# LLDP+ and VDCP:
# VSI Discovery and Configuration Protocol – a Proposal

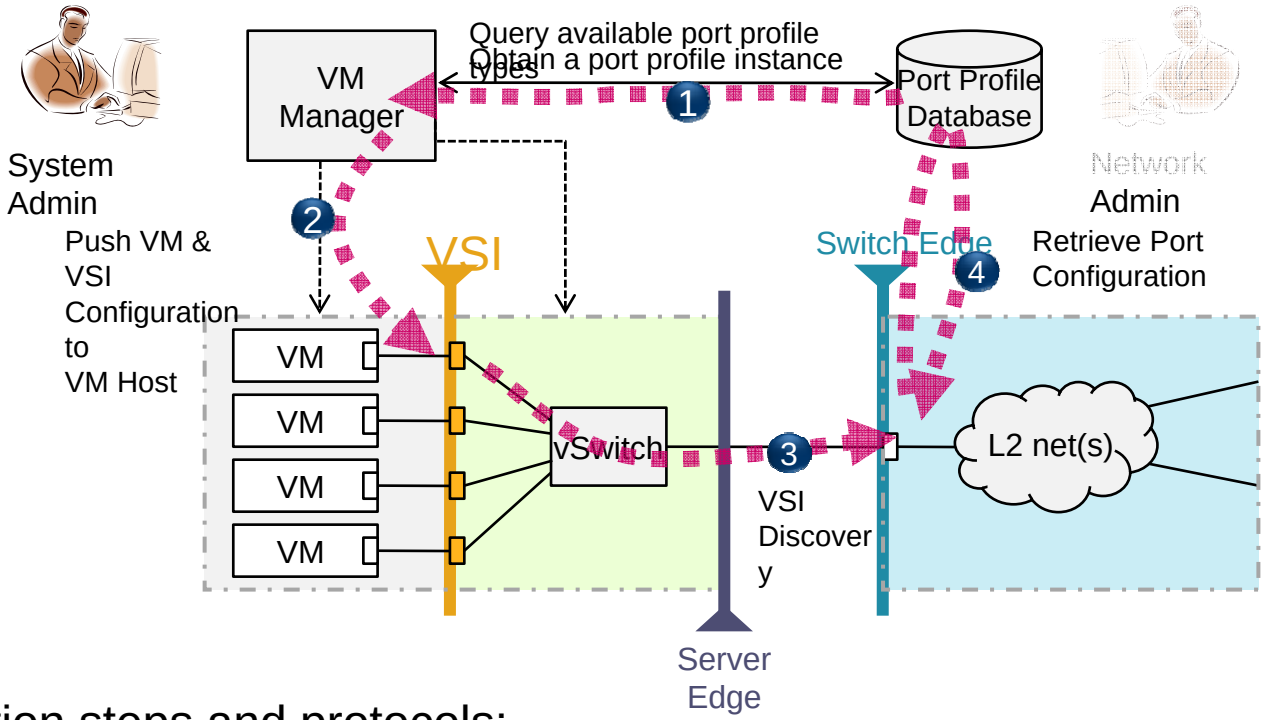
v46
January 21, 2010

**Caitlin Bestler, Aprius**
Uri Elzur, Broadcom
Ilango Ganga, Intel
Manoj Wadekar, Qlogic

# VDCP / LLDP+ Goals

- The NEED: address the emerging needs of VSI Discovery in the Data Center with ample headroom and flexibility
- The proposed Approach: Re-use existing protocols and architecture to minimize development time and effort at the IEEE and in products
- A solution: Upgrade existing protocols (e.g. LLDP) to specifically address VSI Discovery needs
- Provide
  - Scalability
  - Efficiency
  - Timeliness

# VSI Discovery in the context of EVB




Configuration steps and protocols:

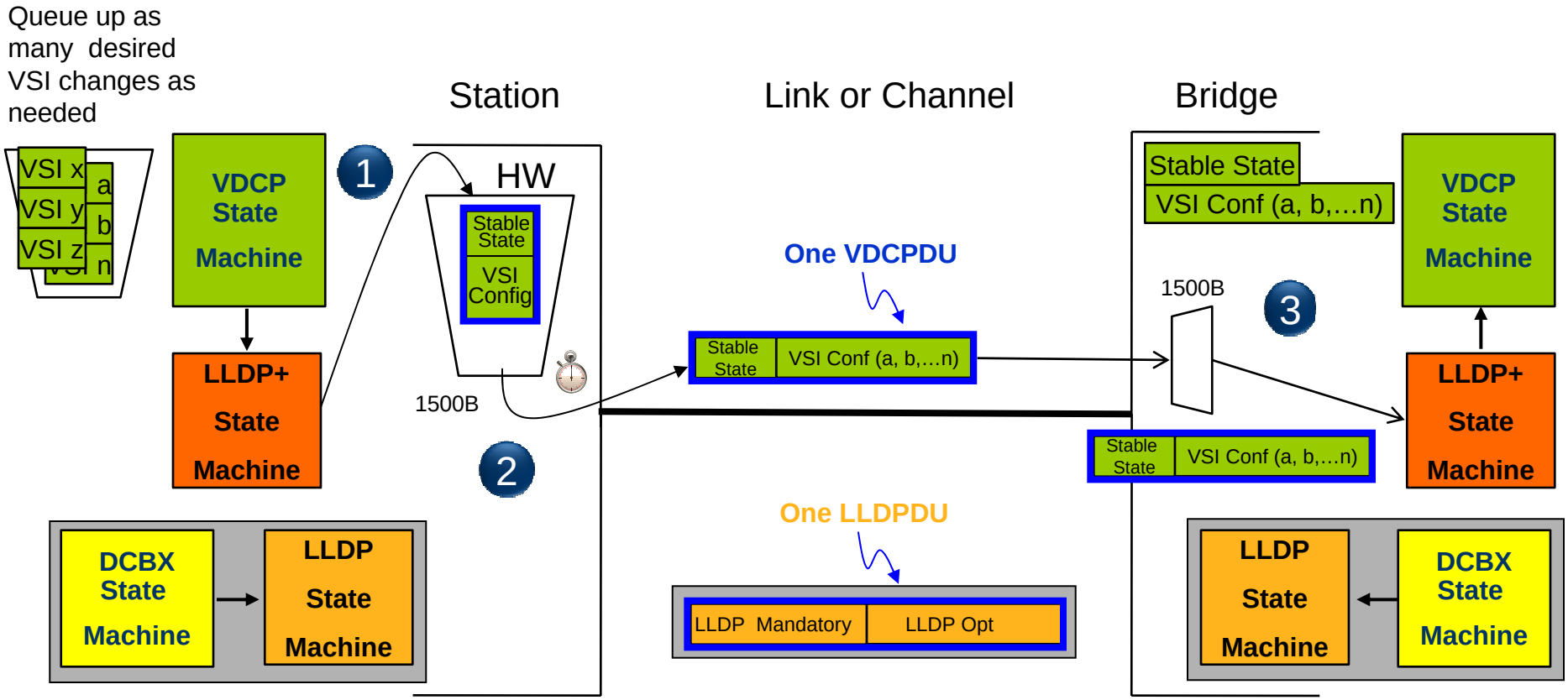
- DCBX (Optional) over LLDP
- EVB Discovery is an ordered set of protocols
  - Multi Channel (Optional) - over LLDP
  - Discovery of Host Switching (VEPA, VEB, Direct Access, PE) – over LLDP
  - VSI Discovery and Configuration ⇨ FOCUS of this presentation

# Consensus and New requirements

- Consensus?
  - To have ALL EXPLICIT VSI state in one message, may require more than 1500B
  - No more than one 1500B message, in each direction, outstanding between link partners at any time
    - Stay away from creating a need for ACK or Sequencing and other Transport attributes
    - Allow for Scalability from small to very large
    - Allow both parties resources to dictate frame exchange rate
  - Stay as close to IEEE 802.1AB-REV LLDP as possible – but extend where needed
    - Must run IEEE 802.1AB-REV LLDP anyhow for DCBx
    - As much re-use of code/HW of existing LLDP implementations for VDCP
  - There are limits to number of VSI state changes/Second that can realistically be supported by HV/Server/Switch/Network/Storage/Management subsystems
  - LLDP can be used for VSI Discovery & Configuration, BUT some bytes may be consumed by Mandatory and Optional TLVs => not an issue if LLDP+ uses a different EtherType
- New Requirements
  - Discovery and Configuration exchange protocol
  - Allow VSI State exchanges with minimal overhead (eliminate unneeded TLVs)
  - Ability to communicate a Digest of all VSI state in a TLV that is < (or even <<) 1500B
  - Allow for faster reaction time than LLDP, but allow for partner controlled rate
- Solution space: Require a completely new protocol or use IEEE 802.1AB-REV as a basis with minor adjustments?

# VDCP / LLDP+
## (single direction shown)

Notes:
Reliability: fast re-xmt

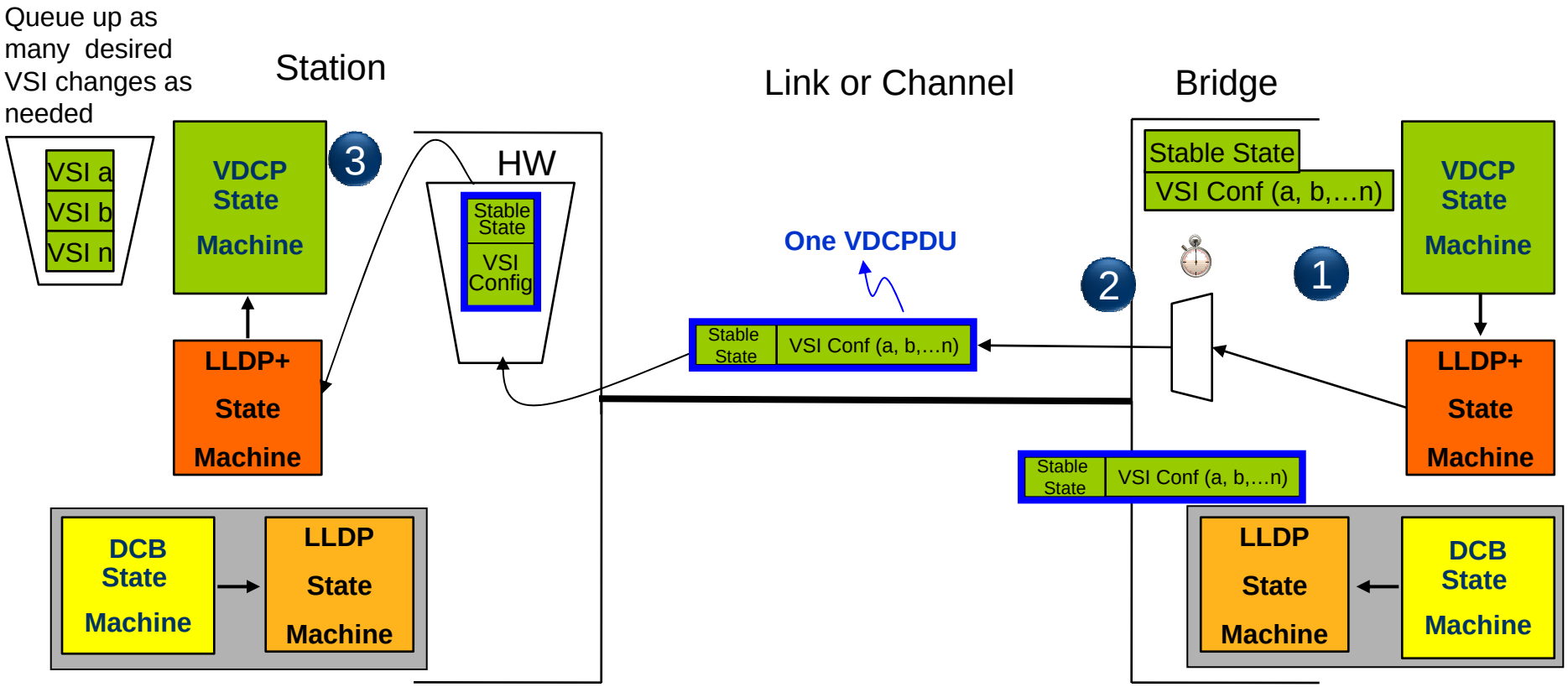



1. VDCP creates a VDCPDU (contains VSI Config TLV with all VSI State Change Requests <MTU, along with a Stable State TLV). VDCP updates the Local MIB and informs LLDP+ of "Local MIB Change" event.

2. LLDP+ transmits the new PDU due to LOCAL MIB Change event.

3. LLDP+ receives the PDU and informs VDCP of a "Remote MIB Change". VDCP consumes VDCPDU, MAY update its local VSI and configure accordingly. VDCP may update the Local MIB and informs LLDP+ of any "Local MIB Change" event.

# VDCP - Bridge response
## (single direction shown)




1. Bridge's VDCP state machine creates a VDCPDU with the VSI Config Request (VSI State MAY be updated!) and the Stable State TLV (Bridge may unilaterally DEASSOCIATE VSIs).

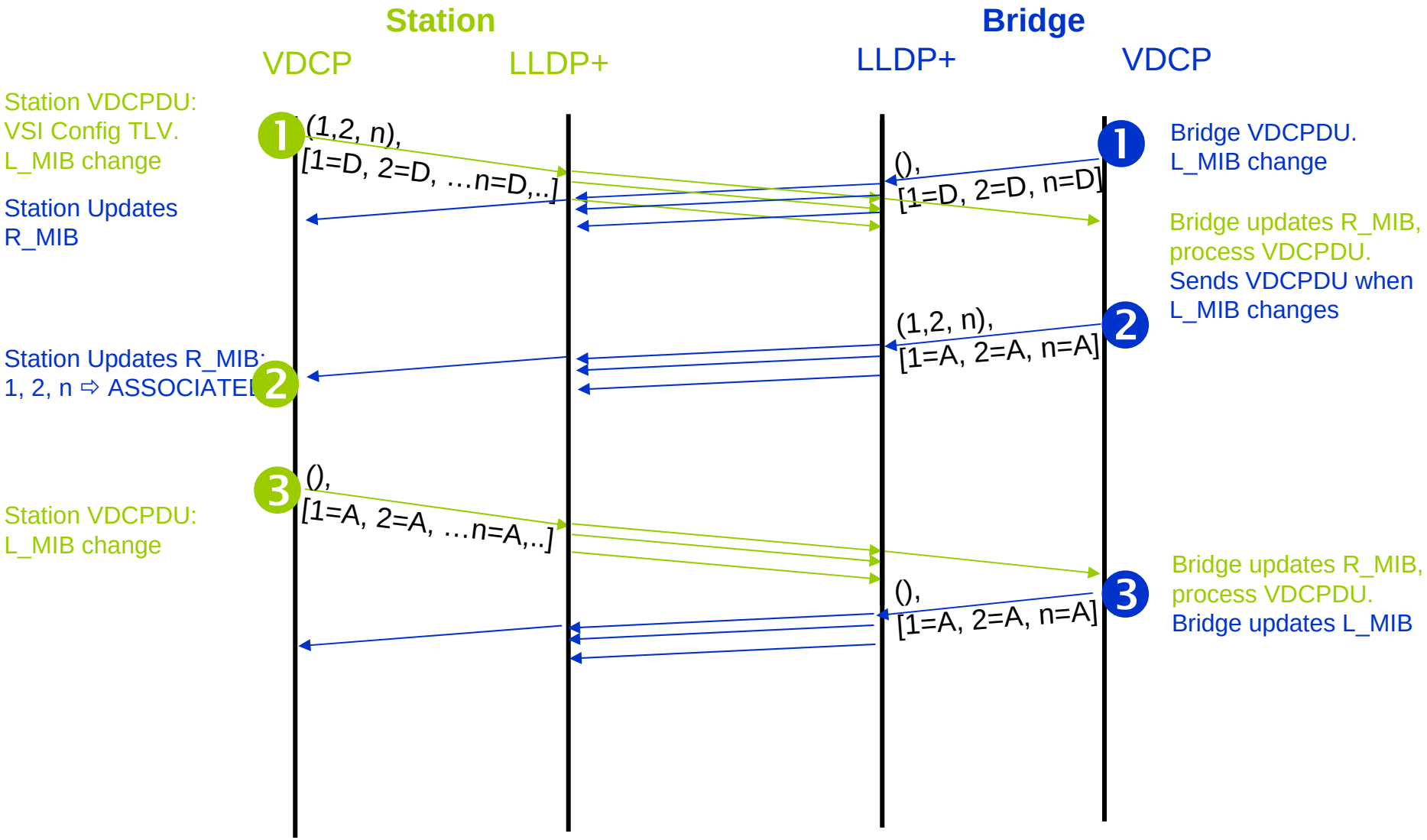
2. VDCP updates the Local MIB and informs LLDP+ of "Local MIB Change" event. VDCPDU is transmitted immediately.

3. LLDP+ receives the PDU and informs VDCP of a "Remote MIB Change". If Bridge provided new info in the VDCPDU, Station MUST reflect this new state back to the Bridge in the first next VDCPDU it transmits.



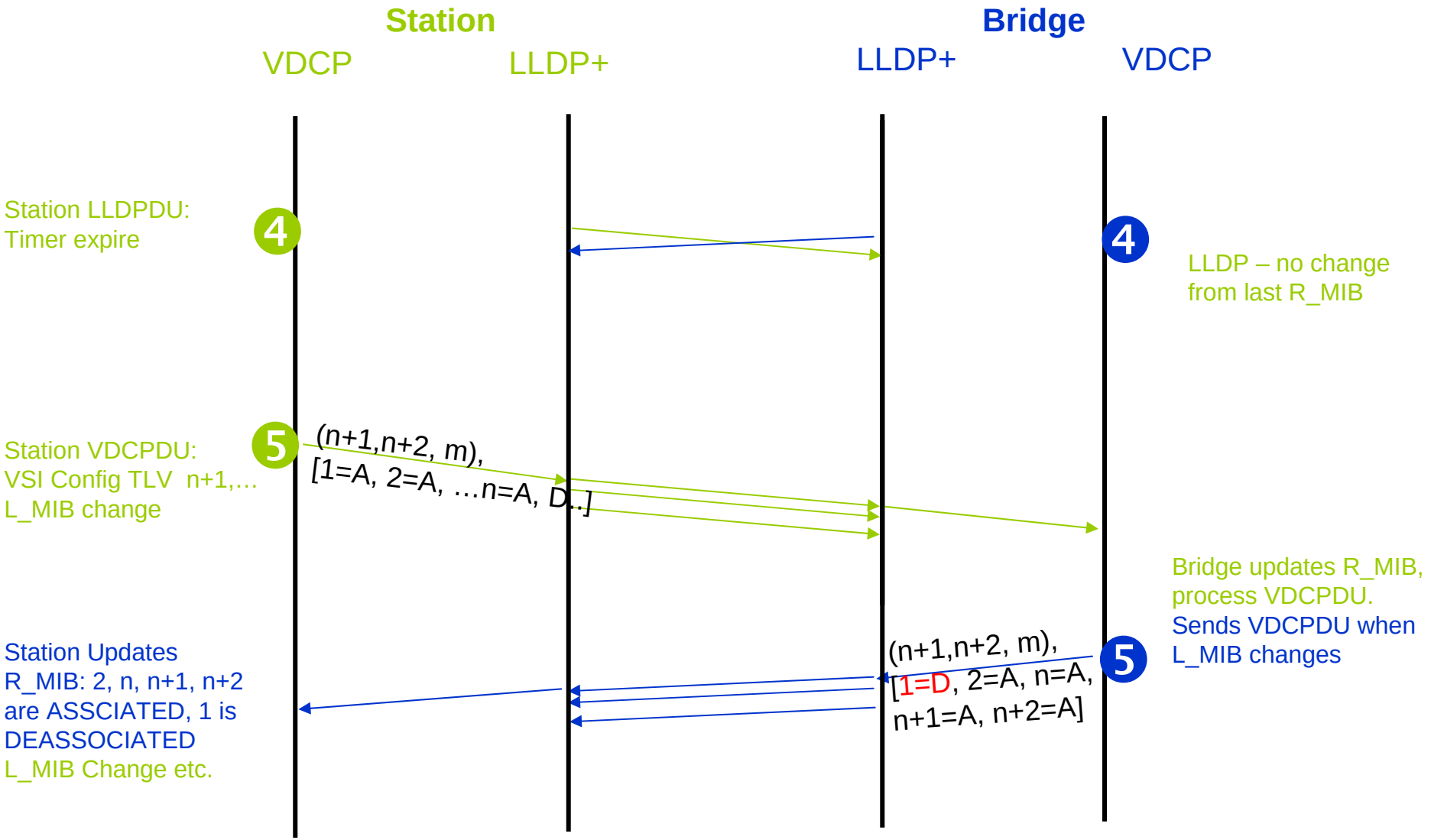
• Station replaces VSI's in VSI Config TLV after VSI reach Stable state (A or D) or may add VSI's if outstanding VSI < SC-max

# VDCP/LLDP+ Exchanges Example




Legend: VSI Config TLV (x,x,…x), Stable State TLV [x,x,…x]

# VDCP/LLDP+ Exchanges Example




Legend: VSI Config TLV (x,x,…x), Stable State TLV [x,x,…x]

# VDCPDU Format (A starting point...)

Frame format for per channel exchanges [B]


Channel Number (opt.)

EVB Status –
- CONFIG REQUEST
- CONFIGURED
- CONFIG REQUEST NACK
- RESET REQUEST
- RESET

VDCP State*** –
- ASSOCIATE REQUEST
- ASSOCIATED
- ASSOCIATE REQUEST NACK
- DEASSOCIATE REQUEST
- DEASSOCIATED

** In some cases SC-max can be equal to V-max. SC-max MUST be set to ensure VDCPDU < MTU*

*** for 1024 VSI*

**** Can be extended to incorporate additional states*

| | Bytes |
|---|---|
| VSI Configuration TLV*** – | |
| • VSI index *(0 … V-n)* | 2 |
| • DB index | 2 |
| • PPID | 1 |
| • MAC address | 4 |
| • VLAN list | 6 |
| • VSI State | 4 |
| • VSI index 1,…. | 1 |
| • : | 18 |
| • VSI index X, …. | |
| VSI Stable State TLV (0, *V-n*) – A/!$D_0$, A/!$D_1$, …, A/!$D_{V\,n}$ | 18 |
| | < 128** |

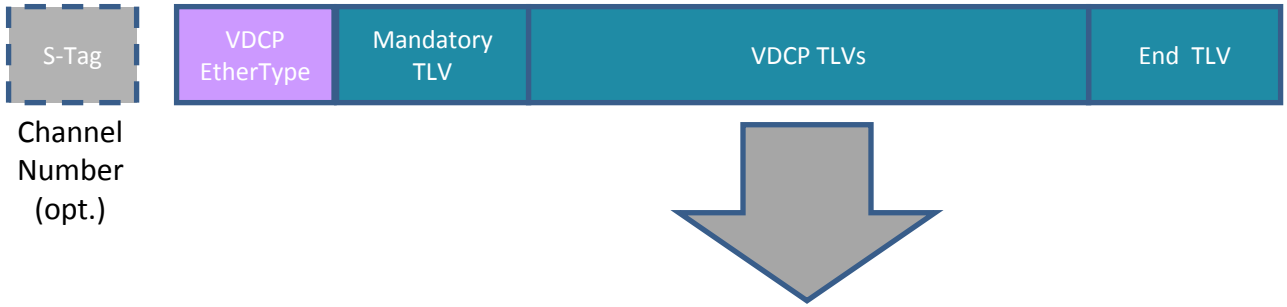
18

As TLV work is going on, only those needed for LLDP+/VDCP are mentioned here

- A TLV is limited to 512B. Allow for multiple VSI Configuration TLVs in one VDCPDU to save overhead or: keep the existing SubType headers (2B per VSI)
- Station initiates VSI Configuration TLV and Bridge responds. Both can initiate Stable State TLV

# VDCPDU Capabilities Format (A starting point…)

Frame format for per channel exchanges [B]




Sent once for configuration information exchange. May be combined with VEB configuration

| VDCP Capabilities TLV – | Bytes |
|---|---|
| •EVB Type (VEPA \| VEB \| VNIC \| … ) | 2 |
| •EVB Type Status | 1 |
| •Max number of VSI supported *(V-max)* | 1 |
| •Current number of VSI *(V-x)* | 2 |
| •Max number of State changing VSI supported *(SC-max)** | 2 |
| •# of VSI State Change request in the VSI Configuration TLV *(SC-m)* | 2 |
| •*# of Data Bases supported (DB-n)* | 2 |
| •*Data base identifier/s (n)* | 1 |
| | n*16 |

OPTIONAL: configurable number of bits per VSI in the Stable State

# Next Page – a more scalable VDCPDU

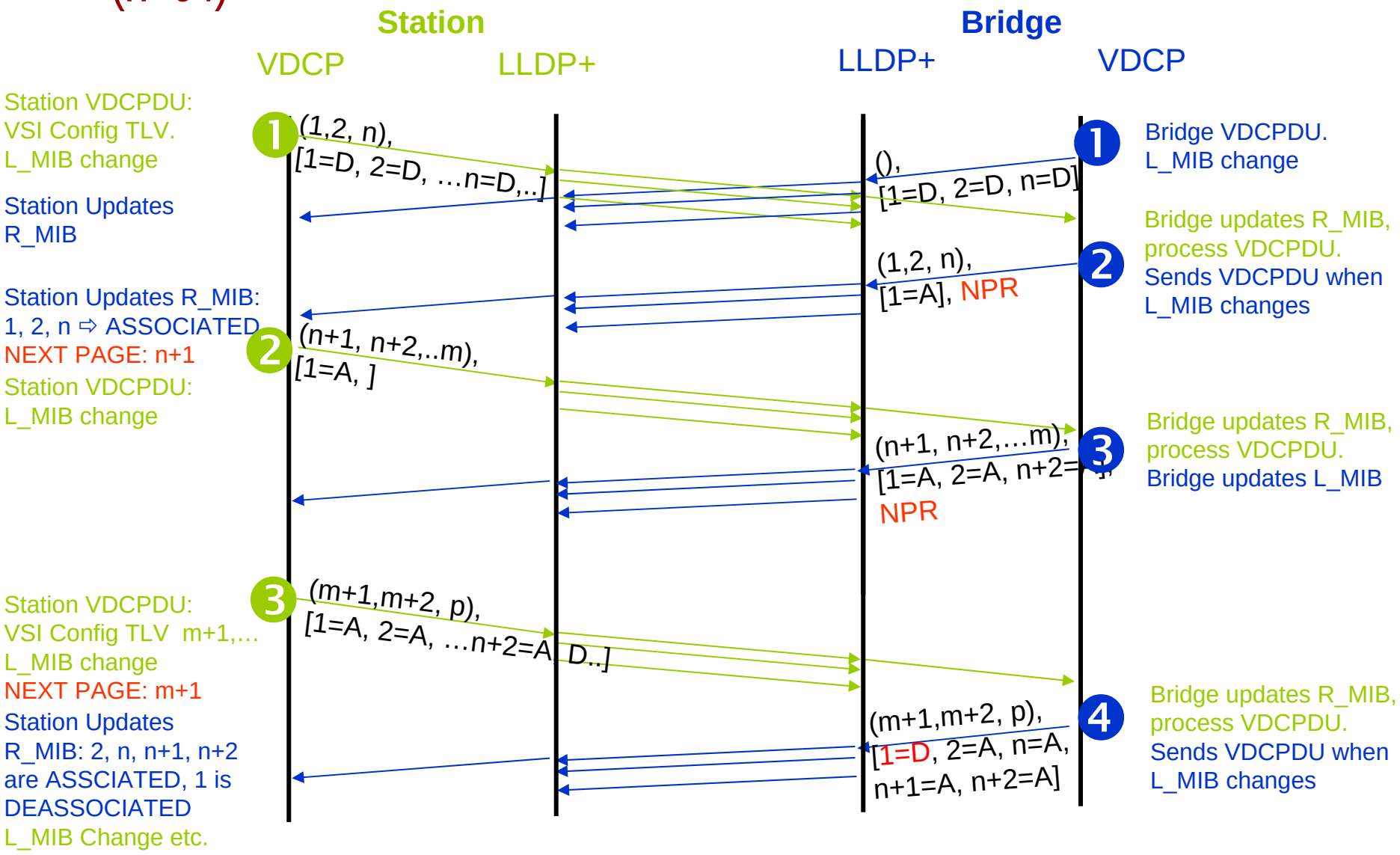
- Motivation: Bridge may experience varying latencies for processing some VSI state change request (e.g. Port Profile DB is distributed).
- Motivation: a Station with a very large number of VSI, may wish to get a high end Bridge to configure a larger number faster

- 2 options to support Next Page are presented below

# A. Next Page – optional mechanism

- MECHANISM: Allow for a larger set of VSI to be in-processing as compared with limit of number of VSI state that fits into a VSI Config TLV
  - For 1500B MTU, basic VDCP allow for ~80 VSIs to simultaneously change state
- Station may be allowed to send new VSI in the VSI Config TLV
  - AFTER: Bridge has sent back to Station a VDCPDU containing earlier sent VSI-s and an indication it has the resources to get more requests
  - AS LONG AS: total number of outstanding VSIs is smaller than SC-max

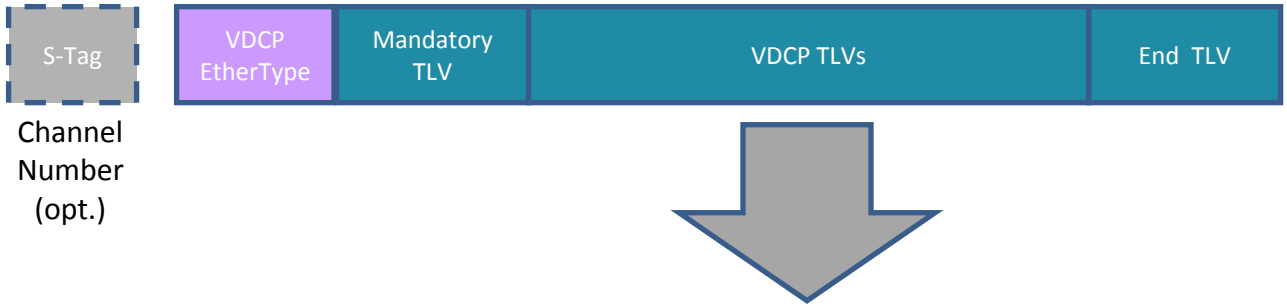
# VDCP/LLDP+ Next Page Exchanges Example (n=64)




Legend: VSI Config TLV (x,x,…x), Stable State TLV [x,x,…x]

# VDCPDU Format (with Next Page)

Frame format for per channel exchanges [B]




EVB Status –

- CONFIG REQUEST
- CONFIGURED
- CONFIG REQUEST NACK
- RESET REQUEST
- RESET

VDCP State*** –

- ASSOCIATE REQUEST
- ASSOCIATED
- ASSOCIATE REQUEST NACK
- DEASSOCIATE REQUEST
- DEASSOCIATED

** In some cases SC-max can be equal to V-max. SC-max MUST be set to ensure VDCPDU < MTU*

*** for 1024 vPorts*

**** Can be extended to incorporate additional states*

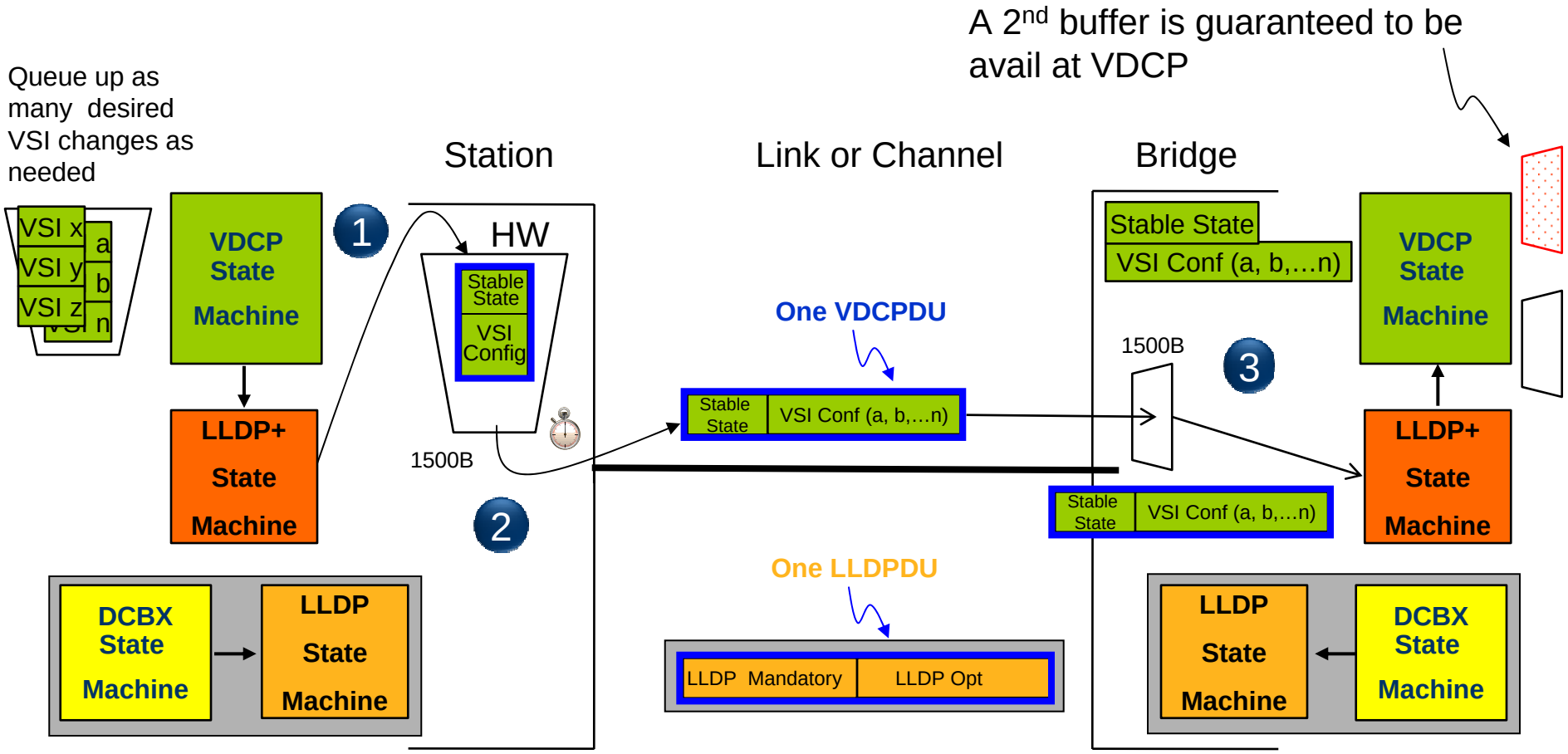
| | Bytes |
|---|---|
| VDCP Capabilities TLV – | 2 |
| •EVB Type (VEPA \| VEB \| VNIC \| … ) | 1 |
| •EVB Type Status | 1 |
| •Mode (Next Page Mode) | 1 |
| •Max number of VSI supported *(V-max)* | 2 |
| •Current number of VSI *(V-n)* | 2 |
| •Max number of State changing VSI supported *(SC-max)** | 2 |
| •# of VSI State Change request in the VSI Configuration TLV *(SC-n)* | 2 |
| VSI Configuration TLV*** – | 2 |
| • VSI index *(0 … V-n)* | 2 |
| • PPID | 4 |
| • MAC address | 6 |
| • VLAN list | 4 |
| • VSI State | 1 |
| • VSI index 1,…. | 17 |
| • : | |
| • VSI index X, …. | 17 |
| **Next Page Ready (NPR)** | 1 |
| VSI Stable State TLV (0, *V-n*) – $A/!D_0$, $A/!D_1$, …, $A/!D_{V_n}$ | < 128** |

17

# B. Next Page – 2 pages w/ implicit Flow Control

- MECHANISM: Bridge guaranteed storage capacity is doubled to 2 * MTU @VDCP
- No change to LLDP+ operation
- Station may send the 2nd page and
  - It triggers the same R_MIB change event
  - it is guaranteed to be stored @VDCP
- Station will only move to a 3rd page after the Bridge's VDCP has indicated a Local MIB change to its LLDP+, in turn triggering sending the 2nd Page back to the Station

# VDCP / LLDP+ with Next Page (implicit Flow Control)
(single direction shown)

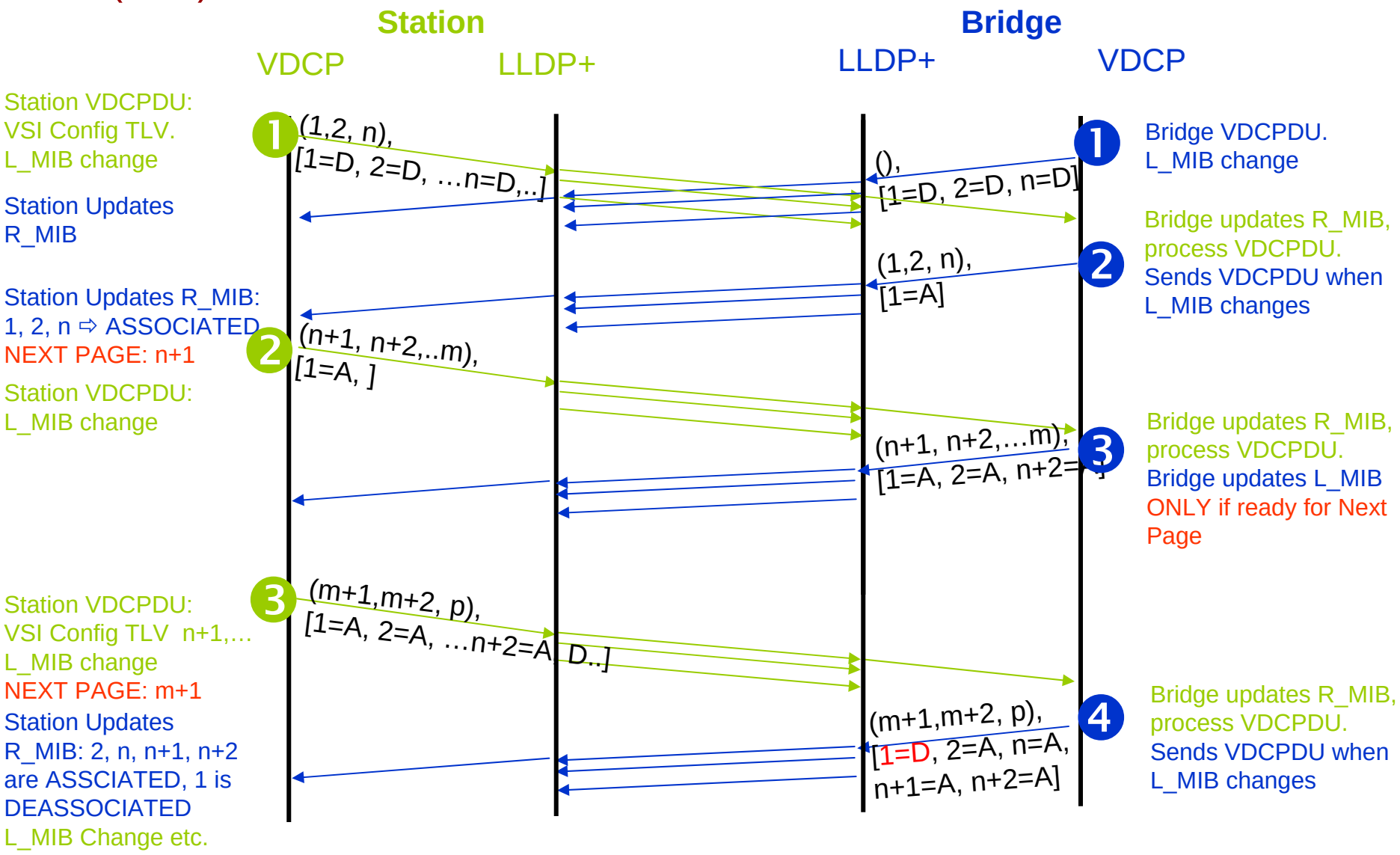



1. VDCP creates a VDCPDU (contains VSI Config TLV with all VSI State Change Requests <MTU, along with a Stable State TLV). VDCP updates the Local MIB and informs LLDP+ of "Local MIB Change" event.

2. LLDP+ transmits the new PDU due to LOCAL MIB Change event.

3. LLDP+ receives the PDU and informs VDCP of a "Remote MIB Change". VDCP consumes VDCPDU, MAY update its local VSI and configure accordingly. VDCP may update the Local MIB and informs LLDP+ of any "Local MIB Change" event.

# VDCP/LLDP+ Next Page Exchanges Example (2nd Option) (n=64)




Legend: VSI Config TLV (x,x,…x), Stable State TLV [x,x,…x]

# LLDP+

- Allow re-use of existing LLDP implementations with a potential parametric changes
  - LLDP+ is identified by a new EtherType or different version number
- Like LLDP
  - Addressing and TLV formatting
  - Only one outstanding message in each direction
  - Fast Re-xmt for reliability
  - Used to convey local configuration information to a link partner
  - Configuration information is exchanged by an external State Machine (DCBX, VDCP)
- Unlike LLDP
  - Optional LLDP TLVs are not carried. Mandatory (TBD)
  - Increased Credit on number of messages / Second

# Summary

## EASE of STANDARDIZATION:

- LLDP+ may require less standardization effort for 802.1 WG

## MEETING THE NEED

- LLDP+ and VDCP address the needs of VSI discovery
- No Need for a new “Transport” protocol

## EASE OF IMPLEMENTATION

- LLDP+ may be implementation friendly for LLDP capable DCB devices
  - High likelihood that existing LLDP implementations may be able to adapt to LLDP+
- Station and Bridge control resources needed

## TIME TO MARKET

- Adopting LLDP+/VDCP for VSI Discovery provides Faster Time to Market