Resource Allocation Protocol (RAP) based on LRP for Distributed Configuration of Time-Sensitive Streams

Version 0.2, November 2017

Author: Feng Chen (chen.feng@siemens.com)
Contributors: Franz-Josef Goetz, Juergen Schmitt, Marcel Kiessling

<Revision History>

V0.1 is the initial version and available at [4]. The comments on V0.1 are documented in [5].

V0.2 contains the following major changes:

a. added description of the parallel L2/L3 reservation in subsection 2.5
b. added requirements on additional RAP features in section 3

Abstract

This document describes a Resource Allocation Protocol (RAP) that conducts distributed resource reservation for streams across a time-sensitive network in a manner comparable to the operation of Multiple Stream Reservation Protocol (MSRP). RAP is intended as an application to be built on top of the Link-local Registration Protocol (LRP), which is under development within the IEEE 802.1CS and provides enhanced support for large databases than the Multiple Registration Protocol (MRP). More importantly, RAP will add support for configuration of TSN features in the fully distributed configuration model, where MSRP currently supports only AVB features.

This document explores the new features of RAP enhancing the capabilities of the fully distributed TSN configuration. It documents the requirements and proposals for features that are contributed in terms of presentations, comments on the previous versions of this document or any other ways.

The purposes are to: 1) arouse interest in deployment of RAP for varied TSN application scenarios, 2) solicit opinions of the proposed features and suggestions for new ideas. It is not the goal of this document to describe the protocol behaviors and operations in style of a specification. The author assumes that the readers have some basic knowledges of MSRP.

1. Introduction

This section describes the solutions currently provided by the AVB and TSN standards (complete or in progress) to meet different requirements of specific markets. The gaps between the distributed and centralized solutions in support of TSN features are analyzed to show the need for a new Resource Allocation Protocol as a TSN-version of stream reservation protocol for fully distributed configuration model. The industrial real-time applications with such needs are also described.

1.1 Two solutions for guaranteed latency

To meet various QoS requirements of applications in different markets, the AVB and TSN standards specify a set of mechanisms and protocols, most of which relate to queuing/transmission functions and resource reservation techniques. To achieve guaranteed QoS, there are generally two categories of solutions as listed in Table 1. Each solution has emerged in different phase of standardizations primarily due to advancements in the queuing and transmission techniques on data-plane, whose properties have a great impact on the choice of an appropriate control-plane technique for configuration purposes.

Table 1: Two types of solutions of AVB/TSN for guaranteed QoS

	Reserved Streams (AVB)	Scheduled Transmissions
Data-plane techniques	Traffic shaping with Credit-based Shaper (802.1Qav)	Time-aware scheduler (802.1Qbv- Scheduled traffic)
Control-plane techniques	Distributed stream configuration (bandwidth reservation and resource allocation) using Stream Reservation Protocol (802.1Qat)	Centralized stream configuration (path/schedule calculation and resource allocation) using management with a central controller (802.1Qcc)
Target Latency	Bounded max. latency	Guaranteed lowest latency

The concept of “Reserved Streams” first emerged in the development of the AVB standards as a solution to provide guaranteed QoS for audio/video streams. The used techniques include traffic shaping with the Credit-based Shaper (CBS) specified in the IEEE Std 802.1Qav-2009 and distributed resource allocation with a Stream Reservation Protocol (SRP) specified in the IEEE Std 802.1 Qat-2010. Such a combination enables dynamic resource allocation for streams that are transmitted with a specified traffic class (Stream Reservation Class A or B) using the CBS and provides automatic stream configuration with Plug-and-Play support for AVB networks.

Pursuing a much lower guaranteed latency to meet tighter real-time requirements of automotive and industrial control applications than the AVB applications, the early work of TSN focused on developing a TDM-like scheme that deploys a repeating schedule to control transmission. Using the scheduled traffic specified in IEEE Std. 802.1Qbv-2015, transmissions of time-critical streams are protected in dedicated time slots to avoid interference with other traffic. However, Qbv provides only scheduling on a traffic class basis, not for individual streams within the same scheduled traffic. If guaranteed lowest latency is required at the stream or stream group level, in-traffic-class interference among time-critical streams has to be further eliminated. This requires a detailed scheduling of each stream at all stream sources (talkers) and high precision time synchronization between bridges and end stations. Calculating transmission schedules for both talkers and network in general is a complex computation task, which requires complete knowledge of stream transmissions on a specific network topology and even detailed knowledge of end station capabilities (like worst-case jitter of injected frames). Furthermore the quality of network synchronization has to be considered for schedule calculation.

Mainly driven by the need for guaranteed lowest latency achievable by applying scheduling to scheduled traffic, an approach that employs a single Central Network Controller (CNC) to conduct centralized computation (for paths, schedules and resources) and to configure bridges through remote management procedures has been specified in the 802.1Qcc. The work defines a set of parameters to be exchanged between end-station and network, which carry the information needed for configuration of the TSN mechanisms developed since AVB. However, the configuration of the TSN features is currently supported only for a network controlled by a CNC. Although an enhanced version of MSRP (MSRPv1) was specified in Qcc, its primary usage is a transfer protocol to carry stream configuration information over the link between each end-station and its nearest bridge that can be directly accessed by CNC. In such cases, MSRPv1 operates in a fundamentally different way from its original version that propagates information across the network and performs resource reservation hop-by-hop on each bridge. If MSRPv1 is applied in a fully distributed manner, it behaves essentially the same as the MSRPv0 with support only for CBS.

1.2 TSN for Industrial control applications with different real-time requirements

It is generally accepted that employing scheduled transmissions to achieve guaranteed lowest latency is best configured in a centralized manner using a CNC. However, there are also many use-cases that can be well satisfied with a more relaxed bounded latency and do not necessarily use the scheduled transmissions relying on a dedicated network controller.

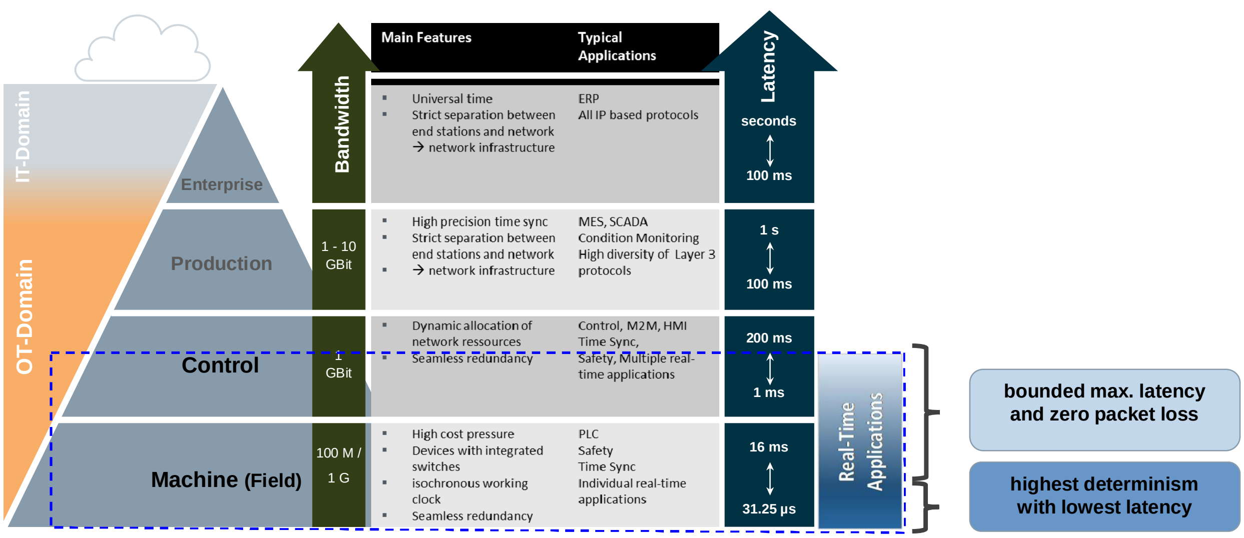

Figure 1 - Industrial real-time applications on the classical automation pyramid

Figure 1 shows the hierarchical organization levels of a factory that is often displayed as "Automation Pyramid". The real-time control applications in need of guaranteed QoS are mostly found on the control level and machine level (also called field level). Depending on the controlling tasks, the response time required by different control applications can vary in a wide range from hundreds of milliseconds down to tens of microseconds.

For hard real-time control systems with very strict low-latency requirements (typically lower than hundreds of µs), such as high-speed motion control on the field level, scheduled transmissions with centralized scheduling provide the ability to minimize the latency to the utmost extent through creating a completely interference-free channel for each stream transmission from talker to listener across the network. In such systems, all end-stations are also required to be strictly time-synchronized with the network components in order to coordinate the transmissions among all the talkers and to align the talker schedules to the network schedules. A detailed knowledge of the timing behavior of the network components is required and cut-through operation is the preferred bridge mode to minimize latency and jitter. To calculate schedules for talkers and network, the CNC must also have knowledge about the requirements and properties of the end-stations that execute real-time applications, whose configurations are typically managed by an engineering tool. To facilitate information exchange between the applications and the network controller, the fully centralized configuration model is specified by the Qcc to exactly address the needs of such systems.

In addition to the specialized hard real-time systems, there also exist a vast number of real-time applications having more relaxed latency requirements, usually in the order of magnitude of milliseconds as typical value for the machine-to-machine communication on the control level. For such applications, applying sophisticated scheduling to minimize interference for lowest latency is not deemed to be necessary. Employing a dedicated CNC would be an overdesign, if guaranteed latency and other performance goals are also achievable using the queuing and transmission mechanisms that do not rely on centralized stream reservation. Even though originally developed for AVB systems, the "reserved streams" concept is attractive to TSN-based industrial systems as well, mainly due to the ability of the stream reservation protocol in support of automatic stream setup with dynamic resource allocation. As described previously, centralized control with a CNC is so far the only option for configuration of the streams that use the queuing and transmission functions provided by the TSN standards, not yet enabled for the distributed scheme. If the stream reservation protocol can be further developed to add support for the TSN QoS functions, it would be an important complement to the current centralized TSN configuration models and expand the usability of TSN techniques for industrial markets.

2. Description of Resource Allocation Protocol Features

RAP is a stream reservation protocol that extends the capabilities of the MSRP previously developed by AVB and provides enhanced service to configure the TSN streams that use the QoS functions defined by the TSN standards. This section provides more technical details on the proposed RAP features that are described in the earlier presentations [2,3].

2.1 RAP Architecture

RAP defines an LRP application that provides the stream registration and resource allocation service for the TSN streams. Unlike MSRP specifying only application-specific components within a common architecture that is defined by MRP for all MRP applications, RAP as a stand-alone protocol is cleanly separated in the architecture from the underlying LRP that provides port-local service with data transport and database synchronization functions on a point-to-point link.

Figure 2 illustrates the architecture of RAP in the case of a two-port bridge and an end station. RAP defines a per-port *RAP-Application* component and on bridges a per-bridge *RAP Attribute Propagation* (*RAP-AP*) component, which are comparable to the *MSRP-application* component and the *MSRP Attribute Propagation* (*MAP*) component respectively[1].

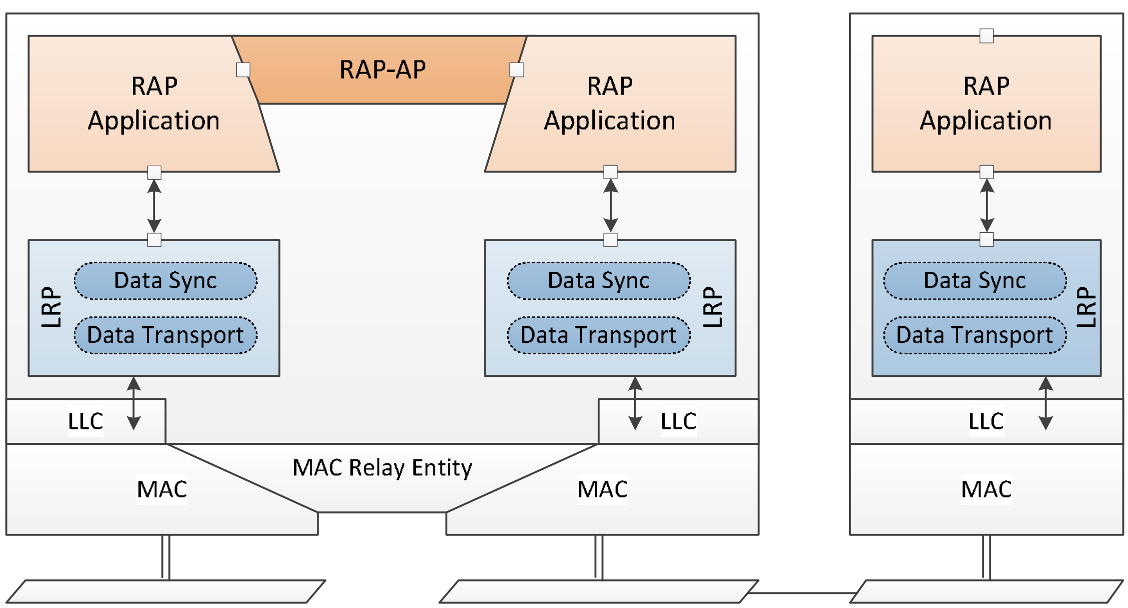

Figure 2 - Architecture of RAP on a Bridge and an end station

The *RAP-Application* component is responsible for the semantics associated with the RAP attribute values and their registration using the interface (in the form of primitives) provided by LRP[2]. The following are defined by the *RAP-Application* component:

---

[1] Outside the architecture of RAP, LRP plays a similar role as the MRP Attribute Declaration (MAD) component. Unlike MAD, LRP is application-neutral and contains no parts to be specified by each LRP application.

[2] The Applicant and Registrar primitives as interface between LRP and the application are not yet specified in the P802.1CS/D1.0 as the current LRP draft at the time of this writing.

a) A set of RAP Attribute types and values
b) The semantics associated with each RAP Attribute types and values
c) The structure and encoding of the records for registration with LRP[3]
d) The application ID for RAP to be identified by LRP
e) The service primitives provided to Talker or Listener applications above RAP (typically located at end station)
f) Establishment of RAP domain boundaries via exchanging the RAP Domain attribute containing SR class characteristics with the RAP-Application at the other end of the same link.

Within a bridge, the RAP-AP component is responsible for adjusting (if needed) and propagating the RAP attributes (except the RAP domain attribute) throughout the network. The following are defined by the *RAP-AP* component:

a) The contexts supported by RAP for attribute propagation, including the Base Spanning Tree Context (by RSTP) and a VLAN context (also with support for redundancy)
b) The rules for adjusting the Talker and Listener attributes before propagating them.
c) The rules for combing Listener attributes from multiple Listeners toward the associated Talker
d) For support of seamless redundancy, the rules for splitting and combining the Talker and Listener attributes, as well as configuring the FRER duplicate filter
e) The rules for updating the Dynamic Reservation Entries
f) Stream bandwidth calculations

Backward-compatibility

Mainly due to the LRP intended as an enhanced new registration protocol, a RAP application running on one side of a local link is not supposed to support direct communication with an MSRP Participant using the MRP procedures on the other side of the same link. However, for a specially designed bridge, RAP can specify a RAP-MSRP adapter, which provides translation between the RAP attributes and the MSRP attributes. One possible location for such an adapter can be located between the RAP-AP and a MSRP Participant implemented on the bridge port that is connected to a MSRP-only device (either bridge or end station), which is illustrated in Figure 3. On that port, the RAP-MSRP adapter can be connected to the MSRP Participant using the service primitives specified by MSRP in 35.2.3 of 802.1Q-2014. In this way, no changes to the present MSRP specifications are needed.

[3] As specified in 6.3.1 of P802.1CS/D1.0, the interaction between LRP and the applications is defined in terms of adding, withdrawing, or altering whole records. As the RAP attributes are organized in accordance to the stream reservation service provided to applications above the RAP following a similar concept as the MSRP attributes, the RAP records represent the data units to be handled by the underlying registration service provided by LRP. RAP will specify the rules of reorganization of the RAP attributes into the RAP records, while each record can be either a subset of the data (fragment) of a RAP attribute or an aggregation of multiple attributes. This part will be added in the future version.

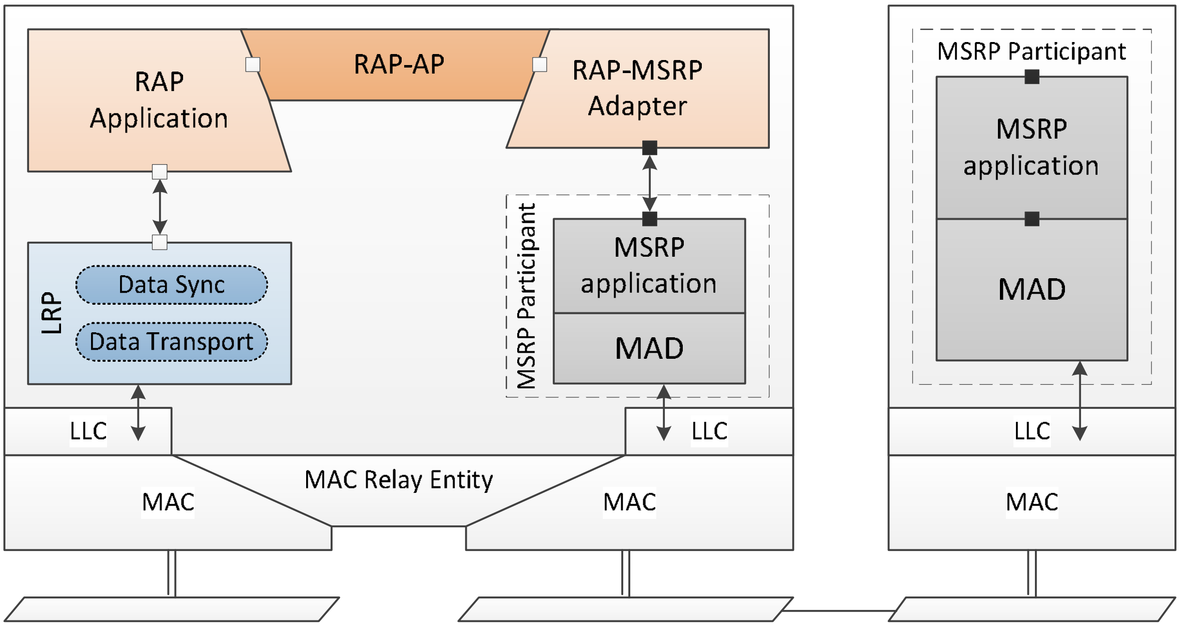

Figure 3 - Bridge in support of both RAP and MSRP for backward-compatibility

2.2 RAP Attributes

Figure 4 shows the organization of the RAP attributes, which extend the MSRPv0 attributes by adding the items needed for RAP to support the features described later in this document and including also some of the items defined by Qcc for MSRPv1. As illustrated in Figure 4, the MSRPv1 attributes contain many other items (in green color), which are defined exclusively for CNC to gather user requirements and to return configuration results back to users in a centralized configuration model. Thus, these items in the MSRPv1 attributes are not supported by a distributed stream reservation protocol.

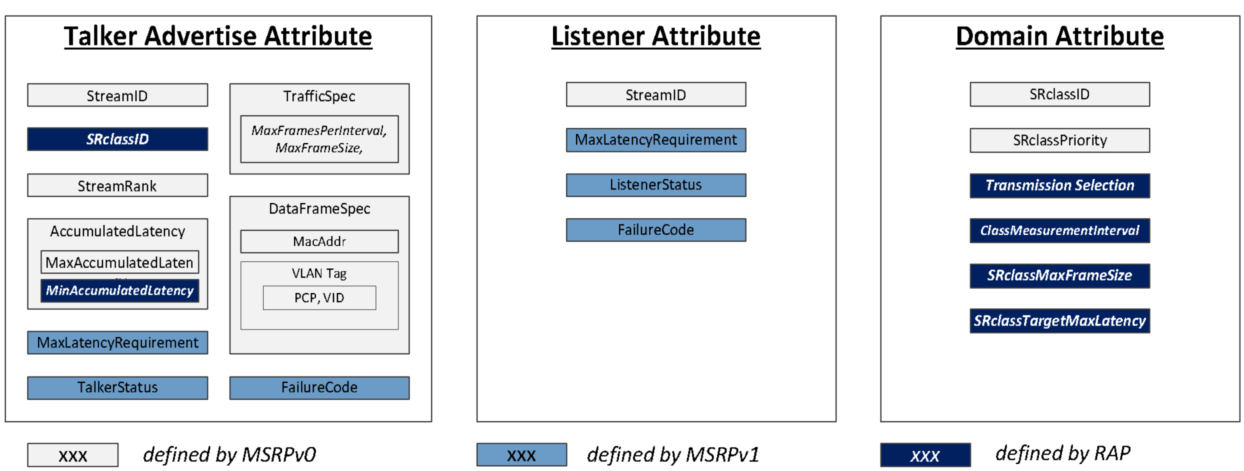

Figure 4 - RAP attributes

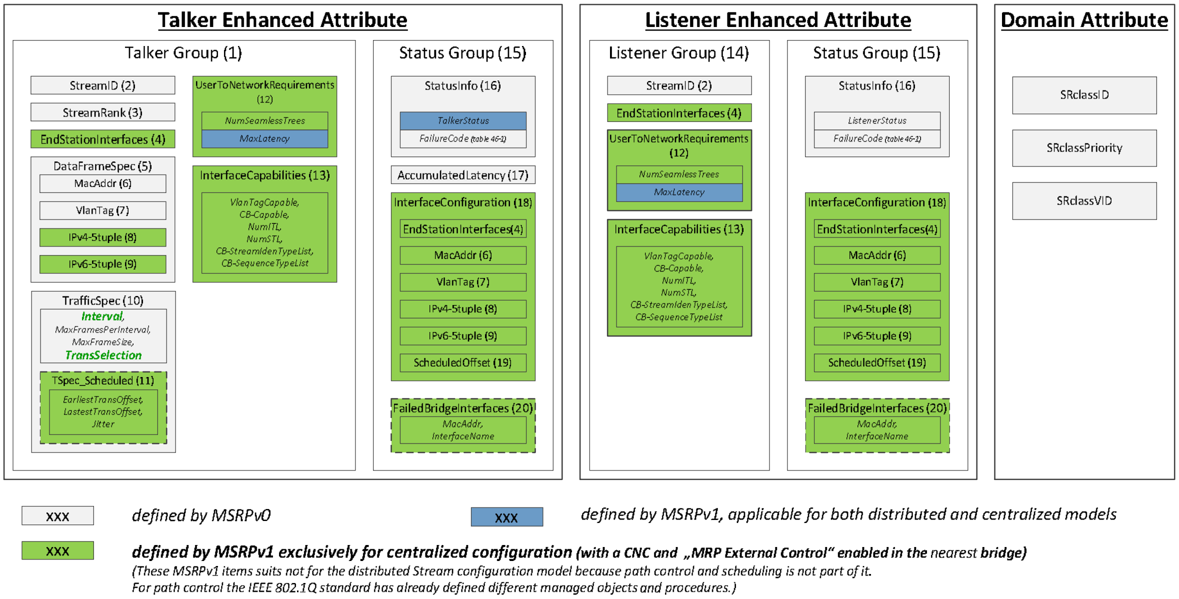

Figure 5 – MSRPv1 attributes

2.3 RAP Domain Detection for Enhanced Stream Reservation Classes

Originally defined by AVB in the IEEE Std 802.1Qav-2009, stream reservation class (SR class) represents a specific traffic class on a Bridge[4] whose bandwidth can be reserved for audio/video (AV) traffic. Two SR classes, SR class A (using priority 3) and SR class B (using priority 2), are specified for use with the CBS in the AVB networks, each supporting a class measurement interval at 125µs and 250µs respectively. The AVB SR classes present rather fixed settings; the shaper is tied to CBS and the class measurement interval is unchangeable by management.

Amended by Qcc, a set of managed objects are added to enable fully configurable SR classes. One could use management procedures to change the default settings of the SR classes A and B, e.g. remapping to a transmission selection algorithm other than CBS or assigning a different priority value. Such configurability also applies to the remaining SR classes from C to G, which are unspecified by AVB. Table 2 shows a comparison between the SR classes for AVB and their enhancements for TSN.

[4] Although traffic class is on a per-Port basis, SR class is currently required to be on a per-Bridge basis, because neither a bridge-internal SRP domain boundary nor behavior for propagating registrations between two Ports on the same Bridge with different SR class settings has been specified by MSRP.

Table 2: Stream Reservation Classes in AVB and TSN

SR class parameters	AVB	TSN (Qcc)
Usable SR classes	SR class A and B (SRclassID 6 and 5)	SR class A to G (SRclassID 6 to 0) in Table 35-7 of Qcc
Transmission selection algorithm	Credit-based shaper	Configurable via managed objects in Table 12-5 of 802.1Q-2014
Class measurement interval	125 µs for class A 250 µs for class B	Configurable via managed object *classMeasurementInterval* in Table 12-4 of Qcc
SR class to Priority mapping	3 for class A 2 for class B	Configurable via managed objects in Table 12-7 of Qcc
Priority to Traffic Class mapping	Default mappings for class A and B in Table 34-1 of 802.1Q-2014, Changeable using managed objects in the Traffic Class Table in 12.6.3	

Based on the enhanced SR classes, RAP is intended to support stream reservation for a given SR class that is managed by the network administrator to use a desired configuration beyond the default settings. As the initial step in the process of stream reservation, domain boundary detection specified in 34.2 of 802.1Q-2014 establishes a reservation domain among contiguous devices that have the same settings for a given SR class and determines the domain boundary. This function is currently specified for the AVB SR classes and hence needs to be first extended for use in RAP to support the enhanced SR classes.

In MSRP, the domain negotiation process is carried out by exchanging the SR class parameters contained in the Domain Attribute between neighboring MSRP participants on each link. Because of almost unchangeable SR class parameters in AVB, the MSRP Domain Attribute carries three parameters as listed in Table 3 and in fact uses simply *SRclassPriority* in the consistency check for a given SR class identified by *SRclassID*. It is worth mentioning that although *SRclassVID* is present in the MSRP Domain Attribute, it is used as an informative value and won't be checked as an SR class parameter in the domain detection procedures.

To establish a domain for enhanced SR classes, the RAP domain detection needs to include also the SR class parameters, which are previously defined in AVB as fixed values and now become configurable. The parameters of the RAP Domain attribute, including two existing items in the MSRP Domain attribute and those additionally added for RAP, are listed in Table 3 and explained as follows.

- *SRclassID*: originally defined for Class A and B in 35.2.2.9.2 of 802.1Q-2014 and now extended by Qcc to add the values for SR classes from C to G.
- *SRclassPriority*: using 3 for SR class A and 2 for SR class B as default. Mapping of *SRclassPriority* to *SRclassID* for each supported SR class is configurable via managed objects in Table 12-7 of Qcc.
- *Transmission selection algorithm*: using CBS as default for SR class A and B. For each supported SR class, one can choose from[5] *Strict Priority (0)*, *Credit-based Shaper (1)* and *Asynchronous*

[5] Some queuing and transmission functions developed by TSN, such as scheduled traffic in IEEE Std. 802.1Qbv-2015 and cyclic queuing and forwarding in IEEE Std. 802.1Qch-2017, are not specified as transmission selection algorithm (Table 8-5) and need to be controlled using the managed objects defined in the corresponding standards.

Traffic Shaping[6] *(3)*, as specified in Table 8-5 of 802.1Q. Mapping a desired transmission selection algorithm to a given SR class is indirectly done via a series of mappings of *SRclassID* to Priority, Priority to Traffic class and Traffic class to Transmission selection algorithm, using corresponding managed objects as described in Table 2.

- *Class measurement interval*: using 125 µs and 250 µs as default value for SR class A and B respectively. Configurable for each supported SR class via the managed object classMeasurementInterval in Table 12-4 of Qcc.
- *SRclass maximum frame size*: represents the maximum frame size allowed for streams associated with this SR class. This parameter provides an upper limit on the frame size of this SR class to the talker, which may use it as a factor in choosing a desired SR class for its stream(s).
- *SRclass target maximum latency*: represents the bounded maximum latency offered by this SR class for its associated streams even if they are transmitted along the longest path in the network. The usability and computability of this parameter rely on the ability of the queuing/transmission function used by this SR class, which is required to provide bounded and topology-independent per-hop latency. Under this condition, the value of this parameter can be easily calculated according to the network diameter (max. hop count) and is typically done by a network administrator offline in the network design phase. At runtime, this value is carried in the Domain attribute to each talker, which may use it to choose a proper SR class for its stream(s). A new managed object is needed for management to configure this parameter.

Table 3: Domain Attributes in MSRP and Enhancements in RAP

MSRP Domain Attribute	RAP Domain Attribute
SRclassID	
SRclassPriority	
SRclassVID *(note: only informative, not checked as an SR class parameter in domain detection)*	- Note: SRclassVID is intended for a different use as described in 2.4)
-	Transmission Selection Algorithm
-	ClassMeasurementInterval
-	SRclassMaxFrameSize
-	SRclassTargetMaxLatency

The per-port per-SR class *RAPdomainBoundaryPort* parameter will be set to false, indicating the port as part of the RAP domain for that SR class, only when all the above parameters in the RAP domain attribute declared by that port for that SR class are found to have the same values as those registered (from its link partner) for the same SR class on that port. As mentioned in the footnote 4, MSRP does not support bridge internal domain boundary port and thus requires the SR class settings to be on a per-system basis. Since the managed objects associated with the parameters in the RAP domain attribute are configurable on a per-port basis, consistency in the SR class settings within a system with multiple ports needs to be taken care of by the management procedures to guarantee proper function of RAP, if the internal domain boundary feature is not supported.

[6] Specified in IEEE P802.1Qcr/D0.1 (work in progress)

2.4 RAP for Seamless Redundancy

Frame Replication and Elimination for Reliability (FRER) specified in the IEEE Std. 802.1CB provides a set of tools for redundant transmission of stream packets over a network. Although FRER is developed mainly as a seamless redundancy technique that can substantially reduce the probability of packet loss caused by equipment failures, it is generally expected to be used in combination with other QoS features provided by the TSN standards to offer bounded latency and zero congestion loss with additional high reliability. In such cases, each stream transmission controlled by the FRER functions also requires that the resources, including bandwidth and buffer space, are reserved at every hop along each of the redundant paths from talker to listener.

Currently, automatic resource allocation for the streams using FRER is only enabled on a network that is configured by a CNC, not yet supported by a stream reservation protocol like MSRP. If RAP is expected to support stream reservation for FRER, the following assumptions need to be made.

- Assumption 1: use pre-established redundant trees for RAP in support of FRER
  RAP itself is not concerned with the establishment of any active topology or VLAN topology for the creation of the multiple paths used for duplicate transmission of data frames. Such tasks are fulfilled by a path control protocol like ISIS-PCR defined in IEEE Std. 802.1Qca-2015 or a central controller through management procedures. This document assumes that a set of redundant trees needed for the operations of FRER and RAP are already established and installed in the network before RAP starts running. The redundant trees could be either static or the Maximally Redundant Trees (MRTs) that are computed and installed using the tools provided by Qca. Each redundant tree is identified by a VID, e.g. using distinct VIDs for MRT-Blue and MRT-Red.

- Assumption 2: use {StreamID, VID} to identify a reservation for a single member stream
  As MSRP has no support for duplicate transmission, each stream is transmitted along a single path from a talker to one (a point-to-point path) or more listeners (a point-to-multipoint path). Thus, a single *StreamID* can be used by MSRP as a control-plane parameter to identify both the path and the reservation made on it for that stream. In 802.1CB, a compound stream, which represents the end-to-end talker-to-listener relationship and is equivalent to the original stream, can be split into multiple member streams that are transmitted along different paths to listener(s). To distinguish among multiple member streams belonging to the same compound stream, FRER defines a *stream_handle* subparameter to identify each member stream, which however is used only internally on a local system. Since reservation is a network global or regional property associated with a single path, neither *StreamID* nor *stream_handle* can be used alone to identify a reservation made for a member stream. Considering that the VLAN ID (VID) is typically used as identifier for different VLAN topologies including that for redundant trees, this document assumes using a combination of *StreamID* (for end-to-end relationship, also for backward-compatibility to non-redundancy cases) and *VID* (for a specific path) to identify a reservation made by RAP for a stream using FRER.

Under the above assumptions, the fundamental task of RAP to support reservation for FRER can be described simply as how RAP propagates registrations for stream reservations and processes the RAP attributes in a multiple tree environment containing redundant trees. As described in 2.1, the RAP-AP component defines attribute propagation and processing rules for each supported RAP-AP context. Thus, the support of RAP for FRER turns into the tasks of defining the RAP-AP functions for a RAP-AP context that contains multiplies VIDs for redundant trees (referred to as a VLAN context with redundant trees).

The stream reservation procedures will be discussed for the following two FRER uses-cases, end-to-end and Ladder redundancy, both of which are already described in Annex D of the 802.1CB standard.

2.4.1 RAP for end-to-end FRER

The system implementing end-to-end FRER relies only on the ability of end stations to conduct the FRER operations including sequencing, splitting and merging of streams. As illustrated in Figure 6, the Talker produces a single compound stream, splits it into two member streams and transmits it along two disjoint paths to the Listener, which finally merges two member streams to a single one. The Red and Blue lines in the Figure represent two redundant trees each with a distinct VID, which are preconfigured and installed on all bridges. Each member stream is required to use either of the VID for transmission of its data frames. There are no FRER functions required for bridges in the network.

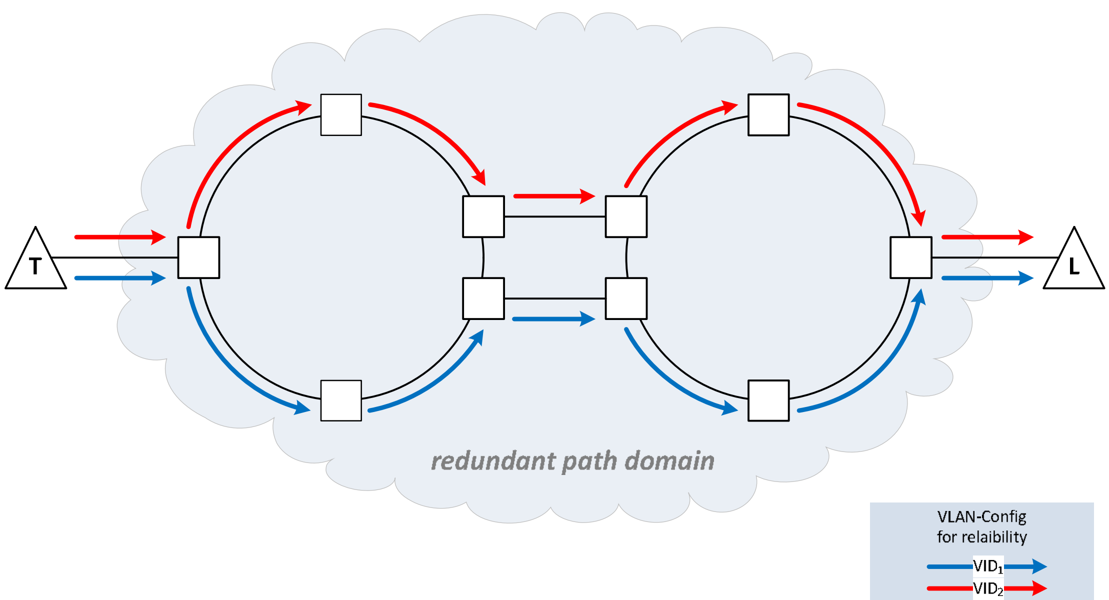

Figure 6 - End-to-end FRER

In such cases, RAP needs to make two reservations with one for each member stream. According to Assumption 2, each reservation will be identified by the {StreamID, VID} pair, which uses the same StreamID value (due to the same end-to-end relationship for both member streams) but a different VID value. The Talker first generates two RAP messages with Talker attribute, which have the same values

(StreamID, SRclassID, DA, priority, TSpec, etc.) but a different VID. The intermediate bridges receiving these Talker attributes need to propagate them only to the ports associated with the VID carried in each received Talker attribute. For the Listener attribute, they simply trace each path back to the Talker on the bridge ports where the corresponding Talker attributes have been previously registered (also called a context defined by Talker registration).

In summary, the reservation process done by RAP for end-to-end FRER is almost the same as MSRP for a single stream, expect that RAP must treat different VIDs that are carried in the Talker attributes but using the same StreamID as distinct reservations. In MSRP, two Talker attributes with the same StreamID but different VID are handled as a failure case relating to the same reservation.

2.4.2 RAP for ladder redundancy

Figure 7 shows a network implementing ladder redundancy that can provide resilience to multiple failures. In contrast to the end-to-end FRER, two member streams, which in the shown case are split from a single stream by the bridge nearest to the Talker, will be repeatedly split and remerged within the network on every bridge located at the junction points of two rings (also called segments). In this way, the network can still provide connectivity even when every segment happens to have a single failure occurring at the same time. Such a form of network is usually referred to as *redundant coupling of rings*.

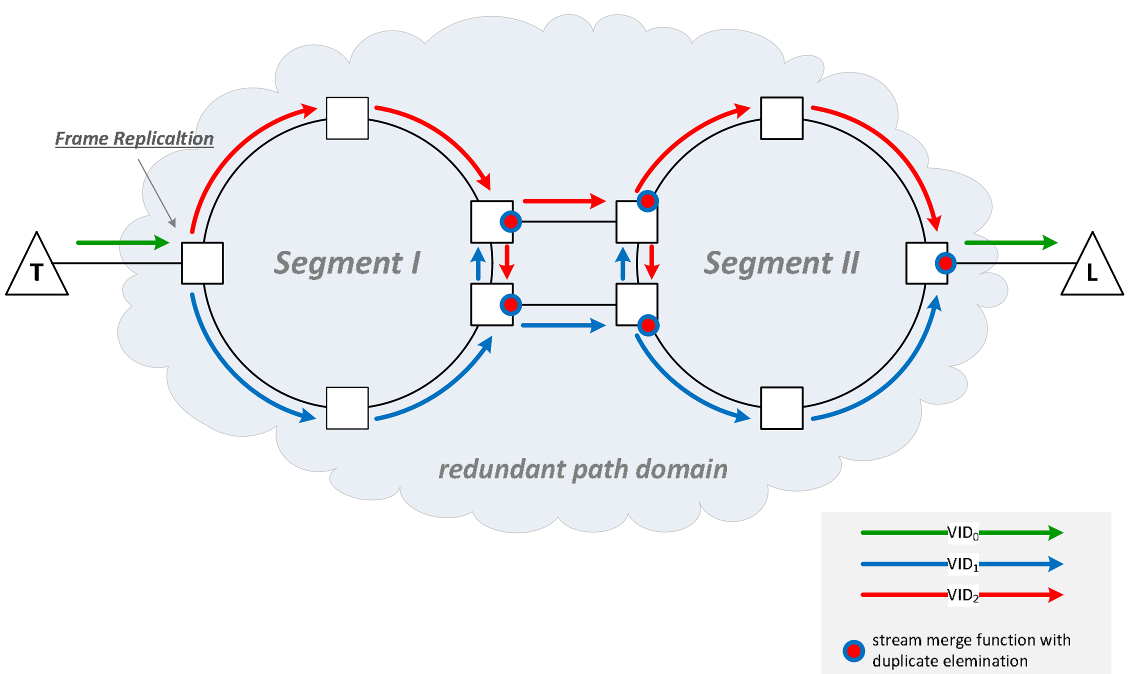

Figure 7 - FRER for ladder redundancy

A more complex example of Ladder redundancy is shown in Figure 8. Rooted at the bridge connecting to the Talker, two redundant trees (MRT-Blue identified by *VID1* and MRT-Red by *VID2*) are formed by a set of SPT (Shortest Path Tree) bridges that are capable of computing MRTs, for streaming from the Talker to two Listeners. Besides two MRT VIDs, *BaseVID* on the Green line represents the Base VID of the same VLAN and is intended for use on the network edges with connections to the end stations outside the SPT region. The VLAN configuration information is stored in the MST Configuration Table on each bridge and is provided to RAP that uses it as context to propagate the attributes.

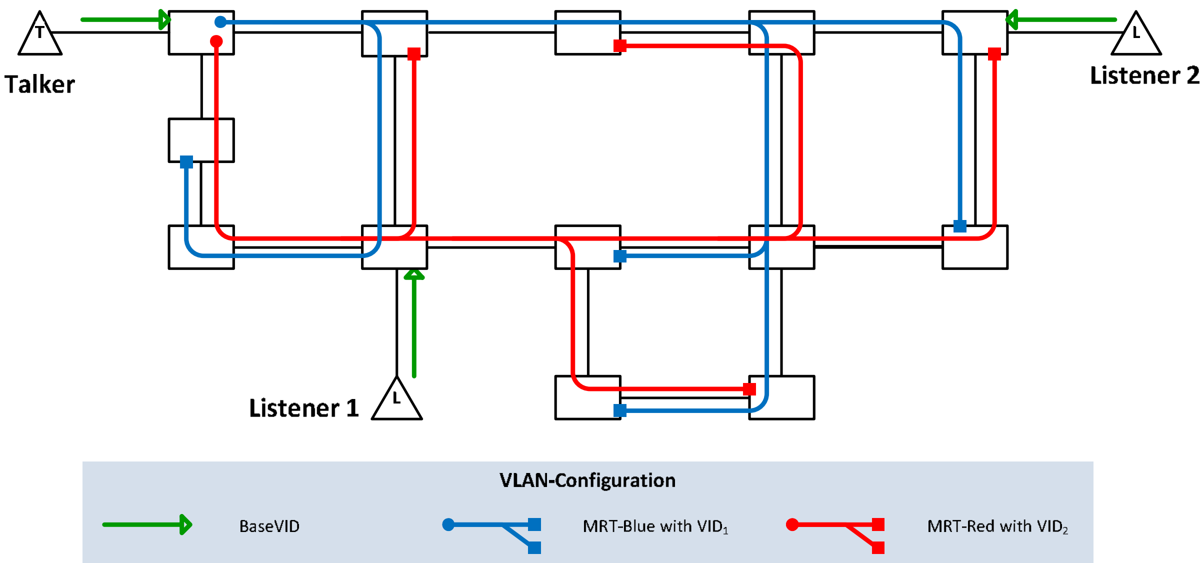

Figure 8 - Example VLAN configuration for ladder redundancy

The propagation of Talker attribute (TA) is illustrated in Figure 9 and described in the following steps.

1. The Talker sends one TA using *BaseVID* (Green TA) to its nearest bridge B1.
2. B1 registers the Green TA on the reception port and then splits it into a Blue TA containing *VID1* and a Red TA containing *VID2.*
    a. The Blue TA will be propagated within the network on each port associated with the MRT-Blue tree identified by *VID1.*
    b. The Red TA will be propagated within the network on each port associated with the MRT-Red tree identified by *VID2.*
3. The bridges with the ports connecting to the Listeners merge the Blue TA and the Red TA into one Green TA containing *BaseVID* and propagate it to the listeners.

During the Talker attribute propagation, the following Talker attribute elements may be changed at a specific location within the network.

- The *VID* field shall take the values as described in the above steps.

- The max and min values of the *AccumulatedLatency* field (see Figure 4) shall be calculated on each path independently. At the TA merge point, i.e. B5 and B8, the Talker attribute propagated to the Listeners shall take the larger value of the *max* values calculated on both Blue and Red paths as *MaxAccumulatedLatency* and the smaller one of the min values as *MinAccumulatedLatency*.
- The *TalkerStatus* field can have the value in *{TalkerReady, TalkerPartialFailed*[7]*, TalkerFailed}* and is assigned by each bridge along two paths independently. At the TA merge point, i.e. B5 and B8, the rules for merging the *TalkerStatus* values received from the both paths to a single value for propagation to the Listeners are as follows.
    - o *TalkerReady* on both Red and Blue => *TalkerReady* on Green
    - o *TalkerFailed* on either Red or Blue => *TalkerPartialFailed* on Green
    - o *TalkerFailed* on both Red or Blue => *TalkerFailed* on Green

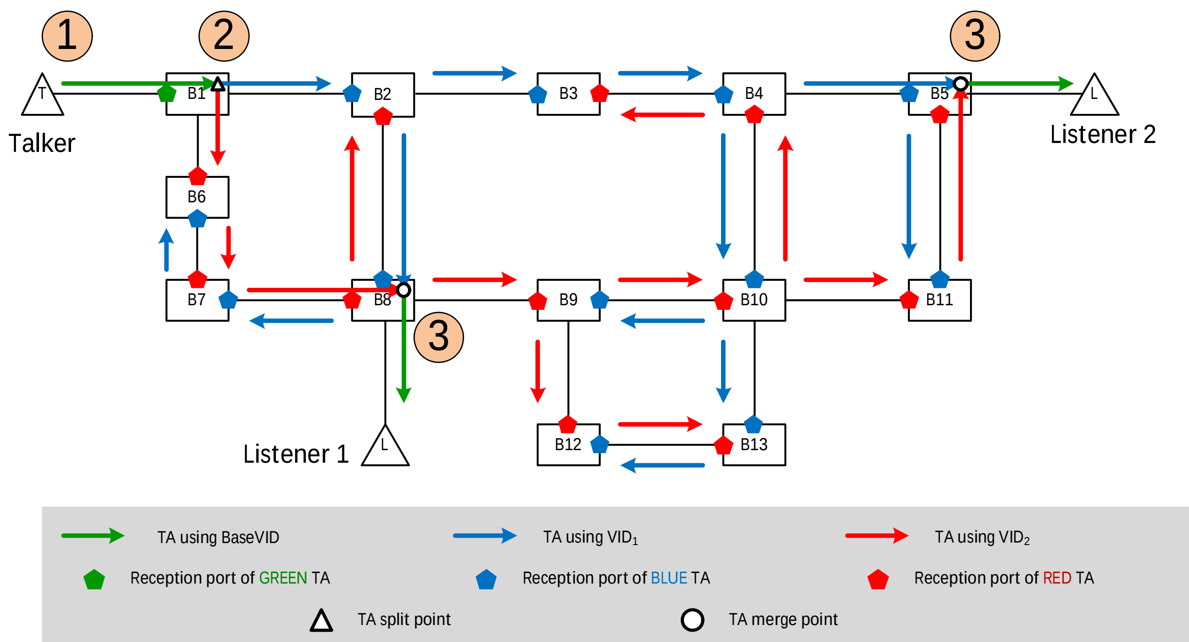

Figure 9 - RAP Talker attribute propagation for ladder redundancy

The propagation of Listener attribute (LA) is illustrated in Figure 10 and described in the following steps. It is worth noting that different to TA, LA contains no VID field and is simply propagated on the bridge ports where the Talker attributes have been previously registered for the same *StreamID*, without using any VLAN context.

[7] The TalkerStatus *TalkerPartialFailed*, which was not defined by MSRP, is needed for RAP to indicate the failure of at least one of the redundant paths to the end station, i.e. the Listener in this case.

1. Each Listener sends one LA to its nearest bridge.
2. Each bridge in the network propagates the LA on the ports that have registered Talker attributes with the same *StreamID*.
3. The FRER duplicate filter for data frames are activated on the reception port of a LA, when the LA needs to be propagated on more than one port on that bridge (also meaning in this case that the bridge has two ports that have registered TA associated with the same *StreamID* but using different *VIDs*.)

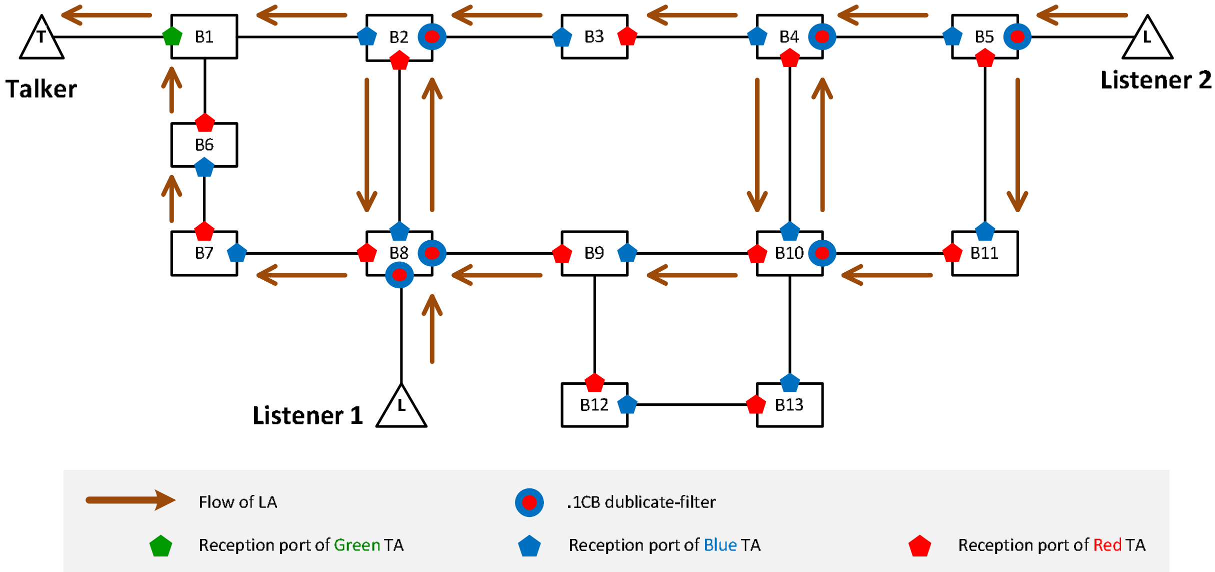

Figure 10 - RAP Listener attribute propagation for ladder redundancy

From the description above, in the case of ladder redundancy, end stations need not to be aware of the redundancy applied in the network (neither on the data plane nor on the control plane). This implies the ability of a RAP-enabled network to provide redundancy service to the end stations that implement MSRP without support for redundancy in the distributed manner.

[Note: as pointed out in the comment RC#13 documented in [5], RAP needs to specify mechanisms to inform the end-stations about the support of (also which type of) redundancy in the network as well as the corresponding VIDs used by the end-station. The next revision of this document is planned to provide more details on this issue.]

2.5 Collaboration with Upper Layer Reservation Protocol

Figure 11 shows a DetNet scenario where two end systems are connected to a DetNet network, which contains multiple L2 sub-networks that are interconnected by routers. The L2 sub-network, e.g. 802.1 TSN, is capable of providing DetNet compatible service for DetNet traffic. Offering the DetNet service with guaranteed latency and zero congestion loss to the data flow between the end systems requires resources to be reserved in all of the intermediate nodes including routers and bridges along the path of the flow.

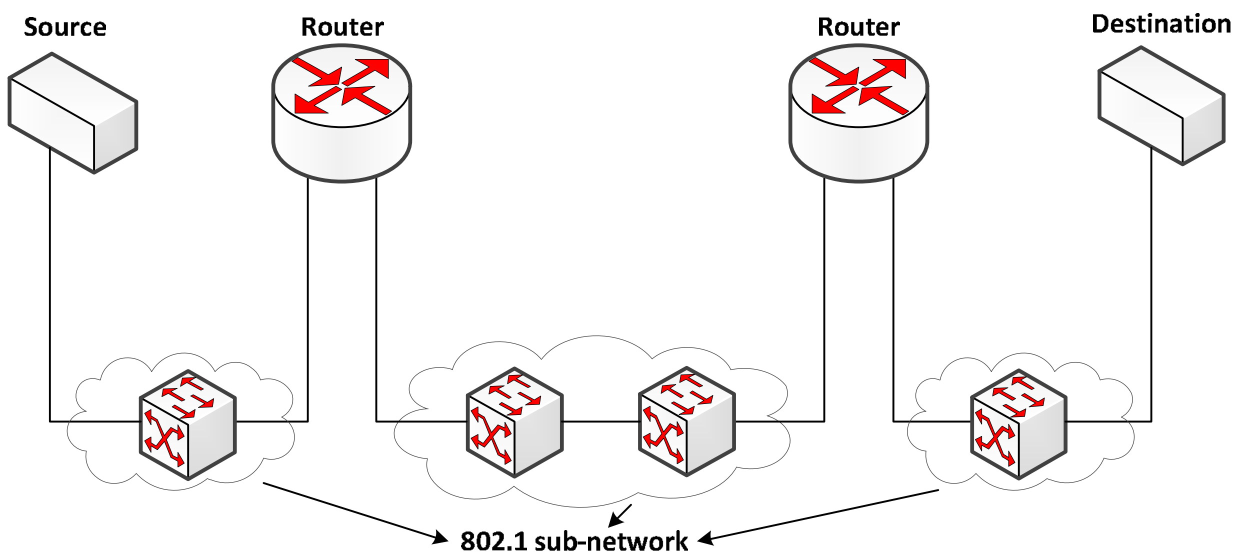

Figure 11 – A DetNet domain containing L2 sub-networks

Intended as a Layer 2 protocol handling resource reservation within a bridged LAN, RAP can be employed in collaboration with a higher layer reservation protocol such as RSVP to reserve resources in such a mixed L2/L3 environment. There are generally two options for RSVP/RAP interworking, either in parallel or serial operation, which are distinguished by whether the higher layer information is carried by the L2 reservation protocol (i.e. RAP) across the L2 sub-network. [8]

Figure 12 shows an example of the parallel operation. The parallel reservation can be viewed as a normal RSVP e2e signaling procedure running in the end systems and routers at Layer-3, which triggers a RAP signaling process within the L2 sub-network. Interaction of two reservation protocols occurs at the ingress and egress of the L2 sub-network, where there are typically end systems or routers that implement both protocols and support functions such as mapping of attributes and coupling of signaling states. Both RSVP and RAP flows physically pass through the L2 sub-network, which however is treated by the RSVP flow as a transparent L2 link between two RSVP peers.

[8] The earlier presentations on mixed L2/L3 reservation are available at <at-feng-SRP-RSVP-r3-060714.pdf> and <at-nfinn-msrp-sbm-rsvp-1207-v1.pdf>.

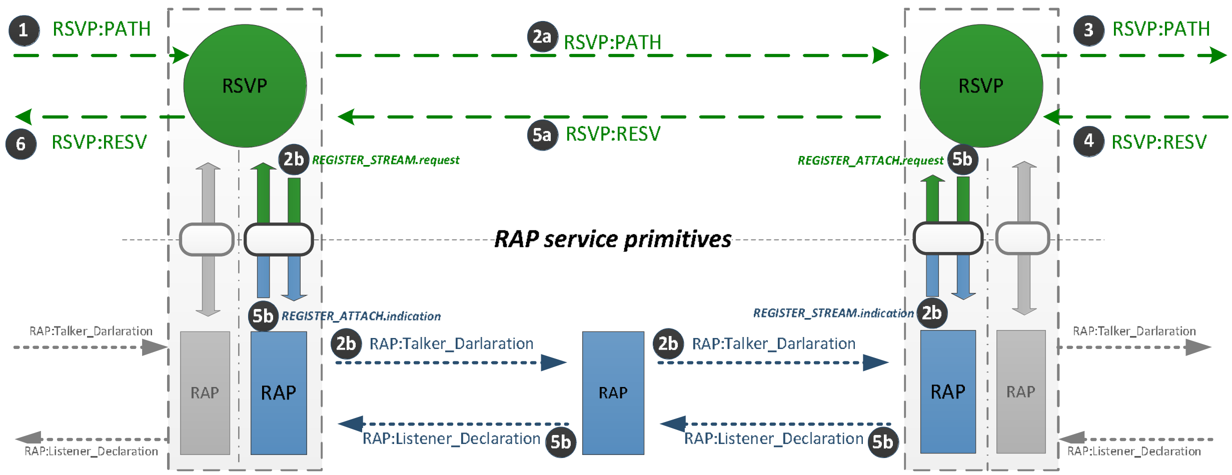

Figure 12 – Parallel operation of RSVP and RAP within a L2 sub-network

The reservation process in the parallel mode can be divided into the following steps, which are numbered in Figure 12 to indicate the sequence of the signaling procedures of two protocols as well as their interdependency. In the following description, two routers at both ends of the L2 sub-network that implement both RSVP and RAP are referred to as inbound/outbound routers according to the direction of the data flow.

1. The downstream *RSVP:PATH* message reaches the inbound router and triggers the parallel reservation.

2. Triggered by the incoming *RSVP:PATH*, the downstream flow will split into two sub-procedures at the inbound router.

   2a. RSVP continues to send the PATH message, which goes across the L2 sub-network to the next peer at the outbound router.

   2b. RSVP *triggers* the L2 registration process by using one of the Talker service primitives, i.e. the *REGISTER_STREAM.request*, provided by RAP. The rules for mapping the RSVP attributes to the RAP attributes are needed here. The RAP entity of the inbound router initiates the stream registration procedures by sending the RAP message with Talker_Declaration within the L2 sub-network.

   The receipt of the Talker attributes at the RAP entity of the outbound router will be notified to the upper layer via the *REGISTER_STREAM.indication* primitive. The Talker declaration type carried in this primitive, either Talker Advertise or Talker Failed, will indicate the the current state of the Talker Declaration registration in the L2 sub-network to the RSVP entity to take appropriate actions.

3. The outbound router refers to the current state of the RAP Talker Declaration progress to update the *RSVP PATH* process. For example, if RAP reports a Declaration type with Talker Failed,

which indicates there are bandwidth constraints or other limitations somewhere along the path in the L2 sub-network, RSVP may issue a *PATH_ERR* message upstream to the sender host. A very strict order between the RSVP and RAP downstream processes, for example RSVP in the outbound router should not send out the *RSVP:PATH* message further downstream until it is once notified by the associated RAP indication primitive, may not be necessary, because no reservation will be made at this stage of either signaling process.

4. The upstream *RSVP:RESV* message is received by the outbound router.

5. Triggered by the *RSVP:RESV* message, the upstream reservation process splits into two flows at the outbound router.

   5a. RSVP continues sending the *RSVP:RESV* message across the L2 sub-network upstream to the inbound router.

   5b. RSVP triggers the L2 reservation process by using the RAP Listener service primitives, i.e. the *REGISTER_ATTACH.request*. The rules for mapping the RSVP attributes to the RAP Listener attributes are needed here. RAP reserves the resources at each bridge hop-by-hop along the reserve path of the associated Talker registration.

   The receipt of the Listener attributes at the RAP entity of the inbound router will be notified to the RSVP layer via the *REGISTER_ATTACH.indication* primitive. The RSVP component will rely on the information received in the indication primitive, i.e. the Listener declaration type, to know about the reservation results within the L2 sub-network and then to determine how it proceeds with the RSVP reservation process.

6. The inbound router assumes a successful reservation (upstream till this router) by reserving the router resources and propagating the *RSVP:RESV* message further upstream, only when both RSVP and RAP reservation procedures are confirmed to be successful (or partially successful). Under this state coupling rule, RSVP will never grant the DetNet flow at the source to start transmission unless a successful reservation is made in the L2 sub-network. A reservation failure status, e.g. a Listener Asking Failed reported by RAP, may result in a *RSVP:RESV_ERR* message sent from the inbound router to the destination instead of an upstream *RSVP:RESV* propagation.

As described above, the parallel operation requires running both protocols in parallel on L2 and L3 in each L2 sub-network located between two L3 nodes. The interaction between two protocols is realized through the service primitives provided by the L2 protocol to the higher layer protocol, which achieves layer independence. In this way, no higher layer information, such as IP addresses, needs to be conveyed on the L2 control layer. The rules and operations needed for translating attributes and coupling states between two protocols need to be specified and executed by the higher layer protocol. This helps ease the workload of the L2 reservation protocol and allows RSVP to collaborate also with the existing MSRP in an AVB sub-network.

3. Requirements on Additional RAP Features

This section provides a brief outline of the proposals for additional RAP features, which have been mentioned in other presentations, submitted as comments on the previous versions of this document or discussed during the IEEE 802.1 TSN F2F meetings/calls. The listed features are described in a concise manner based on the author's personal understanding. In order for the task group to decide which of the proposed features are considered as part of RAP, a contribution with more details for each of the listed features is required.

3.1 RAP for Switchover Redundancy[9]

This feature can be useful for some industrial use-cases, when seamless redundancy is not deemed necessary and instead automatically switching of streams to a backup path in case of failure is sufficient enough to meet their real-time requirements. For such use-cases, RAP can be developed to reserve resources for critical streams on (at least) two paths, one on the primary path (initially activated for stream transmission) and the other on the standby path (initially inactive), typically on a ring topology. When the primary path fails, RAP will switch stream transmissions to the backup path by simply activating the forwarding on the bridges, which have already resources reserved and QoS functions configured for those streams. Such a feature supported by RAP can provide faster switch-over time than the current MSRP, while the latter one needs to first tear down the reservations on the failed path and then make new reservations on the new path recalculated by a given topology control mechanism. If RAP supports switchover redundancy, end-to-end re-reservation for the streams after experiencing a single network failure can be avoided and deterministic service can be further provided for the real-time applications during and after switchover. The needed changes are only applied within (a portion of) the network automatically and are not necessarily perceptible by the applications in end stations that produce or consume the streams.

3.2 DetNet's Needs for A Stream Reservation Protocol

The presentation[10] given by Norman Finn describes the needs of DetNet for additional work to be done in TSN, including a stream reservation protocol built over P802.1CS LRP. The issue whether such a protocol should support both L2 and L3 reservations needs to be decided.

3.3 RAP for Reservation in a Structured Industrial Network

A presentation on additional features needed in RAP will be given by Karl Weber at the Orlando plenary, Nov 2017.

[9] TSN for DetNet use-cases in need of reservation for switchover redundancy are described in <tsn-finn-tsn-detnet-whitepaper-0717-v00.pdf>

[10] The presentation on the needs of DetNet is available as <new-finn-detnet-needs-0917-v01.pdf>

Relevant Documents

[1] http://www.ieee802.org/1/files/public/docs2017/new-chen-RAP-proposal-and-requirements-0517-v02.pdf (presentation on RAP requirements at the Stuttgart interim meeting, May 2017)

[2] http://www.ieee802.org/1/files/public/docs2017/new-kiessling-RAP-poposal-and-features-0517-v01.pdf (presentation on RAP features at the Stuttgart interim meeting, May 2017)

[3] http://www.ieee802.org/1/files/public/docs2017/new-chen-RAP-details-0717-v03.pdf (presentation on RAP features at the St. John's interim meeting, September 2017)

[4] http://www.ieee802.org/1/files/public/docs2017/tsn-chen-RAP-whitepaper-0917-v01.pdf (RAP white paper version 0.1 uploaded in September 2017)

[5] http://www.ieee802.org/1/files/public/docs2017/new-chen-RAP-WP-discussion-TSN-call-1017-v01.pdf (discussion of comments on the RAP white paper V0.1 at the Oct. 02 TSN call)