

# PCE REDUNDANCY PROBLEM AND SOLUTION SUGGESTIONS

DRAFT-FIZGEER-PCE-REDUNDANCY-EXTENSION

- Marina Fizgeer,
- Ribbon Communication
- Marina.Fizgeer@rbbn.com

# PROBLEM ILLUSTRATION

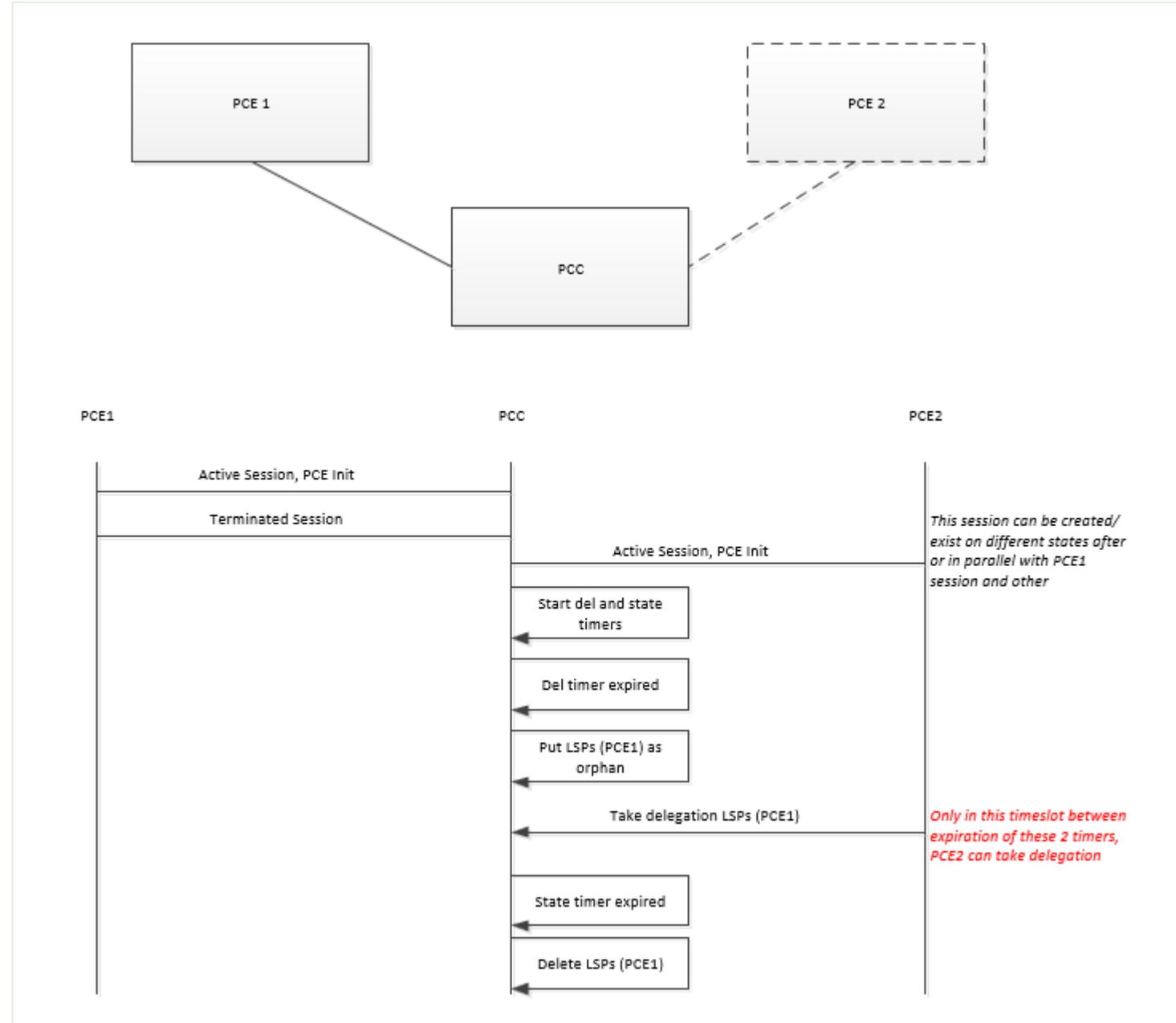
## Network with PCE redundancy

- 2 PCEs are connected to single PCC
  - Both PCEs are stateful and can initiate LSPs (or PCE1 is main, PCE2 is standby)
  - Maybe parallel active sessions or 1 (PCE1) active and 1 inactive (PCE2)
- Session PCE1-PCC is terminated (reason – doesn't matter)
- PCC starts timers
  - Delegation timer – when expired, all LSPs, created by PCE1, will be changed to **orphan**
  - State timer – when expired, all LSPs, created by PCE1 and orphan, will be **deleted**
- Other PCE (PCE2) can require delegation only of orphan LSP
  - It means that PCE2 shall be aware about LSP orphan state and perform delegation request between “delegation timer is expired” and “state timer is not expired”
  - There is no PCEP possibility to inform PCE about timers' states or about LSP state (for example, orphan or not).

### Conclusion – at list one of the below options is needed:

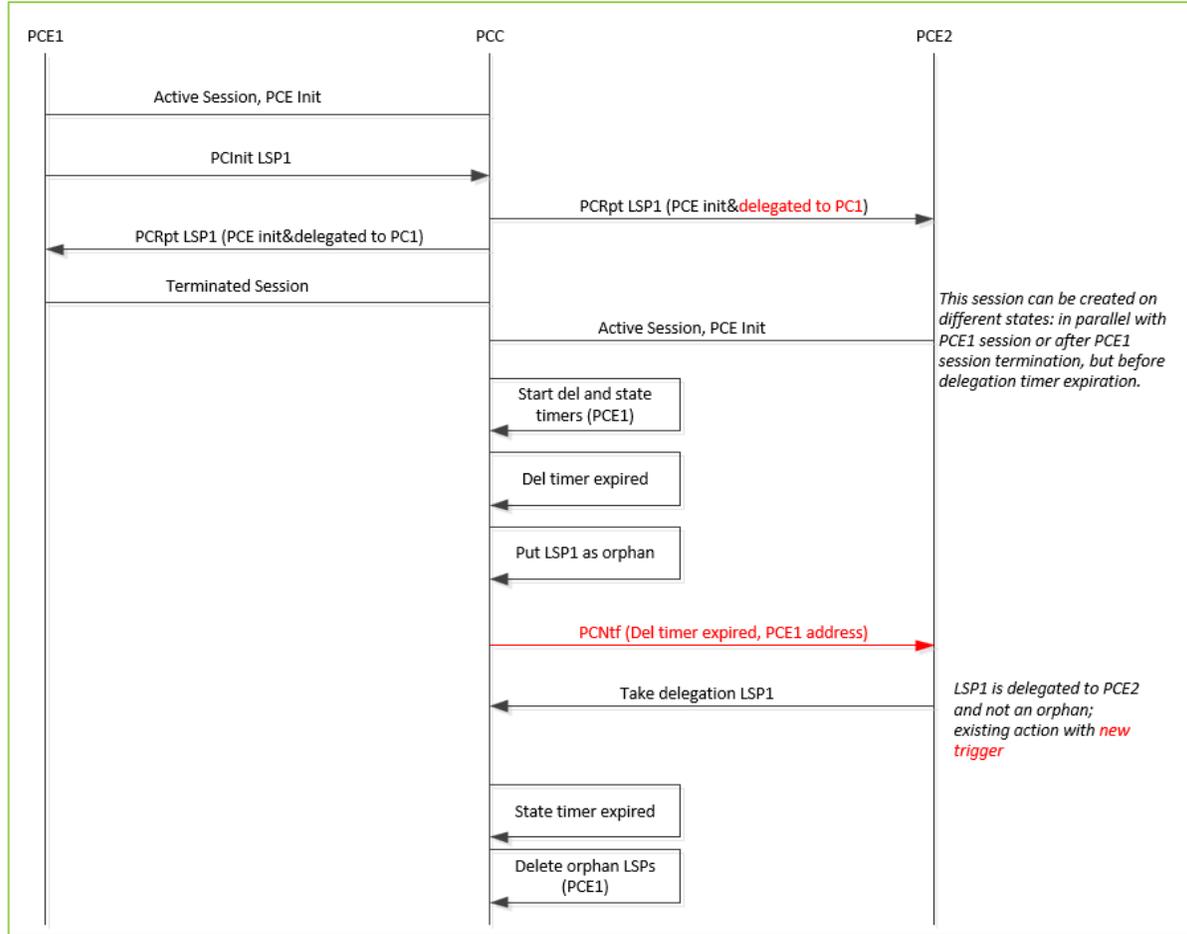
- PCC can somehow redelegate PCE1 initiated LSPs to PCE2
- PCE2 can be aware about timers and/or LSPs state from another session (inactive) and take delegation

Note: PCE2 does not know about the delegation status of LSPs created and/or delegated to other PCEs

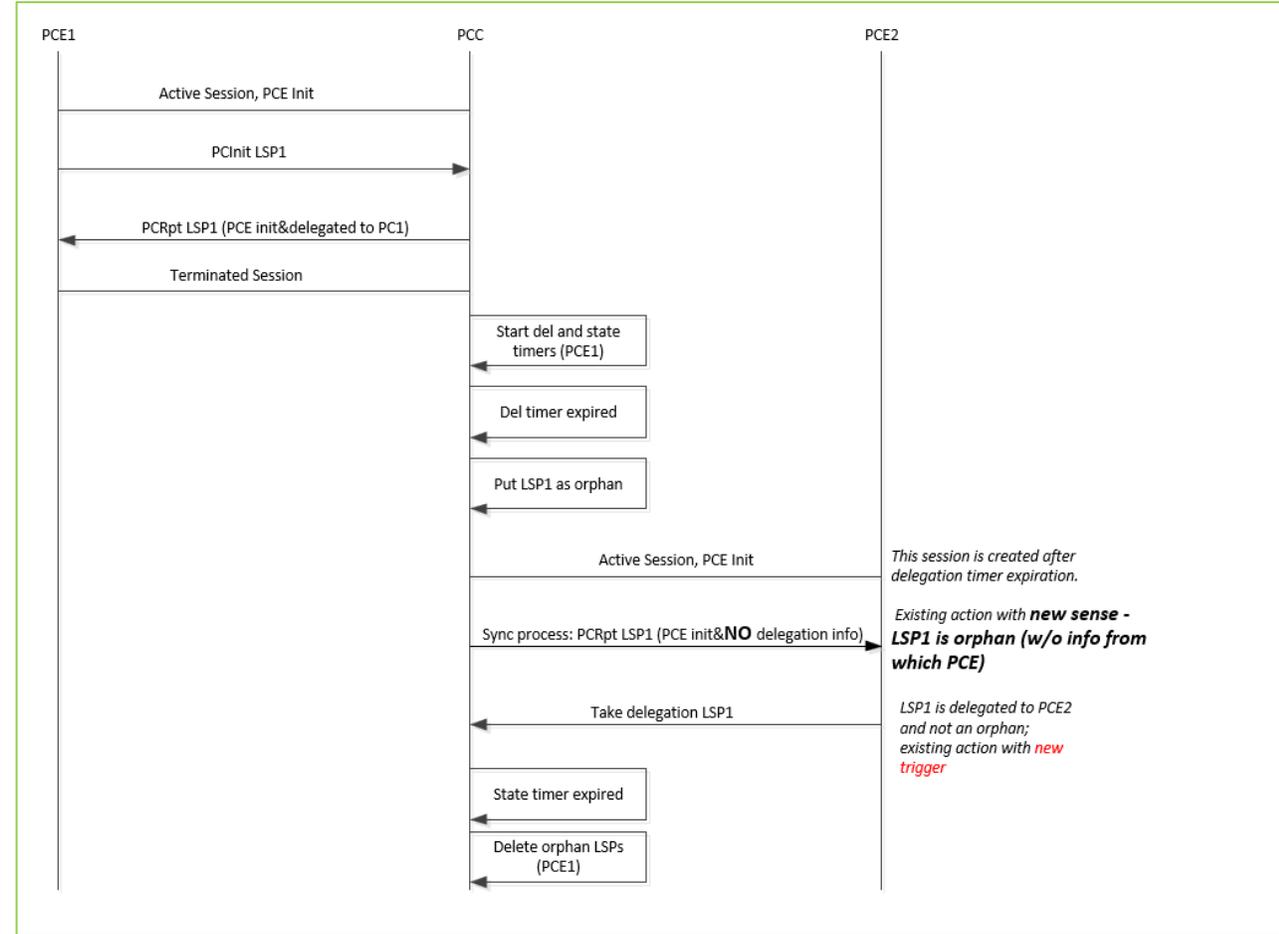


# SOLUTION ILLUSTRATION

## PCE2 is connected before PCE1 del. timer expiration



## PCE2 is connected after PCE1 del. timer expiration



# OUR SUGGESTION

**1. Each PCE with active session shall know about LSP delegation, and if LSP is delegated, to which PCE (IP)**

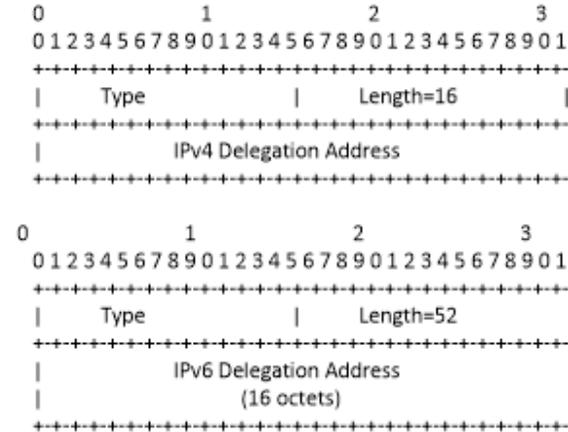
- A new flag is proposed for the STATEFUL-PCE-CAPABILITY TLV

D (DELEGATION-INFO-CAPABILITY): If set, indicates that the PCEP peer supports LSP delegation info.

- Add delegation and PCE address info per LSP in LSP Object:

New optional (MUST for delegated LSPs, MUST NOT for non-delegated to any PCE) TLV in LSP Object with IPv4/IPv6 format

**2. PCC SHALL send this TLV for any delegated LSP to all PCEs with active session (in any PCRpt message)**



**3. New TLV is not needed for orphan LSPs - new session, established after “Delegated timer” is expired (but before “state timer” is expired) for terminated session.**

Here we have 2 options:

- \* During sync process PCE initiated LSPs will be without delegation info, and this can be indication, that LSPs are orphan: LSP (PCE Initiated, No delegation info) = orphan LSP
- \* Add “Last PCE owner IP address to orphan LSP” per orphan LSP – implementation issues, to save last PCE owner for orphan LSPs, or not

Note: LSP delegation info and behavior in this suggestion is relevant for PCE initiated LSP

## OUR SUGGESTION – CONT.

**4. PCC shall send “Delegation to specific PCE is cancelled for all its LSPs” (in other words, LSPs, delegated to this PCE, are changed to orphan):**

- New PCEP message
- PCNtf message with new NT (notification type) and NV (notification value) – preferable:

New NT – Delegation timeout

New NV - TBD

New TLV (sub-TLV) MUST be used for this new NT:

PCE address (was owner) for session with its delegation timer expired

**5. PCC SHALL send this message to all PCEs with active session**

# OUR SUGGESTION – CONT.

## 6. Active PCE – take delegation

- After receiving the PCNtf message with this new NT, the active PCE SHALL/MAY take delegation for all LSPs that were delegated to mentioned in PCNtf PCE (LSP with this IP delegated address) as defined
  - *Here, maybe, new sub-TLV or new flag in PCInit message will help – get all LSPs were delegated to specific IP (like as PLSP ID = 0 in PCInit with remove flag message means deletion for all PCE initiated LSPs)*
    - *New flag in PCInit: D – take delegation, PLSP ID = 0,*
    - *New sub-TLV: Address of “last delegation PCE” (or without, meaning for ALL orphan LSPs)*
- PCC SHALL send list of above LSPs with new delegation address and delegation flag (last delegated PCE) and report redelegated LSPs to all PCEs with active sessions (new delegation PCE)

*Implementation decision: how PCC can know last delegated PCE address for orphan LSP (our implementation – PCC saves this data till LSPs deletion) or perform it for all orphan LSPs*

*Note: if there are more than 2 parallel sessions – first PCE sent get delegation all, will get it ownership, PCC is responsible to lock other PCEs for it*

# SUMMARY

## **This suggestion will work:**

- Multiple parallel sessions in mode active-standby
- Multiple parallel sessions in mode active-active
- Sequential sessions (one session is terminated – another session is established)

# BACKGROUND SLIDE: SOME THOUGHTS ON POSSIBLE SOLUTIONS

1. Add LSP info will help with it, but **can increase the PCEP traffic**, once multiple LSPs are changed their status for orphan (like “sync” processing), via message PCRpt, maybe multiple LSPs in single message (~bulk). Minimal work in PCC (new state value), some new handling in PCE
2. Another option – to add session state, delegation and state timers' info for active PCE about another (down) session (**complicate and not 100% solution as requires full synchronization of timers from different PCEs**)
3. Current option: “other” PCEs shall run the same timers as PCC for session down. **Seems complicated and problematic solution**
4. Possibility for any PCE to take delegation even if this LSP is still delegated to some other PCE. It can be defined as vendor decision how to manage it (explicit user action, internal system decision, what ever), but such action is legal (maybe under some flag or capability, also will help for backward compatibility)
5. Possibility for PCC (for example, under some flag) to redelegate orphan LSPs to some active PCE (from the same PCE group)
6. To use “Speaker entity ID” as PCE initiated LSP owner can solve the problem (if both PCEs have the same entity ID), but defined as optional and cannot be used always with all vendors – **don't help for parallel permanent sessions, as “Speaker entity ID” MUST be unique**
7. Our suggestion (previos slides)

THANK YOU

Marina Fizgeer,

Ribbon Communication

Marina.Fizgeer@rbbn.com