Why Fibre Channel over Ethernet?

R Snively
rsnively@brocade.com

408-333-8135

With grateful acknowledgement to John Hufferd, Anoop Ghanwani, Dave Peterson, and Steve Wilson, all from Brocade.

Introduction

- The 3 classes of interconnects
- Example of cross-pollination: iSCSI
- Our example of cross-pollination: FCoE
- CEE Ethernet
- The model
- FCoE
- FIP

The three classes of interconnects

- Storage
- Clustering
- Networking

Storage

- Requirements
  - Pre-allocated data locations
  - Extreme reliability requirements
  - Large latency multipliers effect performance
  - Simple communications (mostly READ or WRITE)
  - Rapidly increasing scalability requirements
- Examples
  - FIPS-60, ESCON, FICON, FCP over FC, SCSI, SAS, SATA

Fibre Channel, a storage example

- Achieves goals with
  - High reliability links
  - HBA co-processor (simple because protocol is simple)
  - Delivery of data straight to allocated memory
  - Contains context information directly in frame.
  - Flow control at both link level and end-to-end eliminates need to drop frames.
  - Data path topologies tend to be simple, use FSPF routing.
  - Integrated management.

Clustering

- Requirements
  - RDMA
  - Extremely low latency
  - Latency has significant performance impact
  - Large latency multipliers effect performance
  - Simple communications (mostly GET or PUT)
  - Modest scalability requirements
- Examples
  - SCI, HIPPI, InfiniBand, VI, Myranet, others

Networking

- Requirements
  - Data allocation determined by analysis of frames
  - Reliability requirements met only at upper layers.
  - Latency of links is third-order performance impact.
  - Sophisticated application level communications
  - Essentially infinite scalability requirements
- Examples
  - Ethernet, Ethernet, Ethernet, Ethernet, FDDI, SONET, ATM, DSL

Ethernet, a network example

- Achieves goals with
  - Best effort links
  - Simple cheap NICs with host-based frame analysis
  - Delivery of data to buffer, analyzed and moved to destination.
  - Contains context information at higher levels.
  - Flow control at higher levels or not at all. May drop frames.
  - Data path topologies extremely complex.
  - Most management is out of band.

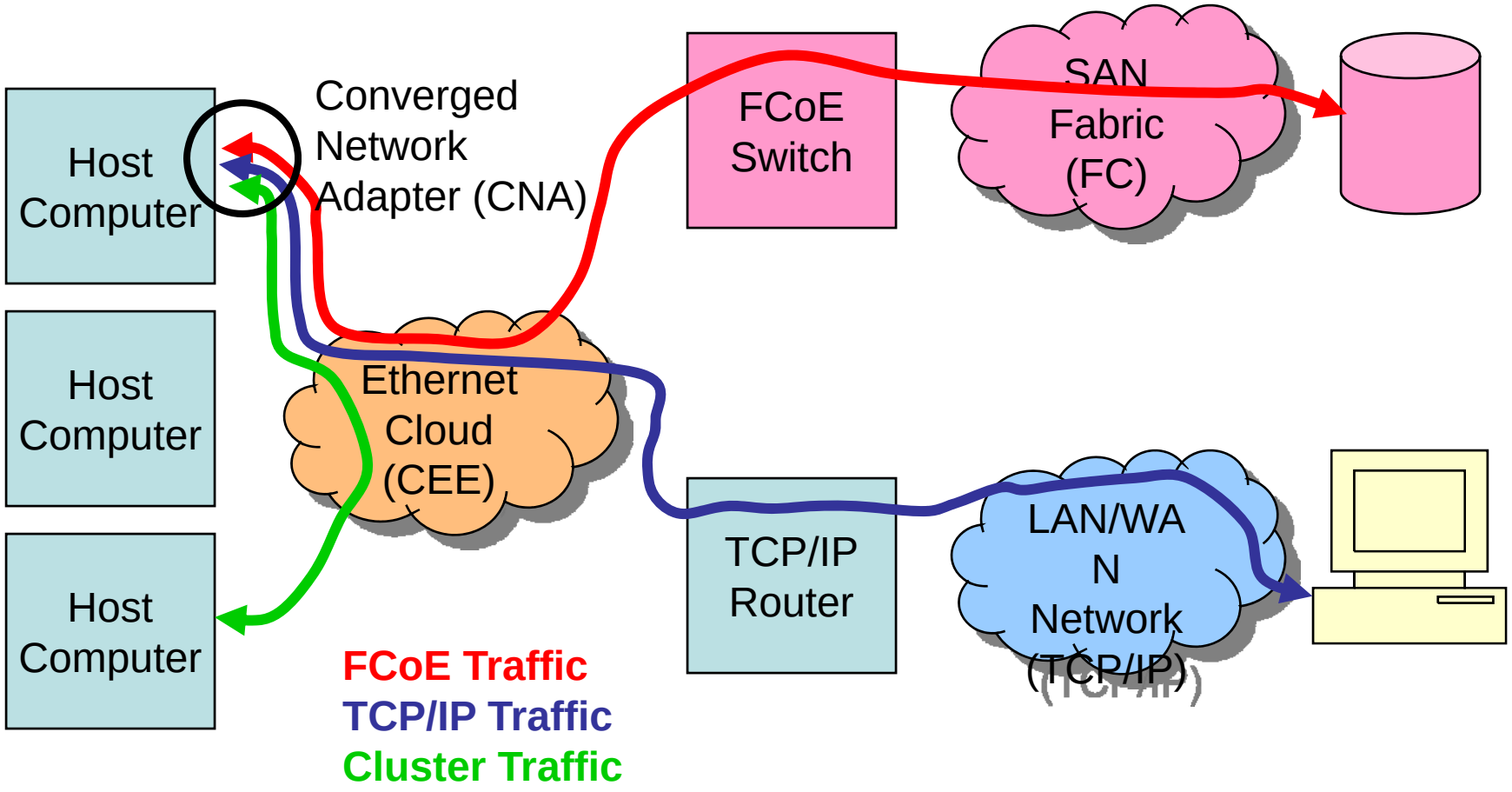
Why shove storage traffic onto a network?

- Because it comes in handy sometimes
  - iSCSI
    - Accepts penalties associated with TCP/IP
    - Accepts penalties associated with latency of networks
    - Gains advantages of LAN/WAN scalability
    - Gains advantages of LAN/WAN distances
    - Low-cost implementations possible if performance goals are modest.

Another way to shove storage traffic on to networks: FCoE

- It is useful enough to justify changing Ethernet to **Converged Enhanced Ethernet**
  - FCoE
    - Accepts scalability penalties of L2-only Ethernet cloud.
    - Accepts mini-jumbo frame requirement for maximum performance.
    - Accepts penalty of installing new lossless Ethernet bridges.
      - Requires additional flow control (PFC)
      - Requires enhanced transmission selection (ETS)
      - Data Center Bridging eXchange (DCBX)
    - Accepts penalty of possible traffic interactions.
    - Gains advantage of one-port-fits-all configuration.
    - Maintains optimum protocol efficiency.
    - Maintains advantage of full TCP/IP and full FC compatibility.

Architectural Overview:



Mini-jumbo frame support

- Mini-jumbo frames (No standard defined)
  - Standard Ethernet Frame

46 - 1500 Data Bytes +18 headers

  - Standard FC frame

0 - 2112 Data + (28 to 52 headers)

  - Need mini-jumbo Ethernet frame to fit FC frame

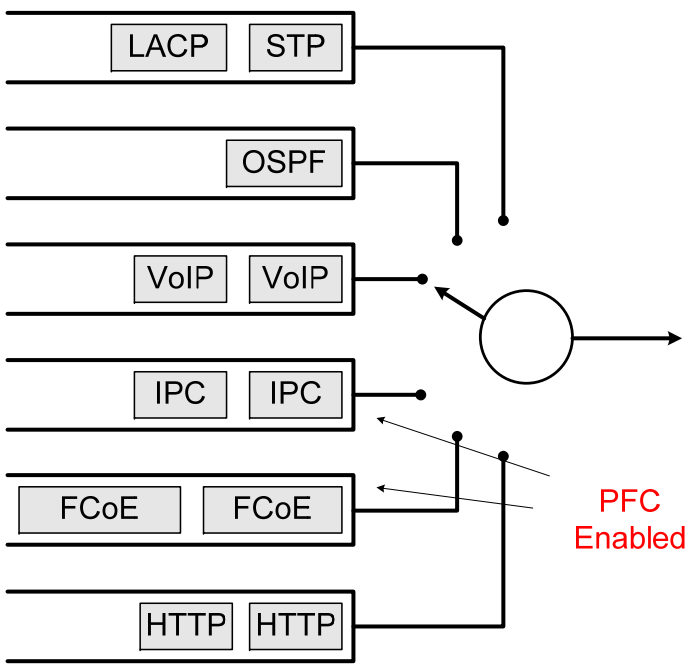
64 – 2200 Bytes required in mini-jumbo frame

CEE Standards developed in IEEE

- Priority Flow Control (IEEE 802.1Qbb)
- Enhanced Transmission Selection (ETS) (IEEE 802.1Qaz)
- DCBX (Probably included in IEEE 802.1Qaz)

- Note that other related work is not required for CEE, though it may ultimately be used in some FCoE implementations.
  - Congestion Notification (IEEE 802.1Qau)
  - Multi-pathing (IETF TRILL project among others)

Priority-based Flow Control (PFC)

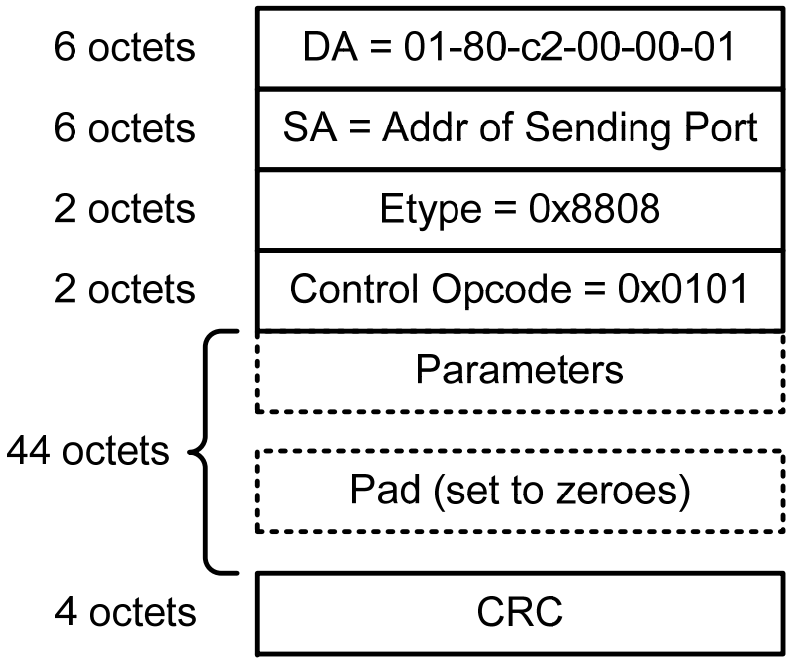


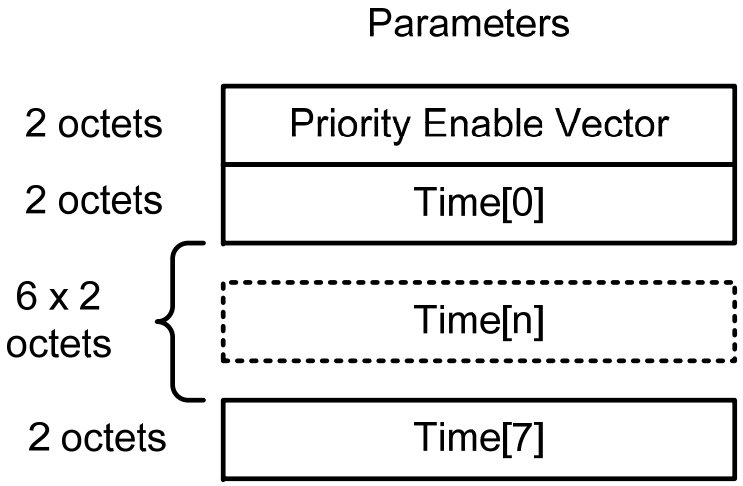
- Why PFC?
  - Link level flow control is the only way to guarantee zero loss due to congestion
  - 802.3x pauses all traffic including control traffic
  - PFC only affects priorities that need it; e.g. the priority used by FCoE

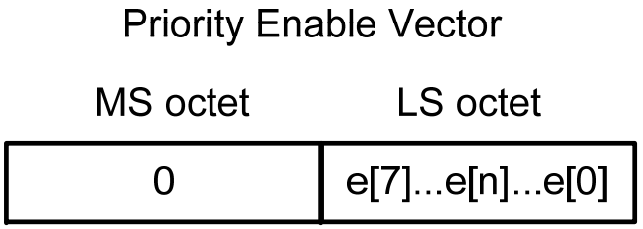
PFC Implementation Conventions

- May be applied to each priority independently; i.e. support enable/disable transmission and receipt of pause frames
  - Both transmission and receipt are enabled/disabled simultaneously
- Frame format for PFC along with Ethertype and MAC address
- Specifies the minimum delay in reaction to pause that a receiver must be able to deal with
- After PFC is negotiated (using DCBX), 802.3x shall not be used
  - Don't generate 802.3x pause frames
  - Ignore received 802.3x pause frames

PFC Frame Format







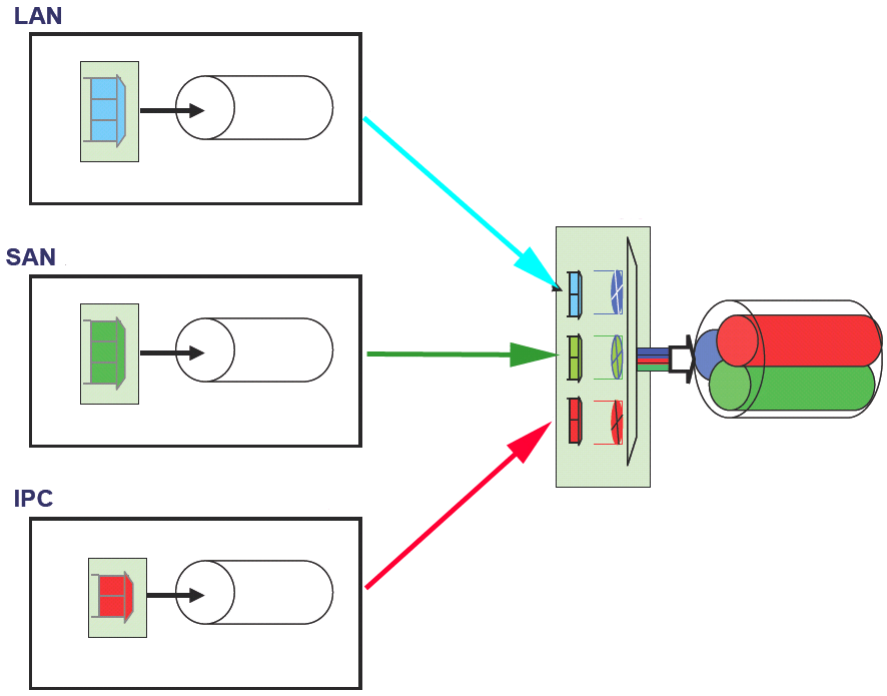
E[n] = 1 => Time[n] valid
E[n] = 0 => Time[n] invalid

- The fields Time[n] for n = 0..7 are always present
- If e[n] = 0, Time[n] is zero on transmit and ignore on receipt

PFC Deployment Considerations

- Pause and link level flow control are known to have several drawbacks
  - Deadlocks – routing and other deadlocks
  - Congestion spreading
- Solutions exist for these problems
  - Use 802.1Qau to deal with sustained congestion
  - Use deadlock avoidance schemes, including simplified topologies and dropping frames in the event a deadlock condition is detected.

Enhanced Transmission Selection



From ETS proposal to IEEE 802.1

- Why ETS?
  - In a converged network, different traffic type have different needs
  - Strict priority which is the default defined in 802.1Q falls short of meeting the needs
  - Several pre-standard solutions exist for addressing the shortcomings of strict priority
- ETS provides a consistent management framework for assigning bandwidth to traffic classes
- It also defines a minimum behavior for the scheduler to ensure some minimum capability across vendors

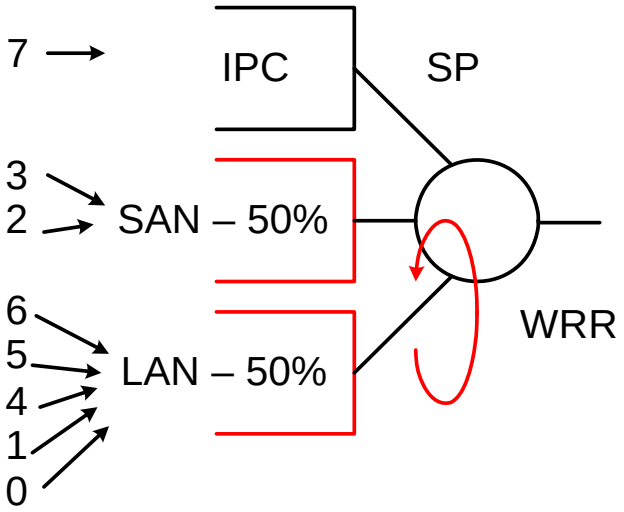
ETS Implementation Agreement

- Combined priority/WRR scheduler
- Each priority is assigned to a priority group
  - A priority group is represented by 4-bits
  - Must be 0..7, 15 (8..14 are reserved)
- A priority may be designated as being strict priority by being assigned PGID 15
- After all priorities in PG15 are served using strict priority, start serving all the other groups in proportion to their weight
- Weight is expressed as a percentage of remaining bandwidth with a resolution of 1%
- Bandwidths must add up to 100%; if they don't behavior is undefined
- A scheduler can pick the closest value that it can support – to within +/- 10%
- Minimum requirement
  - 3 groups -- PGID 15 and 2 priority groups with weights
  - The scheduler should be configurable on a per-port basis

Priority Groups and Bandwidth Assignment – An Example

Priority	PGID	Desc
7	15	IPC
6	1	LAN
5	1	LAN
4	1	LAN
3	0	SAN
2	0	SAN
1	1	LAN
0	1	LAN

PGID	BW%	Desc
15	-	IPC
0	50	SAN
1	50	LAN
-	-	-



- There are 3 priority groups in use – IPC, LAN & WAN
- First all IPC traffic is serviced (pri 7); it is marked as is assigned PGID 15
- Next, the available bandwidth is shared equally by LAN (pri 6,5,4,1,0) & SAN (pri 3,2)
- Within a group, scheduling is not specified; e.g. with the LAN traffic an implementation could map pri 6,5,4,1,0 to a single queue, or map them to separate queues and do round robin or priority between them – hierarchical scheduling

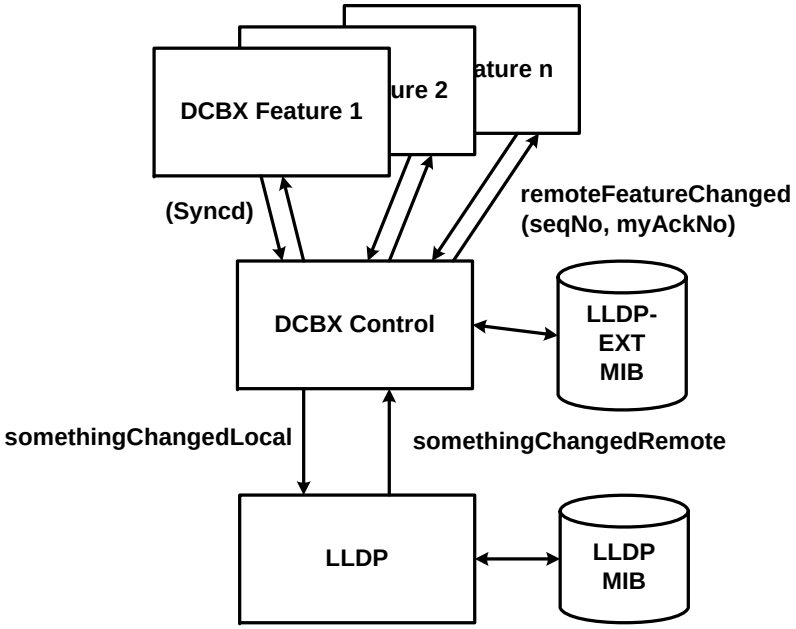
ETS Configuration Guidelines

- Within a PG, PFC must be enabled (or disabled) on all priorities within the group; a mix of PFC and non-PFC priorities within a single PG should be avoided
- The default configuration is the same as 802.1Q – strict priority
  - All priorities are assigned PGID 15
  - The bandwidth table contains only PGID 15 with no bandwidth allocation

Data Center Bridging eXchange (DCBX) Protocol

- Why DCBX?
  - Defines the limits of the CEE-capable cloud
  - Detects mis-configuration between peers
  - May be used to configure a peer
  - Allows for coexistence with legacy devices

DCBX & LLDP



- DCBX has 2 state machines
  - DCBX control
  - DCBX feature
- somethingChangedLocal is used by DCBX control to inform LLDP that a new LLDPDU must be sent
- somethingChangedRemote informs DCBX control that LLDP has received new information from the peer
- remoteFeatureChanged informs DCBX Feature state machine that a remote feature has changed
- seqNo and myAckNo are maintain in DCBX control but visible in DCB feature state machines
- Syncd is maintained in each of the DCBX feature state machines but is also visible in DCBX control
- Extensions are defined to the LLDP MIB for DCBX and each of the features

DCBX Handshaking With Peer

- DCBX control TLV contains a seqNo and ackNo
- Local system initializes and populates a seqNo in LLDP messages
- seqNo is modified when any of the local configuration changes
- The ackNo tells the peer the last seqNo that was received
- In this way, a system knows what information about it has been received by its peer

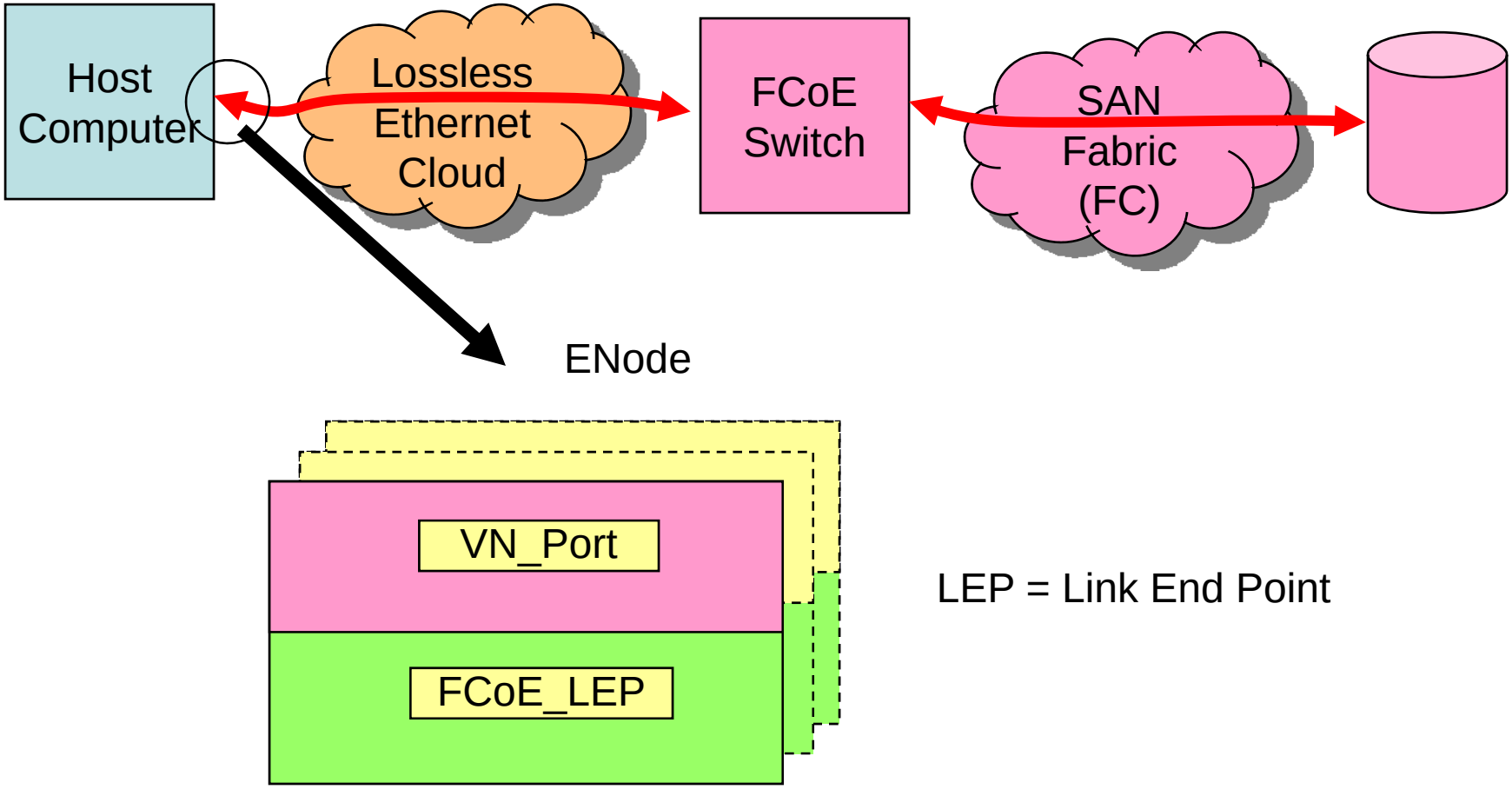
DCBX Sync With Peer

- PFC configuration must match with peer on all priorities, otherwise PFC is disabled on the link and an error is flagged
- ETS configuration need not match
- If one end is "Willing" and the other is "Not Willing", then the "Willing" switch port accepts the configuration from the "Not Willing" port

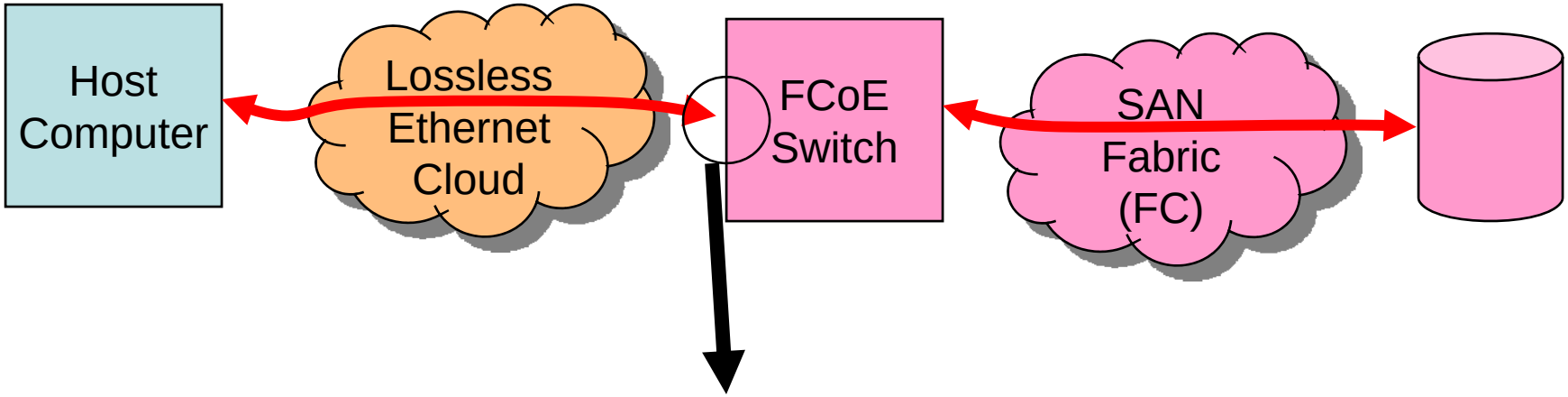
Vocabulary Drill

- INCITS:
  - InterNational Committee for Information Technology Standards. (Accredited by ANSI.)
- INCITS Technical Committee T11
  - Responsible for all Fibre Channel standards
- FC-BB-5
  - The standard that defines the Fibre Channel parts of Fibre Channel over Ethernet
- FCoE
  - In FC-BB-5, officially called the FC-BB_E model.

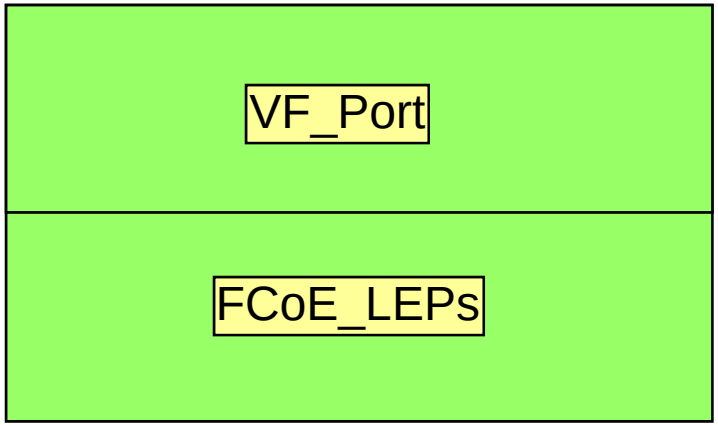
Architectural Overview: CNA



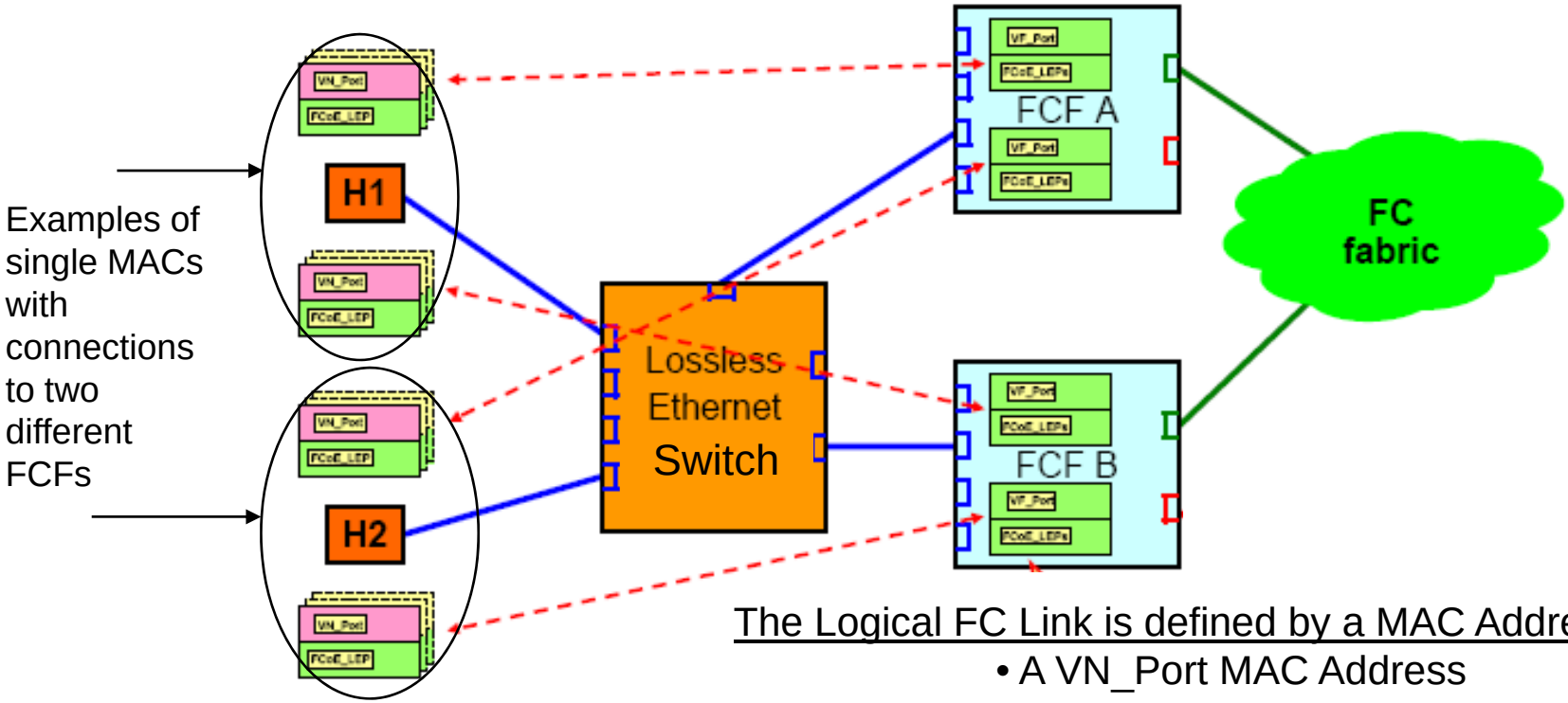
Architectural Overview: FCF



Fibre Channel Forwarder (FCF)



Multiple Logical FC connections via a Single Ethernet MAC



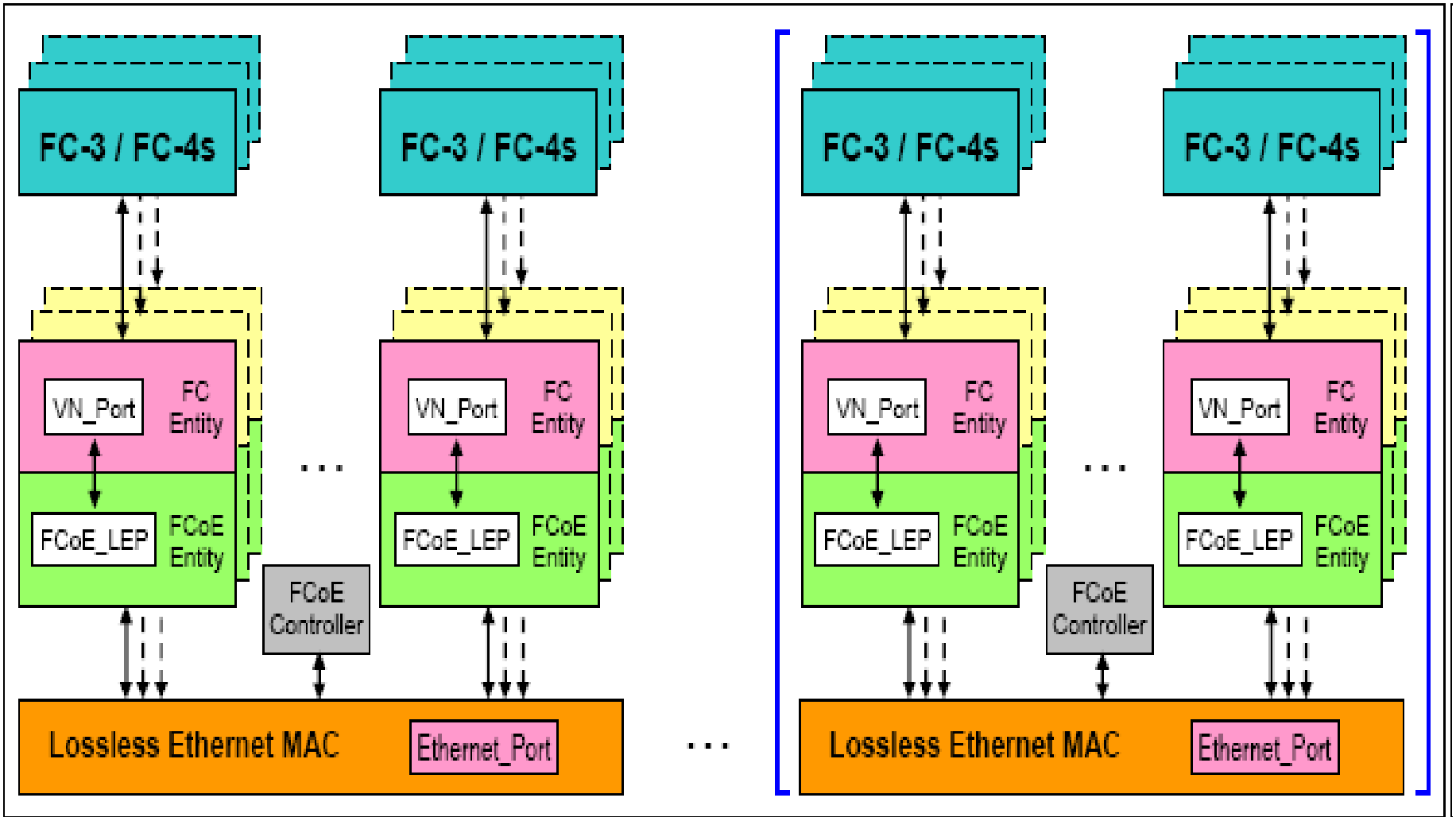
The Logical FC Link is defined by a MAC Address pair

- A VN_Port MAC Address
- A VF_Port MAC Address

For a logical FC link the FCoE Frames are always sent to and received from a specific FCF's MAC Address

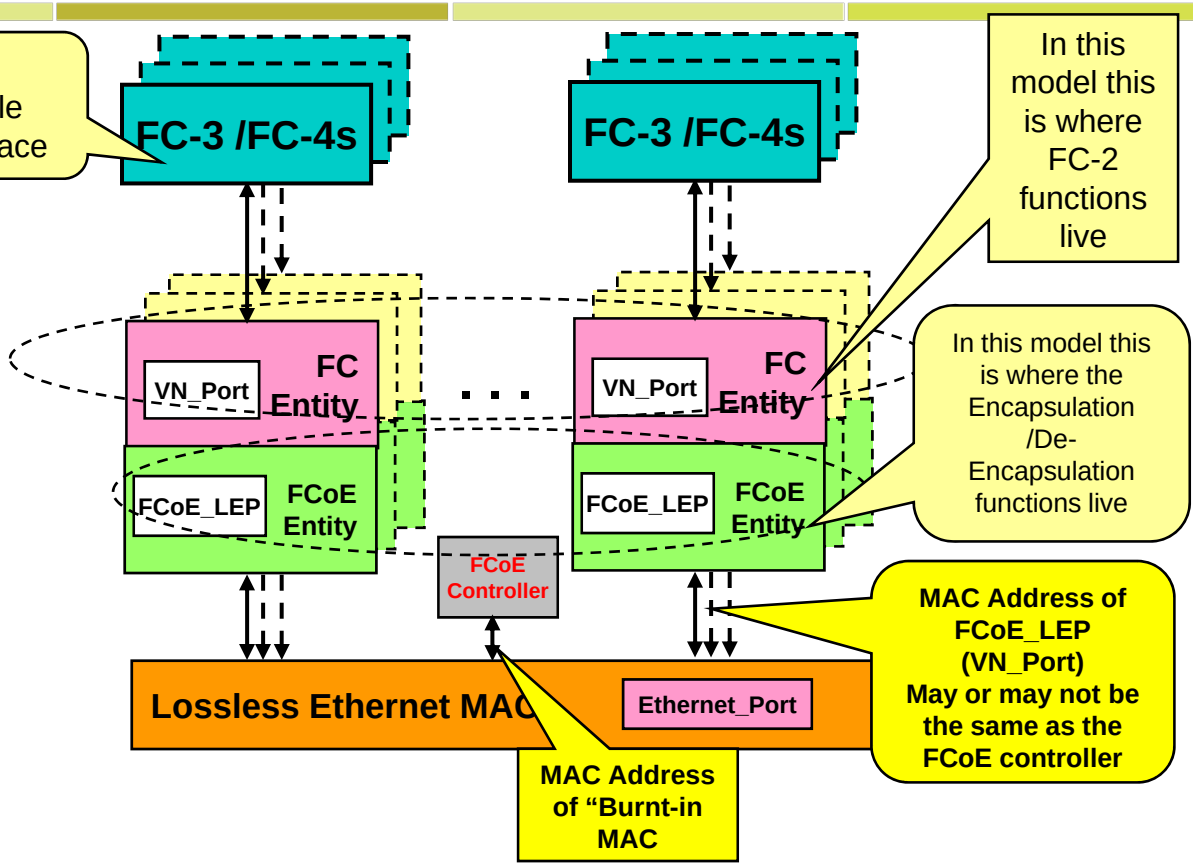
- Therefore, pathing to and from the FC driver is always defined by the MAC Address of the partner FCF's VF_Port

Model for the ENode



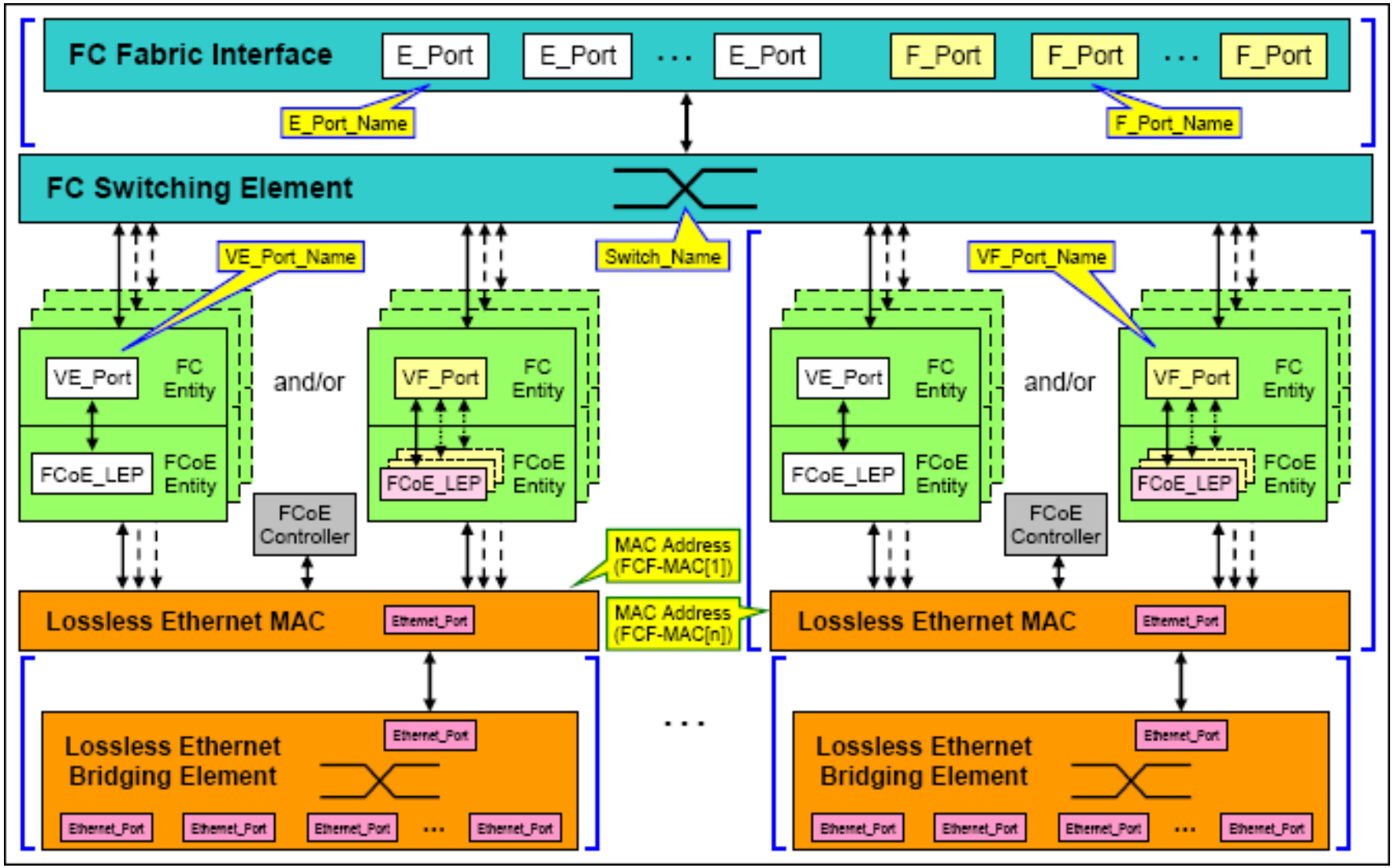
Model of the ENODE with Multiple Logical FC interfaces

- For each logical N_Port (VN_Port) there is one FLOGI and perhaps 100's of FDISC
- Each VN_Port is seen by the Host as a separate (logical) FC connection
- The number of (logical) FC connections is implementation dependent

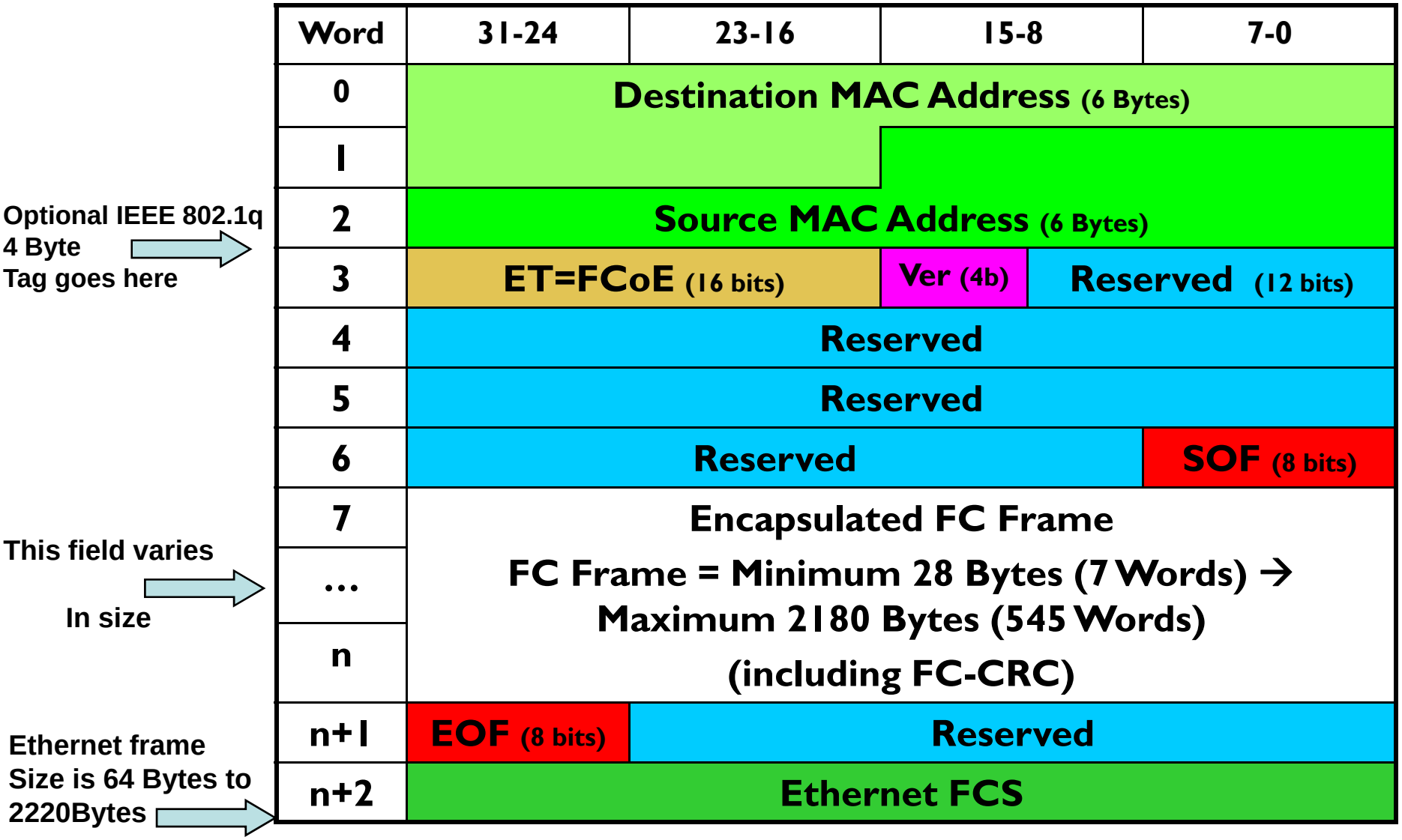


- Only one MAC Address is required for the FCoE Controller and the VN_Ports on a single physical MAC (aka Server Provided MAC Address – SPMA)
- FCF may chose to specify new MAC addresses for each VN_Port (aka Fabric Provided MAC Address – FPMA)

Model for the FCF



FCoE, Ethertype '8906'h



After that, everything is easy

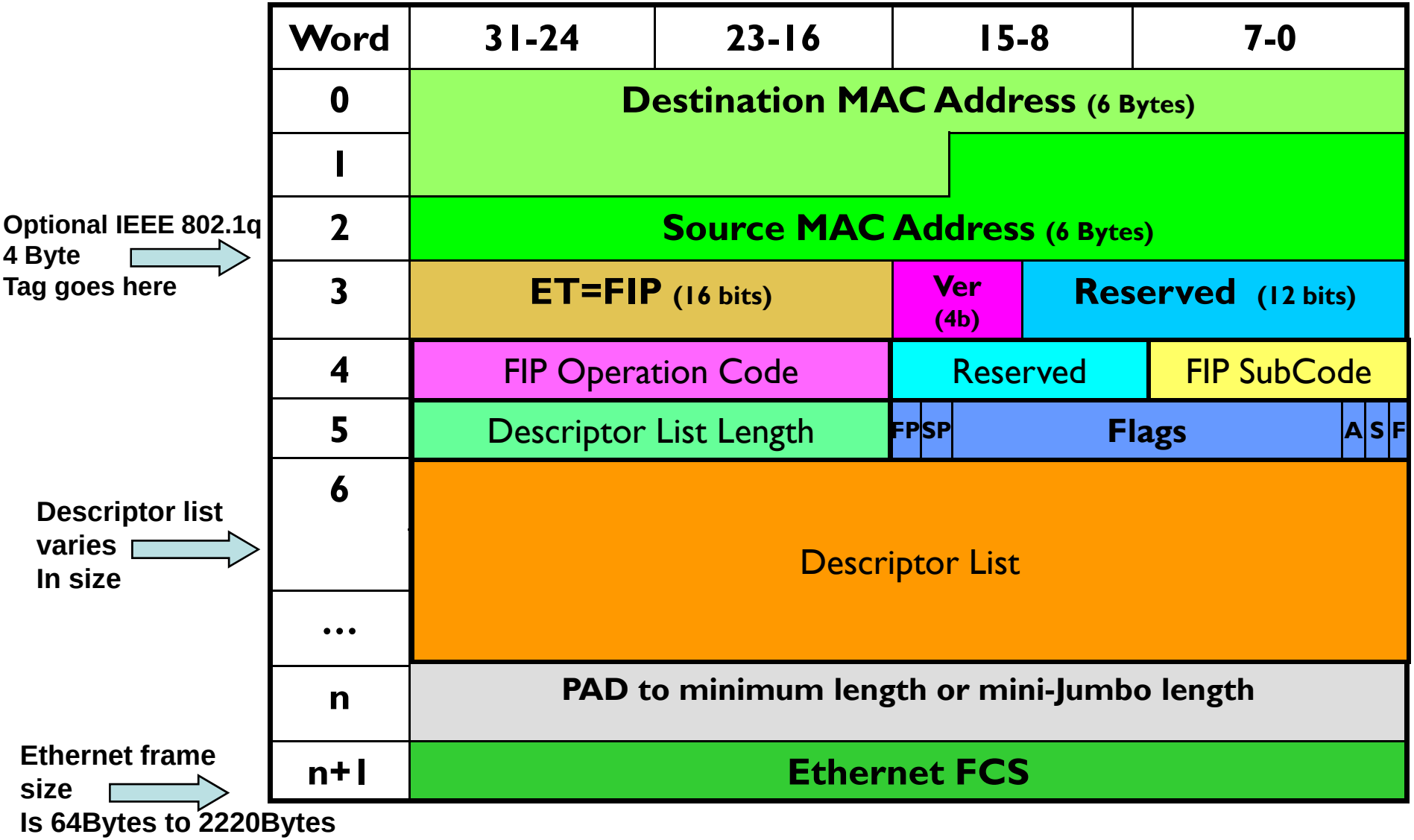
- All Class 2 and 3 protocols work, including:
  - FCP (SCSI Fibre Channel Protocol)
  - FC-SB-3 (FICON)
  - RFC 4338 (Transmission of IPv6, IPv4, and Address Resolution Protocol (ARP) Packets over Fibre Channel)
  - Security
  - etc.
- Link level flow control uses Ethernet conventions

A few little details left...

- How do you find the devices?
- How do you determine the allowable frame sizes?
- How do you associate FC FC_IDs and MAC addresses?
- How do you keep a virtual link alive in an Ethernet L2 cloud?
- How do you clear links?

- Solution is a second protocol.

FIP, Ethertype '8914'h



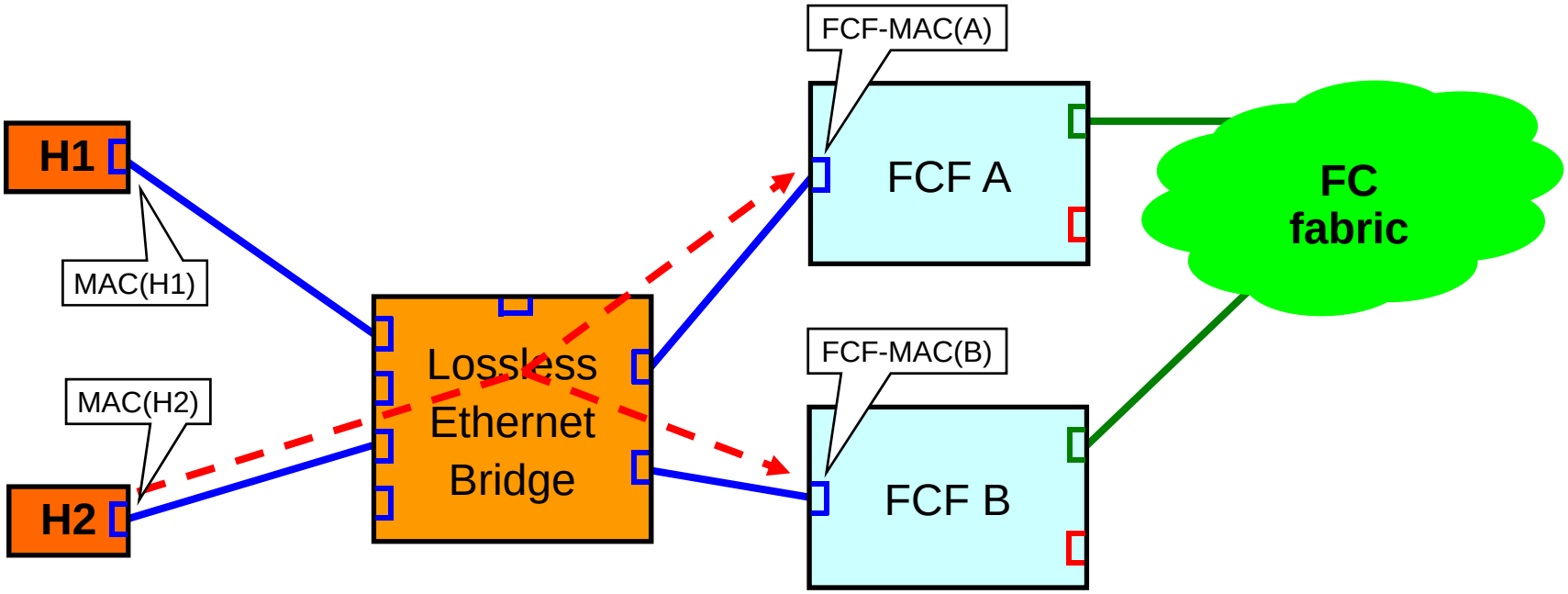
FIP Operation Codes

Operation Code	Subcode	Operation
0001h	01h	Discovery, Solicitation
	02h	Discovery, Advertisement
0002h	01h	Link Service Request
	02h	Link Service Response
0003h	01h	FIP Keep Alive
	02h	FIP Clear Virtual Link
FFF8h .. FFFEh	00h .. FFh	Vendor Specific
All others	All others	Reserved

An example of FIP

- An ENode solicits with a broadcast Solicitation frame to identify any FCFs that it can use to make a virtual connection.
- Applicable FCFs respond with unicast Advertisement frames.
- After deciding which FCFs are appropriate, the ENode sends an FLOGI Request.
- The selected FCF responds with an appropriate FLOGI Response accepting or rejecting the Request.
  - The FLOGI Response assigns the MAC address that the Enode will use for later FCoE frames with the VN_Port.
- And now the FCoE virtual link is up and ready for normal FC use.

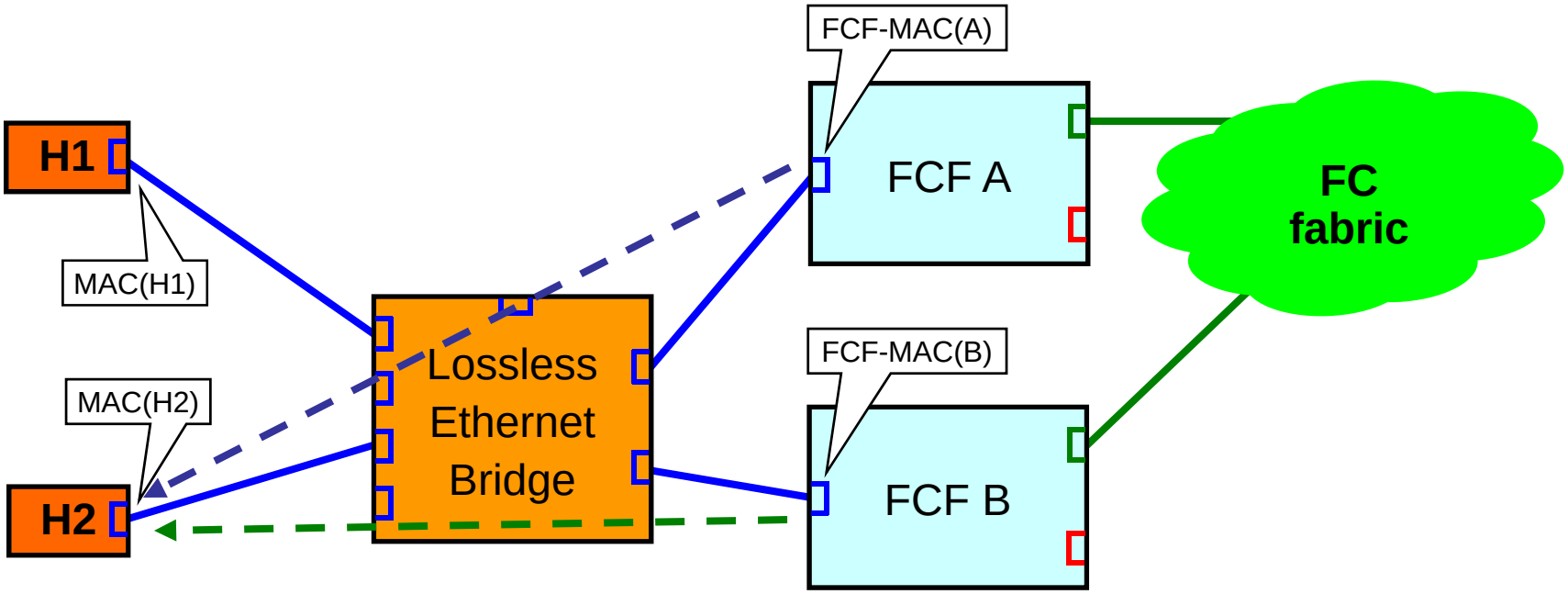
Multicast Solicitation from H2



All-FCF-MACs
MAC(H2)
Solicitation [From ENode, Addr modes supported, MAC(H2), H2 Port_ID,Max_Rcv_Size]

F=0b, this Solicitation solicits VF_Port capable FCF-MACs

Unicast Advertisements from FCFs



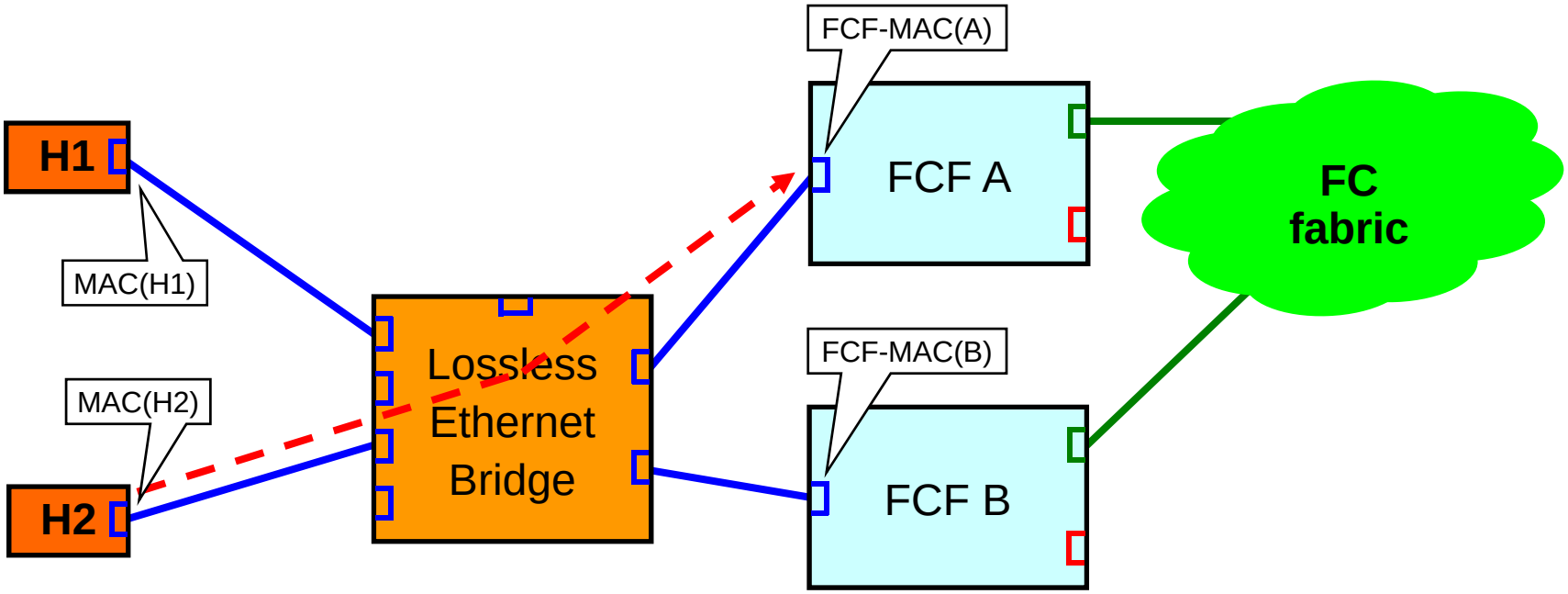
H2's FCF list:

FCF-MAC(A) [J]
FCF-MAC(B) [J]

MAC(H2)
FCF-MAC(A)
Jumbo Advertisement [Solicited, From FCF, Address Modes Supported, Priority, FCF-MAC(A), FC-MAP, Switch_Name, Fabric_Name]

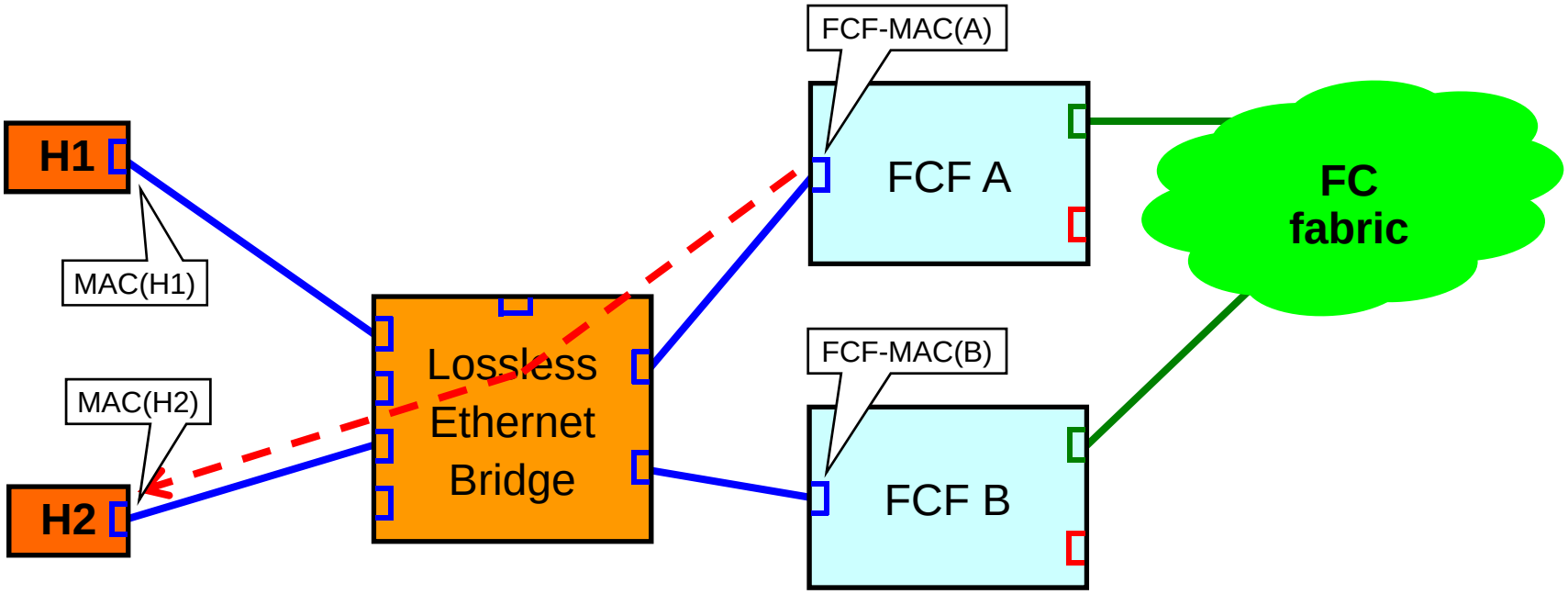
MAC(H2)
FCF-MAC(B)
Jumbo Advertisement [Solicited, From FCF, Address Modes Supported, Priority, FCF-MAC(B), FC-MAP, Switch_Name, Fabric_Name]

FLOGI from H2 to FCF_A



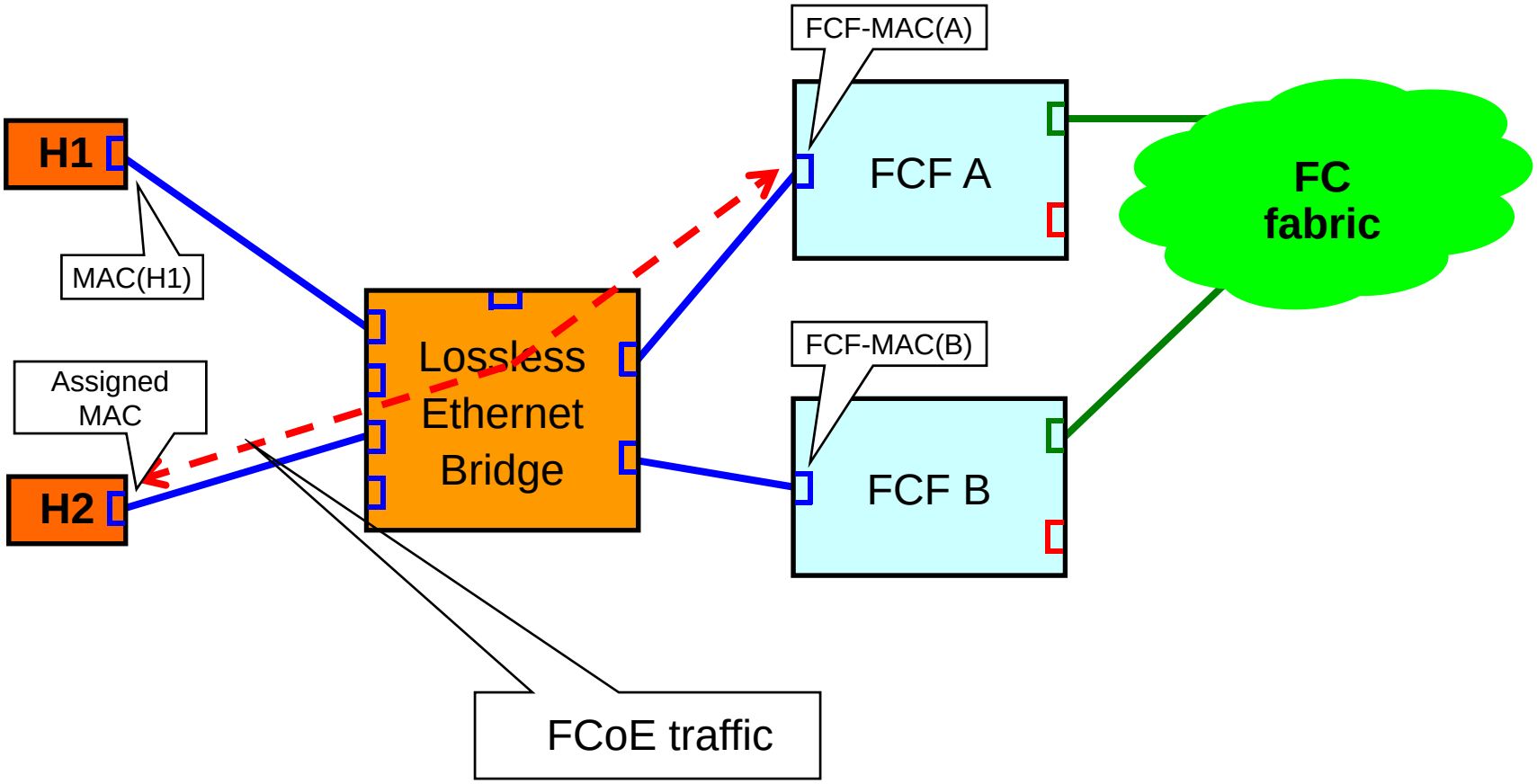
FCF-MAC(A)
MAC(H2)
FLOGI Request [From ENode, Addr mode supported, Proposed MAC address, FLOGI Request information]

FLOGI Response to H2



MAC(H2)
FCF-MAC(A)
FLOGI Response [From FCF, Addr mode assigned, Assigned MAC address, FLOGI Response information]

Resulting virtual FC link runs FCoE



More details of FCoE

- The remainder of this work is “an exercise for the student”.
- References:
  - To become active in IEEE 802.1 and IEEE 802.3 see:
    - http://grouper.ieee.org/groups/802/dots.html
  - To become active in T11:
    - www.t11.org