802.1DD MSRP-like Architecture

A suggestion how to make 802.1DD Architecture more like MSRP

2025-05-22

v01

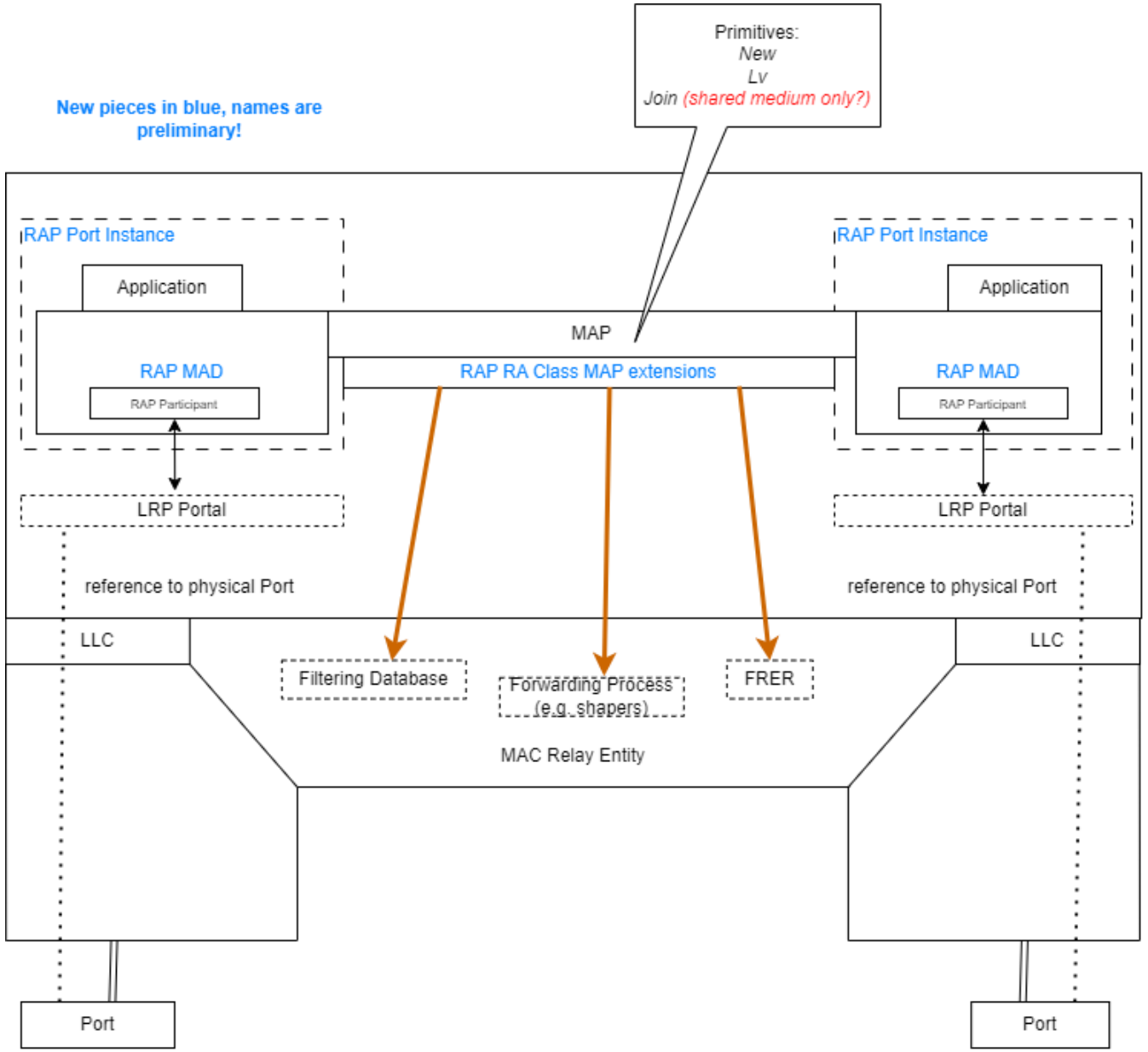
Andreas Meisinger, Siemens AG

RAP Architecture

RAP MAD / (extended RAP Participant)
- RA Class Attribute checks
  - Incoming Attribute Checks
  - Outgoing Attribute Checks
- Per Port Attribute Database
- Datarecord to Attribute Mapping
- Portal Event handling
- MAP Interface handling
- Provide Port to Portal Mapping by LLDP record information

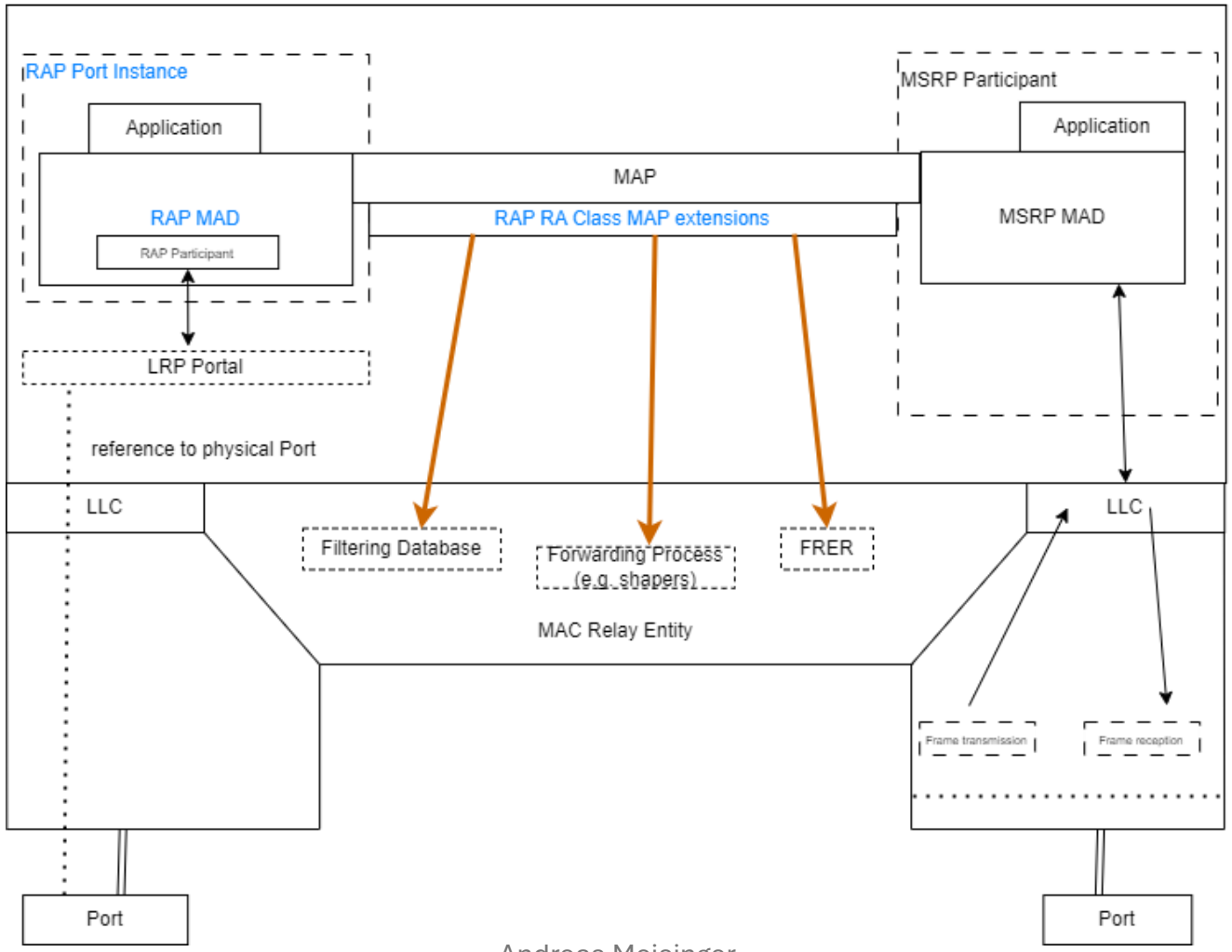
LRP Portal
A LRP portal is always bound to a specific physical port. Configuration of the portal is up to the RAP participant using the portal.

RAP (RA Class) MAP extensions
Extensions to MAP required for different shapers, redundancy, extensibility ...



Architecture combined with MSRP

New pieces in blue, names are preliminary!



Next Steps: Focus on basic cases

- RAP assumes a portal always being associated with a physical port
- RAP manages resources based on ports all the portal specifics and corner cases are kept within 802.1CS
- Redundancy stream resource allocation is expected to be RAP responsibility
- It is to be discussed at a later point in time how much of CB configuration is required to fullfill the resource allocation

Following „Changes“ are required in 802.1Q

- **They are not covered by P802.1DD PAR**

Can those be done in P802.1Q-2022-Rev ?

Issues to solve (technical)

(non comprehensive list)

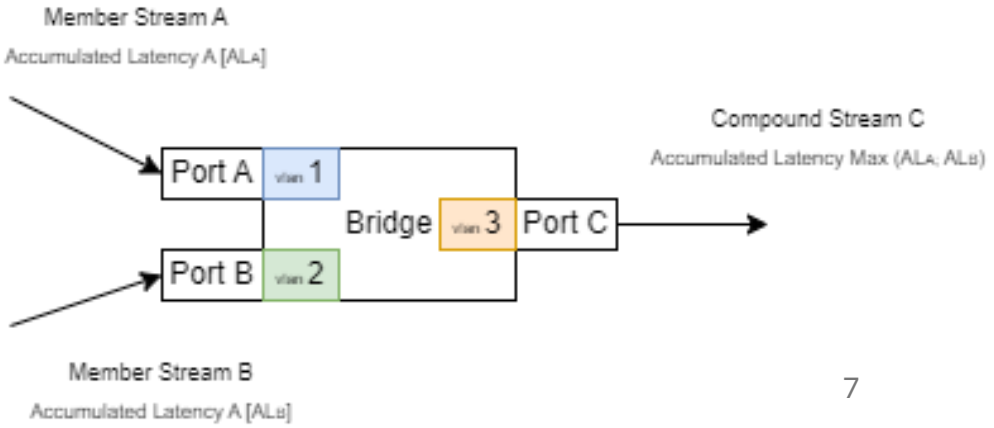
- 10.7.6.12+ MAP Interface references to MRP attribute state machines which can‘t be used without modification with LRP transport
  - Option A: not use the state machine as part of the MAP interface ➔ change in MSRP data architecture?
  - Option B: split Attribute state machine in MRP / Data Flow only parts
  - ?
- According to 35.2.4 MAP is responsible for resource allocation as well as configuration of the required mechanisms.
  - Only CBSA defined
  - Extension for other RA classes by RAP => MAP ressource allocation needs to be modifyable.
  - RAP aims to be extendable for future TSN technologies so an open approach is preferable
  - Inter RA class coordination is not for all combinations possible

Issues to solve (technical)

(non comprehensive list)

- Unclear where exactly (ingress, egress) accumulated latency calculation „MAP function is responsible for adjusting and propagating talker and listener attributes“?
  - Due to latency possibly beeing different between different in <-> out tuples the egress side of MAP would be reasonable

- 10.3.1/8.4 MAP Context is strictly bound to active topology. For RAP redundancy usecases „redundancy context“ has been introduced. This enables rap to adjust the active topology membership of a stream frame using CB mechanisms on specific bridges.
  Note: RAP uses externaly configured active topologies but it may adjust the attributes based on multiple topologies instead of a single one



Issues to solve (formal)

(non comprehensive list)

- **General: chapter 35 does explicitly enforce MSRP/MRP encoding in various places**
- 35.1.3 Behaviour of MRP Clause 10 required

  ➔ this requires some parts of MRP (e.g. MAP) to be modified too
- 35.1.4 SR classes are only allowed to be CBSA based
- 35.2.1.4 hard binding to msrp parameters e.g. msrpEnabledStatus msrpPortEnabledStatus
- 35.2.1.4 msrpNeighborProtocolVersion this parameter is used on other places too it might need at least a translation to some sort of rap version

Issues to solve (formal)

(non comprehensive list)

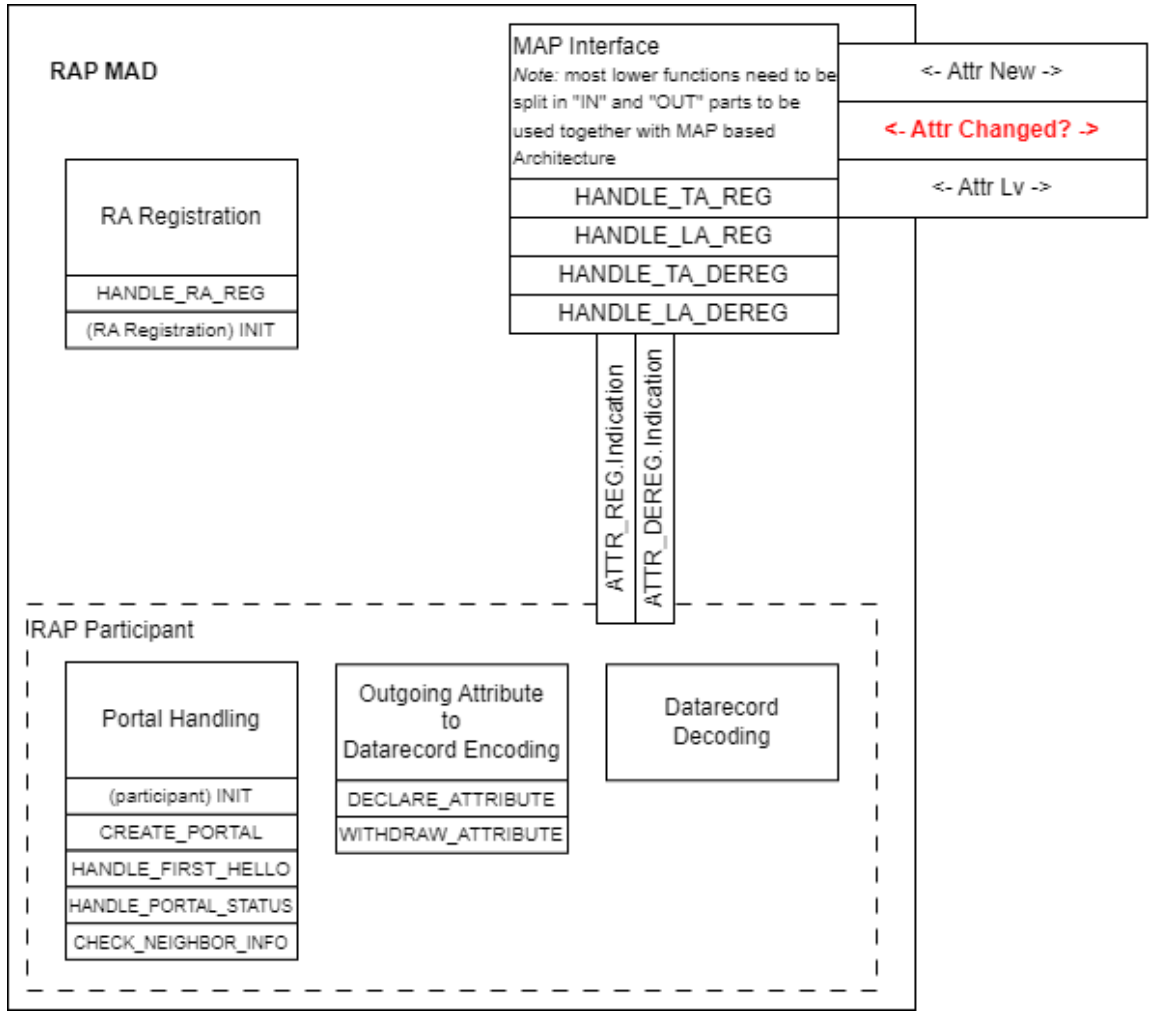
- 35.2.2 does explicitly state MRP requirements
- 35.2.2.4 these attributes might be reused in RAP or specified in a different format.
  - Option a: RAP doesn‘t extend the MSRP Attributes ➔ no support for other shapers than CBSA Possible
  - Option b: RAP tries to define format of other shapers similar to MSRP ➔ limited approach, a lot of redo of current rap draft
  - Option c: RAP supports carrying MSRP attributes enveloped in RAP tlv based format ➔ more overhead but could be compensated by more efficient LRP
  - **Option d: translation between RAP and MSRP attributes (for CBSA SR classes)**

P802.1DD Draft modification

- P802.1DD does already contain all features but in order to fit the proposed architecture some modifications have to be done
- RAP architecture changed from „per bridge“ to „per port“
  - Introduction of RAP MAD
- Propagation and resource allocation is moved to a MAP Extension

RAP MAD

based on previous architecture as well as existing RAP draft



MAP Interface

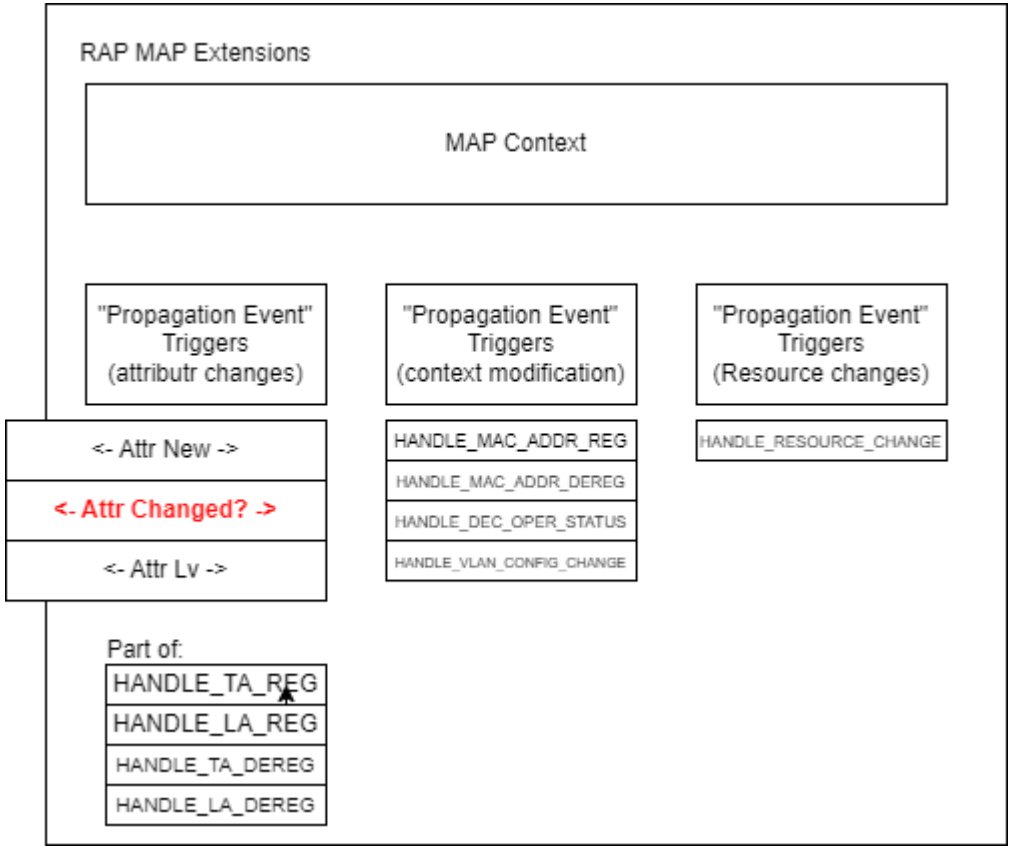
RAP does support changing of (some) stream parameters/attributes

It doesn‘t seem like this is already supported by the MRP MAP interface?

RAP (RA Class) MAP Extensions

Elements from current RAP draft to be considered for integration to MAP

MAP Interface



Thank You