13-134v0

FDF Joining Distributed Switch Fabric Serialization (Part 2)

Patty Driever (IBM)

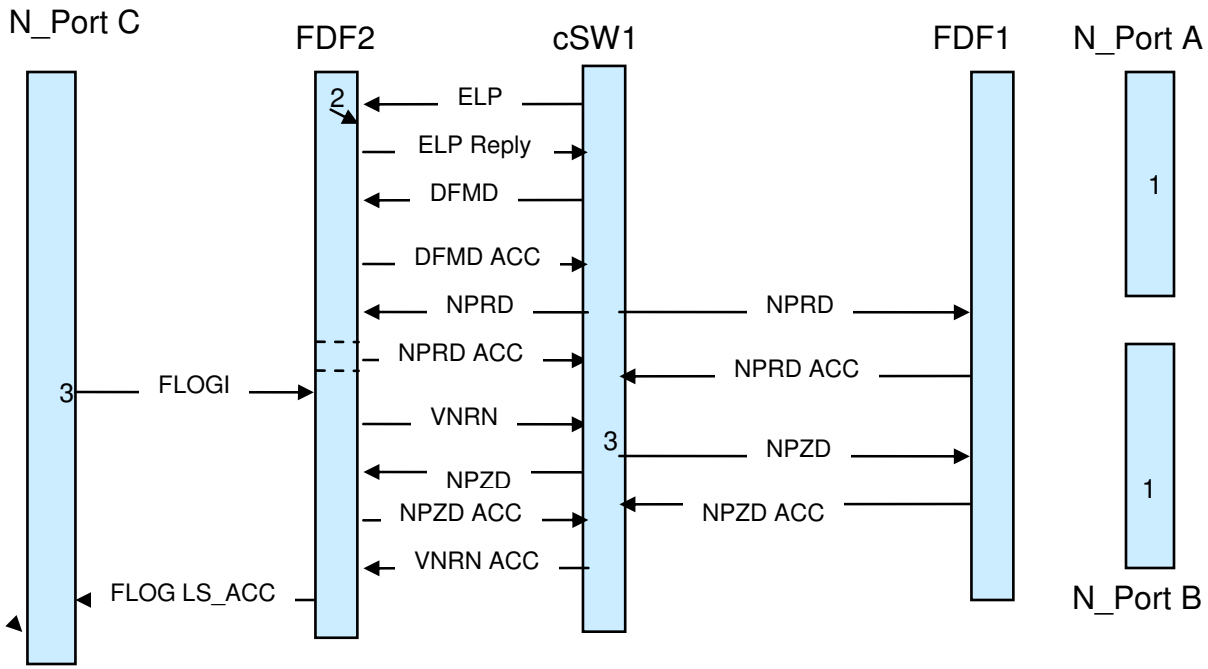
Claudio Santi (Cisco)

Today's Defined Process for Switch Joining Fabric

- **DFMD sent by controlling switch to FCDF newly joining the fabric**
- **NPRDs are generated to communicate routing information**
  - Sent by controlling switch to all FCDFs (current plus new member)
  - Per 12-459v0 adopted text (posted as 13-121v0), after this exchange the new FCDF can accept N_Port logins
- **NPZDs are generated when a new N_Port logs in to an FCDF that is already a part of the fabric**
  - Sent by controlling switch to all FCDFs in the fabric
  - N_Ports can log in to new FCDF once NPRD exchange is completed with controlling switch

Scenario 1 (Good Case – non-cascaded switches)

1. **FDF1 is part of fabric with N_Port IDs A and B**
2. **FDF2 joins fabric, so NPRD is sent to FDF1 informing it of the new switch and to FDF2 to pass routing info**
3. **N_Port ID C logs into FDF2, so NPZD is sent to FDF1 informing it of the new port**

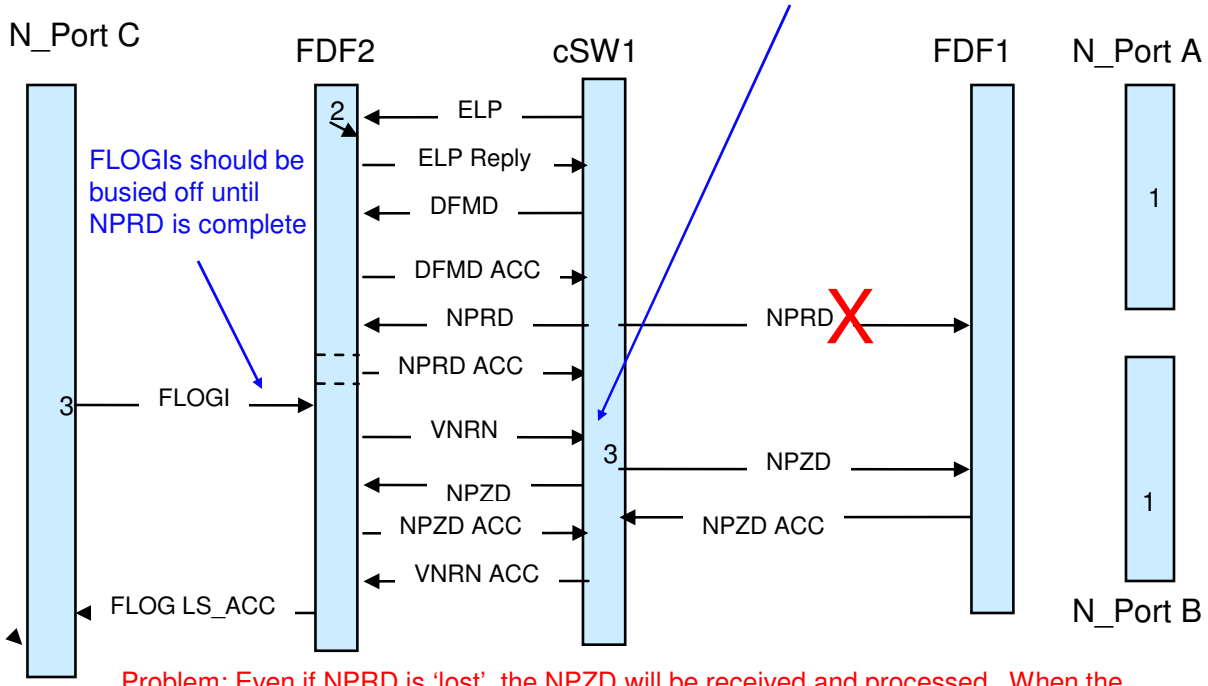


Scenario (NPRD to FDF1 is dropped)

1. **FDF1 is part of fabric with N_Port IDs A and B**
2. **FDF2 joins fabric, so NPRD is sent to FDF1 informing it of the new switch and to FDF2 to pass routing info**
3. **N_Port ID C logs into FDF2, so NPZD is sent to FDF1 informing it of the new port**

Since FLOGIs to the new FCDF are held off until NPRD processing completes, the NPRD timeout must be relatively short in order to allow for case where NPRD needs to be retried

Solution: Do not send NPRD to the newly joining FCDF until NPRDs have been completed with all other FCDFs in the fabric.

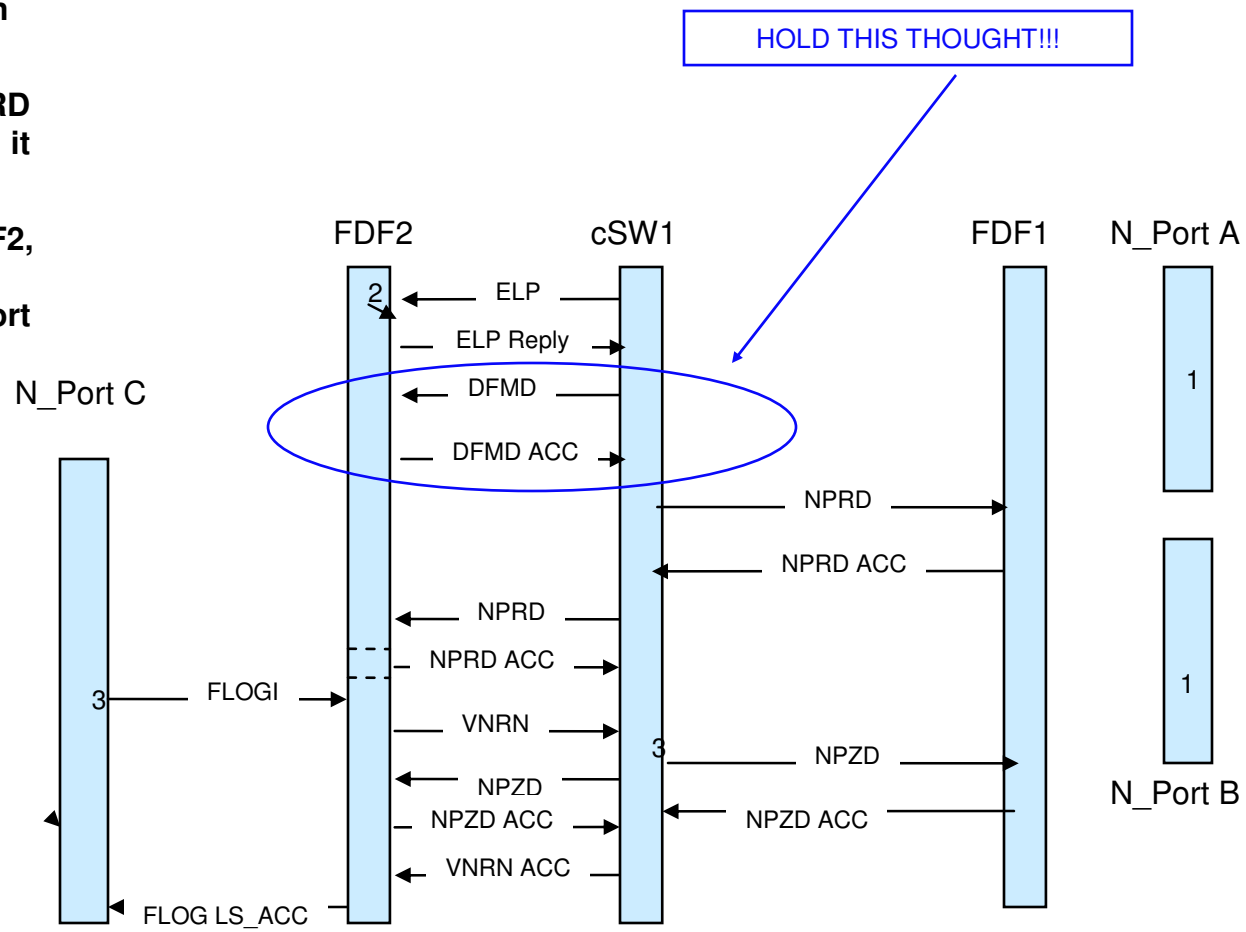


Problem: Even if NPRD is 'lost', the NPZD will be received and processed. When the NPZD completes the VNRN completes and the FLOGI is accepted. N_Port C can now log in with N_Port A, but N_Port A cannot respond until the NPRD times out and is retransmitted and accepted (so it has its routing information)

Scenario 1 (Solution – serialize NPRDs to existing FCDFs before sending NPRD to new FCDF)

1. **FDF1 is part of fabric with N_Port IDs A and B**
2. **FDF2 joins fabric, so NPRD is sent to FDF1 informing it of the new switch**
3. **N_Port ID C logs into FDF2, so NPZD is sent to FDF1 informing it of the new port**

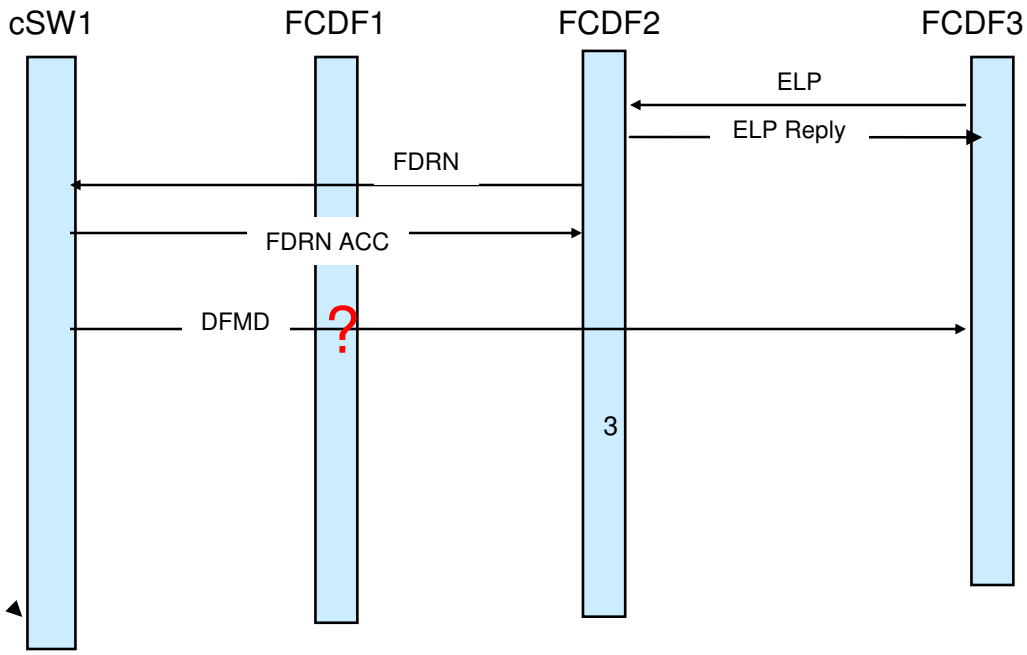
Next Step:
Need to create text to describe this



Scenario 2 (cascaded switches)

1. **FCDF1 and FCDF2 are part of fabric, cascaded together**
2. **FCDF3 joins fabric, again serially cascaded off of FCDF2**
3. **Rule (with resolution of Scenario 1) is DFMD is sent to new switch, and then NPRDs are sent to FCDF1 and FCDF2 and then NPRD is sent to the new FCDF (i.e. FCDF3)**

...BUT FCDF1 doesn't yet have routing information for how to reach FCDF3 at the time the DFMD arrives (i.e. it has not yet received the NPRD)



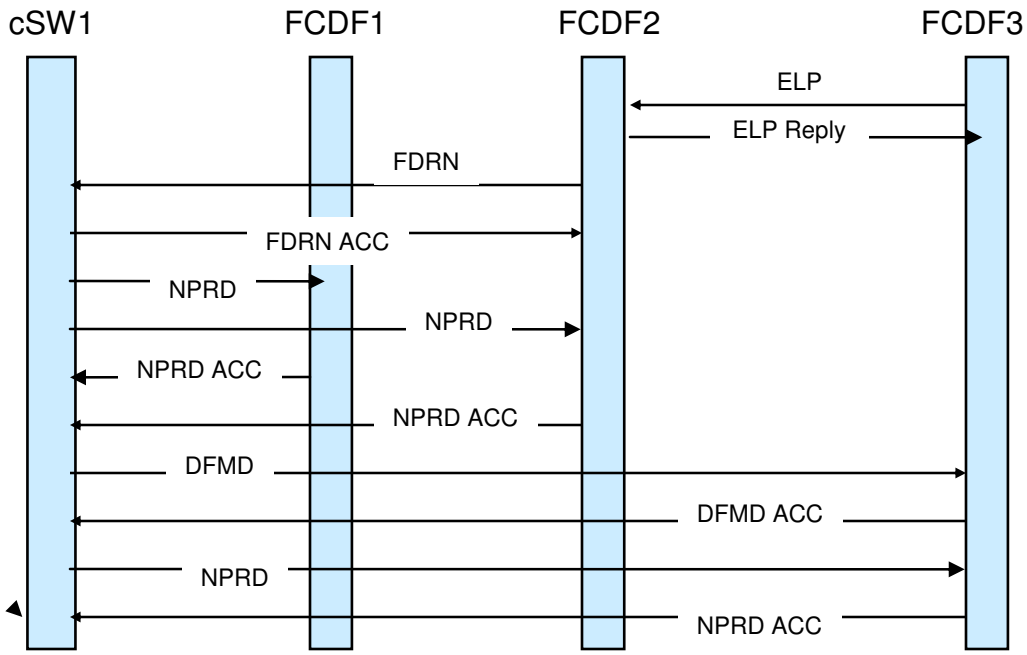
Proposed Solution

- **When a new FCDF joins the fabric:**
  - NPRDs are sent (and completed) to the switches that are already part of the distributed fabric informing them of how to reach the new FCDF
  - DFMD is sent to the newly joining switch to pass the distributed switch membership information to it
  - NPRD is sent to the newly joining switch to pass routing information to all the switches/N_Ports that have previously joined the distributed fabric

Maintains existing order of SW_ILSs to the newly joining FCDF (DFMD followed by NPRD)

Scenario 2 (cascaded switches) Proposed Resolution

1. **FCDF1 and FCDF2 are part of fabric, cascaded together**
2. **FCDF3 joins fabric, again serially cascaded off of FCDF2**
3. **NPRDs are first sent to FDCF1 and FCDF2, followed by DFMD and NPRD to newly joining FCDF3**



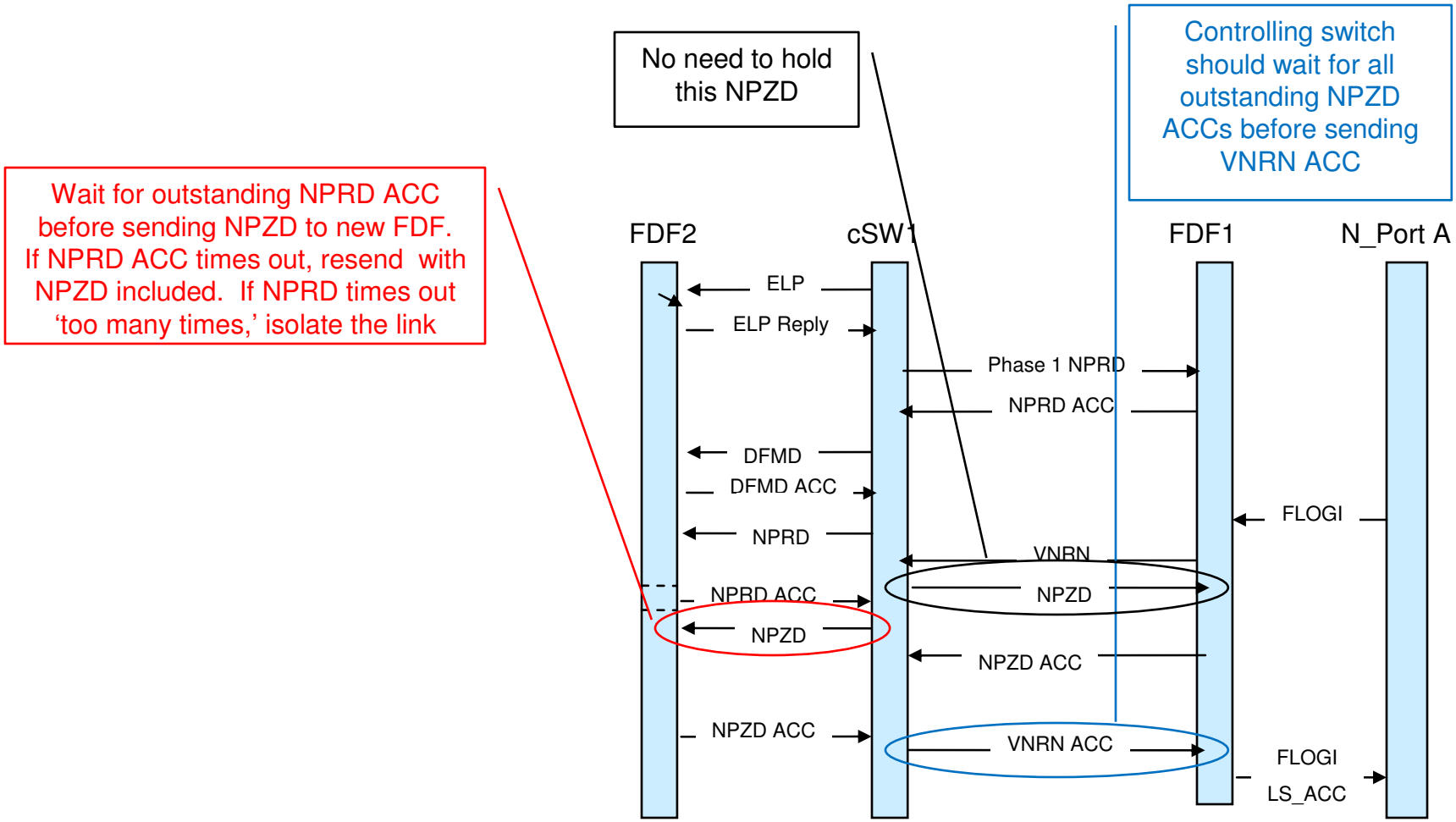
Serialization of NPZDs Relative to NPRDs

- **The initial NPRD received by a joining FDF contains routing information to existing FDFs in the fabric along with currently defined N_Port_ID ranges on those FDFs**
- **The controlling FCF must ensure that this NPRD has been received and processed by this newly joining FDF before sending to it any NPZDs that may be required due to new N_Ports on existing FDFs logging in or out**
  - If this initial NPRD exchange is not successful (e.g. timed out by the cFCF), the cFCF shall retransmit the NPRD, but the retransmitted NPRD will be updated to also include revised N_Port_ID ranges that reflect the information contained in any pending NPZDs
- **Once this initial NPRD has been received and processed by a newly joined FDF, no further serialization of NPZDs with NPRDs is required**
  - Subsequent NPRDs will announce the arrival of new FDFs and changes in routes to existing FDFs
  - Subsequent NPZDs will announce the allocation or deallocation of N_Port_IDs on existing FDFs
  - Serialization rules for NPRD processing when a new FDF joins the fabric (prior slides) eliminate any potential of receiving an NPZD about an N_Port on a 'yet unknown' FDF
  - Sequence numbers on NPZDs eliminate issues of missing or 'out of order' NPZDs
- **While all NPRDs contain a list of N_Port_IDs allocated to each FDF, this information is only *needed* on the first NPRD received by a newly joining FDF**
  - All subsequent changes to the N_Port_ID allocations are communicated via NPZDs, so FDF should never be 'out of synch' with the cFCF regarding reachable N_Port_ID ranges

However....

- **Since the information is passed in each NPRD, and cFCF is the authoritative source of such information:**
  - Is there any value in having the receiving FDF validate the N_Port_ID ranges reported in subsequent NPRDs matches its current mapping?
    - If they did such validation and the mapping doesn't match, what should it do? Report an error? Update its internal mapping to match what's in the NPRD?
  - Should the standard explicitly identify these behaviors?
    - Should add text to 17.7.3.6 (NPRD) to state explicitly that when an NPRD is received that all routing ***and*** N_Port_ID reachability information present in the FDF is replaced.
    - Must also serialize ***all*** NPZDs after ***all*** NPRDs (not just the after the initial NPRD), to ensure integrity of N_Port_ID reachability lists
      - If we didn't serialize, then if an NPZD sent after an NPRD is received and processed ***before*** the NPRD, then the NPRD would overwrite what was learned in the NPZD

NPZD after NPRD Serialization/Consolidation

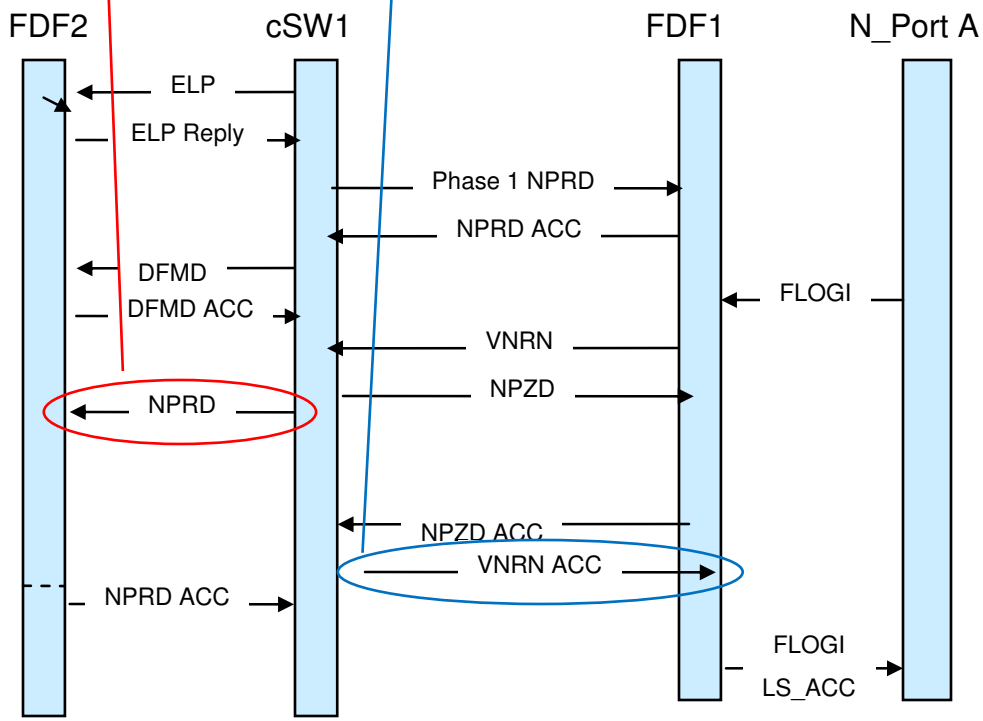


NPRD With Consolidated New NPZD Information

NPRD contains N_Port_IDs in the N_PortIDs Reachability Descriptor that includes pending NPZD info

Q: should this consolidation be mandatory or optional for a cFCF?

With consolidated NPRD, the VNRN ACC can flow before NPRD ACC. If NPRD times out it will be retried including state from any sent NPZDs



Thank you!