

A blurred photograph of a modern office hallway with large glass windows and a central revolving door. Several people in business attire are walking through the hallway, their figures out of focus to convey a sense of movement and activity.

**SIEMENS**

Proposal for splitting IEEE 802.1Qcc into several PAR's

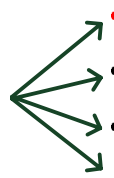
## **MRP++ Transport Protocol for Registration** **MSP Transport Protocol for Reservation**

Franz-Josef Goetz, Siemens AG  
Juergen Schmitt – Siemens AG

# Proposals for new PARs (1)

## Today:

**MRP – Multiple Registration Protocol**  
(framework for all MPR **applications**  
(e.g. MVRP, MMRP, ...) to register  
network attributes)

- 
- **MRRP – Multiple Relation Registration Protocol**  
(used to register relation ID's and to nail down the path in combination with BLCE's)
  - **MVRP – Multiple VLAN Registration Protocol**  
(used to register VID's – e.g. data plane for Streams)
  - **MMRP – Multiple MAC Registration Protocol**  
(used to register MAC addresses or "Group Filtering Behavior")
  - **MSRP – Multiple Stream Reservation Protocol**  
(used to register streams and make the reservation)

## PAR- Proposal for .1Qcc

PAR 1: New definition of .1Qcc

- **Maintenance for the existing MSRPv1 protocol and creating YANG modules for Clause 12 specs.**

## PAR- Proposals for MRP++ and MSP

PAR 2: MRP++

- **MRRP – Multiple Relation Registration application**  
(used to register relation ID's and to nail down the path in combination with BLCE's)
- **MVRP – Multiple VLAN Registration application**  
(used to register VID's – e.g. VID used for Streams)
- **MMRP – Multiple MAC Registration application**  
(used to register "Group Filtering Behavior" or to register MAC addresses)
- **MSRP – Multiple Stream Registration application**  
(used to register **(no reservation)** streams)

**MRP++ - Multiple Registration Protocol**

(Transport protocol for all MPR applications to register network attributes)

PAR 3: MSP

- **MSSP – Multiple Stream Signalling application**  
(used to Do stream **reservation**)

**MSP - Multiple Signalling Protocol** (related to RSVP)

(Transport protocol for all MSP applications to do reservations)

## Proposals for new PARs (2)

### Other PAR-Proposals

(contributed by  
Rodney Cummings)

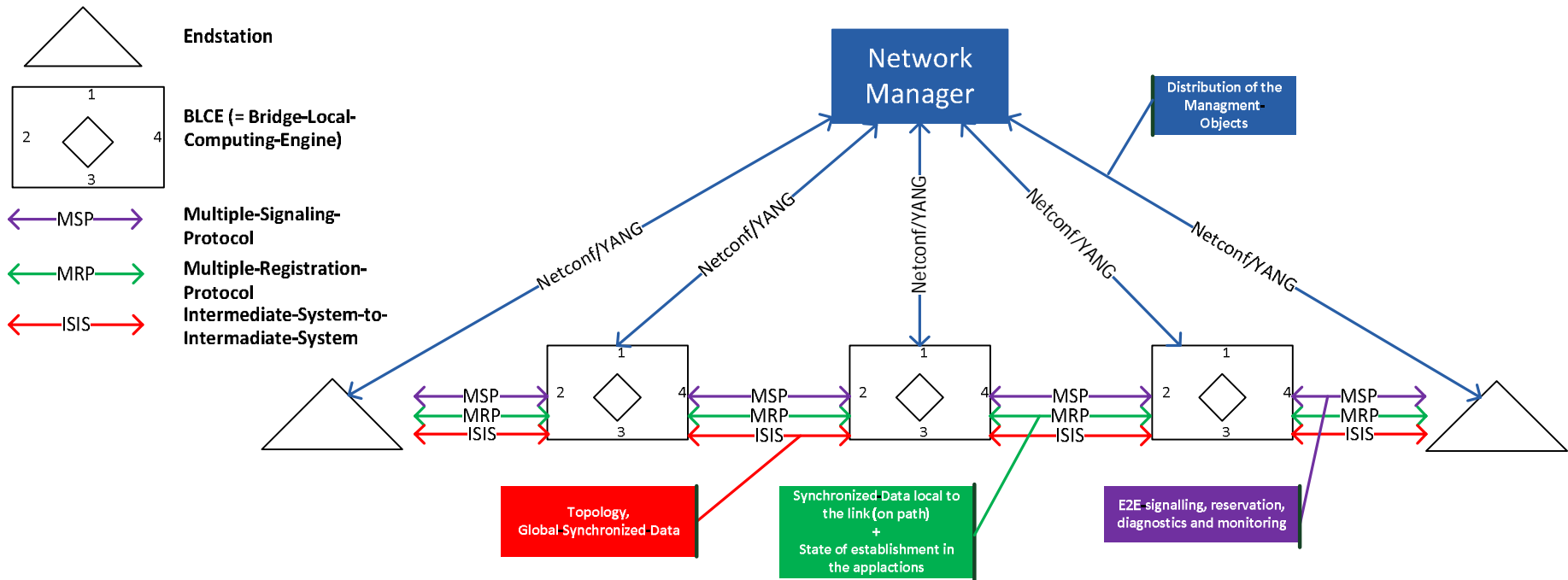
#### PAR 4: „device level“ YANG modules

- „This specifies YANG modules for 802.1Q Clause 12. This is the bottom level YANG (per bridge).“

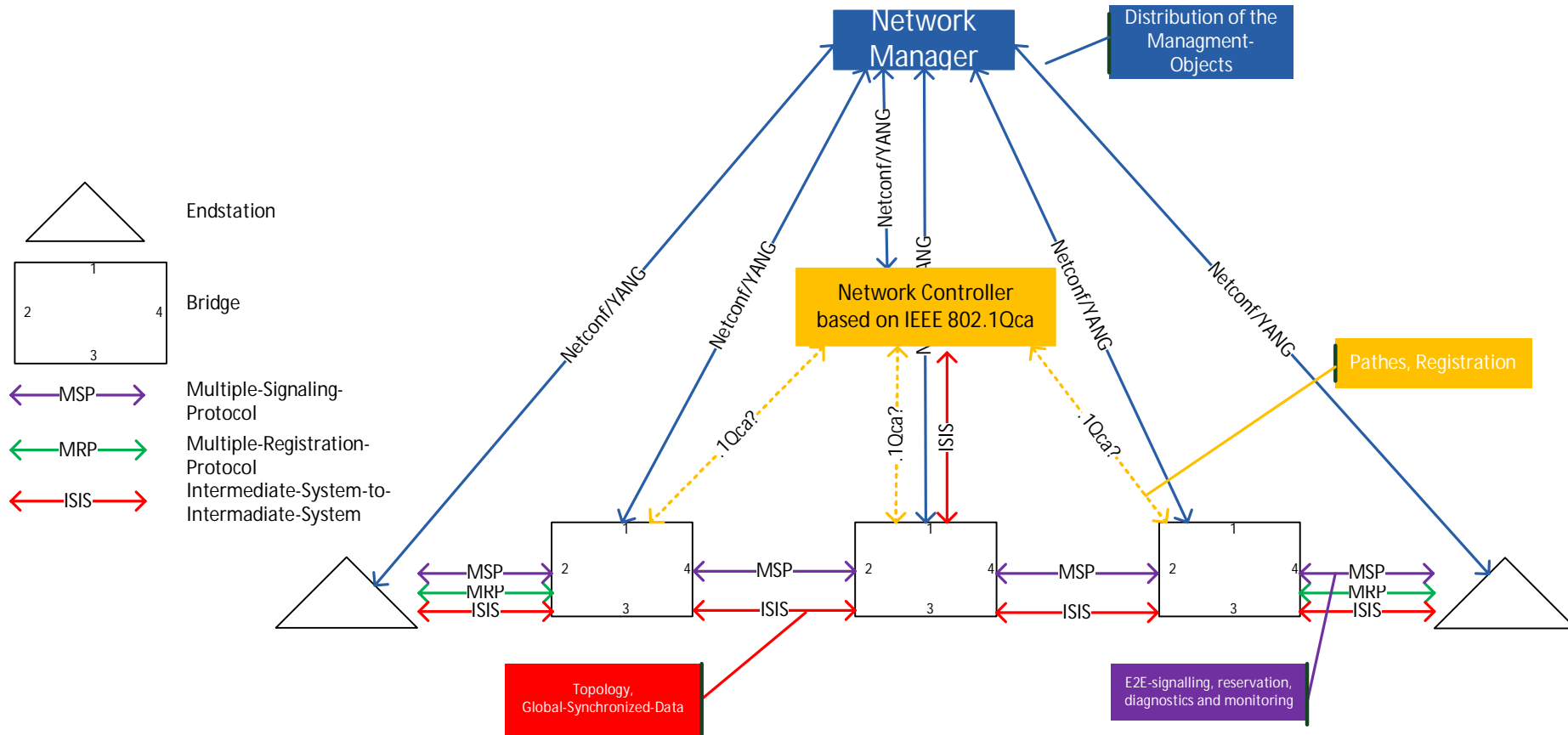
#### PAR 5: „service level“ YANG modules

- This is high level YANG (topology-wide)
- There is an IETF I-D on this topic:  
<https://datatracker.ietf.org/doc/draft-clemm-i2rs-yang-network-topo>  
If we assume this I-D will proceed to RFC, 802.1's role would be to specify an "augment" to this module for 802.1Q bridges. The augment should connect the top-level to the bottom-level.

# Decentralized Controlled Network with Path Computation, Registration & Reservation



# Centralized controlled Network with Path Computation, Registration & Reservation



# Motivation for MRP++ (MRPv2) and MSP

## Motivation for V2 MRP (Multiple Registration Protocol) and V1 MSP (Multiple Signaling Protocol)

	<u>MRP v1</u>	<u>MRP v2</u> "transport-protocol" for applications like MVRP, MMRP, MSRP, ...
<b>Pro (also Supported by new Version)</b>	<b>Cons</b>	<b>Features</b>
Distribution of network attributes over context	No fragmentation - limits the number of attributes. This problem is partly solved by spending one separate frame for each application or application instance. The disadvantage of the current solution that high computing power is required for serialization and deserialization.	<ul style="list-style-type: none"> <li>+ Support Fragmentation</li> <li>+ One MRP frame for all applications (including all attribute lists and states)</li> <li>+ Separate checksum for each attribute list</li> </ul>
One basic mechanism for different applications (MVRP, MMRP, ...) Common architecture (application-->instance-->attribute)	Very complex and intransparent state machines -> difficult to synchronize implementations from different vendors	<ul style="list-style-type: none"> <li>+ Simplified state machine and synchronization mechanism</li> </ul>
	MSRP combines registration and reservation, the attribute size (advertise) is very large and extended the MAP mechanism and introduced four packed events exclusively for MSRP	<ul style="list-style-type: none"> <li>+ MSRPv2 is only a registration protocol to register stream attributes (e.g. TSpec, TC, SR-DA, SR-ID, VID, ...)</li> </ul>
	The pack mechanism from MRP is not practical (only for special use cases)	<ul style="list-style-type: none"> <li>+ By introducing fragmentation the packed mechanism is no longer necessary</li> </ul>
		<ul style="list-style-type: none"> <li>+ Extending existing applications (MVRP, MMRP, MSRP) to support redundancy and seamless redundancy on precalculated trees</li> <li>+ If necessary add a new application like MRRP</li> </ul>
		<ul style="list-style-type: none"> <li>+ Optional support for higher layers like IP (e.g. transport higher layer addresses, QoS specifier, ..) by e.g. using TLV's</li> <li>+ Managed Objects</li> <li>+ TLV's are used to specify the MRP attributes</li> <li>+ The mechanism to synchronize the attribute list on a link is comparable to the synchronization mechanism used by ISIS (ISIS-like)</li> </ul>
		<p><b><u>MSP</u></b> ("RSVP like")</p> <p>("MSP is a separate transport-protocol" for e.g. stream reservation)</p>
	MSRP combines registration and reservation, the attribute size (advertise) is very large and extended the MAP mechanism and introduced four packed events exclusively for MSRP	<ul style="list-style-type: none"> <li>+ MSSP (Multiple Stream Signaling Protocol) is an application for MSP which is used for stream reservation, e2e signalling and diagnostic. The context, which is required for forwarding the signal / reservation, is either built by MRP or ISIS-PCR</li> <li>+ Managed Objects</li> </ul>

# Split the existing MSRP to MSRP / MRP++ and MSSP / MSP

New
Static Information
Dynamic Information

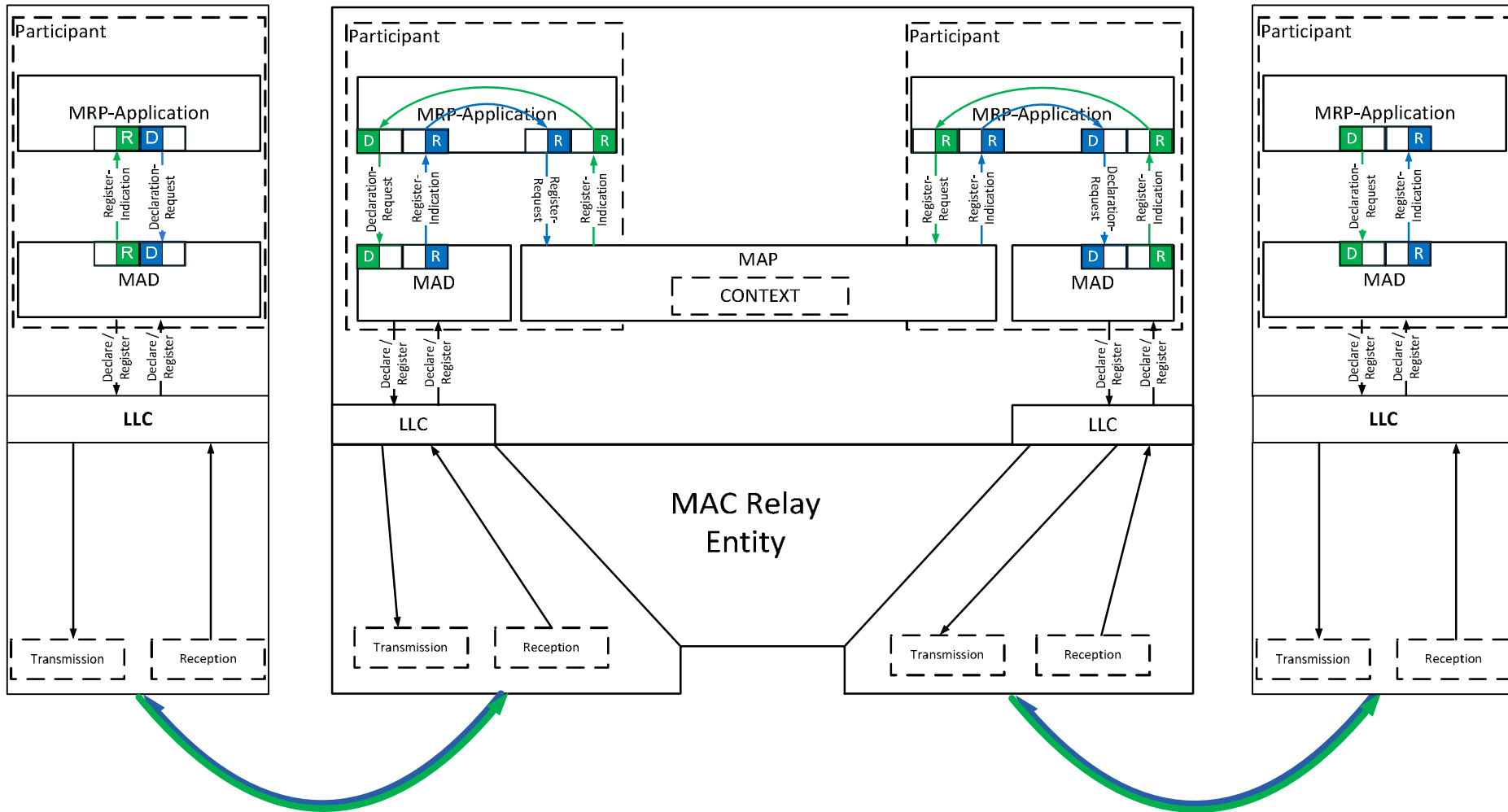
	<i>Talker Advertise</i>		<i>Talker Failed</i>		<i>Listener</i>		<i>Domain</i>
<b>MSRP on MRP</b>	<b>StreamID</b>	Talker Sys-ID	<b>StreamID</b>	Talker Sys-ID	<b>StreamID</b>	Talker Sys-ID	<b>StreamClassID</b>
		Unique-ID		Unique-ID		Unique-ID	<b>StreamClassPriority</b>
	<b>DataFrameParameters</b>	Dest-Address	<b>DataFrameParameters</b>	Dest-Address	<b>FourPackedEvent</b>	Ready /	<b>StreamClassVid</b>
		VID		VID		ReadyFailed /	
	<b>Tspec</b>	MaxFrameSize	<b>Tspec</b>	MaxFrameSize		AskingFailed /	
		MaxInterval		MaxInterval		Ignore	
	<b>PriorityAndRank</b>	DataFramePriority	<b>PriorityAndRank</b>	DataFramePriority			
Rank		Rank					
<b>AccumulatedLatency</b>	portTxMaxLatency	<b>AccumulatedLatency</b>	portTxMaxLatency				
		<b>FailureInformation</b>	BridgeID				
			FailureCode				

	<i>Talker Advertise</i>		<i>Listener</i>		<i>Domain</i>
<b>MSRPv2 on MRP++</b>	<b>StreamID</b>	Talker Sys-ID	<b>StreamID</b>	Talker Sys-ID	<b>StreamClassID</b>
		Unique-ID		Unique-ID	<b>StreamClassPriority</b>
	<b>DataFrameParameters</b>	Dest-Address	<b>Rspec</b>	MinRecvInterval	<b>StreamClassVid</b>
		VID	<b>Listener ID</b>	Listener Sys-ID	
<b>Tspec</b>	MaxFrameSize				
	MaxInterval				
<b>PriorityAndRank</b>	DataFramePriority				
	Rank				

<b>MSSP on MSP</b>	<b>StreamID</b>	Talker Sys-ID	<b>StreamID</b>	Talker Sys-ID	
		Unique-ID		Unique-ID	
	<b>AccumulatedLatency (Calculated downstream)</b>	portTxMinLatency	<b>RequiredLatency (Calculated upstream)</b>	portRxMinLatency	
		portTxMaxLatency		portRxMaxLatency	
	<b>State</b>	ok?	<b>AccumulatedRspec</b>	AccMinRecvInterval	
<b>List&lt;FailureInformation&gt;</b>	BridgeID	<b>State</b>	Ready / ReadyFailed / Failed		
	FailureCode	<b>List&lt;FailureInformation&gt;</b>	BridgeID		
			FailureCode		

# MRP++ Architecture



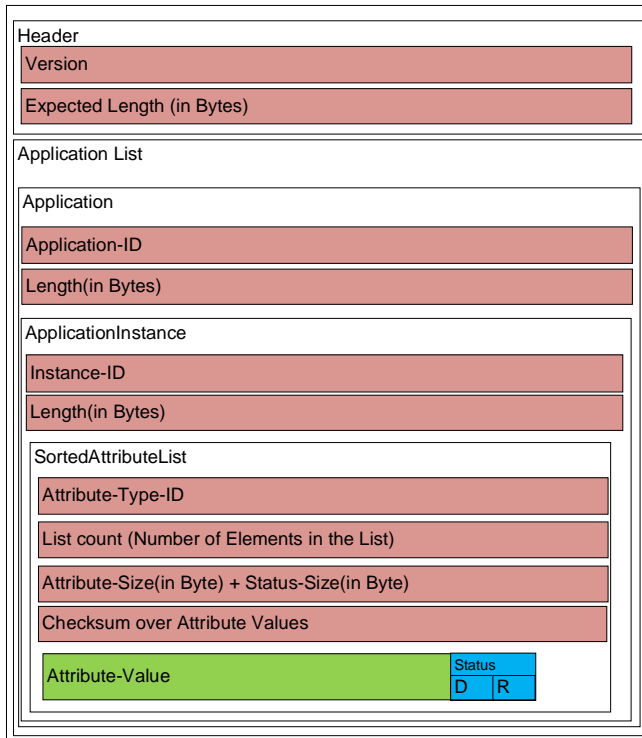


# MRP++ States

<u>State</u>	<u>R</u>	<u>D</u>	<u>Action</u>	<u>Description</u>
Empty	0	0	delete	No Attribute
Declared	0	1	---	Declare / announce a "New Attribute"
Registered	1	0		"New Attribute" registered
Established (Registered & Declared)	1	1	ready	"Attribute" registered and declared

# MRP++ Frame Format

## Frame:



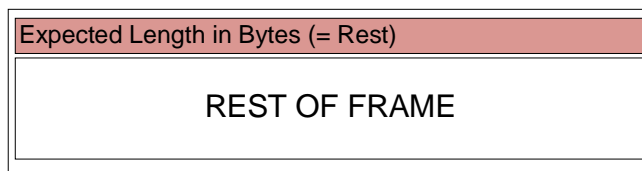
MRP-PDU	→ Header, ApplicationList
Header	→ Version, ExpectedLength
Version	→ UINT8
ExpectedLength	→ Length
Length	→ UINT16
ApplicationList	→ Application*
Application	→ <i>ApplicationId</i> , Length, ApplicationInstance*
ApplicationId	→ ID
ID	→ UINT8
ApplicationInstance	→ InstanceID, Length, SortedAttributeList*
InstanceID	→ UINT16
SortedAttributeList	→ ListHeader, ListBody
ListHeader	→ <i>AttTypeId</i> , ListCount, <i>AttributeSize</i> , Checksum
AttTypeId	→ ID
ListCount	→ <b>UINT8</b>
AttributeSize	→ <b>UINT8</b>
Checksum	→ <b>Fletcher-16</b>
ListBody	→ Attribute*
Attribute	→ Value, State
Value	→ <i>Attribute value defined by Application</i>
State	→ Declarator, Registrar
Declarator	→ BIT
Registrar	→ BIT

*Red: TBD(unsure)*

*Green: Defined By Application*

\* := 0 - N

## Fragment:





## Franz-Josef Goetz

PD TI ATS TM 4 2

Gleiwitzer Str. 555

90475 Nürnberg

Phone: +49 (911) 895-3455

E-Mail: [franz-josef.goetz@siemens.com](mailto:franz-josef.goetz@siemens.com)