>interface *ip-interface-name*

Description This command cross connects a network IP interface to a CCAG SAP using the referenced *ccag-id.* A CCAG network IP interface binding is identified by four items; the *ccag-id,* the CCAG path, the pairing type and the *cc-id.* A network IP interface CCAG port binding supports all the available features as port binding using a Dot1Q virtual interface.

To support cross connection between services and network IP interfaces, the network interface port command allows the binding of the IP interface to a *ccag cc-id*. Similar to service CCAG SAPs, the network IP interface port binding command must reference the *ccag-id*, the CCA path (.a or .b) and the *cc-id* used by the service CCAG SAP on the other CCA path. The pairing type is optional as only *.net-sap* is supported.

The **no** form of the command removes the CCAG binding from the network IP interface.

Parameters **ccag —** The ccag portion of the port binding is required and specifies that the network IP interface is binding to a **ccag** *cc-id*.

ccag-id — The *ccag-id* portion of the port binding is required and specifies which *ccag-id* the network IP interface must be bound to. The specified *ccag-id* must exist on the system or the port binding will fail. The leading dash must be included as a separator between ccag and the *ccag-id*.

Values -1 (dash 1) to -8 (dash 8)

Default **None**

.a | .b — The .a and .b portion of the port binding is required and is used to define the CCA bandwidth path the network IP interface will be associated with. The path association must be specified and .a and .b are mutually exclusive. The .a designation identifies the network IP interface as being on the Alpha path and the .b designation identifies the network IP interface as being on the Beta path. The paired SAP using the same *cc-id* as the bound network IP interface must be associated with the opposite path. The leading period must be included as a separator between the *ccag-id* and the path designator.

Values .a or .b

Default **None**

.net-sap — The .net-sap portion of the network IP interface CCAG binding is optional and is used to explicitly define the pairing type as Net-2-SAP. A cross connection between two network IP interfaces is not currently allowed. The .net-sap pairing type is assumed and does not need to be included in the SAP identification. When specified, the leading period must be used as a separator between the path designator and the pairing type.

Default **.net-sap**

:cc-id — The *:cc-id* portion of the port binding is required and specifies the unique *cc-id* in use by the CCAG network IP interface port binding and the cross connect SAP on the other path.

Values 1 to 4094

Services Commands

Service CCAG SAP Provisioning

Services are provisioned onto a CCAG using a special CCAG SAP definition. CCAG SAPs must reference a *ccag-id*, a CCA path (a or b), a pairing type (sap-sap or sap-net) and a unique *cc-id*. The *ccag-id* identifies the group of CCAs that will be used for forwarding packets associated with the SAP. The path identifies the bandwidth control grouping used to manage CCA egress bandwidth. The pairing type helps the system identify which buffering resources will be used to manage egress queuing of packets. Finally, the *cc-id* is used to explicitly cross connect the SAP to another SAP or network IP interface configured with the same *cc-id*.

sap

Syntax **sap ccag**-*ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id* [**create**]
no sap ccag-*ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id*

Context config>service>epipe

Description This command creates a cross connect SAP on the *ccag-id* referenced in the Epipe service. A CCAG SAP is identified by four items; the *ccag-id*, the CCAG path, the pairing type and the *cc-id*. An Epipe CCAG SAP supports all the available QoS, filtering and accounting features as an Epipe Dot1Q SAP.

The **no** form of the command removes a SAP from a service context. Once removed, all information and resources concerning the SAP is deleted from the system including the CCAG *cc-id* in use on the CCA path.

Parameters **ccag —** The ccag portion of the SAP identifier is required and specifies that the Epipe SAP is of the CCAG type.

-*ccag-id* — The *ccag-id* portion of the SAP identifier is required and specifies which *ccag-id* on which the SAP must be created. The specified *ccag-id* must exist on the system or the SAP creation will fail. The leading dash must be included as a separator between **ccag** and the *ccag-id*.

Values -1 (dash 1) to -8 (dash 8)

Default **None**

.a | .b — The .a and .b portion of the CCAG SAP identifier is required and is used to define the CCA bandwidth path with will be associated with the SAP. The path association must be specified and .a and .b are mutually exclusive. The .a designation identifies the SAP as being on the Alpha path and the .b designation identifies the SAP as being on the Beta path. The paired SAP or network

IP interface using the same *cc-id* as the SAP must be associated with the opposite path. The leading period must be included as a separator between the *ccag-id* and the path designator.

Values .a or .b

Default **None**

.sap-net — The .sap-net portion of the CCAG SAP identifier specifies that the SAP is of the SAP-2-Net pairing type and is required when the *cc-id* is paired with a network IP interface. The pairing type .sap-net is mutually exclusive with pairing type .sap-sap. If .sap-net is not specified, .sap-sap is assumed and does not need to be included in the SAP identification. When specified, the leading period must be used as a separator between the path designator and the pairing type.

Values .sap-net or .sap-sap

Default **.sap-sap**

.sap-sap — The .sap-sap portion of the CCAG SAP identifier is mutually exclusive to .sap-sap and is used to define the pairing type as SAP-2-SAP. The .sap-sap pairing type is only used when the cross connect object sharing the same *cc-id* on the opposite path is a CCAG SAP. If the other cross connect object is a network IP interface, the pairing type must be defined as .sap-net. If .sap-net is not specified, .sap-sap is assumed and does not need to be included in the SAP identification. When specified, the leading period must be used as a separator between the path designator and the pairing type.

Values .sap-net or .sap-sap

Default **.sap-sap**

:cc-id — The *:cc-id* portion of the CCAG SAP identifier is required and specifies the unique *cc-id* in use by the CCAG SAP and the cross connect object on the other path.

Values 1 to 4094

Default **None**

create — Explicitly indicates that the specified CCAG SAP is being created by the **sap** command. Handling the inclusion or exclusion state of the create keyword is dependent on the system environment variable create.

When the system environment variable create is enabled, the system requires the explicit use of the **create** keyword when creating objects such as SAPs. If the keyword is not included and the specified CCAG SAP has not already been created, an error will occur and the CLI will not change context to the specified CCAG SAP instance. This is designed to prevent the inadvertent creation of a CCAG SAP in the event where the wrong CCAG SAP identifier is specified during an attempt to edit an existing CCAG SAP. If the **create** keyword is specified, the CCAG SAP will be created if it does not already exist or if it does exist, the CLI context will change to the specified CCAG SAP.

When the system environment variable create is disabled (using the **no create** command), the system will not require the **create** keyword when creating a CCAG SAP. In the event that the **sap** command is issued with a CCAG SAP identifier that previously had not been created, that CCAG SAP will be created.

Once a CCAG SAP has been created, the **create** keyword is ignored when a **sap** command is executed with that CCAG SAP identifier and the CLI context will change to the specified CCAG SAP.

vsm-cca-xp — In addition to supporting all the features of the existing VSM-CCA, the new VSM-CCA-XP MDA offers a new hybrid mode for simplified provisioning and a higher capacity VSM when inserted on IOM3-XP cards. As with the CSM-CCA MDA, the complete forwarding path bandwidth (in this case 25G) is available allowing single coversations up to 25G on a single MDA.

The use cases for VSM-CCA-XP are nearly identical to the VSM-CCA. When configured as a VSM-CCA-XP port x/x1 and port x/x/2 are interally connected. Therefore, configuration is very similar to a physical loop back port using Ethernet with dot1Q encapsulation. THe use of hybrid port removes the requirement to configure net and sap parameters and simplifies provisioning. The use of the Ethernet VLAN tag is used to connect the SAPs.

VSM-CCA-XP Exceptions:

While LAG is available, LACP is not allowed.

Ethernet CFM is only available when Eth-Rings are configured on the VSM (Ethernet rings use Ethernet MEPS for control).

The new VSM-CCA-XP can be configured as a VSM-CCA MDA to support CCA functions on IOM1, IOM2 and IOM3. On IOM3, the VSM-CCA MDA supports a loop back mode that uses LAG and two ports using Ethernet as the internal connection. The LAG feature also conversations hashing just as the original VSM-CCA. The hybrid port mode eliminates the need to specify network or access modes.

sap

Syntax **sap ccag**-*ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id* [**create**]
no sap ccag-*ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id*

Context config>service>vpls

Description This command creates a cross connect SAP on the *ccag-id* referenced in the VPLS service. A CCAG SAP is identified by four items; the *ccag-id*, the CCAG path, the pairing type and the *cc-id*. A VPLS CCAG SAP supports all the available QoS, filtering and accounting features as a VPLS Dot1Q SAP.

The **no** form of the command removes a SAP from a service context. Once removed, all information and resources concerning the SAP is deleted from the system including the CCAG *cc-id* in use on the CCA path.

Parameters **ccag —** The ccag portion of the SAP identifier is required and specifies that the vpls SAP is of the CCAG type.

-*ccag-id* — Specifies which *ccag-id* on which the SAP must be created. The specified *ccag-id* must exist on the system or the SAP creation will fail. The leading dash must be included as a separator between **ccag** and the *ccag-id*.

Values -1 (dash 1) to -8 (dash 8)

Default **None**

.a | .b — The **.a** and **.b** portion of the CCAG SAP identifier is required and is used to define the CCA bandwidth path with will be associated with the SAP. The path association must be specified and .a and .b are mutually exclusive. The .a designation identifies the SAP as being on the Alpha path

and the .b designation identifies the SAP as being on the Beta path. The paired SAP or network IP interface using the same *cc-id* as the SAP must be associated with the opposite path. The leading period must be included as a separator between the *ccag-id* and the path designator.

Values .a or .b

Default **None**

.sap-net — The .sap-net portion of the CCAG SAP identifier specifies that the SAP is of the SAP-2-Net pairing type and is required when the *cc-id* is paired with a network IP interface. The pairing type .sap-net is mutually exclusive with pairing type .sap-sap. If .sap-net is not specified, .sap-sap is assumed and does not need to be included in the SAP identification. When specified, the leading period must be used as a separator between the path designator and the pairing type.

Values .sap-net or .sap-sap

Default **.sap-sap**

.sap-sap — The .sap-sap portion of the CCAG SAP identifier is mutually exclusive to .sap-sap and is used to define the pairing type as SAP-2-SAP. The .sap-sap pairing type is only used when the cross connect object sharing the same *cc-id* on the opposite path is a CCAG SAP. If the other cross connect object is a network IP interface, the pairing type must be defined as .sap-net. If .sap-net is not specified, .sap-sap is assumed and does not need to be included in the SAP identification. When specified, the leading period must be used as a separator between the path designator and the pairing type.

Values .sap-net or .sap-sap

Default **.sap-sap**

:cc-id — The :*cc-id* portion of the CCAG SAP identifier is required and specifies the unique *cc-id* in use by the CCAG SAP and the cross connect object on the other path.

Values 0 1 to 4094

Default **None**

create — Explicitly indicates that the specified CCAG SAP is being created by the **sap** command. Handling the inclusion or exclusion state of the create keyword is dependent on the system environment variable create.

When the system environment variable create is enabled, the system requires the explicit use of the **create** keyword when creating objects such as SAPs. If the keyword is not included and the specified CCAG SAP has not already been created, an error will occur and the CLI will not change context to the specified CCAG SAP instance. This is designed to prevent the inadvertent creation of a CCAG SAP in the event where the wrong CCAG SAP identifier is specified during an attempt to edit an existing CCAG SAP. If the **create** keyword is specified, the CCAG SAP will be created if it does not already exist or if it does exist, the CLI context will change to the specified CCAG SAP.

When the system environment variable create is disabled (using the **no create** command), the system will not require the **create** keyword when creating a CCAG SAP. In the event that the **sap** command is issued with a CCAG SAP identifier that previously had not been created, that CCAG SAP will be created.

Once a CCAG SAP has been created, the **create** keyword is ignored when a **sap** command is executed with that CCAG SAP identifier and the CLI context will change to the specified CCAG SAP.

sap

Syntax **sap ccag**-*ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id* [**create**]
no sap ccag-*ccag-id*.{**a** | **b**}[.*sap-net* | .*sap-sap*]:*cc-id*

Context config>service>ies>interface

Description This command creates a cross connect SAP on the *ccag-id* referenced in the IES service. A CCAG SAP is identified by four items; the *ccag-id*, the CCAG path, the pairing type and the *cc-id*. A CCAG SAP on an IES IP interface supports all the available QoS, filtering and accounting features as an IES IP interface Dot1Q SAP.

The **no** form of the command removes a SAP from the IES service IP interface context. Once removed, all information and resources concerning the SAP is deleted from the system including the CCAG *cc-id* in use on the CCA path.

Parameters **ccag —** The **ccag** portion of the SAP identifier is required and specifies that the ies SAP is of the CCAG type.

ccag-id — The *ccag-id* portion of the SAP identifier is required and specifies which *ccag-id* on which the SAP must be created. The specified *ccag-id* must exist on the system or the SAP creation will fail. The leading dash must be included as a separator between ccag and the *ccag-id*.

Values -1 (dash 1) to -8 (dash 8)

Default **None**

.a | .b — The .a and .b portion of the CCAG SAP identifier is required and is used to define the CCA bandwidth path with will be associated with the SAP. The path association must be specified and .a and .b are mutually exclusive. The .a designation identifies the SAP as being on the Alpha path and the .b designation identifies the SAP as being on the Beta path. The paired SAP or network IP interface using the same *cc-id* as the SAP must be associated with the opposite path. The leading period must be included as a separator between the *ccag-id* and the path designator.

Values .a or .b

Default **None**

.*sap-sap* — The .sap-sap portion of the CCAG SAP identifier is optional and is used to explicitly define the pairing type as SAP-2-SAP. The .sap-sap pairing type is only used when the cross connect object sharing the same *cc-id* on the opposite path is a CCAG SAP. A cross connection between an IES CCAG SAP and a network IP interface is not currently allowed. If .sap-sap is not specified, .sap-sap is assumed and does not need to be included in the SAP identification. When specified, the leading period must be used as a separator between the path designator and the pairing type.

Default **.sap-sap**

:cc-id — The :*cc-id* portion of the CCAG SAP identifier is required and specifies the unique *cc-id* in use by the CCAG SAP and the cross connect object on the other path.

Values 1 to 4094

Default **None**

create — Explicitly indicates that the specified CCAG SAP is being created by the **sap** command. Handling the inclusion or exclusion state of the create keyword is dependent on the system environment variable create.

When the system environment variable create is enabled, the system requires the explicit use of the **create** keyword when creating objects such as SAPs. If the keyword is not included and the specified CCAG SAP has not already been created, an error will occur and the CLI will not change context to the specified CCAG SAP instance. This is designed to prevent the inadvertent creation of a CCAG SAP in the event where the wrong CCAG SAP identifier is specified during an attempt to edit an existing CCAG SAP. If the **create** keyword is specified, the CCAG SAP will be created if it does not already exist or if it does exist, the CLI context will change to the specified CCAG SAP.

When the system environment variable create is disabled (using the **no create** command), the system will not require the **create** keyword when creating a CCAG SAP. In the event that the **sap** command is issued with a CCAG SAP identifier that previously had not been created, that CCAG SAP will be created.

Once a CCAG SAP has been created, the **create** keyword is ignored when a **sap** command is executed with that CCAG SAP identifier and the CLI context will change to the specified CCAG SAP.

Standards and Protocol Support

Note that the information presented is subject to change without notice.
Alcatel-Lucent assumes no responsibility for inaccuracies contained herein.

Ethernet Standards

IEEE 1588 Precision Clock Synchronization Protocol
IEEE 802.1AB Station and Media Access Control Connectivity Discovery
IEEE 802.1ad Provider Bridges
IEEE 802.1ag Connectivity Fault Management
IEEE 802.1ah Provider Backbone Bridges
IEEE 802.1ak Multiple Registration Protocol
IEEE 802.1aq Shortest Path Bridging
IEEE 802.1ax Link Aggregation
IEEE 802.1D MAC Bridges
IEEE 802.1p Traffic Class Expediting
IEEE 802.1Q Virtual LANs
IEEE 802.1s Multiple Spanning Trees
IEEE 802.1w Rapid Reconfiguration of Spanning Tree
IEEE 802.1X Port Based Network Access Control
IEEE 802.3ab 1000BASE-T
IEEE 802.3ac VLAN Tag
IEEE 802.3ad Link Aggregation
IEEE 802.3ae 10 Gb/s Ethernet
IEEE 802.3ah Ethernet in the First Mile
IEEE 802.3ba 40 Gb/s and 100 Gb/s Ethernet
IEEE 802.3i Ethernet
IEEE 802.3u Fast Ethernet
IEEE 802.3x Ethernet Flow Control
IEEE 802.3z Gigabit Ethernet
ITU-T G.8031 Ethernet Linear Protection Switching
ITU-T G.8032 Ethernet Ring Protection Switching
ITU-T Y.1731 OAM functions and mechanisms for Ethernet based networks

OSPF

RFC 1586 Guidelines for Running OSPF Over Frame Relay Networks
RFC 1765 OSPF Database Overflow
RFC 2328 OSPF Version 2
RFC 3101 The OSPF Not-So-Stubby Area (NSSA) Option
RFC 3509 Alternative Implementations of OSPF Area Border Routers
RFC 3623 Graceful OSPF Restart (Helper Mode)
RFC 3630 Traffic Engineering (TE) Extensions to OSPF Version 2
RFC 4203 OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)
RFC 4222 Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance
RFC 4576 Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS IP Virtual Private Networks (VPNs)
RFC 4970 Extensions to OSPF for Advertising Optional Router Capabilities
RFC 5185 OSPF Multi-Area Adjacency
RFC 5243 OSPF Database Exchange Summary List Optimization
RFC 5250 The OSPF Opaque LSA Option
RFC 5709 OSPFv2 HMAC-SHA Cryptographic Authentication
RFC 6987 OSPF Stub Router Advertisement

BGP

RFC 1397 BGP Default Route Advertisement
RFC 1772 Application of BGP in the Internet
RFC 1965 Confederations for BGP
RFC 1997 BGP Communities Attribute
RFC 2385 Protection of BGP Sessions via MD5
RFC 2439 BGP Route Flap Dampening
RFC 2858 Multiprotocol Extensions for BGP-4
RFC 2918 Route Refresh Capability for BGP-4
RFC 3107 Carrying Label Information in BGP-4
RFC 3392 Capabilities Advertisement with BGP4
RFC 4271 BGP-4 (previously RFC 1771)
RFC 4360 BGP Extended Communities Attribute
RFC 4364 BGP/MPLS IP Virtual Private Networks (VPNs)(previously RFC 2547bis BGP/MPLS VPNs)
RFC 4456 BGP Route Reflection: Alternative to Full-mesh IBGP
RFC 4486 Subcodes for BGP Cease Notification Message
RFC 4577 OSPF as the Provider/ Customer Edge Protocol for BGP/ MPLS IP Virtual Private Networks (VPNs)
RFC 4659 BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN
RFC 4684 Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)
RFC 4724 Graceful Restart Mechanism for BGP – GR helper
RFC 4760 Multi-protocol Extensions for BGP
RFC 4798 Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)
RFC 4893 BGP Support for Four-octet AS Number Space
RFC 5004 Avoid BGP Best Path Transitions from One External to Another
RFC 5065 Confederations for BGP (obsoletes 3065)
RFC 5291 Outbound Route Filtering Capability for BGP-4

RFC 5575 Dissemination of Flow Specification Rules
RFC 5668 4-Octet AS Specific BGP Extended Community
draft-ietf-idr-add-paths Advertisement of Multiple Paths in BGP
draft-ietf-idr-best-external Advertisement of the Best External Route in BGP

IS-IS

ISO/IEC 10589:2002, Second Edition, Nov. 2002 Intermediate System to Intermediate System Intra-Domain Routeing Information Exchange Protocol
RFC 1195 Use of OSI IS-IS for Routing in TCP/IP and Dual Environments
RFC 2973 IS-IS Mesh Groups
RFC 3359 Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System
RFC 3719 Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)
RFC 3787 Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)
RFC 4971 Intermediate System to Intermediate System (IS-IS) Extensions for Advertising Router Information
RFC 5120 M-ISIS: Multi Topology (MT) Routing in IS-IS
RFC 5130 A Policy Control Mechanism in IS-IS Using Administrative Tags
RFC 5301 Dynamic Hostname Exchange Mechanism for IS-IS
RFC 5302 Domain-wide Prefix Distribution with Two-Level IS-IS
RFC 5303 Three-Way Handshake for IS-IS Point-to-Point Adjacencies
RFC 5304 IS-IS Cryptographic Authentication
RFC 5305 IS-IS Extensions for Traffic Engineering TE
RFC 5306 Restart Signaling for IS-IS (Helper Mode)
RFC 5307 IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)
RFC 5309 Point-to-Point Operation over LAN in Link State Routing Protocols
RFC 5310 IS-IS Generic Cryptographic Authentication
RFC 6213 IS-IS BFD-Enabled TLV
RFC 6329 IS-IS Extensions Supporting IEEE 802.1aq Shortest Path Bridging
draft-ietf-isis-mi-02 IS-IS Multi-Instance

IP, LDP, and Segment Routing Fast Reroute (FRR)

RFC 5286 Basic Specification for IP Fast Reroute: Loop-Free Alternates
draft-ietf-isis-segment-routing-extensions-03 IS-IS Extensions for Segment Routing
draft-ietf-rtgwg-lfa-manageability-07 Operational management of Loop Free Alternates
draft-ietf-rtgwg-remote-lfa-09 Remote LFA FRR
draft-katran-mofrr-02 Multicast only Fast Re-Route

IPSec

RFC 2401 Security Architecture for the Internet Protocol
RFC 2406 IP Encapsulating Security Payload (ESP)
RFC 2409 The Internet Key Exchange (IKE)
RFC 2560 X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP
RFC 3706 IKE Dead Peer Detection
RFC 3947 Negotiation of NAT-Traversal in the IKE
RFC 3948 UDP Encapsulation of IPsec ESP Packets
RFC 4210 Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)
RFC 4211 Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)
RFC 5996 Internet Key Exchange Protocol Version 2 (IKEv2)
RFC 5998 An Extension for EAP-Only Authentication in IKEv2
draft-ietf-ipsec-isakmp-xauth-06 Extended Authentication within ISAKMP/Oakley (XAUTH)
draft-ietf-ipsec-isakmp-modecfg-05 The ISAKMP Configuration Method

IPv6

RFC 1981 Path MTU Discovery for IPv6
RFC 2375 IPv6 Multicast Address Assignments
RFC 2460 Internet Protocol, Version 6 (IPv6) Specification
RFC 2461 Neighbor Discovery for IPv6
RFC 2462 IPv6 Stateless Address Auto configuration
RFC 2464 Transmission of IPv6 Packets over Ethernet Networks
RFC 2529 Transmission of IPv6 over IPv4 Domains without Explicit Tunnels
RFC 2545 Use of BGP-4 Multiprotocol Extension for IPv6 Inter-Domain Routing
RFC 2710 Multicast Listener Discovery (MLD) for IPv6
RFC 2740 OSPF for IPv6
RFC 3306 Unicast-Prefix-based IPv6 Multicast Addresses
RFC 3315 Dynamic Host Configuration Protocol for IPv6
RFC 3587 IPv6 Global Unicast Address Format
RFC 3590 Source Address Selection for the Multicast Listener Discovery (MLD) Protocol
RFC 3810 Multicast Listener Discovery Version 2 (MLDv2) for IPv6
RFC 3971 SEcure Neighbor Discovery (SEND)
RFC 3972 Cryptographically Generated Addresses (CGA)
RFC 4007 IPv6 Scoped Address Architecture
RFC 4193 Unique Local IPv6 Unicast Addresses
RFC 4291 IPv6 Addressing Architecture
RFC 4443 Internet Control Message Protocol (ICMPv6)
for the Internet Protocol Version 6 (IPv6) Specification
RFC 4552 Authentication/Confidentiality for OSPFv3

RFC 4659 BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN
RFC 5072 IP Version 6 over PPP
RFC 5095 Deprecation of Type 0 Routing Headers in IPv6
RFC 5187 OSPFv3 Graceful Restart (Helper Mode)
RFC 5308 Routing IPv6 with IS-IS
RFC 5340 OSPF for IPv6
RFC 5838 Support of Address Families in OSPFv3

Multicast

RFC 1112 Host Extensions for IP Multicasting (Snooping)
RFC 2236 Internet Group Management Protocol, (Snooping)
RFC 2362 Protocol Independent Multicast-Sparse Mode (PIMSM)
RFC 3376 Internet Group Management Protocol, Version 3 (Snooping)
RFC 3446 Anycast Rendevous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)
RFC 3618 Multicast Source Discovery Protocol (MSDP)
RFC 3956 Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address
RFC 4601 Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)
RFC 4604 Using IGMPv3 and MLDv2 for Source-Specific Multicast
RFC 4607 Source-Specific Multicast for IP
RFC 4608 Source-Specific Protocol Independent Multicast in 232/8
RFC 4610 Anycast-RP Using Protocol Independent Multicast (PIM)
RFC 4624 Multicast Source Discovery Protocol (MSDP) MIB
RFC 5059 Bootsrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)
RFC 5059 Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)
RFC 5384 The Protocol Independent Multicast (PIM) Join Attribute Format
RFC 5496 The Reverse Path Forwarding (RPF) Vector TLV
RFC 6037 Cisco Systems' Solution for Multicast in MPLS/BGP IP VPNs
RFC 6513 Multicast in MPLS/BGP IP VPNs
RFC 6514 BGP Encodings and Procedures for Multicast in MPLS/ IP VPNs
RFC 6515 IPv4 and IPv6 Infrastructure Addresses in BGP Updates for Multicast VPNs
RFC 6516 IPv6 Multicast MVPN Support Using PIM Control Plane and Selective Provider Multicast Service Interface (S-PMSI) Join Messages
RFC 6625 Wildcards in Multicast VPN Auto-Discover Routes
RFC 6826 Multipoint LDP In-Band Signaling for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Path
RFC 7246 Multipoint Label Distribution Protocol In-Band Signaling in a Virtual Routing and Forwarding (VRF)
RFC 7385 IANA Registry for P-Multicast Service Interface (PMSI) Tunnel Type Code Points
draft-dolganow-l3vpn-mvpn-expl-track-00 Explicit tracking in MPLS/BGP IP VPN

MPLS — GENERAL

RFC 2430 A Provider Architecture DiffServ & TE
RFC 2474 Definition of the DS Field the IPv4 and IPv6 Headers (Rev)
RFC 2597 Assured Forwarding PHB Group (rev3260)
RFC 2598 An Expedited Forwarding PHB
RFC 3031 MPLS Architecture
RFC 3032 MPLS Label Stack Encoding
RFC 3140 Per-Hop Behavior Identification Codes
RFC 3443 Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks
RFC 4023 Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)
RFC 4182 Removing a Restriction on the use of MPLS Explicit NULL
RFC 5332 MPLS Multicast Encapsulations

MPLS — LDP

RFC 3037 LDP Applicability
RFC 3478 Graceful Restart Mechanism for LDP – GR helper
RFC 5036 LDP Specification
RFC 5283 LDP extension for Inter-Area LSP
RFC 5443 LDP IGP Synchronization
RFC 5561 LDP Capabilities
RFC 6388 LDP Extensions for Point-to-Multipoint and Multipoint-to-Multipoint LSP
RFC 6826 Multipoint LDP in-band signaling for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths
draft-ietf-mpls-ldp-ip-pw-capability-09 Disabling IPoMPLS and P2P PW LDP Application's State Advertisement
draft-ietf-mpls-ldp-ipv6-15 Updates to LDP for IPv6
draft-pdutta-mpls-ldp-adj-capability-00 LDP Adjacency Capabilities
draft-pdutta-mpls-ldp-v2-00 LDP Version 2
draft-pdutta-mpls-multi-ldp-instance-00 Multiple LDP Instances
draft-pdutta-mpls-tldp-hello-reduce-04 Targeted LDP Hello Reduction

MPLS/RSVP — TE

RFC 2702 Requirements for Traffic Engineering over MPLS
RFC2747 RSVP Cryptographic Authentication
RFC 2961 RSVP Refresh Overhead Reduction Extensions
RFC3097 RSVP Cryptographic Authentication - Updated Message Type Value
RFC 3209 Extensions to RSVP for Tunnels

RFC 3473 Generalized Multi-Protocol Label Switching (GMPLS) Signaling

Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions – (support of of IF_ID RSVP_HOP object with unnumbered interface and RSVP-TE Graceful Restart Helper Procedures)

RFC 3477 Signalling Unnumbered Links inResourceReSerVationProtocol-

Traffic Engineering (RSVP-TE)

RFC 3564 Requirements for Diff-Serv-aware TE

RFC 3906 Calculating Interior Gateway Protocol (IGP) Routes Over Traffic Engineering Tunnels

RFC 4090 Fast reroute Extensions to RSVP-TE for LSP Tunnels

RFC 4124 Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering

RFC 4125 Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering

RFC 4127 Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering

RFC 4561 Definition of a RRO Node-Id Sub-Object

RFC 4875 Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)

RFC 4950 ICMP Extensions for Multiprotocol Label Switching

RFC 5151 Inter-domain MPLS and GMPLS Traffic Engineering – RSVP-TE Extensions

RFC 5712 MPLS Traffic Engineering Soft Preemption

RFC 5817 Graceful Shutdown in GMPLS Traffic Engineering Networks

draft-newton-mpls-te-dynamic-overbooking-00 A Diffserv-TE Implementation Model to dynamically change booking factors during failure events

MPLS — OAM

RFC 4379 Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures

RFC 6424 Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels

RFC 6425 Detecting Data Plane Failures in Point-to-Multipoint Multiprotocol Label Switching (MPLS) - Extensions to LSP Ping

MPLS — TP (7750/7450 only)

RFC 5586 MPLS Generic Associated Channel

RFC 5921 A Framework for MPLS in Transport Networks

RFC 5960 MPLS Transport Profile Data Plane Architecture

RFC 6370 MPLS-TP Identifiers

RFC 6378 MPLS-TP Linear Protection

RFC 6428 Proactive Connectivity Verification, Continuity Check and Remote Defect indication for MPLS Transport Profile

RFC 6426 MPLS On-Demand Connectivity and Route Tracing

RFC 6478 Pseudowire Status for Static Pseudowires

RFC 7213 MPLS Transport Profile (MPLS-TP) Next-Hop Ethernet Addressing

MPLS — GMPLS

RFC 3471 Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description

RFC 3473 Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions

RFC 4204 Link Management Protocol (LMP)

RFC 4208 Generalized Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model

RFC4872 RSVP-TE Extensions in Support of End to End GMPLS recovery

draft-ietf-ccamp-rsvp-te-srlg-collect-04 RSVP-TE Extensions for Collecting SRLG Information

RIP

RFC 1058 RIP Version 1

RFC 2080 RIPng for IPv6

RFC 2082 RIP-2 MD5 Authentication

RFC 2453 RIP Version 2

TCP/IP

RFC 768 UDP

RFC 791 IP

RFC 792 ICMP

RFC 793 TCP

RFC 826 ARP

RFC 854 Telnet

RFC 951 Bootstrap Protocol (BOOTP)

RFC 1350 The Tftp Protocol (revision 2)

RFC 1519 CIDR

RFC 1542 Clarifications and Extensions for the Bootstrap Protocol

RFC 1812 Requirements for IPv4 Routers

RFC 2347 TFTP option Extension

RFC 2328 TFTP Blocksize Option

RFC 2349 TFTP Timeout Interval and Transfer

Size option

RFC 2401 Security Architecture for Internet Protocol

RFC 2428 FTP Extensions for IPv6 and NATs

RFC 3596 DNS Extensions to Support IP version 6

RFC 5880 Bidirectional Forwarding Detection

RFC 5881 BFD IPv4 and IPv6 (Single Hop)

RFC 5883 BFD for Multihop Paths

VRRP

RFC 2787 Definitions of Managed Objects for the Virtual Router Redundancy Protocol

RFC 3768 Virtual Router Redundancy Protocol

draft-ietf-vrrp-unified-spec-02 Virtual Router Redundancy Protocol Version 3 for IPv4 and IPv6

PPP

RFC 1332 PPP IPCP
RFC 1377 PPP OSINLCP
RFC 1638/2878PPP BCP
RFC 1661 PPP (rev RFC2151)
RFC 1662 PPP in HDLC-like Framing
RFC 1877 PPP Internet Protocol Control Protocol Extensions for Name Server Addresses
RFC 1989 PPP Link Quality Monitoring
RFC 1990 The PPP Multilink Protocol (MP)
RFC 1994 PPP Challenge Handshake Authentication Protocol (CHAP)
RFC 2516 A Method for Transmitting PPP Over Ethernet
RFC 2615 PPP over SONET/SDH
RFC 2686 The Multi-Class Extension to Multi-Link PPP

Frame Relay

FRF.1.2 - PVC User-to-Network Interface (UNI) Implementation Agreement
FRF.5 - Frame Relay/ATM PVC Network Interworking Implementation
ANSI T1.617 Annex D, DSS1 — Signalling Specification For Frame Relay Bearer Service.
FRF2.2 PVC Network-to- Network Interface (NNI) Implementation Agreement.
FRF.12 Frame Relay Fragmentation Implementation Agreement
FRF.16.1 Multilink Frame Relay UNI/NNI Implementation Agreement
ITU-T Q.933, Annex A Additional procedures for Permanent Virtual Connection (PVC) status management

ATM

RFC 1626 Default IP MTU for use over ATM AAL5
RFC 2514 Definitions of Textual Conventions and OBJECT_IDENTITIES for ATM Management
RFC 2515 Definition of Managed Objects for ATM Management
RFC 2684 Multiprotocol Encapsulation over ATM Adaptation Layer 5
AF-TM-0121.000 Traffic Management Specification Version 4.1
ITU-T Recommendation I.610 B-ISDN Operation and Maintenance Principles and Functions version 11/95
ITU-T Recommendation I.432.1 BISDN user-network interface – Physical layer specification: General characteristics
GR-1248-CORE Generic Requirements for Operations of ATM Network Elements (NEs). Issue 3
GR-1113-CORE Bellcore, Asynchronous Transfer Mode (ATM) and ATM Adaptation Layer (AAL) Protocols Generic Requirements, Issue 1
AF-ILMI-0065.000 Integrated Local Management Interface (ILMI) Version 4.0
AF-TM-0150.00 Addendum to Traffic Management v4.1 optional minimum desired cell rate indication for UBR
AF-PHY-0086.001 Inverse Multiplexing for ATM (IMA) Specification Version 1.1

DHCP

RFC 2131 Dynamic HostConfiguration Protocol (REV)
RFC 3046 DHCP Relay Agent Information Option (Option 82)
RFC 1534 Interoperation between DHCP and BOOTP

Policy Management and Credit Control

3GPP TS 29.212 Policy and Charging Control (PCC) over Gx/Sd Reference Point (Release 11 and Release 12) - Gx support as it applies to wireline environment (BNG)
RFC 3588 Diameter Base Protocol
RFC 4006 Diameter Credit Control Application

NAT

RFC 5382 NAT Behavioral Requirements fr TCP
RFC 5508 NAT Behavioral Requirements for ICMP
RFC 6146 Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers
RFC 6333 Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion
RFC 6334 Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual-Stack Lite
RFC 6888 Common Requirements For Carrier-Grade NATs (CGNs)

VPLS

RFC 4761 Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling
RFC 4762 Virtual Private LAN Services Using LDP
RFC 5501 Requirements for Multicast Support in Virtual Private LAN Services
RFC 6074 Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)
RFC 7041 Extensions to the Virtual Private LAN Service (VPLS) Provider Edge (PE) Model for Provider Backbone Bridging
RFC 7117 Multicast in Virtual Private LAN Service (VPLS)

Pseudowire

RFC 3985 Pseudo Wire Emulation Edge-to-Edge (PWE3)
RFC 4385 Pseudo Wire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN
RFC 3916 Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)
RFC 4717 Encapsulation Methods for Transport ATM over MPLS Networks
RFC 4816 PWE3 ATM Transparent Cell Transport Service
RFC 4448 Encapsulation Methods for Transport of Ethernet over MPLS Networks
RFC 4619 Encapsulation Methods for Transport of Frame Relay over MPLS Networks
RFC 4446 IANA Allocations for PWE3
RFC 4447 Pseudowire Setup and Maintenance Using LDP

RFC 5085 Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires
RFC 5659 An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge
RFC 5885 Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)
RFC 6073 Segmented Pseudowire
RFC 6310 Pseudowire (PW) OAM Message Mapping
RFC 6391 Flow Aware Transport of Pseudowires over an MPLS PSN
RFC 6575 ARP Mediation for IP Interworking of Layer 2 VPN
RFC 6718 Pseudowire Redundancy
RFC 6829 Label Switched Path (LSP) Ping for Pseudowire Forwarding Equivalence Classes (FECs) Advertised over IPv6
RFC 6870 Pseudowire Preferential Forwarding Status bit
RFC 7023 MPLS and Ethernet OAM Interworking
RFC 7267 Dynamic Placement of Multi-Segment Pseudowires
draft-ietf-l2vpn-vpws-iw-oam-04 OAM Procedures for VPWS Interworking
MFA Forum 9.0.0 The Use of Virtual trunks for ATM/MPLS Control Plane Interworking
MFA Forum 12.0.0 Multiservice Interworking - Ethernet over MPLS
MFA Forum 13.0.0 Fault Management for Multiservice Interworking v1.0
MFA Forum 16.0.0 Multiservice Interworking - IP over MPLS

ANCP/L2CP

RFC 5851 ANCP framework
draft-ietf-ancp-protocol-02 ANCP Protocol

Voice /Video Performance:

ITU-T G.107 The E Model- A computational model for use in planning.
ETSI TS 101 329-5 Annex E extensions- QoS Measurement for VoIP - Method for determining an Equipment Impairment Factor using Passive Monitoring
ITU-T Rec. P.564 Conformance testing for voice over IP transmission quality assessment models
ITU-T G.1020, Appendix I Performance Parameter Definitions for Quality of Speech and other Voiceband Applications Utilizing IP Networks- Mean Absolute Packet Delay Variation& Markov Models.
RFC 3550, Appendix A.8 RTP: A Transport Protocol for Real-Time Applications- Estimating the Interarrival Jitter.

Circuit Emulation

RFC 4553 Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)
RFC 5086 Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)
MEF-8 Implementation Agreement for the Emulation of PDH Circuits over Metro Ethernet Networks, October 2004
RFC 5287 Control Protocol Extensions for the Setup of Time-Division Multiplexing (TDM) Pseudowires in MPLS Networks

SONET/SDH

ITU-G.841 Telecommunication Standardization Section of ITU, Types and Characteristics of SDH Networks Protection Architecture, issued in October 1998 and as augmented by Corrigendum1 issued in July 2002

AAA

RFC 2865 Remote Authentication Dial In User Service
RFC 2866 RADIUS Accounting
draft-grant-tacacs-02 The TACACS+ Protocol

SSH

RFC 4250 The Secure Shell (SSH) Protocol Assigned Numbers
RFC 4251 The Secure Shell (SSH) Protocol Architecture
RFC 4254 The Secure Shell (SSH) Connection Protocol

OpenFlow

ONF OpenFlow Switch Specification Version 1.3.1 (Hybrid-switch/ FlowTable)

Timing

GR-253-CORE SONET Transport Systems: Common Generic Criteria. Issue 3, September 2000
ITU-T G.781 Telecommunication Standardization Section of ITU, Synchronization layer functions, issued 09/2008
ITU-T G.813 Telecommunication Standardization Section of ITU, Timing characteristics of SDH equipment slave clocks (SEC), issued 03/2003.
GR-1244-CORE Clocks for the Synchronized Network: Common Generic Criteria, Issue 3, May 2005
ITU-T G.8261 Telecommunication Standardization Section of ITU, Timing and synchronization aspects in packet networks, issued 04/2008.
ITU-T G.8262 Telecommunication Standardization Section of ITU, Timing characteristics of synchronous Ethernet equipment slave clock (EEC), issued 08/2007.
ITU-T G.8264 Telecommunication Standardization Section of ITU, Distribution of timing information through packet networks, issued 10/ 2008.
ITU-T G.8265.1 Telecommunication Standardization Section of ITU, Precision time protocol telecom profile for frequency synchronization, issued 10/2010.
IEEE 1588-2008 IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems

Network Management

ITU-T X.721 Information technology-OSI-Structure of Management Information

ITU-T X.734 Information technology-OSI-Systems Management: Event Report Management Function

M.3100/3120 Equipment and Connection Models

TMF 509/613 Network Connectivity Model

RFC 1157 SNMPv1

RFC 1215 A Convention for Defining Traps for use with the SNMP

RFC 1657 BGP4-MIB

RFC 1724 RIPv2-MIB

RFC 1850 OSPF-MIB

RFC 1907 SNMPv2-MIB

RFC 2011 IP-MIB

RFC 2138 RADIUS

RFC 2206 RSVP-MIB

RFC 2452 IPv6 Management Information Base for the Transmission Control Protocol

RFC 2465 Management Information Base for IPv6: Textual Conventions and General Group

RFC 2558 SONET-MIB

RFC 2571 SNMP-FRAMEWORKMIB

RFC 2572 SNMP-MPD-MIB

RFC 2573 SNMP-TARGET-&-NOTIFICATION-MIB

RFC 2574 SNMP-USER-BASED-SMMIB

RFC 2575 SNMP-VIEW-BASEDACM-MIB

RFC 2576 SNMP-COMMUNITY-MIB

RFC 2578 Structure of Management Information Version 2 (SMIv2)

RFC 2665 EtherLike-MIB

RFC 2819 RMON-MIB

RFC 2863 IF-MIB

RFC 2864 INVERTED-STACK-MIB

RFC 2987 VRRP-MIB

RFC 3014 NOTIFICATION-LOGMIB

RFC 3019 IP Version 6 Management Information Base for The Multicast Listener Discovery Protocol

RFC 3164 Syslog

RFC 3273 HCRMON-MIB

RFC 3411 An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks

RFC 3412 Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)

RFC 3413 Simple Network Management Protocol (SNMP) Applications

RFC 3414 User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)

RFC 3418 SNMP MIB

RFC 3826 The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model

RFC 4113 Management Information Base for the User Datagram Protocol (UDP)

RFC 4292 IP Forwarding Table MIB

RFC 4293 MIB for the Internet Protocol

RFC 5101 Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information

RFC 6241 Network Configuration Protocol (NETCONF)

RFC 6242 Using the NETCONF Protocol over Secure Shell (SSH)

draft-ietf-bfd-mib-00 Bidirectional Forwarding Detection Management Information Base

draft-ietf-isis-wg-mib-06 Management Information Base for Intermediate System to Intermediate System (IS-IS)

draft-ietf-ospf-mib-update-04 OSPF Version 2 Management Information Base

draft-ietf-mboned-msdp-mib-01 Multicast Source Discovery protocol MIB

draft-ietf-mpls-lsr-mib-06 Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base

draft-ietf-mpls-te-mib-04 Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base

draft-ietf-mpls-ldp-mib-07 Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)

IEEE 802.3ad MIB

Customer documentation and product support

Customer documentation

http://documentation.alcatel-lucent.com

Technical support

http://support.alcatel-lucent.com

Documentation feedback

documentation.feedback@alcatel-lucent.com

3HE 09858 AAAA TQZZA 01