46. Time-Sensitive Networking (TSN) configuration

Change the introductory text of Clause 46 as follows:

Time-Sensitive Networking (TSN) is a collection of features in IEEE 802.1 standards that provide the following:

- — Time synchronization among Bridges and end stations
- — Significant reduction in frame loss due to faults in network equipment
- — Significant reduction in, or the elimination of, frame loss due to egress Port congestion
- — Bounded latency

This clause provides specifications for the configuration of TSN features in a network. The configuration process begins when Talkers and Listeners pass their requirements to the network and proceeds with the configuration of TSN features in Bridges along a tree from each Talker to its Listener(s).

46.1 Overview of TSN configuration

Change 46.1 as follows:

46.1.1 ~~User/Network Interface (UNI)~~Streams, Talkers, and Listeners

TSN configuration uses the concept of a Stream that is transmitted by a Talker to one or more Listeners. The Talkers and Listeners are located within end stations.

This clause specifies configuration information~~ that is exchanged over a User/Network Interface (UNI). The user side of the interface represents~~ for Streams requested by Talkers and Listeners, or by network management entities acting on their behalf.~~ The network side of the interface represents the Bridges that transfer frames of the Stream from each Talker to its Listeners. Each user specifies~~ Stream requirements~~ for its data, but~~ are initially specified,by, or for, Talker and Listener without detailed knowledge of the network. ~~The network~~ Network entities or administrators obtain~~s~~ these requirements~~ from users~~, analyze~~s~~ the topology and~~ TSN~~ capabilities of the Bridges, and configure~~s the~~ Bridges to meet~~ user~~ the requirements. The network returns the success or failure of each Stream's configuration to~~ the user~~ its Talker(s) and Listener(s).

46.1.2 ~~Modeling of user/network~~TSN configuration~~ information~~ protocols

A variety of protocols can be used for the exchange of configuration information~~ over the TSN UNI~~ (e.g., signaling protocols, remote network management protocols).~~ These protocols can exchange the configuration information as text or as binary fields. To enable flexible integration of TSN configuration into a variety of protocols, 46.2 specifies the TSN user/network configuration information in a manner that is independent of schema, encoding, or protocol.~~

Specific TSN-capable products list the ~~user/network~~ protocol that is supported as part of their conformance [e.g., 5.18.3, item c) in 5.29]. Each user/network protocol will specify a specific schema and/or encoding for the configuration information in 46.2. Examples of these protocols are described for each of the TSN configuration models in 46.1.3.

46.1.3 TSN configuration models

~~This subclause describes three~~Three models for TSN user/network configuration are described. These models provide an architectural context for subsequent specifications. Each model specification shows the logical flow of~~ user/network~~ configuration information between various entities in the network.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

46.1.3.1 Fully distributed model

In the fully distributed model, the end stations that contain~~ users of Streams (i.e.,~~ Talkers and Listeners~~)~~ communicate the user requirements directly~~ over the TSN user/network protocol~~ to the neighboring Bridge. The network is configured in a fully distributed manner, without a centralized network configuration entity. The distributed network configuration is performed using a protocol that propagates~~ TSN user/network~~ configuration information along the active topology for the Stream (i.e., Bridges in the tree from Talker to Listeners).

As~~ user~~ Stream requirements propagate through each Bridge, management of the Bridge's resources is effectively performed locally. This local management is limited to the information that the Bridge has knowledge of and does not necessarily include knowledge of the entire network.

Figure 46-1 provides a graphical representation of the fully distributed model.

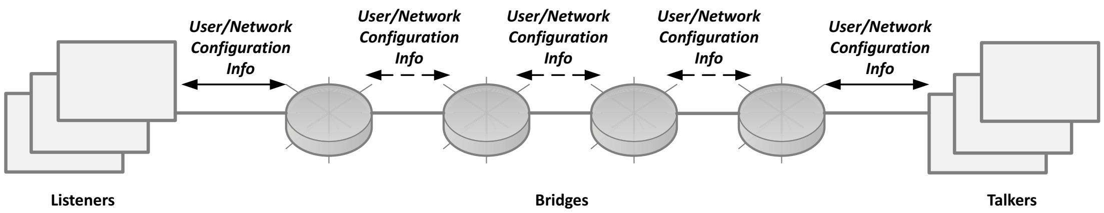

Figure 46-1—Fully distributed model

In the figure, the solid arrows represent the protocol that is used~~ as the UNI for~~ to exchange of~~ configuration~~ information between Talkers/Listeners (users) and Bridges (network). This~~ configuration~~ information is specified in 46.2.

In the figure, the dashed arrows represent the protocol that propagates~~ configuration~~ information within the network. This protocol carries~~ the TSN user/network~~ Stream configuration information (46.2) as well as additional information~~ that is~~ specific to network configuration.

The following TSN features can be configured by Bridges using this model:

a) Credit-based shaper algorithm (8.6.8.2) and its configuration (Clause 34)

The Stream Reservation Protocol (SRP) of Clause 35 can be used~~ as the UNI~~ by Talkers and Listeners, and to propagate configuration information throughout the network of Bridges.~~ SRP exchanges configuration information as binary fields using the Type-Length-Value (TLV) technique. Using this technique, the protocol's top-level message contains a list of one or more TLVs. Each TLV consists of a Type field that specifies what the Value field contains, a Length field that specifies the number of octets in the Value field, and the Value field. In SRP specifications, each TLV Type identifies one of the groups specified in 46.2, and the TLV Value contains a binary representation of the elements in that group.~~

46.1.3.2 Centralized network/distributed user model

Some TSN use cases are computationally complex. For example, for scheduled traffic (8.6.8.4), computation of the gate control list of each Port can take significant time. For such use cases, it is helpful to centralize the computation in a single entity (Bridge or end station), rather than perform the computation in all Bridges.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

Some TSN use cases can benefit from a complete knowledge of all Streams in the network. For example, if the bandwidth for multiple Streams is greater than the available bandwidth along the shortest path between Talkers and Listeners, it is helpful to forward a subset of those Streams along a path other than the shortest. For these use cases, a centralized entity can gather information for the entire network in order to find the best configuration.

~~The~~ In the centralized network/distributed user model, ~~is similar to the fully distributed model in that~~ end stations communicate their Talker/Listener Stream requirements directly ~~over the TSN UNI~~ to the neighboring Bridge, just as in the fully distributed model. However, in the centralized network/distributed user model, a Centralized Network Configuration (CNC, 46.1.6) entity configures Bridges to pass Talker/Listener Stream requirements directly to the CNC, rather than propagating that information along the path to be taken by Stream data. ~~In contrast, in the centralized network/distributed user model, the configuration information is directed to/from a Centralized Network Configuration (CNC) entity.~~ All configuration of Bridges for ~~TSN~~ Streams is performed by this CNC using a remote network management protocol.

The CNC has a complete view of the physical topology of the network as well as the capabilities of each Bridge. This enables the CNC to centralize complex computations. The CNC can exist in either an end station or a Bridge.

~~The CNC knows the address of all Bridges at the edge of the network (those with an end station connected). The CNC configures those edge Bridges to act as a proxy, transferring Talker/Listener information directly between the edge Bridge and the CNC, rather than propagate the information to the interior of the network.~~

Figure 46-2 provides a graphical representation of the centralized network/distributed user model.

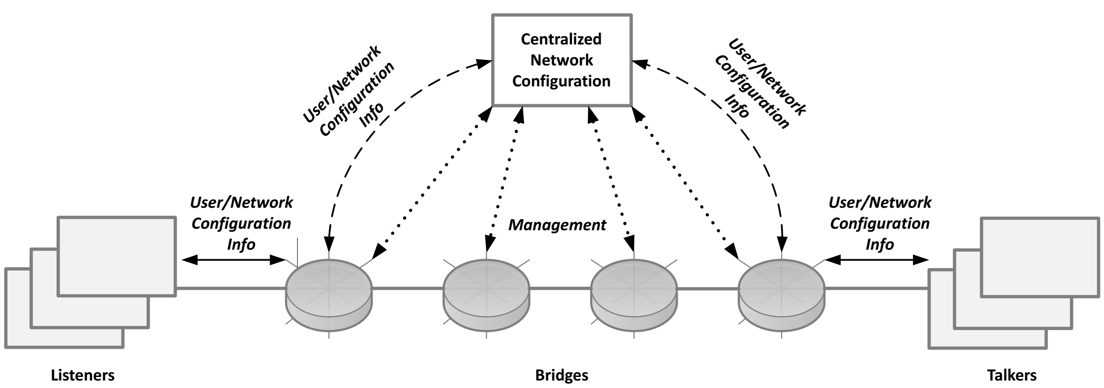

Figure 46-2—Centralized network/distributed user model

In the figure, the solid arrows represent the protocol that is used ~~as the UNI for~~ to exchange ~~of configuration~~ Stream information between Talkers/Listeners (users) and Bridges (network). This ~~configuration~~ information is specified in 46.2.

In the figure, the dashed arrows represent the protocol that transfers Stream configuration information between ~~edge~~ Bridges and the CNC. This ~~configuration~~ information is specified in 46.2.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

In the figure, dotted arrows represent the remote network management protocol. The CNC acts as the management client, and each Bridge acts as the management server. The CNC uses remote management to discover physical topology, retrieve Bridge capabilities, and configure TSN features in each Bridge. Talkers and Listeners are not required to participate in this remote network management protocol. The information carried by the remote network management protocol is specified in Clause 12.

NOTE 1—If the Talker/Listener protocol of the fully distributed model is selected to be the same as the Talker/Listener protocol of the centralized network/distributed user model, end stations can support both models without explicit knowledge of how the network is configured.

The following TSN features can be configured by the CNC using this model:

a) Credit-based shaper algorithm (8.6.8.2) and its configuration (Clause 34)
b) Frame preemption (6.7.2)
c) Scheduled traffic (8.6.8.4, 8.6.9)
d) Frame Replication and Elimination for Reliability (IEEE Std 802.1CB)
e) Per-Stream Filtering and Policing (8.6.5.1)
f) Cyclic queuing and forwarding (Annex T)

SRP (Clause 35) can be used as the ~~UNI~~ Talker/Listener protocol (solid arrows of Figure 46-2). SRP's MRP External Control (12.32.4) feature can be used to exchange configuration information with the CNC component (dashed arrows of Figure 46-2). SRP exchanges configuration information using the TLV technique to reference elements in 46.2 (see 46.1.3.1). Examples of a remote network management protocol (dotted arrows of Figure 46-2) include Simple Network Management Protocol (SNMP), NETCONF (IETF RFC 6241 [B39]), and RESTCONF (IETF RFC 8040 [B47]).

NOTE 2—NETCONF and RESTCONF specify a startup datastore: nonvolatile configuration that is applied when the Bridge powers on. The startup datastore feature enables a ~~TSN~~ CNC to configure Bridges and then remove itself from the network. SNMP does not specify a startup datastore feature. If SNMP is used by a ~~TSN~~ CNC, this can be mitigated by a) using a proprietary (Bridge-specific) startup datastore feature or b) ensuring that the ~~TSN~~ CNC is always active in the network in order to reconfigure Bridges that cycle power.

46.1.3.3 Fully centralized model

Many TSN use cases require significant user configuration in the end stations that act as Talkers and Listeners. For example, in many automotive and industrial control applications, the timing of physical inputs and outputs (I/Os) is determined by the physical environment under control, and the timing requirements for TSN Streams are derived from that I/O timing. In some use cases, these I/O timing requirements can be computationally complex and involve detailed knowledge of the application software/hardware within each end station.

In order to accommodate this sort of TSN use case, the fully centralized model enables a Centralized User Configuration (CUC, 46.1.5) entity to discover end stations, retrieve end station capabilities and user requirements, and configure TSN features in end stations. The protocols that the CUC uses for this purpose are specific to the user application and outside the scope of this standard.

From a network perspective, the primary difference between the fully centralized model and the centralized network/distributed user model is that all user requirements are exchanged between the CNC and CUC. ~~Therefore, the TSN UNI exists between the CNC and CUC.~~

Figure 46-3 provides a graphical representation of the fully centralized model.

In the figure, the solid arrows represent the ~~protocol that is used as the UNI for~~ exchange of configuration information between the CUC and the CNC. This configuration information is specified in 46.2.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

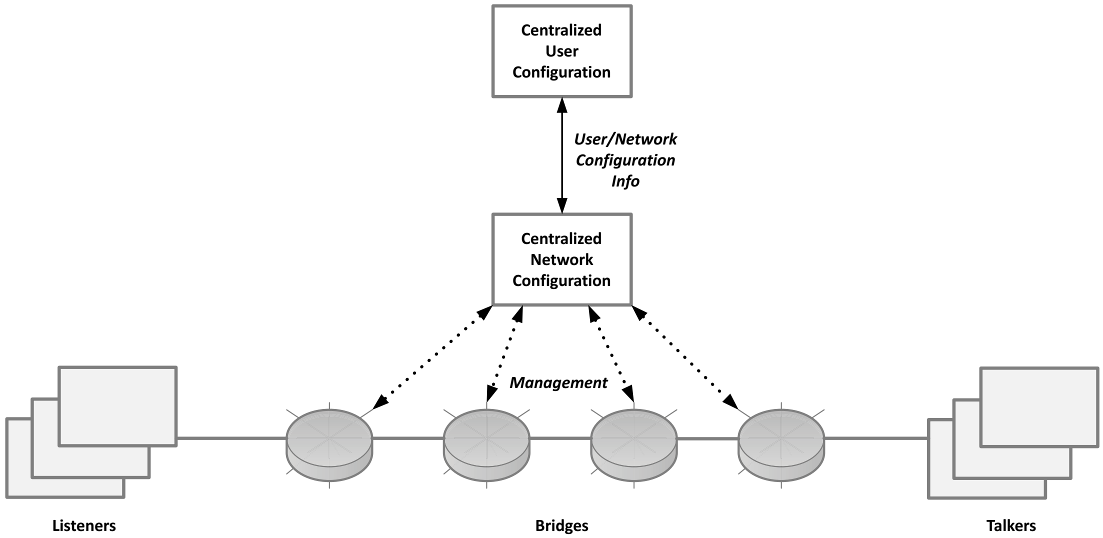

Figure 46-3—Fully centralized model

In the figure, the dotted arrows represent the remote network management protocol. The CNC acts as the management client, and each Bridge acts as the management server. The CNC uses remote management to discover physical topology, retrieve Bridge capabilities, and configure TSN features in each Bridge. Talkers and Listeners are not required to participate in this remote network management protocol. The information carried by the remote network management protocol is specified in Clause 12.

In this fully centralized model, a protocol is used between the CUC and end stations (Talkers and Listeners) to retrieve end station capabilities and requirements and to configure the end stations. ~~Since that protocol is user-to-user, its configuration information is considered to be outside the scope of this standard, and it is not shown in Figure 46-3.~~

The following TSN features can be configured by the CNC using this model:

a) Credit-based shaper algorithm (8.6.8.2) and its configuration (Clause 34)
b) Frame preemption (6.7.2)
c) Scheduled traffic (8.6.8.4, 8.6.9)
d) Frame Replication and Elimination for Reliability (IEEE Std 802.1CB)
e) Per-Stream Filtering and Policing (8.6.5.1)
f) Cyclic queuing and forwarding (Annex T)

YANG (IETF RFC 7950) is a data modeling language used to model configuration data and state data for remote network management protocols. The remote network management protocol uses a specific encoding such as XML or JSON. For a particular feature, a YANG module specifies the organization and rules for the feature's management data, and a mapping from YANG to the specific encoding enables the data to be understood correctly by both client (e.g., network manager) and server (e.g., Bridge). ~~Technically speaking, the TSN user/network configuration is not network management, in that information is exchanged between user and network, and not between a network manager and the network's Bridges (Clause 12). Nevertheless, the concepts are sufficiently similar that YANG is useful for modeling the configuration and state data for the TSN user/network configuration information.~~

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

In order to support the use of YANG-based protocols for the fully centralized model, 46.3 specifies a YANG module. The YANG module specifies a YANG typedef/grouping for each group of information in 46.2.

~~NOTE—At the time that this clause was developed, specific protocol implementations for the fully centralized model were a work in progress. One protocol explored for the UNI between CUC and CNC is RESTCONF (IETF RFC 8040 [B47]). A complete YANG module for the TSN UNI can be specified in a document other than IEEE Std 802.1Q. In order to conform with this clause, the complete TSN UNI YANG module imports the YANG module of 46.3 for use within its containers and lists. The complete TSN UNI YANG module will presumably specify features outside the scope of this clause, such as operations to control the deployment of Stream configuration to the network. The JSON encoding can be used with RESTCONF. Although the TSN UNI is technically not network management, use of RESTCONF provides a simple and effective application programming interface (API) for TSN configuration.~~

For an informative example workflow using the fully centralized model, refer to U.2.

46.1.4 Stream identification and transformation

TSN configuration uses the concept of a Stream of data that is transmitted by a Talker to one or more Listeners.~~ The Talkers and Listeners are end stations.~~

In order to apply TSN behavior to a Stream~~s~~ (e.g., to reserve~~d~~ bandwidth~~ guarantees~~), the network~~ must be able to~~ distinguishes ~~one ~~Streams from~~ another Stream and distinguish Streams~~ each other and from non-TSN traffic (e.g., best-effort). ~~Therefore, e~~Each frame of~~ the~~ a Stream~~ must~~ contains fields~~ in its header~~ that uniquely identify the Stream.

The goal of TSN configuration is to allow Talkers and Listeners to use their existing transport layer and application layer protocols for data, rather than requiring a TSN-specific frame format. TSN achieves this goal by identifying each Stream using fields from well-established frame formats such as Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and IEEE 802.1 (i.e., MAC addresses and VLAN identifier).

As the frames of each Stream cross the user/network boundary, the identification of the Stream in its frames can be different between the network and the user. For example, the user can use UDP without an awareness of VLAN IDs, but the network can require a specific VLAN ID in order to apply TSN features. In order to support this sort of difference in frame format, the TSN user/network configuration information (46.2) provides features to enable transformation of the Stream's identification at the user/network boundary. The user identification translates to/from the network identification at the boundary, either within the end station or a nearest Bridge. This transformation has the benefit of allowing the user's identification to match its higher layer application protocol and the network's identification to match the bridging technology.

Stream transformation can be accomplished using the functions specified in IEEE Std 802.1CB. The functions of IEEE Std 802.1CB can be implemented in the end station (Talker/Listener) or within the nearest Bridge. The descriptions in this clause focus on Stream transformation in the end station and use features of the TSN user/network configuration information (46.2).

NOTE 1—In this clause, Stream transformation refers to changes to the fields of a frame that identify the Stream. IEEE Std 802.1CB specifies Stream transformation for identification as well as for frame replication and elimination (redundancy).

NOTE 2—Stream transformation is an optional capability of end stations and Bridges. If stream transformation is not supported, the user's identification of the Stream must be the same as the network's identification, and the user must use an identification that is consistent with bridging as specified for TSN features in this standard (e.g., VLAN tag and group destination MAC address).

Figure 46-4 provides an example of Stream transformation in the Talker end station. Stream transformation in the Listener end station is similar. The example of Figure 46-4 assumes use of the centralized network/ distributed user model (46.1.3.2). Use of the fully centralized model (46.1.3.3) is similar.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

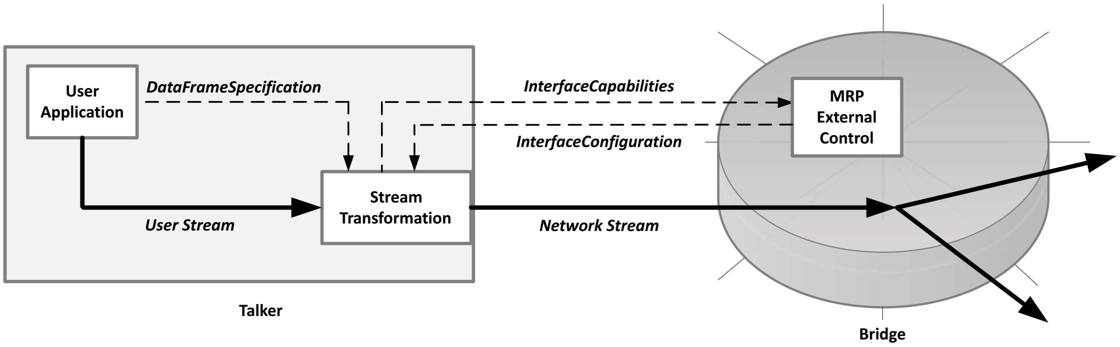

Figure 46-4—Example of Stream transformation in Talker end station

For Stream transformation in the end station, the end station’s interface to the network can provide the transformation capability, acting as a network entity within the user’s end station.

The identification of the Stream in the user originates from the User Application block, specified by the Talker as the DataFrameSpecification (46.2.3.4). The end station knows this user identification, but not the identification that the network requires. To negotiate the network identification, the Talker uses SRP to transmit InterfaceCapabilities (46.2.3.7) that describe its Stream transformation capabilities to the nearest Bridge. The Bridge uses MRP External Control (12.32.4) to send the InterfaceCapabilities to a CNC. The CNC consults its configuration of network identification and uses MRP External Control to send InterfaceConfiguration (46.2.5.3) along with successful Status (46.2.5) back to the Bridge. When the Bridge receives this information, it propagates back to the Talker using SRP. The InterfaceConfiguration provides the network identification, which the end station uses to perform Stream transformation for data frames.

NOTE 3—The network identification typically entails allocation of a group MAC address for the Stream. If a CNC is used, the CNC can allocate a group MAC address from a pool that it maintains.

Figure 46-5 provides an example of IEEE 802.1CB functions within the Stream Transformation block in the Talker end station. The example assumes that the user identification uses an Internet Protocol (IP) packet for identification and that the frame conveying the IP packet does not use the appropriate MAC address and VLAN tag for TSN features in Bridges~~ (e.g., IP packet unicast destination MAC address, untagged)~~. The IEEE 802.1CB function for IP Stream identification uses fields of the IP packet to identify the packet as a specific TSN stream. That stream identification is then applied to the IEEE 802.1CB function for Active Destination MAC and VLAN Stream identification to replace the destination MAC address and add a VLAN tag for TSN Bridge features. The IP fields of the packet are not changed. The IEEE 802.1CB functions can be implemented in software (e.g., operating system driver) or in hardware.

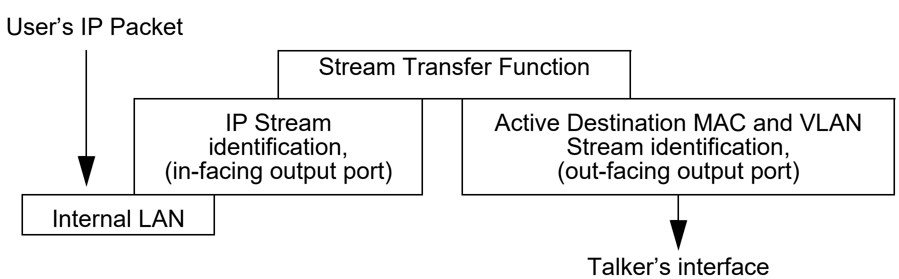

Figure 46-5—Example of IEEE 802.1CB functions in Talker end station

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

Figure 46-6 provides a corresponding example of IEEE 802.1CB functions within the Stream Transformation block in the Listener end station. In this direction, the IEEE 802.1CB function for Active Destination MAC and VLAN Stream identification provides both functions. The IEEE 802.1CB function uses the group destination MAC address and VLAN tag of the received frame to identify a specific Stream. The IEEE 802.1CB function then transforms the destination MAC address and VLAN tag to restore the Stream's frame to its original format (as transmitted by the Talker).

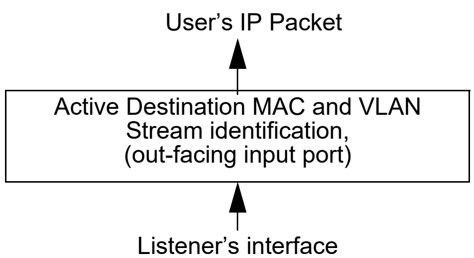

Figure 46-6—Example of IEEE 802.1CB functions in Listener end station

Insert 46.1.5, 46.1.6, and 46.1.7 after 46.1.4 as follows:

46.1.5 Centralized User Configuration

A Centralized User Configuration (CUC) delivers user requirements to the CNC. The CUC delivers information for configuring TSN features to end stations. It is a logical entity that can be located in any ~~station of a~~ network system.

The CUC is responsible for:

a) Reconciling the requirements from Talkers and Listeners to Stream requirements, if necessary.
b) Sending the Stream requirements to the CNC.
c) Receiving the end station communication-configuration from the CNC.
d) Distributing the end station communication-configuration to Talkers and Listeners.

NOTE–It is the responsibility of the CNC to ensure that Streams are assigned a unique StreamID group. For this a remote procedure call (RPC) RequestFreeStreamId (46.2.7.5) is available so the CUC can request a free StreamID from the CNC.

Stream requirements, in the context of the CUC, result from combining the Stream requirements of one Talker with the Stream requirements of one or multiple Listeners that, together, apply to form a Stream. Reconciling the requirements for the Stream does not change the parameters in the Stream request originating from the Talker or the Listener(s).

The end station communication-configuration that is received by the CUC from the CNC and then distributed to the Talkers and Listeners does not directly configure features on the end stations. It consists of configuration information that a CUC can provide for a Talker and Listeners to configure the Stream. An end station could, for example, make use of the information it receives in the communication-configuration from the CUC to configure an application in a way that ensures different TSN Streams are sent by the application in a specific order that correlates with the expected Stream's transmission on the network.

A CUC affects only one Configuration Domain. Talkers and Listeners can only make use of the CUC to reconcile their Stream requirements into a Stream request, if they are part of the same Configuration Domain. If a Talker wants to communicate with one or more Listeners in a different Configuration Domain, this needs to be done through dedicated inter-domain communication mechanisms. Such inter-domain communication mechanisms are not specified by this standard.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

The protocols that the CUC uses for communication with end stations are not specified by this standard. A CUC exchanges information with a CNC in order to configure TSN features on behalf of its end stations. It communicates with the CNC through the CUC-CNC interface specified in 46.2. The CUC can request computation of paths and configurations for Streams in the following ways:

e) Request computation of the paths and configurations for a set of Streams, using the protocol operation described in 46.2.7.1. The computation is performed by the CNC on the complete set of Streams of this request. This allows for optimized scheduling of Streams in the network.
f) Request computation of the paths and configurations for new or modified Streams, using the protocol operation described in 46.2.7.2. The computation is performed by the CNC on all Streams in a Configuration Domain that have a StreamStatus (46.2.3.8) of either planned or modified.
g) Request computation of the paths and configurations for all Streams of a CUC, using the protocol operation described in 46.2.7.3. The computation is performed by the CNC on all Streams in a Configuration Domain that belong to the CUC specified in the request.
h) Request the joining addition of a set of Listeners to an already existing Stream. The paths are extended to allow forwarding of the Stream to the new Listeners. Computation for the changes has to be triggered via RPC.
i) Request the removal of an existing Stream, using the protocol operation described in 46.2.8.1.
j) Request the removal of one or more Listeners from an existing Stream. Computation for the changes has to be triggered via RPC.

A CUC can be present for initial configuration, to manage changes to a running network, or both. Multiple CUCs can co-exist and operate in parallel in the same Configuration Domain as shown in Figure 46-3.

46.1.6 Centralized Network Configuration

The Centralized Network Configuration (CNC) is a logical entity that configures network resources on behalf of applications (users) and can be located in any station of a network.

The CNC is responsible for:

a) Receiving the Stream requirements for one or more Streams from the corresponding CUC.
b) Providing a way for a CUC to request a free StreamID.
c) Assigning a unique destination MAC address in the Configuration Domain it is responsible for to each of the requested Streams.
d) Computing paths for requested Streams.
e) Performing computation of scheduling and/or shaping configuration for the requested Streams.
f) Configuring the network devices to provide the required resources for the Streams (e.g. FDB entries, configuration of transmission gates, etc.), using remote management.
g) Providing the end station communication-configuration for the Streams to the corresponding CUC. If the paths for the Streams impact existing Streams the CNC is also responsible for providing that information to the CUCs that originally requested the impacted Streams.
h) Removing of Streams as requested by a CUC.
i) Discovering physical topology, using remote management.
j) Retrieving Station capabilities, using remote management.

The CNC communicates with a CUC through the CUC-CNC interface specified in 46.2. It communicates with the stations using the managed objects defined in this and other IEEE 802.1 standards. There can only be one active CNC per Configuration Domain.

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

46.1.7 Configuration Domain

A Configuration Domain is a set of stations that are under a common configuration and management scheme, and a single administration. The Configuration Domain provides boundary information for the common management scheme and ~~in~~ to support of the responsibilities of the CUC and CNC regarding Streams. Whether a CNC and one or more CUCs are present in a Configuration Domain depends on the TSN configuration model (46.1.3) that is used in the domain (e.g., whether the fully centralized model or a different configuration model is used). The CNC and the CUCs required for the configuration of a Configuration Domain affect only one Configuration Domain.

46.2 User/network configuration information

~~This subclause specifies~~ the user/network configuration information that is used for the three TSN configuration models (46.1.3)~~. The semantics for the TSN user/network configuration information is specified independent of schema, encoding, or protocol.~~

~~A schema or encoding of a protocol for the TSN user/network configuration information will reference 46.2 as part of its normative specifications.~~ For the two distributed models, SRP specifies TLVs in 35.2.2.10 that reference information in 46.2. For the fully centralized model, the YANG module in 46.3 references information in 46.2.

Within this subclause, the word element refers to a single item of information used for TSN configuration. The word group refers to a collection of related elements. ~~Groups are organized hierarchically, such that~~ a group can be contained within another group. A single low-level group can be contained within multiple higher level groups. The dot-separated notation is used to refer to a specific element in text. For example, “Talker.StreamID.UniqueId” refers to the UniqueId element of the StreamID group that is contained within the Talker group.

~~This subclause specifies each group in a table. Each row of the table specifies an element of the group. Each element’s row specifies its~~ name, data type, and a reference to normative text that specifies its semantics (i.e., meaning). The data type of each element uses one of the values from 46.2.1. Other specifications such as conformance (i.e., required or optional), direction of transfer (i.e., user to network or network to user), default value, and range limitations are specified with normative text instead of with the table.

Each element name uses a camel-case naming convention (e.g., “MacAddress”) to align with naming conventions used in other clauses of IEEE Std 802.1Q. A specific protocol can use a different naming convention (e.g., 46.3) as long as the protocol’s name for the element can be associated with the element’s specification in 46.2.

46.2.1 Data types

~~The data type of each element is limited to semantics, independent of a specific encoding or protocol.~~ Data types in the tables include the following:

a) Boolean
b) int8, for a signed 8-bit integer
c) int16, for a signed 16-bit integer
d) int32, for a signed 32-bit integer
e) uint8, for an unsigned 8-bit integer
f) uint16, for an unsigned 16-bit integer
g) uint32, for an unsigned 32-bit integer
h) string

Copyright © 2023 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.