

12. Bridge management

12.1.1 Configuration management

Insert the following bullet item, relettered if necessary to follow the existing list.

- g) The ability to create and delete the functional elements of congestion notification and to control their operation.

12.2 Managed objects

Insert the following bullet item, relettered if necessary to follow the existing list.

- j) The congestion notification entities (12.17).

Insert a new Clause 12.17 as follows:

12.17 Congestion notification managed objects

A number of the variables that implement congestion notification, including CPs, RPs, and CND defense, are manageable objects¹. There are a number of managed objects, each including a number of variables. The managed objects are the:

- a) CN component managed object (12.17.2);
- b) CN component priority managed object (12.17.3);
- c) CN Port priority managed object (12.17.4);
- d) Congestion Point managed object (12.17.5);
- e) Reaction Point port priority managed object (12.17.6); and
- f) Reaction Point group managed object (12.17.7).

NOTE—If multiple managed objects are altered over a period of time, then between the time the first and last object has been altered, operation of the state machines could produce unexpected results. This standard assumes that any number of managed objects can be altered as an atomic operation, so that no inconsistent intermediate states can occur. See 17.7.16 for one mechanism to ensure consistency.

12.17.1 Overriding variables

CN component managed objects (12.17.2), CN component priority managed objects (12.17.3), CN Port priority managed objects (12.17.4), and Congestion Point managed objects (12.17.5) override and are overridden by each other as follows:

- a) Altering the CN component managed object (12.17.2) or CN component priority managed object (12.17.3) does not alter the values of any of the other managed objects (12.17.4, 12.17.5, 12.17.7).
- b) If `cngMasterEnable` (32.2.1) is FALSE, all congestion notification activity is suppressed; the Priority Regeneration Table (6.9.3) operates normally, and CNMs, CN-TAGs, and LLDP Congestion Notification TLVs are never generated and are always ignored on receipt. If TRUE, the other managed objects control the operation of congestion notification.

1. The managed objects in this subclause are documented in a manner that is not parallel to that of the managed objects included in IEEE Std. 802.1Q-2005 and certain other of its amendments. It is intended that managed object definitions in future amendments to IEEE Std. 802.1Q will follow the format in this subclause. The resultant inconsistencies will be resolved in a future edition of IEEE Std. 802.1Q, insofar as this can be done without invalidating the MIB definitions in Clause 17, some of which depend on the format of Clause 12 used in IEEE Std. 802.1Q-2005.

- c) The Port/priority and component objects override each other as shown in Table 12-1 and Table 12-2. In each case, if the Port/priority object contains the specified value, the component or component/priority object controls the specified function.

Table 12-1—LLDP instance selection managed object overrides

(per-port per-priority) cnpdLldpInstanceChoice 32.4.4	(component per-priority) cncpLldpInstanceChoice 32.3.6	Congestion Notification TLV is sent/received	LLDP instance is selected by
cnlNone	any	No	none selected
cnlAdmin	any	Yes	(per-port per-priority) cnpdLldpInstanceSelector 32.4.5
cnlComponent	cnlNone	No	none selected
cnlComponent	cnlAdmin	Yes	(component per-priority) cncpLldpInstanceSelector 32.3.7

Table 12-2—CND defense mode selection managed object overrides

(per-port per-priority) cnpdDefModeChoice 32.4.1	(component per-priority) cncpDefModeChoice 32.3.1	CND defense mode is selected by	Alternate priority is selected by
cpcAdmin	any	(per-port per-priority) cnpdAdminDefenseMode 32.4.2	(per-port per-priority) cnpdAlternatePriority 32.4.6
cpcAuto	any	(per-port per-priority) cnpdAutoDefenseMode 32.4.3	(component per-priority) cncpAutoAltPri 32.3.3
cpcComp	cpcAdmin	(component per-priority) cncpAdminDefenseMode 32.3.4	(component per-priority) cncpAlternatePriority 32.3.2
cpcComp	cpcAuto	(per-port per-priority) cnpdAutoDefenseMode 32.4.3	(component per-priority) cncpAutoAltPri 32.3.3

12.17.2 CN component managed object

A single instance of the CN component managed object shall be implemented by a bridge component or end station that is congestion aware. It comprises all of the variables included in the CN component variables (32.2), as illustrated in Table 12-3. An end station may omit the managed objects noted “C” in the Conformance column of Table 12-3 if it does not support Congestion Points (CPs), and may in any case omit the managed object marked “e”.

Table 12-3—CN component managed object row elements

Name	Data type	Operations supported ^a	Conformance ^b	References
cngMasterEnable	Boolean	RW	BE	32.2.1
cngCnmTransmitPriority	unsigned integer [0..7]	R	BC	32.2.2
cngDiscardedFrames	counter	R	BC	32.2.3
cngErroredPortList	list	R	Be	32.2.4

- a R = Read only access;
RW = Read/Write access
- b B = Required for a Bridge or Bridge Component that is congestion aware;
C = Required for an end station that implements one or more CPs
E = Required for an end station that is congestion aware
e = Optional for an end station that is congestion aware

12.17.3 CN component priority managed object

The CN component priority managed object contains the managed objects that control a single CNPV for all Ports in an end station or bridge component. It comprises all of the variables included in the Congestion notification per-CNPV variables (32.3), as illustrated in Table 12-4. In one common use case for congestion notification, every Port of a bridge or end station has the same number of CPs, each configured for the same set of priority values. This managed object facilitates configuring that case. These objects can override, and can be overridden by, the CN Port priority managed objects (12.17.4) as described in 12.17.1.

Table 12-4—CN component priority managed object row elements

Name	Data type	Operations supported ^a	Conformance ^b	References
cncpDefModeChoice	enum {cpcAdmin, cpcAuto}	RW	BE	32.3.1
cncpAlternatePriority	integer [0..7]	RW	BC	32.3.2
cncpAutoAltPri	integer [0..7]	R	BC	32.3.3
cncpAdminDefenseMode	enum {cptDisabled, cptInterior, cptInteriorReady, cptEdge}	RW	BE	32.3.4
cncpCreation	enum {cncpAutoEnable, cncpAutoDisable}	RW	BE	32.3.5
cncpLldpInstanceChoice	enum {cnlNone, cnlAdmin}	RW	BE	32.3.6
cncpLldpInstanceSelector	802.1AB LLDP instance selector	RW	BE	32.3.7

- a R = Read only access;
RW = Read/Write access
- b B = Required for a Bridge or Bridge Component that is congestion aware;
C = Required for an end station that implements one or more CPs
E = Required for an end station that is congestion aware

A CN component priority managed object shall be implemented by a bridge component or end station that is congestion aware for each priority value that can be a CNPV. An end station may omit the managed objects noted “C” in the Conformance column in Table 12-4 if it does not support Congestion Points (CPs). The operations that can be performed on a congestion aware system’s CN component priority managed object are:

- a) Create CN component priority managed object (12.17.3.1); and
- b) Delete CN component priority managed object (12.17.3.2).

12.17.3.1 Create CN component priority managed object

Creating a CN component priority managed object creates an instance of each of the Congestion notification per-CNPV variables (32.3), and also creates the corresponding CN Port priority managed objects (12.17.4) and all their dependent managed objects and variables, on every Port in the bridge component or end station, as illustrated in CN Port priority managed object. Depending on the value of cncpCreation (32.3.5), creating a CN component priority managed object can make the selected priority a CNPV throughout the bridge component or end station.

12.17.3.2 Delete CN component priority managed object

Deleting a CN component priority managed object deletes all of the CN component variables (32.2), and also deletes the corresponding CN Port priority managed objects (12.17.4) and all their dependent managed objects and variables, on all Ports in the bridge component or end station, thus making the priority not a CNPV throughout the bridge component or end station.

12.17.4 CN Port priority managed object

There is one CN Port priority managed object per Port per priority in a congestion aware end station or bridge component. It comprises some of the variables included in the CND defense per-Port per-CNPV variables (32.4), as illustrated in Table 12-5. These objects can override, and can be overridden by, the CN component managed objects (12.17.2) and CN component priority managed objects (12.17.3) as described in 12.17.1.

Table 12-5—CN Port priority managed object row elements

Name	Data type	Operations supported ^a	Conformance ^b	References
cnpdDefModeChoice	enum{cpcAdmin, cpcAuto, cpcComp}	RW	BE	32.4.1
cnpdAdminDefenseMode	enum{cptDisabled, cptInterior, cptInteriorReady, cptEdge}	RW	BE	32.4.2
cnpdAutoDefenseMode	enum{cptDisabled, cptInterior, cptInteriorReady, cptEdge}	R	BE	32.4.3
cnpdLldpInstanceChoice	enum{cnlNone, cnlAdmin}	RW	BE	32.4.4
cnpdLldpInstanceSelector	802.1AB LLDP instance selector	RW	BE	32.4.5
cnpdAlternatePriority	integer [0..7]	RW	BC	32.4.6

- a R = Read only access;
RW = Read/Write access

- 1 b B = Required for a Bridge or Bridge Component that is congestion aware;
2 C = Required for an end station that implements one or more CPs
3 E = Required for an end station that is congestion aware
4

5
6 A CN Port priority managed object is created or deleted when the corresponding Port or CN component
7 priority managed object is created or deleted, and its initial state on creation is determined by `cnpcCreation`
8 (32.3.5).
9

10 The CN Port priority managed object shall be implemented by a bridge component or end station that is
11 congestion aware. An end station may omit the managed objects noted “C” in the Conformance column in
12 Table 12-5 if it does not support Congestion Points (CPs).
13

14 **12.17.5 Congestion Point managed object**

15
16 There is one Congestion Point managed object for each CP in a bridge component or end station. It
17 comprises some of the variables included in the Congestion Point variables (32.8), as illustrated in
18 Table 12-6.
19

20 **Table 12-6—Congestion Point managed object row elements**

Name	Data type	Operations supported ^a	Conformance ^b	References
<code>cpMacAddress</code>	MAC address	R	BC	32.8.1
<code>cpId</code>	octet string (size = 8)	R	BC	32.8.2
<code>cpQSp</code>	unsigned integer	R	BC	32.8.3
<code>cpW</code>	real number	RW	BC	32.8.6
<code>cpSampleBase</code>	unsigned integer	RW	BC	32.8.11
<code>cpDiscardedFrames</code>	counter	R	BC	32.8.12
<code>cpTransmittedFrames</code>	counter	R	BC	32.8.13
<code>cpTransmittedCnms</code>	counter	R	BC	32.8.14

- 21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38 a R = Read only access;
39 RW = Read/Write access
40 b B = Required for a Bridge or Bridge Component that is congestion aware;
41 C = Required for an end station that implements one or more CPs
42

43 The Congestion Point managed object shall be implemented by a congestion aware bridge component. It
44 shall be implemented by an end station that supports Congestion Points (CPs).
45

46 NOTE—The Recommended priority to traffic class mappings in Table 8-2 can assign more than one CNPV to the same
47 traffic class, the same queue, and hence the same CP. There can be only one CP controlling a given queue, and that CP
48 has one set of controlling managed objects, not one set per CNPV. That set of managed objects can be accessed using
49 any of the CNPV values assigned to the CP’s queue. Thus, changing a managed object for one CNPV changes the
50 managed object for all CNPVs assigned to the same queue.
51
52
53
54

12.17.6 Reaction Point port priority managed object

A congestion aware end station shall implement one Reaction Point port priority managed object for each CNPV on each port. The Reaction Point port priority managed object controls the creation of RPs on the port and CNPV, as illustrated in Table 12-7.

Table 12-7—Reaction Point port priority managed object row elements

Name	Data type	Operations supported ^a	Conformance ^b	References
rpppMaxRps	unsigned integer	RW	E	32.10.1
rpppCreatedRps	counter	R	E	32.10.2
rpppRpCentiseconds	unsigned integer	R	E	32.10.3

a R = Read only access;
RW = Read/Write access

b E = Required for an end station that is congestion aware

12.17.7 Reaction Point group managed object

There is one Reaction Point group managed object for each set of Reaction Point group variables (32.11), as illustrated in Table 12-8. The Reaction Point group managed object shall be implemented by an end station that is congestion aware.

Table 12-8—Reaction Point group managed object row elements

Name	Data type	Operations supported ^a	Conformance ^b	References
rpgEnable	Boolean	RW	E	32.11.1
rpgTimeReset	unsigned integer	RW	E	32.11.2
rpgByteReset	unsigned integer	RW	E	32.11.3
rpgThreshold	unsigned integer	RW	E	32.11.4
rpgMaxRate	unsigned integer	RW	E	32.11.5
rpgAiRate	unsigned integer	RW	E	32.11.6
rpgHaiRate	unsigned integer	RW	E	32.11.7
rpgGd	real number	RW	E	32.11.8
rpgMinDecFac	real number	RW	E	32.11.9
rpgMinRate	unsigned integer	RW	E	32.11.10

a RW = Read/Write access

b E = Required for an end station that is congestion aware