802.1Qbv: Dynamic Configuration of Scheduling Windows

Rodney Cummings

National Instruments

Franz-Josef Goetz

Siemens AG

Agenda

- Need for dynamic configuration of 802.1Qbv
- Which existing protocol?
- Technical proposal

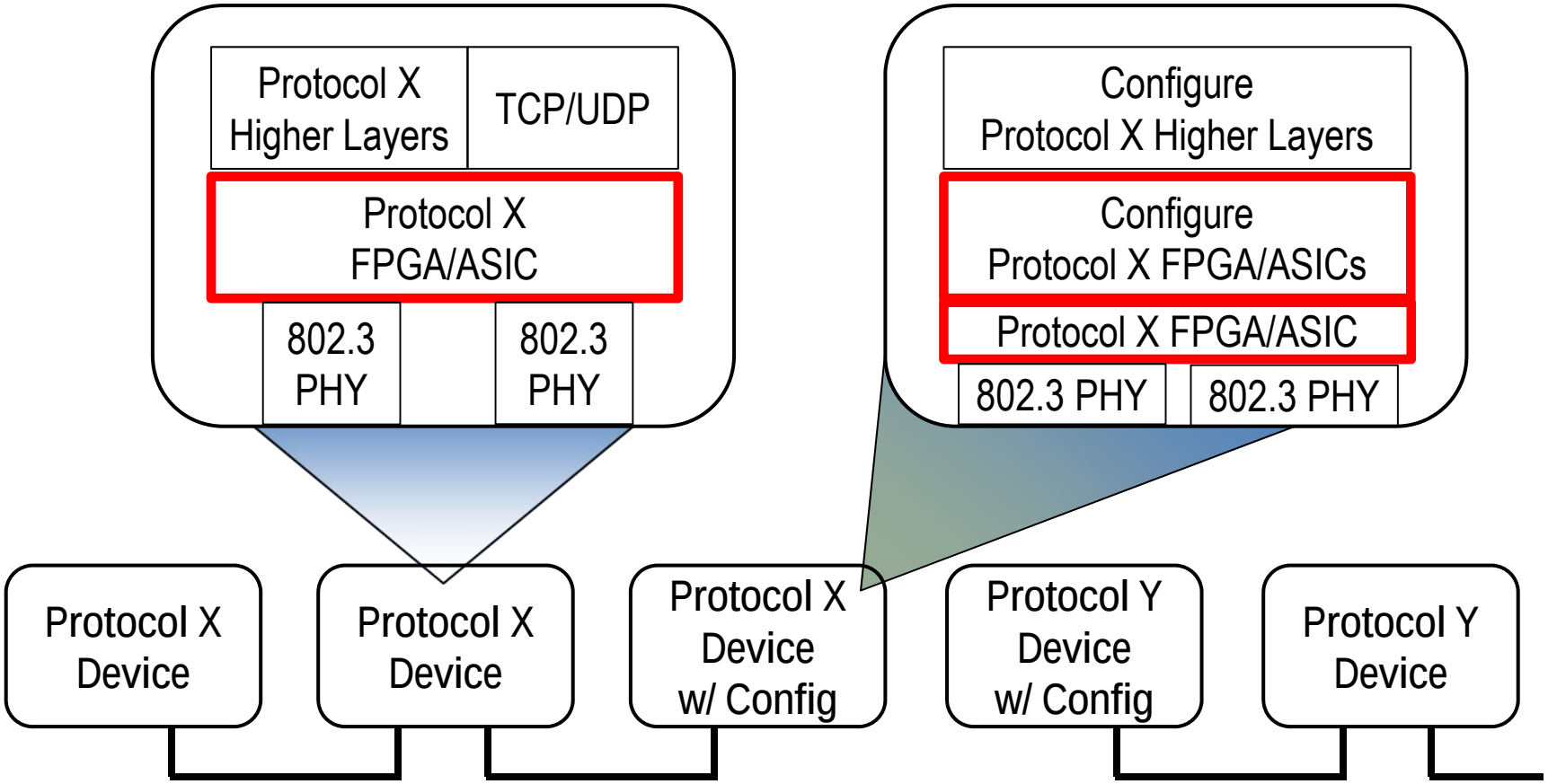
Profile of Industrial Ethernet User

- Not a networking expert
- Not dependent on IT
- No explicit configuration of bridge
  - Networking software may do this, hidden from user

Some Applications are Plug&Play

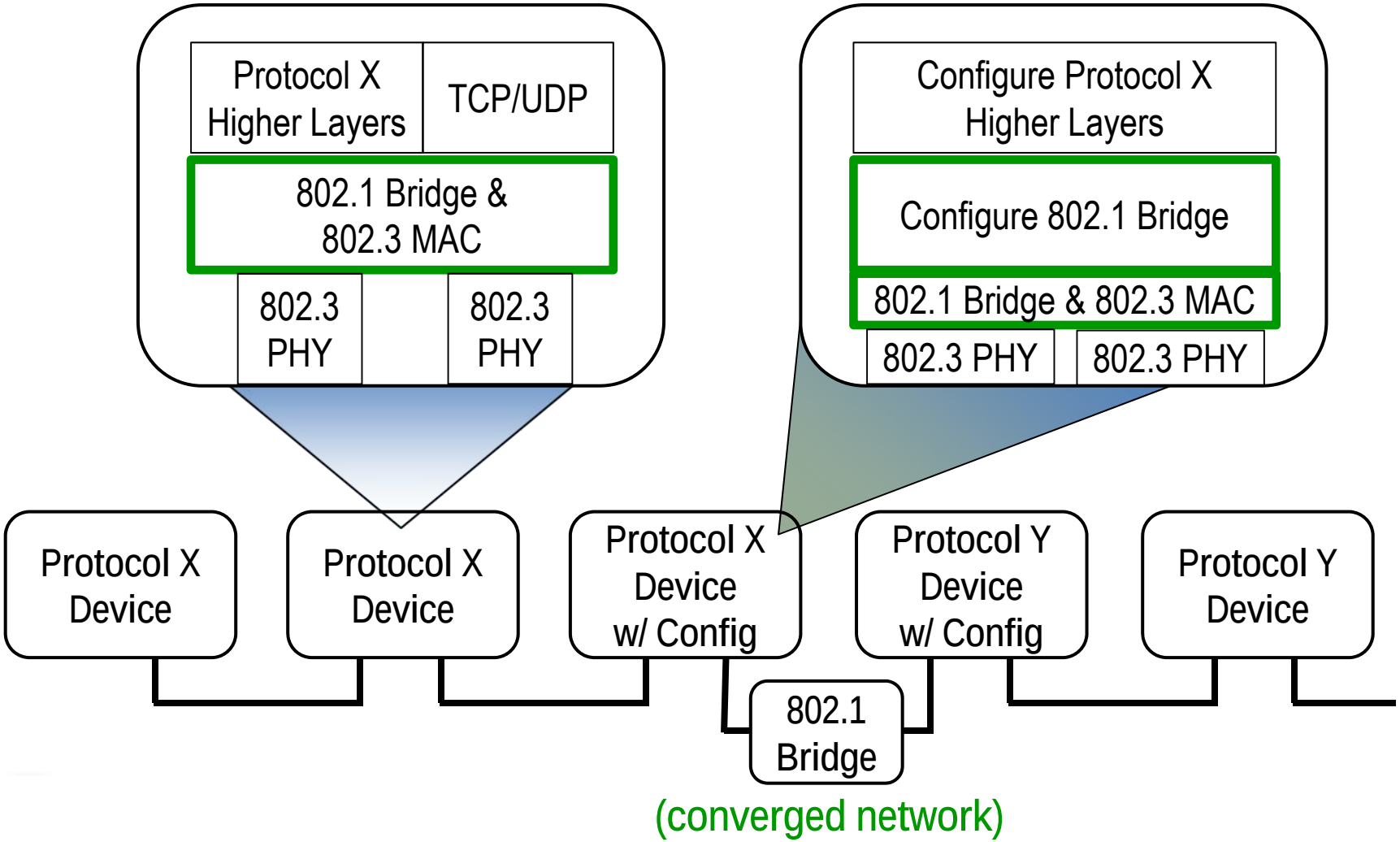
- E.g. Heavy vehicles
  - Plug “trailer” or “implement” into “tractor”

Industrial Ethernet Past



(distinct networks)

Industrial Ethernet Future



Delegate Bridge Config to Protocol X?

- Each Protocol X standard currently specifies its own config protocol, XML schema, etc
  - Typically not SNMP
- If 802.1Qbv window config is static only (MIB-only)…
  - ...no clear solution for configuration of 802.1 bridges
- Likely result: Each Protocol X standard entity will extend Protocol X to configure 802.1Qbv windows
  - Many different solutions
  - Back to the past… bridges specific to Protocol X

Recent Examples in 802.1

- 802.1Qat (MSRP, current AVB config)
  - Dynamic: Excellent fit to industrial Ethernet user
  - Sets precedent for AVB usability
- 802.1aq (SPB)
  - IS-IS dynamically configures routes
  - Supports dynamic or static assignment of frames to routes
    - Clause 27 intro: “To allow SPB to support plug-and-play operation for some VLANs, while providing the administrative controls and scalability required for large scale operations… (SPBV and SPBM)…”
  - Users can avoid complex bridge configuration
    - Subclause 27.1 item n): “Bridges do not have to be individually configured before being added to a network, other than having their MAC Addresses assigned through normal procedures.”

Routing and Scheduling

- Routing = Where
- Scheduling = When

- Closely related for industrial/embedded applications
  - Both required to calculate latency precisely
  - Intuitive to use same protocol for both

Proposed Roadmap for Scheduling

1. Procedures and managed objects
   - Work in scope of 802.1Qbv PAR
   - Static (offline) use cases fully supported
2. Extend existing protocol for *distribution*
   - Calculation of scheduling windows occurs on end station(s)
     - Above 802.1 (in Protocol X)
   - 802.1 protocol distributes windows to bridges
   - Work in scope of 802.1Qbv PAR

     OR

     Create new PAR for this work

Which Existing Protocol?

Importance of End Stations (ES)

- Calculation of scheduling windows in ES
  - Either static or dynamic
- Topology is a required input to this calculation
- For smaller/simpler applications, likely to have windows in ES more complex than windows in bridges
  - Tradeoff ease-of-configuration against ideal performance
  - E.g. Multiple windows in ES, One window in bridges

Assumptions

- Gating cycle: Repeating list of windows that gate queues on/off
- Gating cycle per bridge per port (egress)
- Gating cycle per end station port
- Two information exchanges
  - End station (ES) to nearest bridge
  - Bridge to bridge
- For existing protocol
  - No adverse effect on existing applications
  - Extension for scheduling windows works independently

Implications

- For future 802.1 configuration that includes end stations, which protocol(s) to use?
  - Past tended to use MRP as basis (e.g. MSRP)
  - Future desire to use IS-IS as basis
- This presentation explores question for 802.1Qbv, but answer is likely to apply to future ideas
  - 802.1Qbu (Preemption), MSRP upgrade, new shapers, …
- Dual goals for subsequent slides
  - Complete proposal for 802.1Qbv to show viability
  - Brainstorming on future direction for 802.1

Bridge Protocol: ISIS-SPB In Scope?

- IS-IS is a link state routing protocol
  - "IS-IS is easily extended to carry the required Ethernet address, VLAN, and Service membership information introduced by SPB" (802.1aq clause 28 intro)
- Distribute info without calculation?
  - 802.1aq 27.5: "ISIS-SPB can be viewed primarily as a means of sharing information between nodes in a network"
    - Each node has a copy of all node's info
    - Copies kept in sync
  - Recent email on 802.1 reflector regarding IS-IS:
    "THERE IS NOTHING THAT SAYS EVERY NODE HAS TO PERFORM BOTH INFORMATION DISTRIBUTION AND APPLICATION COMPUTATION"

ES Protocol: What Does SPB Use?

- Bridge to bridge is IS-IS
- End station (ES) to bridge is MRP
  - Translation at boundary of SPT Region
  - MMRP translates to SPBV MAC Address sub-TLV
    - 802.1aq 28.10
  - MVRP translates to SPBV SPVIDs
    - 802.1aq 27.12 and 27.13

ES Protocol: Options

- MRP?
  - Precedent in SPB for translating MRP at edge of SPT Region
- LLDP?
  - Primarily an ES-to-bridge discovery protocol
  - “not intended to act as a configuration protocol for remote systems” (802.1AB 6.2)
- ES-IS (ISO 9542, reprinted as RFC 995)?
  - Primarily an ES-to-bridge discovery protocol
  - Assumes ES does not participate in information distribution
    - Exchange of IS-IS TLVs not supported

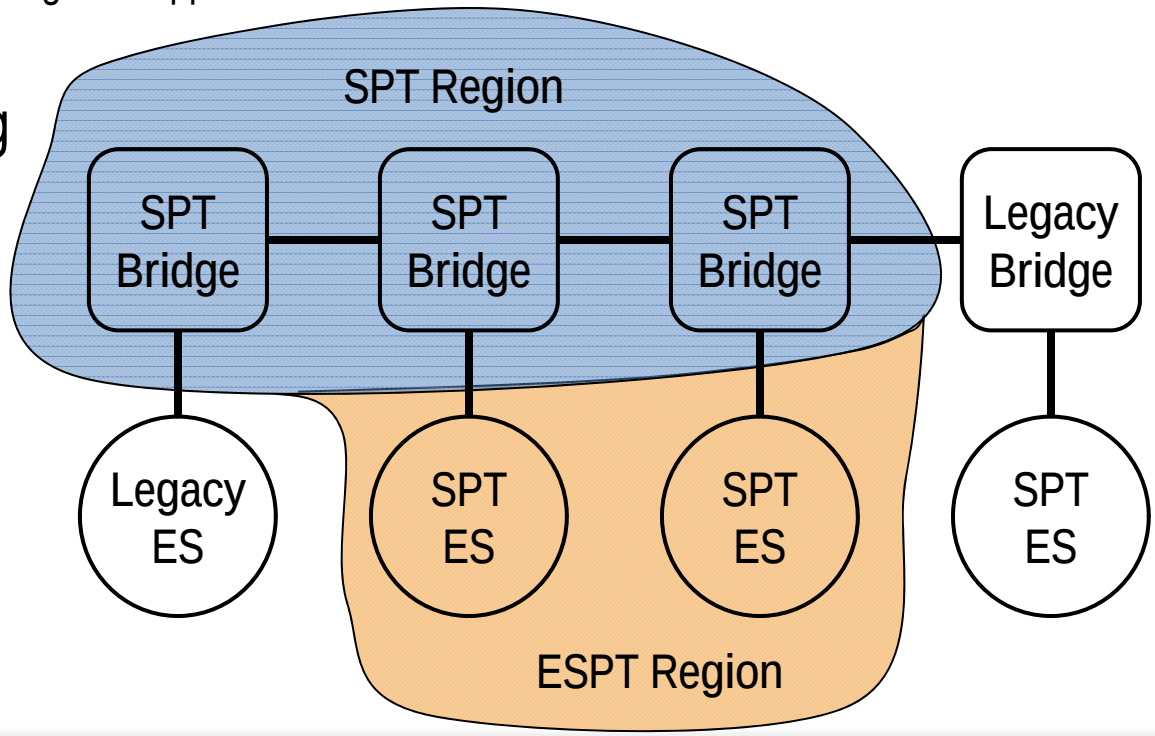
ES Protocol: ISIS-SPB (1 of 2)

- Idea: Extend ISIS-SPB to end stations
- ES exchanges ISIS-SPB PDUs
  - Looks like single port bridge (no transit)

- Subsequent slides use this option for ES to bridge
  - Preceding options could also be extended to support IS-IS sub-TLV exchange

ES Protocol: ISIS-SPB (2 of 2)

- Extended SPT Region (ESPT): SPT Region plus ES that are…
  - directly adjacent
  - using ISIS-SPB (SPT capable)
  - still outside SPT Region (e.g. VID translation boundary)
    - avoid breaking existing SPB applications
- SPT = Routing
- ESPT = Non-routing



Technical Proposal: Extend ISIS-SPB

Overview of ISIS-SPB Extensions

- Formation of ESPT Region
- Sub-TLV to configure windows in remote port
- Sub-TLVs to synchronize gating cycles in ESPT Region

- Focus on configuration not data transfer
  - E.g. Not discussing SPBV versus SPBM

Formation of ESPT Region (1 of 2)

- Formation of SPT Region: 802.1aq 28.2
  - Formation of ESPT Region re-uses methodology
- ES sends IS-IS Hello PDU with NLPID 0xC1 (ISIS-SPB protocol ID)
- ES's IS-IS Hello PDU uses MCID values that avoid match with SPT bridges, to remain outside SPT Region
  - Configuration Name "IEEE 802.1 ES xyz", where xyz is ES's MAC address as a hex string
  - All other MCID elements zero
  - Similar to legacy (MST) bridge MCID per 802.1Q 13.8

Formation of ESPT Region (2 of 2)

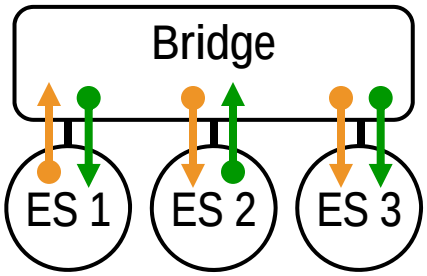
- ESPT-capable bridge evaluates adjacent ES
  - Distinguishes between ES that is ESPT-capable or not
    - Using NLPID and Config Name
  - If ES is ESPT-capable, it is included in ESPT Region
    - Bridge exchanges IS-IS PDU with ES (e.g. scheduling sub-TLVs)
    - Bridge forwards all IS-IS PDU to ES, so ES has all info (e.g. topology)
    - ES will not send sub-TLV that apply only within SPT Region (e.g. routing)
- ESPT-incapable SPT Bridge ignores ES
  - Existing 802.1aq bridge unaffected by new scheduling sub-TLVs
- ESPT Region used to distribute scheduling windows

Configuring Windows from ES (1 of 2)

- Typical IS-IS methodology:
  - Distribute my info
  - Gather info from other nodes
  - Sync all info to perform calculation
- ISIS-SPB extended with membership
  - E.g. MAC address membership
    - ES outside SPT Region sends MMRP PDU to SPT Bridge (‘my info’)
    - Bridges in SPT Region distribute in ISIS-SPB sub-TLV; no calculation
- Proposal: Configure windows similar to membership
  - ES in ESPT Region sends ISIS-SPB sub-TLVs (‘my info’)
  - Bridges distribute windows for ports in ESPT Region

Configuring Windows from ES (2 of 2)

- Windows from ES can be entire gating cycle, or subset
  - E.g. ES 1 sends windows for itself and ES 2 (six egress 'dots')
  - E.g. ES 1 sends its windows (orange dots), and ES 2 sends its windows (green dots)



- Bridge merges windows from different ES to form cycle
  - E.g. Egress from bridge to ES 3 merges two sets of windows to form a single gating cycle
  - Protocols above 802.1 must ensure windows do not overlap
  - Nevertheless, 802.1 specifies a tie-breaker algorithm
    - E.g. Lowest ES MAC address wins

SPB Schedule Window sub-TLV (1 of 2)

- Configure windows for specific egress port
- IS-IS Node Info Extension (per RFC 6329)
  - Configures port (IS-IS adjacency), but generated from node (software on ES)
- Carried within MT-Capability TLV
  - Overload Flag = 1
    - SPT ES set true to specify IS-IS “no transit” semantics
- Multiple sub-TLV used to configure entire ESPT Region
  - ES software sends list of Schedule Window sub-TLV, one for each port in region

SPB Schedule Window sub-TLV (2 of 2)

Name	Octet	Len	Notes
Type (5)	1	1	Next MT-Cap value from IANA
Length (3n+18)	2	1	
System ID	3-8	6	Address of bridge to configure
Port Number	9-10	2	Port number within bridge
Format	11	1	Format of subsequent bytes (reserved)
Interval Resolution	12	2 bits	0=1µs, 1=10µs, 2=100µs, 3=1000µs
Start Time	13-20	8	
Window Tuple 1, Gates	21	1	8 flags, one per queue, 0=close, 1=open
Win Tuple 1, Interval	22-23	2	
Win Tuple n, Gates	3n+18	1	
Win Tuple n, Interval	(3n+19)-(3n+20)	2	

SPB Schedule Cycle sub-TLV (1 of 2)

- Bridge reports current gating cycle for each of its ports
  - Merged from SPB Schedule Window sub-TLVs for port
- Typical IS-IS: Report my info for others to use
- IS-IS Adjacency Info Extension (per RFC 6329)
- Carried within Extended IS Reachability TLV or MT IS TLV

SPB Schedule Cycle sub-TLV (2 of 2)

Name	Octet	Len	Notes
Type (31)	1	1	Next adjacency value from IANA
Length (3n+11)	2	1	
Port Number	3-4	2	Bridge address is in IS-IS header
Use Flag	5	1 bit	Is gating cycle in use (running)?
Static Flag	5	1 bit	1=static, 0=dynamic, static (mgt) always wins over dynamic (SPB Sched Config)
Interval Resolution	5	2 bits	0=1µs, 1=10µs, 2=100µs, 3=1000µs
Start Time	6-13	8	
Window Tuple 1, Gates	14	1	8 flags, one per queue, 0=close, 1=open
Win Tuple 1, Interval	15-16	2	
Win Tuple n, Gates	3n+11	1	
Win Tuple n, Interval	(3n+12)-(3n+13)	2	

SPB Schedule Digest sub-TLV (1 of 2)

- Sync all gating cycles in ESPT Region (including ES)
- IS-IS Hello Extension (per RFC 6329)
- Carried within MT-Port-Cap TLV
- Leverage SPB Agreement Digest concept
  - Digest Port Count (instead of edge count)
  - Calculate signature (compression) of each port's cycle
  - Sum port signatures to create 32-byte digest
- Use Flag: Logical AND of Use Flag of all ports in region
  - ES use to determine when ESPT Region is up and running

SPB Schedule Digest sub-TLV (2 of 2)

Name	Octet	Len	Notes
Type (7)	1	1	Next MT-Port-Cap value from IANA
Length (33)	2	1	
V	3	1 bit	Same semantics as SPB Digest
A	3	2 bits	Same semantics as SPB Digest
D	3	2 bits	Same semantics as SPB Digest
Use Flag	3	1 bit	Logical AND of all ports in region
Schedule Digest	4	32	Similar to 802.1aq 28.4

Memory Considerations

- Protocol design requirement from 802.1aq 27.1.l):
  "The memory requirements associated with each Bridge Port are either a constant or a linear function of the number of Bridges and LANs in the network."
- Analogy: SPB Link Metric sub-TLV
  - One per port in region, but constant
- Proposed SPB Schedule Window & Cycle sub-TLVs
  - Linear function of the number of windows (~ 3n+20)
  - Example: 128 windows, 100 ports → 40 kBytes
- If memory a concern, each bridge can store its ports only
  - Store other ports as digest signature only (constant)
  - Disadvantage: smarter bridges in future (calculation)

Thank you