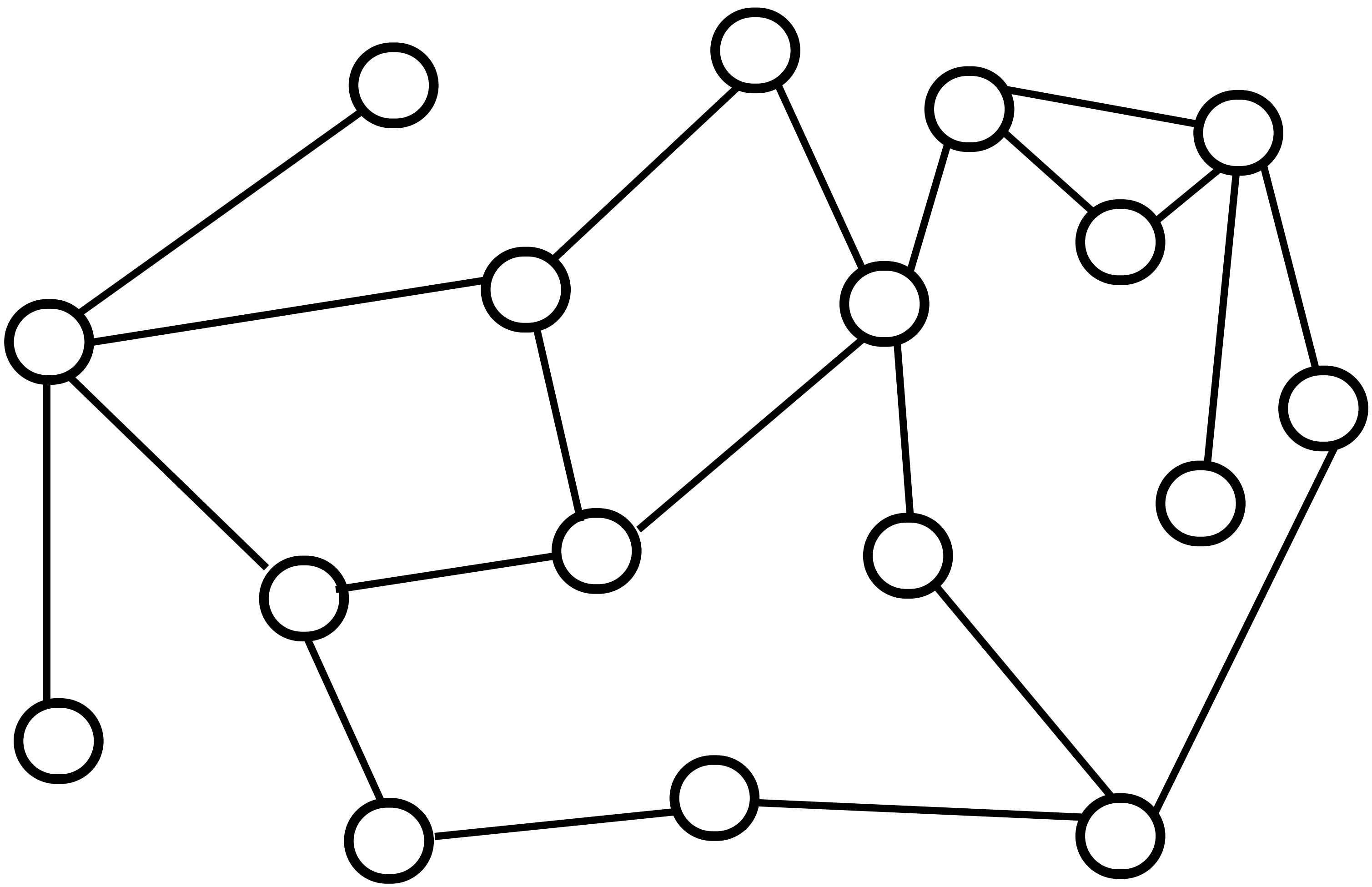
Server Cache Synchronization Protocol (SCSP)

Presented By: James V. Luciani

Ascom Nexion

Goals

- **Don't re-invent the wheel**
  - Use a well known and tested database synchronization technology which lends itself to code reuse such as that found in link state protocols
- **Allow for synchronization of servers in an *arbitrary topology* of servers**
- **Use a server synchronization approach which is easily adapted to multiple protocols (e.g., MARS, ATMARP, etc.)**
- **Impose minimal burden on the clients**
  - Old clients must work in the new environment and new clients are not precluded from working in the old environment
  - A client should be permitted but not required to use multiple servers
- **Impose minimal burden on the server**
- **Allow for a high degree of scalability**
- **Allow for autoconfiguration**
- **Allow for authentication using native protocol techniques**
  - E.G., use existing NHRP authentication extension (clear text or MD5)



SCSP Allows For An Arbitrary Server Topology

A Frame Of Reference

Server Group

- The SCSP synchronizes caches of a set of server entities which are bound to a SG through some commonality (e.g., membership in a LIS).

Local Server (LS)

- All statements about SCSP are made from the perspective of an LS.

Directly Connected Server (DCS)

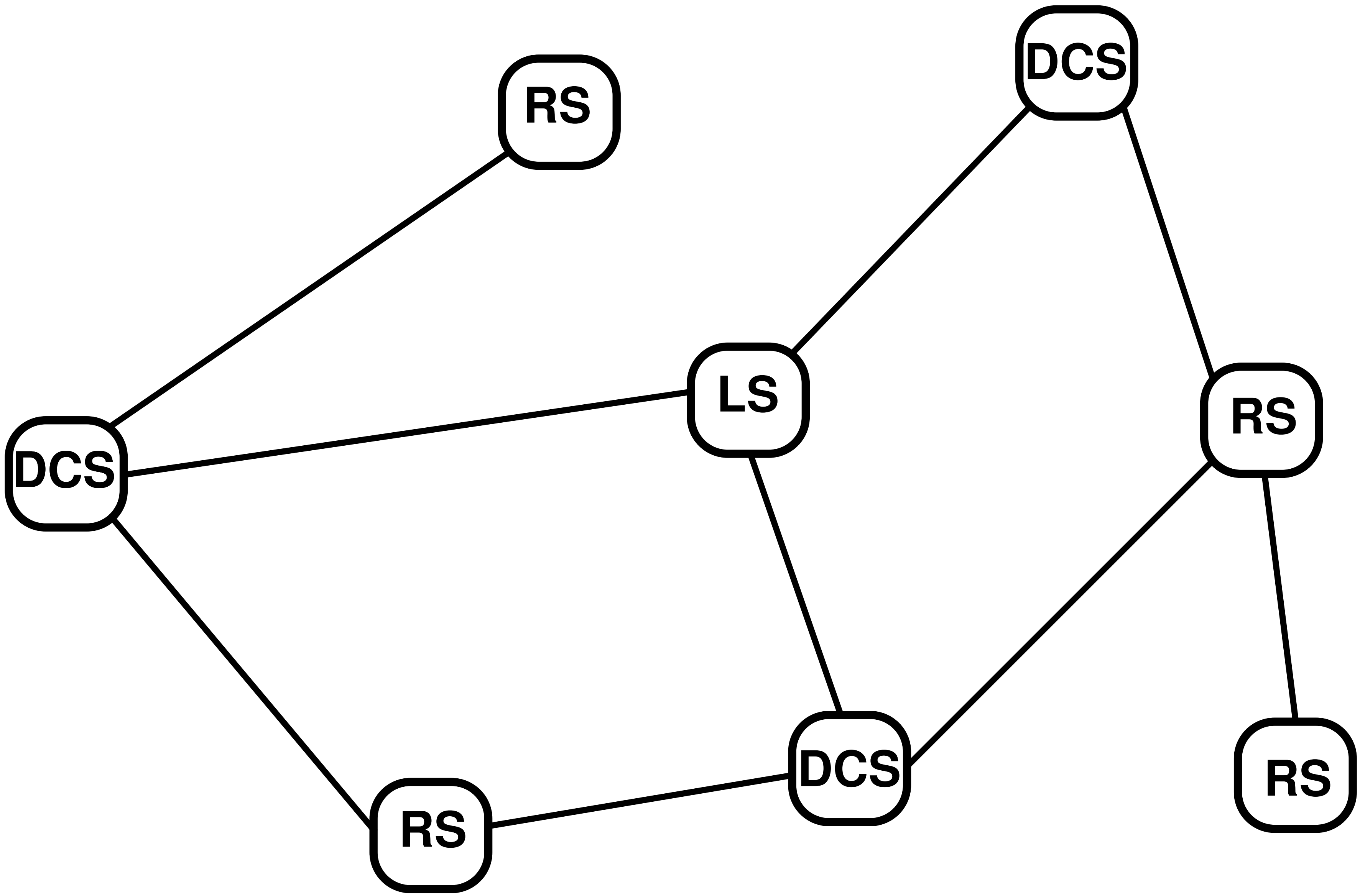
- The DCS is a server which is directly connected to the LS. For example, there exists a VC between the LS and DCS. A DCS is always one hop away from an LS.

Remote Server (RS)

- Any server in the SG that is not an LS or DCS.

The terms LS, DCS, and RS are terms created to give a frame of reference when talking about SCSP.

- Every server is an LS with respect to the SCSP code running on that server.
- Every server is a DCS with respect to the frame of reference of every other server to which it is directly connected.



An Example Of A Server Group As Seen By An LS

3 Basic Message Types

Hello

- Used to ascertain whether a DCS is operational and whether the connection between the LS and DCS is bidirectional, unidirectional, or non-functional

Client State Update (CSU)

- Used to update the state of cache entries in servers for a given SG
- CSU Reply acknowledges a CSU by merely returning the CSU message received and marking the message as being an ack/reply.

Cache Alignment (CA)

- Allows an LS to synchronize its cache with that of the cache of its DCSs
- Client State Update Solicit (CSUS) messages have the same format and they are used to request Client State Updates

SCSP Protocol Basics

- **An LS maintains a Hello Finite State Machine (HFSM) and a Cache Alignment Finite State Machine (CAFSM) for each DCS**
  - An HFSM keeps track of the connectivity state between an LS and the corresponding DCS
  - A CAFSM keeps track of the state of the alignment of the caches in both LS and the corresponding DCS
- **When an LS decides to communicate via SCSP to one of its DCSs, the HFSM and CAFSM are placed in the DOWN state.**
- **Every LS must be configured with knowledge of its DCSs since there is no broadcast media in an NBMA**
  - the method of this configuration is beyond the scope if this work but might include an autoconfiguration service and/or manual configuration

■ HFSM - Keeping Track Of Connectivity

- ❑ The LS maintains an HFSM for each of its DCSs. The HFSM starts in the DOWN state. When in the DOWN state the HFSM may only transition to the WAITING state.
- ❑ After the LS acquires NBMA level connectivity to that DCS, the corresponding HFSM transitions to the WAITING state at which point the LS starts sending and/or receiving Hello messages to/from that DCS.
- ❑ If the LS receives a Hello message from the DCS with a zero in the Receiver ID field of the Hello message then the HFSM transitions to the UNIDIRECTIONAL CONNECTION state.
- ❑ If the LS receives a Hello message from the DCS with its LSID included in the Receiver ID field of the Hello message then the HFSM transitions to the BIDIRECTIONAL CONNECTION state.
- ❑ Abnormal events such as a malformed packet cause the HFSM to transition to the WAITING state.
- ❑ Loss of NBMA level connectivity cause the HFSM to transition to the DOWN state.

CAFSM - Keeping Track Of Cache Alignment

- **The LS maintains a CAFSM for each of its DCSs which starts in the DOWN state.**
- **When the HFSM associated with a particular DCS transitions to the BIDIRECTIONAL CONNECTION state, the CAFSM transitions to the MASTER/SLAVE NEGOTIATION STATE.**
- **When a master is chosen, the CAFSM transitions to the CACHE SUMMARIZE state**
  - The LS and DCS exchange Client State Advertisement Summary (CSAS) records which are included in Cache Alignment (CA) messages. A CSAS record contains a summary of a server's cache entry.
  - The LS keeps a list of CSAS records which are "more up to date" in the DCS's cache then they are in the LS's cache. This list is called a CRL.

CAFSM - Continued

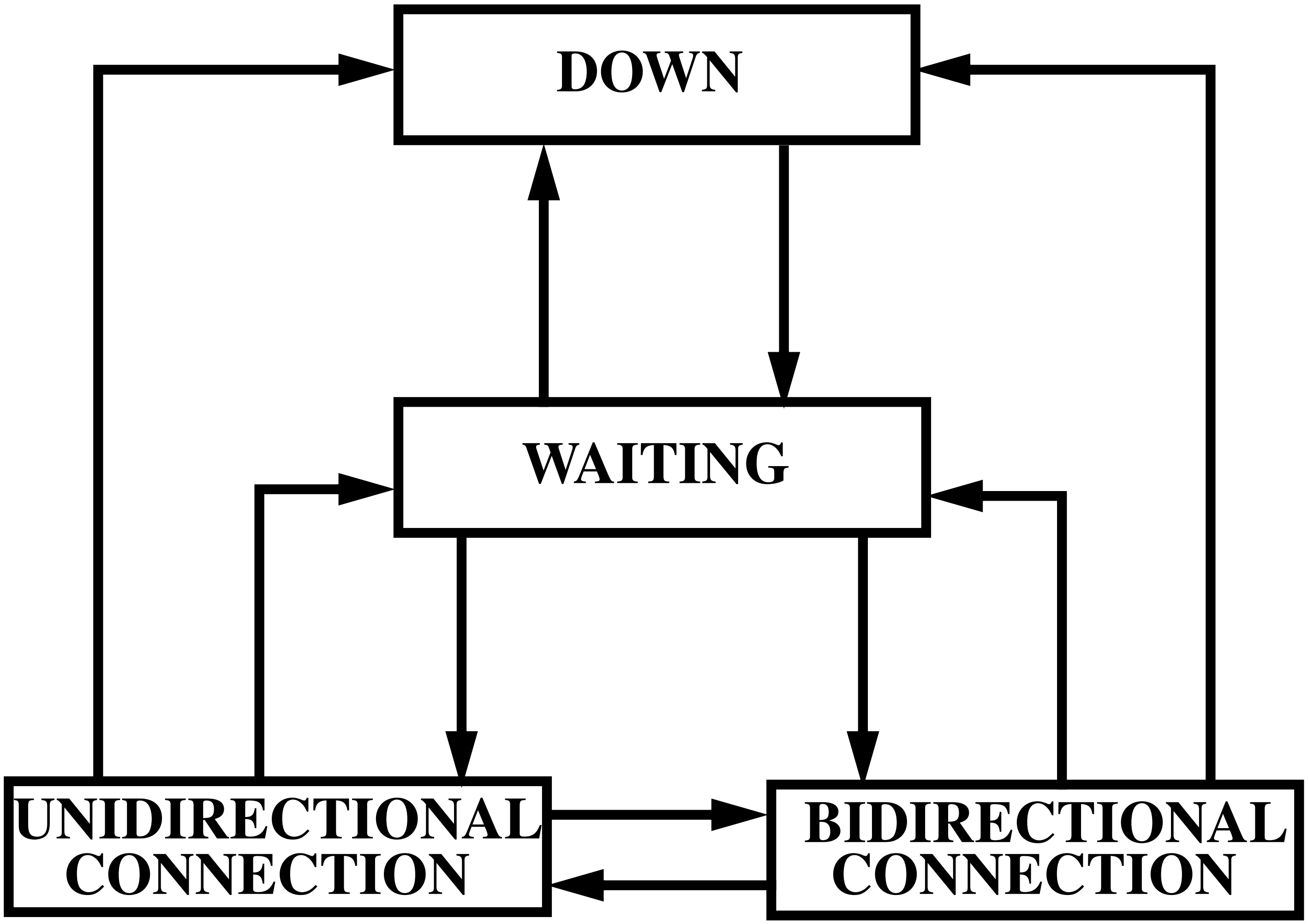
- **When the cache summarization process is complete, the CAFSM transitions into the CACHE UPDATE state.**
  - At this point, the LS sends one or more Client State Update Solicit (CSUS) messages to the DCS which contain the CSAS records in the CRL. The DCS responds by returning Client State Update (CSU) messages to the LS.
  - CSU messages contain Client State Advertisement (CSA) records and CSA records contain the client state. When the DCS responds to the LS's CSUS messages, it places a CSA records in a CSU message for each CSAS in the CRL.
  - The "more up to date" information is then copied into the LS's cache.
- **When the update is complete, the CAFSM transition into the ALIGNED state**

Aligned State

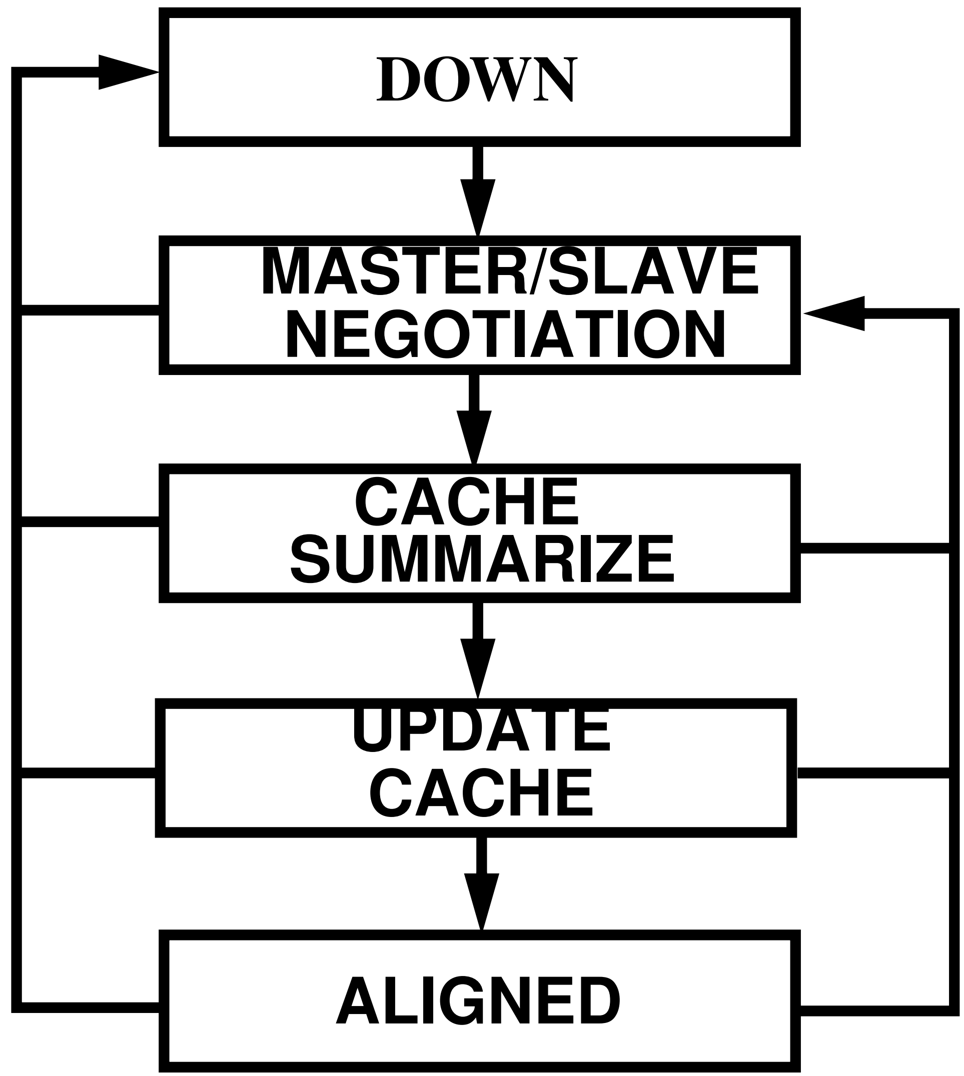
- **When the CAFSM is in the ALIGNED state the communication between the LS and the DCS associated with that CAFSM is in the "*steady state*" condition**
- **In the steady state, only CSU messages are exchanged between LS and that DCS. There are 3 basic reasons to exchange CSU messages in this case:**
  - LS receives a request to add an entry to its cache (e.g., NHRP Registration Request)
  - LS receives a request to remove an entry from its cache (e.g., NHRP Purge Request)
  - LS has a cache entry which has timed out, has been refreshed, or has been administratively modified
- **It is recommended that servers only exchange state for clients which are explicitly registered within the server group**
  - That is, servers should only synchronize information which is local to their server group and not on a global basis. For example in NHRP, only state for clients in a LIS should be synchronized within the LIS otherwise update rates would be quite large regardless of the synchronization technology.

Issues

- **Should Cache Alignment messages contain Client State Advertisements (CSA) records or Client State Advertisement Summary (CSAS) records**
  - Add bit to hello to negotiate for CSA rather than CSAS
  - if CSA then skip CSUS exchange stage but potentially much more traffic
  - if CSAS then only get what you need but potentially takes longer to align caches
- **Counter Wrap**
  - To lollipop or not to lollipop... that is the question.
- **Add bit to Hello to allow pt-mpt**
  - Receiver ID <> 0 or LSID is acceptable when this is negotiated



Hello Finite State Machine (HFSM)



Cache Alignment Finite State Machine (CAFSM)