Simple 2-Trunk Intermediate System (S2IS)

Norman Finn
Rev 2

July 3, 2013

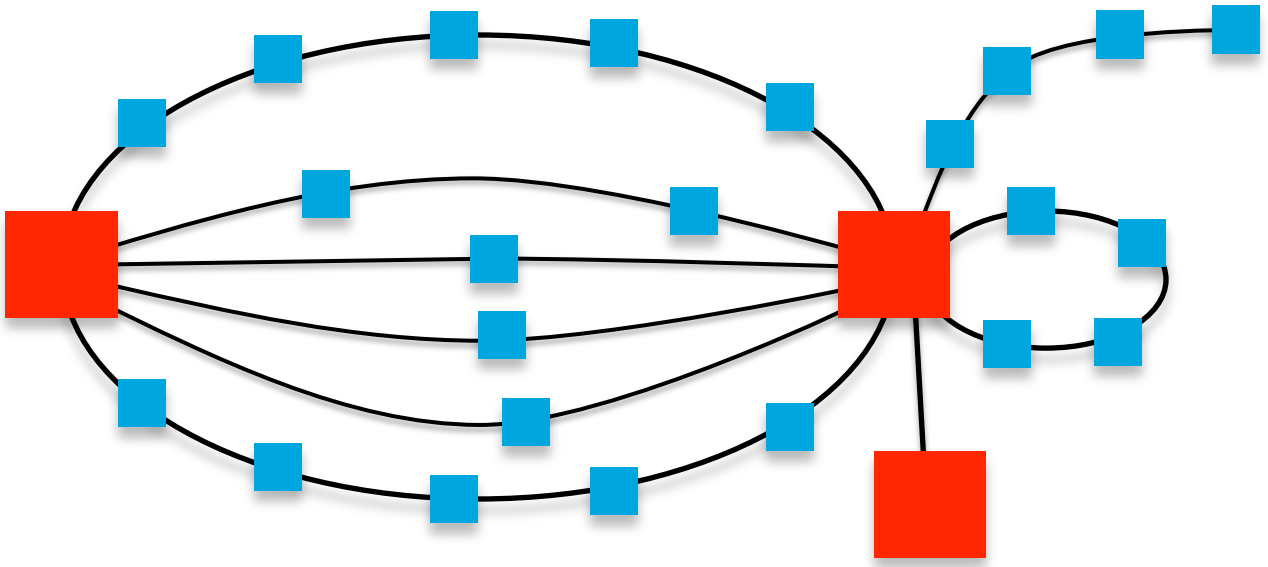
ISO/IEC 10589 ISIS

- ISIS is increasingly important to IEEE 802.1 networks
  - Basis for optimal forwarding for SPBV and SPBM networks.
  - Basis for topology discovery
  - Basis for multiple path determination (P802.1Qca)
  - May be basis for resource allocation (P802.1Qcc, TBD)
- Reaction to ISIS by key players in ODVA and ISA-100:
  - **"I am reluctant to run spanning tree. Don't talk to me about ISIS!"**
- Why this reaction?
  - Every ISIS participant must maintain a complete ISIS LSA database.
  - The LSA database scales with the size of the network, and can be multiple Mbytes in size.
  - The Dijkstra algorithm is perceived as being complex and time consuming.
  - There is no good open source version of ISIS SPB.
  - One must read parts of 4 or 5 standards to implement SPB.

Consider a device with two “trunk” ports

- A Simple 2-port Intermediate System (S2IS, or “stoos”) is connected to:
  - Any number of End Systems
  - 0, 1, or 2 Intermediate Systems, each of which can be either a S2IS or a Full IS.
- **An Intermediate System that has three or more IS neighbors is a Full IS, and not a S2IS.**

Example topology

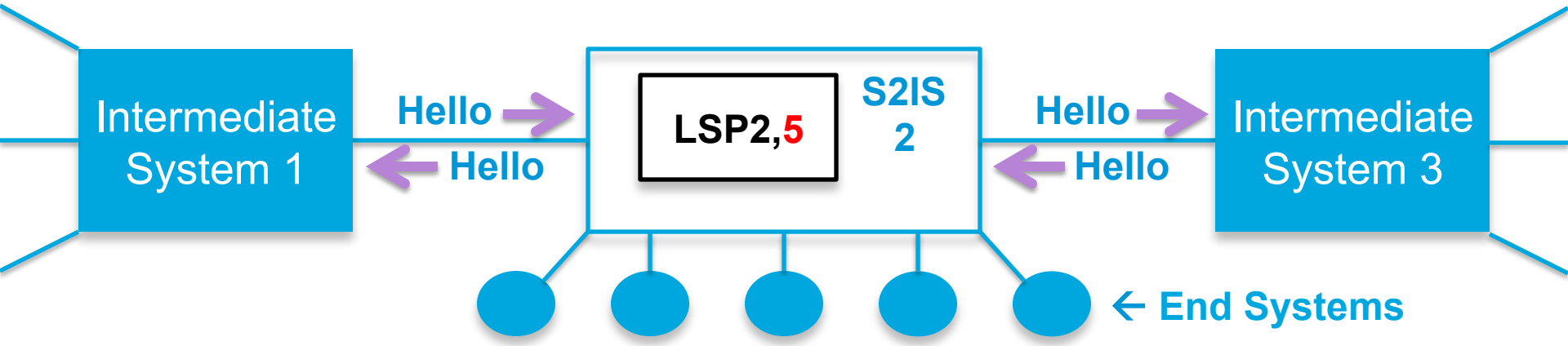


- As in organic chemistry, the hydrogen atoms (End Systems) are not shown.

What if … ?

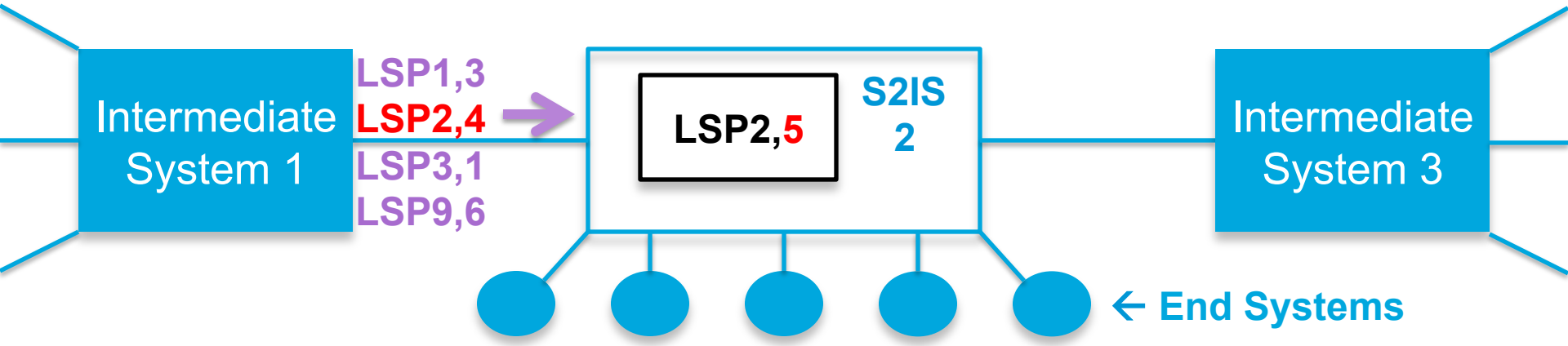
- We will not address, at this time, any question of how a device with 2 or fewer IS neighbors determines whether it should be a Full IS or a S2IS, or what happens if a S2IS-only implementation discovers it has more than 2 IS neighbors.
- We will simply assume that:
  - A full IS can have 0 or more IS neighbors.
  - A S2IS can have 0, 1, or 2 IS neighbors.
  - Either can have any number of end station neighbors.

Link State Database



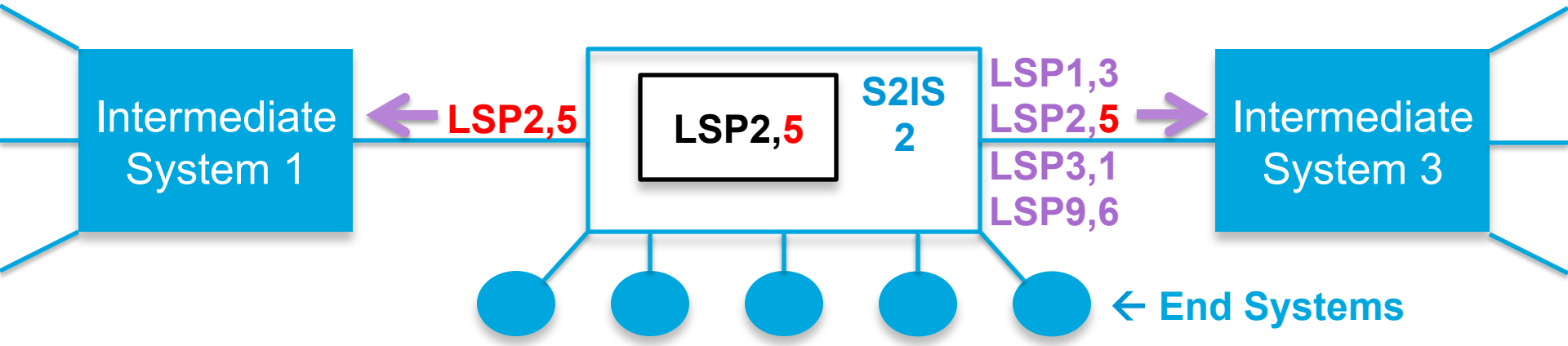
- A S2IS establishes itself as a neighbor of other ISs with the exchange of Hello PDUs in the normal manner for any IS.
- A S2IS keeps a Link State Database exactly according to ISO/IEC 10589, except that it **maintains only its own LSP(s)**. The S2IS **never retains LSPs received from its IS neighbors**.
  - "LSP = Link State PDU" = all or part of an IS's state. An IS can have one or more LSPs, and together, they include at least a list of all of its neighbor ISs, and may include other information of interest to other ISs.

Link State Database



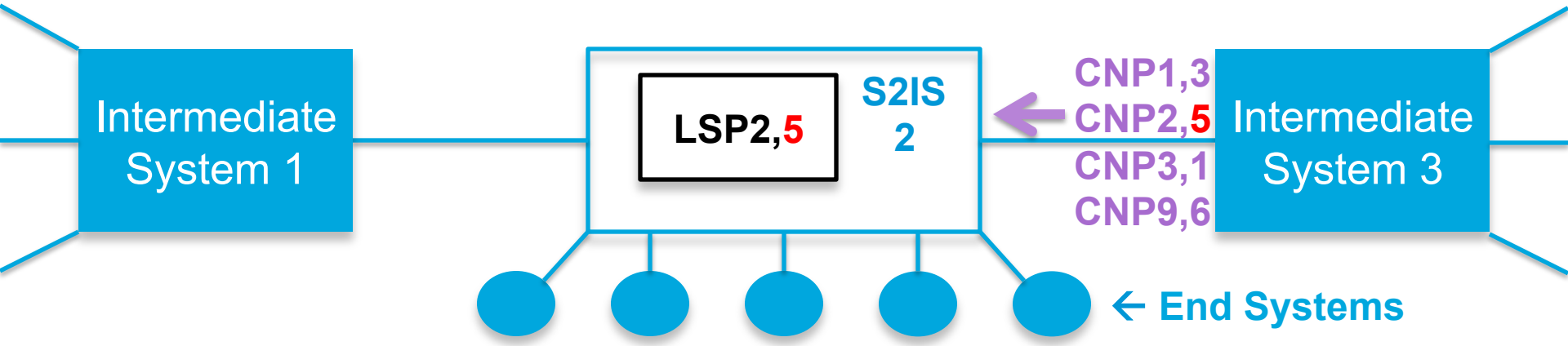
- When a S2IS receives one of its own LSPs, or receives a SNP (Sequence Numbers PDU) for one of its own LSPs, it handles that LSP or SNP in the normal manner prescribed by ISO/IEC 10589.
- When a S2IS receives one or more LSPs or SNPs that are not its own, it queues transmission on its other IS port a frame carrying those LSPs or SNPs. That is, non-self LSPs and SNPs pass transparently through the S2IS as a normal transmission by the S2IS. (I.e., the source address changes to that of the S2IS.)

Link State Database



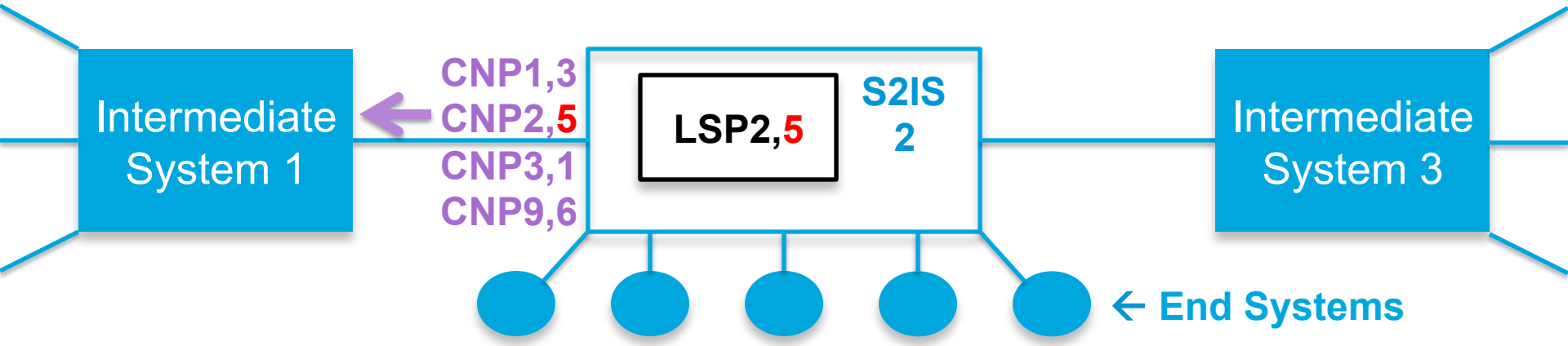
- When a S2IS receives one of its own LSPs, or receives a SNP (Sequence Numbers PDU) for one of its own LSPs, it handles that LSP or SNP in the normal manner prescribed by ISO/IEC 10589.
- When a S2IS receives one or more LSPs or SNPs that are not its own, it queues transmission on its other IS port a frame carrying those LSPs or SNPs. That is, non-self LSPs and SNPs pass transparently through the S2IS as a normal transmission by the S2IS. (I.e., the source address changes to that of the S2IS.)

Link State Database



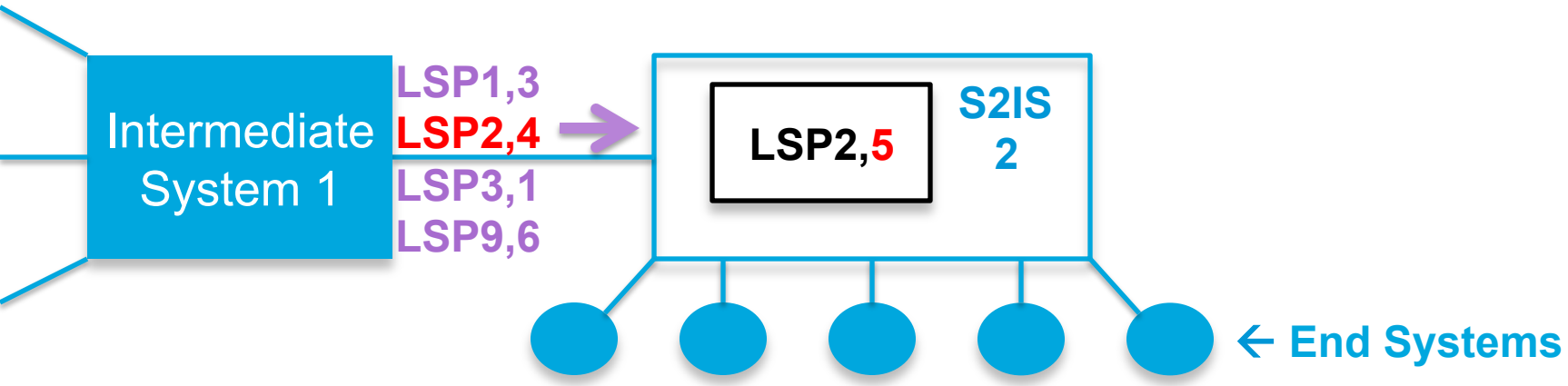
- When a S2IS receives one of its own LSPs, or receives a SNP (Sequence Numbers PDU) for one of its own LSPs, it handles that LSP or SNP in the normal manner prescribed by ISO/IEC 10589.
- When a S2IS receives one or more LSPs or SNPs that are not its own, it queues transmission on its other IS port a frame carrying those LSPs or SNPs. That is, non-self LSPs and SNPs pass transparently through the S2IS as a normal transmission by the S2IS. (I.e., the source address changes to that of the S2IS.)

Link State Database



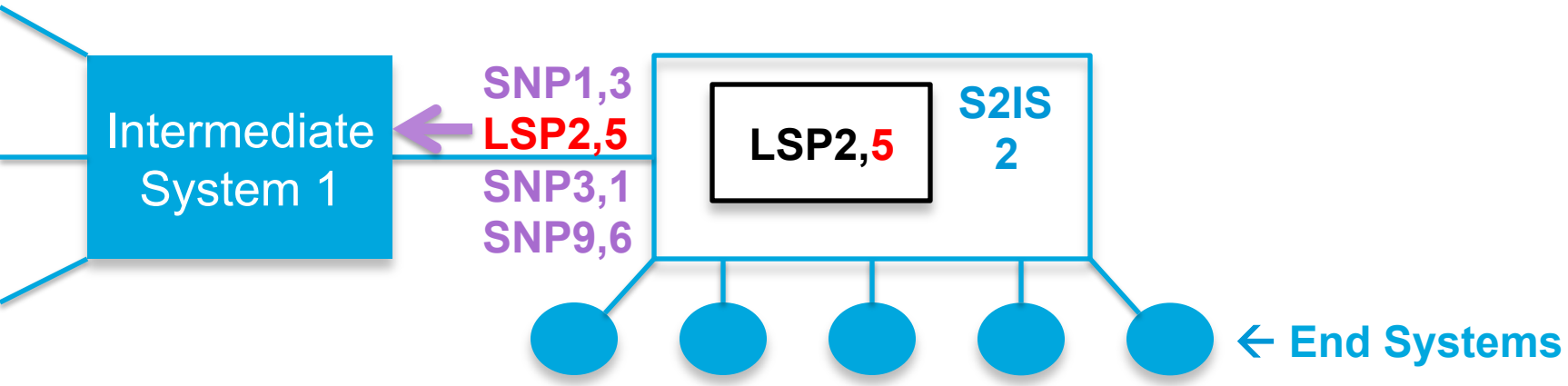
- When a S2IS receives one of its own LSPs, or receives a SNP (Sequence Numbers PDU) for one of its own LSPs, it handles that LSP or SNP in the normal manner prescribed by ISO/IEC 10589.
- When a S2IS receives one or more LSPs or SNPs that are not its own, it queues transmission on its other IS port a frame carrying those LSPs or SNPs. That is, non-self LSPs and SNPs pass transparently through the S2IS as a normal transmission by the S2IS. (I.e., the source address changes to that of the S2IS.)

Link State Database



- When a S2IS receives one of its own LSPs, or receives a SNP (Sequence Numbers PDU) for one of its own LSPs, it handles that LSP or SNP in the normal manner prescribed by ISO/IEC 10589.
- If an S2IS has no second IS neighbor, it replies to LSPs not its own with SNPs, just like a Full IS, even though it does not retain any LSPs except its own in its Link State Database.

Link State Database



- When a S2IS receives one of its own LSPs, or receives a SNP (Sequence Numbers PDU) for one of its own LSPs, it handles that LSP or SNP in the normal manner prescribed by ISO/IEC 10589.
- If an S2IS has no second IS neighbor, it replies to LSPs not its own with SNPs, just like a Full IS, even though it does not retain any LSPs except its own in its Link State Database.

Obvious things to do

There are more details than are shown here:

- The start-up (or state change) condition for a S2IS is very, very simple. It transmits only one LSP, in order to keep from having to implement a timer. Convergence slows if PDUs are lost, but periodic CSNPs by the Full ISs recover.
- “Healing” a link in the middle of a chain connecting Full ISs is not discovered until the next periodic CSNP. Complexity can be added to the S2IS to avoid this. This can be discussed, later.
- If some number of S2ISs are joined in a ring without a Full ISIS, the LSPs do not circulate forever; each S2IS stops its own LSPs as they return.
  - Yes, a faulty S2IS could allow an LSP to circulate forever. Any faulty IS could chatter incessantly, for that matter.

Given that …

- Every Full IS has a normal Link State Database that includes all of the S2ISs as if they were also Full ISs.
- A Full IS behaves perfectly normally, with no additional behavior required (except that the recommended practice of sending periodic Complete SNPs is required, instead of recommended).
- But, since S2ISs do not have the complete Link State Database, they cannot:
  - Determine whether a given trunk port is or is not blocked for a given SPB spanning tree, and thus whether to forward or filter a frame received from a trunk port.
  - Run a Dijkstra calculation to determine on which trunk port to send a packet (frame) received from one of its attached End Stations.

So what do we do in the data plane?

- Lots of possibilities to think about, and to talk about.
- The Full ISs could do all of the spanning tree blocking, and forwarding becomes less than optimal.
- A Full IS could “adopt” one or more S2ISs and tell them what VLAN IDs to block on what port(s), or even download a full Filtering Database.
- S2ISs could send all unknowns out both trunk ports, learn from received traffic, and some or all of the learned information when told to do so by a Full IS.
- The S2IS could run SRP, MVRP, etc., or it could depend on the Full IS for assistance. (If this sounds like SDN …)
- The S2ISs may not do SPB at all – they may do ISO/IEC 62439 HSRP or ITU-T G.8032 rings.

Summary

- Adding just a little bit of code to a ring- or chain-capable device allows us to use ISIS to observe the complete network topology.
- This makes it much easier to retrofit existing ring- or chain-based networks with ISIS and TSN protocols.
- This makes it possible to build SPB networks that have most of the benefits of SPB without requiring the expense, both in unit cost and in ongoing management cost, of running full-up SPB in all bridges.
- This may be of some benefit to L3 ISIS – that is not clear.
- IEEE 802.1 may be the wrong place to define the S2IS. We should talk to the IETF and decide where (and of course, whether) the work should be done.

Thank you.