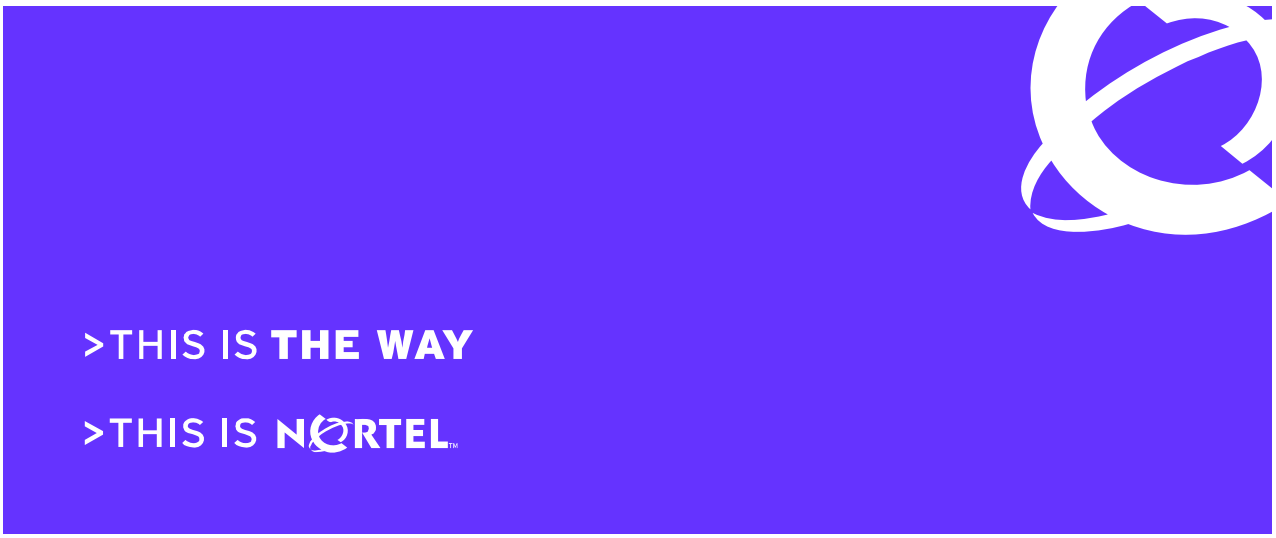


Nortel Multiservice Switch 7400/15000/20000

Operations: Ethernet Line Virtual Services

NN10600-822

Document status: Standard
Document issue: 7.2S1
Document date: March 2006
Product release: PCR7.2 and up
Job function: Operations
Type: NTP
Language type: U.S. English

Contents

What's new 7
EVLS policing 7

Ethernet virtual line services configuration 8

EVLS deployment and migration from the Ethernet over ATM service 11
Removing the EoAtm feature 13
Configuring EVLS software 14

EVLS configuration 15
Configuring EVLS on an Ethernet interface 19
Configuring a LAN-to-LAN EVC 21
Configuring a LAN-to-VLAN EVC 23
Configuring a VLAN-to-LAN EVC 26
Configuring a 1:1 VLAN-to-VLAN EVC 29
Configuring an N:1 VLAN-to-VLAN EVC 31
Configuring a LAN-to-ATM EVC for LAN-to-LAN transport 33
Configuring a LAN-to-ATM EVC for LAN-to-VLAN transport 35
Configuring a 1:1 VLAN-to-ATM EVC for LAN-to-VLAN transport 38
Configuring a 1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport 41
Configuring an N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport 43

EVLS traffic management configuration 45
Configuring EVLS core traffic management 47
Configuring the EVLS policer for an EVC at the calling or called end 50
Enabling the EVLS policer for an EVC 52

Ethernet transport system (ETS) configuration 54
Configuring ETS software 56
Configuring ETS on the node 57
Configuring ETS for end-to-end EVLS ATM transport 58
Configuring ETS for single-ended EVLS ATM transport 62

EVLS or ETS removal 66
Removing EVLS 67
Removing ETS 68

Nortel Multiservice Switch 7400/15000/20000
Operations: Ethernet Line Virtual Services
NN10600-822 7.2S1 Standard
PCR7.2 and up March 2006

Troubleshooting EVLS **70**
Verifying link side connectivity 74
Verifying LAN-to-LAN EVC connectivity 75
Verifying LAN-to-VLAN EVC connectivity 77
Verifying VLAN-to-LAN EVC connectivity 79
Verifying 1:1 VLAN-to-VLAN EVC connectivity 81
Verifying N:1 VLAN-to-VLAN EVC connectivity 83
Verifying LAN-to-ATM EVC connectivity 85
Verifying 1:1 VLAN-to-ATM EVC connectivity 87
Verifying N:1 VLAN-to-ATM EVC connectivity 89
Verifying ETS access connectivity 91
Verifying ETS end-to-end EVLS ATM transport connectivity 93
Verifying ETS single-ended EVLS ATM transport connectivity 95
Verifying traffic flow over a LAN-to-LAN EVC 97
Verifying traffic flow over a LAN-to-VLAN EVC 99
Verifying traffic flow over a VLAN-to-LAN EVC 101
Verifying traffic flow over a VLAN-to-VLAN EVC 103
Verifying traffic flow over a LAN-to-ATM EVC 105
Verifying traffic flow over a VLAN-to-ATM EVC 107
Verifying policer configuration 109
Verifying policer operation 111
Isolating EVLS problems 113
Diagnosing EVC connectivity faults 114
Diagnosing traffic loss related to congestion 116
Isolating the point of congestion 118
Diagnosing traffic loss unrelated to congestion 120
Handling problems 123
Diagnosing no traffic enforcement 128
Diagnosing that all traffic is tagged by policer 131
Diagnosing that all traffic is discarded by policer 133
Diagnosing that the connection is not meeting service level agreement (SLA) 135
Diagnosing policer admitting excess traffic as CIR conformant 137
Diagnosing policer admitting non-conformant traffic as EIR conformant 139

EVLS overview **141**
Why use EVLS? 141
Terminology 142
How does EVLS work? 143
Supported cards 144
Physical network 146
Carrier grade functionality 147
Timers 148
Addressing 148

Frame data flows **150**
EVLS frame formats 151
Ethernet policy rules 152
LAN-to-ATM EVC for LAN-to-LAN transport (Ethernet-to-ATM direction) 153
LAN-to-ATM EVC for LAN-to-LAN transport (ATM-to-Ethernet direction) 154
LAN-to-ATM EVC for LAN-to-VLAN transport (Ethernet-to-ATM direction) 155
LAN-to-ATM EVC for LAN-to-VLAN transport (ATM-to-Ethernet direction) 158
VLAN-to-ATM EVC for LAN-to-VLAN transport (Ethernet-to-ATM direction) 158
VLAN-to-ATM EVC for LAN-to-VLAN transport (ATM-to-Ethernet direction) 162
VLAN 162
Port-VLAN 162
1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (Ethernet-to-ATM direction) 165
1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (ATM-to-Ethernet direction) 166
N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (Ethernet-to-ATM direction) 167
N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (ATM-to-Ethernet direction) 170
Packet modification and RFC2684 encapsulation 172
LLC encapsulation for bridged protocols 173

EVLS solution for end-to-end EVC **174**
What is the EVLS solution for end-to-end EVC? 174
Benefits of the EVLS solution for end-to-end EVC 174
Configurations of the EVLS solution for end-to-end EVC 175
LAN-to-LAN connectivity 175
LAN-to-VLAN connectivity 176
1:1 VLAN-to-VLAN connectivity 177
N:1 VLAN-to-VLAN connectivity 178

EVLS solution for single-ended EVC **181**
What is the EVLS solution for single-ended EVC? 181
Performance enhancement solution 181
Benefits of the EVLS solution for single-ended EVC 182
Configurations of the EVLS solution for single-ended EVC 183
LAN-to-ATM EVC 183
1:1 VLAN-to-ATM EVC 184
N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport 185

EVLS traffic management **188**
Traffic management overview 188
Traffic management features 188
Ethernet traffic classes 189
Network engineering 189
Ethernet to ATM bandwidth parameters mapping 190
Traffic parameter mapping options 190

Overhead factor 192
Overhead calculation 193
Example of conversion from Ethernet to ATM traffic parameters 195
EVLS policing 196
Overview 197
Benefits 198
Policer algorithm 199
Policer configuration 200
Policing granularity 206
Policer parameters and operational representation 207
Policing statistics 208
Traffic management functions 208
Ingress node traffic management 209
Egress node congestion management 212

Procedure conventions 217
Operational mode 217
Provisioning mode 218
Activating configuration changes 218

What's new

The following feature was added to this document:

- EVLS policing (page 7)

Attention: To ensure that you are using the most current version of an NTP, check the current NTP list in NN10600-000 *Nortel Multiservice Switch 7400/15000/20000 What's New*.

EVLS policing

The following section was added for this feature:

- EVLS traffic management configuration (page 45)

The following section was updated for this feature:

- Ethernet virtual line services configuration (page 8)

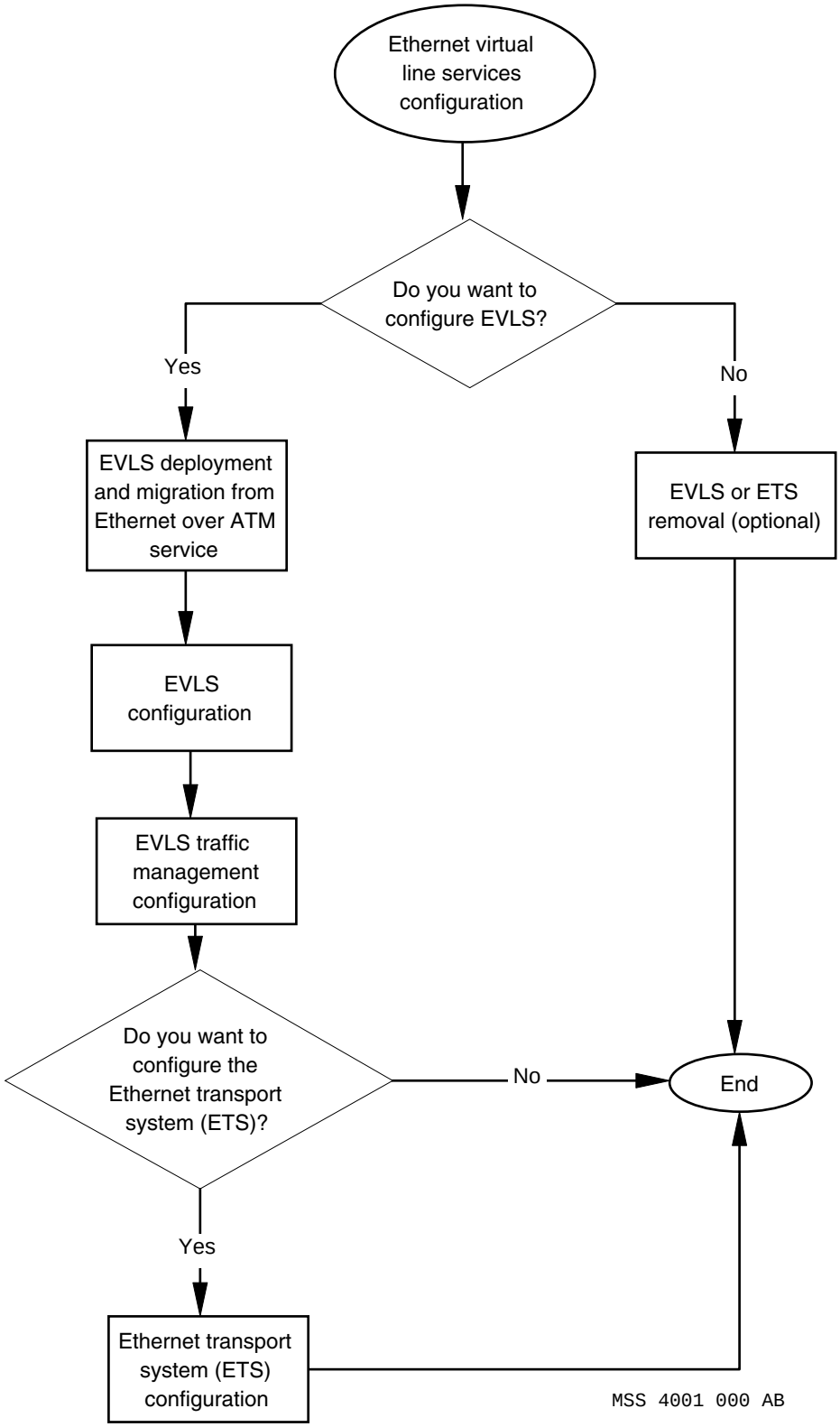
Ethernet virtual line services configuration

Configure the Ethernet interface to carry Ethernet traffic to and from the Ethernet physical media for the Ethernet virtual line service (EVLS).

Ethernet virtual line services configuration tasks

This work flow shows you the sequence of tasks you perform to configure the Ethernet interface. To link to any procedure, go to Ethernet virtual line services configuration task navigation (page 10).

Ethernet virtual line services configuration tasks



Ethernet virtual line services configuration task navigation

- EVLS deployment and migration from the Ethernet over ATM service (page 11)
- EVLS configuration (page 15)
- EVLS traffic management configuration (page 45)
- Ethernet transport system (ETS) configuration (page 54)
- EVLS or ETS removal (page 66)

EVLS deployment and migration from the Ethernet over ATM service

Ethernet virtual line service (EVLS) is a replacement for the previously available Ethernet over ATM (EoAtm) feature introduced in releases 5.1 and 5.2. If the EoAtm feature has been deployed, an upgrade from EoAtm to EVLS is mandatory. In order to deploy the EVLS feature, the software must be upgraded and the EVLS feature must be turned on.

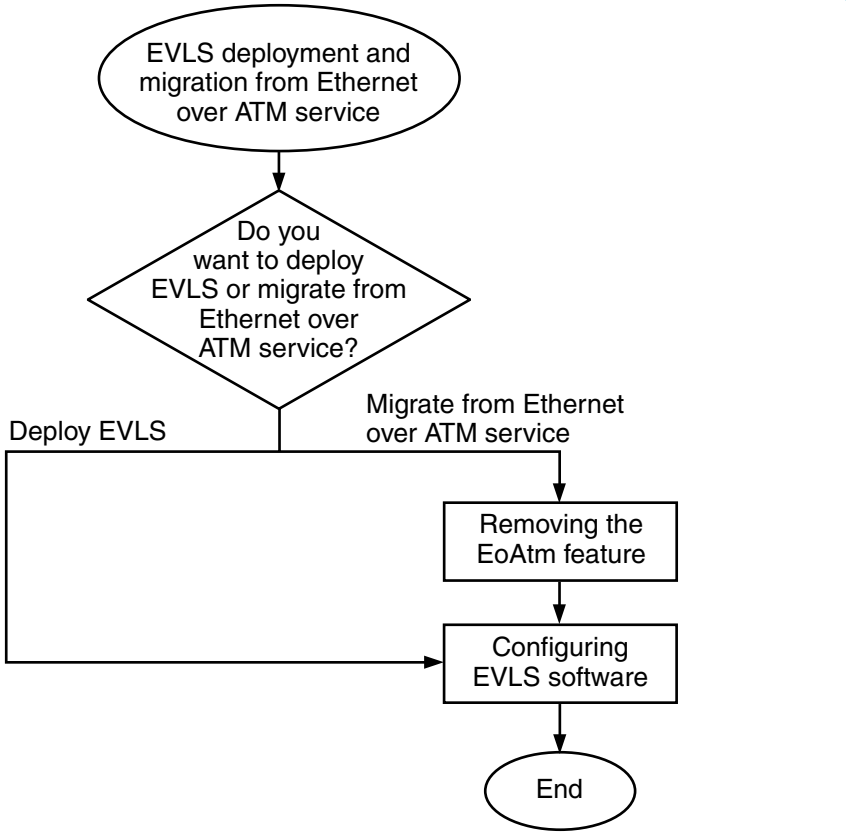
Prerequisites to EVLS deployment and migration from the Ethernet over ATM service

- The software needs to be updated on the shelf. See NN10600-272 *Nortel Multiservice Switch 7400/15000/20000 Upgrading Software*.

EVLS deployment and migration from the Ethernet over ATM service procedures

This task flow shows you the sequence of procedures you perform to deploy EVLS and migrate from the Ethernet over ATM service. To link to any procedure, go to EVLS deployment and migration from the Ethernet over ATM service procedure navigation (page 12).

EVLS deployment and migration from the Ethernet over ATM service procedures



EVLS deployment and migration from the Ethernet over ATM service procedure navigation

- Removing the EoAtm feature (page 13)
- Configuring EVLS software (page 14)

Removing the EoAtm feature

If you have the Ethernet over ATM (EoAtm) feature configured, you need to remove this feature before adding the EVLS feature.

Procedure steps

Step	Action
1	Remove the Ethernet over ATM feature. **`set Sw Lpt/<lpt> featureList ~eoatm`**
	--End--

Variable definitions

Variable	Value
<lpt>	is the instance value of the *LogicalProcessorType* component.

Procedure job aid

EoAtm feature component hierarchy

Configuring EVLS software

Configure the Ethernet FP to load Ethernet virtual line service (EVLS) software on a single Multiservice Switch node.

Procedure steps

Step	Action
1	Add a software logical processor type to a new Ethernet FP. If the Ethernet FP already exists, proceed to step 2. **`add Sw Lpt/<lpt>`**
2	Associate the logical processor type with the Ethernet FP. **`set Lp/<lp> logicalProcessorType Sw Lpt/<lpt>`**
3	Set the feature list to a new logical processor type. **`set Sw Lpt/<lpt> featureList Evls`**
4	Repeat step 1 through step 3 on a second Multiservice Switch node.
	--End--

Variable definitions

Variable	Value
<lp>	is the instance value of the *LogicalProcessor* component.
<lpt>	is the instance value of the *LogicalProcessorType* component.

Procedure job aid

EVLS software component hierarchy

EVLS configuration

Configure the Ethernet virtual line service (EVLS) to offer one or more users the ability to connect to another Ethernet interface, via single-ended or end-to-end connections, across an ATM network.

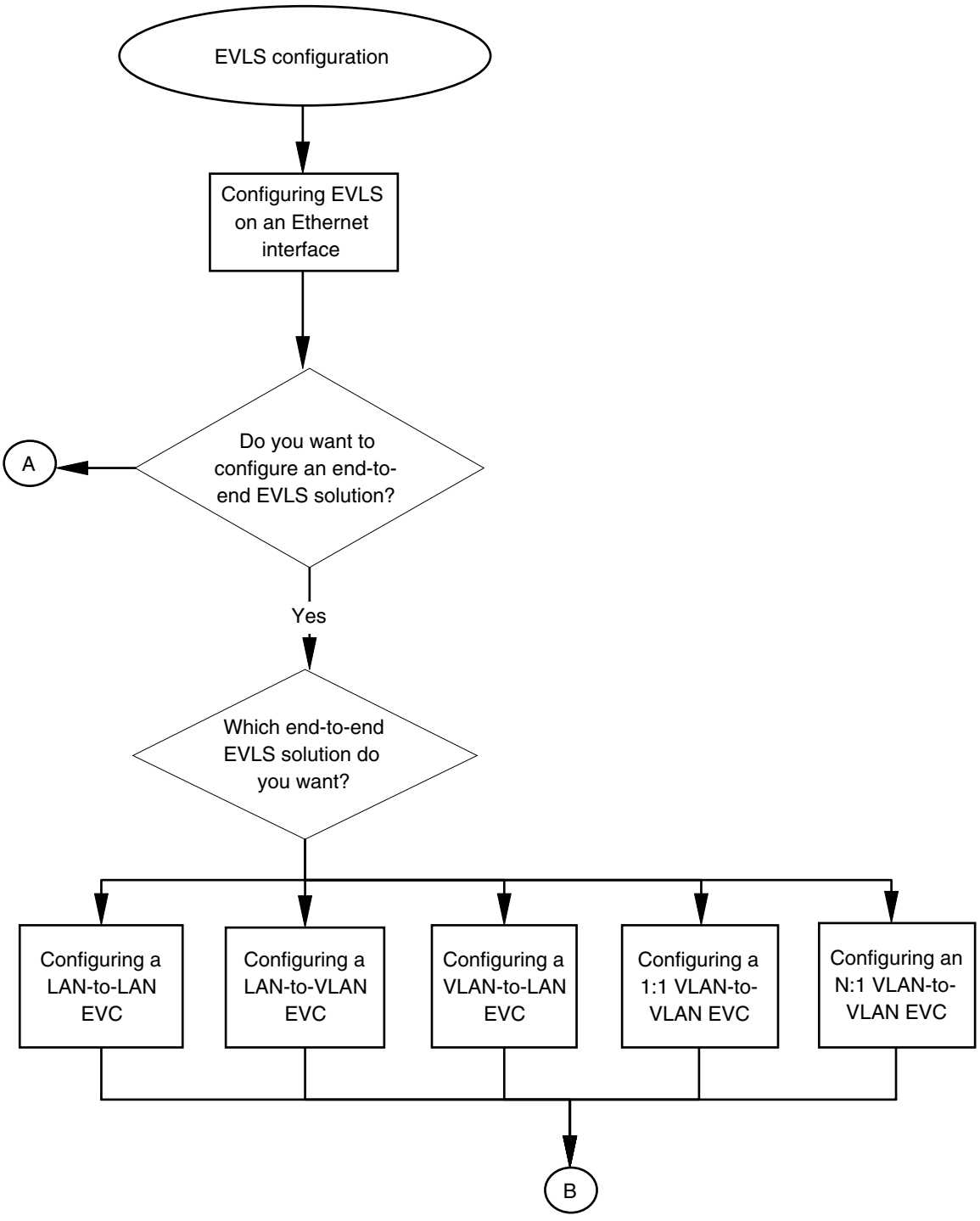
Prerequisites to EVLS configuration

- The Ethernet function processor must be configured, as well as the physical Ethernet ports. See NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference*.
- Logical processor must be added and linked to the Ethernet FP. See NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures*.
- Ensure the ATM network, including PNNI and UNI, is installed on the node. See NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

EVLS configuration procedures

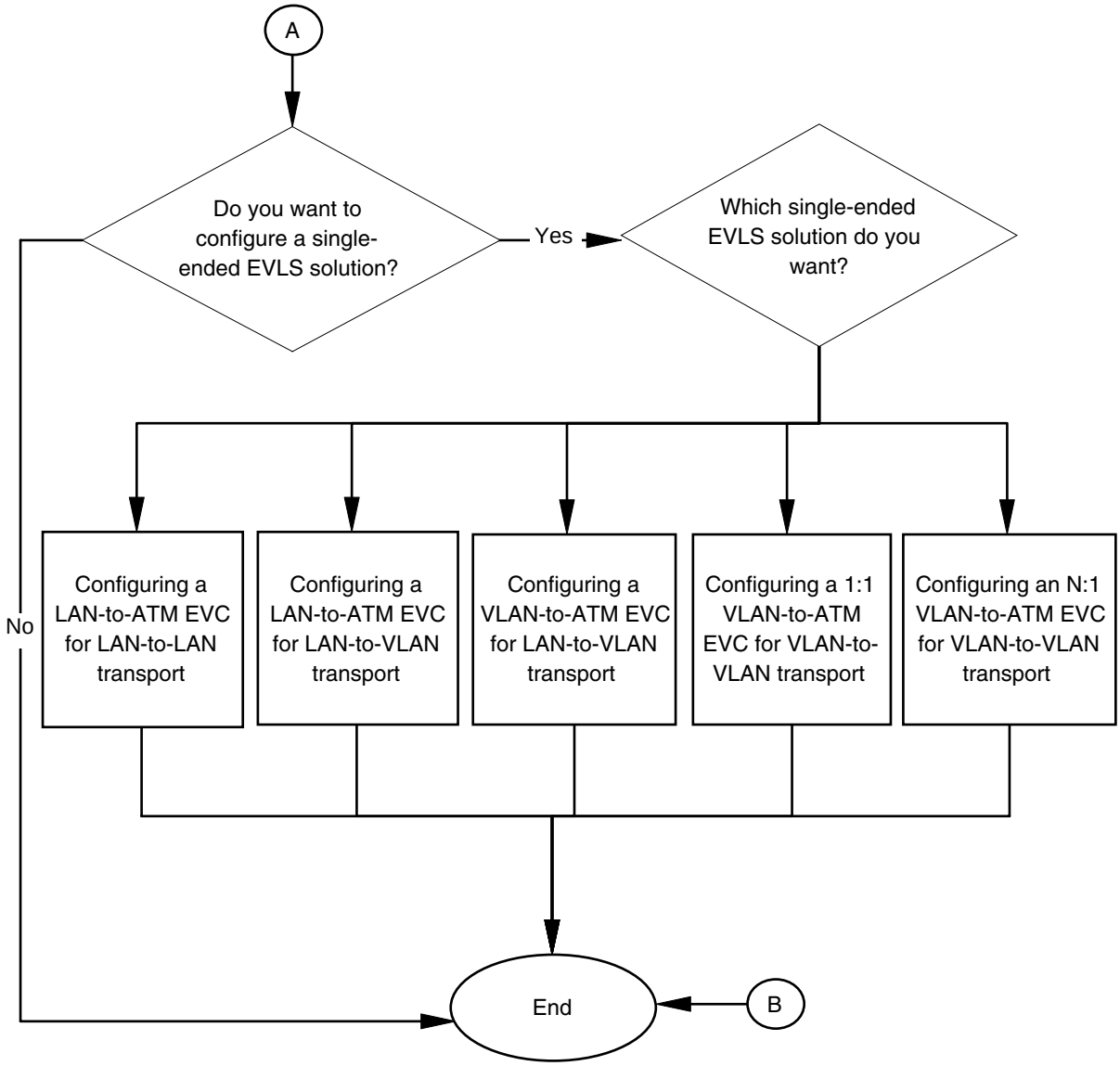
This task flow shows you the sequence of procedures you perform to configure the EVLS. To link to any procedure, go to EVLS configuration procedure navigation (page 17).

EVLS configuration procedures, part 1 of 2



MSS 4001 038 AA

EVLS configuration procedures, part 2 of 2



MSS 4001 039 AA

EVLS configuration procedure navigation

- Configuring EVLS on an Ethernet interface (page 19)
- Configuring a LAN-to-LAN EVC (page 21)
- Configuring a LAN-to-VLAN EVC (page 23)
- Configuring a VLAN-to-LAN EVC (page 26)

- Configuring a 1:1 VLAN-to-VLAN EVC (page 29)
- Configuring an N:1 VLAN-to-VLAN EVC (page 31)
- Configuring a LAN-to-ATM EVC for LAN-to-LAN transport (page 33)
- Configuring a LAN-to-ATM EVC for LAN-to-VLAN transport (page 35)
- Configuring a 1:1 VLAN-to-ATM EVC for LAN-to-VLAN transport (page 38)
- Configuring a 1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (page 41)
- Configuring an N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (page 43)

Configuring EVLS on an Ethernet interface

Configure the Ethernet virtual line service (EVLS) on an Ethernet interface.

Procedure steps

Step	Action
1	Add the Ethernet virtual line service (EVLS) to the Ethernet interface. **`add La/<la> Evls`** The *LocalAddr* component is automatically added. The default *nsapAddress* is provided.
2	Optionally, set the *softPvcHoldOffTime* attribute. **`set La/<la> Evls softPvcHoldOffTime <softPvcHold>`**
3	Optionally, set the *softPvcRetryPeriod* attribute. **`set La/<la> Evls softPvcRetryPeriod <softPvcRetry>`**
4	Optionally, set the *EthernetLocalAddress* component. **`set La/<la> Evls LocalAddr nsapAddress <nsapAddress>`**
5	Optionally, add the Ethernet policy. **`add La/<la> Policy`**
6	Optionally, set the unknown VLAN ID treatment. **`set La/<la> Policy unknownVlanIdTreatment <unknownVlanIdTreatment>`**
7	If the value of the *unknownVlanIdTreatment* attribute is accept, set the *maxFrameSize* attribute. **`set Lp/<lp> Eth/<v> maxFrameSize <frame_value>`**

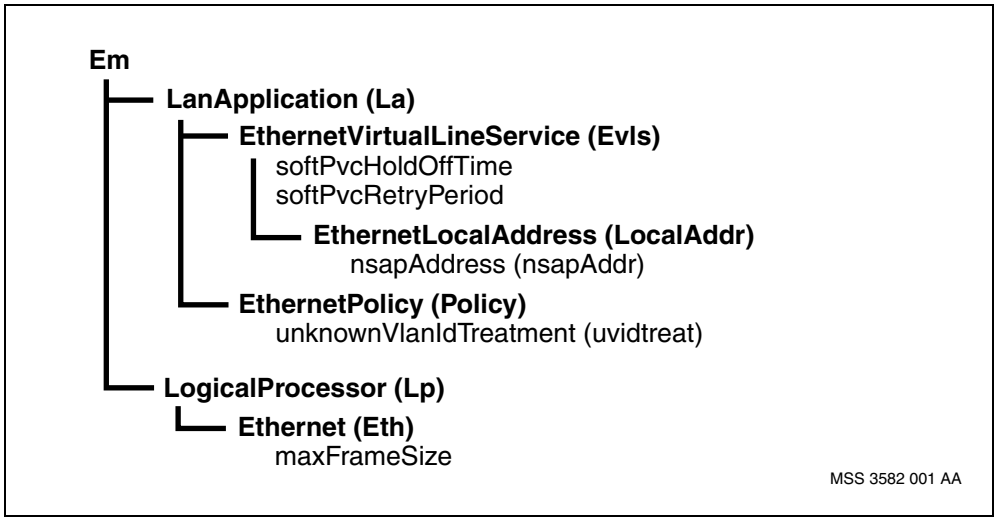
--End--

Variable definitions

Variable	Value
<frame_value>	specifies the maximum frame size of the port.
<la>	is the instance value of the *LanApplication* component.
<lp>	is the instance value of the *LogicalProcessor* component.
<nsapAddress>	is the address of the Ethernet interface. This address is displayed under Artg Dna/* Dest/*.
<softPvcHold>	specifies the time in milliseconds to hold off retries following the release of a connection.
<softPvcRetry>	specifies the time interval in seconds at which a SPVC connection attempts re-establishment after the calling end of the SPVC initiates the setup request and receives a release message.
<unknownVlanId Treatment>	specifies the treatment of IEEE 802.1Q VLAN-tagged traffic received from the interface that has no VLAN instance configured. The possible values are accept and discard.
<v>	is the number of the Ethernet port on the function processor.

Procedure job aid

EVLS on an Ethernet interface component hierarchy



Configuring a LAN-to-LAN EVC

Configure a LAN-to-LAN Ethernet virtual connection (EVC) between the calling end and the called end.

Procedure steps

Step	Action
1	For the called end, complete step 2. For the calling end, complete step 2 to step 7.
2	Add an *EthernetVirtualConnection (Evc)* to the LAN application. **`add La/<la> Evc`**
3	Add an SPVC. **`add La/<la> Evc Spvc/<Spvc>`**
4	Add the source EVC. **`add La/<la> Evc SrcEvc`**
5	Set the remote address. **`set La/<la> Evc SrcEvc remoteAddress <remoteAddress>`** The remote address should represent the NSAP address of the far-end Ethernet interface within the ATM network.
6	Set the remote VLAN ID to zero. **`set La/<la> Evc SrcEvc remoteVlanId 0`**
7	Set the *preserveVlanTag* attribute. **`set La/<la> Evc presVlanTag <presVlanTag>`**

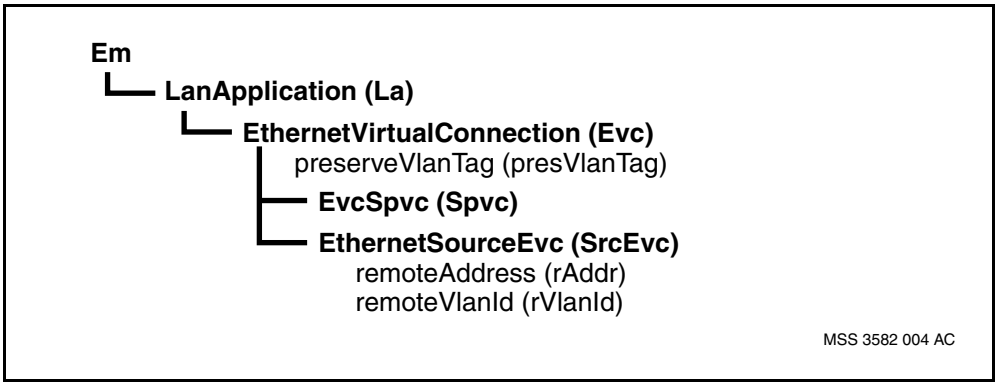
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<presVlanTag>	is the *preserveVlanTag* attribute. This attribute specifies if the EVC, in the direction from the Ethernet interface, preserves or removes the VLAN tag.
<remoteAddress>	is the NSAP address of the far-end Ethernet interface.
<Spvc>	is the instance value of the *Spvc* component.

Procedure job aid

LAN-to-LAN EVC component hierarchy



Configuring a LAN-to-VLAN EVC

Configure a LAN-to-VLAN Ethernet virtual connection (EVC) between the calling end and the called end.

Prerequisites

- Ensure the *Vlan* component is added at the called end.

Procedure steps

Step	Action
1	To configure a LAN as a calling end, complete step 2 to step 6. To configure a VLAN as a called end, complete step 7.
2	Add an *EthernetVirtualConnection (Evc)* to the LAN application. **`add La/<la> Evc`**
3	Add an SPVC. **`add La/<la> Evc Spvc/<Spvc>`**
4	Add the source EVC. **`add La/<la> Evc SrcEvc`**
5	Set the remote address. **`set La/<la> Evc SrcEvc remoteAddress <remoteAddress>`** The remote address should represent the NSAP address of the far-end Ethernet interface within the ATM network.
6	Set the remote VLAN ID. **`set La/<la> Evc SrcEvc remoteVlanId <remoteVlanId>`**
7	Add the EVC to the VLAN at the called end. **`add La/<la> Vlan/<vlan> Evc`**
	--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<remoteAddress>	is the NSAP address of the far-end Ethernet interface.

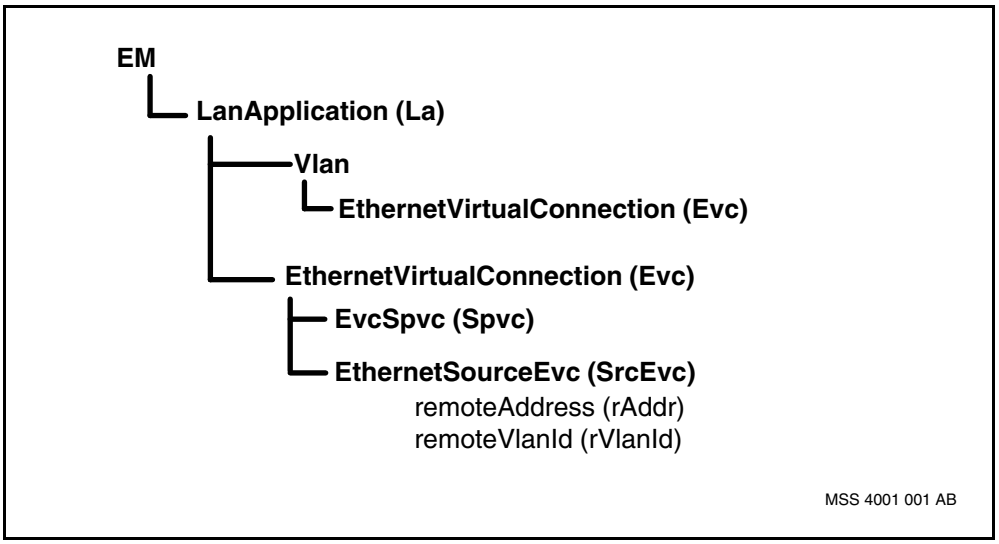
(1 of 2)

Variable	Value
<remoteVlanId>	is the instance value of the *remoteVlanId* attribute. The *remoteVlanId* attribute specifies the identifier of the VLAN at the far-end Ethernet interface that terminates the EVC. The value must be greater than zero.
<Spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

(2 of 2)

Procedure job aid

LAN-to-VLAN EVC component hierarchy



Configuring a VLAN-to-LAN EVC

Configure a VLAN-to-LAN Ethernet virtual connection (EVC) between the calling end and the called end.

Prerequisites

- Ensure the *Vlan* component is added to the calling end.

Procedure steps

Step	Action
1	To configure a VLAN as a calling end, complete step 2 to step 7. To configure a LAN as a called end, complete step 8.
2	Add the EVC to the VLAN. **`add La/<la> Vlan/<vlan> Evc`**
3	Add the EVC SPVC. **`add La/<la> Vlan/<vlan> Evc Spvc/<Spvc>`**
4	Add the source EVC. **`add La/<la> Vlan/<vlan> Evc SrcEvc`**
5	Set the remote address. **`set La/<la> Vlan/<vlan> Evc SrcEvc remoteAddress <remoteAddress>`** The remote address should represent the NSAP address of the far-end Ethernet interface within the ATM network.
6	Set the remote VLAN ID to zero. **`set La/<la> Vlan/<vlan> Evc SrcEvc remoteVlanId 0`**
7	Set the *preserveVlanTag* attribute. **`set La/<la> Vlan/<vlan> Evc presVlanTag <presVlanTag>`**
8	Add an *EthernetVirtualConnection (Evc)* to the LAN application at the called end. **`add La/<la> Evc`**

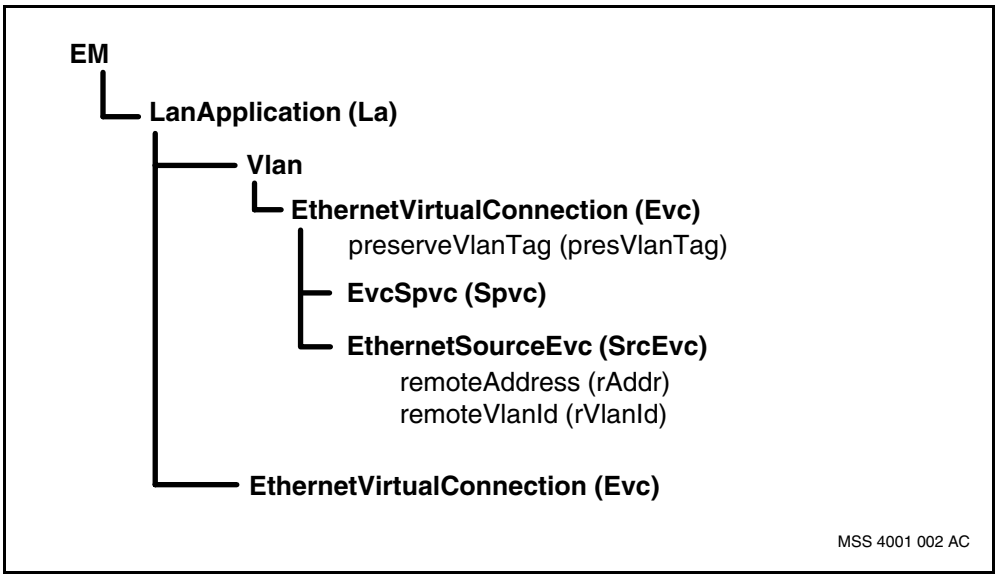
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<presVlanTag>	is the *preserveVlanTag* attribute. This attribute specifies if the EVC, in the direction from the Ethernet interface, preserves or removes the VLAN tag.
<remoteAddress>	is the NSAP address of the far-end Ethernet interface.
<Spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

VLAN-to-LAN EVC component hierarchy



Configuring a 1:1 VLAN-to-VLAN EVC

Configure a 1:1 VLAN-to-VLAN Ethernet virtual connection (EVC) to connect an EVC soft permanent virtual circuit (SPVC) to another VLAN across an ATM private network-to-network interface (PNNI) network.

Prerequisites

- Ensure the *Vlan* component is added to both the calling and called ends.

Procedure steps

Step	Action
1	For the called end, complete step 2. For the calling end, complete step 2 to step 6.
2	Add the EVC to the VLAN. **`add La/<la> Vlan/<vlan> Evc`**
3	Add the EVC SPVC. **`add La/<la> Vlan/<vlan> Evc Spvc/<Spvc>`**
4	Add source EVC. **`add La/<la> Vlan/<vlan> Evc SrcEvc`**
5	Set the remote address of the far-end Ethernet interface. **`set La/<la> Vlan/<vlan> Evc SrcEvc remoteAddress <remoteAddress>`**
6	Set the remote VLAN ID. **`set La/<la> Vlan/<vlan> Evc SrcEvc remoteVlanId <remoteVlanId>`**

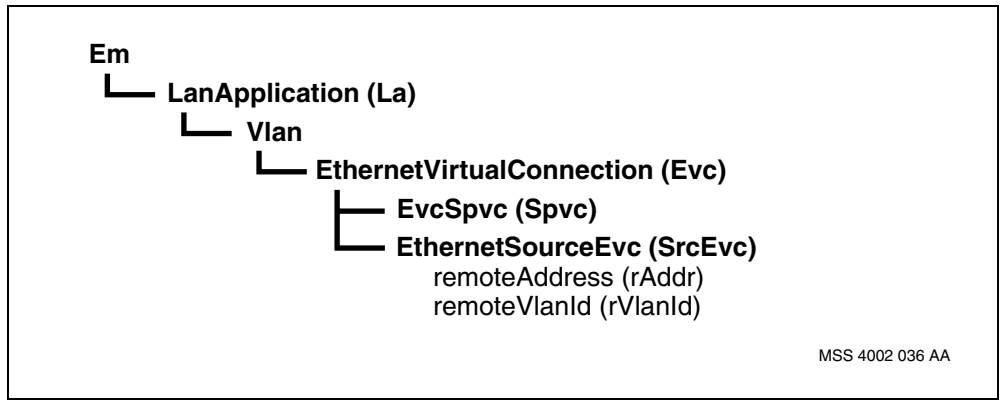
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<remoteAddress>	is the NSAP address of the far-end Ethernet interface.
<remoteVlanId>	is the instance value of the *remoteVlanId* attribute. The *remoteVlanId* attribute specifies the identifier of the VLAN at the far-end Ethernet interface that terminates the EVC. The value must be greater than zero.
<Spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

1:1 VLAN-to-VLAN EVC component hierarchy



Configuring an N:1 VLAN-to-VLAN EVC

Configure an N:1 VLAN-to-VLAN Ethernet virtual connection (EVC) and ensure that the *LanApplication* at both ends of the network is configured as a calling end. The EVC soft permanent virtual circuit (SPVC) is established using the Ethernet transport system (ETS). The ETS must be configured to establish the virtual channel connection (VCC) on the ATM trunk.

Prerequisites

- Configure the Ethernet transport system (ETS). See Ethernet transport system (ETS) configuration (page 54).
- Steps for configuring a 1:1 VLAN-to-VLAN EVC must be performed and both ends must be configured as calling ends. See Configuring a 1:1 VLAN-to-VLAN EVC (page 29).

Procedure steps

Step	Action
1	Set the VLAN to terminate on the Ethernet transport system (ETS). **`set La/<la> Vlan/<vlan> Evc aggregation enabled`**
2	Repeat step 1 for each additional VLAN.
3	Repeat step 1 and step 2 to configure the other calling end on a second Multiservice Switch node.

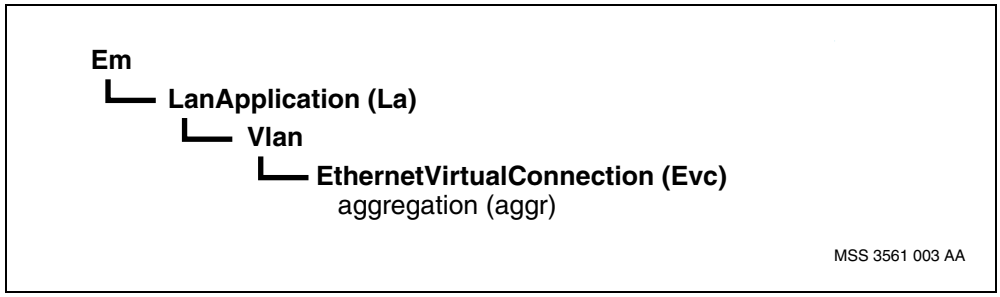
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component aggregated on the VCC.

Procedure job aid

N:1 VLAN-to-VLAN EVC component hierarchy



Configuring a LAN-to-ATM EVC for LAN-to-LAN transport

Configure a LAN-to-ATM Ethernet virtual connection (EVC) for LAN-to-LAN transport to terminate on an ATM virtual channel connection (VCC). The VCC can use one of the following protocols: ATM inter-network interface (AINI), interim inter-switch signalling protocol (IISP), private network-to-network interface (PNNI), or user-to-network interface (UNI). The EVC is configured as the calling end.

Prerequisites

- The ATM UNI must be configured. See NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

Procedure steps

Step	Action
1	Add an Ethernet virtual connection (EVC) to the LAN application. **`add La/<la> Evc`**
2	Add an SPVC. **`add La/<la> Evc Spvc/<Spvc>`**
3	Add the source EVC. **`add La/<la> Evc SrcEvc`**
4	Set the remote address. **`set La/<la> Evc SrcEvc remoteAddress <remoteAddress>`** The remote address should represent the NSAP address of the called-end ATM interface within the ATM network.
5	Set the remote VLAN ID to zero. **`set La/<la> Evc SrcEvc remoteVlanId 0`**
6	Set the *preserveVlanTag* attribute. **`set La/<la> Evc presVlanTag <presVlanTag>`**
7	Add the source VCC to the SPVC. **`add La/<la> Evc Spvc/<Spvc> SrcVcc`**
8	Set the Vpi.Vci for the far-end ATM UNI. **`set La/<la> Evc Spvc/<Spvc> SrcVcc calledVpiVci <calledVpiVci>`**
9	Repeat step 1 to step 8 to configure the other calling end on a second Multiservice Switch node.

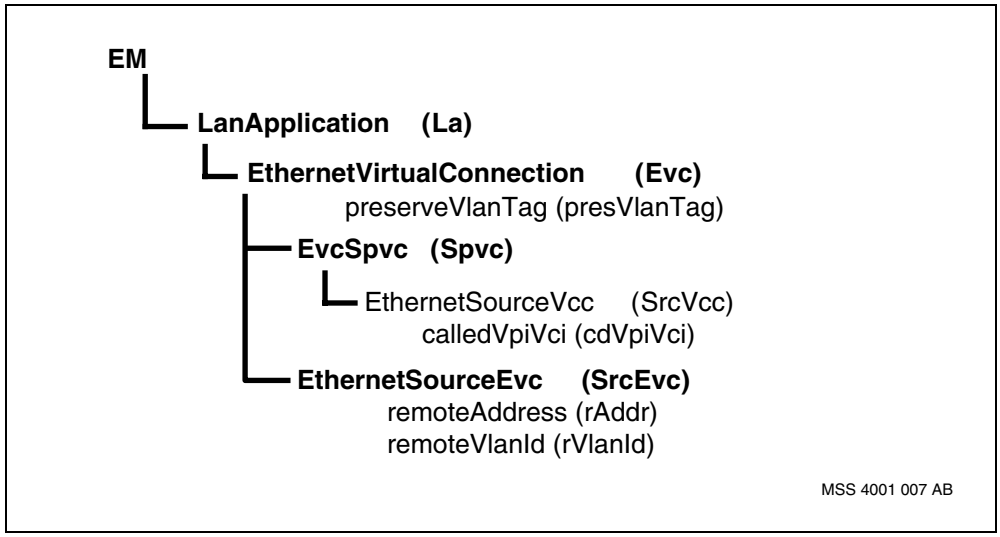
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<presVlanTag>	is the *preserveVlanTag* attribute. This attribute specifies if the EVC, in the direction from the Ethernet interface, preserves or removes the VLAN tag.
<remoteAddress>	is the NSAP address of the called-end ATM interface.
<Spvc>	is the instance value of the *Spvc* component.

Procedure job aid

LAN-to-ATM EVC for LAN-to-LAN transport component hierarchy



Configuring a LAN-to-ATM EVC for LAN-to-VLAN transport

Configure a LAN-to-ATM EVC for LAN-to-VLAN transport to terminate on an ATM virtual channel connection (VCC). The VCC can use one of the following protocols: ATM inter-network interface (AINI), interim inter-switch signalling protocol (IISP), private network-to-network interface (PNNI), or user-to-network interface (UNI). The EVC is configured as the calling end.

Procedure steps

Step	Action
1	Add an Ethernet virtual connection (EVC) to the LAN application. **`add La/<la> Evc`**
2	Add an SPVC. **`add La/<la> Evc Spvc/<Spvc>`**
3	Add the source EVC. **`add La/<la> Evc SrcEvc`**
4	Set the remote address. **`set La/<la> Evc SrcEvc remoteAddress <remoteAddress>`** The remote address should represent the NSAP address of the called-end ATM interface within the ATM network.
5	Set the remote VLAN ID to a value greater than zero. **`set La/<la> Evc SrcEvc remoteVlanId <remoteVlanId>`**
6	Add the source VCC to the SPVC. **`add La/<la> Evc Spvc/<Spvc> SrcVcc`**
7	Set the Vpi.Vci for the called-end ATM interface. **`set La/<la> Evc Spvc/<Spvc> SrcVcc calledVpiVci <calledVpiVci>`**

--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<remoteAddress>	is the NSAP address of the called-end ATM interface.

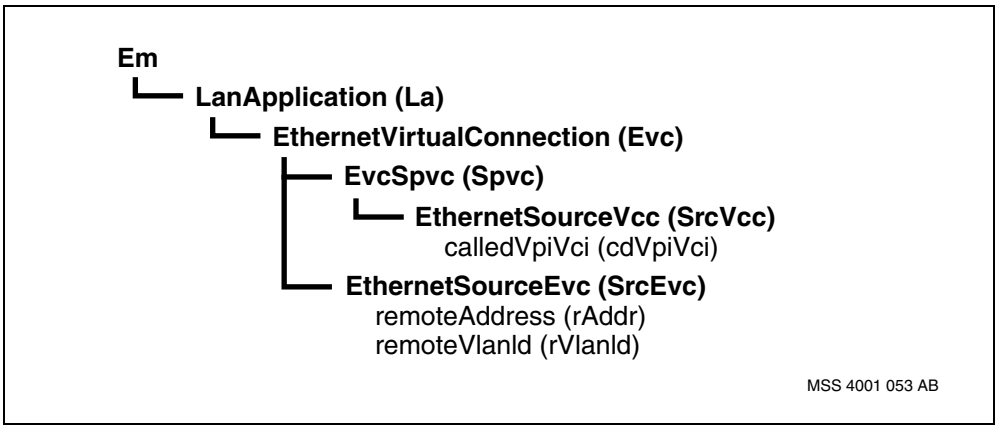
(1 of 2)

Variable	Value
<remoteVlanId>	is the instance value of the *remoteVlanId* attribute. The *remoteVlanId* attribute specifies the identifier of the VLAN at the far-end Ethernet interface. The value must be greater than zero.
<Spvc>	is the instance value of the *Spvc* component.

(2 of 2)

Procedure job aid

LAN-to-ATM EVC for LAN-to-VLAN transport component hierarchy



Configuring a 1:1 VLAN-to-ATM EVC for LAN-to-VLAN transport

Configure a 1:1 VLAN-to-ATM EVC for LAN-to-VLAN transport to terminate on an ATM virtual channel connection (VCC).

Procedure steps

Step	Action
1	Add the EVC to the VLAN. **`add La/<la> Vlan/<vlan> Evc`**
2	Add the EVC SPVC. **`add La/<la> Vlan/<vlan> Evc Spvc/<Spvc>`**
3	Add the source EVC. **`add La/<la> Vlan/<vlan> Evc SrcEvc`**
4	Set the remote address. **`set La/<la> Vlan/<vlan> Evc SrcEvc remoteAddress <remoteAddress>`** The remote address should represent the NSAP address of the called-end ATM interface within the ATM network.
5	Set the remote VLAN ID to zero. **`set La/<la> Vlan/<vlan> Evc SrcEvc remoteVlanId 0`**
6	Set the *preserveVlanTag* attribute. **`set La/<la> Vlan/<vlan> Evc presVlanTag <presVlanTag>`**
7	Add the source VCC to the SPVC. **`add La/<la> Vlan/<vlan> Evc Spvc/<Spvc> SrcVcc`**
8	Set the Vpi.Vci for the far-end ATM UNI. **`set La/<la> Vlan/<vlan> Evc Spvc/<Spvc> SrcVcc calledVpiVci <calledVpiVci>`**

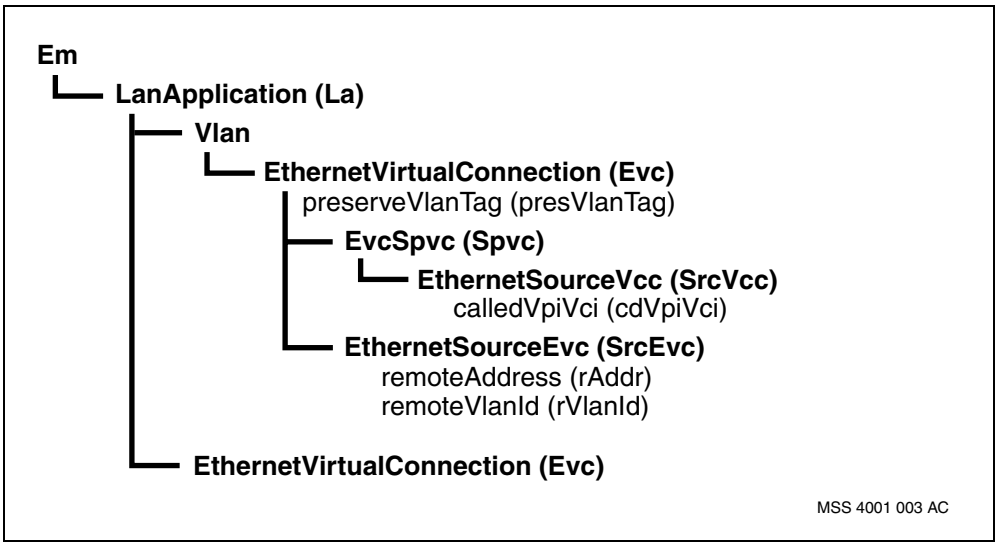
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<presVlanTag>	is the *preserveVlanTag* attribute. This attribute specifies if the EVC, in the direction from the Ethernet interface, preserves or removes the VLAN tag.
<remoteAddress>	is the NSAP address of the called-end ATM interface.
<Spvc>	is the instance value of the *Spvc* component.

Procedure job aid

1:1 VLAN-to-ATM EVC for LAN-to-VLAN transport component hierarchy



Configuring a 1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport

Configure a 1:1 VLAN-to-ATM Ethernet virtual connection (EVC) for VLAN-to-VLAN transport to terminate on an ATM virtual channel connection (VCC). The VCC can use one of the following protocols: ATM inter-network interface (AINI), interim inter-switch signalling protocol (IISP), private network-to-network interface (PNNI), or user-to-network interface (UNI). The EVC is configured as the calling end.

Procedure steps

Step	Action
1	Add the EVC to the VLAN. **add La/<la> Vlan/<vlan> Evc**
2	Add the EVC SPVC. **add La/<la> Vlan/<vlan> Evc Spvc/<Spvc>**
3	Add source EVC. **add La/<la> Vlan/<vlan> Evc SrcEvc**
4	Set the remote address. **set La/<la> Vlan/<vlan> Evc SrcEvc remoteAddress <remoteAddress>** The remote address should represent the NSAP address of the called-end ATM interface within the ATM network.
5	Set the remote VLAN ID. **set La/<la> Vlan/<vlan> Evc SrcEvc remoteVlanId <remoteVlanId>**
6	Add the source VCC to the SPVC. **add La/<la> Vlan/<vlan> Evc Spvc/<Spvc> SrcVcc**
7	Set the Vpi.Vci for the far-end ATM UNI. **set La/<la> Vlan/<vlan> Evc Spvc/<Spvc> SrcVcc calledVpiVci <calledVpiVci>**

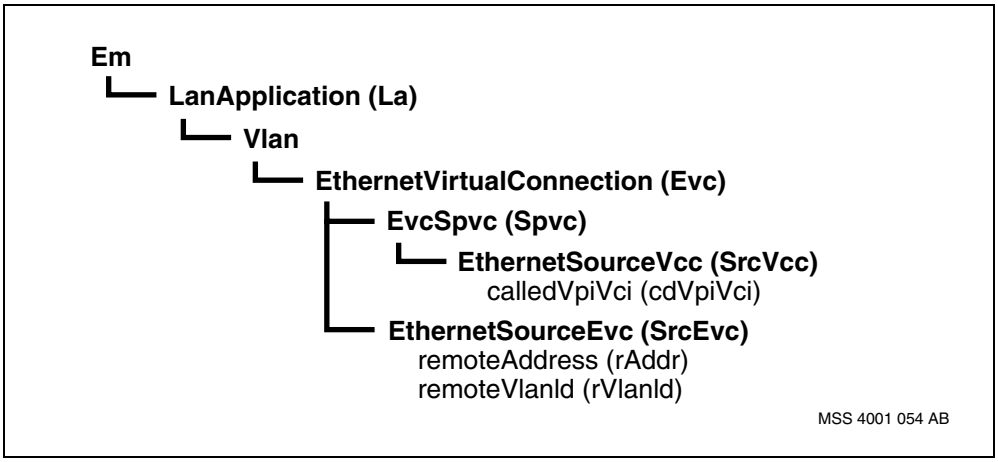
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<remoteAddress>	is the NSAP address of the called-end ATM interface.
<remoteVlanId>	is the instance value of the *remoteVlanId* attribute. The *remoteVlanId* attribute specifies the identifier of the VLAN at the far-end Ethernet interface. The value must be greater than zero.
<Spvc>	is the instance value of the *Spvc* component.

Procedure job aid

1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport component hierarchy



Configuring an N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport

Configure an N:1 VLAN-to-ATM Ethernet virtual connection (EVC) for VLAN-to-VLAN transport to terminate on an ATM virtual channel connection (VCC). The VCC can use one of the following protocols: ATM inter-network interface (AINI), interim inter-switch signalling protocol (IISP), private network-to-network interface (PNNI), or user-to-network interface (UNI). The EVC is configured as the calling end.

Prerequisites

- Ensure that Configuring ETS for single-ended EVLS ATM transport (page 62) has been completed.
- Steps for configuring a 1:1 VLAN-to-VLAN EVC must be performed and both ends must be configured as calling ends. See Configuring a 1:1 VLAN-to-VLAN EVC (page 29).

Procedure steps

Step	Action
1	Set the VLAN to terminate on the Ethernet transport system (ETS). **`set La/<la> Vlan/<vlan> Evc aggregation enabled`**
2	Repeat step 1 for each additional VLAN.
3	Repeat step 1 and step 2 to configure the other calling end on a second Multiservice Switch node.

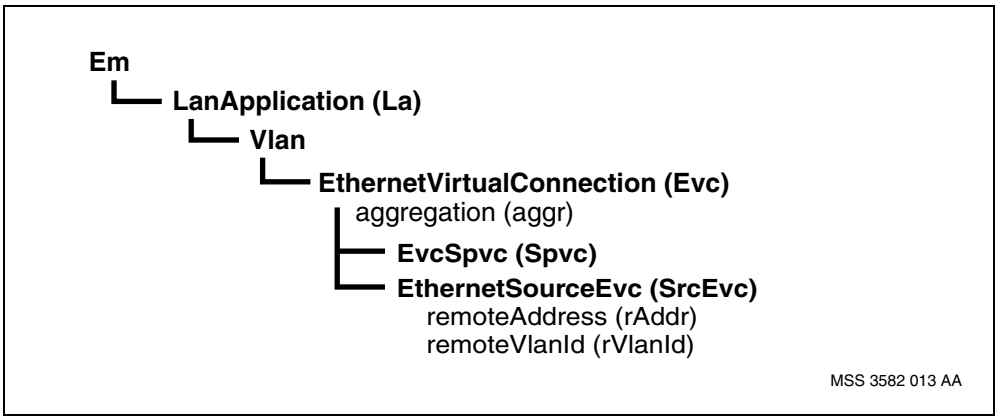
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<remoteAddress>	is the NSAP address of the far-end ATM interface.
<remoteVlanId>	is the instance value of the *remoteVlanId* attribute. The *remoteVlanId* attribute specifies the identifier of the VLAN at the far-end Ethernet interface.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component aggregating on the VCC.

Procedure job aid

N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport component hierarchy



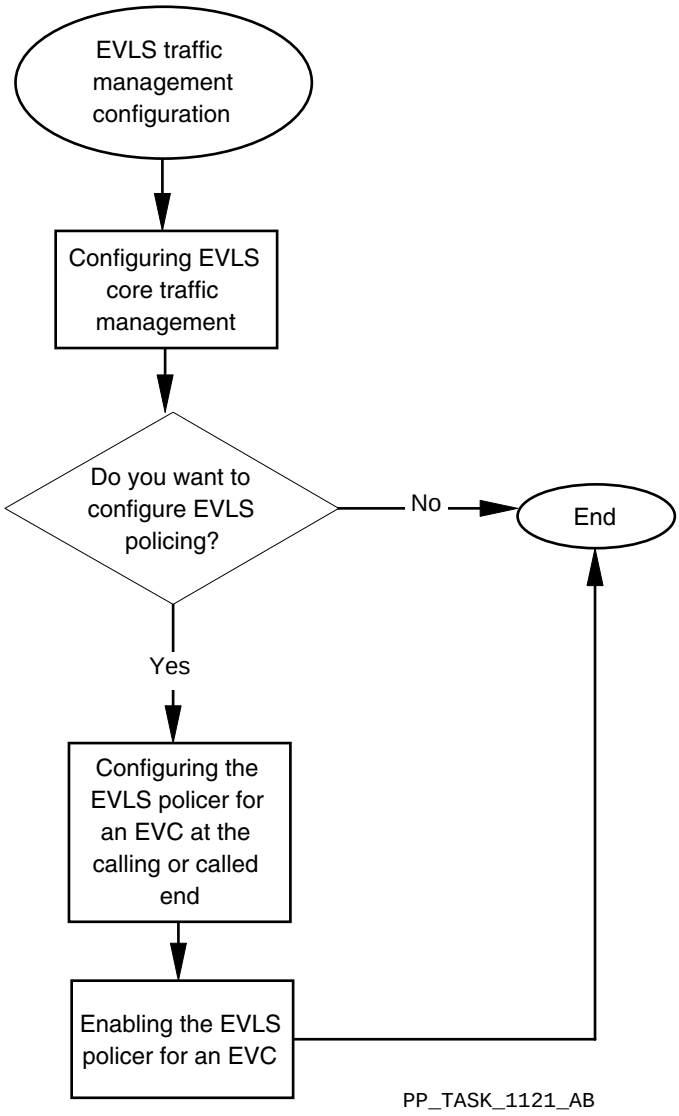
EVLS traffic management configuration

Configure EVLS traffic management to specify quality of service (QoS) parameters of the connection and to enable the network to meet the service level agreement (SLA).

EVLS traffic management configuration procedures

This task flow shows you the sequence of procedures you perform to configure EVLS traffic management. To link to any procedure, go to EVLS traffic management configuration procedure navigation (page 46).

EVLS traffic management configuration procedures



EVLS traffic management configuration procedure navigation

- Configuring EVLS core traffic management (page 47)
- Configuring the EVLS policer for an EVC at the calling or called end (page 50)
- Enabling the EVLS policer for an EVC (page 52)

Configuring EVLS core traffic management

Configure Ethernet virtual line service (EVLS) core traffic management at the calling end.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Add the Ethernet quality of service (EQoS). **add La/<la> Evc EQoS/<EQoS>** or **add La/<la> Vlan/<vlan> Evc EQoS/<EQoS>**
2	Set the Ethernet per-hop-behavior (ePHB). The *ePhb* attribute represents the Ethernet PHB for both directions of the connection. **set La/<la> Evc EQoS/<EQoS> ePhb <Phb>** or **set La/<la> Vlan/<vlan> Evc EQoS/<EQoS> ePhb <Phb>**
3	Optionally, set the *committedInformationRate (cir)*, *excessInformationRate (eir)*, *committedBurstSize (cbs)*, and *averageFrameSize* (afs) attributes. These attributes are used to derive ATM traffic management parameters. Traffic management techniques include the connection admission control (CAC) on an ATM interface. **set La/<la> Evc EQoS/<EQoS> cir <cir>** **set La/<la> Evc EQoS/<EQoS> eir <eir>** **set La/<la> Evc EQoS/<EQoS> cbs <cbs>** **set La/<la> Evc EQoS/<EQoS> afs <afs>** or **set La/<la> Vlan/<vlan> Evc EQoS/<EQoS> cir <cir>** **set La/<la> Vlan/<vlan> Evc EQoS/<EQoS> eir <eir>** **set La/<la> Vlan/<vlan> Evc EQoS/<EQoS> cbs <cbs>** **set La/<la> Vlan/<vlan> Evc EQoS/<EQoS> afs <afs>**
4	Optionally, set the partial packet discard. **set La/<la> Evc EQoS/<EQoS> packetWiseDiscard <pd>** or

```
set La/<la> Vlan/<vlan> Evc EQoS/<EQoS>
packetWiseDiscard <pd>
```

5 If there is another calling end, repeat step 1 to step 4 to configure it on a second Multiservice Switch node.

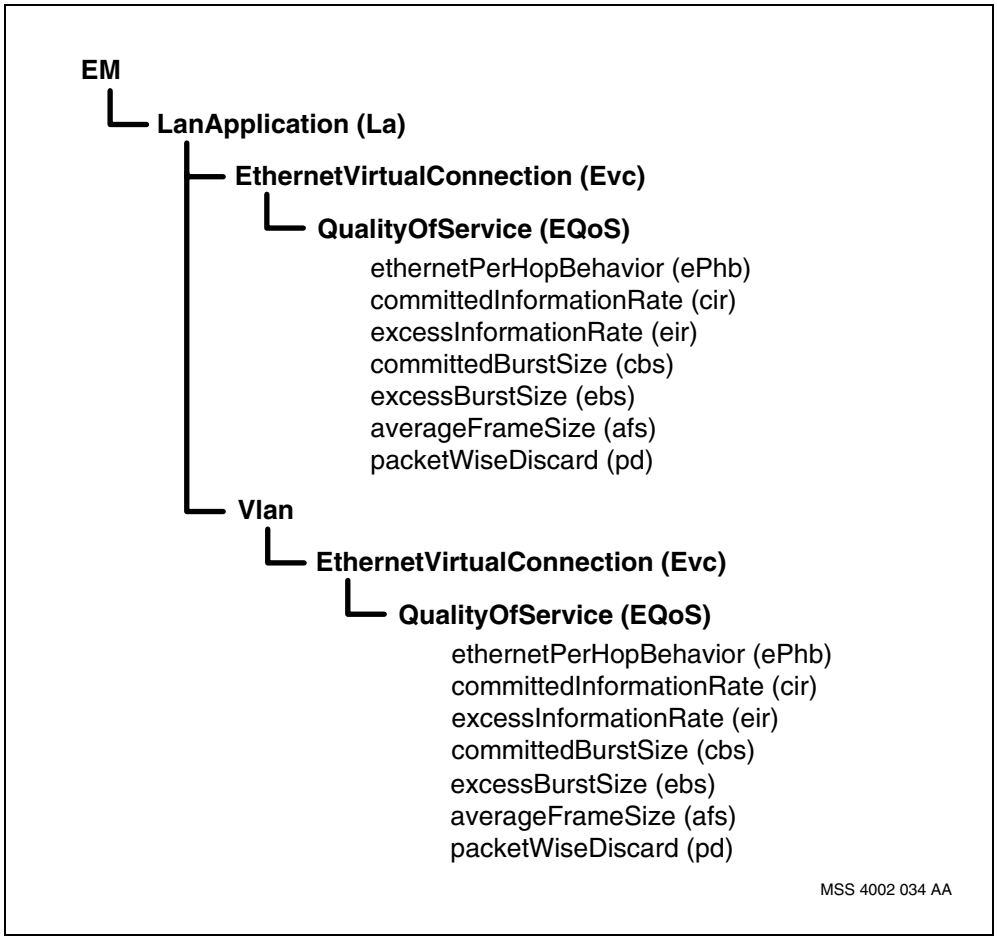
--End--

Variable definitions

Variable	Value
<afs>	is the average frame size that needs to be synchronized with the far-end EVC value. Default value is 512 bytes.
<cbs>	represents the maximum number of bits available for use by sequential service frames sent at the Ethernet interface speed to remain CIR conformant.
<cir>	represents the average rate up to which the network delivers service frames and meets the performance objectives defined by the Ethernet per hop behavior of this connection.
<eir>	represents the average rate up to which the network delivers service frames without any performance objectives.
<EQoS>	is the instance of the *EQoS* component.
<la>	is the instance value of the *LanApplication* component.
<pd>	is the packetWiseDiscard function for the VC. The default value is enabled.
<Phb>	is the *ePhb* attribute with a value of either EF, AF31, AF21, AF11, or DF. DF is the default value. This attribute specifies a mapping from a *PHB* attribute value to ATM service categories.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

EVLS core traffic management configuration component hierarchy



Configuring the EVLS policer for an EVC at the calling or called end

Optionally, configure the Ethernet virtual line service (EVLS) policer on each Ethernet virtual connection (EVC) at the calling end, called end, or both ends to specify parameters that define bandwidth requirements of a service level agreement.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Add EVLS policing to an EVC. **`add La/<la> Evc EvcPolicer`** or **`add La/<la> Vlan/<vlan> Evc EvcPolicer`**
2	Configure the *committedInformationRate (cir)*, *excessInformationRate (eir)*, *committedBurstSize (cbs)*, and *averageFrameSize* (afs) attributes. **`set La/<la> Evc EvcPolicer cir <cir>`** **`set La/<la> Evc EvcPolicer eir <eir>`** **`set La/<la> Evc EvcPolicer cbs <cbs>`** **`set La/<la> Evc EvcPolicer ebs <ebs>`** or **`set La/<la> Vlan/<vlan> Evc EvcPolicer cir <cir>`** **`set La/<la> Vlan/<vlan> Evc EvcPolicer eir <eir>`** **`set La/<la> Vlan/<vlan> Evc EvcPolicer cbs <cbs>`** **`set La/<la> Vlan/<vlan> Evc EvcPolicer ebs <ebs>`**
3	Repeat step 1 to step 2 if you want to configure the other end of the end-to-end Ethernet connection.

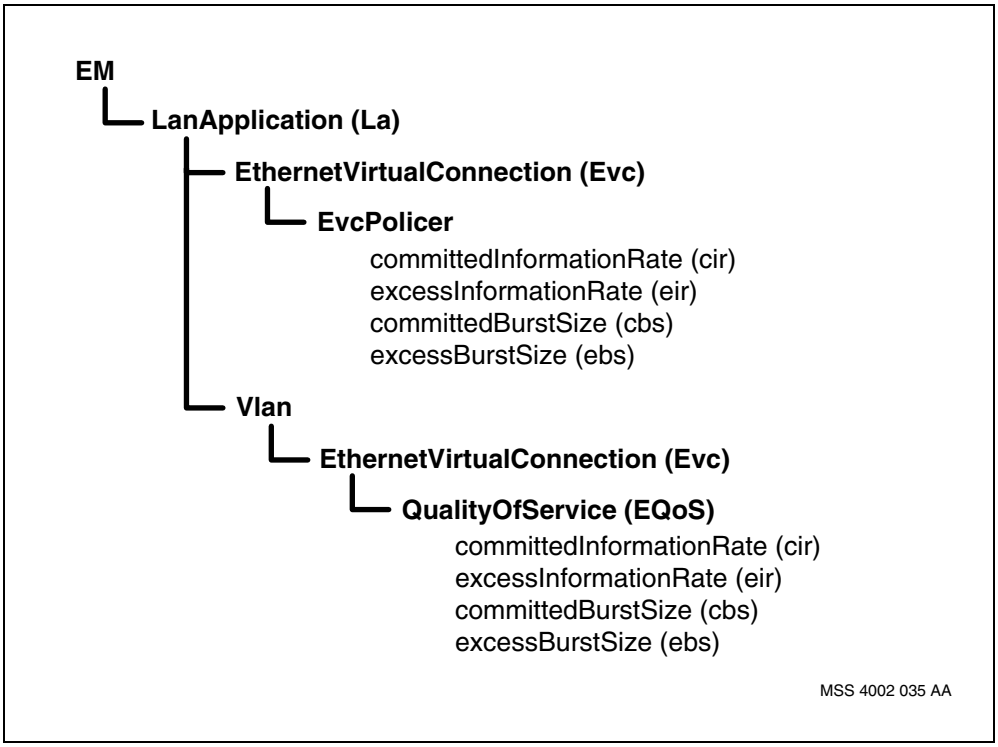
--End--

Variable definitions

Variable	Value
<ebs>	is the maximum amount of data (in bits) available for a burst of ingress Ethernet frames sent at the Ethernet access rate to remain EIR conformant.
<cbs>	represents the maximum number of bits available for use by sequential service frames sent at the Ethernet interface speed to remain CIR conformant.
<cir>	represents the average rate up to which the network delivers service frames and meets the performance objectives defined by the Ethernet per hop behavior of this connection.
<eir>	represents the average rate up to which the network delivers service frames without any performance objectives.
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

EVLS policer for an EVC at the calling end configuration component hierarchy



Enabling the EVLS policer for an EVC

Enable the Ethernet virtual line service (EVLS) policer for an Ethernet virtual connection to start traffic enforcement.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Enable EVLS policing for an EVC. **`set La/<la> Evc EvcPolicer policing enabled`** or **`set La/<la> Vlan/<vlan> Evc EvcPolicer policing enabled`**

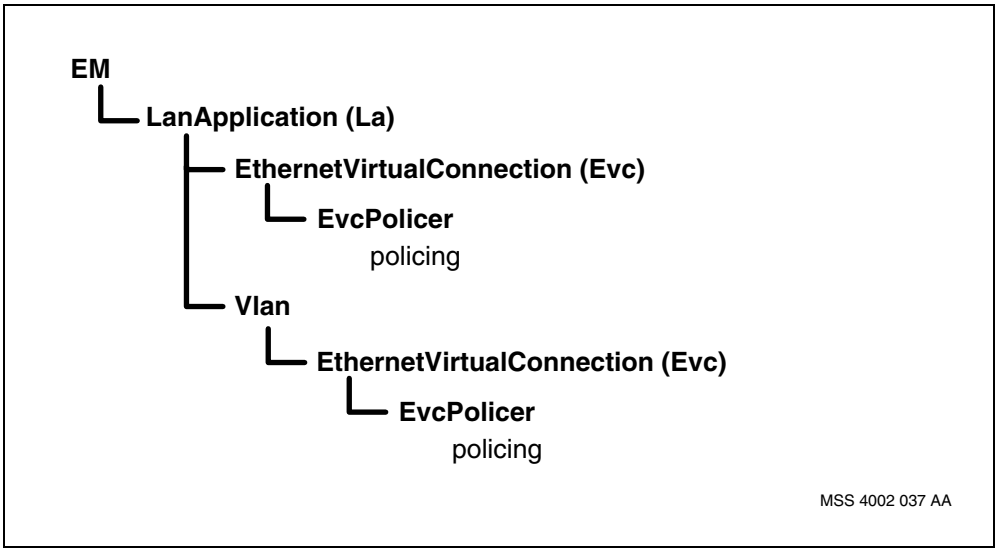
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

EVLS policer for an EVC component hierarchy



Ethernet transport system (ETS) configuration

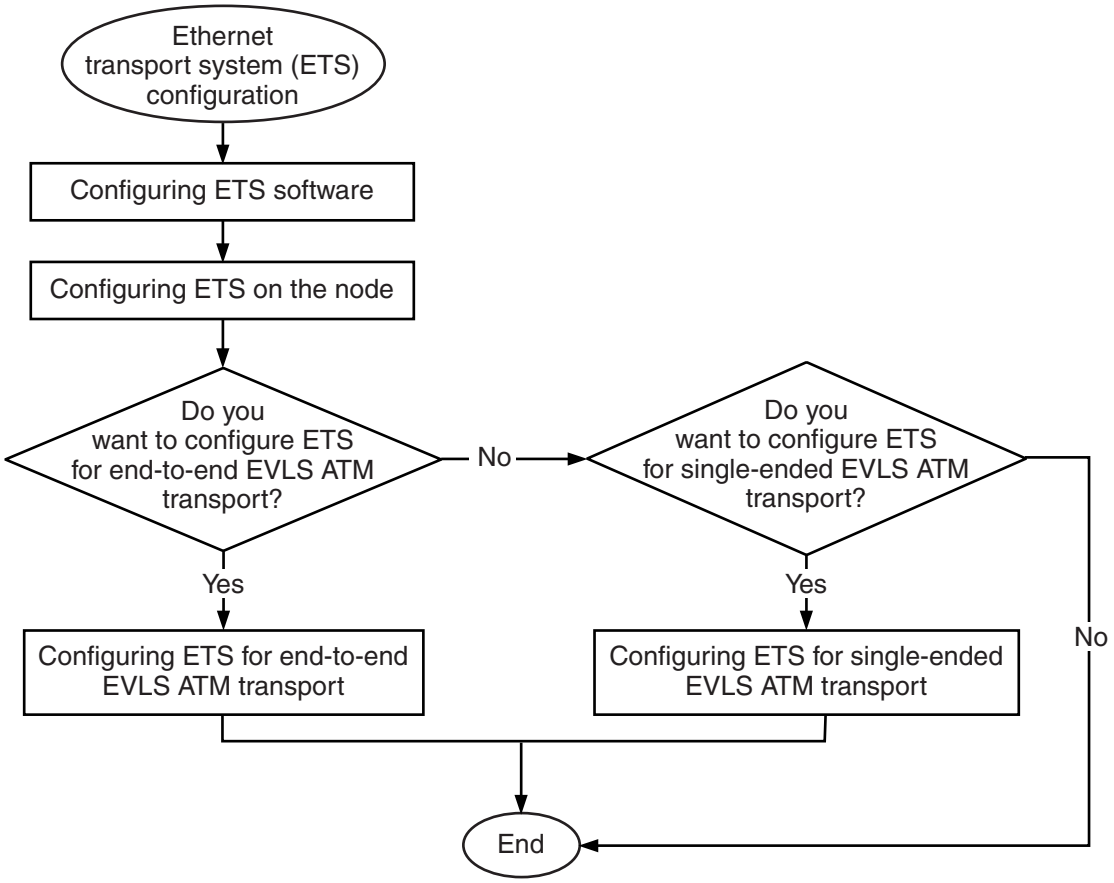
The Ethernet transport system (ETS) is a requirement if you want to configure Ethernet virtual connections (EVC) to take advantage of VLAN aggregation. ETS is only fully supported on the following control processor (CP) cards:

- CP2 for Multiservice Switch 7400
- CP3 for Multiservice Switch 15000 and Multiservice Switch 20000

Ethernet transport system (ETS) configuration procedures

This task flow shows you the sequence of procedures you perform to configure the Ethernet transport system (ETS). To link to any procedure, go to Ethernet transport system (ETS) procedure navigation (page 55).

Ethernet transport system (ETS) configuration procedures



Ethernet transport system (ETS) procedure navigation

- Configuring ETS software (page 56)
- Configuring ETS on the node (page 57)
- Configuring ETS for end-to-end EVLS ATM transport (page 58)
- Configuring ETS for single-ended EVLS ATM transport (page 62)

Configuring ETS software

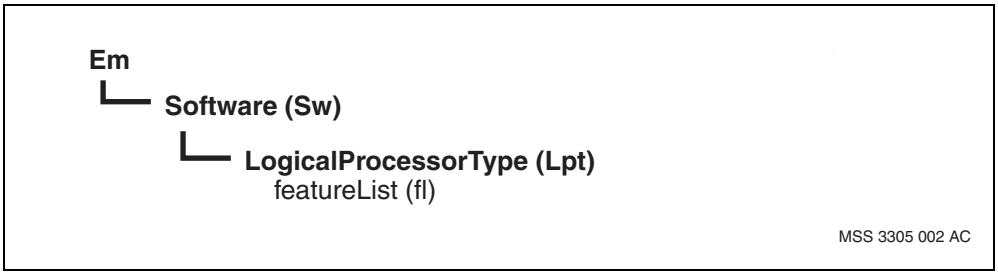
Configure the CP to load the ETS software on a single Multiservice Switch node.

Procedure steps

Step	Action
1	Provision the feature set on the CP. `set Sw Lpt/CP featureList Ets`
	--End--

Procedure job aid

ETS software component hierarchy



Configuring ETS on the node

Configure the Ethernet transport system to provide aggregation of traffic from multiple VLANs on a single Ethernet interface across a single ATM VCC.

Procedure steps

Step	Action
1	Add the ETS to the node. **`add Ets`**
2	Optionally, set the local NSAP address of the ETS within the ATM network. **`set Ets nsapAddr <nsapA>`**

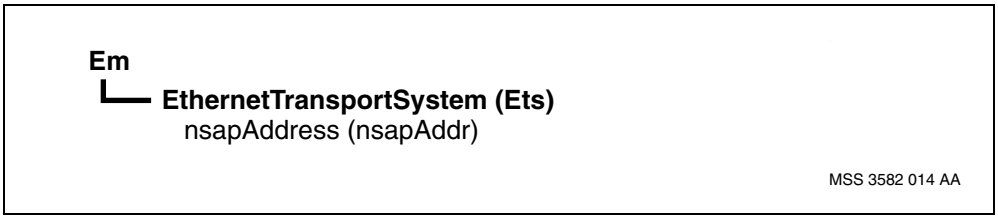
--End--

Variable definitions

Variable	Value
<nsapA>	is the value that represents the local address of the ETS (for example, the called endpoint) within the ATM network. If no NSAP address is configured, a system supplied value is used.

Procedure job aid

ETS on the node component hierarchy



Configuring ETS for end-to-end EVLS ATM transport

Configure an end-to-end EVLS ATM transport to provide aggregation of traffic from multiple VLANs between a pair of peer Ethernet interfaces across a single ATM VCC. The initial portion of this procedure involves configuring the called end of the ETS on the second Multiservice Switch node. The entire procedure involves configuring the calling end of the ETS on the first Multiservice Switch node.

Procedure steps

Step	Action
1	For the called end, complete step 2 to step 5. For the calling end, complete step 2 to step 10.
2	Add the *EvlsAtmTransport (EAtmTrsp)* component. **add Ets EAtmTrsp/<transport>**
3	Set the LAN application for the *EAtmTrsp* component. **set Ets EAtmTrsp/<transport> lanApplication La/<la>**
4	Set the prefix NSAP address for the *EAtmTrsp* component. **set Ets EAtmTrsp/<transport> prefixNsapAddress <prefixNsapA>**
5	Add the *EthSourceEvlsAtmTransport (SrcEAtmTrsp)* component to the EVLS ATM transport. **add Ets EAtmTrsp/<transport> SrcEAtmTrsp**
6	Set the remote address. **set Ets EAtmTrsp/<transport> SrcEAtmTrsp remoteAddress <remoteAddress>**
7	Set the remote EVLS ATM transport ID. **set Ets EAtmTrsp/<transport> SrcEAtmTrsp rEAtmTrspId <remoteEAtmTrspId>**
8	Optionally, set the retryPeriod for the SPVC. **set Ets EAtmTrsp/<transport> SrcEAtmTrsp retryPeriod <retryPeriod>**
9	Add the *EthernetOverAtmSpvc (Spvc)* component. **add Ets EAtmTrsp/<transport> Spvc/<Spvc>**
10	Set the SPVC Tm parameters. **set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm trafficDescType <trafficDescType>**

```
set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
trafficDescParm <trafficDescParm>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
atmServiceCategory <atmServiceCategory>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
bearerClassBbc <bearerClassBbc>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
transferCapabilityBbc <transferCapabilityBbc>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm bestEffort
<bestEffort>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
txPacketWiseDiscard <txPacketWiseDiscard>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
rxPacketWiseDiscard <rxPacketWiseDiscard>
```

--End--

Variable definitions

Variable	Value
<atmServiceCategory>	is the attribute value of the ATM service category for both directions of the connection.
<bearerClassBbc>	is the attribute value of the bearer capability for this connection.
<bestEffort>	is the attribute value of the best effort parameter in the ATM traffic descriptor information element.
<la>	is the instance number of the *LanApplication* component that the Ethernet Transport System uses.
<prefixNsapA>	is the attribute that specifies the 40-digit NSAP of the far-end *La Evls* component. The ETS uses this attribute to match an *EvlsAtmTransport* component with an EVC configured to use aggregation, comparing it with the EVC remoteAddress. The value of this attribute must be unique among all *Ets* components that share the same *La* value.
<remoteAddress>	is the NSAP address of the far-end ETS.
<remoteEAtmTrspId>	is the attribute value of the *EvlsAtmTransport* component on the far-end ETS to which this Atm transport is established.
<retryPeriod>	is the interval at which an SPVC connection attempts to re-establish after calling end of the SPVC initiates the setup request and receives a release message.

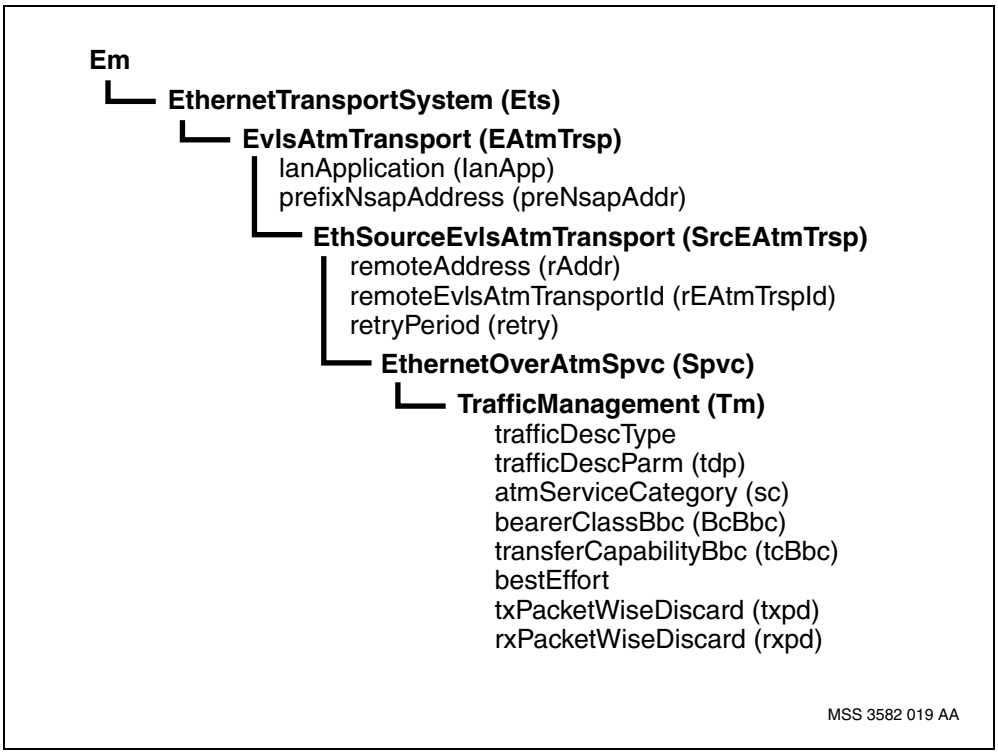
(1 of 2)

Variable	Value
<rxPacketWiseDiscard>	is the attribute value that indicates whether packet wise discard functions are enabled or disabled in the receive data direction for this connection.
<spvc>	is the instance value of the *Spvc* component.
<trafficDescParm>	is one of five transmit traffic parameters: peak cell rate (PCR), sustained cell rate (SCR), minimum cell rate (MCR), actual cell rate (ECR), and actual shaping rate.
<trafficDescType>	is the type of traffic management which is applied to the receive and transmit directions of this connection within the ATM network. The default value is 3.
<transferCapabilityBbc>	is the attribute value of the transfer capability for this connection.
<transport>	is the instance value of the *EAtmTrsp* component with a value from 1 to 256.
<txPacketWiseDiscard>	is the attribute value that indicates whether packet wise discard functions are enabled or disabled in the transmit data direction for this connection. The default value is enabled.

(2 of 2)

Procedure job aid

ETS for end-to-end EVLS ATM transport component hierarchy



Configuring ETS for single-ended EVLS ATM transport

Configure a single-ended EVLS ATM transport to provide aggregation of traffic from multiple VLANs from a single Ethernet interface to an ATM UNI across a single ATM VCC. The entire procedure involves configuring the calling end of the ETS on the Multiservice Switch node.

Procedure steps

Step	Action
1	Add the *EvlsAtmTransport (EAtmTrsp)* component. **add Ets EAtmTrsp/<transport>**
2	Set the LAN application for the *EAtmTrsp* component. **set Ets EAtmTrsp/<transport> lanApplication La/<la>**
3	Set the prefix NSAP address for the *EAtmTrsp* component. **set Ets EAtmTrsp/<transport> prefixNsapAddress <prefixNsapA>**
4	Add the source EVLS ATM transport component to the EVLS ATM transport. **add Ets EAtmTrsp/<transport> SrcEAtmTrsp**
5	Set the remote address. **set Ets EAtmTrsp/<transport> SrcEAtmTrsp remoteAddress <remoteAddress>**
6	Set the remote EVLS ATM transport ID. **set Ets EAtmTrsp/<transport> SrcEAtmTrsp rEAtmTrspId <remoteEAtmTrspId>**
7	Optionally, set the retryPeriod for the SPVC. **set Ets EAtmTrsp/<transport> SrcEAtmTrsp retryPeriod <retryPeriod>**
8	Add the *EthernetOverAtmSpvc* (*Spvc*) component. **add Ets EAtmTrsp/<transport> Spvc/<spvc>**
9	Set the SPVC Tm parameters. **set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm trafficDescType <trafficDescType>** **set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm trafficDescParm <trafficDescParm>** **set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm atmServiceCategory <atmServiceCategory>** **set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm bearerClassBbc <bearerClassBbc>**

```
set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
transferCapabilityBbc <transferCapabilityBbc>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm bestEffort
<bestEffort>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
txPacketWiseDiscard <txPacketWiseDiscard>

set Ets EAtmTrsp/<transport> Spvc/<spvc> Tm
rxPacketWiseDiscard <rxPacketWiseDiscard>
```

10 Add the source VCC to the *Ets* component.

```
add Ets EAtmTrsp/<transport> Spvc/<spvc> SrcVcc
```

11 Set the VPI.VCI for the ATM UNI.

```
set Ets EAtmTrsp/<transport> Spvc/<spvc> SrcVcc
calledVpiVci <calledVpiVci>
```

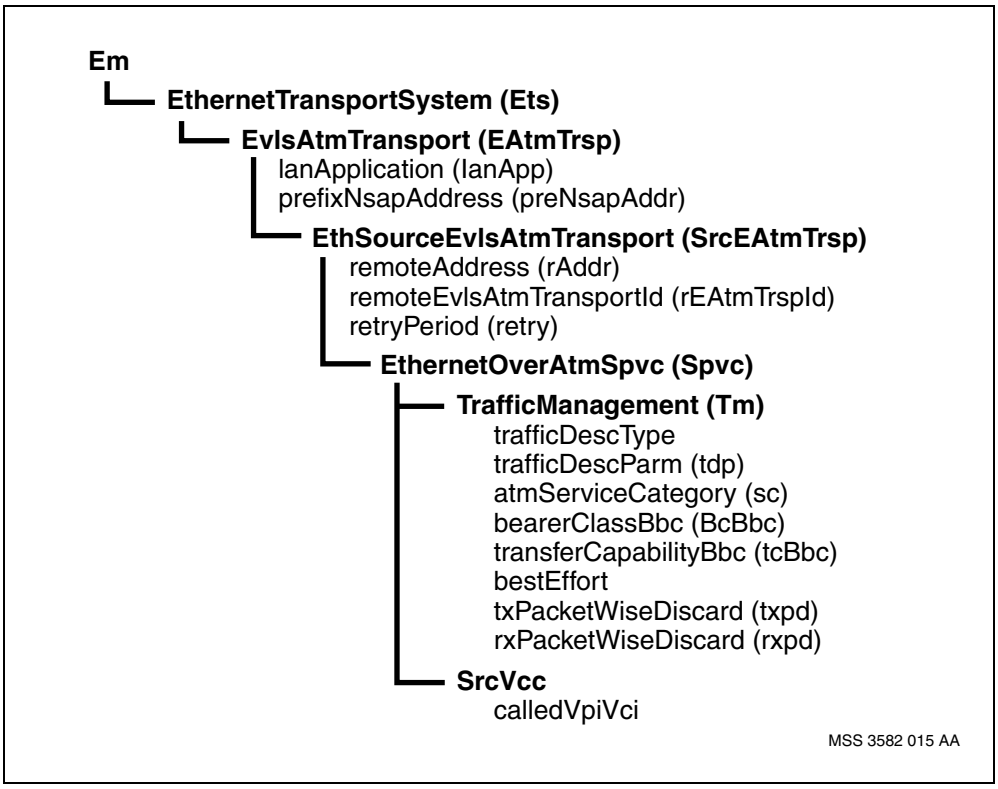
--End--

Variable definitions

Variable	Value
<atmServiceCategory>	is the instance value of the ATM service category for both directions of the connection.
<bearerClassBbc>	is the instance value of the bearer capability for this connection.
<bestEffort>	is the instance value of the best effort parameter in the ATM traffic descriptor information element.
<calledVpiVci>	is the attribute that specifies the far-end ATM UNI VCC that is the called end of this EVC.
<la>	is the instance number of the *LanApplication* component that the Ethernet Transport System uses.
<prefixNsapA>	is the attribute that specifies the 40-digit NSAP of the far-end *La Evls* component. The ETS uses this attribute to match an *EvlsAtmTransport* component with an EVC configured to use aggregation, comparing it with the EVC remoteAddress. The value of this attribute must be unique among all *Ets* components that share the same *La* value.
<remoteAddress>	is the NSAP address of the far-end ATM UNI or ATM PNNI.
<remoteEAtmTrspId>	is the attribute value of the *EvlsAtmTransport* component on the far-end ETS to which this Atm transport is established.
<retryPeriod>	is the interval at which an SPVC connection attempts to re-establish after calling end of the SPVC initiates the setup request and receives a release message.
<rxPacketWiseDiscard>	is the attribute value that indicates whether packet wise discard functions are enabled or disabled in the receive data direction for this connection.
<spvc>	is the instance value of the *Spvc* component.
<trafficDescParm>	is one of five transmit traffic parameters: peak cell rate (PCR), sustained cell rate (SCR), minimum cell rate (MCR), actual cell rate (ECR), and actual shaping rate.
<trafficDescType>	is the type of traffic management which is applied to the receive and transmit directions of this connection within the ATM network. The default value is 3.
<transferCapabilityBbc>	is the attribute value of the transfer capability for this connection.
<transport>	is the instance value of the *EAtmTrsp* component with a value from 1 to 256.
<txPacketWiseDiscard>	is the instance value that indicates whether packet wise discard functions are enabled or disabled in the transmit data direction for this connection. The default value is enabled.

Procedure job aid

ETS for single-ended EVLS ATM transport component hierarchy



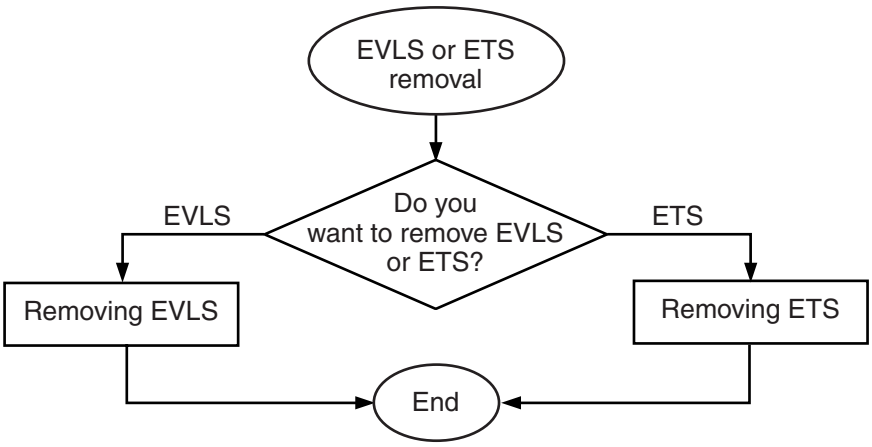
EVLS or ETS removal

You can remove the Ethernet virtual line service (EVLS) or the Ethernet transport system (ETS) functionality, or both.

EVLS or ETS removal procedures

This task flow shows you the sequence of procedures you perform to remove EVLS or ETS. To link to any procedure, go to EVLS or ETS removal procedure navigation (page 66).

EVLS or ETS removal procedures



MSS 3605 001 AA

EVLS or ETS removal procedure navigation

- Removing EVLS (page 67)
- Removing ETS (page 68)

Removing EVLS

Remove this feature if you no longer want to take advantage of the Ethernet virtual line service (EVLS) functionality.

Procedure steps

Step	Action

CAUTION
Risk of loss of service
The removal of the EVLS from the node causes the shelf to reset.

1 Delete EVLS from all LanApplications on the FP.

```
delete La/<la> Evls
```

2 Remove EVLS software from the Ethernet FP.

```
set Sw Lpt/<lpt> featureList ~evls
```

--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<lpt>	is the instance value of the *LogicalProcessorType* component.

Procedure job aid

EVLS component hierarchy

Removing ETS

Remove this feature if you no longer want to take advantage of Ethernet transport system (ETS) functionality.

Procedure steps

Step	Action

CAUTION
Risk of loss of service
The removal of the ETS from the node causes the shelf to reset.

1 Delete the Ethernet transport system (ETS) from the node.

```
delete Ets
```

2 Disable aggregation for all VLAN EVCs on the node.

```
set La/<la> Vlan/<vlan> Evc aggregation disabled
```

3 Remove the Ethernet transport system (ETS) from the CP.

```
set Sw Lpt/<lpt> featureList ~ets
```

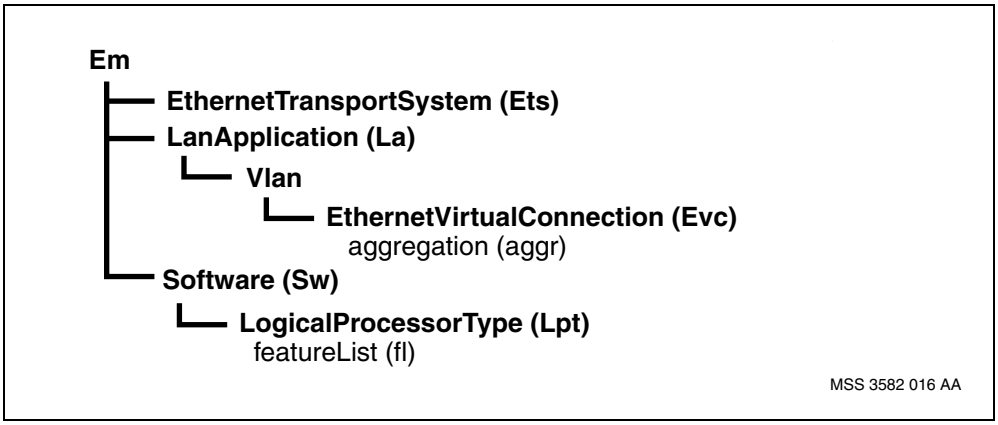
--End--

Variable definitions

Variable	Value
<la>	is the instance number of the *LanApplication* component that the Ethernet Transport System uses.
<lpt>	is the instance value of the *LogicalProcessorType* component.
<vlan>	is the instance value of the *Vlan* component aggregating on the VCC.

Procedure job aid

ETS component hierarchy



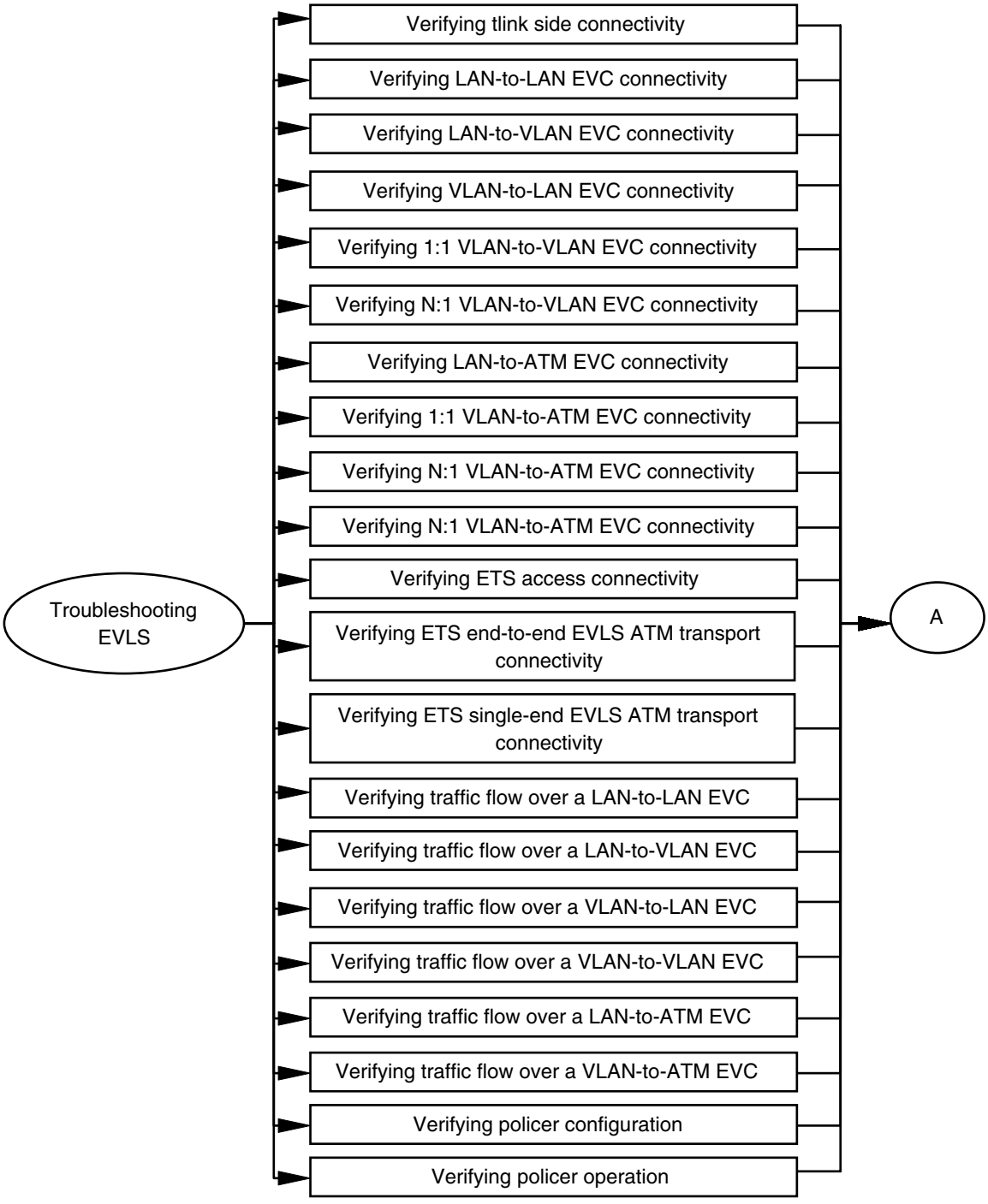
Troubleshooting EVLS

Troubleshoot the Ethernet virtual line services (EVLS) to determine the possible reasons behind connection failure, congestion, and data loss.

Troubleshooting EVLS procedures

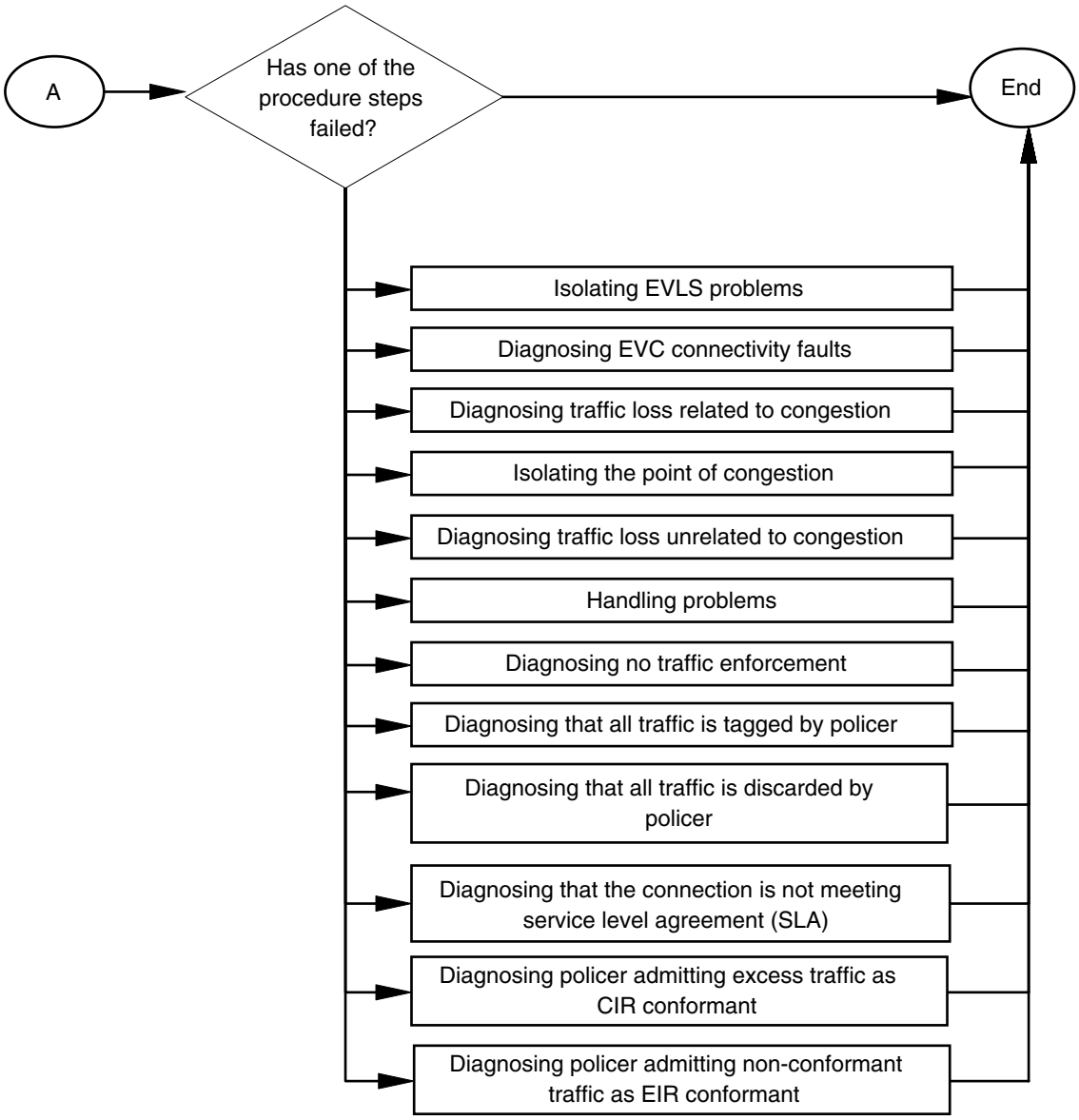
This task flow shows you the procedures you perform to troubleshoot EVLS. To link to any procedure, go to Troubleshooting EVLS procedure navigation (page 72).

Troubleshooting EVLS procedures, part 1 of 2



MSS 4001 037 AB

Troubleshooting EVLS procedures, part 2 of 2



MSS 3346 001 AC

Troubleshooting EVLS procedure navigation

- Verifying link side connectivity (page 74)
- Verifying LAN-to-LAN EVC connectivity (page 75)
- Verifying LAN-to-VLAN EVC connectivity (page 77)
- Verifying VLAN-to-LAN EVC connectivity (page 79)

- Verifying 1:1 VLAN-to-VLAN EVC connectivity (page 81)
- Verifying N:1 VLAN-to-VLAN EVC connectivity (page 83)
- Verifying LAN-to-ATM EVC connectivity (page 85)
- Verifying 1:1 VLAN-to-ATM EVC connectivity (page 87)
- Verifying N:1 VLAN-to-ATM EVC connectivity (page 89)
- Verifying ETS access connectivity (page 91)
- Verifying ETS end-to-end EVLS ATM transport connectivity (page 93)
- Verifying ETS single-ended EVLS ATM transport connectivity (page 95)
- Verifying traffic flow over a LAN-to-LAN EVC (page 97)
- Verifying traffic flow over a LAN-to-VLAN EVC (page 99)
- Verifying traffic flow over a VLAN-to-LAN EVC (page 101)
- Verifying traffic flow over a VLAN-to-VLAN EVC (page 103)
- Verifying traffic flow over a LAN-to-ATM EVC (page 105)
- Verifying traffic flow over a VLAN-to-ATM EVC (page 107)
- Verifying policer configuration (page 109)
- Verifying policer operation (page 111)
- Isolating EVLS problems (page 113)
- Diagnosing EVC connectivity faults (page 114)
- Diagnosing traffic loss related to congestion (page 116)
- Isolating the point of congestion (page 118)
- Diagnosing traffic loss unrelated to congestion (page 120)
- Handling problems (page 123)
- Diagnosing no traffic enforcement (page 128)
- Diagnosing that all traffic is tagged by policer (page 131)
- Diagnosing that all traffic is discarded by policer (page 133)
- Diagnosing that the connection is not meeting service level agreement (SLA) (page 135)
- Diagnosing policer admitting excess traffic as CIR conformant (page 137)
- Diagnosing policer admitting non-conformant traffic as EIR conformant (page 139)

Verifying link side connectivity

Ensure that the provider edge (PE) Ethernet interface is connected to the customer edge (CE) Ethernet interface.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

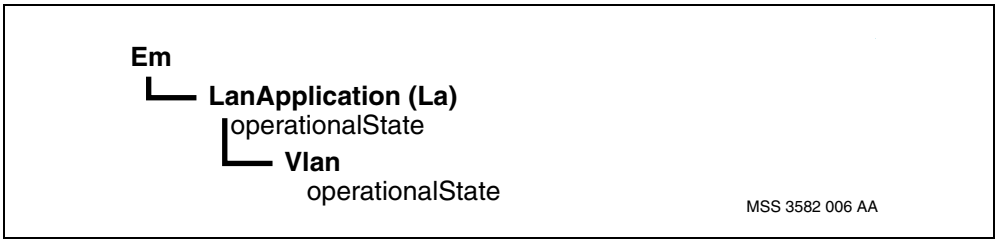
Step	Action
1	Ensure that the Ethernet interface and VLANs are operational. **`display La/<la> operationalState`** **`display La/<la> Vlan/<vlan> operationalState`**
	--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Link side connectivity component hierarchy



Verifying LAN-to-LAN EVC connectivity

Verify that an EVC configured for LAN-to-LAN connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Ensure that either of the two end-to-end EVCs are operational. The value of the *usageState* attribute must be busy. **`display La/<la> Evc usageState`**
2	Verify that the EVC SPVC is connected. **`display La/<la> Evc Spvc/<spvc> spvcStatus`**
3	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Evc Spvc/<spvc>`**
4	Determine which ATM connection is connected to the SPVC. **`display La/<la> Evc Spvc/<spvc> AtmCon`**
5	Determine the far-end EVC that is connected to this EVC. **`display La/<la> Evc remoteAddress, remoteVlanId`**
6	Repeat step 1 and step 2 to verify that the far-end EVC is connected.

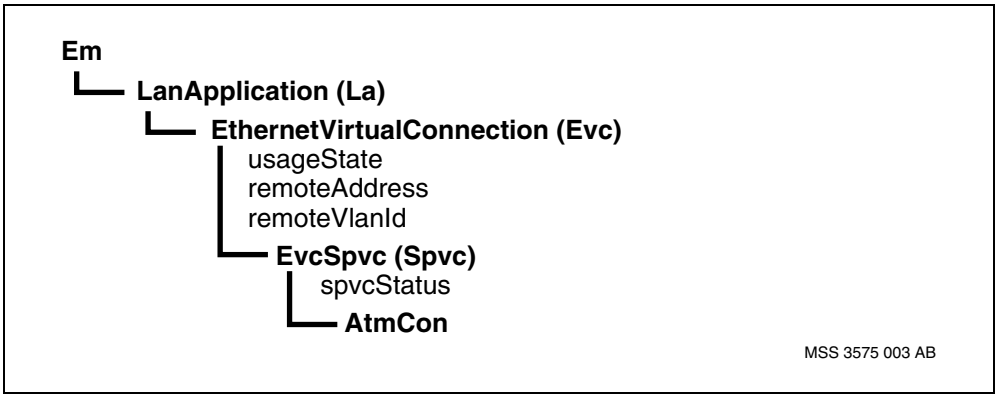
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.

Procedure job aid

LAN-to-LAN EVC connectivity component hierarchy



Verifying LAN-to-VLAN EVC connectivity

Verify that an EVC configured for LAN-to-VLAN connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	To verify from the LAN end of the EVC, complete step 2 to step 8. To verify from the VLAN end of the EVC, complete step 7 to step 11, step 2, and step 3.
2	Ensure that the LAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Evc usageState`**
3	Verify that the EVC SPVC is connected. **`display La/<la> Evc Spvc/<spvc> spvcStatus`**
4	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Evc Spvc/<spvc>`**
5	Determine which ATM connection is connected to the SPVC. **`display La/<la> Evc Spvc/<spvc> AtmCon`**
6	Determine the far-end EVC that is connected to this EVC. **`display La/<la> Evc remoteAddress, remoteVlanId`**
7	Ensure that the VLAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Vlan/<vlan> Evc usageState`**
8	Verify that the EVC SPVC is connected. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> spvcStatus`**
9	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
10	Determine which ATM connection is connected to the SPVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> AtmCon`**
11	Determine the far-end EVC that is connected to this EVC. **`display La/<la> Vlan/<vlan> Evc remoteAddress, remoteVlanId`**

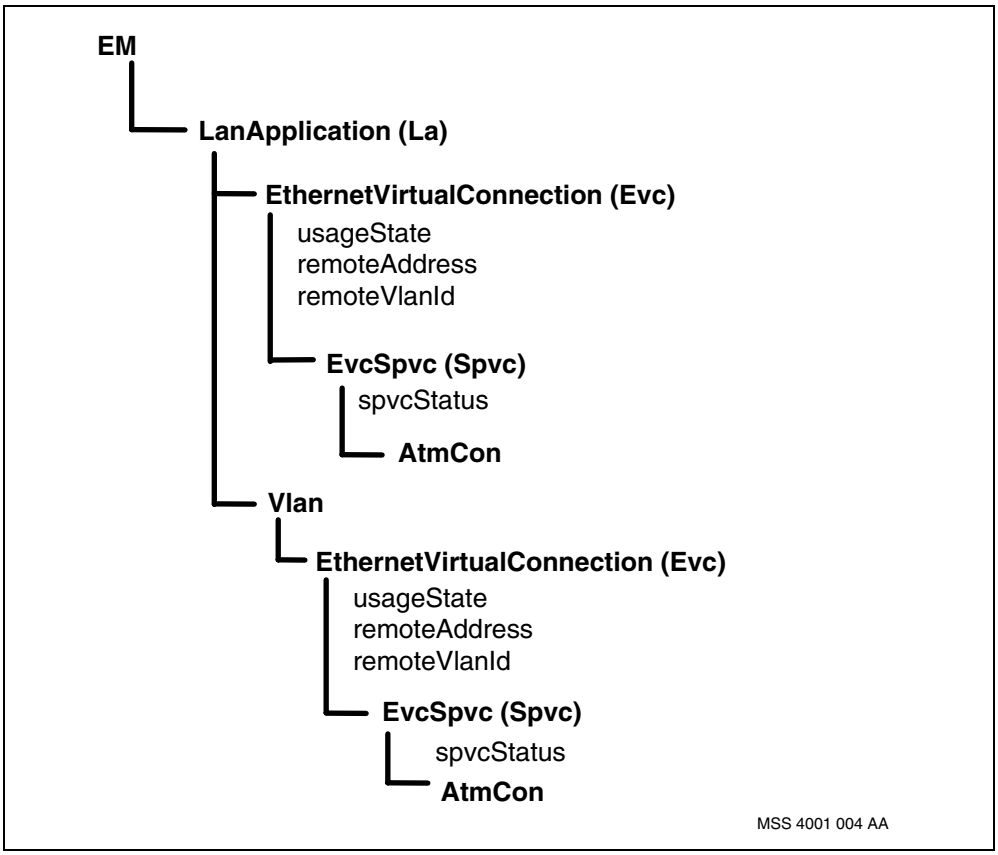
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

LAN-to-VLAN EVC connectivity component hierarchy



Verifying VLAN-to-LAN EVC connectivity

Verify that an EVC configured for VLAN-to-LAN connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	To verify the VLAN end of the EVC, complete step 2 to step 8. To verify the LAN end of the EVC, complete step 7 to step 11, step 2 and step 3.
2	Ensure that VLAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Vlan/<vlan> Evc usageState`**
3	Verify that the EVC SPVC is connected. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> spvcStatus`**
4	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
5	Determine which ATM connection is connected to the SPVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> AtmCon`**
6	Determine the far-end EVC that is connected to this EVC. **`display La/<la> Vlan/<vlan> Evc remoteAddress, remoteVlanId`**
7	Ensure that the LAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Evc usageState`**
8	Verify that the EVC SPVC is connected. **`display La/<la> Evc Spvc/<spvc> spvcStatus`**
9	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Evc Spvc/<spvc>`**
10	Determine which ATM connection is connected to the SPVC. **`display La/<la> Evc Spvc/<spvc> AtmCon`**
11	Determine the far-end EVC that is connected to this EVC. **`display La/<la> Evc remoteAddress, remoteVlanId`**

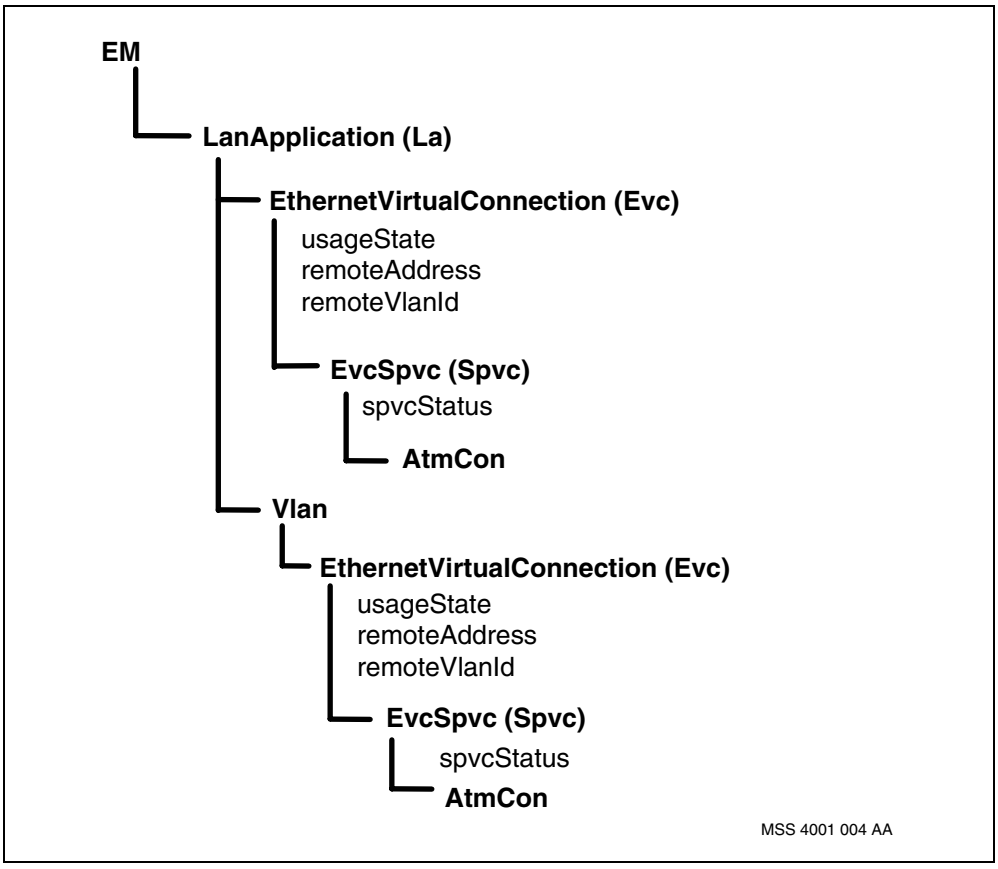
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

VLAN-to-LAN EVC connectivity component hierarchy



Verifying 1:1 VLAN-to-VLAN EVC connectivity

Verify that an EVC configured for VLAN-to-VLAN connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Ensure that either of the two end-to-end VLAN EVCs are operational. The value of the *usageState* attribute must be busy. **`display La/<la> Vlan/<vlan> Evc usageState`**
2	Verify that the EVC SPVC is connected. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> spvcStatus`**
3	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
4	Determine which ATM connection is connected to the SPVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> AtmCon`**
5	Determine the far-end EVC that is connected to this EVC. **`display La/<la> Vlan/<vlan> Evc remoteAddress, remoteVlanId`**
6	Repeat step 1 and step 2 to verify that the far-end EVC is connected.

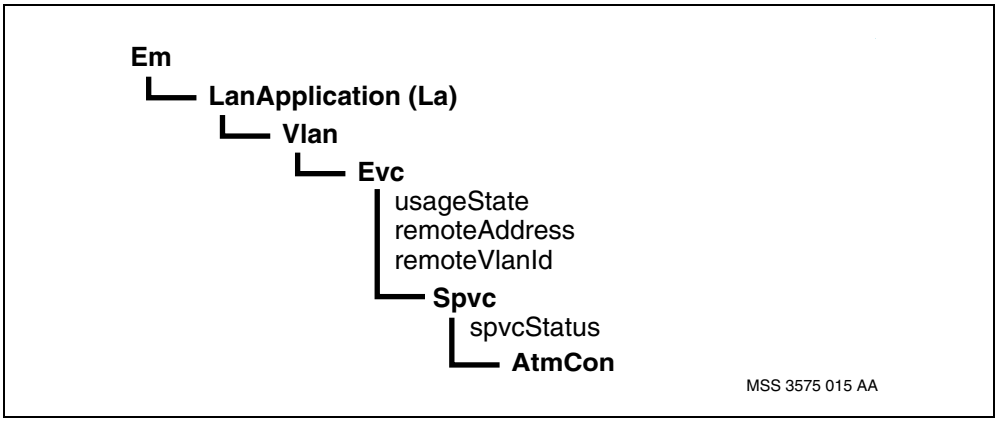
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

1:1 VLAN-to-VLAN EVC connectivity component hierarchy



Verifying N:1 VLAN-to-VLAN EVC connectivity

Verify that the end-to-end EVC configured for aggregation is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.
- Verify ETS connectivity. See Verifying ETS access connectivity (page 91) and Verifying ETS end-to-end EVLS ATM transport connectivity (page 93).
- Verify that the end-to-end EVC configured for aggregation is operational.

Procedure steps

Step	Action
1	Ensure that either of the two end-to-end VLAN EVCs are operational. The value of the *usageState* attribute must be busy. **`display La/<la> Vlan/<vlan> Evc usageState`**
2	Verify the status of the EVC SPVC is aggregationConnected. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> aggregationStatus`**
3	Verify that an *EtsTrspCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
4	Determine the ETS EVLS ATM transport that has admitted the EVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> EtsTrspCon`**
5	Determine the far-end Ethernet interface. **`display La/<la> Vlan/<vlan> Evc remoteAddress, remoteVlanId`**
6	Repeat step 1 through step 4 for the far-end EVC.

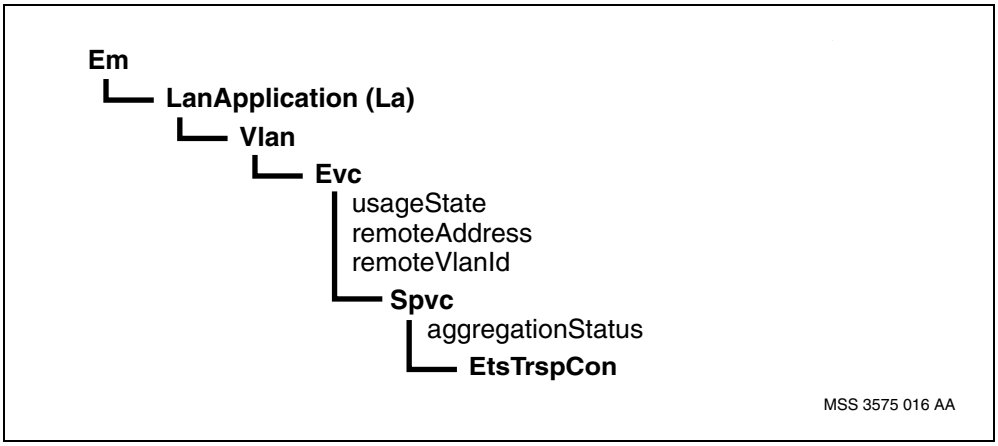
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component aggregating on the VCC.

Procedure job aid

N:1 VLAN-to-VLAN EVC connectivity component hierarchy



Verifying LAN-to-ATM EVC connectivity

Verify that the EVC configured for LAN-to-ATM connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Ensure that the single-ended LAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Evc usageState`**
2	Verify that the EVC SPVC is connected. **`display La/<la> Evc Spvc/<spvc> spvcStatus`**
3	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Evc Spvc/<spvc>`**
4	Determine which ATM connection is connected to the SPVC. **`display La/<la> Evc Spvc/<spvc> AtmCon`**
5	Verify ATM network connectivity. Ensure that the ATM UNI VCC is operational. See NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

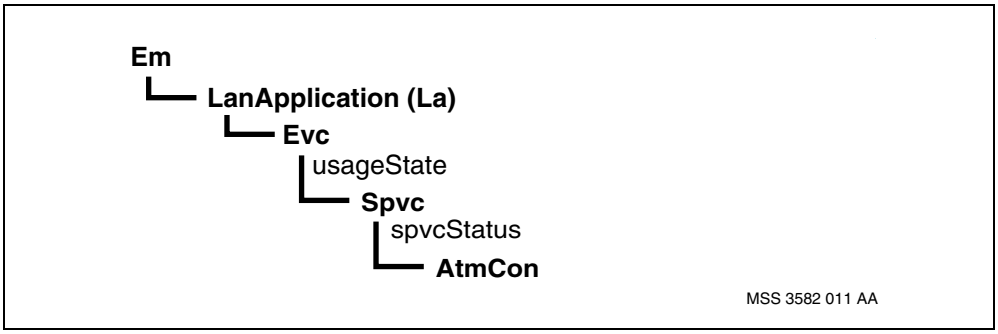
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.

Procedure job aid

LAN-to-ATM EVC connectivity component hierarchy



Verifying 1:1 VLAN-to-ATM EVC connectivity

Verify that the EVC configured for 1:1 VLAN-to-ATM connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Ensure that the single-ended VLAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Vlan/<vlan> Evc usageState`**
2	Verify that the EVC SPVC is connected. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> spvcStatus`**
3	Verify that an *AtmCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
4	Determine which ATM connection is connected to the SPVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> AtmCon`**
5	Verify ATM network connectivity. Ensure that the ATM UNI VCC is operational. See NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
6	If you are using a single-ended EVC in a performance enhancement configuration, verify that the traffic flows across the ATM SPVC connecting the two UNIs.
7	Repeat step 1 to step 5 to verify the connectivity of the far-end single-ended EVC.

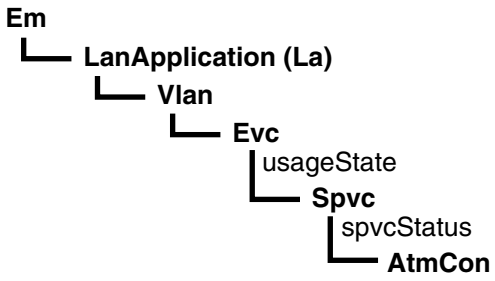
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

1:1 VLAN-to-ATM EVC connectivity component hierarchy



Verifying N:1 VLAN-to-ATM EVC connectivity

Verify that the EVC configured for N:1 VLAN-to-ATM connectivity is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Ensure that the single-ended VLAN EVC is operational. The value of the *usageState* attribute must be busy. **`display La/<la> Vlan/<vlan> Evc usageState`**
2	Verify the status of the EVC SPVC is aggregationConnected. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> aggregationStatus`**
3	Verify that an *EtsTrspCon* subcomponent exists under the configured EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
4	Determine the ETS EVLS ATM transport that has admitted the EVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> EtsTrspCon`**
5	Verify ETS access connectivity. See Verifying ETS access connectivity (page 91).
6	Verify ETS single-ended EVLS ATM transport connectivity. See Verifying ETS single-ended EVLS ATM transport connectivity (page 95).

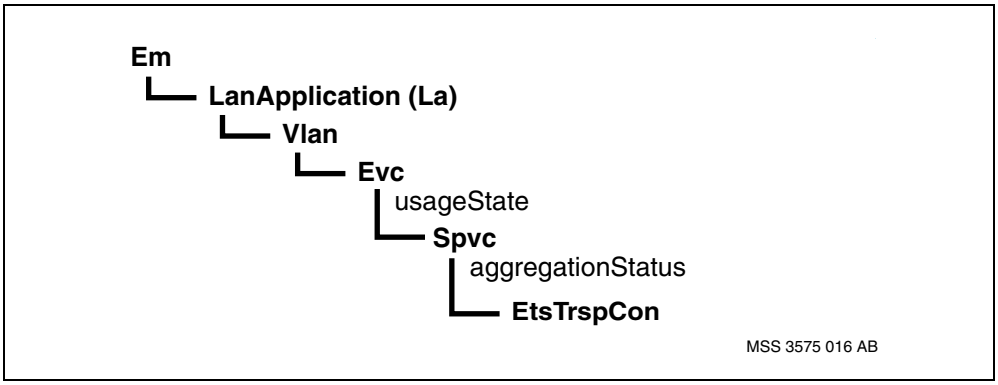
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component aggregating on the VCC.

Procedure job aid

N:1 VLAN-to-ATM EVC connectivity component hierarchy



Verifying ETS access connectivity

Verify that an EVC has established connectivity over an ETS EVLS ATM transport.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Verify that an *EtsTrspCon* subcomponent exists under the EVC SPVC. **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
2	Determine which *Ets EvlsAtmTransport Spvc* component the EVC is connected to. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> EtsTrspCon`**
3	Verify that the *Ets EvlsAtmTransport* component is operational. **`display Ets EvlsAtmTransport/<transport> osiState`**
4	Verify that one of the service categories of the *admittedConnections* attribute has incremented by 1. **`display Ets EvlsAtmTransport/<transport> admittedConnections`**
5	Repeat step 1 through step 4 at the far-end ETS.

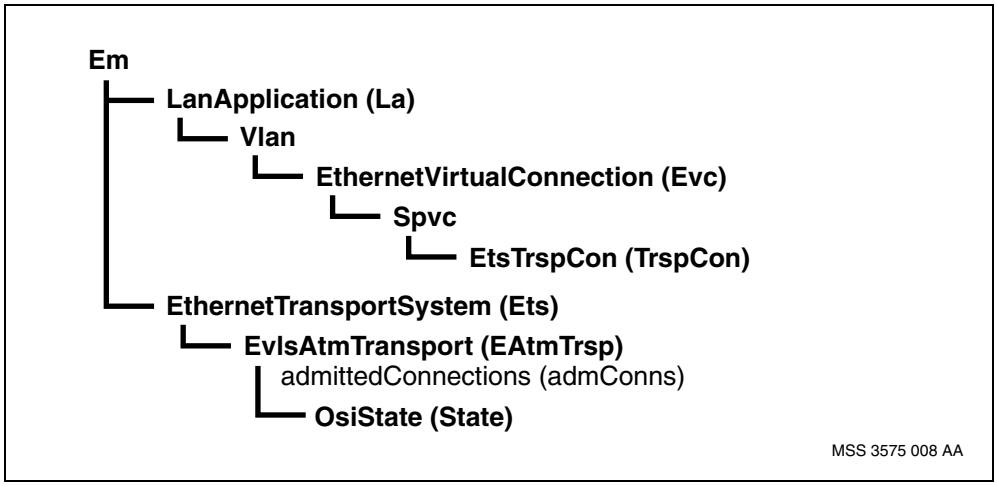
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component aggregating on the VCC.

Procedure job aid

ETS access connectivity component hierarchy



Verifying ETS end-to-end EVLS ATM transport connectivity

Verify that the ETS EVLS ATM transport establishes end-to-end connectivity to another ETS EVLS ATM transport on another node over an ATM network.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Verify that the status of the EVLS ATM transport SPVC is connected. **display Ets EvlsAtmTransport/<transport> Spvc/<spvc> spvcStatus**
2	Verify that the *AtmConnection* component exists under each of the *EvlsAtmTransport Spvc* components. **list Ets EvlsAtmTransport/<transport> Spvc/<spvc>**
3	Determine which ATM connection is connected to the SPVC. **display Ets EvlsAtmTransport/<transport> Spvc/<spvc> AtmConnection**
4	Choose one of the SPVCs from step 2 to determine the far-end *EvlsAtmTransport* to which this *EvlsAtmTransport* component is connected. **display Ets EvlsAtmTransport/<transport> remoteAddress, remoteEvlsAtmTransportId**

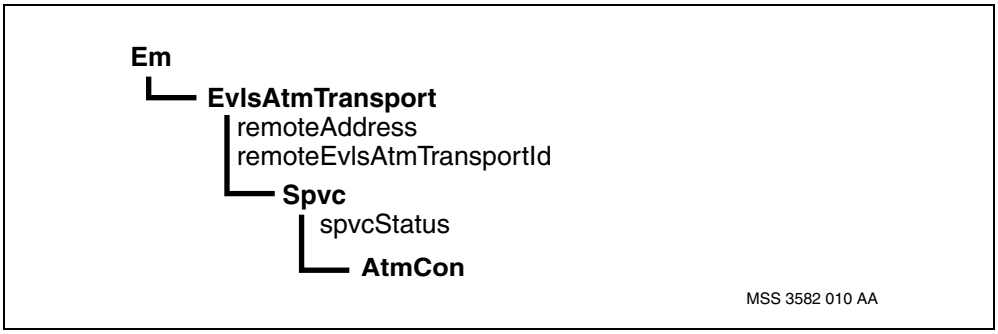
--End--

Variable definitions

Variable	Value
<spvc>	is the instance value of the *Spvc* component.
<transport>	is the instance value of the EVLS ATM transport.

Procedure job aid

ETS end-to-end EVLS ATM transport connectivity component hierarchy



Verifying ETS single-ended EVLS ATM transport connectivity

Verify that a single-ended ETS-to-ATM UNI is operational.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Verify that the status of the EVLS ATM transport SPVC is connected. **`display Ets EvlsAtmTransport/<transport> Spvc/<spvc> spvcStatus`**
2	Verify that the *AtmConnection* subcomponent exists under each of the configured EvlsAtmTransport SPVCs. **`list Ets EvlsAtmTransport/<transport> Spvc/<spvc>`**
3	Determine which ATM connection is connected to the SPVC. **`display Ets EvlsAtmTransport/<transport> Spvc/<spvc> AtmConnection`**
4	Verify ATM network connectivity. Ensure that the ATM UNI VCC is operational. See NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
5	If you are using a single-ended EVLS ATM transport in a performance enhancement configuration, verify that the traffic flows across the ATM SPVC connecting the two UNIs.
6	Repeat step 1 to step 4 to verify the connectivity of the far-end single-ended EVLS ATM transport used in a performance enhancement configuration.

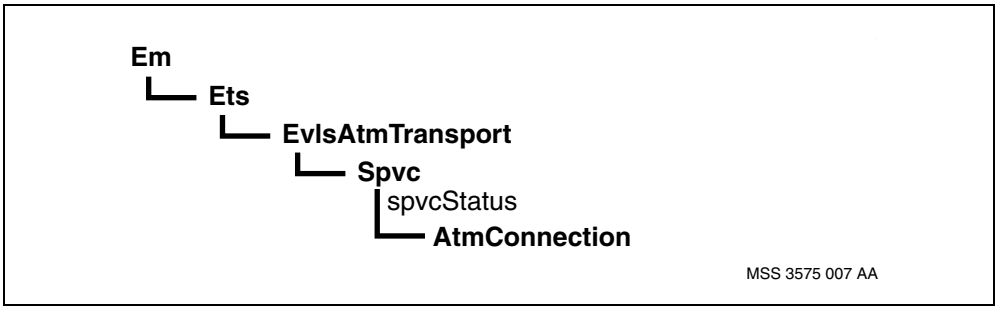
--End--

Variable definitions

Variable	Value
<spvc>	is the instance value of the *Spvc* component.
<transport>	is the instance value of the EVLS ATM transport.

Procedure job aid

ETS single-ended EVLS ATM transport component hierarchy



Verifying traffic flow over a LAN-to-LAN EVC

Compare the received frames of the local EVC against the transmitted frames of the far-end EVC. After the traffic has stopped, ensure the value of the *txFrames* and the *rxFrames* attributes are both increasing at both ends of the EVC.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.
- Verify that the LAN-to-LAN is connected. See Verifying LAN-to-LAN EVC connectivity (page 75).

Procedure steps

Step	Action
1	Choose one of the two end-to-end EVCs and verify the received frames from the local EVC statistics. **`display La/<la> Stats rxFrames`**
2	Verify transmission frames from the far-end EVC statistics. **`display La/<la> Stats txFrames`**
3	Ensure that the value of the *rxFrames* attribute of the local EVC displayed in step 1 and the value of the *txFrames* attribute of the far-end EVC displayed in step 2 are both increasing.
4	Repeat step 1 through step 3 for the far-end EVC.

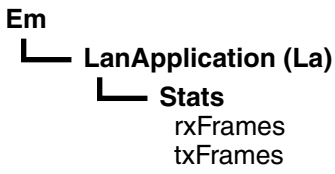
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.

Procedure job aid

Traffic flow over a LAN-to-LAN EVC component hierarchy



Verifying traffic flow over a LAN-to-VLAN EVC

Compare the received frames of the local EVC against the transmitted frames of the far-end EVC. After the traffic has stopped, ensure the value of the *txFrames* and the *rxFrames* attributes are both increasing at both ends of the EVC.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.
- Verify that the EVC is connected. See Verifying LAN-to-VLAN EVC connectivity (page 77).

Procedure steps

Step	Action
1	Verify the number of frames received from the local Ethernet interface. **`display La/<la> Stats rxFrames`**
2	Verify the number of frames transmitted to the far-end Ethernet interface. **`display La/<la> Vlan/<vlan> Stats txFrames`**
3	Ensure that the value of the *rxFrames* attribute of the local EVC displayed in step 1 and the value of the *txFrames* attribute of the far-end EVC displayed in step 2 are both increasing.
4	Verify the number of frames received from the far-end Ethernet interface. **`display La/<la> Vlan/<vlan> Stats rxFrames`**
5	Verify the number of frames transmitted to the local Ethernet interface. **`display La/<la> Stats txFrames`**
6	Ensure that the value of the *rxFrames* attribute of the far-end EVC displayed in step 4 and the value of the *txFrames* attribute of the local EVC displayed in step 5 are both increasing.

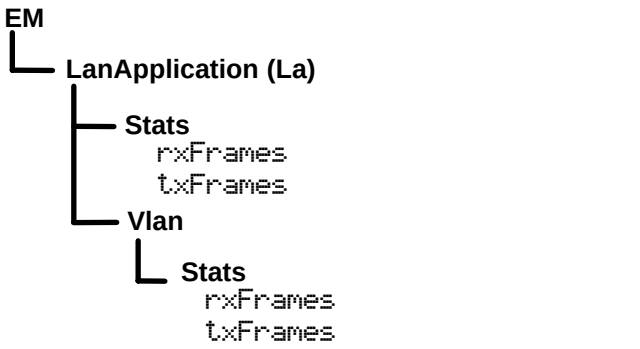
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Traffic flow over a LAN-to-VLAN EVC component hierarchy



Verifying traffic flow over a VLAN-to-LAN EVC

Compare the received frames of the local EVC against the transmitted frames of the far-end EVC. After the traffic has stopped, ensure the value of the *txFrames* and the *rxFrames* attributes are both increasing at both ends of the EVC.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.
- Verify that the EVC is connected. See Verifying VLAN-to-LAN EVC connectivity (page 79).

Procedure steps

Step	Action
1	Verify the number of frames received from the local Ethernet interface. **`display La/<la> Vlan/<vlan> Stats rxFrames`**
2	Verify the number of frames transmitted to the far-end Ethernet interface. **`display La/<la> Stats txFrames`**
3	Ensure that the value of the *rxFrames* attribute of the local EVC displayed in step 1 and the value of the *txFrames* of the far-end EVC displayed in step 2 are both increasing.
4	Verify the number of frames received from the far-end Ethernet interface. **`display La/<la> Stats rxFrames`**
5	Verify the number of frames transmitted to the local Ethernet interface. **`display La/<la> Vlan/<vlan> Stats txFrames`**
6	Ensure that the value of the *rxFrames* of the far-end EVC displayed in step 4 and the value of the *txFrames* of the local EVC displayed in step 5 are both increasing.

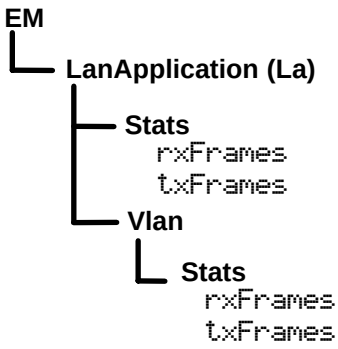
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Traffic flow over a VLAN-to-LAN EVC component hierarchy



MSS 4001 005 AA

Verifying traffic flow over a VLAN-to-VLAN EVC

Compare the received frames of the local EVC against the transmitted frames of the far-end EVC. After the traffic has stopped, ensure the value of the *txFrames* and the *rxFrames* attributes are both increasing at both ends of the EVC.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.
- Verify that the 1:1 VLAN-to-VLAN EVC or N:1 VLAN-to-VLAN EVC is connected. See Verifying 1:1 VLAN-to-VLAN EVC connectivity (page 81) or Verifying N:1 VLAN-to-VLAN EVC connectivity (page 83).

Procedure steps

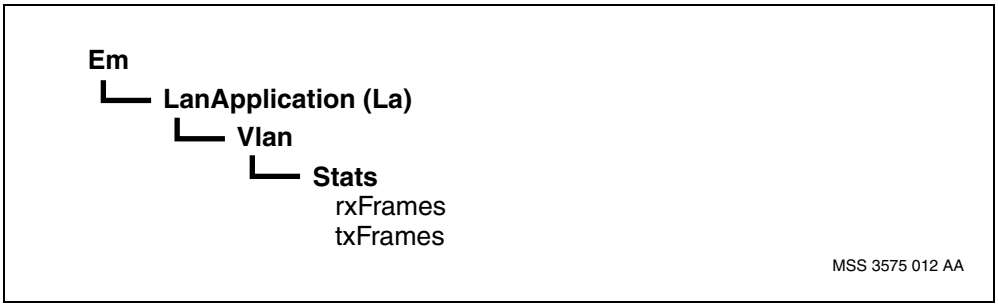
Step	Action
1	Choose one of the two end-to-end EVCs and verify the rxFrames of the local EVC statistics. **`display La/<la> Vlan/<vlan> Stats rxFrames`**
2	Verify txFrames of the far-end EVC statistics. **`display La/<la> Vlan/<vlan> Stats txFrames`**
3	Ensure that the rxFrames of the local EVC displayed in step 1 and the txFrames of the far-end EVC displayed in step 2 are both increasing.
4	Repeat step 1 through step 3 for the far-end EVC.
	--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Traffic flow over a VLAN-to-VLAN EVC component hierarchy



Verifying traffic flow over a LAN-to-ATM EVC

Compare the rxFrames of the local EVC against the txFrames of the far-end EVC. After the traffic has stopped, convert the received frames to cells and make sure the rxFrames correspond to the txCell.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.
- Verify that the single-ended EVC for LAN-to-ATM UNI VCC connectivity is connected. See Verifying LAN-to-ATM EVC connectivity (page 85).

Procedure steps

Step	Action
1	Determine the rxFrames of the EVC statistics. **`display La/<la> Stats rxFrames`**
2	Determine which ATM connection is connected to the SPVC. **`display La/<la> Evc Spvc/<spvc> AtmCon`**
3	Determine the txCell of the far-end ATM UNI VCC statistics. **`display AtmIf/<atmif> Vcc/<vcc> Stats txCell`**
4	Calculate the frames into their respective number of cells. Ensure the rxFrames value corresponds to the txCell value.
5	If you are using a single-ended EVC in a performance enhancement configuration, verify that the traffic flows across the ATM SPVC connecting the two UNIs.

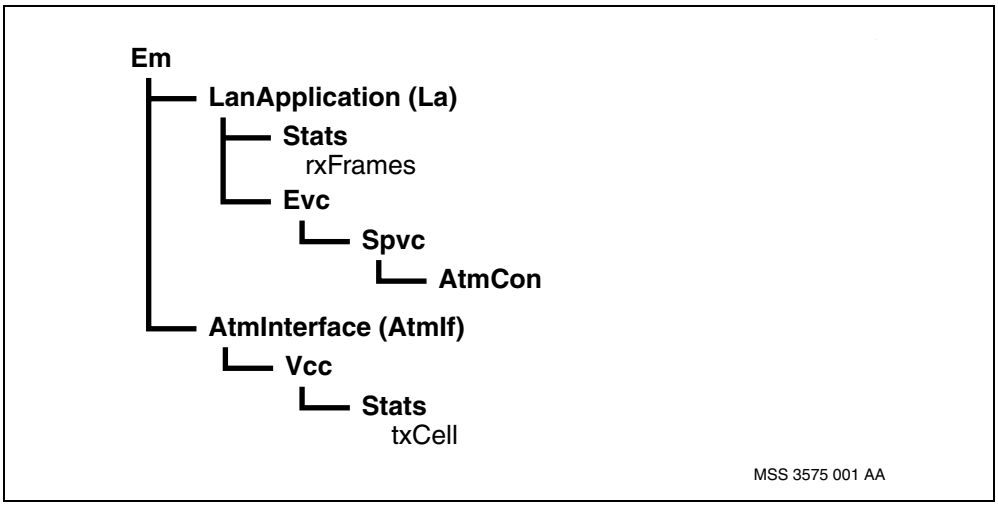
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vcc>	is the instance value of the *Vcc* component.

Procedure job aid

Traffic flow over a LAN-to-ATM EVC component hierarchy



Verifying traffic flow over a VLAN-to-ATM EVC

Compare the rxFrames of the local EVC against the txFrames of the far-end EVC. After the traffic has stopped, convert the received frames to cells and make sure the rxFrames correspond to the txCell.

Prerequisites

- Verify that the Ethernet port is physically connected to the CE device and is operational.

Procedure steps

Step	Action
1	Determine the rxFrames of the EVC statistics. **`display La/<la> Vlan/<vlan> Stats rxFrames`**
2	Determine which ATM connection is connected to the SPVC. **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> AtmCon`**
3	Determine the txCell of the far-end ATM UNI VCC statistics. **`display AtmIf/<atmif> Vcc/<vcc> Stats txCell`**
4	Calculate the frames into their respective number of cells. Ensure the rxFrames value corresponds to the txCell value.
5	If you are using a single-ended EVC in a performance enhancement configuration, verify that the traffic flows across the ATM SPVC connecting the two UNIs.

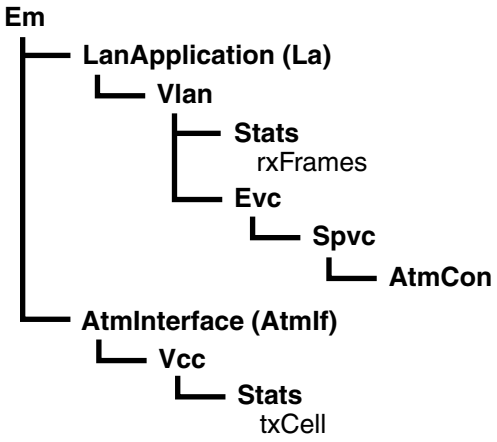
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vcc>	is the instance value of the *Vcc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Traffic flow over a VLAN-to-ATM EVC component hierarchy



Verifying policer configuration

This procedure displays the actual policer parameters used by Multiservice Switch node for traffic enforcement. The node rounds up the configured policer parameters to the nearest value supported by hardware. This procedure may be performed for both the called and calling ends. The displayed values should be close to the parameters configured by the user.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Display the EvcPolicer attributes. **display La/<la> Evc EvcPolicer ecir** **display La/<la> Evc EvcPolicer ecbs** **display La/<la> Evc EvcPolicer eeir** **display La/<la> Evc EvcPolicer eebs** or **display La/<la> Vlan/<vlan> Evc EvcPolicer ecir** **display La/<la> Vlan/<vlan> Evc EvcPolicer ecbs** **display La/<la> Vlan/<vlan> Evc EvcPolicer eeir** **display La/<la> Vlan/<vlan> Evc EvcPolicer eebs**

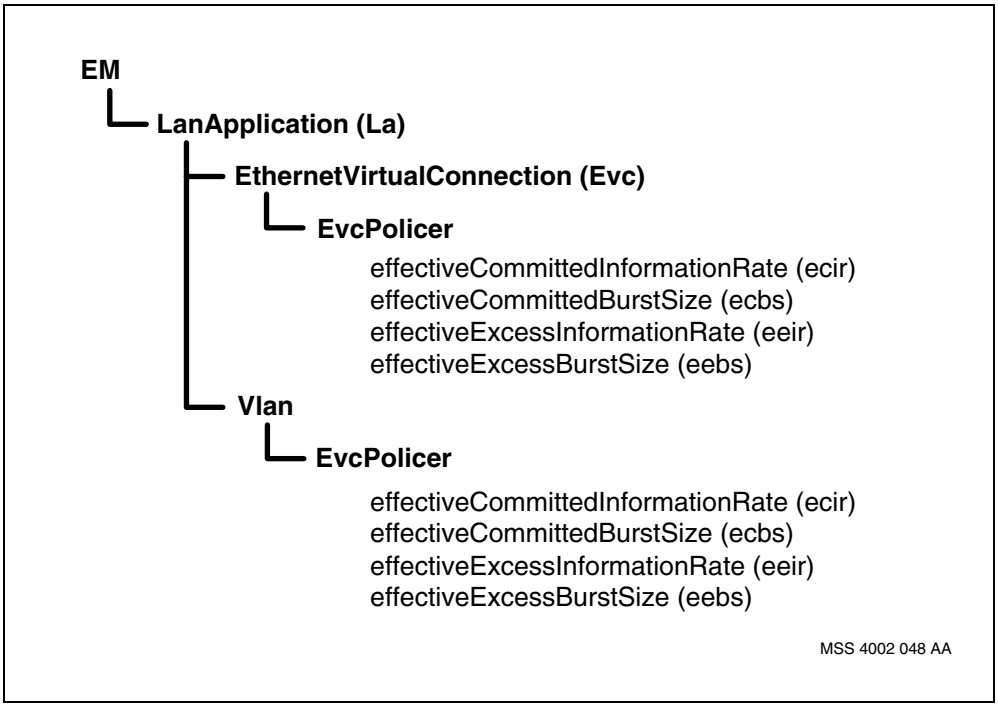
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Policer configuration component hierarchy



Verifying policer operation

Display statistics and confirm that one or both values are incrementing when traffic is received by the policer.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Verify received traffic by the network. **display La/<la> Stats rxFrames** or **display La/<la> Vlan/<vlan> Stats rxFrames** Verify that the traffic is incrementing.
2	If you verified that the traffic is incrementing in step 1, then display the *cirConformantFrame* attribute count. **display La/<la> Evc EvcPolicer cirComformantFrames** or **display La/<la> Vlan/<vlan> Evc EvcPolicer cirComformantFrames**
3	If you verified that the traffic is incrementing in step 1, then display the *eirConformantFrame* attribute count. **display La/<la> Evc EvcPolicer eirComformantFrames** or **display La/<la> Vlan/<vlan> Evc EvcPolicer eirComformantFrames**

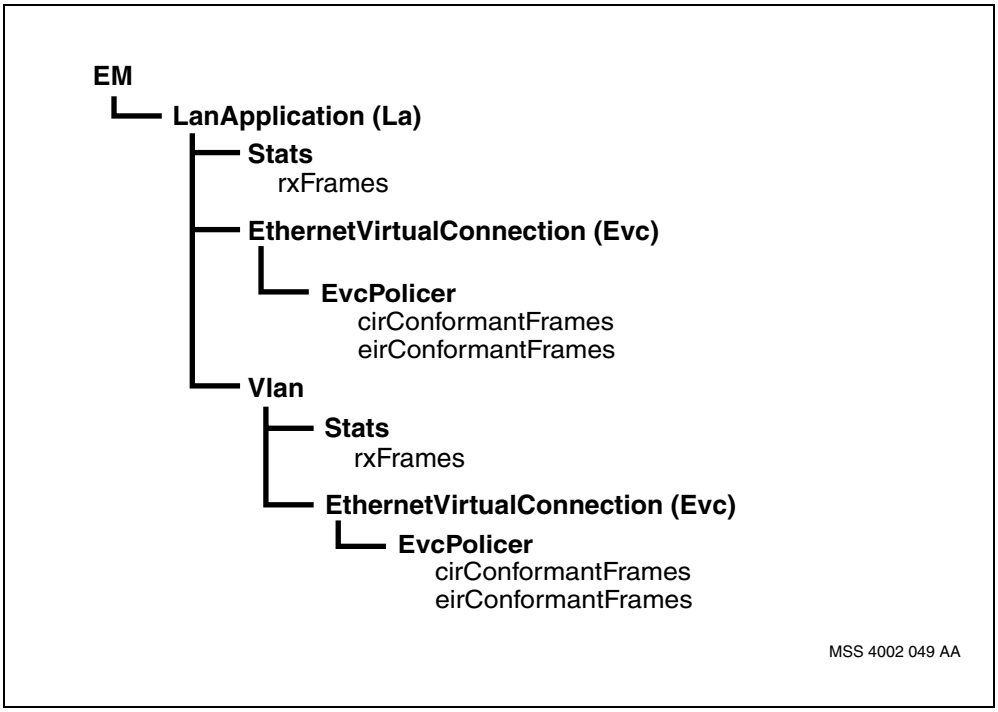
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Policer operation component hierarchy



Isolating EVLS problems

Since a LAN connection spans both a LAN and an ATM network, you need to isolate the problem to the specific portion of the network. The generic troubleshooting guidelines are described below:

Procedure steps

Step	Action
1	Analyze alarms generated by Nortel Multiservice Switch components. See NN10600-500 *Nortel Multiservice Switch 6400/7400/15000/20000 Alarms Reference*.
2	Display and analyze LAN service level operational statistics. See NN10600-702 *Nortel Multiservice Switch 7400/15000/20000 ATM Routing and Signalling Fundamentals*.
3	Identify corrective actions by referring to NN10600-700 *Nortel Multiservice Switch 7400/15000/20000 ATM Technology Fundamentals*.
4	Use the test and loop capability for an ATM virtual channel connection (VCC), if third party equipment is being used. See NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
5	Identify the ATM related cause codes by referring to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

--End--

Diagnosing EVC connectivity faults

When traffic loss is occurring, check the EVC status first. For non-aggregating EVC, the SPVC is automatically created at the called end, in which case the SPVC information cannot be used to diagnose EVC connectivity faults. The calling end is the only source of information in this case.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Verify that an *SPVC* subcomponent exists under the EVC. **`list La/<la> Evc Spvc/<spvc>`** **`list La/<la> Vlan/<vlan> Evc Spvc/<spvc>`**
2	Determine whether the endpoint of the connection is the calling end. **`display La/<la> Evc Spvc/<spvc> endPoint`** **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> endPoint`** If this endpoint is determined not to be the calling end, repeat step 1 and step 2 on the other Multiservice Switch node and proceed with the remaining steps.
3	Check the SPVC status. **`display La/<la> Evc Spvc/<spvc> spvcStatus`** **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> spvcStatus`**
4	If the spvcStatus is serviceNotReady, check the administrative control for all components used to establish the EVC. Ensure all components used to establish the EVC are unlocked. If the spvcStatus is setupFailed, there are two possible causes. One is caused by an ATM network problem. See NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*. Another possible cause is that the Ethernet interface at the far-end is inactive.
5	Determine the lastFailureCause. **`display La/<la> Evc Spvc/<spvc> lastFailureCause`** **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> lastFailureCause`**
6	See Handling problems (page 123) for a list of possible solutions.

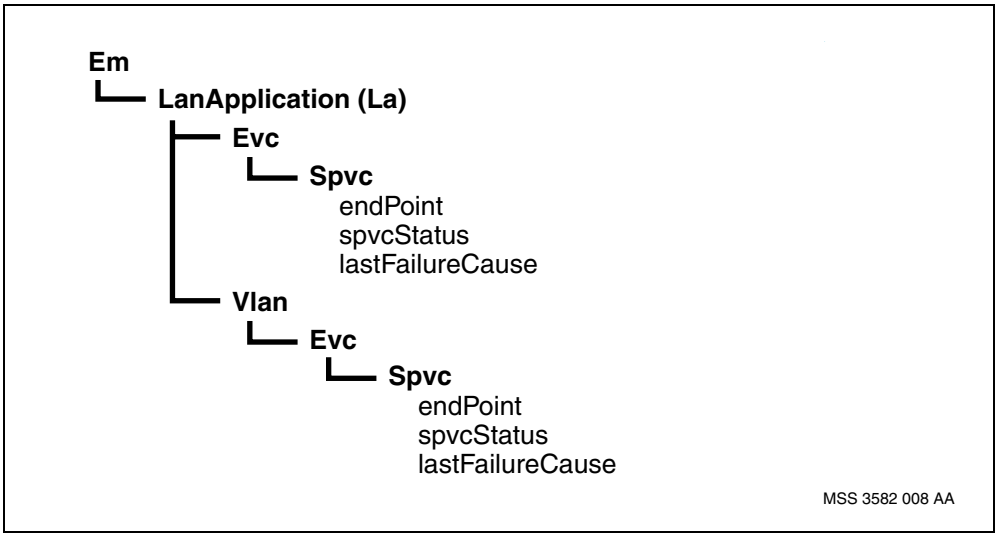
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

EVC connectivity faults component hierarchy



Diagnosing traffic loss related to congestion

Determine whether traffic loss is due to link or network congestion. Use statistics on the Ethernet interface and ATM network to detect traffic loss.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Verify the LanApplication statistics for discards in the direction from the Ethernet interface. **`display La/<la> Stats rxDiscFrames, txDiscFrames`** **`display La/<la> Vlan/<vlan> Stats rxDiscFrames, txDiscFrames`** If the local Ethernet interface is congested, it can be diagnosed that FP congestion causes Ethernet interface queue congestion resulting in potential traffic loss.
2	Determine the number of frames received by the local EVC. **`display La/<la> Stats rxFrames`** **`display La/<la> Vlan/<vlan> Stats rxFrames`**
3	Determine the number of frames transmitted by the far-end EVC. **`display La/<la> Stats txFrames`** **`display La/<la> Vlan/<vlan> Stats txFrames`**
4	Compare the rxFrames of the local EVC against the txFrames of the far-end EVC. After the traffic is stopped, if the txFrames is less than rxFrames, traffic loss may be due to ATM network congestion.
5	If traffic loss is determined to be a result of congestion, see Isolating the point of congestion (page 118).

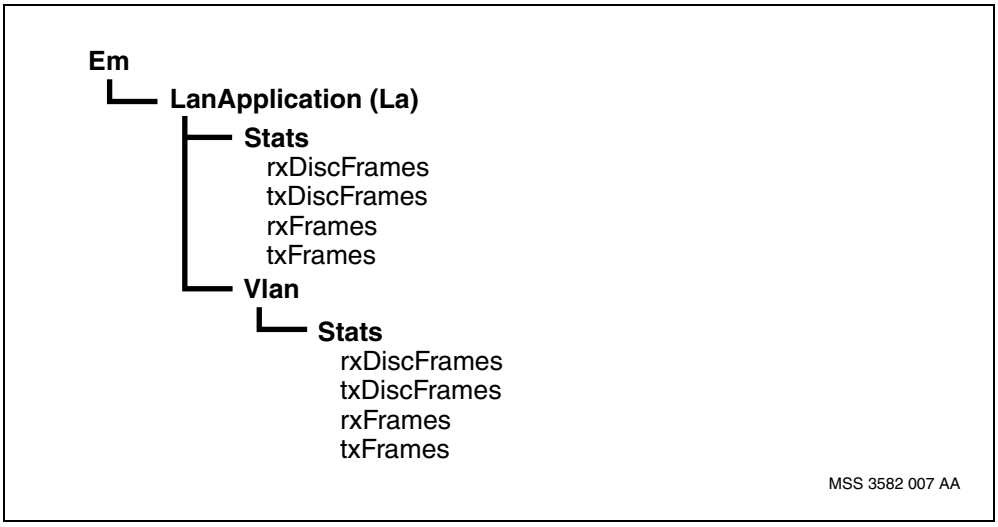
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Traffic loss related to congestion component hierarchy



Isolating the point of congestion

You can isolate the point of congestion when data loss is observed on an EVC connection. Use statistics on the Ethernet interface and ATM network to detect traffic loss.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Prerequisites

- You must determine whether traffic loss is due to congestion. See Diagnosing traffic loss related to congestion (page 116).

Procedure steps

Step	Action
1	Determine to which ATM next hop this SPVC is established. **`display La/<la> Evc Spvc/<spvc> AtmCon nextHop`** **`display La/<la> Vlan/<vlan> Evc Spvc/<spvc> AtmCon nextHop`**
2	Diagnose ATM network traffic loss starting at the ATM next hop determined above by referring to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

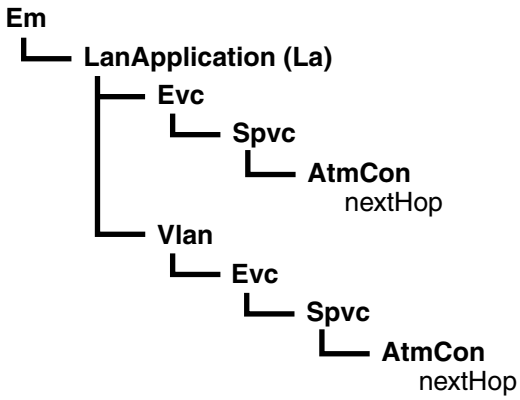
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<spvc>	is the instance value of the *Spvc* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Isolating the point of congestion component hierarchy



Diagnosing traffic loss unrelated to congestion

There are a number of possible causes for traffic loss when congestion has been eliminated. Some possible causes are that the *La* or *Vlan* component is not operational, the SPVC connection is down, the maximum frame size on the Ethernet port is misconfigured, the *unknownVlanIdTreatment* attribute is misconfigured, or the policer is discarding frames.

Procedure steps

Step	Action
1	For an end-to-end configuration, determine whether the *La* or *Vlan* component is locked. If it is locked, the *rxDiscFrames* attribute value increments. **display La/<la> Stats rxFrames, rxDiscFrames** **display La/<la> Vlan/<vlan> Stats rxFrames, rxDiscFrames**
2	Determine whether the traffic is being sent to the other end by performing this command on the far-end Ethernet interface. **display La/<la> Stats rxFrames, txFrames** **display La/<la> Vlan/<vlan> Stats rxFrames, txFrames**
3	Determine whether connectivity fault is due to an inactive SPVC under the *La* or *Vlan* component. **display La/<la> Evc vcDownDiscards** **display La/<la> Vlan/<vlan> Evc vcDownDiscards**
4	Determine whether the frames are too long. **display Lp/<lp> Ethernet/<ethernet> framesTooLong**
5	Determine that the value of the *averageFrameSize (afs)* attribute reflects the traffic profile received by the EVC. **display La/<la> Evc EQoS/<EQoS> afs <afs>** **display La/<la> Vlan/<vlan> Evc EQoS/<EQoS> afs <afs>**
6	Determine whether the Ethernet policy is influencing the traffic loss. **display La/<la> Stats EthIfStatsOper**
7	Determine whether the EVLS policer is influencing traffic loss. **display La/<la> Evc EvcPolicer discardedNonConformantFrames** **display La/<la> Vlan/<vlan> Evc EvcPolicer discardedNonConformantFrames**

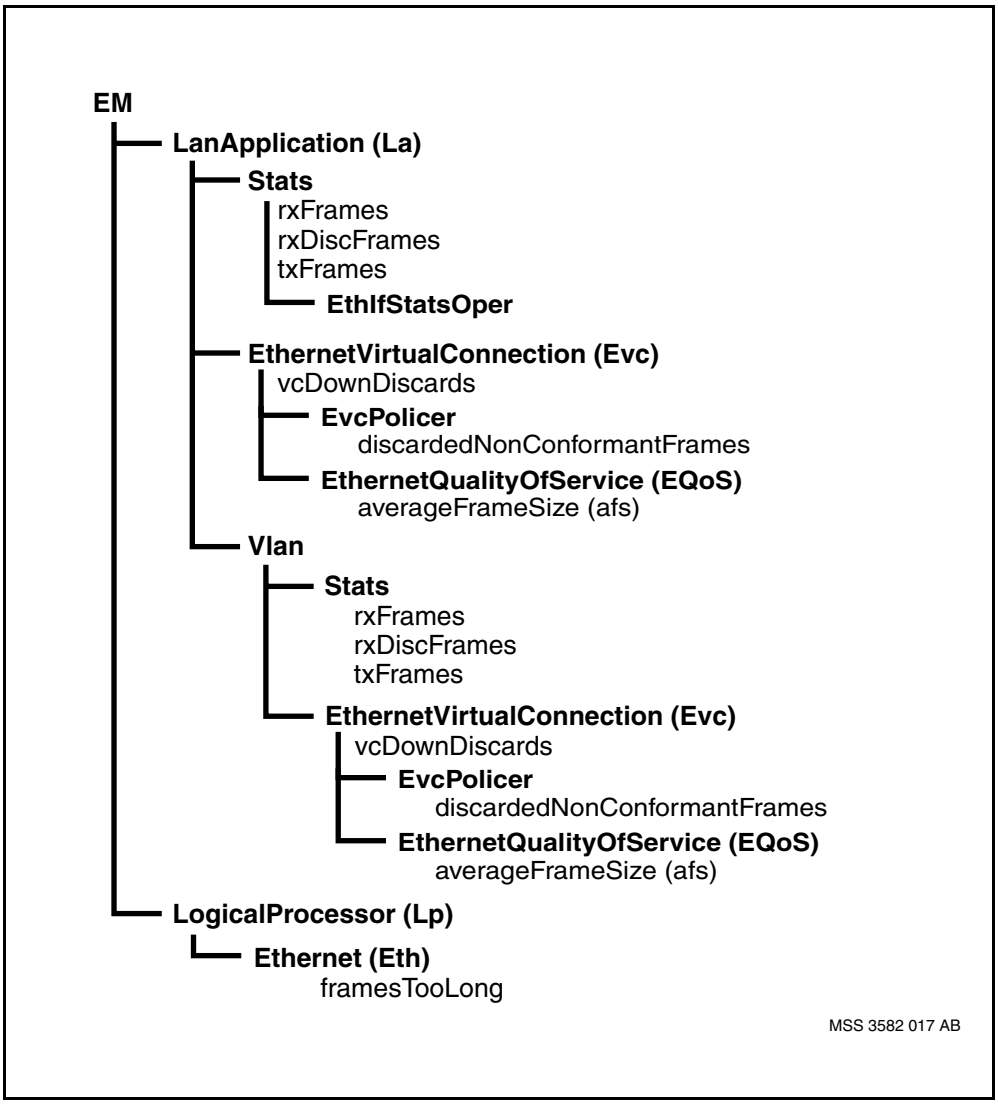
--End--

Variable definitions

Variable	Value
<afs>	is the average frame size that needs to be synchronized with the far-end EVC value. Default value is 512 bytes.
<la>	is the instance value of the *LanApplication* component.
<lp>	is the instance value of the *LogicalProcessor* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Diagnosing traffic loss unrelated to congestion component hierarchy



Handling problems

The table Handling problems (page 123) provides guidelines on how to respond to problems that may occur with EVC connections.

Handling problems

Problem	Probable causes	Corrective measures
Ethernet port physically connected to the CE device is not operational	There is an operational affecting condition.yip	Ensure that the Ethernet port is unlocked. Refer to NN10600-520 *Nortel Multiservice Switch 7400/15000/20000 Fault and Performance Management: Troubleshooting*.
	Faulty cable.	Replace the cable. Refer to NN10600-175 *Nortel Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade* or NN10600-130 *Nortel Multiservice Switch 15000/20000 Hardware Installation, Maintenance, and Upgrade*.
	Port configuration does not match CE port configuration.	Check the autonegotiation settings at both ends. Refer to Checking the status of FP Ethernet ports in NN10600-520 *Nortel Multiservice Switch 7400/15000/20000 Fault and Performance Management: Troubleshooting*.
La Evc or Vlan Evc can not establish connectivity	Ethernet port, *La*, or *Vlan* component is locked.	Ensure that all components used to establish the EVC are unlocked.
	Under *Evc Spvc* component: lastFailureCause = 3 (No route to destination)	Ensure that the called-end Ethernet interface is configured. Ensure that there is a path between the Ethernet interfaces. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/ 20000 ATM Fault and Performance Management*. Ensure that the calling-end EVC is configured with the correct NSAP and remoteVlanId of the called-end EVC.
	Under *Evc Spvc* component: lastFailureCause = 17 (User is busy)	Ensure that the called end has not accepted another call. Ensure that there is no failure in the ATM network. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
(1 of 5)		

Handling problems (continued)

Problem	Probable causes	Corrective measures
La Evc or Vlan Evc can not establish connectivity (continued)	Under *Evc Spvc* component: lastFailureCause = 21 (Call is rejected)	Ensure that the call is made to a called end instead of a calling end. Ensure that the called end is an EVLS. Ensure that there is no failure in the ATM network. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Evc Spvc* component: lastFailureCause = 45 (No vpi.vci is available)	Ensure that the called vpi.vci value is available under the ATM interface. Ensure that there is no failure in the ATM network. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Evc Spvc* component: lastFailureCause = 37 (Cell rate is unavailable)	Ensure that the parameter configured under *EQoS* component can be accepted in the tandem ATM card. Provide additional resource in the ATM network. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Evc Spvc* component: lastFailureCause = 49 (QoS is unavailable)	
	Other cause codes.	Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*
La Evc or Vlan Evc connectivity failure	Under *Evc Spvc* component: lastFailureCause = 18 (No user response)	Ensure that the called-end Ethernet interface is configured.
	Under *Evc Spvc* component: lastFailureCause = 21 (call is rejected)	Ensure that all components used to establish the EVC are unlocked.
	Other cause codes.	Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*
ETS EvlsAtmTransport can not establish connectivity	*EvlsAtmTransport* component is locked.	Ensure that the *EvlsAtmTransport* component is unlocked.

Handling problems (continued)

Problem	Probable causes	Corrective measures
ETS EvlsAtmTransport can not establish connectivity (continued)	Under *Spvc* component: LastFailureCauseCode = 3 (no route to destination)	Ensure that the called-end Ets is configured. Ensure that the calling-end Ets EvlsAtmTrsp is configured with the correct NSAP and remoteEvlsAtmTrspId of the called-end Ets. Ensure that there is a path between the Ets endpoints. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Spvc* component: LastFailureCauseCode = 17 (user is busy)	Ensure that the called end has not accepted the call. Ensure that there is no failure in the ATM network. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Spvc* component: LastFailureCauseCode = 21 (call is rejected)	Ensure that the call is made to a called end instead of a calling end. Ensure that there is no failure in the ATM network. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Spvc* component: LastFailureCauseCode = 34 (requested vpi/vci is already being used)	Change the *calledvpivci* attribute under the *SrcVcc* component in the single-ended configuration. Ensure that the requested vpivci is available.
	Under *Spvc* component: LastFailureCauseCode = 37 (cell rate is unavailable)	Ensure that the parameter configured under *Tm* component can be accepted on the tandem ATM card. Provide additional resource in ATM. Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
	Under *Spvc* component: LastFailureCauseCode = 49 (QoS is available)	
	Under *Spvc* component: LastFailureCauseCode = 45 (No vpi.vci is available)	Ensure that the called vpi.vci value is available under the ATM interface.

Handling problems (continued)

Problem	Probable causes	Corrective measures
ETS EvlsAtmTransport can not establish connectivity (continued)	No EVC registered to this *EvlsAtmTransport* component.	Ensure that there is at least one aggregated EVC that is registered to this *EvlsAtmTransport* component.
	Other cause codes.	Refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.
EVC using aggregation can not establish connectivity	Aggregation is not enabled in the access EVC.	Set the *EVC aggregation* attribute to enabled.
	aggregationLastFailureCause = noNsapPrefixMatch	Configure an EvlsAtmTransport with a prefixNsapAddress that matches the EVC remoteAddress. Ensure that the *La* component for this EVC is configured in the *EtsEAtmTrsp lanApplication* attribute. Ensure that the EVC remoteAddress is configured with the activeNsapAddress of the far-end Ethernet interface.
	aggregationLastFailureCause = etsNotReady	This is a transient condition. Wait for Ets to become ready.
	aggregationLastFailureCause = spvcNotAvail	Ensure that the Ets EvlsAtmTransport Spvc is operational.
Traffic is discarded due to congestion	Ingress Ethernet interface congestion.	Reduce the amount of traffic sent to network from CE. Increase the network capacity to support more offered load.
	Egress Ethernet interface congestion.	Re-engineering the network to reduce traffic to the port. Reduce the number of VLANs by moving some VLANs to another port or FP.
	ATM network congestion.	Check for traffic loss in the ATM portion of the network. For details, refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/ 20000 ATM Fault and Performance Management*.
Traffic is discarded due to non-congestion conditions	*La* or *Vlan* component is not operational.	Ensure that all components used to establish the EVC are unlocked.

(4 of 5)

Handling problems (continued)

Problem	Probable causes	Corrective measures
Traffic is discarded due to non-congestion conditions (continued)	EVC SPVC is not connected.	Follow the fault recovery procedures for EVC connectivity failures. Refer to end-to-end and single-ended troubleshooting procedures within this chapter.
	Traffic is discarded by the port.	Refer to NN10600-520 *Nortel Multiservice Switch 7400/15000/20000 Fault and Performance Management: Troubleshooting*.
	Ethernet policy to treat unknown VLAN traffic is configured to discard.	Ensure that the correct VLAN is configured on the Ethernet interface. Ensure that the *unknownVlanIdTreatment* attribute value is set to accept and an EVC is configured under the *La* component.
(5 of 5)		

Diagnosing no traffic enforcement

The policer does not cause any traffic tagging or discards. This behavior may be observed by noting all received CIR frames are accepted as conformant traffic. Traffic enforcement may not take place because of several reasons: the *policer* component may not be added, configured properly, or enabled. The CIR may be set equal to the Ethernet access rate, in which case all traffic would be conformant.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Determine count of received frames by displaying *rxFrames*. **display La/<la> EthStats rxFrames** or **display La/<la> Vlan/<vlan> EthStats rxFrames**
2	Confirm the problem. Verify that all received frames are accepted as CIR conformant traffic. Display cirConformantFrames and compare it to the value in step 1. **display La/<la> Evc EvcPolicer cirConformantFrames** or **display La/<la> Vlan/<vlan> Evc EvcPolicer cirConformantFrames**
3	Diagnose the problem. Verify that the policer is added to the EVC. **list La/<la> Evc** or **list La/<la> Vlan/<vlan> Evc** If the policer is not added, see the section EVLS traffic management configuration (page 45) to configure it.
4	Verify that the policer is enabled for the EVC. **display La/<la> Evc EvcPolicer policing** or **display La/<la> Vlan/<vlan> Evc EvcPolicer policing** If the policer is not enabled, see the section EVLS traffic management configuration (page 45) to configure it.

5 Verify that CIR is set to Ethernet access rate. If the CIR value is the same as the lineSpeed value, configure a smaller value for CIR. Note that the available Ethernet access rate capacity is less than the port rate because of the exclusion of the inter-frame-gap, preamble, and SFD field (20 bytes per frame).

```
display La/<la> Evc EvcPolicer cir
```

or

```
display La/<la> Vlan/<vlan> Evc EvcPolicer cir
```

```
display Lp/<lp> Ethernet/<ethernet> lineSpeed
```

6 Verify that the rate of ingress traffic is less than the effective CIR value.

```
display La/<la> Evc EvcPolicer ecir
```

or

```
display La/<la> Vlan/<vlan> Evc EvcPolicer ecir
```

Using a protocol analyzer, in a line drop configuration, ensure that the amount of traffic arriving at the Multiservice Switch Ethernet interface is greater than the EVC policer's effective CIR value. If the rate of traffic is in excess of the configured CIR value on the Multiservice Switch node, contact Nortel Technical Support for assistance.

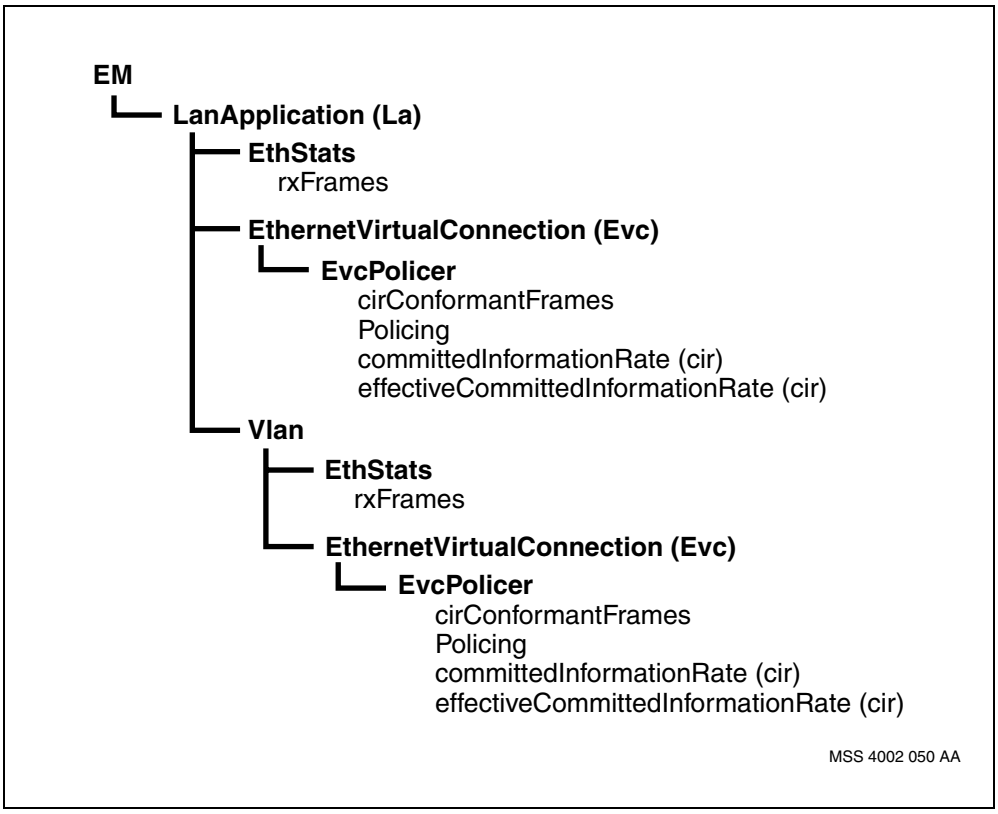
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<lp>	is the instance value of the *LogicalProcessor* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

No traffic enforcement component hierarchy



Diagnosing that all traffic is tagged by policer

The policer may admit and tag all incoming frames if CIR is zero and EIR is set to the Ethernet port line rate.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Determine count of received frames by displaying *rxFrames*. **display La/<la> EthStats rxFrames** or **display La/<la> Vlan/<vlan> EthStats rxFrames**
2	Confirm the problem. Verify that all received frames are accepted as EIR conformant traffic. Display eirConformantFrames and compare it to the value in step 1. **display La/<la> Evc EvcPolicer eirConformantFrames** or **display La/<la> Vlan/<vlan> Evc EvcPolicer eirConformantFrames**
3	Diagnose the problem. Verify that CIR and CBS are both non-zero. **display La/<la> Evc EvcPolicer ecir** **display La/<la> Evc EvcPolicer ecbs** or **display La/<la> Vlan/<vlan> Evc EvcPolicer ecir** **display La/<la> Vlan/<vlan> Evc EvcPolicer ecbs**
4	Verify that EIR is set to the Ethernet port access rate. Note that the available Ethernet access rate capacity is less than the port rate because of the exclusion of the inter-frame-gap, preamble, and SFD field (20 bytes per frame). **display Lp/<lp> Eth/<ethernet> actualLineSpeed** **display La/<la> Evc EvcPolicer eeir** or **display La/<la> Vlan/<vlan> Evc EvcPolicer eeir**

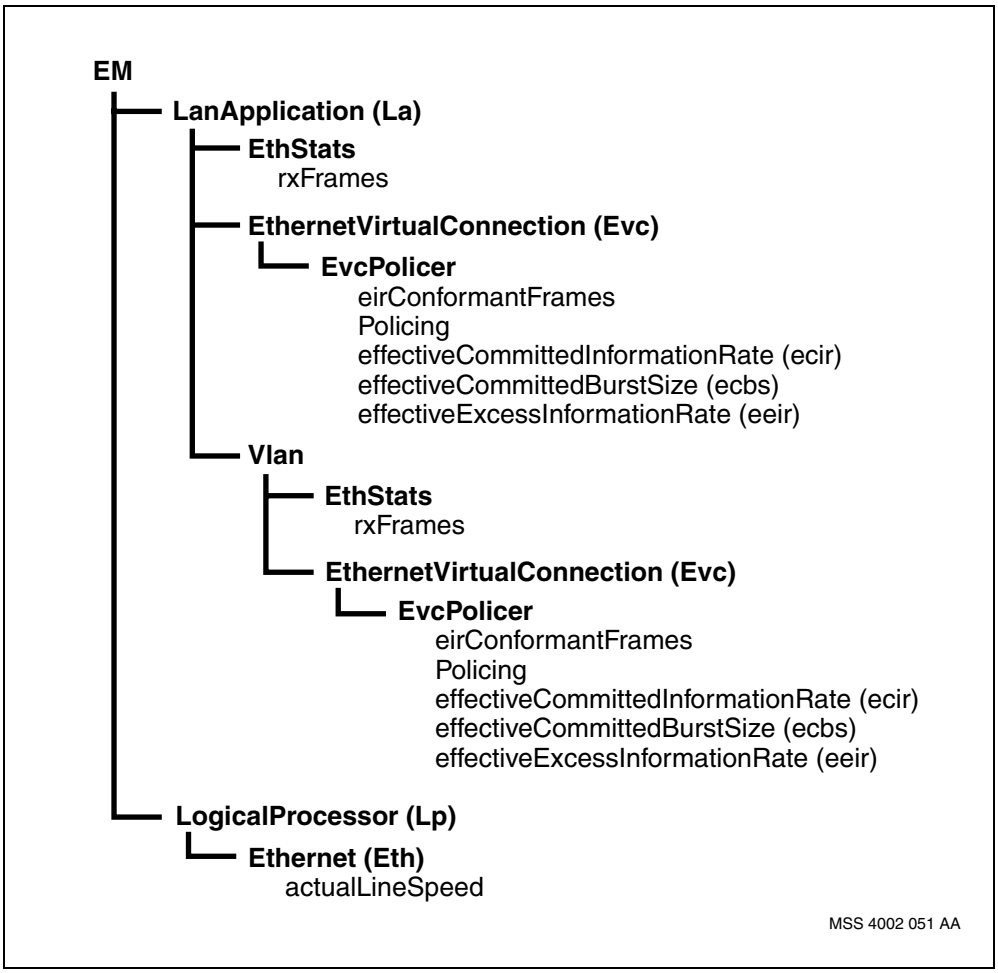
--End--

Variable definitions

Variable	Value
<ethernet>	is the instance value of the *Ethernet* component.
<la>	is the instance value of the *LanApplication* component.
<lp>	is the instance value of the *LogicalProcessor* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

All traffic is tagged by policer component hierarchy



Diagnosing that all traffic is discarded by policer

The policer discards all traffic if both CIR and EIR are set to zero.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Display rxFrames. **display La/<la> EthStats rxFrames** or **display La/<la> Vlan/<vlan> EthStats rxFrames**
2	Display the discardedNonConformantFrames and confirm that it is equal to the received rxFrames value in step 1. **display La/<la> Evc EvcPolicer discardedNonConformantFrames** or **display La/<la> Vlan/<vlan> Evc EvcPolicer discardedNonConformantFrames**
3	Verify that both CIR or CBS, as well as EIR or EBS are non-zero. **display La/<la> Evc EvcPolicer cir** **display La/<la> Evc EvcPolicer cbs** **display La/<la> Evc EvcPolicer eir** **display La/<la> Evc EvcPolicer ebs** or **display La/<la> Vlan/<vlan> Evc EvcPolicer cir** **display La/<la> Vlan/<vlan> Evc EvcPolicer cbs** **display La/<la> Vlan/<vlan> Evc EvcPolicer eir** **display La/<la> Vlan/<vlan> Evc EvcPolicer ebs**

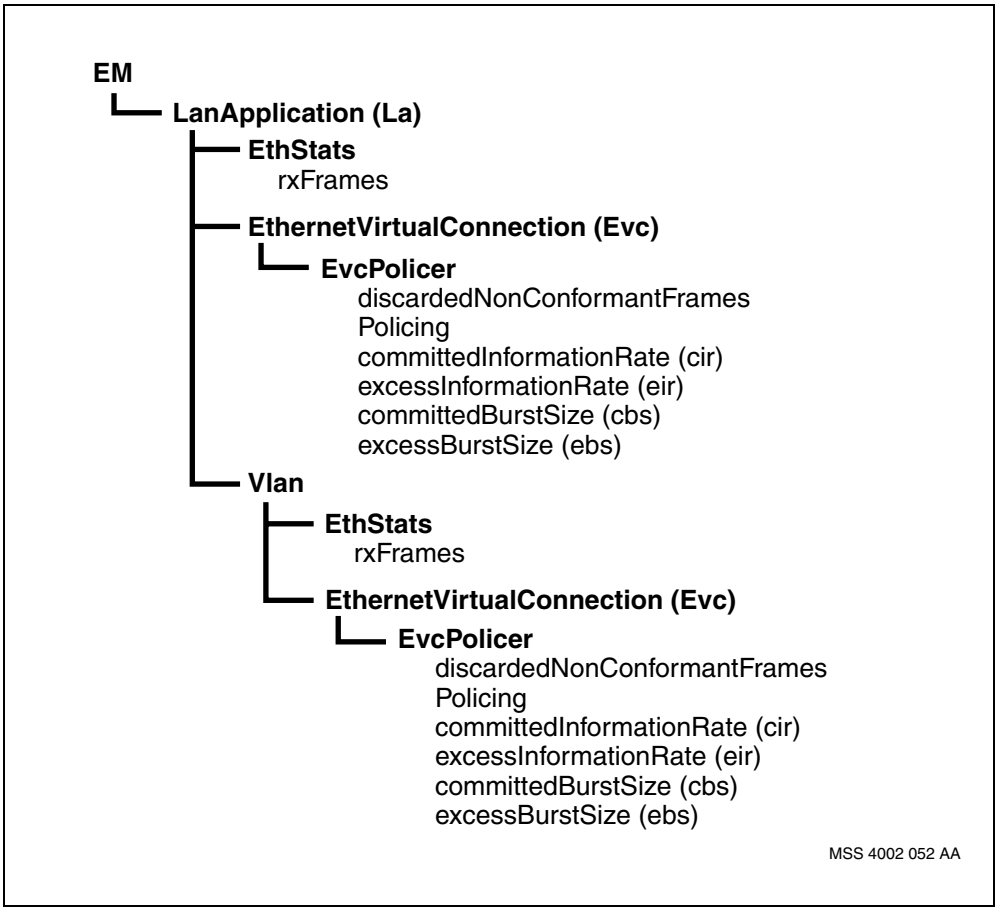
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

All traffic is discarded by policer component hierarchy



Diagnosing that the connection is not meeting service level agreement (SLA)

The user is experiencing larger than expected network traffic loss. The cause may be that the user traffic exceeds the contract, causing tagging or discarding traffic by the policer.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Display policer tagging statistics. **`display La/<la> Evc EvcPolicer eirConformantFrames`** or **`display La/<la> Vlan/<vlan> Evc EvcPolicer eirConformantFrames`** If the value is incrementing, it indicates that the user is sending traffic above CIR rate. Tagged traffic has no performance guarantee, and is discarded by the network before conformant traffic.
2	Display policer discard statistics. **`display La/<la> Evc EvcPolicer discardedNonConformantFrames`** or **`display La/<la> Vlan/<vlan> Evc EvcPolicer discardedNonConformantFrames`** If the value is incrementing, it indicates that the user is sending traffic above the CIR + EIR rate. This traffic is discarded by the policer at network ingress.

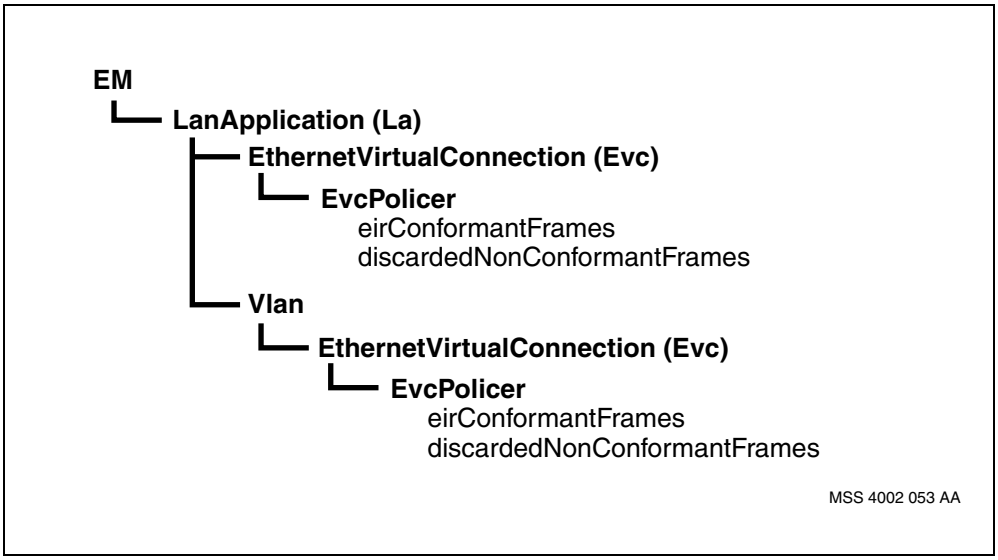
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Connection is not meeting SLA component hierarchy



Diagnosing policer admitting excess traffic as CIR conformant

The policer may admit traffic above the configured committed information rate and burst size. The reason is that the actual values used by the policer may be slightly larger that the configured values because of hardware granularity. See section Policing granularity (page 206) for more information.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Verify that the operational values of CIR and CBS are within the acceptable round granularity for the FP. **Display La/<la> Vlan/<vlan> Evc EvcPolicer effectiveCommittedInformationRate** **Display La/<la> Vlan/<vlan> Evc EvcPolicer effectiveCommittedBurstSize** or **Display La/<la> Evc EvcPolicer effectiveCommittedInformationRate** **Display La/<la> Evc EvcPolicer effectiveCommittedBurstSize**

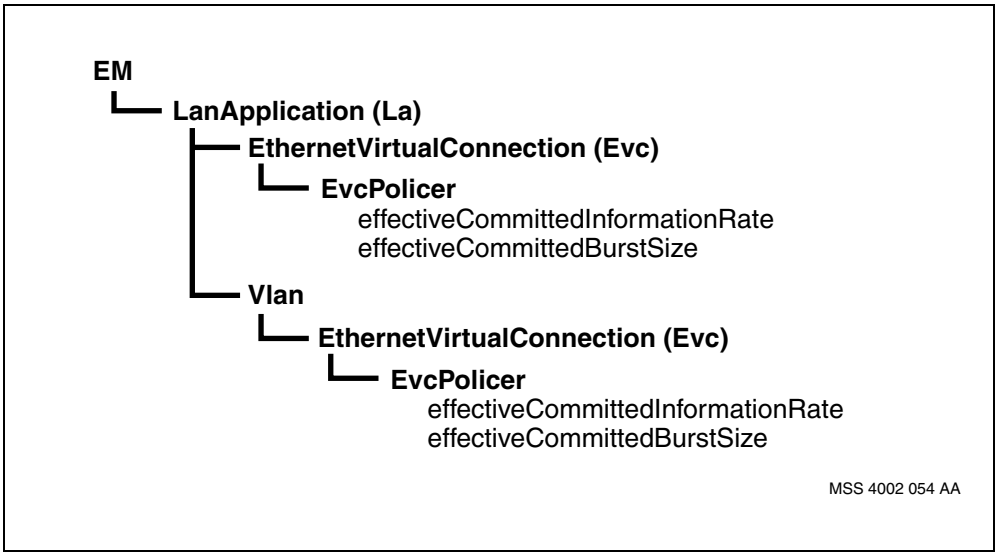
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Policer admitting excess traffic as CIR conformant component hierarchy



Diagnosing policer admitting non-conformant traffic as EIR conformant

The policer may admit traffic above the configured excess information rate and burst size. See section Policing granularity (page 206) for more information.

For an EVC under the *La* component, perform the commands for the *La* component. For an EVC under the *Vlan* component, perform the commands for the *Vlan* component.

Procedure steps

Step	Action
1	Verify that the operational values of EIR and EBS are within the acceptable round granularity for the FP. **Display La/<la> Vlan/<vlan> Evc EvcPolicer effectiveExcessInformationRate** **Display La/<la> Vlan/<vlan> Evc EvcPolicer effectiveExcessBurstSize** or **Display La/<la> Evc EvcPolicer effectiveExcessInformationRate** **Display La/<la> Evc EvcPolicer effectiveExcessBurstSize**

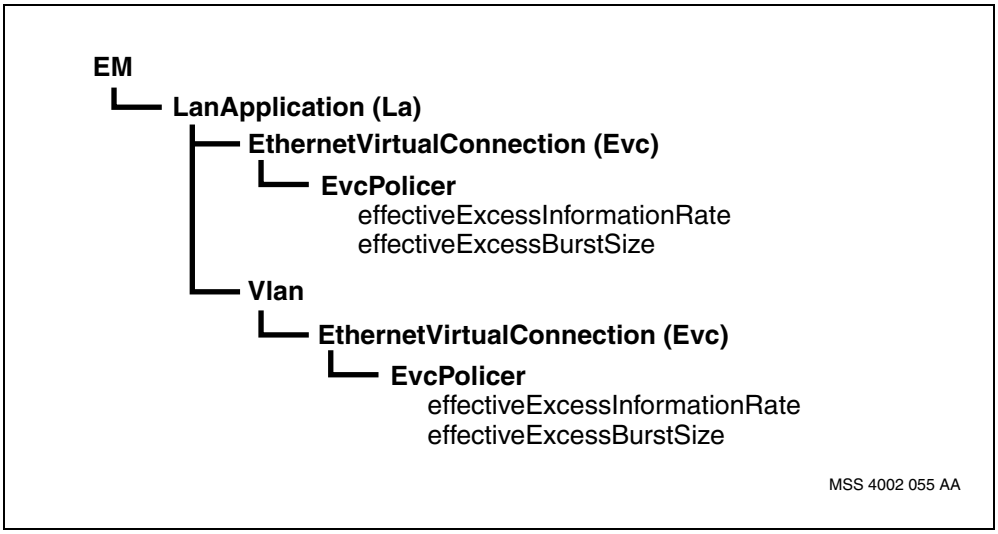
--End--

Variable definitions

Variable	Value
<la>	is the instance value of the *LanApplication* component.
<vlan>	is the instance value of the *Vlan* component.

Procedure job aid

Policer admitting non-conformant traffic as EIR conformant component hierarchy



EVLS overview

The Ethernet virtual line service (EVLS) provides transparent tunneling of both IEEE 802.3 and Ethernet v2.0 traffic over a dedicated Ethernet virtual connection (EVC). The EVC can be established between a pair of Ethernet LANs, or a pair of Ethernet virtual LANs (VLANs), or between an Ethernet LAN or a VLAN and a tandem ATM interface. Each Ethernet LAN or VLAN is logically associated with the EVC, such that either of the following connections can be established:

- LAN-to-LAN, LAN-to-VLAN, or VLAN-to-VLAN peer connectivity can be established using the end-to-end EVC configuration.
- LAN-to-ATM VCC or VLAN-to-ATM VCC connectivity can be established using the single-ended configuration.

Attention: In this section, statements related to LAN-to-VLAN also apply to VLAN-to-LAN.

See the following sections for more detailed conceptual information about EVLS:

- Why use EVLS? (page 141)
- Terminology (page 142)
- How does EVLS work? (page 143)
- Carrier grade functionality (page 147)
- Timers (page 148)
- Addressing (page 148)

Why use EVLS?

Ethernet has emerged as both an important service offer for implementing VPNs within a local Metro area and an access mechanism for higher layer services with local and wide area scope. However, it often becomes a challenge to extend the Ethernet VPN service over the WAN between pockets of Metro Ethernet, or delivering Ethernet-based end customers to layer 3 service points of presence outside of the local Metro. Ethernet virtual line

service (EVLS) provides the ability to extend LAN/MAN/WAN reachability via Ethernet across a resilient ATM WAN. ATM network infrastructure is leveraged to cost effectively transport Ethernet traffic between a pair of Ethernet LANs, between a pair of Ethernet VLANs, and between an Ethernet LAN and an Ethernet VLAN on Ethernet interfaces.

EVLS offers the following additional benefits:

- Enables customer devices to view the core network as a single point-to-point Ethernet wire.
- Provides multiple users on a single Ethernet interface connectivity to another Ethernet interface via LANs across an ATM network, enabling optimization of Ethernet resource utilization.
- Introduces the Ethernet transport system (ETS) to provide aggregation of traffic from multiple VLANs on a single port across a single ATM virtual channel connection (VCC).
- Provides flexibility to interoperate with third-party equipment at the remote access point of the ATM network by encapsulating Ethernet frames using RFC2684.
- Enables other services to co-exist on the same Ethernet interface. With the use of VLANs, EVLS, IP, and IP VPN services can be offered to provide a rich suite of network solutions to address both layer 2 and layer 3 needs.
- Provides transparent tunnelling of IEEE 802.3 and Ethernet v2.0 traffic.

Terminology

Refer to the following list for term definitions:

- LAN-to-LAN EVC. For an end-to-end configuration using LAN connectivity with non-aggregation transport, an EVC is established over a transport that is dedicated to this LAN and terminates on a far-end LAN. This EVC is referred to as a LAN-to-LAN EVC.
- LAN-to-VLAN EVC. For an end-to-end configuration using LAN connectivity with a non-aggregation transport, an EVC is established over a transport that is dedicated to this LAN and terminates on a far-end VLAN. This EVC is referred to as a LAN-to-VLAN EVC.
- 1:1 VLAN-to-VLAN EVC. For an end-to-end configuration using VLAN connectivity with a non-aggregation transport, an EVC is established over a transport that is dedicated to this VLAN and terminates on a far-end VLAN. This EVC is referred to as a 1:1 VLAN-to-VLAN EVC.
- N:1 VLAN-to-VLAN EVC. For an end-to-end configuration using VLAN connectivity with an aggregation transport, an EVC is established over a transport that is shared with other EVCs using ETS, and terminates on a far-end VLAN. This EVC is referred to as an N:1 VLAN-to-VLAN EVC.

- LAN-to-ATM EVC. For a single-ended configuration using LAN connectivity with a non-aggregation transport, an EVC is established over a transport that is dedicated to this LAN and terminates on an ATM VCC. Routing protocols for the ATM VCC can be one of the following: ATM inter-network interface (AINI), interim inter-switch signalling protocol (IISP), private network-to-network interface (PNNI), and user-to-network interface (UNI). This EVC is referred to a LAN-to-ATM EVC.
- 1:1 VLAN-to-ATM EVC. For a single-ended configuration using VLAN connectivity with a non-aggregation transport, an EVC is established over a transport that is dedicated to this VLAN and terminates on an ATM VCC. Routing protocols for the ATM VCC can be one of the following: AINI, IISP, PNNI, and UNI. This EVC is referred to as a 1:1 VLAN-to-ATM EVC.
- N:1 VLAN-to-ATM EVC. For a single-ended configuration using VLAN connectivity with an aggregation transport, an EVC is established over a transport that is shared with other EVCs using ETS. The transport terminates on an ATM VCC. Routing protocols for the ATM VCC can be one of the following: AINI, IISP, PNNI, and UNI. This EVC is referred to as an N:1 VLAN-to-ATM EVC.

How does EVLS work?

The Ethernet virtual line service (EVLS) establishes an Ethernet virtual connection (EVC) between ports or virtual local area networks (VLANs) using an ATM soft permanent virtual circuit (SPVC) VCC. The ATM SPVC can be configured to terminate on either another EVC or on a tandem ATM interface. For a list of the function processors (FPs) supporting EVLS, see table FPs supporting EVLS (page 144). If the ATM SPVC is configured to terminate on another EVC it is referred to as an end-to-end EVLS solution. If the ATM SPVC is configured to terminate on a tandem ATM interface, it is referred to as a single-ended EVLS solution.

The Ethernet virtual connection (EVC) uses an ATM VCC to transport layer 2 Ethernet traffic with RFC2684 bridged mode LLC/SNAP encapsulation, which is then carried as AAL5 frames across the ATM network. The AAL5 segmentation and reassembly (SAR) function is provided by the tandem ATM FPs. For frames egressing onto the ATM link, the ATM FP segments the AAL5 frames. For cells ingressing from the ATM link, the ATM FP reassembles the AAL5 frames. The layer 2 Ethernet headers remain unaltered by and between the peer ATM tandem FPs. The Ethernet frames are then transmitted to the Ethernet interface unaltered from the way they arrived at the ingress Ethernet interface, with one exception. This exception is the case of a VLAN-to-VLAN EVC, where the VLAN identifier (VID) that is received by the EVC ingress is swapped with the VID to which it is transmitted at the EVC egress. For a VLAN-to-LAN EVC, the VLAN tag may or may not be present. If the VLAN tag is present, it may or may not be modified depending on the configuration used. For information on data flow, see Frame data flows (page 150).

The Ethernet transport system (ETS) provides aggregation of traffic from multiple VLANs on a single port across a single ATM VCC. The traffic that is aggregated on any single ETS *EvlsAtmTransport* SPVC is all from the same Ethernet interface. Each VLAN can be configured to use aggregation.

When a VLAN solicits for transport, ETS searches for an *EvlsAtmTransport* with matching *lanApplication* and *prefixNsapAddress* attributes. If a match is found, then the VLAN is successfully linked to that *EvlsAtmTransport* instance.

EVLS is supported on Ethernet interfaces in both port-mode and VLAN-mode. When services are configured on dedicated VLANs, EVLS can coexist with other Multiservice Switch solutions on an Ethernet interface in VLAN-mode.

There is no layer 2 bridging or layer 3 routing provided or supported by this feature. Therefore, auto-discovery of peer layer 2 circuits is not required. Each Ethernet port and VLAN is directly linked to its own dedicated EVC.

The EVC SPVC that interconnects Ethernet ports and VLANs supports resiliency during ATM network failures. The EVC SPVC attempts the initial call setup after the LanApplication service has received all of its provisioning data.

Operation of EVLS is shown in the figure EVLS end-to-end view (page 146).

Attention: EVLS does not support edge-based routing (EBR) for the SPVC.

Supported cards

Table FPs supporting EVLS (page 144) lists the function processors (FPs) that are recommended when configuring EVLS.

FPs supporting EVLS

FPs supported	PEC
Ethernet PQC12 FPs	
4-port 10/100 BaseT Ethernet	NTNQ95
8-port 10/100 BaseT Ethernet	NTNQ92
Ethernet FQM FP	
4-port Gigabit Ethernet	NTHW49
Multiservice Switch 15000/20000 ATM trunk FPs	
1-port OC-48/STM-16 ATM FP with APS (PQC2)	NTHW01
4-port DS3Ch ATM FP with IMA (PQC2)	NTHR31DA
4-port OC-3/SMT-1 multimode ATM (PQC12)	NTHW05

(1 of 2)

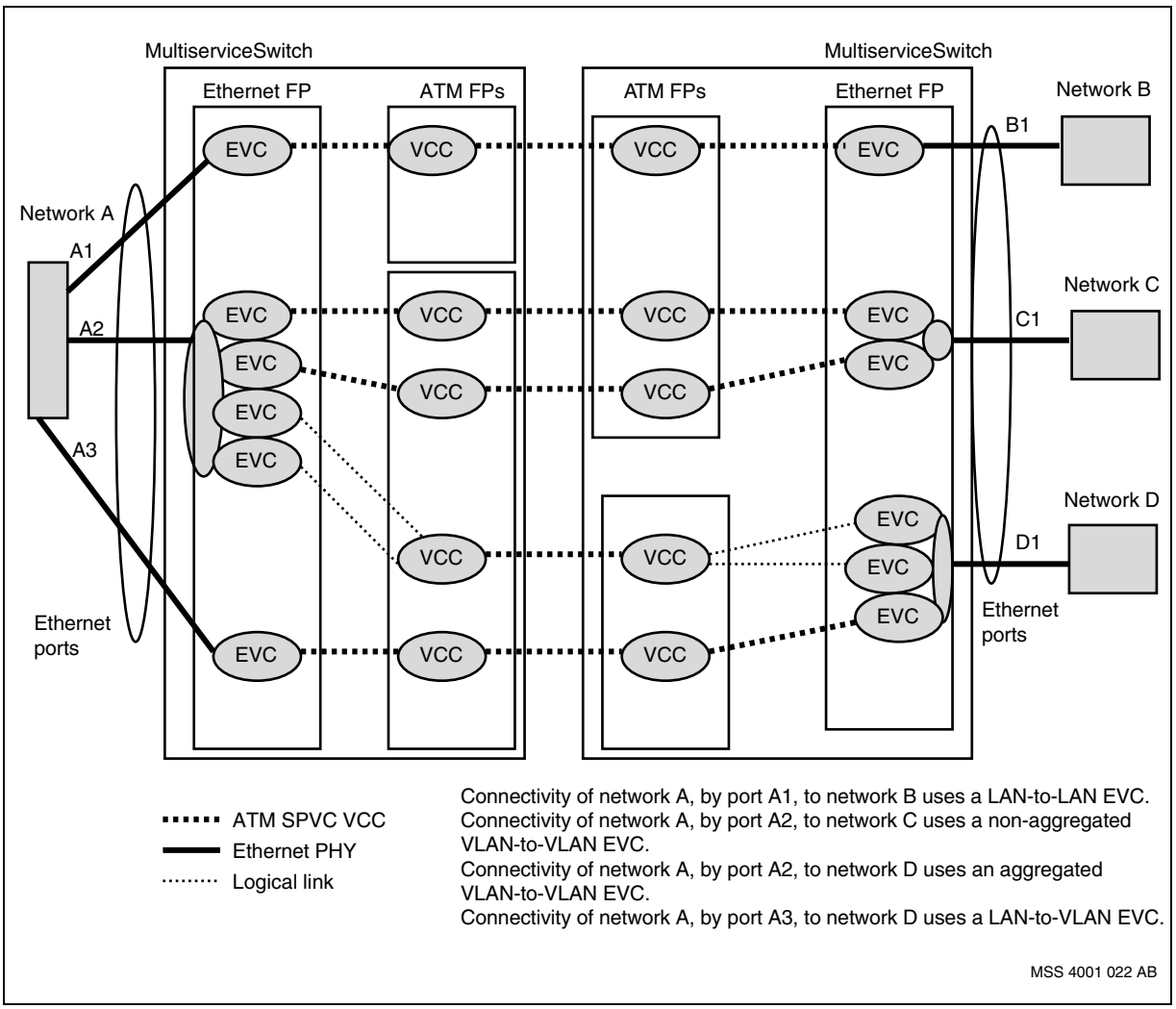
FPs supporting EVLS (continued)

FPs supported	PEC
4-port OC-3/SMT-1 single-mode intermediate reach ATM (PQC12)	NTHW15
4-port OC-3/SMT-1 multimode ATM (PQC2)	NTHR17
4-port OC-3/STM-1 single-mode intermediate reach ATM (PQC2)	NTHR21
4-port MR POS and ATM (GQM)	NTHW46
4-port OC-12/STM-4 ATM (PQC12)	NTHW86
12-port DS3 ATM (PQC2)	NTHR23
12-port E3 ATM (PQC2)	NTHR25
16-port OC-3/STM-1 POS and ATM (GQM)	NTHW44
16-port OC-3/STM-1 ATM FP with MT-RJ connectors (PQC2)	NTHW21
16-port OC-3/STM1 ATM FP with LC connectors (PQC2)	NTHW31
Multiservice Switch 7400 ATM trunk FPs	
2-port OC-3 ATM IP multimode (PQC2)	NTNQ65
2-port OC-3 ATM IP single mode (PQC2)	NTNQ66
2-port STM-1 electrical ATM IP	NTNQ90
2-port STM-1 electrical channelized CES/ATM/IMA (PQC2)	NTNQ91
2-port STM-1 optical channelized CES/ATM/IMA (PQC2)	NTNQ96
3-port DS3 ATM single mode (PQC2)	NTNQ68
3-port E3 ATM single mode (PQC2)	NTNQ67
8-port DS1 MSA (PQC2)	NTNQ61
8-port E1 MSA (PQC2)	NTNQ60
32-port DS1 MSA 1-slot	NTNQ94
32-port DS1 MSA 2-slot	NTNQ74
32-port DS1 MSA 2-slot with 2-port (protected) OC-3/STM-1 multimode	NTNQ76
32-port DS1 MSA 2-slot with 2-port (protected) OC-3/STM-1 single mode	NTNQ78
32-port E1 MSA 1-slot	NTNQ93
32-port E1 MSA 2-slot	NTNQ69
32-port E1 MSA 2-slot with 2-port (protected) OC-3/STM-1 multimode	NTNQ71
32-port E1 MSA 2-slot with 2-port (protected) OC-3/STM-1 single mode	NTNQ73
(2 of 2)	

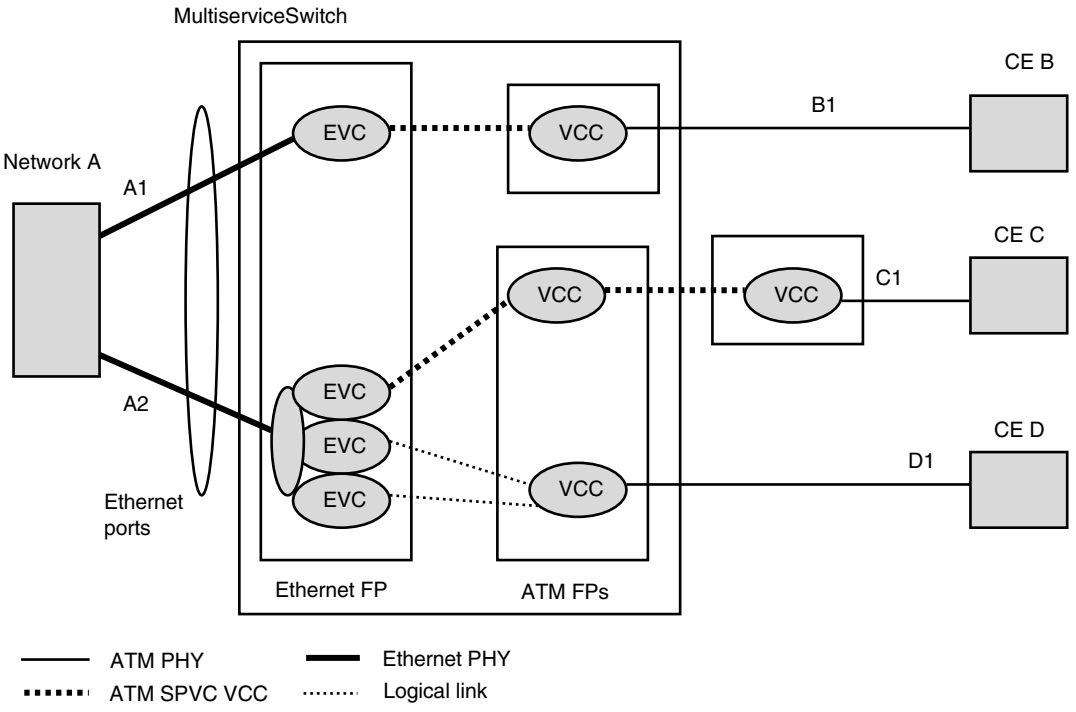
Physical network

On the physical network, connectivity of Ethernet LAN segments is typically collocated with the bridge, router, or switch that links the segments together. If these segments are not collocated, but in disparate locations, interconnecting segments typically require customer equipment to connect to the WAN data circuit-terminating equipment (DCE) using a frame relay or ATM layer 2 media. EVLS enables customer equipment (CE) to connect to the WAN DCE using native Ethernet layer 2 media while maintaining the existing ATM core network as the transport technology. See figures EVLS end-to-end view (page 146) and EVLS single-ended view (page 147) for more information.

EVLS end-to-end view



EVLS single-ended view



Connectivity of network A, by port A1, to customer equipment (CE) B uses a LAN-to-ATM EVC for LAN-to-LAN transport or LAN-to-VLAN transport.

Connectivity of network A, by port A2, to CE C uses a non-aggregated VLAN-to-ATM EVC for LAN-to-VLAN transport or VLAN-to-VLAN transport.

Connectivity of network A, by port A2, to CE D uses an aggregated VLAN-to-ATM EVC for VLAN-to-VLAN transport.

MSS 4001 023 AB

Carrier grade functionality

The carrier-grade functionality supported by this feature includes:

- CP switchover (CPSO) behavior: supported in warm standby mode only. Administrative states of all relevant components are reset to their default upon CPSO. Operational states will, however, reflect the current operational status of the component.
- FP switchover behavior: ATM FP switchovers are not impacted.
- Software upgrade behavior: hitless software migration (HSM) is not supported for EVLS.
- Line automatic protection switching is not supported.

Timers

The Ethernet virtual connection (EVC) SPVC that interconnects Ethernet ports supports resiliency during ATM network failures. The VCC attempts the initial call setup once the LanApplication service has received all of its provisioning data. If the first call setup attempt fails, the EVC SPVC retries to connect to the remote endpoint indefinitely. If a call fails after the VCC has been successfully established, the EVC SPVC attempts to reconnect to the remote called endpoint indefinitely, every 20 seconds by default (SPVC retry timer). A holding time is used to handle the failure cases when receiving a release.

The EVC SPVC retry timer is configured per module. This timer is started after a release message is received by the LAN application. Upon its expiry, EVC starts the connection setup procedures. The timer is used by the EVC that has configured *Spvc* components as a calling endpoint.

The hold-off timer is configured for every Ethernet interface. The hold-off timer specifies how long to hold off retries following the release of a connection. Since EVC SPVCs retries are paced, in cases where there are many connections, the entire process for n connections can take up to n connections divided by 40 seconds.

Addressing

In Nortel Multiservice Switch ATM networks, the only address format used is the one that adheres to OSI network service access point (NSAP) format. NSAP is a 40-digit long (20 bytes) address that follows a certain format.

Ethernet virtual line service (EVLS) uses NSAP addressing. It is necessary to use NSAP addressing in order for an EVC to interwork with ATM using SPVCs. ATM inter-network interface (AINI), interim interswitch signalling protocol (IISP), private network-to-network interface (PNNI) and user-to-network interface (UNI) can be used to setup SPVCs. The NSAP address combined with a connection ID uniquely identifies any Ethernet virtual connection (EVC) SPVC connection in the PNNI1.0 network.

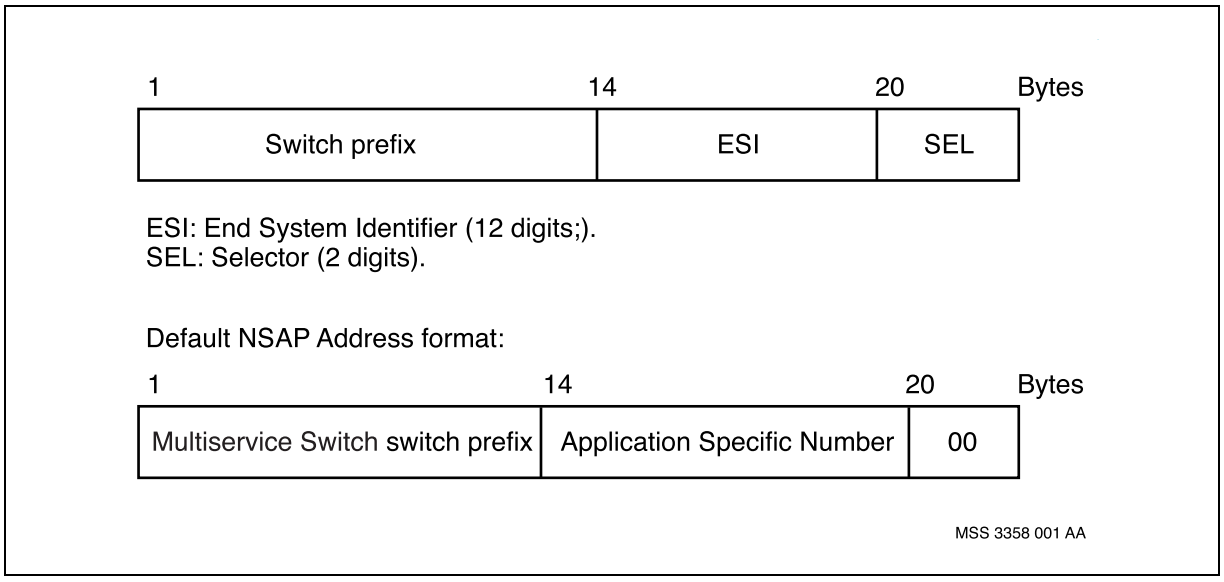
The address is constructed using binary-coded decimal (BCD) encoding (each two digits occupy one byte). The first two digits of the NSAP address contains an Authority and Format Identifier (AFI) value. AFI indicates the type of address encapsulated inside this NSAP address (i.e. DCC, ICD, or E.164). The last byte is a Selector (SEL) byte which has local significance. The ESI (End System Identifier) can be a MAC address. All other values of the AFI byte are reserved. This means that if an ATM switch receives a call request with an address that begins with a value of AFI other than the ones defined (39 for DDC; 47 for ICD) it can reject the call without violating ATMF standards compliance.

The NSAP address can be configured per LanApplication and per Ets, but it is recommended to allow EVLS and the ETS to generate the default NSAP address. Here is how the default NSAP address is built:

- Switch prefix (13 bytes) - the Multiservice Switch node prefix, as configured under the *Rtg* component.
- End System identifier (ESI) - the application specific number. The 7 bytes are constructed from the following:
  - 4 bytes system defined: LanApplication and EthTransportSystem system defined (0x0020480d)
  - 4 bits application ID: LanApplication ID (0x8), EthTransportSystem ID (0xB)
  - 12 bits application instance: not applicable for EthTransportSystem
  - 1 byte selective set to 0

For a breakdown of the NSAP address, see the figure NSAP address format (page 149).

NSAP address format



Frame data flows

The Ethernet service provides a cut-through frame data flow from an Ethernet LAN to an ATM VCC and from an ATM VCC to an Ethernet LAN. See the following sections for more information:

- EVLS frame formats (page 151)
- LAN-to-ATM EVC for LAN-to-LAN transport (Ethernet-to-ATM direction) (page 153)
- LAN-to-ATM EVC for LAN-to-LAN transport (ATM-to-Ethernet direction) (page 154)
- LAN-to-ATM EVC for LAN-to-VLAN transport (Ethernet-to-ATM direction) (page 155)
- LAN-to-ATM EVC for LAN-to-VLAN transport (ATM-to-Ethernet direction) (page 158)
- VLAN-to-ATM EVC for LAN-to-VLAN transport (Ethernet-to-ATM direction) (page 158)
- VLAN-to-ATM EVC for LAN-to-VLAN transport (ATM-to-Ethernet direction) (page 162)
- 1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (Ethernet-to-ATM direction) (page 165)
- 1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (ATM-to-Ethernet direction) (page 166)
- N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (Ethernet-to-ATM direction) (page 167)
- N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (ATM-to-Ethernet direction) (page 170)
- Packet modification and RFC2684 encapsulation (page 172)

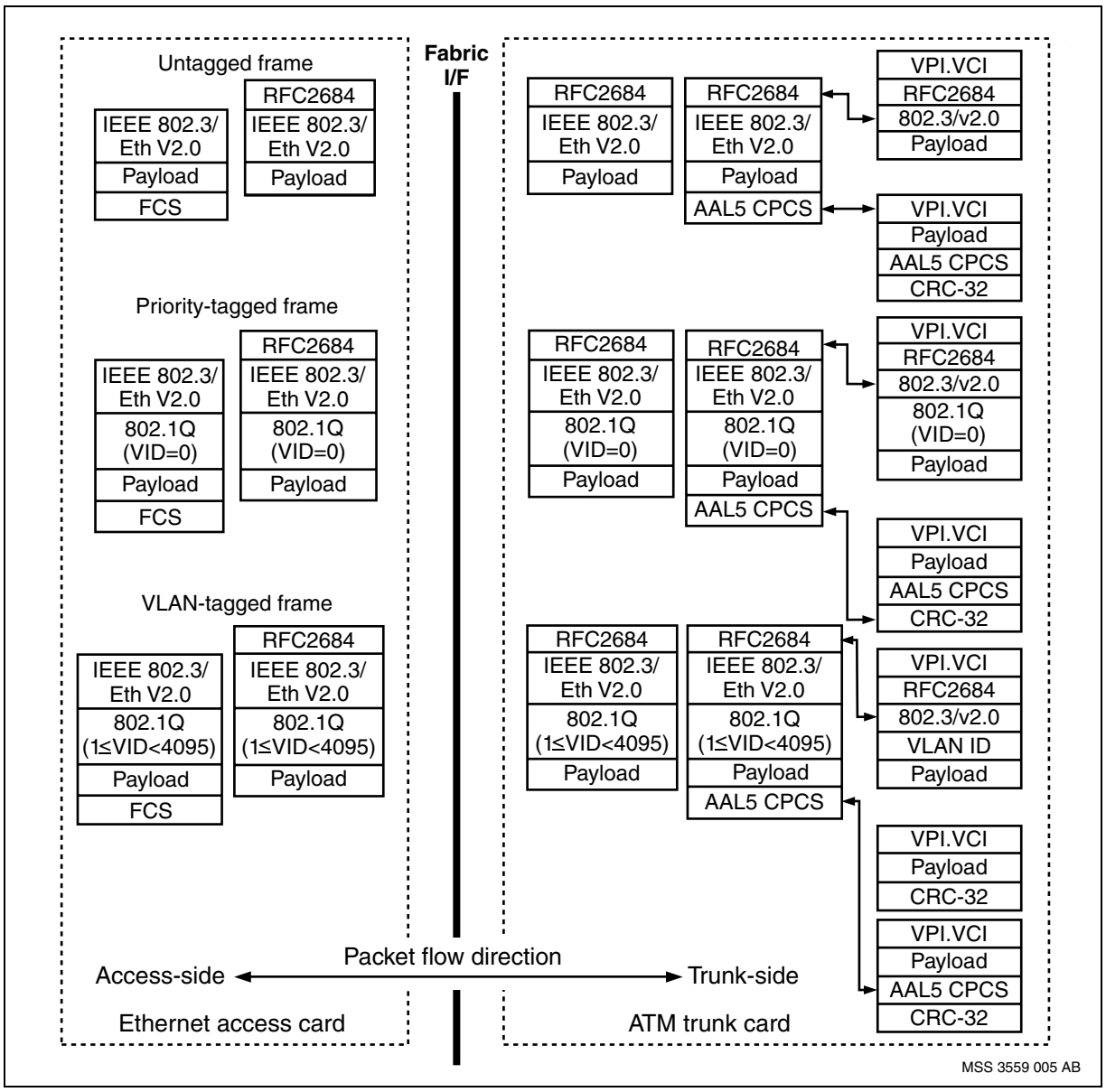
EVLS frame formats

On reception of traffic from an Ethernet interface, both Ethernet v2.0 and IEEE 802.3 layer 2 header encapsulation are accepted. On transmission of traffic to the interface, Ethernet v2.0 and IEEE 802.3 layer 2 header encapsulation is sent.

When EVLS is configured on a port-mode Ethernet interface, the unknown VLAN ID treatment behavior is to accept all traffic (VLAN-tagged, untagged, and priority-tagged traffic).

See the figure EVLS frame formats (page 152) for the data flow for all three types of frame formats.

EVLS frame formats



Ethernet policy rules

When EVLS is configured on a port-mode Ethernet interface, the *unknownVlanIdTreatment* attribute should be set to accept. In this case, all traffic received from the Ethernet interface is accepted. If the *unknownVlanIdTreatment* attribute is not set to accept, the default behavior is to discard all VLAN-tagged traffic, and to accept only untagged and priority-tagged traffic.

In VLAN-mode, the default behavior for the *unknownVlanIdTreatment* is to discard traffic received for a VLAN that is not configured on the Ethernet interface.

For more information about Ethernet policy rules, see the Understanding Ethernet services chapter in NN10600-821 *Nortel Multiservice Switch 7400/15000/20000 Operations: Ethernet Services*.

LAN-to-ATM EVC for LAN-to-LAN transport (Ethernet-to-ATM direction)

The ingress framer function block of the Ethernet FP performs frame delineation and validation, keeping numerous counts of both good and error frames. Statistics collected by the ingress framer are also provided.

The PHY device checks the FCS of the Ethernet frames and strips the FCS before forwarding the frame onto the ingress framer pre-classification block.

The pre-classification function block on the ingress datapath compares the MAC Destination Address of the frame with the local port MAC address. If the addresses match, the frame is silently discarded. If the addresses do not match, the frame is further classified. The pre-classification function block assigns a fixed internal QoS corresponding to an emission priority (EP) and discard priority (DP) based on Ethernet perHopBehavior (PHB) for all received traffic per EVC.

The ingress network processor (NP) uses a configured single LAN or single VLAN to a single VCC (1:1) mapping to determine the egress VCC on the egress ATM FP.

If the ingress NP cannot forward a frame because either the VCC is down or the egress ATM FP is congested, the frame is discarded by the NP and statistics are collected.

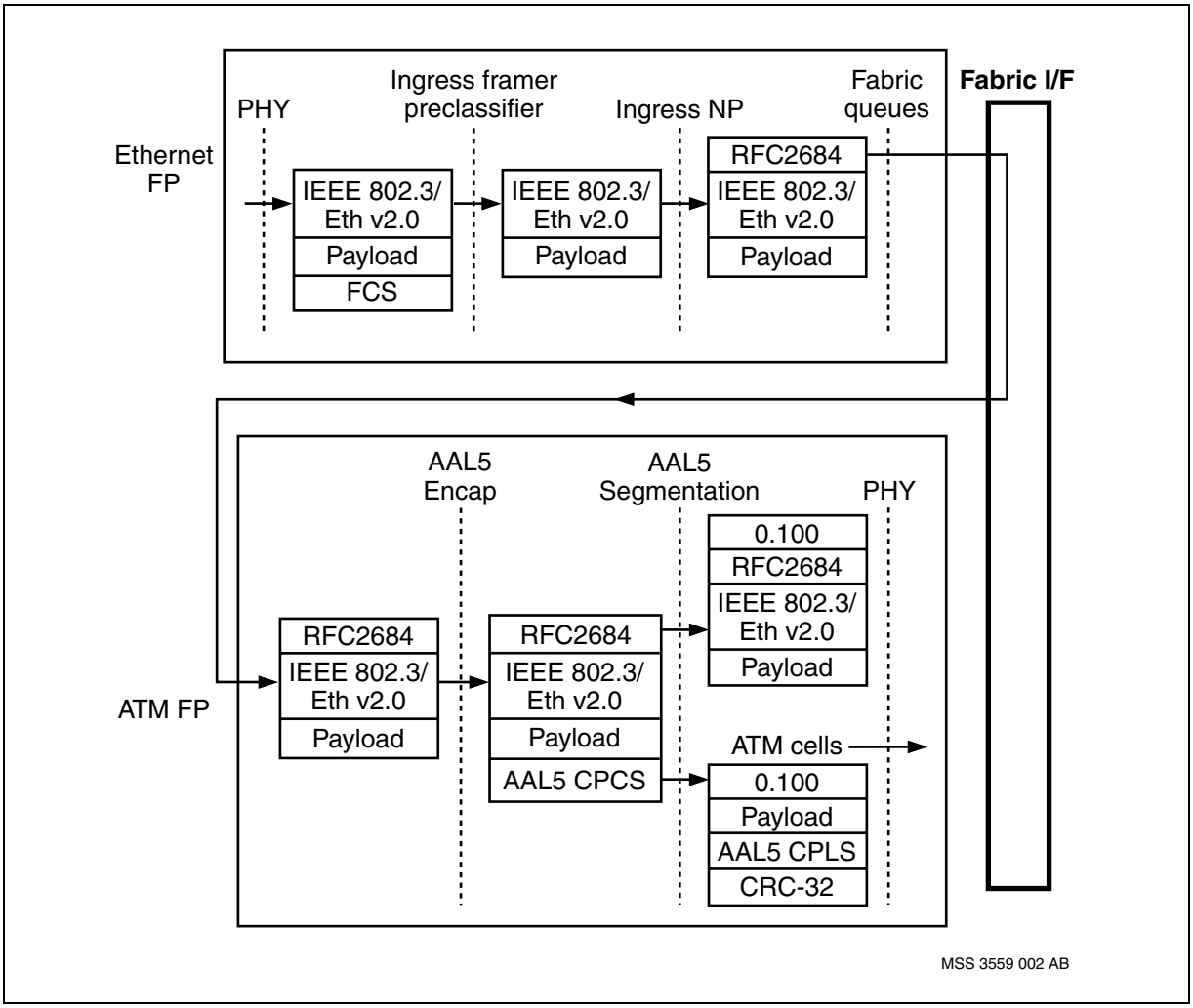
If the ingress NP can forward a frame, the frame is encapsulated using RFC2684 LLC encapsulation for bridged protocols.

On the ATM FP, frames are encapsulated with an AAL5 CPCS-PDU trailer and segmented into ATM cells by the AAL5 segmentation functional block.

The resulting ATM cells are placed on the outgoing VCC via the ATM traffic management device.

See the figure Ethernet-to-ATM direction frame flow (LAN-to-ATM EVC for LAN-to-LAN transport) (page 154) for more information.

Ethernet-to-ATM direction frame flow (LAN-to-ATM EVC for LAN-to-LAN transport)



LAN-to-ATM EVC for LAN-to-LAN transport (ATM-to-Ethernet direction)

The ATM VCC on the ATM FP is configured to forward traffic to the egress Ethernet FP. Received cells on this VCC go through the AAL5 reassembly function block and assembled frames are forwarded to the LAN EVC on the egress Ethernet FP.

The egress NP removes the RFC2684 encapsulation and forwards the resulting packet to the Ethernet interface. The egress PHY device recalculates and adds the FCS to the packet before transmitting the frame on the port.

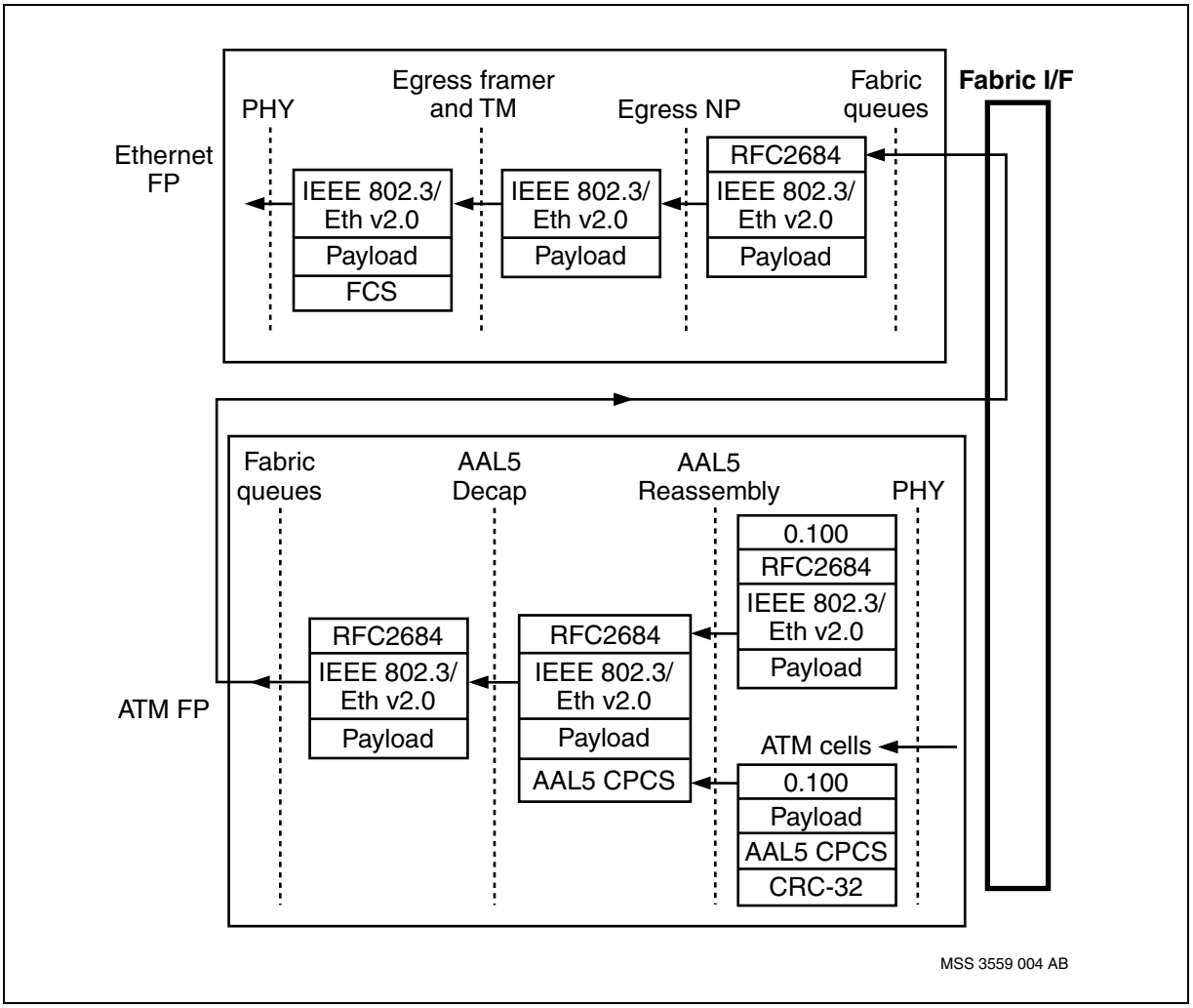
The host processor on the Ethernet FP and on the ATM FP is not involved in frame forwarding. It executes the control function by performing the following:

- EVC SPVC setup

- frame datapath configuration
- statistics collection and aggregation

Refer to the figure ATM-to-Ethernet direction frame flow (LAN-to-ATM EVC for LAN-to-LAN transport) (page 155) for a visual representation of egress traffic and the path it takes.

ATM-to-Ethernet direction frame flow (LAN-to-ATM EVC for LAN-to-LAN transport)



LAN-to-ATM EVC for LAN-to-VLAN transport (Ethernet-to-ATM direction)

The ingress framer function block of the Ethernet FP performs frame delineation and validation, keeping numerous counts of both good and errored frames. Statistics collected by the ingress framer are also provided.

The PHY device checks the FCS of the Ethernet frames and strips the FCS before forwarding the frame onto the ingress frame pre-classification block.

The pre-classification function block on the ingress datapath is disabled. This means that all received frames are assigned a fixed internal QoS corresponding to an emission priority (EP) and discard priority (DP) based on Ethernet perHopBehavior (PHB) for all received traffic per EVC.

The ingress network processor (NP) uses a configured single LAN to a single VCC (1:1) mapping to determine the egress VCC on the egress ATM FP.

If the ingress NP cannot forward a frame because either the VCC is down or the egress ATM FP is congested, the frame is discarded by the NP and statistics are collected.

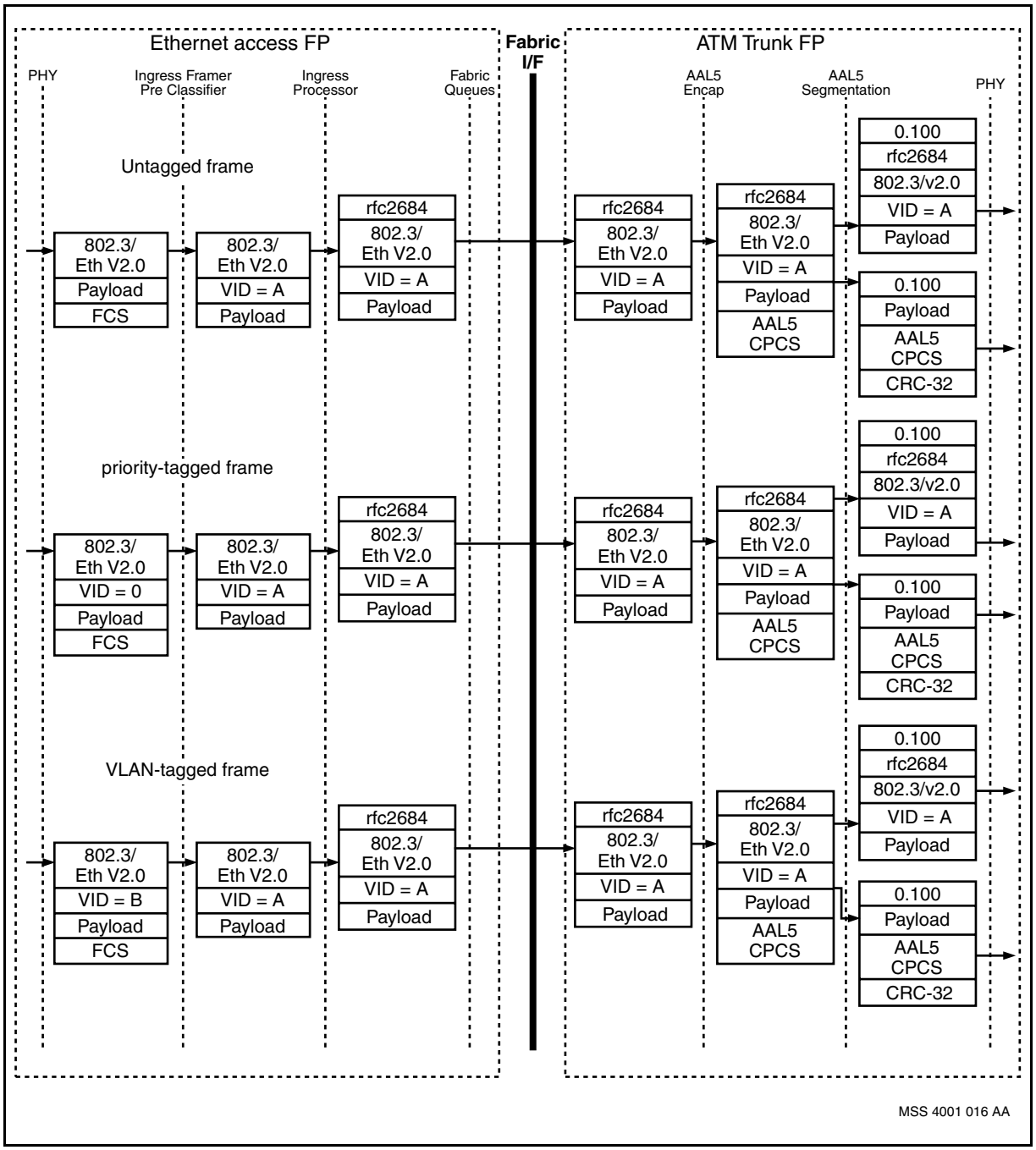
If the ingress NP can forward a frame, it adds a VLAN tag with the VLAN ID set to the remote VLAN for untagged traffic and it swaps the VLAN tag with the remote VLAN tag for priority-tagged and VLAN-tagged traffic.

The frame is encapsulated using RFC2684 LLC encapsulation for bridged protocols.

On the ATM FP, frames are encapsulated with an AAL5 CPCS-PDU trailer and segmented into ATM cells by the AAL5 segmentation functional block. The resulting ATM cells are placed on the outgoing VCC via the ATM traffic management device.

See Ethernet-to-ATM direction frame flow (LAN-to-ATM EVC for LAN-to-VLAN transport) (page 157) for more information.

Ethernet-to-ATM direction frame flow (LAN-to-ATM EVC for LAN-to-VLAN transport)



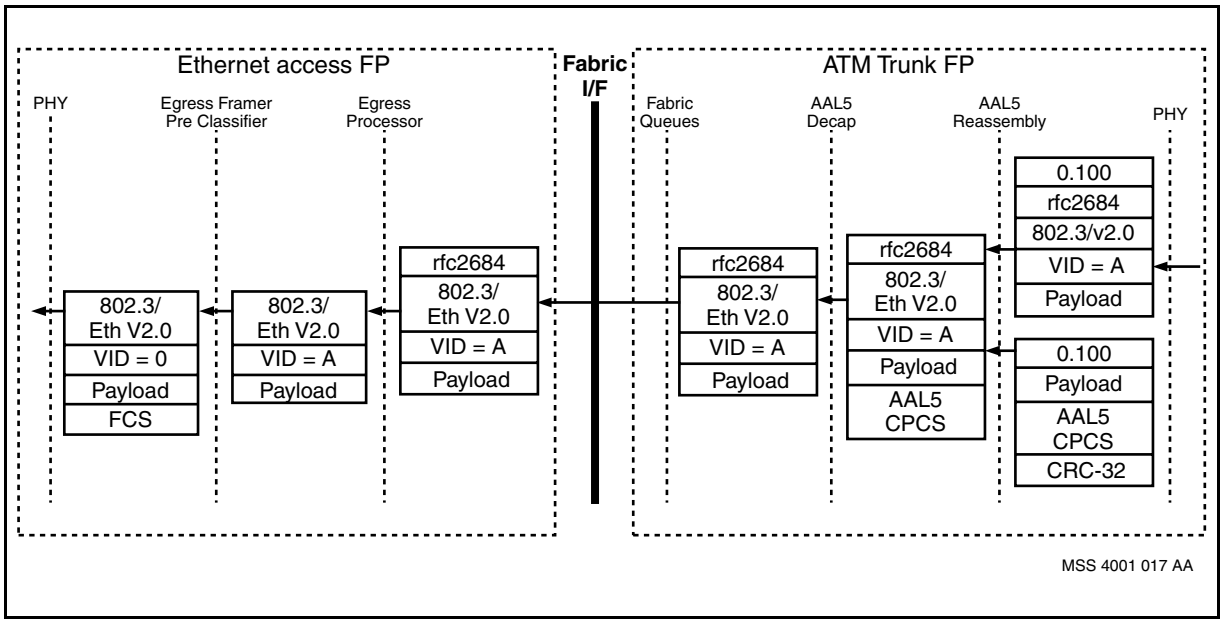
LAN-to-ATM EVC for LAN-to-VLAN transport (ATM-to-Ethernet direction)

The ATM VCC on the ATM FP is configured to forward traffic to the egress Ethernet FP. Received cells on this VCC go through the AAL5 reassembly function block and assembled frames are forwarded to the LAN EVC on the egress Ethernet FP.

The egress NP removes the RFC2684 encapsulation and forwards the resulting packet to the Ethernet interface. The egress PHY device recalculates and adds the FCS to the packet before transmitting the frame on the port.

Refer to the figure ATM-to-Ethernet direction frame flow (LAN-to-ATM EVC for LAN-to-VLAN transport) (page 158) for a visual representation of egress traffic and the path it takes.

ATM-to-Ethernet direction frame flow (LAN-to-ATM EVC for LAN-to-VLAN transport)



VLAN-to-ATM EVC for LAN-to-VLAN transport (Ethernet-to-ATM direction)

The ingress framer function block of the Ethernet FP performs frame delineation and validation, keeping numerous counts of both good and errored frames. Statistics collected by the ingress framer are also provided.

The PHY device checks the FCS of the Ethernet frames and strips the FCS before forwarding the frame onto the ingress frame pre-classification block.

The pre-classification function block on the ingress datapath is disabled. This means that all received frames are assigned a fixed internal QoS corresponding to an emission priority (EP) and discard priority (DP) based on Ethernet perHopBehavior (PHB) for all received traffic per EVC.

The ingress network processor (NP) uses a configured VLAN to single VCC (1:1) mapping to determine the egress VCC on the egress ATM FP.

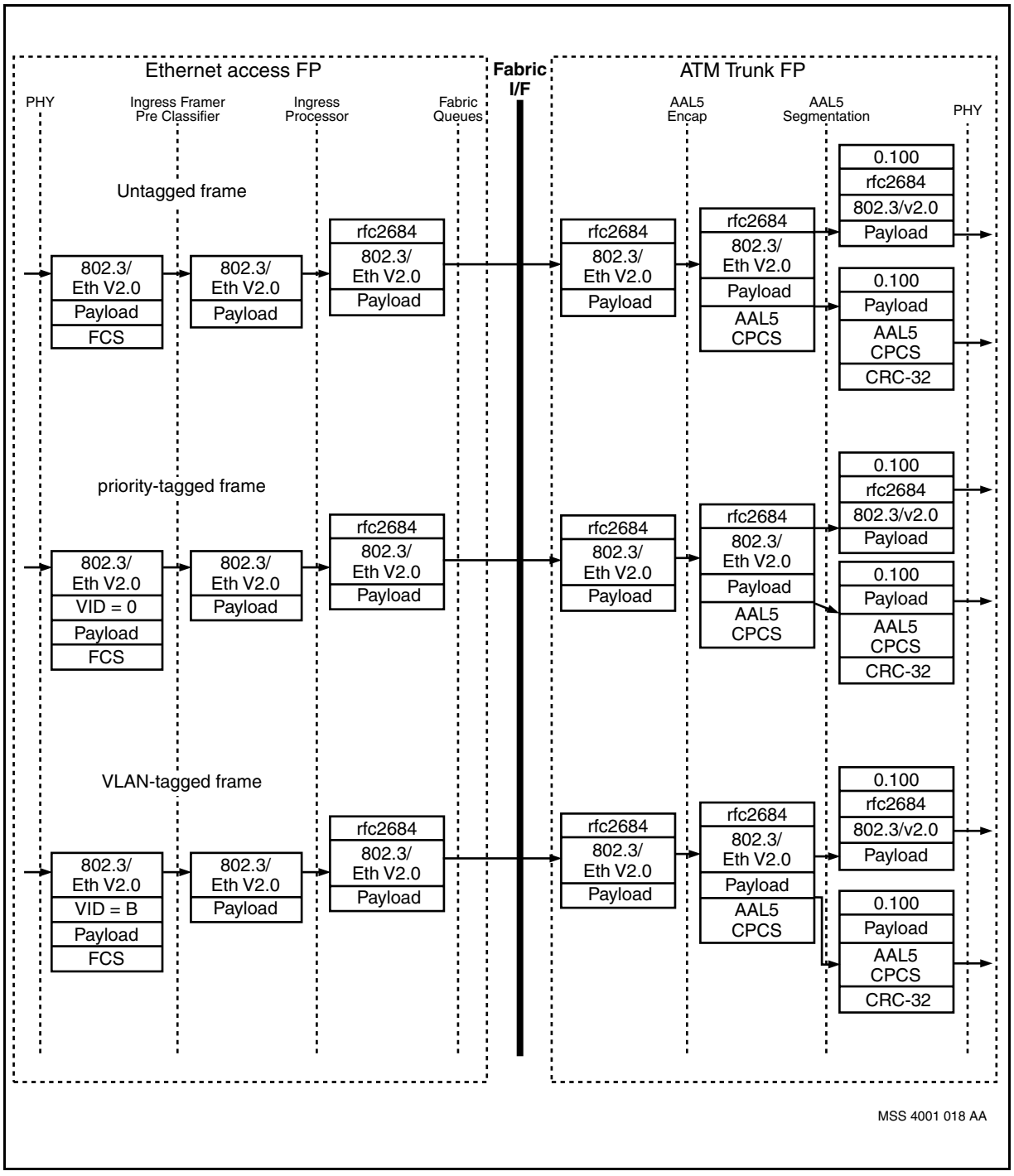
If the ingress NP can not forward a frame because either the VCC is down or the egress ATM FP is congested, the frame is discarded by the NP. If the ingress NP can forward a frame, it is encapsulated using RFC2684 LLC encapsulation for bridged protocols.

If the *preserveVlanTag* attribute under the *La Vlan Evc* component is set to disabled, the untagged traffic is forwarded unmodified. In the case of priority-tagged and VLAN-tagged traffic, the VLAN tag is removed (refer to the figure, Ethernet-to-ATM direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with preserveVlanTag attribute disabled (page 160).

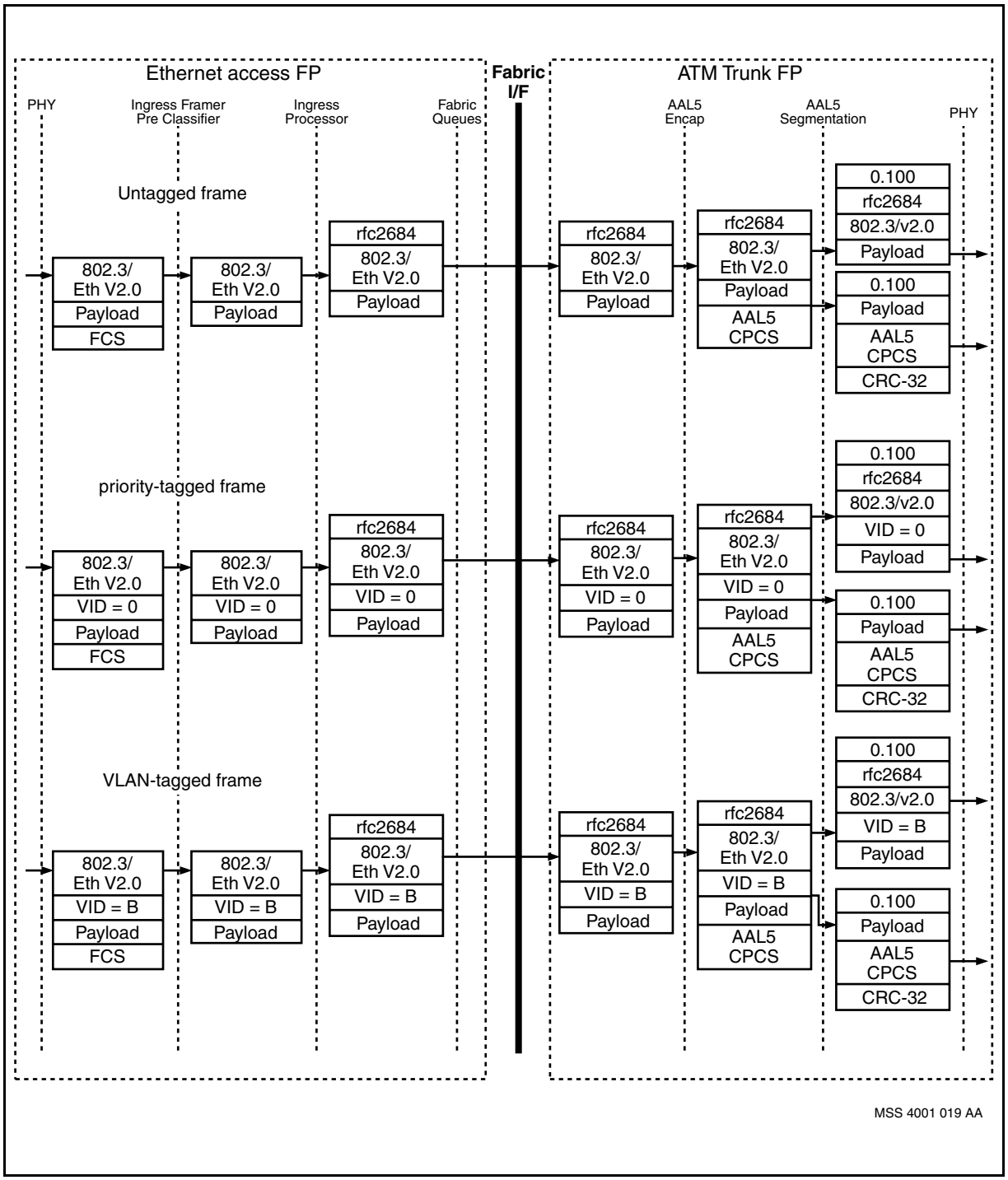
If the *preserveVlanTag* attribute under the *La Vlan Evc* component is set to enabled, all type of traffic (untagged, priority-tagged, and VLAN-tagged) is forwarded unmodified (refer to the figure, Ethernet-to-ATM direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with preserveVlanTag attribute enabled (page 161).

On the ATM FP, frames are encapsulated with an AAL5 CPCS-PDU trailer and segmented into ATM cells by the AAL5 segmentation functional block. The resulting ATM cells are placed on the outgoing VCC via the ATM traffic management device.

Ethernet-to-ATM direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with *preserveVlanTag* attribute disabled



Ethernet-to-ATM direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with *preserveVlanTag* attribute enabled



VLAN-to-ATM EVC for LAN-to-VLAN transport (ATM-to-Ethernet direction)

The ATM VCC on the ATM FP is configured to forward AAL5 reassembled frames to the Ethernet FP. Received cells are processed by the AAL5 reassembly function block. Assembled frames are forwarded to the VLAN EVC on the Ethernet FP.

The egress FP removes the RFC2684 encapsulation and forwards the resulting packet to the Ethernet interface.

VLAN

If the egress processor can forward a frame and the *remoteVlanId* attribute under the *La Vlan Evc EthernetSourceEvc* component is set to 0, it adds a VLAN tag with VLAN ID set to local VLAN ID for untagged traffic and swaps the VLAN tag with a local VLAN ID for p-tagged and VLAN-tagged traffic. See the figure, ATM-to-Ethernet direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with remoteVlanID attribute equal to zero (page 163).

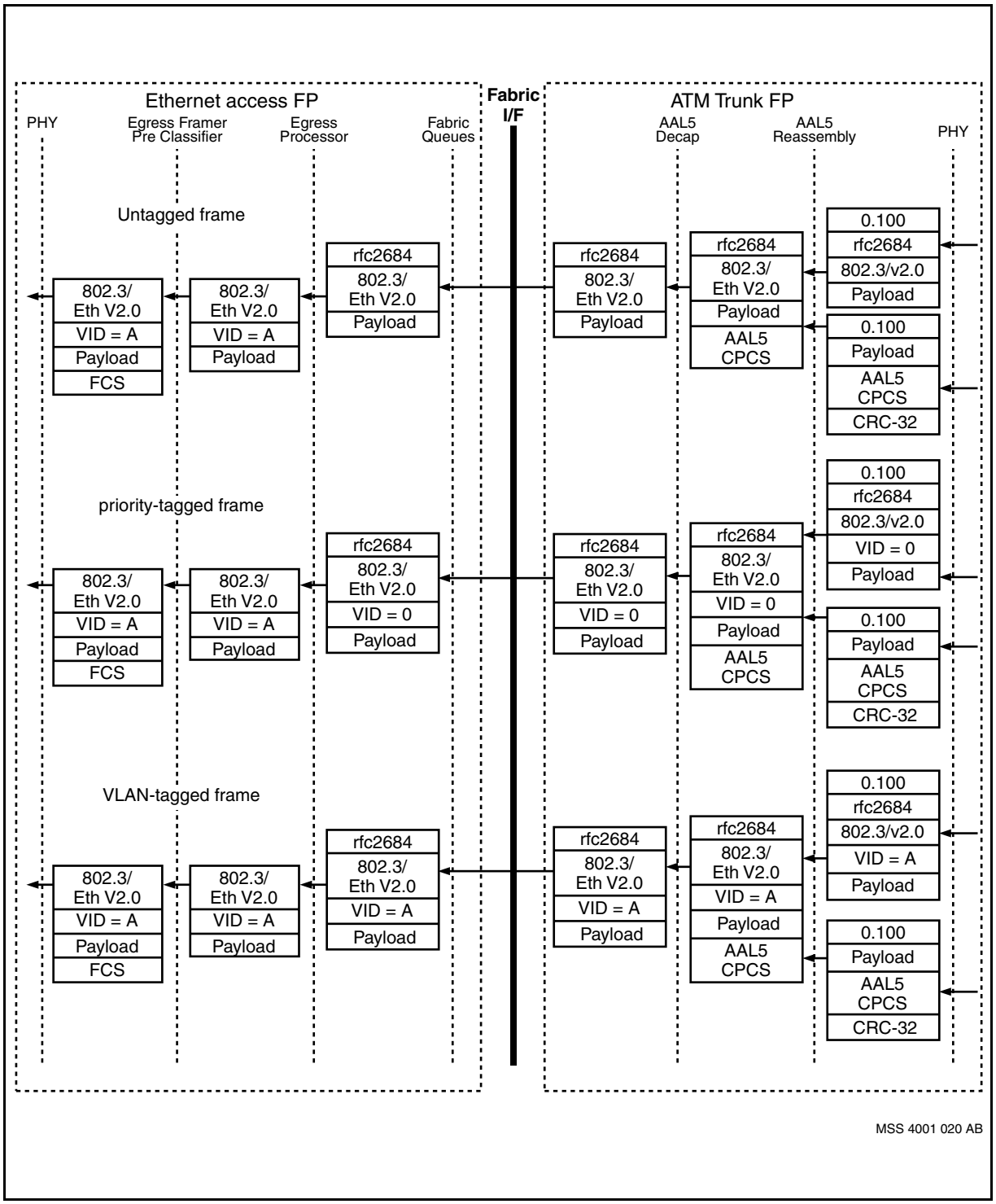
Port-VLAN

If the *egressTagRule* attribute under the *La EthernetPolicy* component is set to tagged, then the datapath is the same as any other VLAN as described the section, VLAN (page 162).

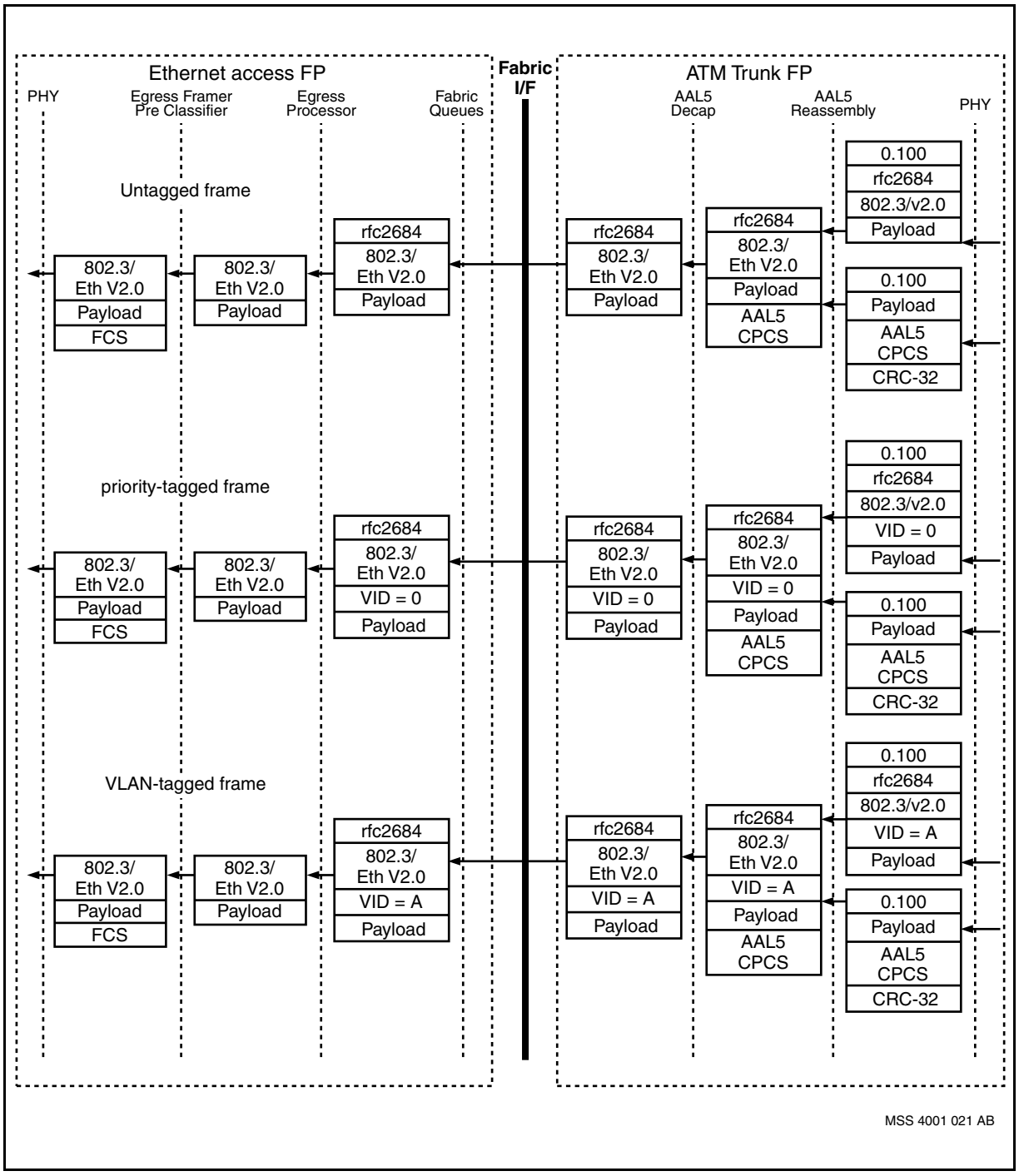
If *egressTagRule* attribute under the *La EthernetPolicy* component is set to untagged, the untagged traffic is forwarded unmodified. In the case of p-tagged and VLAN-tagged traffic, the VLAN tag is removed. For details on the frame flow for VLAN-to-ATM EVC when the *egressTagRule* attribute is set to untagged, see the figure, ATM-to-Ethernet direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with egressTagRule attribute is set to untagged (page 164).

The egress PHY device recalculates and adds the FCS to the packet before transmitting the frame on the port.

ATM-to-Ethernet direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with *remoteVlanID* attribute equal to zero



ATM-to-Ethernet direction frame flow (VLAN-to-ATM EVC for LAN-to-VLAN transport) with *egressTagRule* attribute is set to untagged



1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (Ethernet-to-ATM direction)

The ingress framer function block of the Ethernet FP performs frame delineation and validation, keeping numerous counts of both good and errored frames.

The PHY device checks the FCS of the Ethernet frames and strips the FCS before forwarding the frame onto the ingress frame pre-classification block.

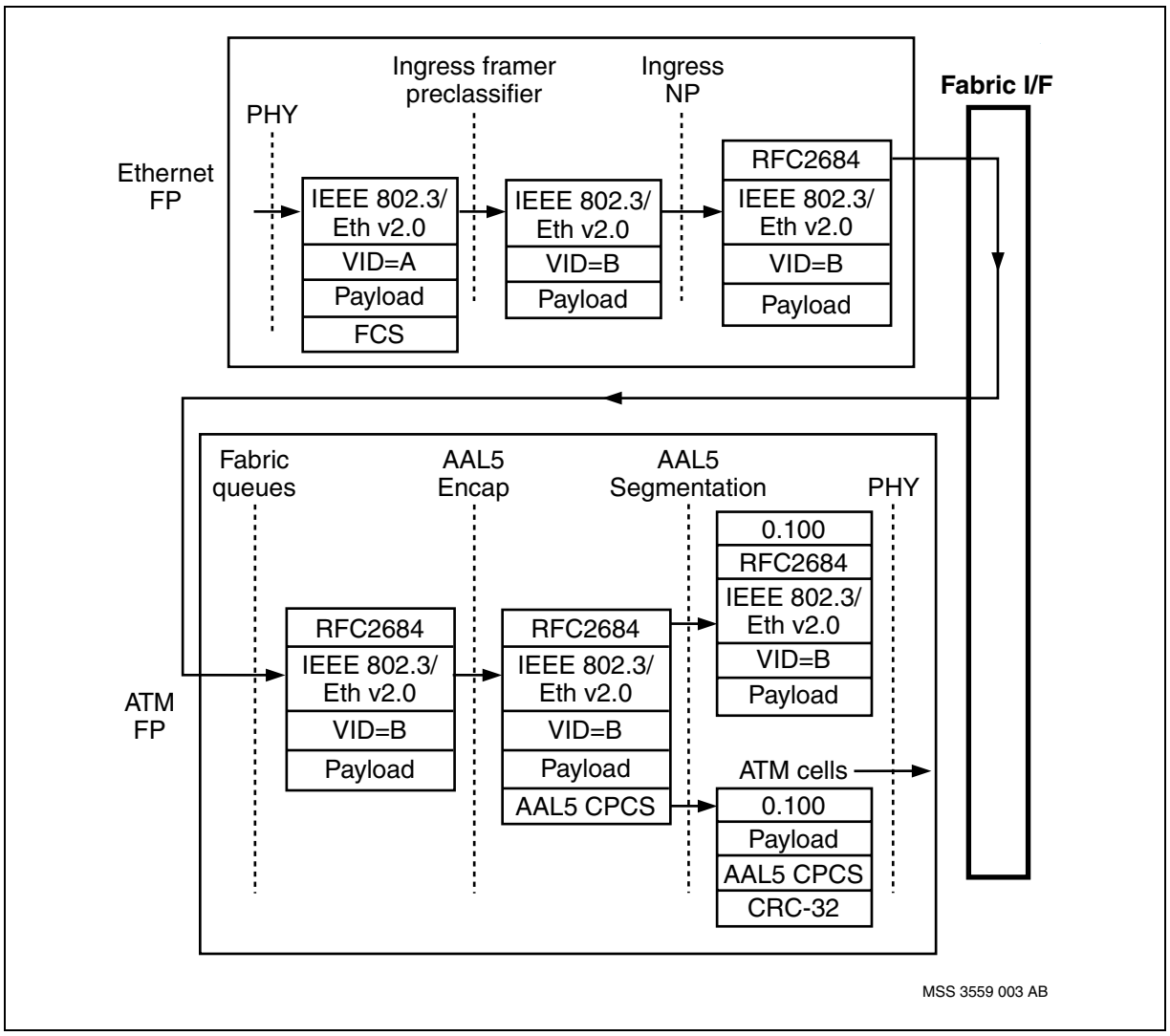
The pre-classification function block on the ingress datapath compares the MAC Destination Address of the frame with the local port MAC address. If the addresses match, the frame is silently discarded. If the addresses do not match, the frame is further classified. The pre-classification function block assigns a fixed internal QoS corresponding to an emission priority (EP) and discard priority (DP) based on Ethernet perHopBehavior (PHB) for all received traffic per EVC.

The ingress network processor (NP) uses a configured one-to-one VLAN to VCC on the tandem ATM FP. If the ingress NP can not forward a frame because either the VCC is down or the egress ATM FP is congested, the frame is discarded by the NP. If the ingress NP can forward a frame, it is encapsulated using RFC2684 LLC encapsulation for bridged protocols.

On the ATM FP, frames are encapsulated with an AAL5 CPCS-PDU trailer and segmented into ATM cells by the AAL5 segmentation functional block. The resulting ATM cells are placed on the outgoing VCC via the ATM traffic management device.

See the figure Ethernet-to-ATM direction frame flow (1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport) (page 166).

Ethernet-to-ATM direction frame flow (1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport)

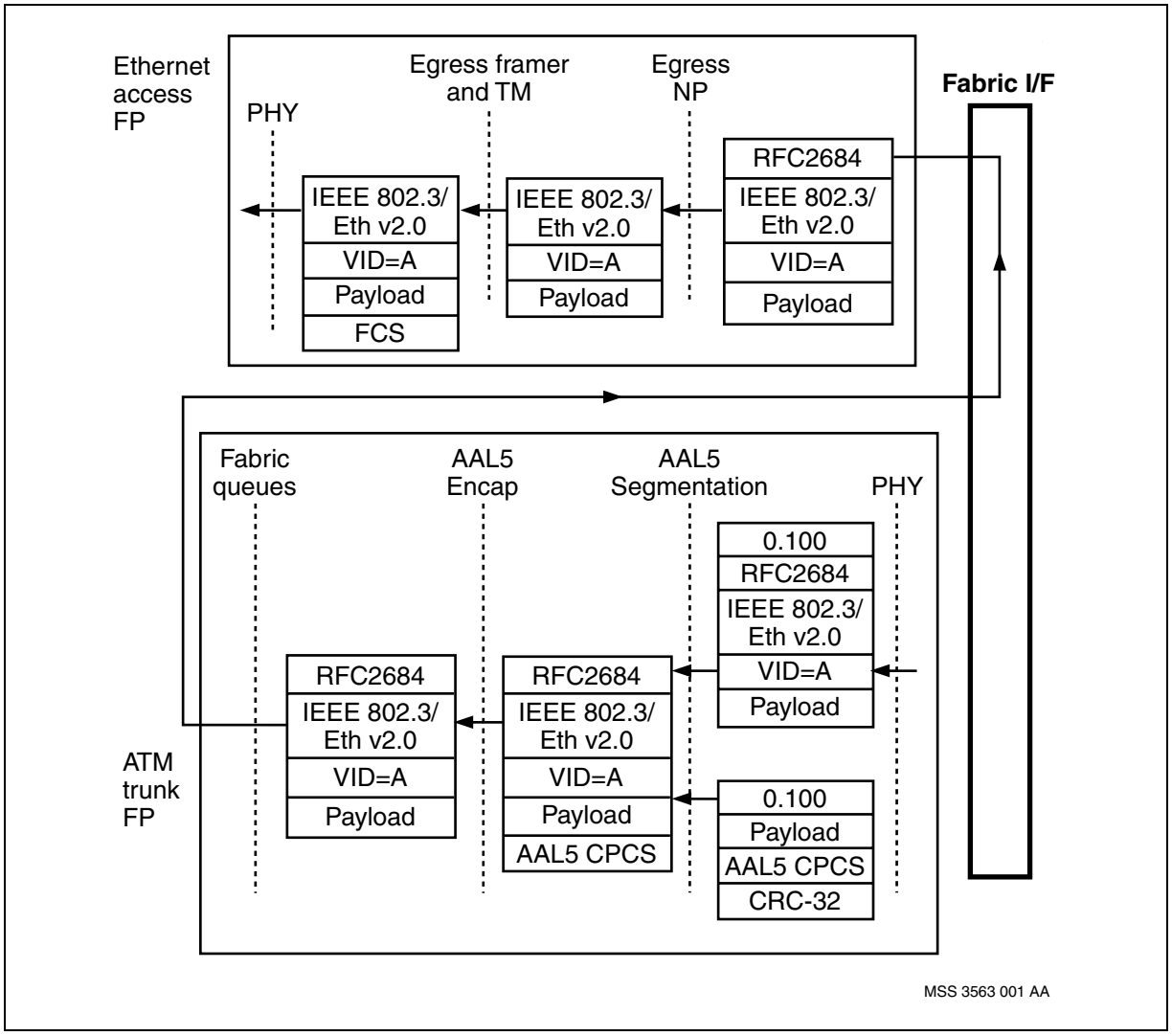


1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (ATM-to-Ethernet direction)

ATM VCC on the ATM FP is configured with internal forwarding information pointing to VLAN associated with the ingress VCC.

See ATM-to-Ethernet direction frame flow (1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport) (page 167) for more information.

ATM-to-Ethernet direction frame flow (1:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport)



N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (Ethernet-to-ATM direction)

The ingress framer function block of the Ethernet FP performs frame delineation and validation, keeping numerous counts of both good and errored frames.

The PHY device checks the FCS of the Ethernet frames and strips the FCS before forwarding the frame onto the ingress frame pre-classification block.

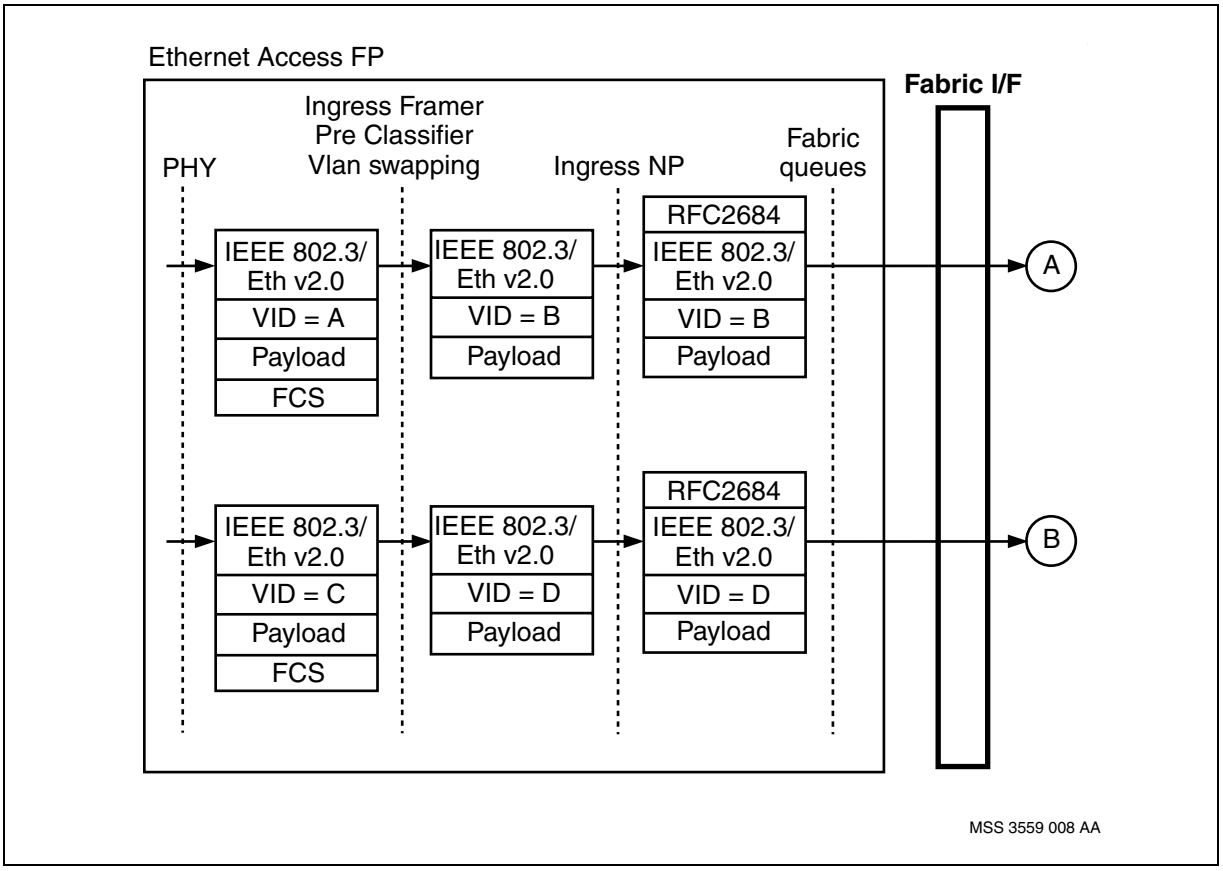
The pre-classification function block on the ingress datapath compares the MAC Destination Address of the frame with the local port MAC address. If the addresses match, the frame is silently discarded. If the addresses do not match, the frame is further classified. The pre-classification function block assigns a fixed internal QoS corresponding to an emission priority (EP) and discard priority (DP) based on Ethernet perHopBehavior (PHB) for all received traffic per EVC.

The ingress network processor (NP) uses a configured VLAN to VCC mapping to determine the egress VCC on the egress ATM FP. If the ingress NP can not forward a frame because either the VCC is down or the egress ATM FP is congested, the frame is discarded by the NP. If the ingress NP can forward a frame, it is encapsulated using RFC2684 LLC encapsulation for bridged protocols.

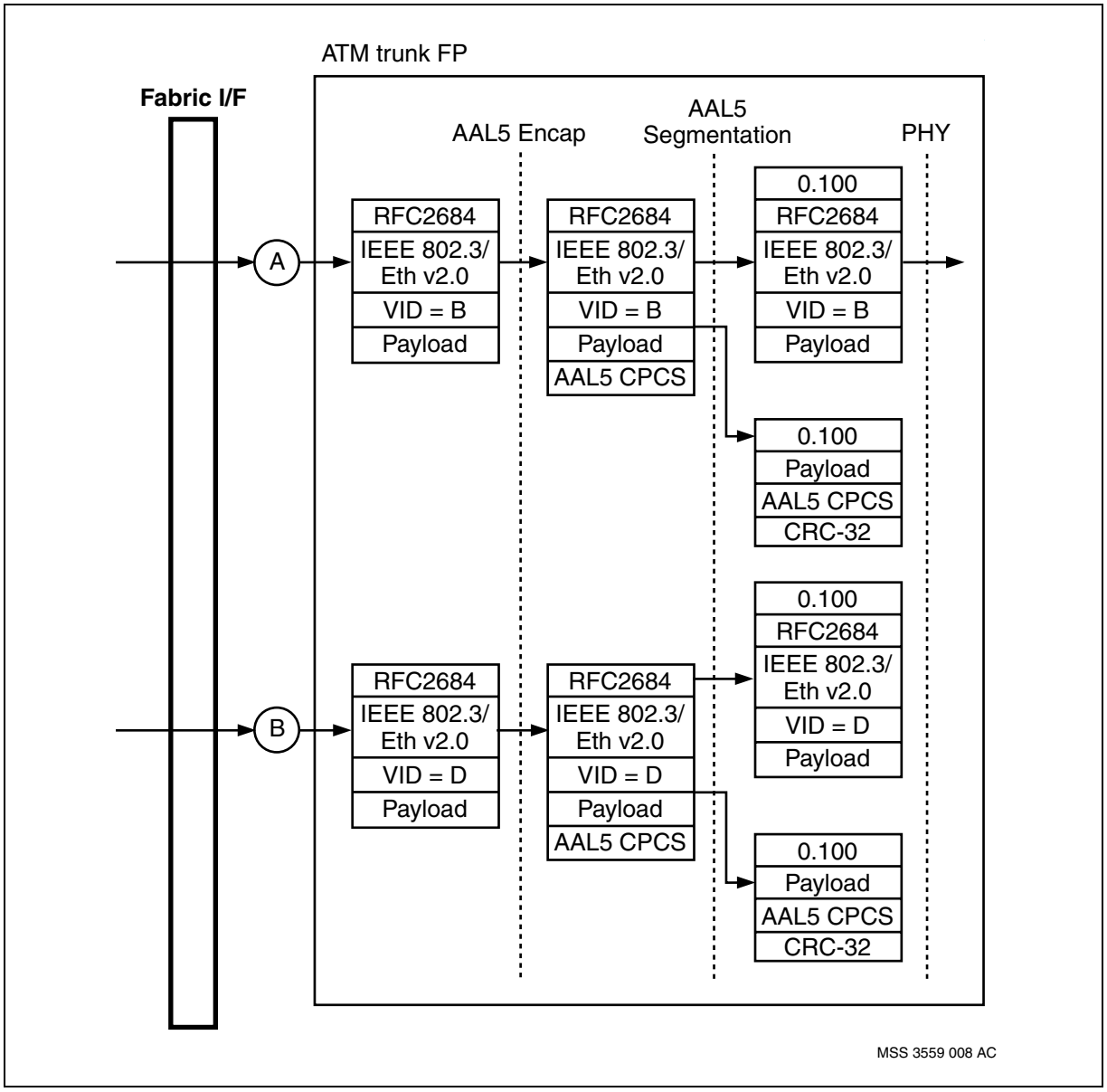
On the ATM FP, frames are encapsulated with an AAL5 CPCS-PDU trailer and segmented into ATM cells by the AAL5 segmentation functional block. The resulting ATM cells are placed on the outgoing VCC via the ATM traffic management device.

See the figure Ethernet-to-ATM direction frame flow (N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport): Part 1 of 2 (page 169).

Ethernet-to-ATM direction frame flow (N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport): Part 1 of 2



Ethernet-to-ATM direction frame flow (N:1 VLAN-to-ATM EVC for VLAN-to-VLAN): Part 2 of 2



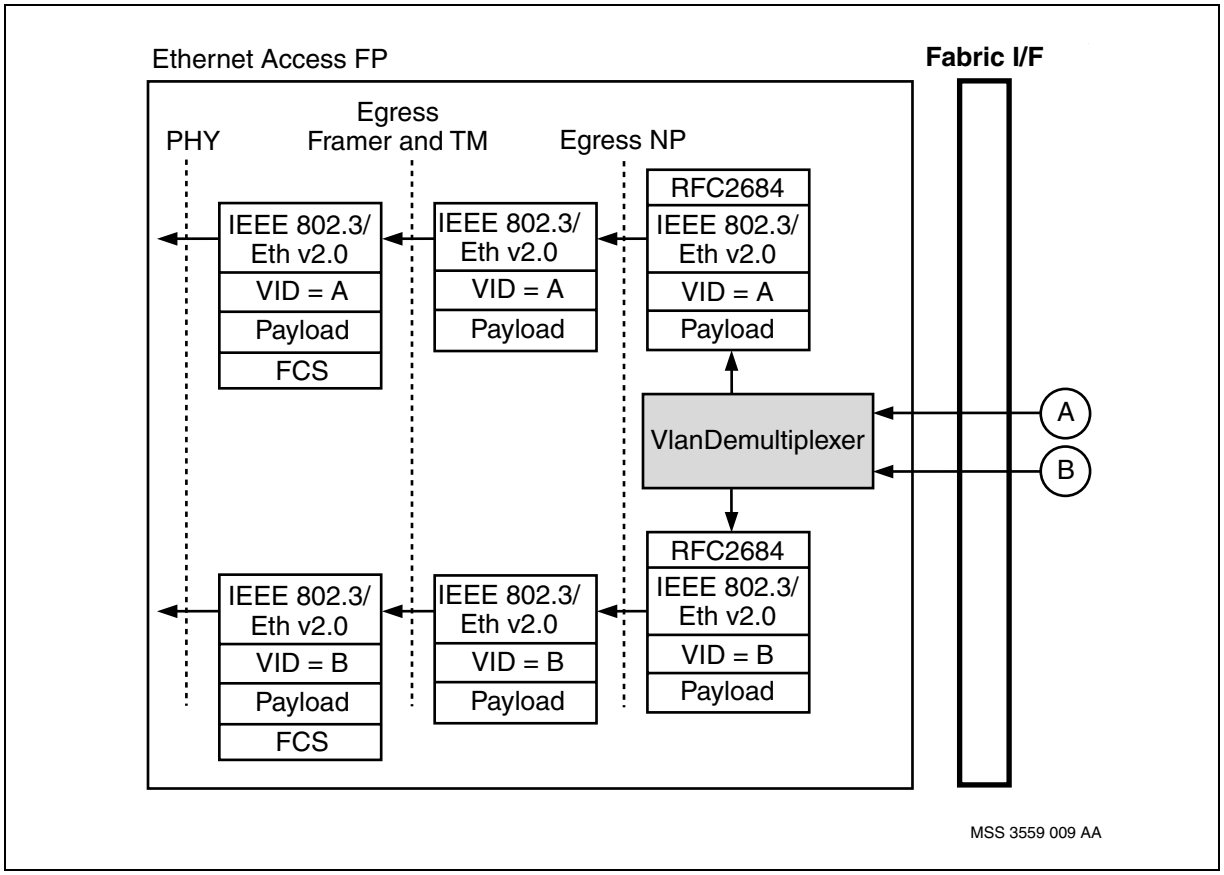
N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (ATM-to-Ethernet direction)

The ATM transport VCC on the ATM FP is configured to forward AAL5 reassembled frames to the Ethernet FP. Received cells are processed by the AAL5 reassembly function block. Assembled frames are forwarded to the egress network processor (NP) on the Ethernet FP. The NP uses the EVLS VLAN demultiplexing table to identify and forward VLAN traffic to the corresponding VLAN EVC on the Ethernet interface.

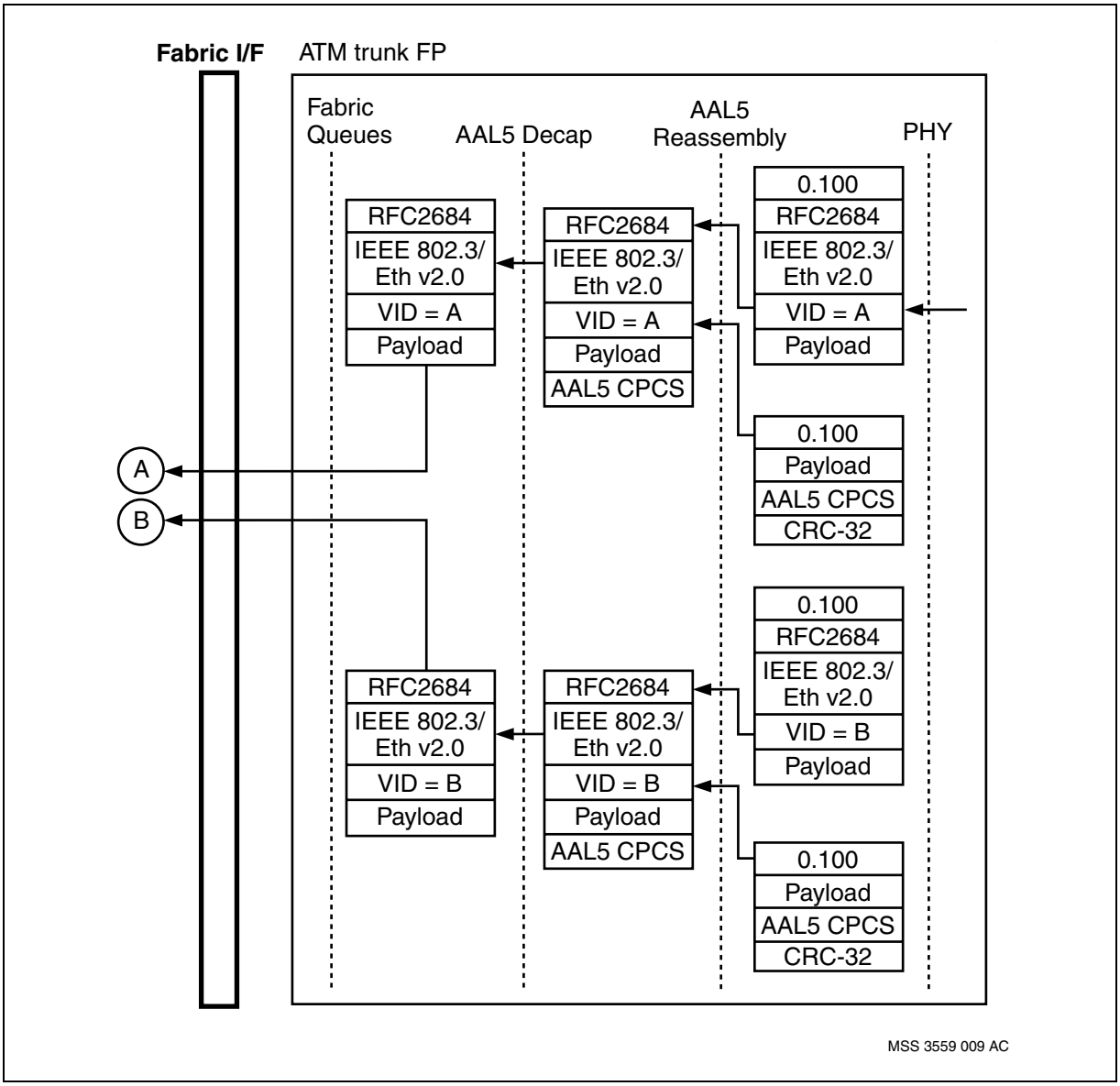
The egress FP removes the RFC2684 encapsulation and forwards the resulting packet to the Ethernet interface. The egress PHY device recalculates and adds the FCS to the packet before transmitting the frame on the port.

See ATM-to-Ethernet direction frame flow (N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport): Part 1 of 2 (page 171) for more information.

ATM-to-Ethernet direction frame flow (N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport): Part 1 of 2



ATM-to-Ethernet direction frame flow (N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport): Part 2 of 2



Packet modification and RFC2684 encapsulation

When an Ethernet frame is received from an Ethernet interface, the FCS is validated and stripped. The remainder of the frame may be further manipulated by EVLS based on the frame format and the Ethernet interface operating mode. EVLS uses RFC2684 LLC encapsulation for bridged PDUs to encapsulate the packet. The RFC2684 encapsulated packet is then forwarded to the ATM trunk FP for AAL5 transport. Each of these stages is described here.

RFC2684 describes two encapsulation methods for carrying network interconnect traffic over AAL5 ATM VCCs. Only one of these two methods is currently supported. This method is needed when more than one protocol might be carried over the same VC. With this method, the receiver requires that each AAL5 CPCS PDU explicitly identify the protocol of the routed or bridged PDU. This method is called LLC encapsulation and is the supported encapsulation method used by EVLS in this release. LLC encapsulation specifies both routed and bridged PDU formats. EVLS uses the bridged PDU format exclusively. See LLC encapsulation for bridged protocols (page 173) for more information on the supported encapsulation method.

LLC encapsulation for bridged protocols

EVLS uses LLC encapsulation for bridged PDUs which requires that the SNAP header follow the LLC header. In the table Frame format for bridged Ethernet/802.3 PDUs without FCS (page 173), the LLC header is encoded as 0xAA-AA-03. The OUI value in the SNAP header is encoded as 0x00-80-C2 to represent the IEEE 802.1 organization code for MAC bridged PDUs. EVLS transports IEEE 802.3/Ethernet frames without FCS. As such, the type of the bridged media specified by the two octet PID in the SNAP header is encoded as 0x00-07. See Frame format for bridged Ethernet/802.3 PDUs without FCS (page 173).

Frame format for bridged Ethernet/802.3 PDUs without FCS

Field	Encoding
LLC	0xAA-AA-03
OUI	0x00-80-C2
PID	0x00-07
PAD	0x00-00
MAC destination address	MAC DA (48 bits)
(remainder of MAC frame)	

EVLS solution for end-to-end EVC

The Ethernet virtual line service (EVLS) solution for end-to-end Ethernet virtual connection (EVC), formerly known as the direct connection solution, establishes an EVC between a pair of Ethernet interfaces. This solution originates and terminates the EVC on an Ethernet interface.

- What is the EVLS solution for end-to-end EVC? (page 174)
- Benefits of the EVLS solution for end-to-end EVC (page 174)
- Configurations of the EVLS solution for end-to-end EVC (page 175)

What is the EVLS solution for end-to-end EVC?

The EVLS solution for end-to-end EVC establishes an EVC using an ATM soft permanent virtual circuit (SPVC) VCC. The EVC originates and terminates on an Ethernet interface. The following configurations are supported for this solution:

- LAN-to-LAN connectivity (page 175)
- LAN-to-VLAN connectivity (page 176)
- 1:1 VLAN-to-VLAN connectivity (page 177)
- N:1 VLAN-to-VLAN connectivity (page 178)

Benefits of the EVLS solution for end-to-end EVC

The EVLS solution for end-to-end EVC offers the following benefits:

1. Ethernet virtual line service leverages existing ATM network infrastructure to cost effectively transport Ethernet traffic between a pair of Ethernet ports and a pair of Ethernet VLANs on Ethernet interfaces.
2. EVLS enables customer devices linked to this service to view the core network as a single point-to-point Ethernet wire.
3. EVLS provides the ability to offer many users on a single Ethernet interface connectivity to another Ethernet interface, via VLANs, across an ATM network. This enables the optimization of Ethernet resource utilization. Scalability of the ATM network is enhanced by enabling the aggregation of multiple VLANs at the access across the ATM network using shared VCCs.

4 Encapsulation of Ethernet frames using RFC2684 provides flexibility to interoperate with third party equipment at the remote access point of the ATM network.

For information on configuring the EVLS solution for end-to-end EVC, see EVLS configuration (page 15).

Configurations of the EVLS solution for end-to-end EVC

The following subsections describe three possible configurations available for the end-to-end EVC.

LAN-to-LAN connectivity

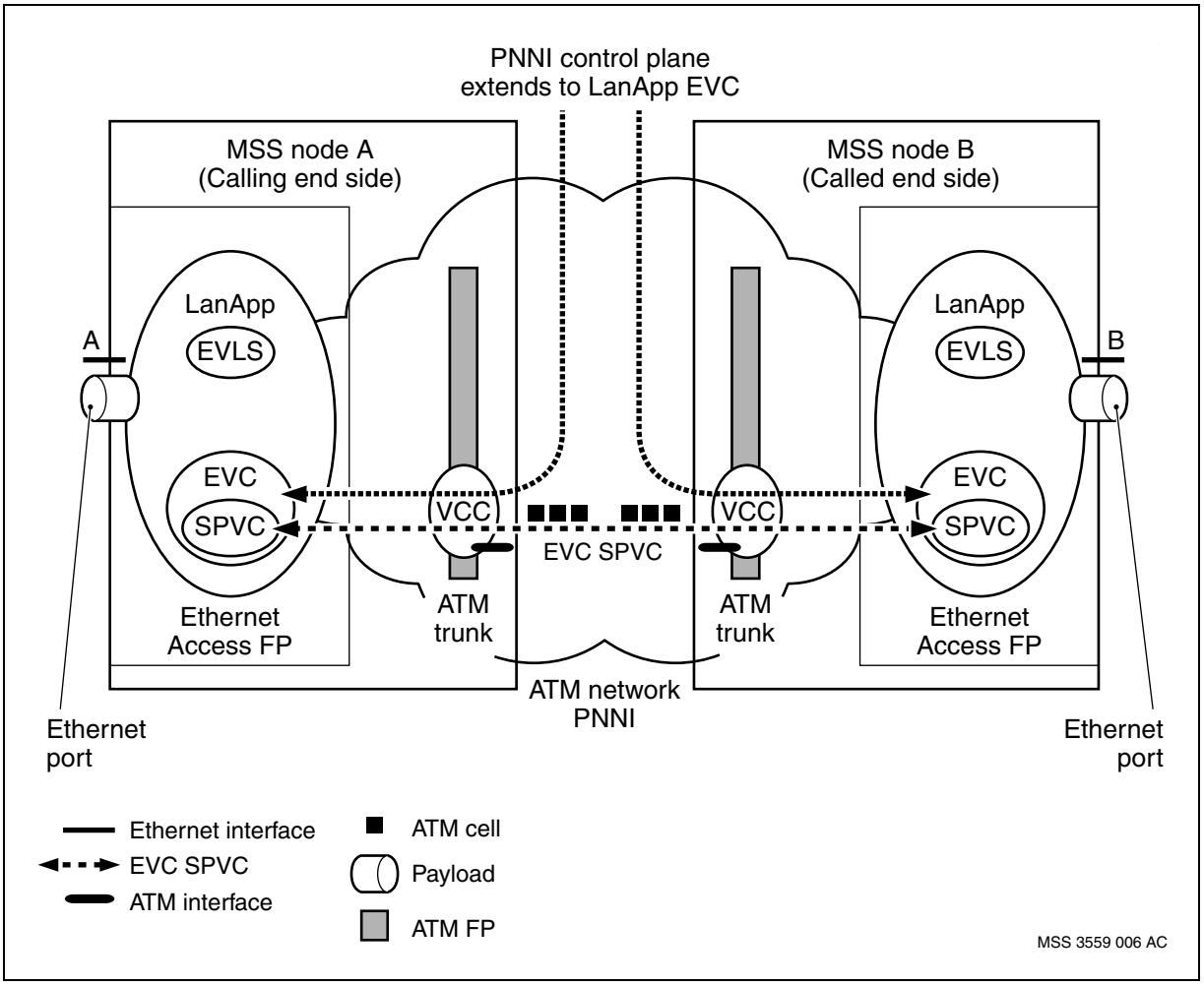
EVLS LAN-to-LAN connectivity establishes an EVC using a single ATM SPVC VCC, referred to as an EVC SPVC. This EVC SPVC is configured to originate on an Ethernet interface operating in port-mode and terminate on another Ethernet interface also operating in port-mode.

The maximum number of EVC SPVCs corresponds to the number of ports on the Ethernet FP. This implies that each EVC instance can only support one ATM service category because each EVC can only support one SPVC.

The *LanApp Evc SrcEvc remoteAddress* attribute is used to specify the EVC called end, at the configured EVC calling end. The *remoteAddress* attribute specifies the far-end Ethernet interface.

See the figure LAN-to-LAN connectivity (page 176).

LAN-to-LAN connectivity



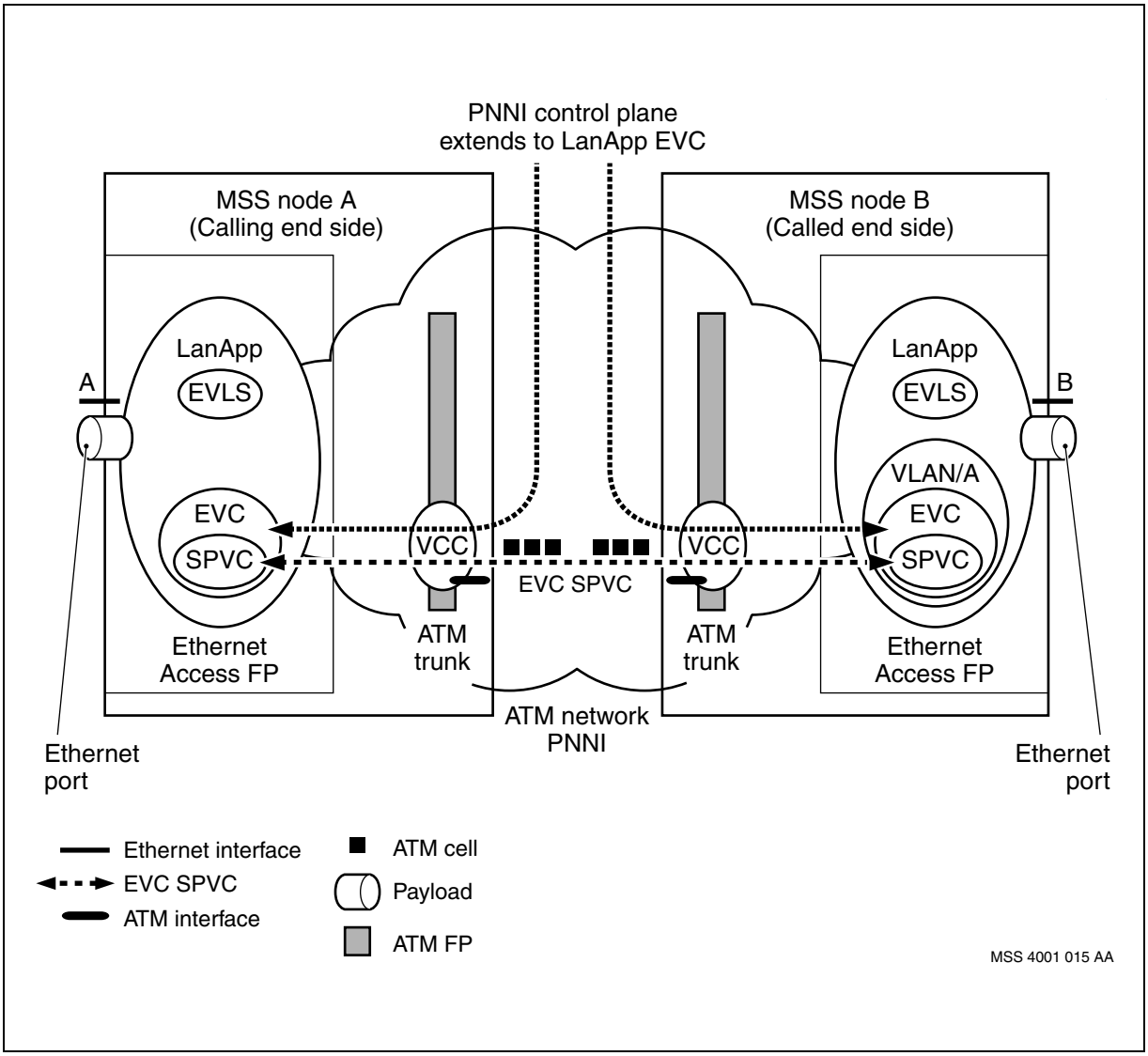
LAN-to-VLAN connectivity

EVLS LAN-to-VLAN connectivity establishes an EVC using a single ATM SPVC VCC, referred to as an EVC SPVC. This EVC SPVC is configured to either:

- originate on an Ethernet interface operating in port-mode and terminate on another Ethernet interface operating in VLAN-mode
- originate on an Ethernet interface operating in VLAN-mode and terminate on an Ethernet interface operating in port-mode

See the figure LAN-to-VLAN connectivity (page 177).

LAN-to-VLAN connectivity

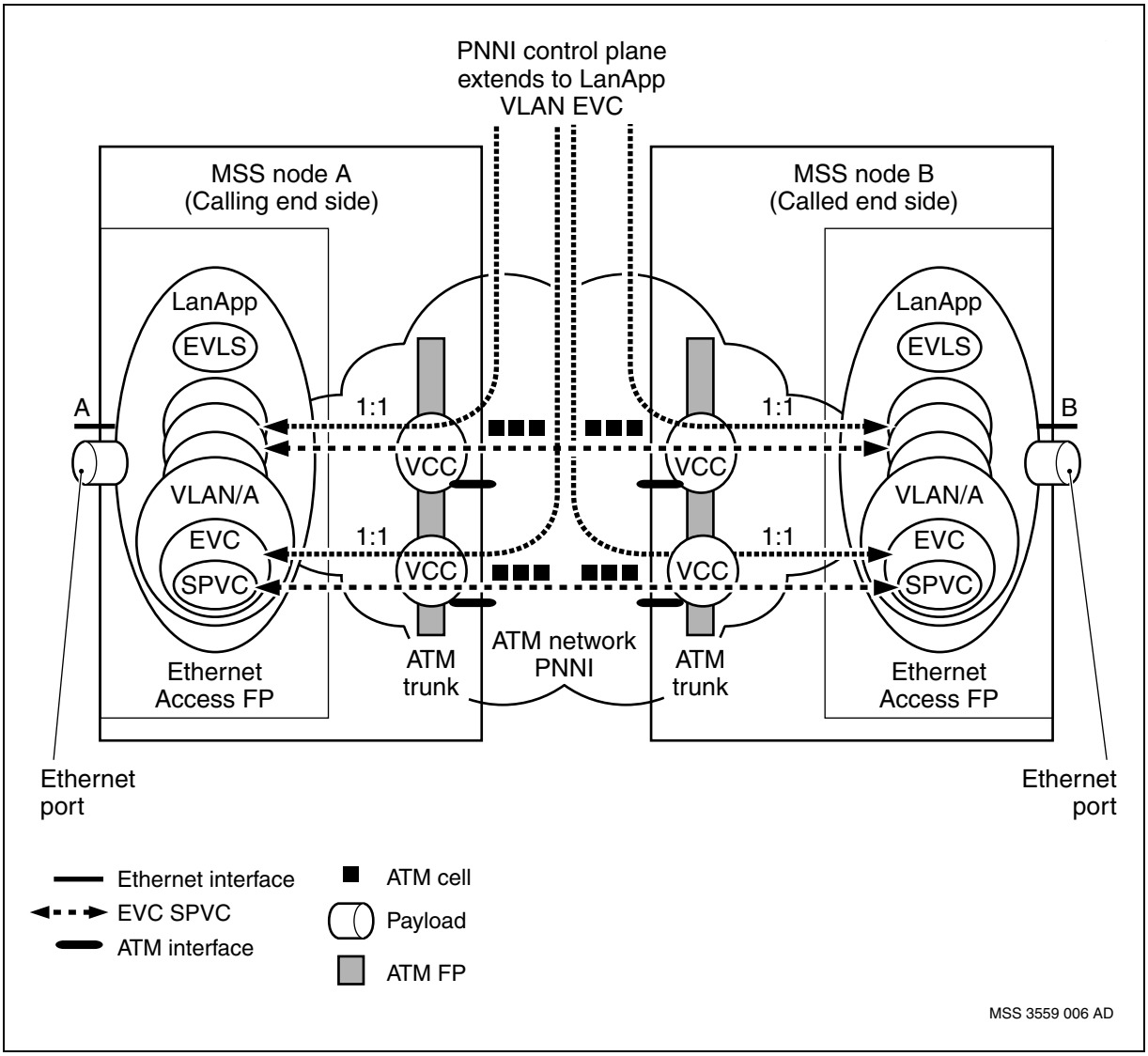


1:1 VLAN-to-VLAN connectivity

An Ethernet virtual connection (EVC) SPVC can be configured on each VLAN. Each EVC instance supports exactly one ATM service category because it is established using exactly one ATM SPVC. The *LanApp Vlan Evc SrcEvc* component is used to specify the EVC called end, at the configured EVC calling end. The *remoteAddress* attribute specifies the far-end Ethernet interface. The *remoteVlanId* attribute specifies the LAN (when set to the port-mode value of 1) or VLAN identifier at the far-end Ethernet interface that terminates the EVC. For the 1:1 VLAN-to-VLAN connectivity option, the *remoteVlanId* attribute must be set to the value of the far-end VLAN identifier.

See the figure 1:1 VLAN-to-VLAN connectivity (page 178).

1:1 VLAN-to-VLAN connectivity



N:1 VLAN-to-VLAN connectivity

The Ethernet transport system (ETS) provides aggregation of traffic from multiple VLANs on a single Ethernet interface across a single ATM VCC. The figure N:1 VLAN-to-VLAN connectivity (page 180) illustrates a scenario in which traffic from multiple VLANs on a single Ethernet interface is aggregated by ETS onto a single VCC. All traffic that is aggregated onto a single ETS SPVC originates from the same Ethernet interface.

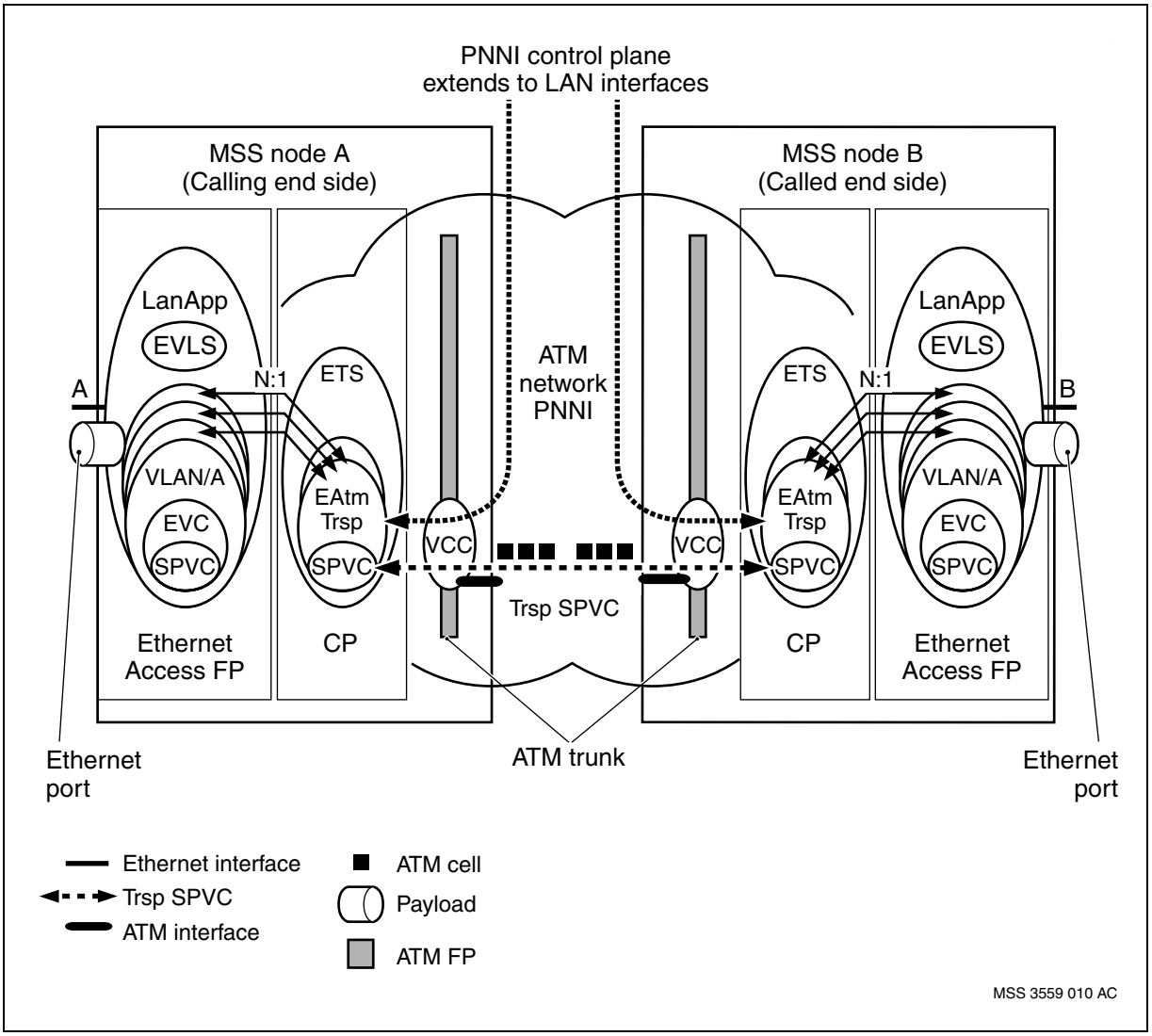
Each *EvlsAtmTransport* instance supports exactly one ATM service category because it is established using exactly one SPVC. Each *EvlsAtmTransport* instance is linked to exactly one Ethernet interface, represented by a *LanApplication* component instance, through the *EvlsAtmTransport*

lanApplication attribute. This attribute defines which set of VLANs on a single Ethernet interface can use the *EvlsAtmTransport Spvc* as transport. The *EvlsAtmTransport SrcEAtmTransport remoteAddress* and *remoteAtmEvcId* attributes are used to define the called end of a configured calling SPVC. The *remoteAddress* attribute defines the ETS (node) for which the SPVC is destined and the *remoteEvlsAtmTransport* attribute defines the *EvlsAtmTransport* instance the dynamic called SPVC is created under.

VLANs are configured to use aggregation by setting the *LanApplication Vlan Evc aggregation* attribute to enabled. When aggregation is enabled for a VLAN, that VLAN solicits the ETS for a transport VC to carry its traffic to the configured destination with the specified Ethernet quality of service (QoS). If a transport VC is found, the VLAN is successfully linked to that transport VC. A match is found when the EVC originates from a LanApplication that matches the *lanApplication* linked to the *EvlsAtmTransport*, and the *remoteAddress* of the EVC matches the *prefixNsapAddress* of the *EvlsAtmTransport* exactly. Each *EvlsAtmTransport* instance has a unique *lanApplication-prefixNsapAddress* pair, the VLAN search for transport provides at most one match. Even if there is no enabled SPVC, the VLAN is linked to the *EvlsAtmTransport* instance if a match is found. For ETS network engineering guidelines, refer to *Nortel Multiservice Switch 7400/15000/20000 Ethernet and EVLS Engineering Guidelines*.

See the figure N:1 VLAN-to-VLAN connectivity (page 180).

N:1 VLAN-to-VLAN connectivity



EVLS solution for single-ended EVC

The EVLS solution for single-ended EVC allows the transport of Ethernet frames between an Ethernet interface and an ATM UNI. This solution is achieved by configuring the Ethernet virtual connection (EVC) SPVC to terminate on an ATM interface (AtmIf) instead of an Ethernet interface (LanApplication). This solution was formerly known as the hairpin solution.

- What is the EVLS solution for single-ended EVC? (page 181)
- Benefits of the EVLS solution for single-ended EVC (page 182)
- Configurations of the EVLS solution for single-ended EVC (page 183)

What is the EVLS solution for single-ended EVC?

The EVLS solution for single-ended EVC establishes an Ethernet virtual connection (EVC) using an ATM soft permanent virtual circuit (SPVC) VCC. The EVC is configured to terminate on a tandem ATM interface. The following configurations are supported for this solution:

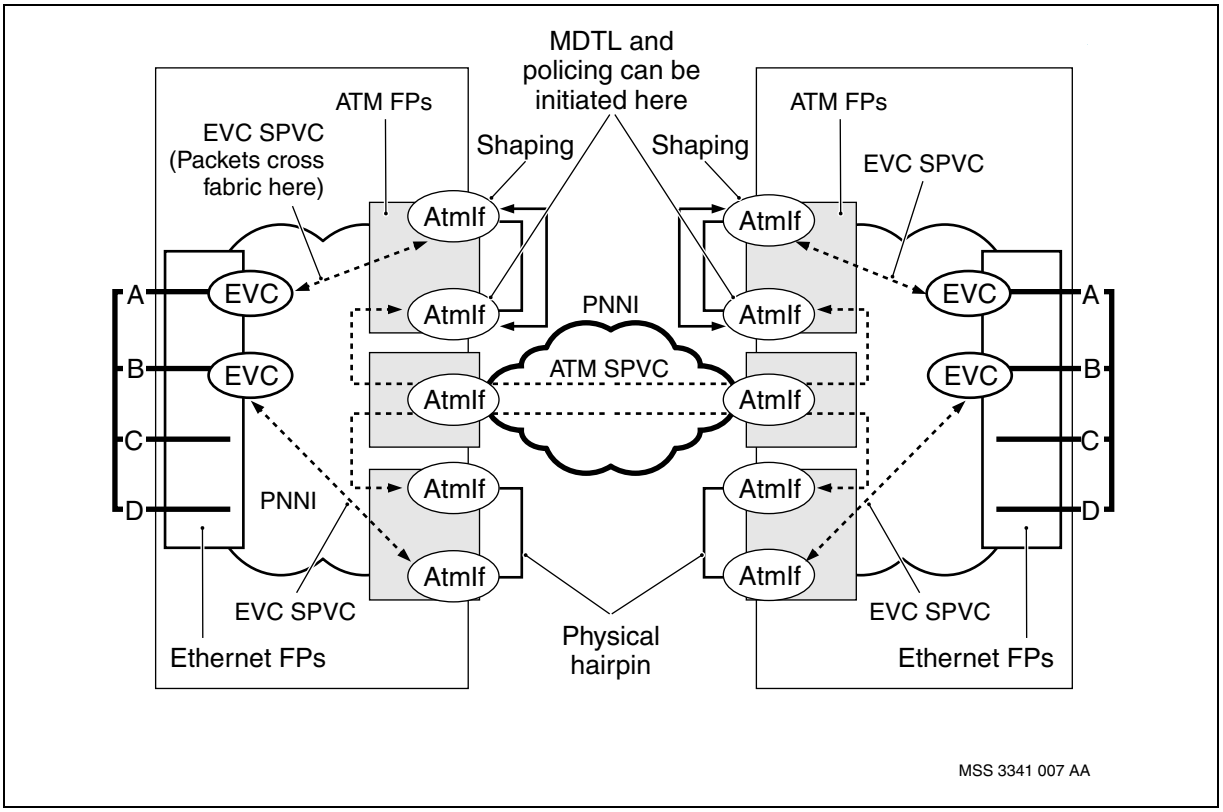
- LAN-to-ATM EVC (page 183)
- 1:1 VLAN-to-ATM EVC (page 184)
- N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (page 185)

Performance enhancement solution

A performance enhancement solution can be achieved by using the single-ended EVLS solution. The EVC must terminate on an ATM interface that is hairpinned to a second ATM interface. Both sides of the hairpinned interface must be configured as ATM UNI interfaces. With this configuration, MDTL, EBR, and policing of Ethernet traffic based on ATM cell rates are supported at the hairpinned ATM interface.

See the figure EVLS performance enhancement solution (page 182) for a graphical representation of this solution.

EVLS performance enhancement solution



Failure in an EVC segment or the tandem ATM segment(s) is not relayed between the segments. As a result, traffic accepted by an EVC may be discarded in the network.

Proper network engineering of this solution requires that both configured EVCs ATM SPVC traffic management parameters, and the configured tandem ATM VCC(s) traffic management parameters all match.

Benefits of the EVLS solution for single-ended EVC

The EVLS solution for single-ended EVC offers the following benefits:

1 The AtmIf configuration takes advantage of the policing and shaping features offered by the ATM function processor (FP). The policing and shaping of Ethernet traffic enforces bandwidth subscriptions and protects the integrity of the ATM network.

2 Source and destination ATM SPVC segment endpoints (MDTL) can be selected manually. MDTL can be used to specify the AtmIf-to-AtmIf VCC segment. For more information, see NN10600-702 *Nortel Multiservice Switch 7400/15000/20000 ATM Routing and Signalling Fundamentals*.

3 The maximum throughput for ATM connections is supported.

For information on configuring the EVLS solution for single-ended EVC, see EVLS configuration (page 15).

Configurations of the EVLS solution for single-ended EVC

The following subsections describe three possible configurations available for the EVLS solution for single-ended EVC.

LAN-to-ATM EVC

The LAN-to-ATM EVC establishes an EVC using a single ATM SPVC VCC, referred to as an EVC SPVC. This EVC SPVC is configured to originate on an Ethernet interface operating in port-mode and terminate on an ATM UNI interface. This configuration is used to support either:

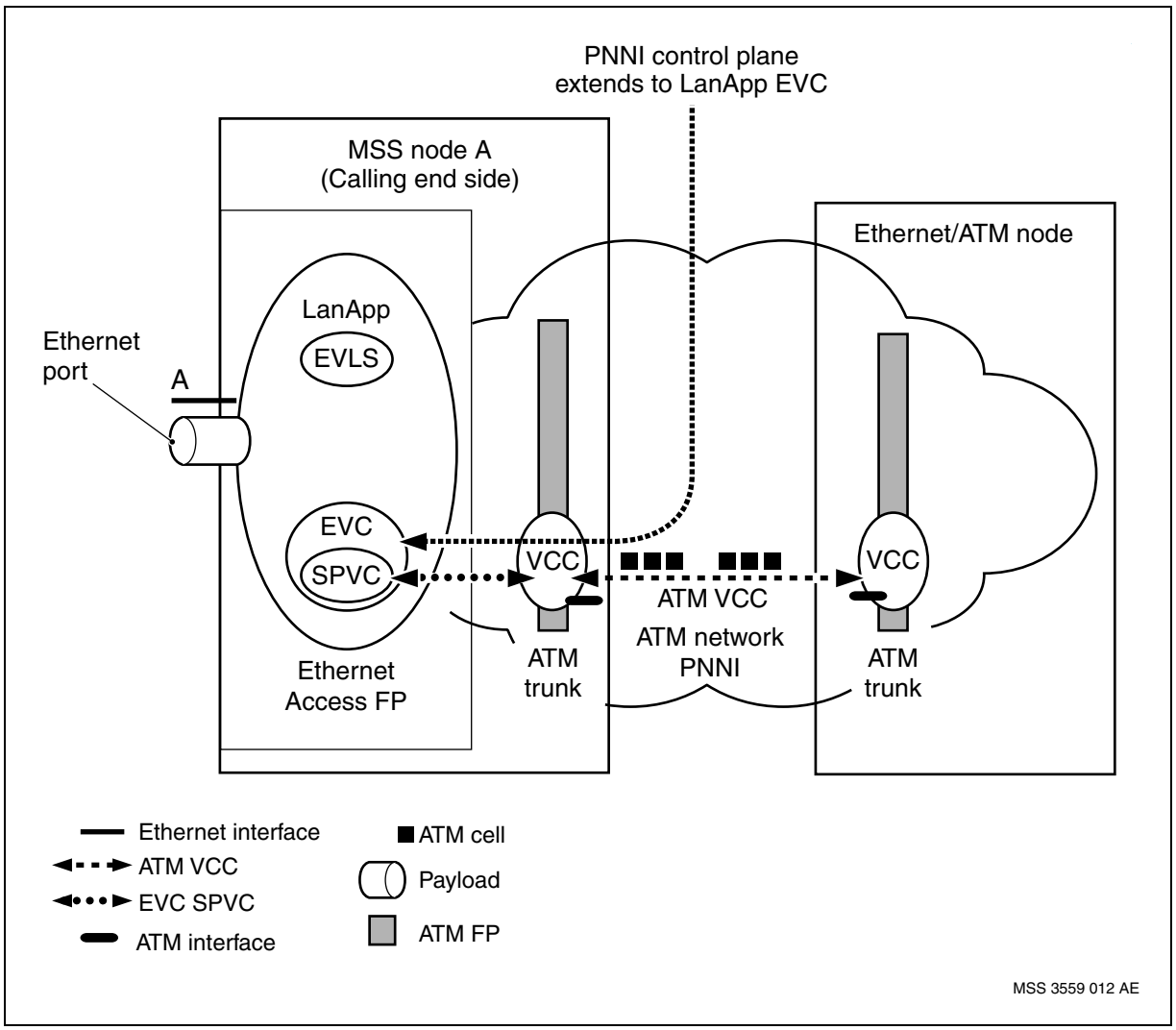
- LAN-to-LAN transport
- LAN-to-VLAN transport

The maximum number of EVC SPVCs corresponds to the number of ports on the Ethernet FP. This implies that each EVC instance can only support one ATM service category because each EVC can only support one SPVC.

The *LanApp Evc SrcEvc remoteAddress* attribute is used to specify the ATM UNI at the far-end, at the configured EVC calling end. The *remoteAddress* attribute specifies the far-end ATM UNI interface.

See the figure LAN-to-ATM EVC (page 184).

LAN-to-ATM EVC



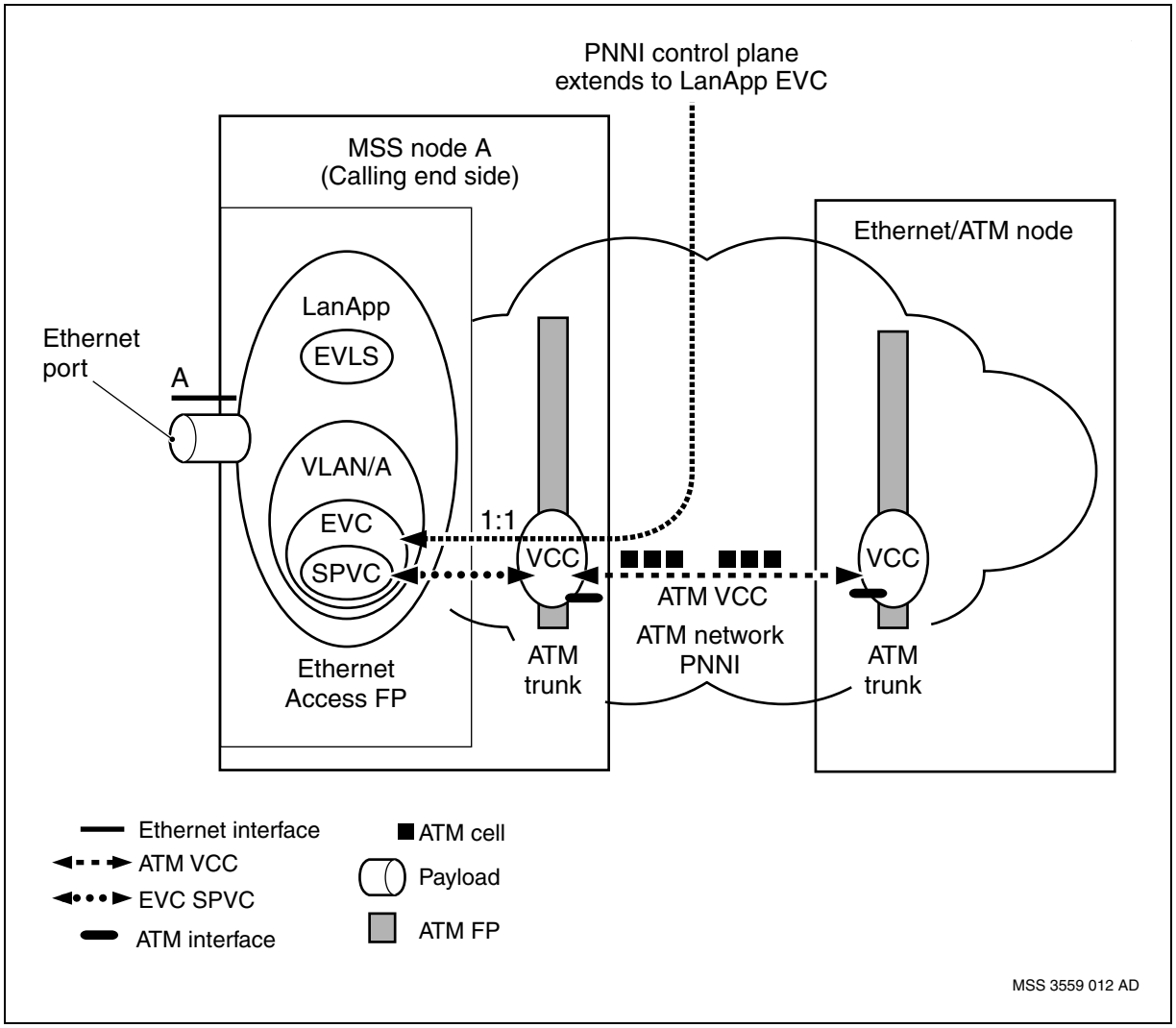
1:1 VLAN-to-ATM EVC

An Ethernet virtual connection (EVC) SPVC can be configured on each VLAN. Each EVC instance supports exactly one ATM service category because it is established using exactly one ATM SPVC. The *LanApp Vlan Evc SrcEvc* component is used to specify the EVC called end, at the configured EVC calling end. The *remoteAddress* attribute specifies the far-end ATM UNI interface. The *remoteVlanId* attribute does not have a default value and therefore needs to be configured. The *calledVpiVci* attribute under the *SrcVcc* component needs to be configured. This configuration is used to support either:

- VLAN-to-VLAN transport
- LAN-to-VLAN transport

See the figure 1:1 VLAN-to-ATM EVC (page 185).

1:1 VLAN-to-ATM EVC



N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport

The Ethernet transport system (ETS) provides aggregation of traffic from multiple VLANs on a single Ethernet interface across a single ATM VCC. The figure N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (page 187) illustrates a scenario in which traffic from multiple VLANs on a single Ethernet interface is aggregated by ETS onto a single VCC. All traffic that is aggregated onto a single ETS SPVC originates from the same Ethernet interface.

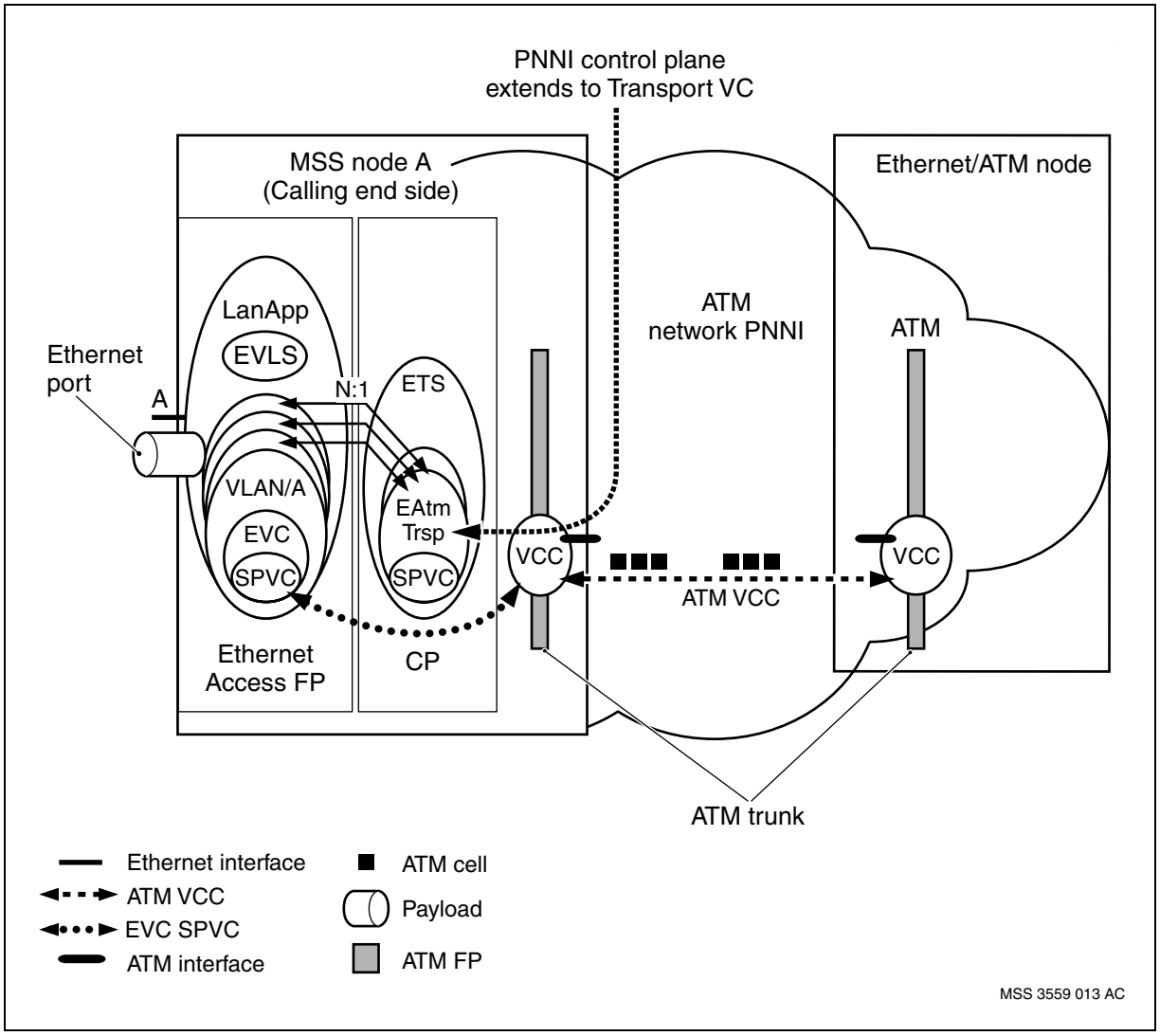
Each *EvlsAtmTransport* instance supports exactly one ATM service category because it is established using exactly one SPVC. Each *EvlsAtmTransport* instance is linked to exactly one Ethernet interface, represented by a *LanApplication* component instance, through the *EvlsAtmTransport*

lanApplication attribute. This attribute defines which set of VLANs on a single Ethernet interface can use the *EvlsAtmTransport Spvc* as transport. The *remoteAddress* attribute is used to define the called end of a configured calling SPVC. The *remoteAddress* attribute defines the remote ATM UNI interface for which the SPVC is destined. The *calledVpiVci* attribute under the *SrcVcc* component needs to be configured.

VLANs are configured to use aggregation by setting the *LanApplication Vlan Evc aggregation* attribute to enabled. When aggregation is enabled for a VLAN, that VLAN solicits the ETS for a transport VC to carry its traffic to the configured destination with the specified Ethernet quality of service (QoS). If a transport VC is found, the VLAN is successfully linked to that transport VC. A match is found when the EVC originates from a LanApplication that matches the *lanApplication* linked to the *EvlsAtmTransport*, and the *remoteAddress* of the EVC matches the *prefixNsapAddress* of the *EvlsAtmTransport* exactly. Each *EvlsAtmTransport* instance has a unique *lanApplication-prefixNsapAddress* pair, the VLAN search for transport provides at most one match. Even if there is no enabled SPVC, the VLAN is linked to the *EvlsAtmTransport* instance if a match is found. For ETS network engineering guidelines, refer to *Nortel Multiservice Switch 7400/15000/20000 Ethernet and EVLS Engineering Guidelines*.

See the figure N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport (page 187).

N:1 VLAN-to-ATM EVC for VLAN-to-VLAN transport



EVLS traffic management

This section describes the traffic management features of the Ethernet virtual line service (EVLS) solution.

Navigation

- Traffic management overview (page 188)
- Ethernet to ATM bandwidth parameters mapping (page 190)
- EVLS policing (page 196)
- Traffic management functions (page 208)

Traffic management overview

Traffic management (TM) functions enable the service provider to offer differentiated services with service level agreements (SLAs), manage network resources according to the provider policy, and perform effective congestion control functions. They also promote fairness by protecting conformant and well-behaved traffic from aggressive users sending traffic in excess of their contract.

The EVLS edge traffic management is based on the industry standard Differentiated Services model. It contains a rich set of functions, including traffic classification, policing, QoS mapping, egress queuing, scheduling, and congestion control. The traffic management functions are implemented at both the ingress and egress EVLS nodes, and on both the Ethernet and ATM FPs. The EVLS solution also takes advantage of all existing Nortel Multiservice Switch ATM networking and traffic management features at the core.

Traffic management features

Traffic management is derived primarily from the Ethernet virtual connection (EVC) configuration. An EVC can carry traffic from a single VLAN or port. An EVC is configured with one of five classes which determine the quality of service treatment of the EVC frames. Traffic management parameters align with the traffic management standard created by the Metro Ethernet Forum (MEF.10).

Single-ended configuration of ATM SPCVs in non-aggregated mode allows the use of standard ATM traffic management functions on the ATM side of the connection. Performance enhancement option such as hairpin also enables additional ATM traffic management functions such as traffic shaping and policing of VCC at the ATM layer.

In addition to traffic shaping and policing, congestion management and network engineering are performed by the ATM FP. For information about traffic shaping or policing, see NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*. For information about ATM queuing, see NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*.

Ethernet traffic classes

The Ethernet service is assigned to one of the multiple traffic classes, depending on its performance requirements. The traffic class determines the frames priority treatment both at the EVLS edge nodes and in the ATM core network. The traffic class is not explicitly configured. It is indicated to the system by the *EthernetPerHopBehavior* attribute. Examples of supported applications and their traffic classes and PHB assignments are:

- Premium class: Traffic streams of telephony and constant rate services are assigned the *EthernetPerHopBehavior* of EF.
- Gold class: Delay sensitive multimedia applications such as packet voice with speech activity detection and video conferencing are assigned the *EthernetPerHopBehavior* of AF31.
- Silver class: Delay tolerant interactive data applications are assigned the *EthernetPerHopBehavior* AF21.
- Bronze class: High throughput data are assigned the *EthernetPerHopBehavior* of AF11
- Standard class: Best effort data applications are assigned the *EthernetPerHopBehavior* of DF

The traffic class provides a convenient method to characterize the traffic performance requirements, and allows for future flexibility when multiple PHBs may be normalized to the same traffic class. The supported Ethernet traffic, classes are consistent with the Multiservice Switch IP differentiated services (DiffServ) classes.

Network engineering

Proper network engineering is critical in ensuring optimal use of network resources while minimizing undesirable frame discards by the network. The offered load by an Ethernet FP can exceed the capacity of an ATM FP. It is

essential to engineer the ingress Ethernet traffic in such a way that it maximizes the efficiency of the ATM FPs, while minimizing Ethernet discards by the Ethernet FP or the ATM FP due to congestion.

Ethernet to ATM bandwidth parameters mapping

The Ethernet and ATM services use different standard traffic parameters to characterize connection. In addition, the EVLS solution encapsulation introduces overhead, which must be taken into account while performing parameters mapping between the Ethernet and ATM services.

For non-aggregated EVC, parameters mapping from Ethernet to ATM is performed automatically by the access node, and is used for establishing the ATM VCC connection.

The user needs only to configure the Ethernet service class (ePhb) and average frame size required for the overhead calculations.

For the aggregated EVC, the ATM VCC is configured by the user. In this case, the mappings provided in this section can serve as guidelines for estimating the ATM VCC bandwidth based on the traffic parameters of the EVCs sharing the aggregate.

The conversion from Ethernet parameters to ATM can not be exact because each set of parameters describes a given connection differently. However, one set of parameters can be used to estimate the equivalent of the other set.

Traffic parameter mapping options

The objective is deriving ATM VCC traffic parameters from the configured EVC traffic parameters.

Ethernet EVC uses the following parameters:

- CIR: committed information rate
- EIR: excess information rate
- CBS: committed burst size
- EBS: excess burst size

ATM VCC uses the following parameters:

- PCR: peak cell rate in cells/second
- CDVT: cell delay variation tolerance
- SCR: sustainable cell rate in cells/second
- MBS: maximum burst size in cells
- MDCR: minimal desired cell rate in cells/second

As engineering guidelines, the Ethernet traffic parameters should correspond to the EVC service class as indicated below:

- ETH EF class specifies CIR
- AF31 and AF21, specify CIR, CBS, EIR
- AF11 specifies CIR, EIR
- DF specifies EIR

However, the switch software is tolerant to variations to the above rules, and the above recommendations are not enforced by the system.

A summary of the traffic parameter mapping rules is provided in the table Ethernet to ATM traffic mapping options (page 191). All four ATM service categories are supported. If any of the resulting ATM rates is greater than ATM link rate, it should be capped at ATM link Rate.

The mapped parameters are multiplied by an estimated overhead factor as indicated. For more information about overhead calculations, see Overhead factor (page 192).

Ethernet to ATM traffic mapping options

1. Ethernet PerHopBehavior = EF and service category = CBR
PCR_{0+1} = (CIR + EIR) x overheadFactor(n) / (53 x 8) cells/s
PCR cannot be equal to 0 (CIR > 0 or EIR > 0).
CDVT value is configurable and typically very small (ex: in the order of 1 msec.).
2. Ethernet PerHopBehavior = AF31 and service category = rt-VBR (VBR.3)
PCR_{0+1} = (CIR + EIR) x overheadFactor(n) / (53 x 8) cells/s
CDVT value is configurable.
CIR cannot be equal to 0 (CIR > 0).
SCR_0 = CIR x overheadFactor(n) / (53 x 8) cells/s
MBS_0 = [CBS/(1 - CIR/AR) + 1] x overheadFactor(n) / (53 x 8) cells/s
3. Ethernet PerHopBehavior = AF21 and service category = nrt-VBR (VBR.3)
PCR_{0+1} = (CIR + EIR) x overheadFactor (n) / (53 x 8) cells/s
CDVT value is configurable.
CIR cannot be equal to 0 (CIR > 0).
SCR_0 = CIR x overheadFactor(n) / (53 x 8) cells/s
MBS_0 = [CBS/(1 - CIR/AR) + 1] x overheadFactor(n) / (53 x 8) cells/s
(1 of 2)

Ethernet to ATM traffic mapping options (continued)

4. Ethernet PerHopBehavior = AF11 and service category = UBR+ PCR_{0+1} = (CIR + EIR) x overheadFactor(n) / (53 x 8) cells/s CDVT value is configurable. CIR cannot be equal to 0 (CIR > 0). MDCR = CIR x overheadFactor(n) / (53 x 8) cells/s
5. Ethernet PerHopBehavior = DF and service category = UBR PCR_{0+1} = (CIR + EIR) x overheadFactor(n) / (53 x 8) cells/s CDVT value is configurable. PCR can be equal to 0 (CIR and EIR can be equal to 0).
overheadFactor(n) = ceiling(($n + h'_E + h_{rfc2684} + h_{trailer} - h_{FCS}$) / 48) x 53 / ($n + h'_E$) where: ceiling(y) gives the number y that is round off to the next largest interger ($n + h'_E$) represent the average frame size CIR and EIR are in bit/s CBS is in bits
(2 of 2)

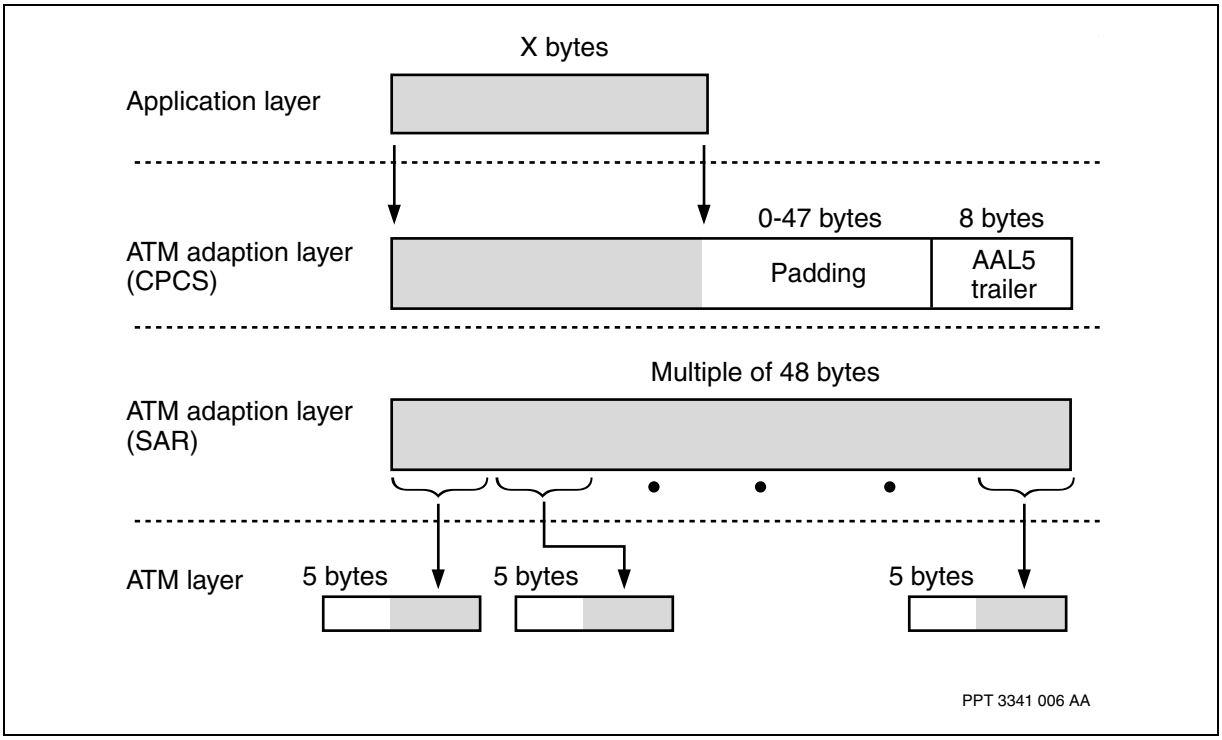
Overhead factor

Due to the overhead difference between Ethernet and ATM, an overhead factor is necessary when converting from one set of traffic parameters to the other. The encapsulation of an application layer frame into ATM cells at the ATM layer encounters the following types of overhead:

- ATM Adaptation layer (common part convergence sublayer (CPCS)) overhead is responsible for adding 0 to 47 bytes of padding and an 8 byte trailer to the frame. The amount of padding results in the total length of the CPCS layer frame size to be a multiple of 48.
- ATM Adaptation layer (segmentation and reassembly sublayer (SAR)) is responsible for partitioning the CPCS layer frame into 48 byte payloads and does not add additional overhead.
- ATM layer is responsible for appending a 5 byte ATM header that includes the VPI and VCI fields to the 48 byte payload.

See the figure Overhead encountered for frame-based services using AAL5 (page 193) for more information.

Overhead encountered for frame-based services using AAL5



Overhead calculation

Algorithms that determine the overhead incurred by the Ethernet to ATM service interworking function are presented here. The types of overhead considered include the Ethernet overhead and the overhead introduced by the ATM adaptation (AAL5) and ATM layers as shown in table Ethernet to ATM service interworking overhead summary (page 195).

To transport a data unit of length n, IEEE 802.3 (Ethernet) adds a header of length h_E.

Counting preamble, tagging, and FCS, h_E = 30 bytes

The IWF encapsulates the MAC frame starting at the MAC destination address and uses this information as the payload of the AAL5 CPCS-PDU. As part of the MAC overhead, h'_E has to be counted towards counting the overall overhead.

- h'_E = 22 bytes if the MAC FCS is preserved (h_{FCS} is 4 bytes)
- $h'_E - h_{FCS}$ = 18 bytes if the MAC FCS is not preserved.

AAL5 adds two types of overhead:

- AAL5 trailer, $h_{trailer}$ = 8 bytes
- AAL5 SAR overhead, h_{SAR} = 0 to 47 bytes

An additional overhead due to encapsulating of higher layer protocols associated with LLC encapsulation: Counting RFC2684 bridged mode Vc encapsulation, $h_{rfc2684}$ = 10 bytes (8 + 2 "Padding").

Attention: Required to have the AAL5 CPCS-PDU length be an integer with multiples of 48 bytes.

Number of cells needed to transport n bytes of information:

- N_{cells} = Ceiling (($n + h_{rfc2684} + h'_E - h_{FCS} + ht_{railer}$) / 48)

Ceiling (x) is equal to the smallest integer that is greater or equal to x.

Attention: Cell size, excluding overhead, is equal to 48 bytes.

Attention: Assuming the average frame size ($n + h'_E$), including h_{MAC} (12 bytes), $h_{MACLength/Type}$ (2 bytes), h_{VLAN} (4 bytes) if included and the h_{FCS} (4 bytes).

A correction factor is needed when rates are mapped between Ethernet and ATM. Assuming a data unit of length n, the correction factor is overheadFactor = N_{cells} x 53 / ($n + h'_E$).

In the Ethernet to ATM direction, multiply Ethernet rate values by the overheadFactor. For example, SCR = overheadFactor x CIR (note that SCR is in cells/sec, while CIR is in bytes/sec). This analysis does not include the additional overhead due to encapsulating of higher layer protocols, for example IP. No overhead is associated with the VC multiplexing encapsulation.

The ATM overhead introduced by the ATM adaptation layer accounts for the 8 byte AAL5 trailer and the AAL5 padding bytes which are dependent upon the frame size. As well, the ATM overhead also accounts for the 5 byte ATM cell header.

If these parameters are going to be used for ATM UPC traffic policing, the average frame size attribute should be set to the maximum value.

A summary of equations used to account for the overhead is provided below:
overheadFactor (n) = ceiling (n) x 53 / ($n + h'_E$)
where: ceiling (n) gives the number n rounded off to the next larger integer
ceiling (n) = (($n + h'_E + h_{rfc2684} + h_{trailer} - h_{FCS}$) / 48) cells

An overhead summary for selected data units based on the equations presented above is shown in table Ethernet to ATM service interworking overhead summary (page 195).

Ethernet to ATM service interworking overhead summary

Average frame size (bytes)	User Data "n" (bytes)	Number of cells (cells)	Total ATM bytes to transfer DU	Total Ethernet bytes to transfer DU	Overhead factor (n) (cells/byte)
64	42	2	106	64	1.66
128	106	3	159	128	1.24
256	234	6	318	256	1.24
512	490	11	583	512	1.14
1522	1500	32	1696	1522	1.11

Example of conversion from Ethernet to ATM traffic parameters

For this example, an Ethernet connection is carried over an ATM connection. The Ethernet traffic parameters are:

- Average frame size is 512 bytes
- Ethernet PHB is AF21
- With a CIR of 10000 Kbps, EIR of 10000 Kbps, and CBS of 800000 bits, the overhead factor is 1.14.

The computed ATM traffic parameters are:

- Service category is nrt-VBR
- PCR_{0+1} is 53711 cells/s
- SCR is 26856 cells/s
- MBS is 2171 cells

A summary of the traffic parameter conversion for some mapping options is presented in the Tables, Computed ATM traffic descriptor parameters on Multiservice Switch 15000/20000 node (page 196) and Computed ATM traffic descriptor parameters on Multiservice Switch 7400 node (page 196).

Computed ATM traffic descriptor parameters on Multiservice Switch 15000/20000 node

Average frame size (bytes)	CIR (Kbps)	CBS (bits)	EIR (Kbps)	PCR_{0+1} (cells/s)	SCR (cells/s) (VBR only)	MBS_0 (cells) (VBR only)	MDCR (cells/s) UBR+ only)
64	10000	800000	10000	78125	39063	3157	39063
128	10000	800000	10000	58594	29297	2368	29297
256	10000	800000	10000	58594	29297	2368	29297
512	10000	800000	10000	53711	26856	2171	26856
1518	10000	800000	10000	52701	26351	2130	26351
1522	10000	800000	10000	52563	26282	2124	26282
1600	10000	800000	10000	53125	26563	2147	26563
2048	10000	800000	10000	52491	26246	2121	26246
4096	10000	800000	10000	52491	26246	2121	26246
9182	10000	800000	10000	52277	26139	2113	26139

Computed ATM traffic descriptor parameters on Multiservice Switch 7400 node

Average frame size (bytes)	CIR (Kbps)	CBS (bits)	EIR (Kbps)	PCR_{0+1} (cells/s)	SCR (cells/s) (VBR only)	MBS_0 (cells) (VBR only)	MDCR (cells/s) UBR+ only)
64	10000	800000	10000	78125	39063	3473	39063
128	10000	800000	10000	58594	29297	2605	29297
256	10000	800000	10000	58594	29297	2605	29297
512	10000	800000	10000	53711	26856	2388	26856
1518	10000	800000	10000	52701	26351	2343	26351

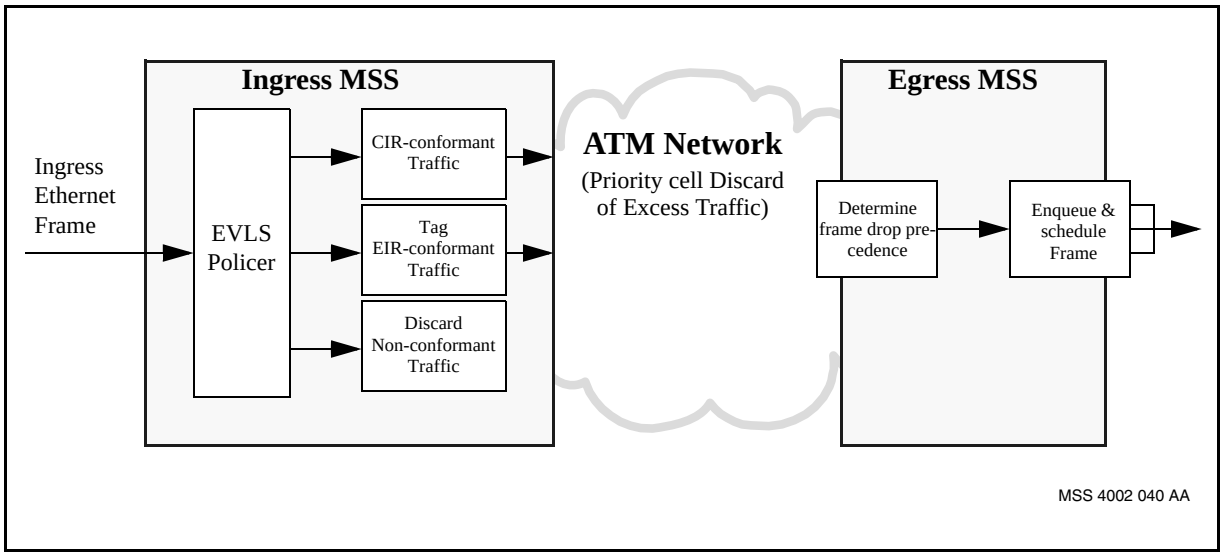
EVLS policing

The policing capability enables the service provider to enforce and monitor a service level agreement (SLA) with the end-user according to the traffic contract. In addition, the policing feature promotes fairness in allocating network resources, and protects well-behaved users from aggressive ones who exceed their traffic contract.

Overview

Using this feature, the network provider can monitor traffic flow entering the network for each Ethernet Virtual Connection (EVC), and either increase the drop precedence or discard frames violating the service agreement. The frames exceeding the committed information rate are optionally tagged and forwarded through the network with no performance guarantees (see figure Multiservice Switch Ethernet policing (page 197)).

Multiservice Switch Ethernet policing



The policing feature supports all Multiservice Switch EVLS configurations, including single VLAN to ATM VCC mapping, Ethernet port to VCC mapping, the aggregated Ethernet Transport Service (ETS) with multiple VLANs sharing an ATM VCC, and single-ended and end-to-end solutions. It is available on both the Multiservice Switch 7400 4pEth100BaseT and 8pEth100BaseT FPs, and on the Multiservice Switch 15000 and 20000 4pGe FP.

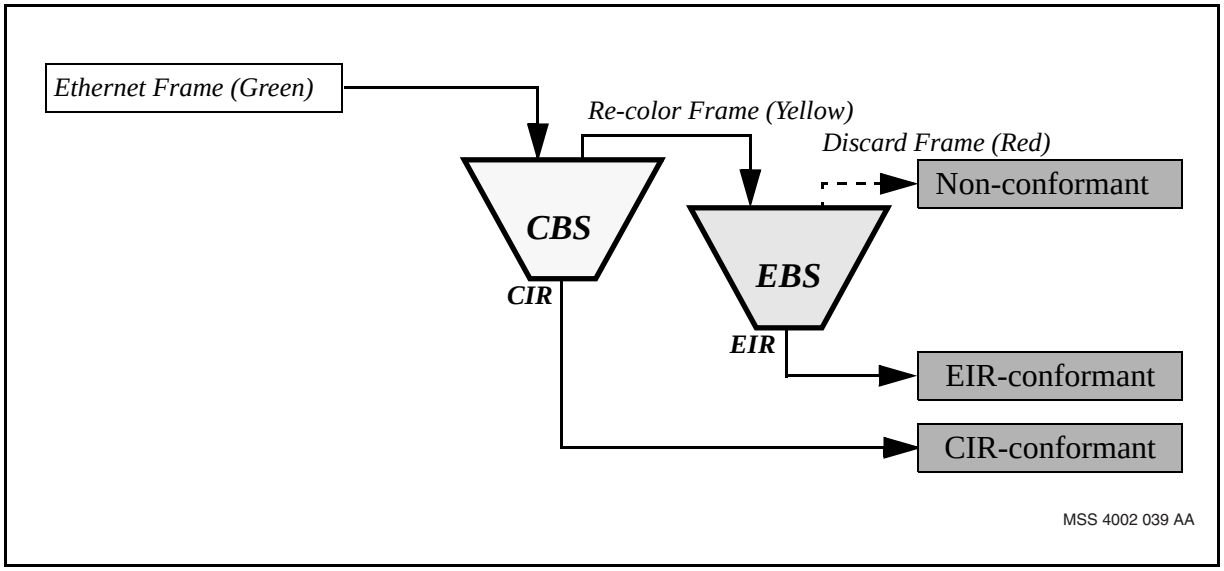
Figure Multiservice Switch Ethernet Dual Rate Traffic Policing (page 198) provides the general example of how the Policing feature can be used to control the amount of traffic entering the network. The Policer uses a dual leaky bucket algorithm, similar to the one implemented for the MSS FR and IP solutions. The configured EVC traffic contract determines the leakage rate and size of each of the two buckets. All incoming Ethernet frames are considered to have the Green color. The Ethernet frame is first evaluated against the Committed Information Rate (CIR) bucket. If the bucket has enough tokens, the frame is accepted as CIR-conformant, and its color remains green. Otherwise, the frame is re-colored Yellow, and evaluated against the Excess Information Rate (EIR) bucket. If accepted as EIR-

conformant, the frame color remains yellow, is tagged with high drop precedence, and forwarded through the network. Frames that exceed the EIR rate, are re-colored red, and are discarded as non-conformant.

Statistics are maintained for incoming frame counts, EIR-conformant frames, and non-conformant discarded frames by the policer. These statistics are important for SLA monitoring, and determining the suitability of the traffic contract for the actual customer traffic.

Similar to the EVC, the policing function is bidirectional. Traffic conformance can be enforced at either or, more commonly, both ends of the connection. Furthermore, the traffic contracts can be asymmetrical, varying by traffic direction.

Multiservice Switch Ethernet Dual Rate Traffic Policing



Benefits

The Multiservice Switch policer provides a large degree of configuration flexibility to suit service provider policies and networking objectives. These include per EVC (port and VLAN) policing, single or double rates enforcement, discard or tagging of non CIR-conformant traffic, a wide range of traffic rates and burst sizes, and independence of policer configuration from the ATM core service.

The service provider benefits in several ways. First, the ability of offering new and enhanced types of Ethernet services with different service level agreements (SLAs). Second, the policer enables the provider to restrict the amount of traffic admitted to the network from specific customers or classes, which results in overall performance improvement and fairness to all network users. The excess traffic can be tagged as low priority and forwarded as best

effort or discarded, at user option. Third, the policer provides detailed statistics, which can be used for monitoring and reporting the SLAs, up-selling the end-user a bigger traffic contract, and for network engineering purposes.

Policer algorithm

Multiservice Switch Ethernet traffic policer implements the Metro Ethernet Forum (MEF) standard algorithm. This algorithm is similar in principle to the well known Frame Relay traffic management algorithm, and the algorithm implemented in the Multiservice Switch IP Policer. This algorithm uses following parameters:

- Committed Information Rate (CIR). This is the average rate (expressed in bits/s) at which the network agrees to transfer information under normal conditions, and meets the performance objectives.
- Committed Burst Size (CBS). The CBS is the maximum amount of data (in bits) available for a burst of ingress Ethernet frames sent at the Ethernet access rate to remain CIR-conformant.
- Excess Information Rate (EIR). This is the rate (expressed in bits/s) up to which the network may deliver Ethernet frames without any performance objectives.
- Excess Burst Size (EBS). The EBS is the maximum amount of data (in bits) available for a burst of ingress Ethernet frames sent at the Ethernet access rate to remain EIR-conformant.

The policing parameters include the frame payload, plus the Ethernet MAC header of 18 bytes for untagged frames, and 22 bytes for IEEE 802.1Q-tagged frames. The Ethernet header consists of the MAC destination address (6 bytes), MAC source address (6 bytes), Protocol Type (2 bytes), and FCS (4 bytes). The optional Q-tag is 4 bytes. The parameters exclude the inter-frame-gap (12 bytes), and the preamble (8 bytes).

This traffic management algorithm utilizes the leaky bucket mechanism. It uses two leaky buckets. The first bucket controls CIR and the second bucket controls EIR. See the figure Multiservice Switch Ethernet Dual Rate Traffic Policing (page 198) for general operation of the policer.

The policer operates in the color-blind mode, meaning that all incoming Ethernet traffic is assumed to be green, and is directed to the CIR bucket. CIR-conformant traffic remains colored green. Non CIR-conformant (i.e., excess) traffic that is EIR-conformant is colored yellow and forwarded through the network on a best effort basis with no performance guarantee. Non-conformant traffic from the EIR bucket is discarded by the policer.

Statistics are maintained per policer for the total amount of ingress traffic received by an EVC, CIR-conformant (green) traffic, EIR-conformant (yellow) traffic, and discarded non-conformant (red) traffic.

The bit count limit for the CIR bucket is set to the Committed Burst Size (CBS) and the bit count limit for the EIR bucket is set to the Excess Burst Size (EBS). The initial bit count for each bucket is set to 0.

When a frame arrives at the CIR bucket and the bucket is not full (the current bit count in the bucket is less then the committed burst size), the frame adds n-bits (n = frame size*8) to the bit count and the frame is forwarded. If the CIR bucket was full when the frame arrived, then the frame is not CIR-conformant, and is assigned the yellow color and directed to the EIR bucket.

When a frame arrives at the EIR bucket and the bucket is not full (the current bit count in the bucket is less then the excess burst size), the frame adds n-bits (n = frame size*8) to the bit count and the frame is forwarded. If the EIR bucket was full when the frame arrived, the frame is discarded.

The buckets are leaking at the CIR and EIR rates respectively. This means that the bit count in the buckets is decremented at the CIR and EIR rate.

The user configured traffic parameters are used to configure the bucket parameters as well as algorithm operation. Depending on the traffic parameters one or both rates can be enforced and different actions can be taken for non CIR-conformant frames.

Similar to the EVC, the policing function is bidirectional. Traffic conformance can be enforced at either or, more commonly, both ends of the connection. Furthermore, the traffic contracts can vary by traffic direction.

Policer configuration

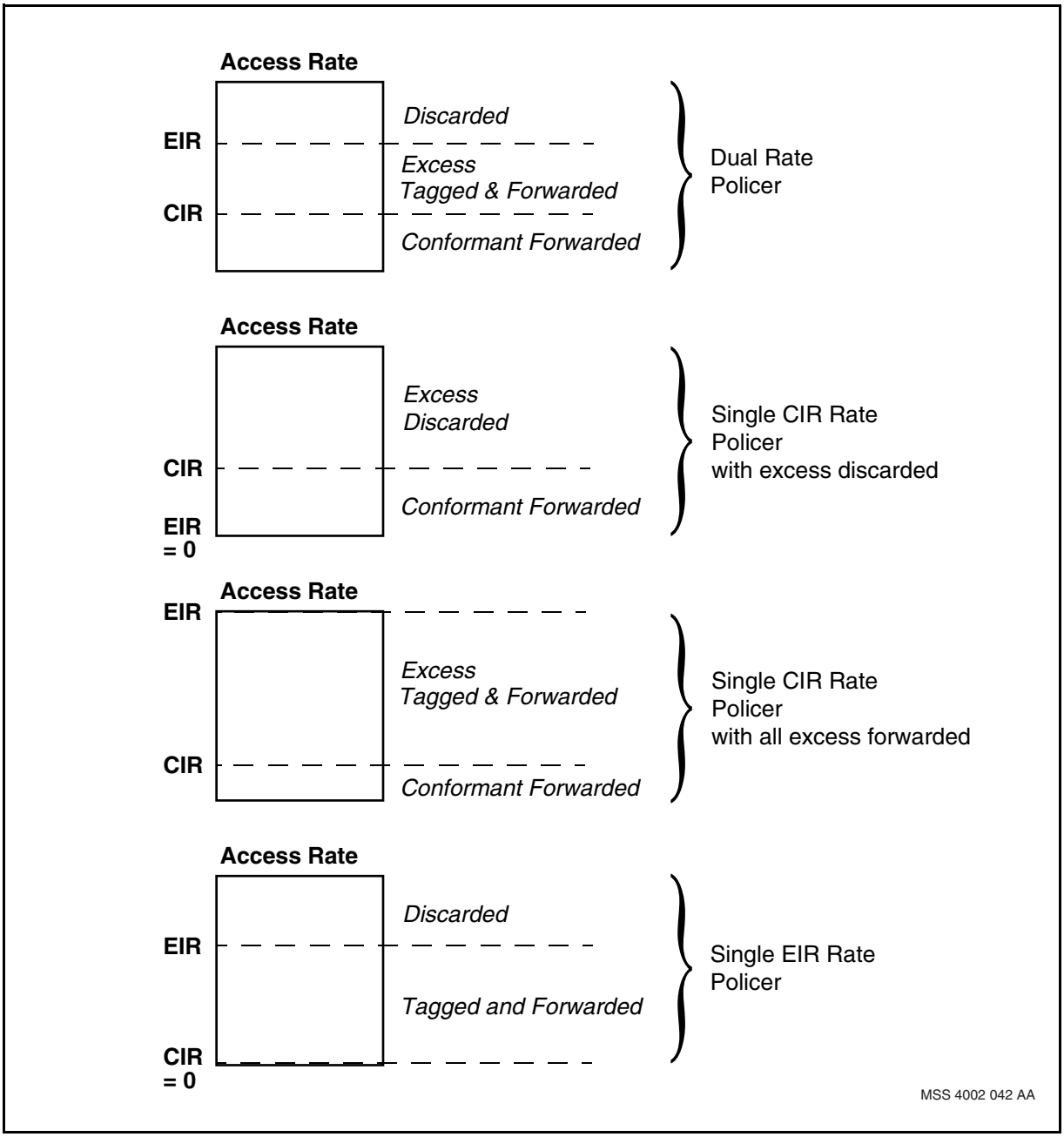
The policer operation is determined by the four configurable traffic parameters CIR, CBS, EIR, and EBS. These parameters are provisioned under the *EvcPolicer* component. The *EvcPolicer* component can be optionally added and enabled at one or both ends of an EVC.

The CIR and EIR buckets are independent, and either bucket can be configured with the special rate and size value of zero, to reject all traffic, or set to line rate to accept all traffic directed at the bucket.

The choice of the bucket parameters enable a large amount of flexibility in tailoring the policer operation to the service provider requirements. For example, traffic may be policed using a single rate or dual rates, and non CIR-conformant traffic may be tagged with low priority or discarded. The figure Policer types (page 201) shows different types of traffic policers that can be configured by adjusting policer parameters.

The ingress traffic policer is enabled per EVC, by provisioning the policing attribute under the *EvcPolicer* component.

Policer types



The policing traffic parameters / options apply to the entire Ethernet Virtual Connection (port or VLAN), and are independent of the connection Ethernet Per Hop Behavior (ePHB). This provides separation between the access and core traffic treatment, and increases network engineering flexibility. This separation is also necessary in the ETS aggregation case, when a single ATM VCC is used for transporting traffic from multiple VLANs with different service classes and policing requirements.

The following default policing behaviors (see table Recommended policing behavior for the different Ethernet PHBs (page 202)) are recommended as good engineering practice for the non-aggregated case, and for the aggregated ETS case when all the constituent VLANs are configured with the same ePhb value.

Recommended policing behavior for the different Ethernet PHBs

ePhb	Traffic class and example	Policing behavior
EF	Premium (telephone)	Single CIR rate with excess discarded
AF31	Gold (multimedia)	Dual rate policer
AF21	Silver (transactions)	Dual rate policer
AF11	Bronze (high throughput data)	Dual rate policer
DF	Standard (best effort)	Single EIR rate

Note that the non CIR-conformant AF11 traffic cannot be tagged for differentiated forwarding treatment, because all traffic is mapped to ATM UBR CLP1. But the dual rates policer configuration is recommended for the AF11 class for the purpose of statistics collection and SLA monitoring. The ATM network typically promises some QoS targets for the traffic within the minimum guaranteed rate (which maps to the ATM UBR+ minimum desired cell rate).

The following sections describe the policer parameters selection to achieve different policing behaviors.

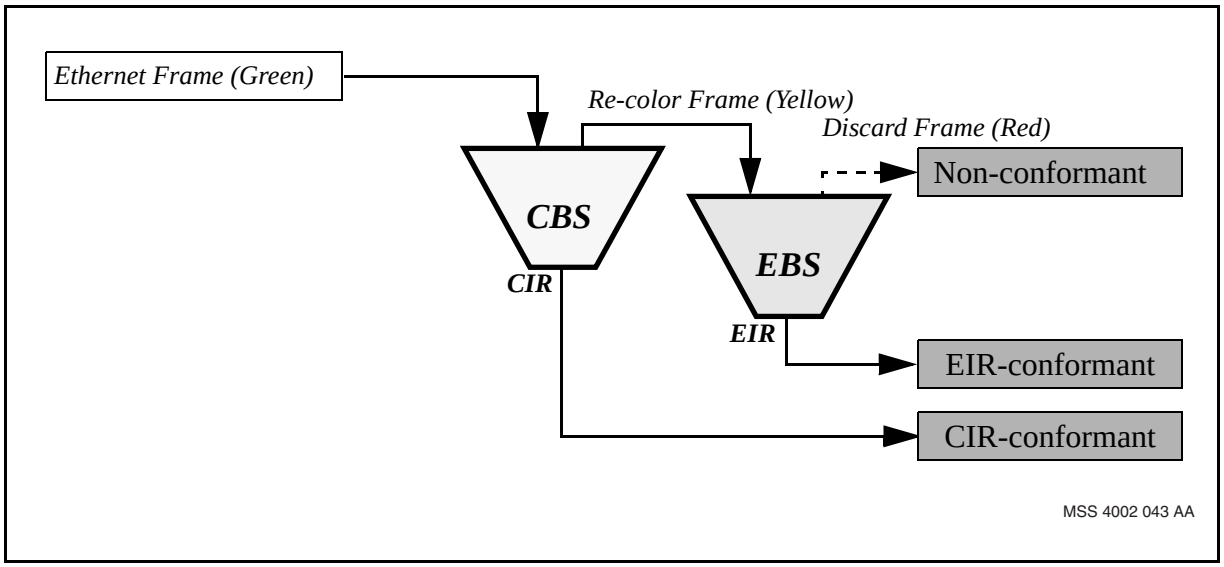
- Dual rate policer (page 202)
- Single CIR rate policer with all excess discarded (page 203)
- Single CIR rate with all excess forwarded policer (page 204)
- Single EIR rate policer (page 205)

Refer to EVLS traffic management configuration (page 45) for the provisioning procedures.

Dual rate policer

The dual rate policer enforces both rates. First frames are directed to the CIR bucket, CIR-conforming traffic is forwarded, excess traffic is tagged and directed to the EIR bucket. EIR-conformant traffic, (for example, traffic accepted by the EIR bucket) is forwarded. Non-conformant traffic, that is traffic in excess of the EIR bucket, is discarded.

Dual rate policer



To achieve the dual rate policer the parameters have to be set according to table Dual rate policer configuration (page 203).

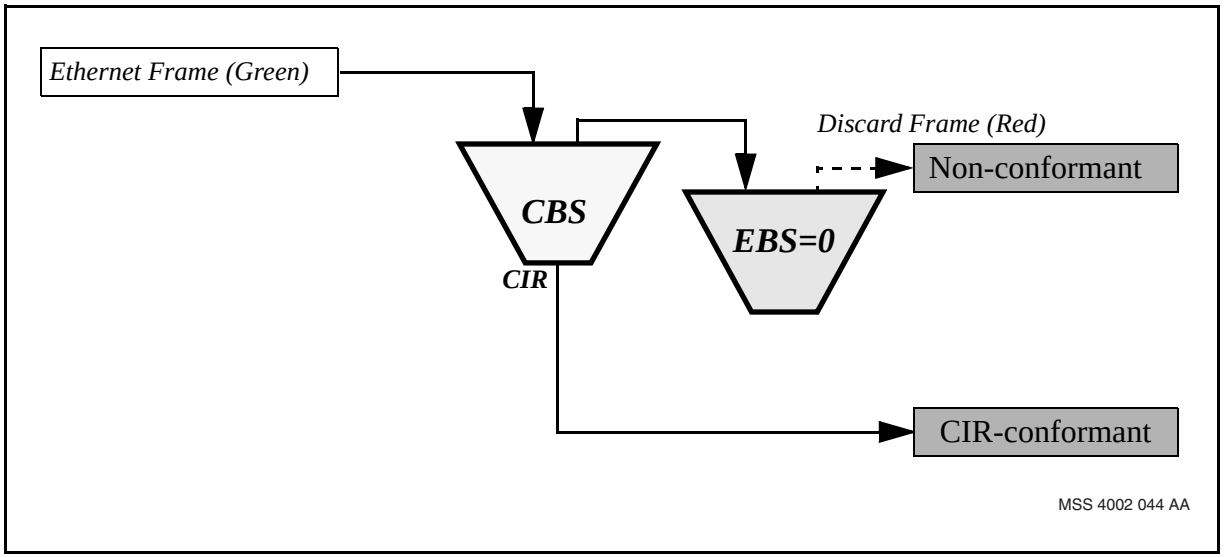
Dual rate policer configuration

	Parameter name	Paramenter value
Policer parameters	committedInformationRate (cir)	non zero
	committedBurstSize (cbs)	non zero
	excessInformationRate (eir)	non zero
	excessBurstSize (ebs)	non zero

Single CIR rate policer with all excess discarded

The single CIR rate policer enforces only CIR rates. All frames are directed to the CIR bucket. Non CIR-conformant frames are directed to the EIR bucket. Since the EIR bucket is zero, all excess frames are discarded.

Single CIR rate policer with all excess discarded



To achieve the single CIR rate policer with all excess traffic discarded the parameters have to be set according to table Single rate CIR only policer configuration (page 204).

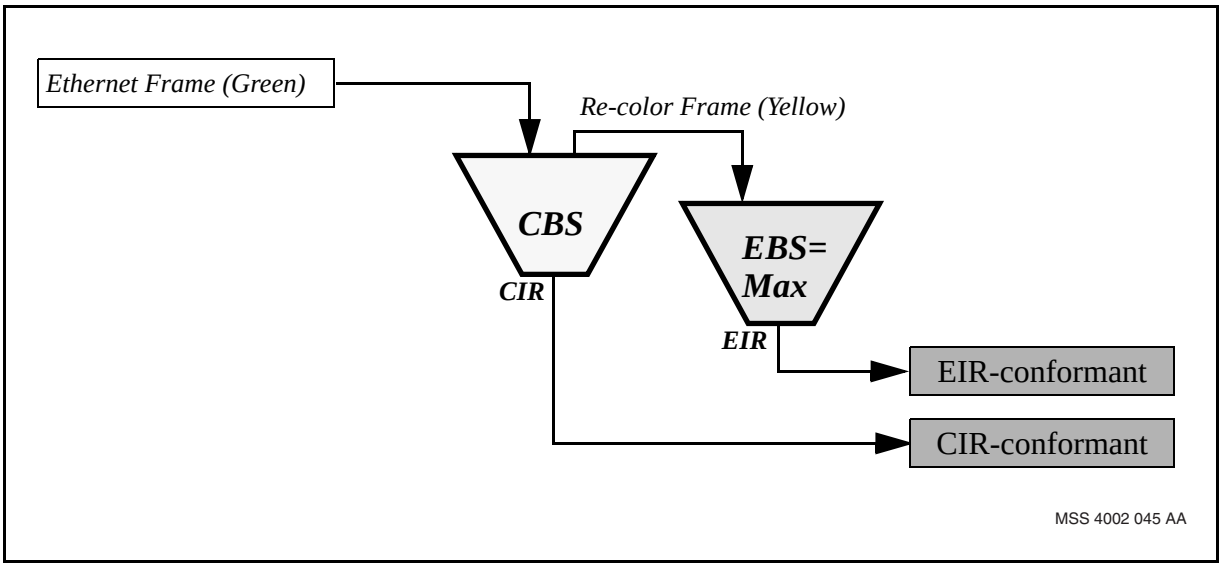
Single rate CIR only policer configuration

	Parameter name	Paramenter value
Policer parameters	committedInformationRate (cir)	non zero
	committedBurstSize (cbs)	non zero
	excessInformationRate (eir)	zero
	excessBurstSize (ebs)	zero

Single CIR rate with all excess forwarded policer

The single CIR rate policer enforces only CIR rates. All frames are directed to the CIR bucket. Non CIR-conformant frames are directed to the EIR bucket. Since the EIR bucket is set to the maximum value, all excess frames are forwarded.

Single CIR rate with all excess forwarded policer



To achieve the single CIR rate policer with excess traffic forwarded the parameters have to be set according to table Single rate CIR with excess forwarded policer configuration (page 205).

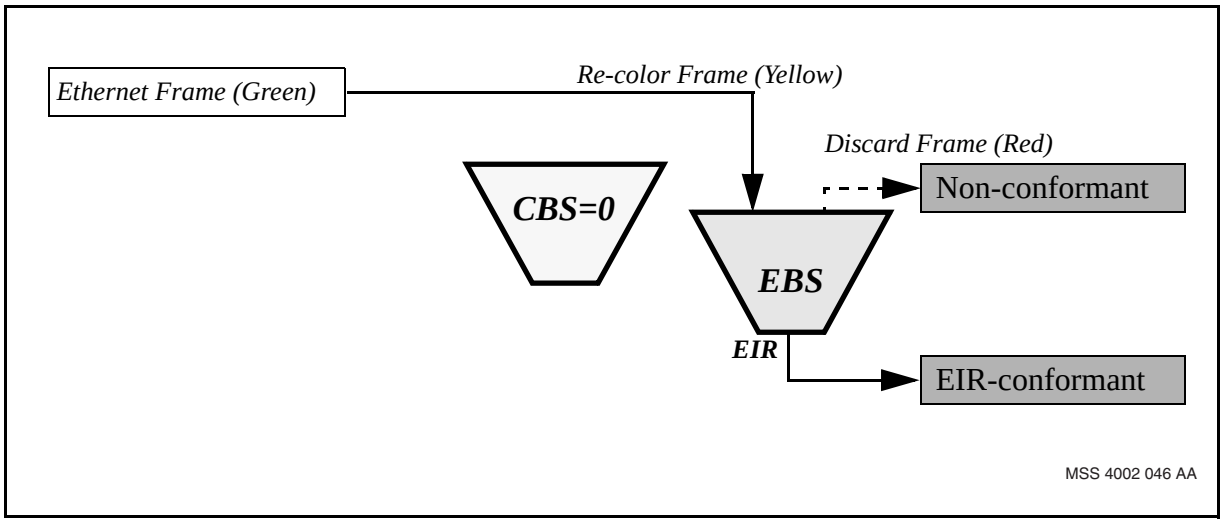
Single rate CIR with excess forwarded policer configuration

	Parameter name	Paramenter value
Policer parameters	committedInformationRate (cir)	non zero
	committedBurstSize (cbs)	non zero
	excessInformationRate (eir)	maximum allowed value (Ethernet access rate)
	excessBurstSize (ebs)	maximum allowed value

Single EIR rate policer

The Single EIR rate policer enforces only EIR rate. All frames are directed to the EIR bucket. Non-conformant frames are discarded.

Single EIR rate policer



To achieve the single EIR rate policer, the parameters have to be set according to table Single EIR policer configuration (page 206).

Single EIR policer configuration

	Parameter name	Paramenter value
Policer parameters	committedInformationRate (cir)	zero
	committedBurstSize (cbs)	zero
	excessInformationRate (eir)	non zero
	excessBurstSize (ebs)	non zero

Policing granularity

4p/8pEth100BaseT FP Granularity

The policer on the 4p/8pEth 100BaseT FPs supports CIR and EIR rates ranging between 7 Kbits per second and the 100BaseT access rate, in 1 Kbit per second increments. Aconfigured information rate value less then 7 Kbits/s is rounded up to 7 Kbits/s. For all other values, the policer may introduce a small error, by rounding the user configured rate up by an amount of no more then 0.1% of the configured value.

The burst sizes of the 4p/8pEth 100BaseT can be configured in the range between 0 and 100 Mbit, in 1 bit increments. The policer introduces a small error to ensure that conformant customer traffic is always accepted by the policer. The error consists of two components: a fixed increase in the burst size of 188 bytes, and a relative increase of less then 0.025% of the configured burst size. The bucket size (burst size) may be set to zero to force all frames

to be non-conforming. This error may cause the policer to admit a small amount of non-conformant traffic, but it will never result in over-policing conformant user traffic which arrives within the configured rate.

4pGe FP Granularity

The 4pGe FP policer algorithm supports rates in the range of 0 Mbits/s to 1,000 Mbits/s in increments of 1 Mbit/s. A rate of zero indicates that all frames are non-conforming to the respective bucket. Other rates are rounded up to the nearest supported value. For example, a customer configured CIR of 29,155 Kbits/s is rounded up by the algorithm to 30 Mbits/s. The algorithm introduces an error of no more than 1% of the configured rate. This error may cause the policer to admit a small amount of non-conformant traffic, but it will never result in over-policing conformant user traffic which arrives within the configured rate.

The 4pGe Policer burst sizes can be configured in the range between 0 and 100 Mbits, in increments of 1 byte.

Policer parameters and operational representation

The policer parameters, cir, eir, cbs and ebs are provisioned. Refer to EVLS traffic management configuration (page 45) for descriptions of these attributes. Due to variance in the granularity of the rates and bucket sizes that are supported by each FP type, the values that are configured may not be reflected exactly in their respective operational attributes. There attributes are *effectiveCommittedInformationRate (effcir), effectiveExcessInformationRate (effeir), effectiveCommittedBurstSize (effcbs), and effectiveExcessBurstSize (effebs)*. The values shown in the operational attributes are the values enforced by the policer. These values are rounded up to the nearest value supported by the FP. See section Policing granularity (page 206) for a description of the policing granularity for each FP type.

Each of the four policing parameters is independently configurable. There is no enforced relationship between them. Since each parameter is configured independently, each bucket is rate enforced over a different time interval. The time interval used for each bucket is derived from the relationships illustrated in figure Policer time intervals derivation (page 207).

Policer time intervals derivation

$$T_{CIR\text{-}bucket} = \frac{CBS}{CIR} \qquad T_{EIR\text{-}bucket} = \frac{EBS}{EIR}$$

MSS 4002 047 AA

When CIR or EIR is set to a rate of zero, the respective bucket size is set to a value of zero, irrespective of the value of the CBS or EBS attribute. Setting a rate to zero has the same effect as setting the respective CBS or EBS attribute to a value of zero. A bucket size of zero effectively disallows any traffic from being accepted as conformant to that bucket.

A bucket size of zero does not disable rate enforcement for that bucket. To disable policing for a particular bucket, set the corresponding rate to the Ethernet port link rate.

Policing statistics

When the policing is activated on an EVC, the statistics related to the policer operation are collected. They are available under the *EvcPolicer* component.

The following statistics are collected:

- Total number of accepted CIR-conformant frames (green) by the CIR policer bucket
- Total number of tagged EIR-conformant frames (yellow) by the CIR policer bucket
- Total number of discarded non-conformant frames (red) by the policer

The policer statistics are provided in addition to the existing total number of received frame and byte counts available under the EthernetStatistics component.

Traffic management functions

This section provides a walkthrough of the EVLS traffic management functions from the Multiservice Switch EVLS ingress node to the egress node.

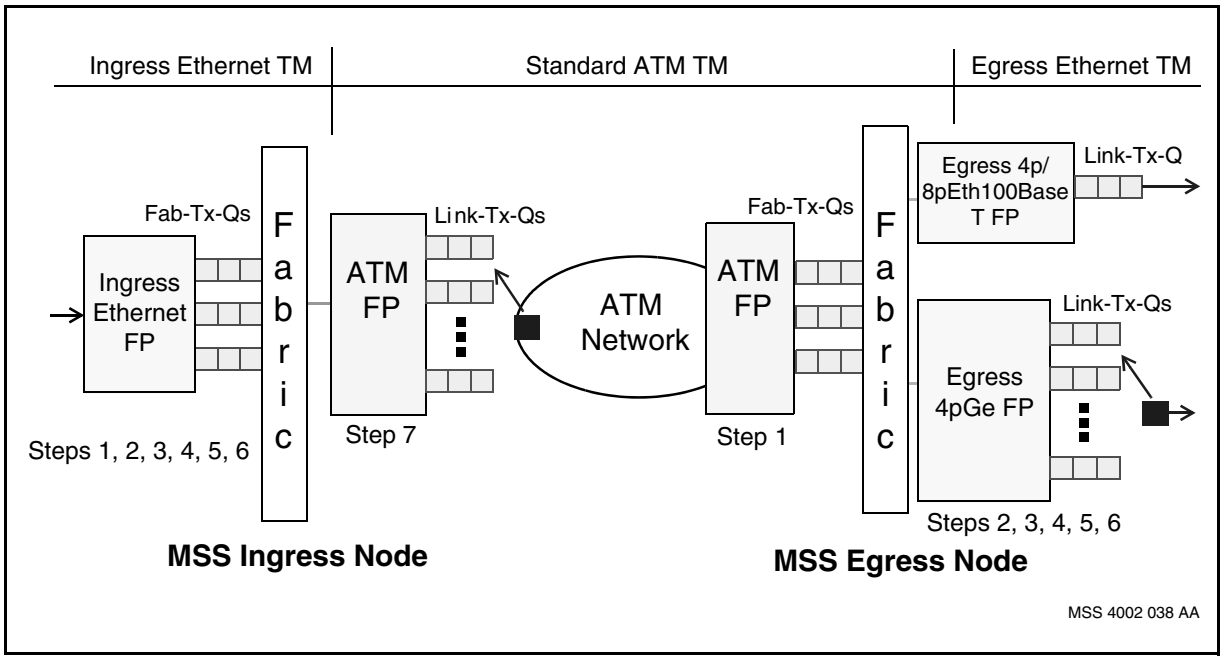
The Multiservice Switch ingress node performs the QoS functions of classifying, optional policing, and QoS mapping to the ATM core network. The Multiservice Switch egress node reclassifies the traffic based on its ATM QoS indicators and/or VLAN configuration information, and forwards the frame onto the Ethernet egress link accordingly.

The Ethernet frame scheduling treatment through the network is determined primarily by its service class (configured as the *ePhb* attribute of the *Evc* component), and its conformance to the policing function.

Multiservice Switch supports service differentiation and congestion control at the network ingress, core, and egress points using the mechanisms shown inEdge-to-edge congestion control mechanism (page 209). Different mechanisms take place at the various frame forwarding points. The functions are broken down into four sequential components: Ingress Ethernet Access

FP, Egress ATM Trunk FP, Ingress ATM Trunk FP, and Egress Ethernet Access FP. The following sections describe the traffic management and congestion control mechanisms of the FPs.

Edge-to-edge congestion control mechanism



Ingress node traffic management

The ingress Ethernet FP performs the following seven functions in tandem. See EVLS traffic management processing on the ingress EVLS node (page 209). The first six functions are performed at the Ethernet access FP, while step seven is performed at the ATM trunk FP.

EVLS traffic management processing on the ingress EVLS node

Step	Title
1	Classification
2	Policing
3	Select drop precedence
4	Select scheduling class
5	Outgoing ATM VCC selection
6	Internal congestion control
7	ATM traffic management

1. Classification

This function determines the Ethernet Virtual Connection (EVC) identifier, based on incoming port or VLAN identifier. Frames of each EVC receive the same QoS treatment, defined by the *ePhb* provisioned attribute of the *Evc EQos* components. Possible values are any of the five values represented on table Ingress Ethernet FP congestion control at the Ingress Multiservice Switch node (page 212).

2. Policing

This is an optional feature, per EVC. If enabled, policing would apply to all frames received by the EVC. Depending on the policer configuration, either dual rates or a single rate may be enforced, and non CIR-conformant traffic may be tagged (colored yellow) or discarded.

Case 1- EVC ethernet policing is not enabled: All frames are assigned “green” color by default.

Case 2- EVC ethernet policing is enabled: A frame is assigned a “green”, “yellow”, or “red” color by the policing mechanism. The “green” frames are CIR conformant. The “yellow” tagged frames are EIR conformant. The “red” tagged frames are discarded.

3. Select drop precedence

The drop precedence (DP) determines the loss sensitivity of the frame relative to other frames in the various internal and fabric queues of the ingress Ethernet FP. In general, frames with a DP of low are less likely to be discarded when the queue is congested. Frames with a DP of high have a lower probability of being forwarded and are most likely to be discarded. The DP is assigned a value of low (1), medium (2), or high (3).

The frame drop precedence (DP) is assigned according to table Ingress Ethernet FP congestion control at the Ingress Multiservice Switch node (page 212). For CIR-conformant traffic, the DP is determined based on the EVC ePhb provisioned value. For EIR-conformant traffic, the DP is always set to “high”, irrespective of the ePhb value.

4. Select scheduling class

Map ePhb to one of the three fabric transmit priority queues according to Table Ingress Ethernet FP congestion control at the Ingress Multiservice Switch node (page 212). Note that the highest priority queue is not used for forwarding EVLS traffic.

5. Outgoing ATM VCC selection

This function maps the Ethernet service class to the ATM VCC that meets its performance requirements. Each EVC is associated with exactly one ATM VCC. This mapping is determined automatically by the system for a non-aggregated EVC based on its configured ePhb.

The aggregated EVC case uses the Ethernet Transport System (ETS). In that case the ATM ServiceCategory, ATM TrafficDescriptor, and bandwidth parameters of the ATM VCC are manually configured by setting the attributes of the *Ets EAtmTrsp Spvc Tm* component. The ATM VCC may be set to any desired value, but the recommendation is that the VCC class can meet the most stringent QoS requirements of its EVCs.

In the non-aggregated case, the ATM ServiceCategory and ATM TrafficDescriptor of the ATM VCC are configured automatically using the *ePhb* attribute of the *Evc eQos* component according the mapping defined on table Ingress Ethernet FP congestion control at the Ingress Multiservice Switch node (page 212). The Ethernet bandwidth parameters used to size the ATM VCC are manually configured by setting the attributes of the *Evc eQos* component.

6. Internal congestion control

The frame is sent across the fabric using the Low or Medium fabric-transmit-queue selected in step 4. First, check the frame drop precedence against the congestion level of the selected queue to determine whether the frame should be forwarded or dropped. Note that the fabric transmit queues are seldom congested because of the Multiservice Switch fabric speed-up and predominantly output-buffered switching architecture. On the Multiservice Switch 7K these queues could also build up in the unlikely case of switching bus overload.

7. ATM egress traffic management

The Egress ATM trunk FP performs standard MSS ATM traffic management functions:

- The AAL5 frame is segmented into ATM cells.
- The ATM VCC service category and traffic parameters are used for determining the cells scheduling treatment, such as emission class and per-VC queuing configuration.
- The ATM cell CLP indication is determined from the ingress frame DP using the standard MSS ATM mapping. The DP values of 1 and 2 are mapped to CLP 0. The DP value of 3 is mapped to CLP 1. This means that Ethernet CIR-conformant traffic of classes EF, AF31, and AF21 is mapped to CLP 0, whereas AF11, DF, and EIR-conformant traffic of all classes is mapped to CLP 1.
- The frame drop precedence is preserved from the ingress Ethernet FP (see Ingress Ethernet FP congestion control at the Ingress Multiservice Switch node (page 212)) on most ATM FPs, including the GQM, AQM, and QRD FPs. The exception is the APC-based FPs, which support only two drop precedence per queue based on the ATM CLP value.

- ATM networking manages the edge-to-edge traffic management through the ATM core network. The ATM networking function utilizes all the existing Multiservice Switch traffic management capabilities, including PNNI routing, priority scheduling, and congestion control. This function is driven by the assigned VCC service category and ATM CLP indication.

Ingress Ethernet FP congestion control at the Ingress Multiservice Switch node

Ethernet PerHopBehavior (ePhb)	Frame DP (non-Policed or CIR-conformant)	Frame DP (Policed EIR-conformant)	Fabric Transmit queue	ATM service category
EF	1 (Low)	3 (High)	Medium	CBR
AF31	1 (Low)	3 (High)	Medium	rt-VBR
AF21	2 (Medium)	3 (High)	Low	nrt-VBR
AF11	3 (High)	3 (High)	Low	UBR+
DF	3 (High)	3 (High)	Low	UBR

Egress node congestion management

The Egress Node performs the following six processing steps on the ATM and Ethernet FPs. See EVLS traffic management processing on the egress EVLS node (page 212). The first step is performed at the ATM ingress FP, while the Egress Ethernet FP performs the other five functions in tandem.

EVLS traffic management processing on the egress EVLS node

Step	Title
1	ATM ingress traffic management
2	Classification
3	Select drop precedence
4	Select scheduling class
5	Egress interface congestion control
6	Ethernet flow control

1. ATM ingress traffic management

The Ingress ATM trunk FP performs standard MSS traffic management functions:

- AAL5 frame assembly.
- Determining the frame Fabric Transmit Queue from the VCC service category.

- Determining the frame DP from the VCC service category and CLP setting (see table Ingress ATM trunk FP congestion control at the egress Multiservice Switch node (page 213)). Note the CLP may be set to 1 because of several reasons: non CIR-conformant tagging by the ingress Ethernet policer, the UBR or UBR+ ATM service category, or tagging by an ATM network policer in the core (or at the edge node hairpin).
- Forwarding the frame across the fabric to the egress Ethernet FP.

Ingress ATM trunk FP congestion control at the egress Multiservice Switch node

ATM service category	ATM CLP	Frame Discard Priority (DP)	Fabric transmit queue
CBR	0	1	High
CBR	1	3	High
rt-VBR	0	1	Medium
rt-VBR	1	3	Medium
nrt-VBR	0	2	Low
nrt-VBR	1	3	Low
UBR+	1	3	Low
UBR	1	3	Low

2. Classification

The classification function is different for the aggregated and non-aggregated EVC cases.

Case 1 - EVC is not aggregated: Map ATM ServiceCategory and ATM TrafficDescriptor to PHB according to table Mapping of ATM service category to ePhb at the egress Ethernet FP (page 213).

Mapping of ATM service category to ePhb at the egress Ethernet FP

ATM service category	Ethernet PerHopBehavior (ePhb)
CBR	EF
rt-VBR	AF31
nrt-VBR	AF21

(1 of 2)

Mapping of ATM service category to ePhb at the egress Ethernet FP

ATM service category	Ethernet PerHopBehavior (ePhb)
UBR+	AF11
UBR	DF
(2 of 2)	

Case 2 - EVC is aggregated: For an EVC configured to use aggregation, the determination of the ePhb of a frame is FP type dependent. On the 4pGe FP, its frame's ePhb is determined based on the configured EVC ePhb (it is recommended to configure the VLAN with same ePhb values at the ingress and egress nodes for QoS consistency, but this guideline is not enforced by the system). On the 4p/8pEth100BaseT FP, its frame's ePhb is determined based on the ATM VCC service category.

3. Select drop precedence

If the ATM CLP= 0, then determine the DP according to the ePhb value; otherwise DP is set to "high" (see table Frame drop precedence determination at the egress Ethernet FP (page 215)). Note that for EVCs, using either non-aggregation or aggregation transport options, the frame drop precedence is set to the highest value of DP 3 if its arriving cells had the CLP set to 1, irrespective of scheduling class.

4. Select scheduling class

Map ePhb to SC8Q (logical scheduling class for 8 queues) according to table Emission priority mapping at the egress 4pGe FP (page 215).

5. Egress interface congestion control

Map Scheduling Class to Emission Priority according to a fixed system mapping. SC8Q (0,1,2,3,4,5,6,7) maps to EP (7,6,5,4,3,2,1,0), respectively. See table Emission priority mapping at the egress 4pGe FP (page 215).

Check the assigned drop precedence against the congestion level of the selected queue to determine whether the frame should be forwarded or dropped.

Both the 4p/8pEth100BaseT and 4pGe FP's hardware implement similar advanced queuing and scheduling mechanisms towards the egress link, that includes deep buffers, priority discards, WRED, and a weighted fair queuing scheduler. However, in this release, the 4p/8pEth100BaseT FP software implements a simpler queuing mechanism. Therefore, we will describe the mechanisms separately.

- The 4pGe FP traffic management device is the Frame Queue Manager (FQM) FPGA. The SC8Q is mapped to an FQM emission queue using

fixed mapping (see table Emission priority mapping at the egress 4pGe FP (page 215)). Next, the frame is enqueued onto the egress FQM queue. Refer to the 4pGe section in NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference*, which describes the FQM queue size and scheduling operations and configurations.

- The 4p/8pEth100BaseT FPs implement a single queue on the Ethernet Traffic Manager (ETM) device for enqueuing traffic to each Ethernet egress link. The queue supports three drop precedence. The frame DP value is preserved from the ATM ingress trunk FP (see table Ingress ATM trunk FP congestion control at the egress Multiservice Switch node (page 213)). Note that the VLANs in the aggregated ETS mode may receive different discard treatment based only on the frame conformance / CLP indication, not the ePhb of the individual VLANs

Frame drop precedence determination at the egress Ethernet FP

ePhb	ATM CLP	Ethernet frame (DP)
EF	0	1 (Low)
EF	1	3 (High)
AF31	0	1 (Low)
AF31	1	3 (High)
AF21	0	2 (Medium)
AF21	1	3 (High)
AF11	1	3 (High)
DF	1	3 (High)

Emission priority mapping at the egress 4pGe FP

ePhb	Scheduling class (SC8Q)	Emission priority queue (EP)
EF	5	2
AF31	4	3
AF21	2	5
AF11	1	6
DF	0	7

6. Ethernet flow control

The Ethernet flow control function is performed by the Ethernet physical device. The device stops sending traffic when it receives a pause frame for the downstream node. This function is only supported on the 4pGe FP.

Procedure conventions

This document uses the following procedure conventions:

- You can enter commands using full component and attribute names, or you can abbreviate them. The commands used in the procedures contain the full component and attribute names in the first instance. In the second instance, the component and attribute names are abbreviated. For more information on abbreviating component and attribute names, see NN10600-060 *Nortel Multiservice Switch 7400/15000/20000 Component Reference*. All component and attribute names are formatted in italics.
- The introduction of every procedure states whether you must perform the procedure in operational mode or provisioning mode. For more information on these modes, see Operational mode (page 217) or Provisioning mode (page 218).
- When you complete a procedure, you can verify your changes and then activate them as the new node configuration. For more information on completing configuration changes and exiting provisioning mode, see Activating configuration changes (page 218).

Operational mode

Procedures contained within this document can either be performed in operational mode or provisioning mode. When you initially log into a node, you are in operational mode. Nortel Multiservice Switch systems use the following command prompt when you are in operational mode:

```
#>
```

where:
is the current command number

In operational mode, you work with operational components and attributes. In operational mode, you can

- list operational components and display operational attributes to determine the current operating parameters for the node
- control the state of parts of the node by locking and unlocking components

- set certain operational attributes and enter commands to perform diagnostic tests

Provisioning mode

To change from operational mode to provisioning mode, type the following command at the operator prompt:

```
start Prov
```

Only one user can be in provisioning mode at a time. Nortel Multiservice Switch systems use the following command prompt whenever you are in provisioning mode:

```
PROV #>
```

where:

is the current command number

In provisioning mode, you work with the provisionable components and attributes that contain the current and future configurations of the node. You can add and delete components, and display and set provisionable attributes. For information on completing the configuration changes, exiting provisioning mode, and returning to operational mode see Activating configuration changes (page 218).

For information on operational and provisionable attributes, see NN10600-060 *Nortel Multiservice Switch 7400/15000/20000 Component Reference*.

Activating configuration changes

Several procedures in this document ask that you complete the configuration changes. When you complete the configuration changes, you are activating the configuration changes, confirming that you want to activate them, and saving the changes. You are instructed to complete the configuration changes only at the end of procedures that you perform in provisioning mode.

CAUTION

Activating a provisioning view can affect service

Activating a provisioning view can result in a CP reload or restart, causing all services on the node to fail. See NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*, for more information.

CAUTION
Risk of service failure
When you activate the provisioning changes (see step 3), you have 20 minutes to confirm these changes. If you do not confirm these changes within 20 minutes, the shelf resets and all services on the node fail.

1 Verify that the provisioning changes you have made are acceptable.

```
check Prov
```

Correct any errors and then verify the provisioning changes again.

2 If you want to store the provisioning changes in a file, save the provisioning view.

```
save -f(<filename>) Prov
```

3 If you want these changes as well as other changes made in the edit view to take effect immediately, activate, confirm, and commit the provisioning changes.

```
activate Prov
confirm Prov
commit Prov
```

4 End the provisioning session.

```
end Prov
```

Nortel Multiservice Switch 7400/15000/20000

Operations: Ethernet Line Virtual Services

Publication: NN10600-822
Document status: Standard
Document issue: 7.2S1
Document date: March 2006
Product release: PCR7.2 and up
Job function: Operations
Type: NTP
Language type: U.S. English