

Forces Protocol

Jamal Hadi Salim

(for the ForCES Protocol Team)

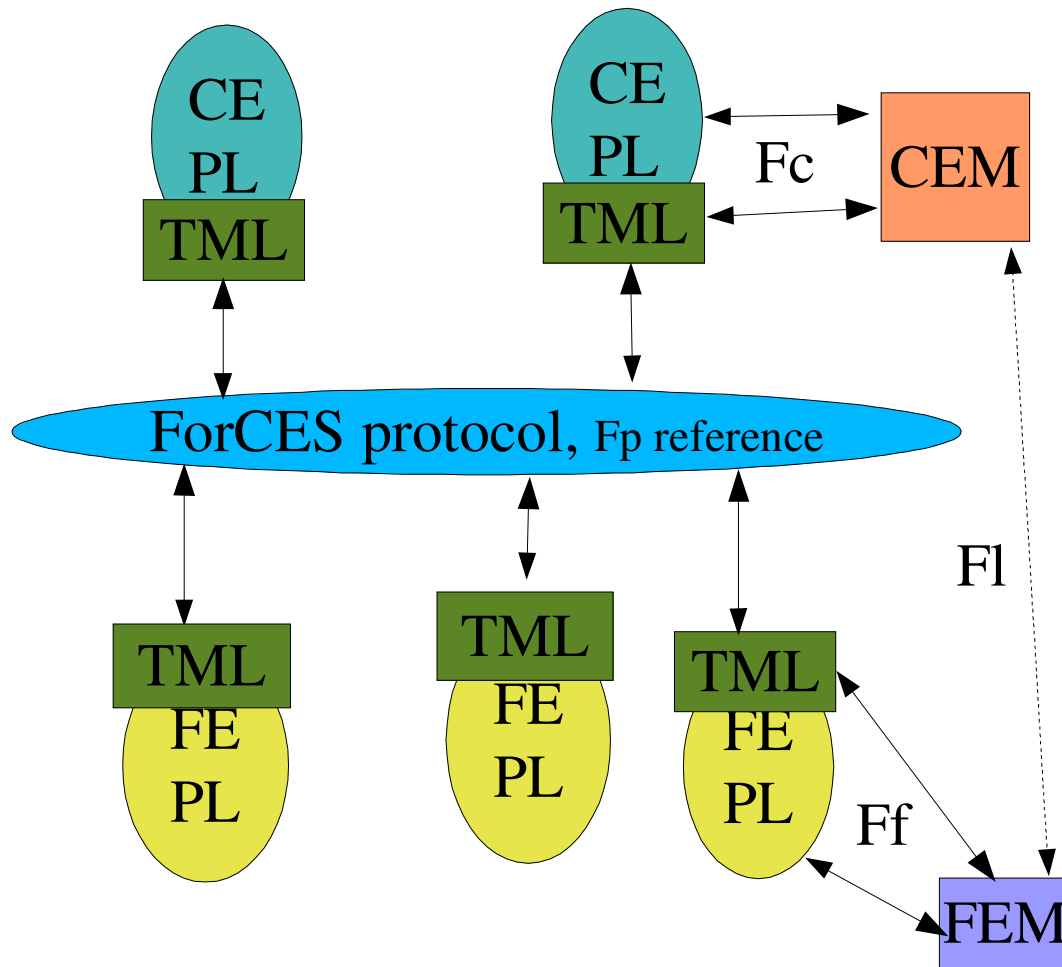
IETF61, Washington, DC

November 2004

Team Members (alphabetical order)

- Ligang Dong, Zhejiang Gongshang University
- Avri Doria, ETRI
- Ram Gopal, Nokia
- Robert Haas, IBM
- Jamal Hadi Salim, ZNYX Networks
- Hormuzd M Khosravi, Intel
- Weiming Wang, Zhejiang Gongshang University

ForCES overview



Outside of ForCES charter:

- CE-CE communication
- FE-FE communication
- CEM-FEM
- CEM
- FEM

TML

- Resolves underlying Transport issues
 - Reliability
 - Security
 - Congestion control
 - Address Mapping
 - Timeliness
 - Prioritization
 - Encapsulation
 - High Availability facilitator

TML

- Several proposals made to date
 - TIPC: L2/L1
 - Draft-maloy-tipc-01
 - Unicast, multicast, broadcast
 - draft-audu-forces-iptml-00
 - IP level
 - Draft-khosravi-forces-tcptml-00
 - IP Level

PL-PL Logical Interaction



- PL Termination always at LFB
 - Simplifies grammar and architecture
 - There are a few exceptions such as a heartbeat and association messages
 - Still needs to be discussed

Core LFBs

- Needed for termination of messages addressed to the FE or CE
- Three LFBs at the moment
 - FE Object LFB
 - Maintains attributes relevant to the FE
 - FE Protocol Object LFB
 - Maintains attributes relevant to the ForCES protocol
 - CE Object LFB
 - Maintains attributes relevant to the CE

Core LFBs: FE Protocol Object LFB

- Assigned class ID 1 and instance Id 1
- Responsible for:
 - Protocol events that can be subscribed to
 - Heartbeats, TML events, etc
 - Protocol capabilities
 - Version, TML capability
 - Protocol attributes
 - Association timer, heartbeat interval, Primary CE, alternate
 - Alternate CEs, failover policy, restart policy, etc

Core LFBs: FE Object LFB

- Assigned class ID 2 and instance Id 1
- Responsible for
 - FE Events management
 - FE Status changes, DOS alerts, Capability changes
 - FE attributes
 - FE status, Instantiated LFBs and topologies, FE Model etc
 - FE Capabilities
 - Supported LFBs by FE, their occurrence limits, etc

Core LFBs: CE Object LFB

- Assigned class ID 3 and instance ID 1
- Responsible for
 - CE Events management
 - CE Status changes, Capability changes
- This LFB is still under discussion
- Maybe even a CE Protocol LFB maybe needed to reciprocate FE protocol Object

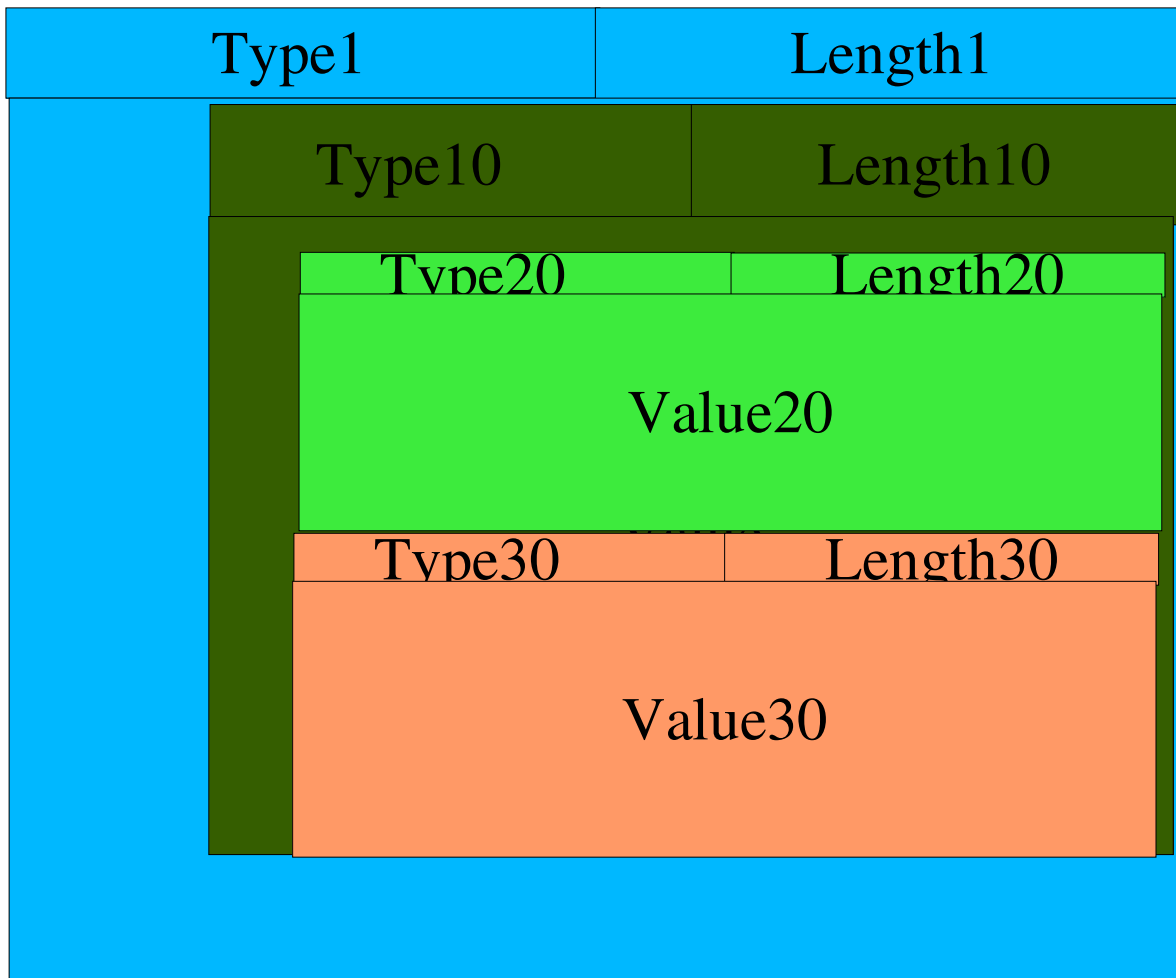
Protocol Grammar

- PL level PDU := PLHdr<*LFBSelect*>+
- LFBSelect := LFBClassID LFBInstance
<*OPER*>+
- OPER := <OPERATION [*path-data*]*>+

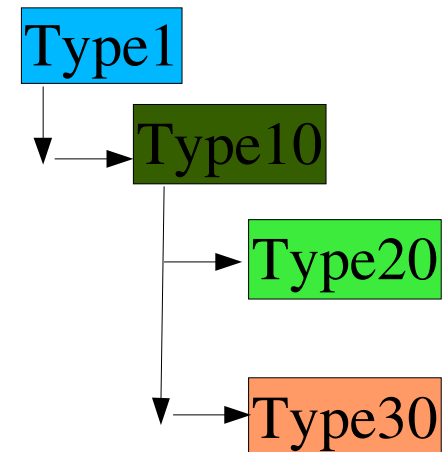
Protocol Grammar

- PLHdr defines message type and CE/FE Ids, etc
- LFBClassID is a 32 bit unique identifier given at LFB definition time
- LFBInstance is a 32 bit ID for an instance of LFB class creates at runtime
- Operation is one of *ADD*, *DEL*, *GET*, *ADVERTISE*, *CANCEL*, etc
- Path-data is what the operation is applied on
 - Could be attribute, capability, event

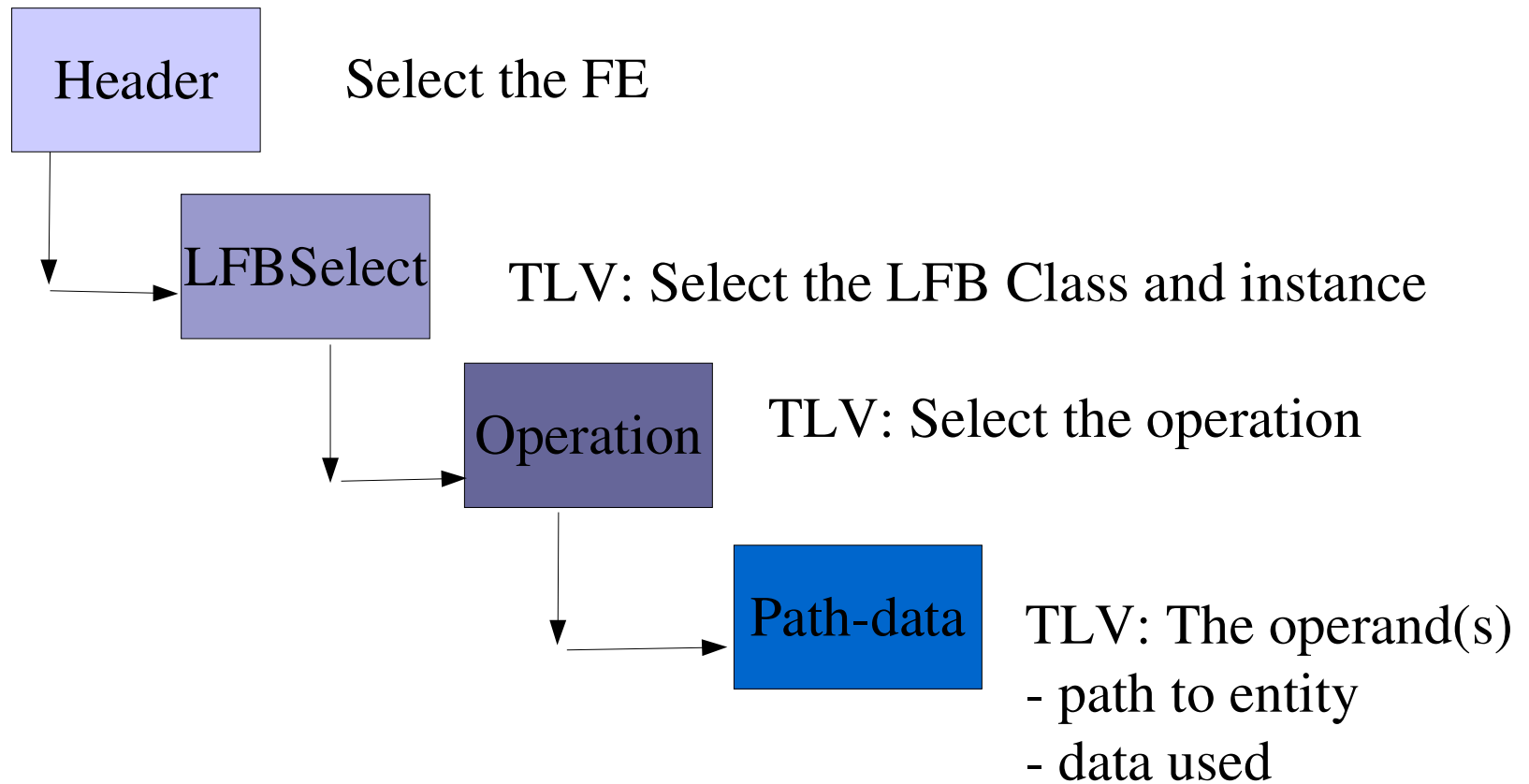
Nested TLVs



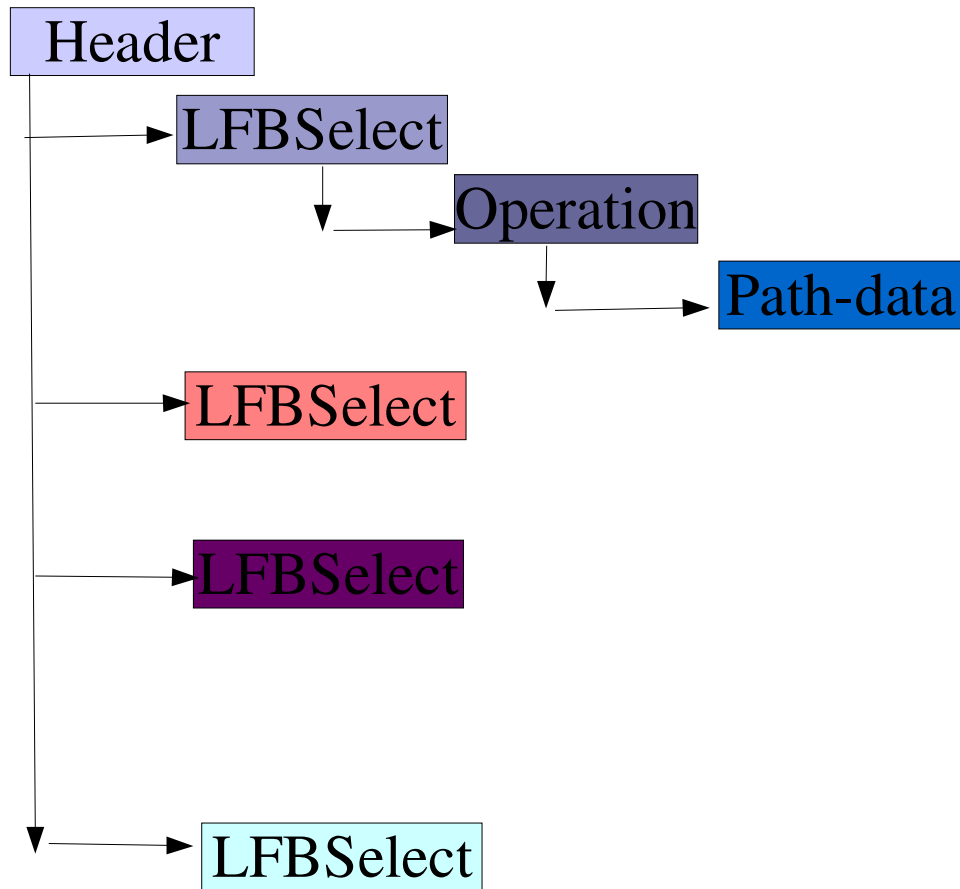
- Very extensible
 - Fits very nicely in BNF description of PL protocol
 - Maps nicely to xml definitions/layering



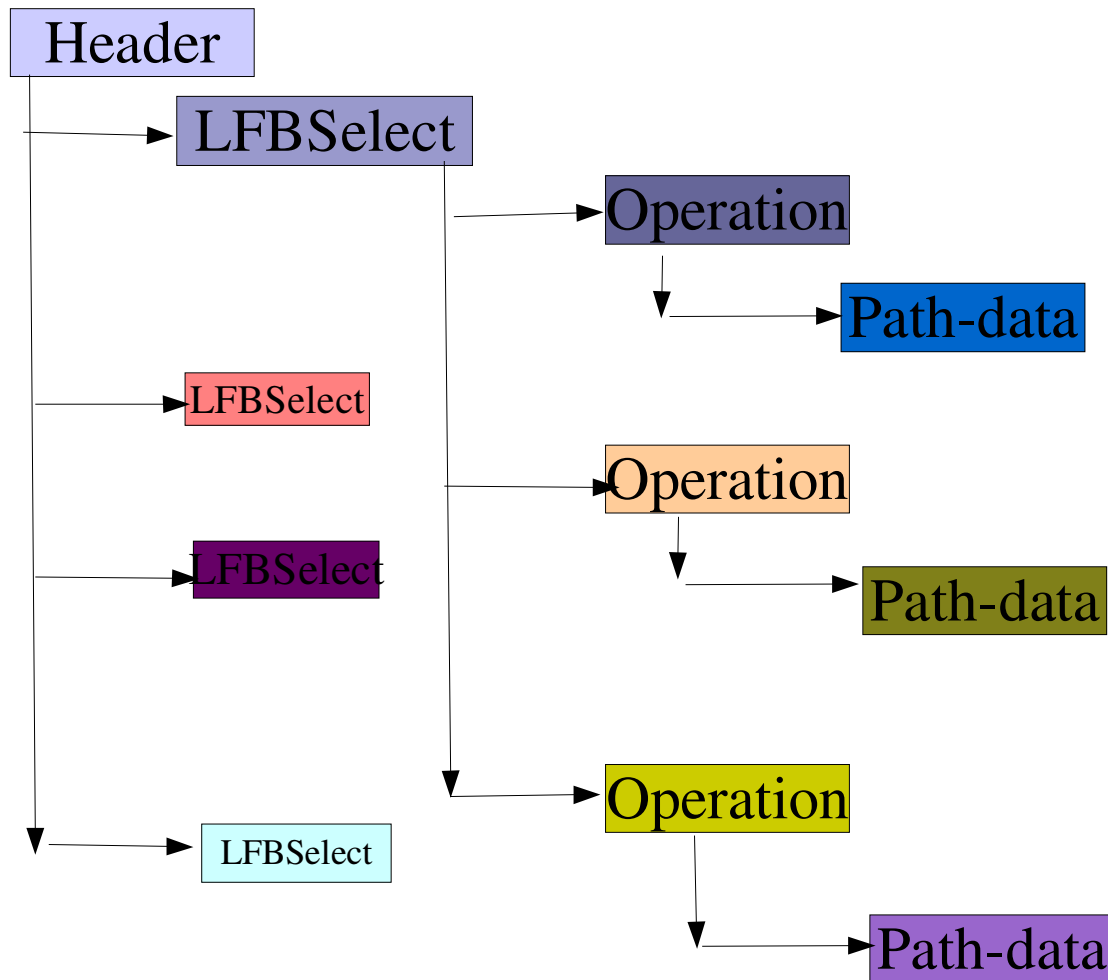
PL Message Layout



Batching: Multiple LFB selections



Batching: Multiple operations



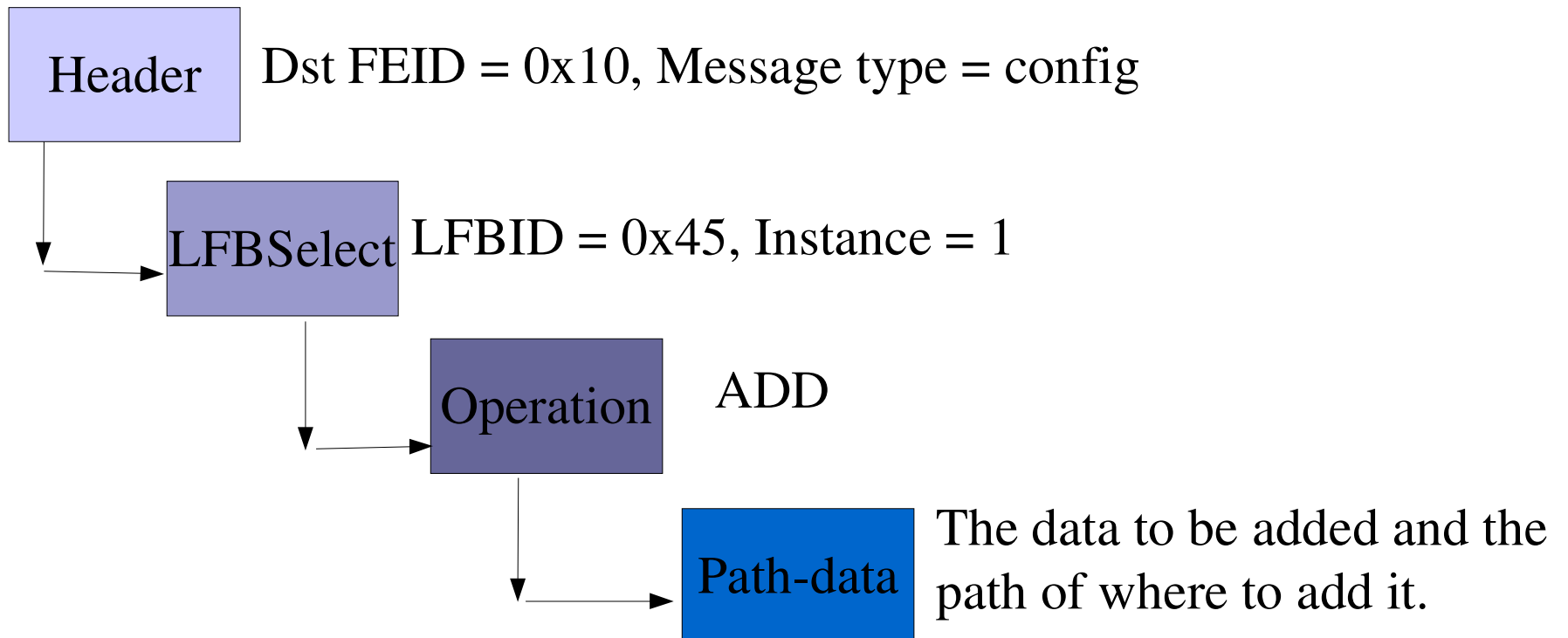
PL Message Types

- Association
 - To join and leave NE (by FE)
- Configuration
 - To configure attributes, capabilities and events in an LFB instance
- Query
 - To query for configured attributes, capabilities and events

PL Message Types

- Event Notification
 - Used to notify about events to registered users
- Packet redirection
 - Packet redirects from FE to registered LFBs in CE
- Heartbeat
 - Heartbeat between CE and FE

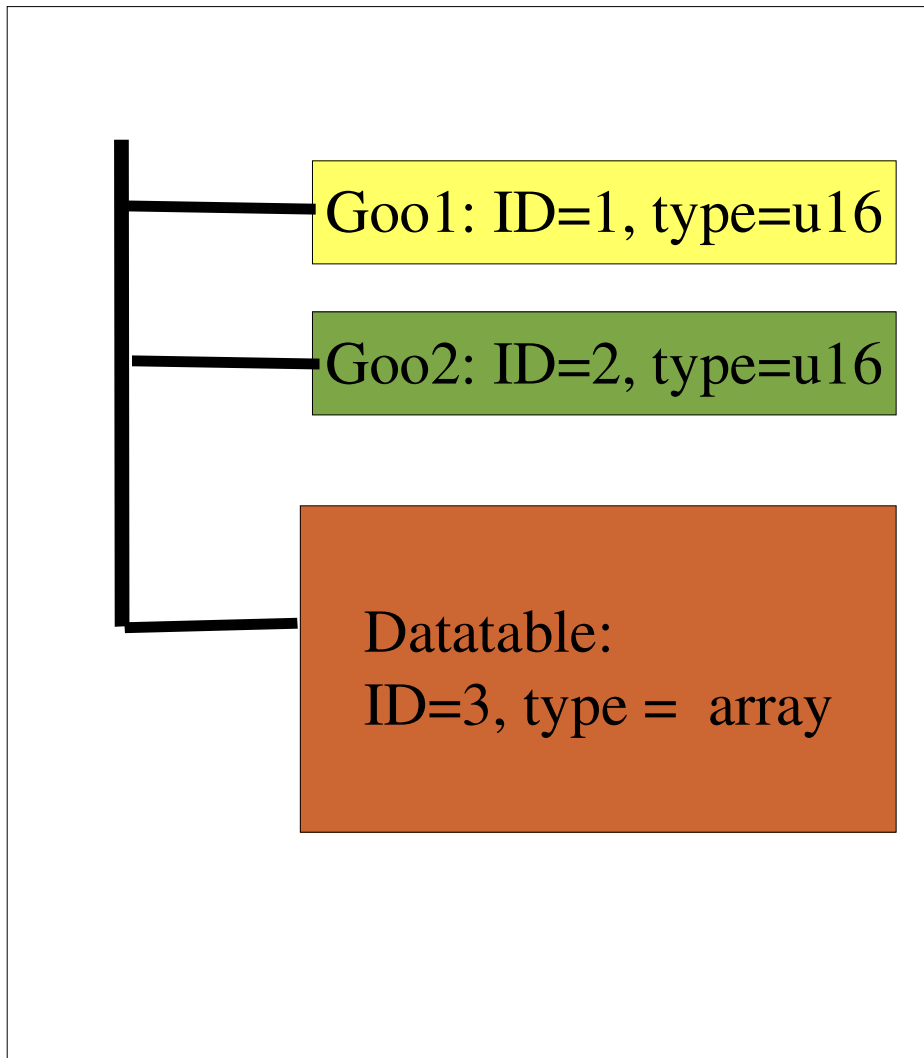
An Example PL Message Layout



Path Data

- A path is a *map* to targeted element/entity
 - Path is a series of 32 bit Identifiers
 - XML definition of LFB requires all elements that can be targeted in a path to have a name and a 32 bit id.
 - A path is not unlike a SNMP OID
 - Element targeted maybe capability, attribute, or event(under discussion)
- Carries associated data where needed
 - Example: Config will have data, but not query.

Example: Attributes for NOP LFB



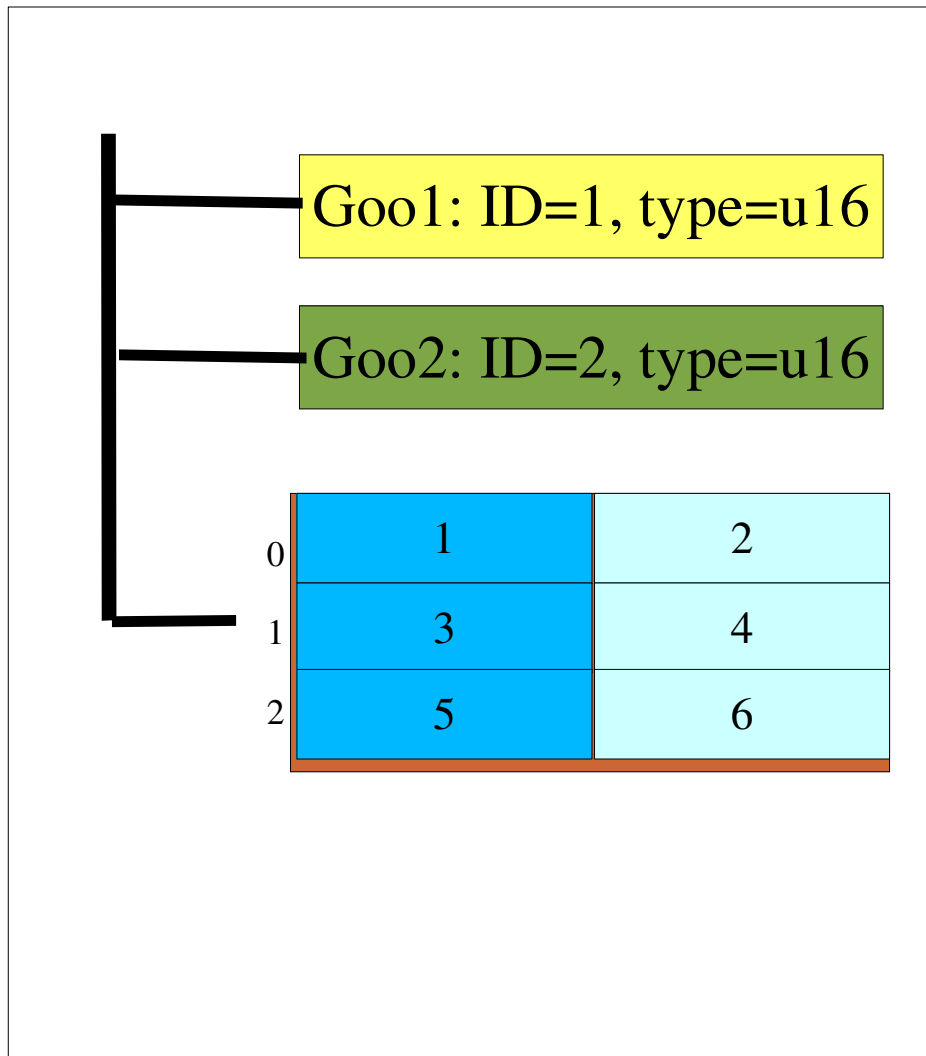
Datatable row constitutes

Foo1: ID=1,
type u32

Foo2: ID=2,
type u32

- Table entries are created at runtime
- Table entries of arrays have index references

Example: Attributes for NOP LFB



- To access datatable
 - ID = 3
- To access first row of datatable
 - ID = 3,0
- To access foo2 in second row of datatable
 - ID=3,1,2

Table operations: Under discussion

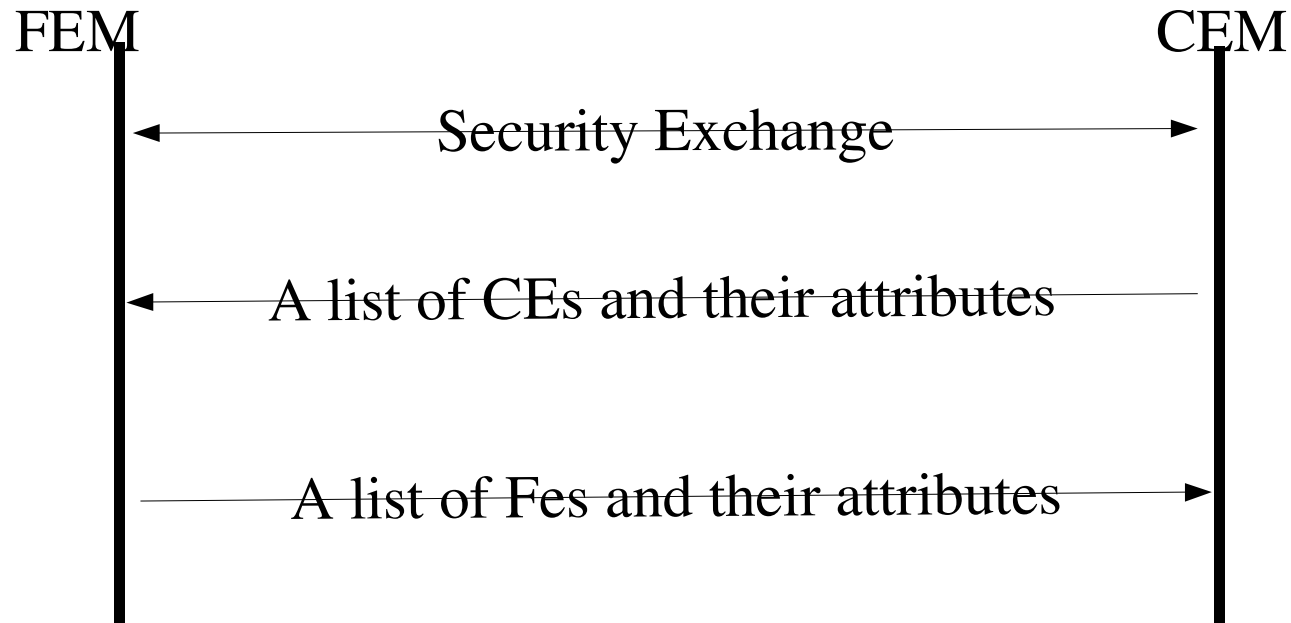
- Block operations
 - Update multiple rows with specified values
 - Eg *start at index = x, for 3 rows*
 - Eg *start at index = x, end at index = y*
 - Update multiple columns with specified values
 - Eg *start at index=x, for 3 rows, and ID=2(foo2)*
- Content based access
 - Example: *for all entries foo1=2*

Open Issues

- These are show stoppers
 - Consensus on path
 - Two proposals in place at the moment
 - Packing/transporting of path referenced data
 - A scheme proposed by Steve/Zsolt insufficient for table hierachies
 - TLV for referenced data (may not be efficient)
- Resolvable
 - Operations on block data
 - Multicasting to LFB instances (Robert will present)

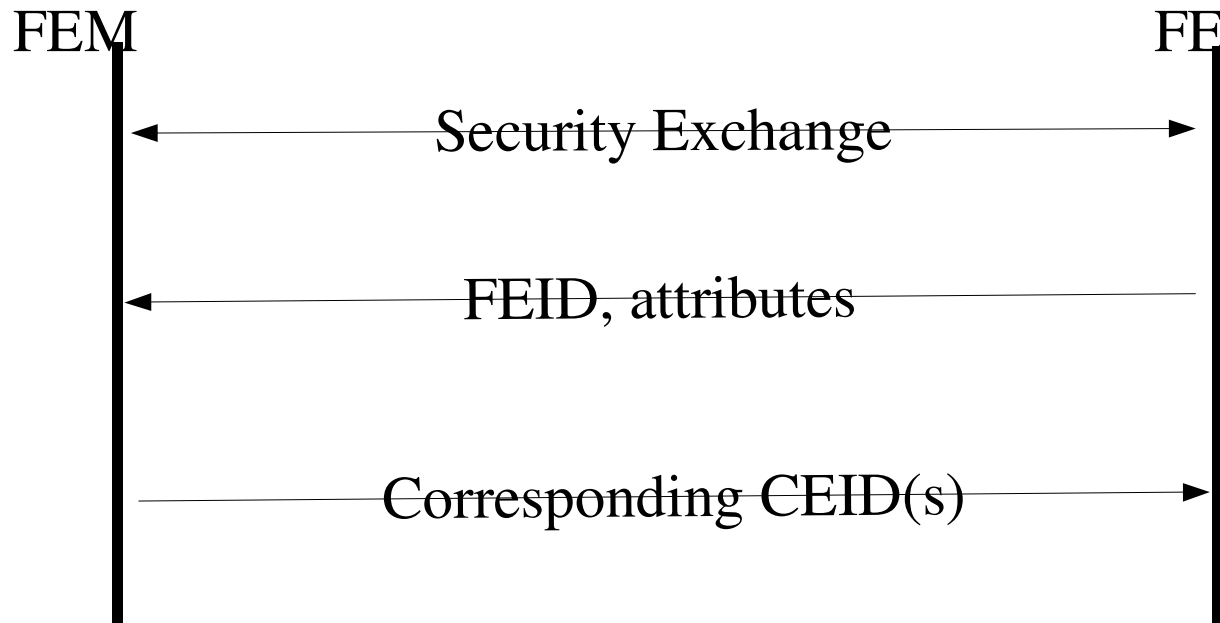
Backtrack

F1 Reference point



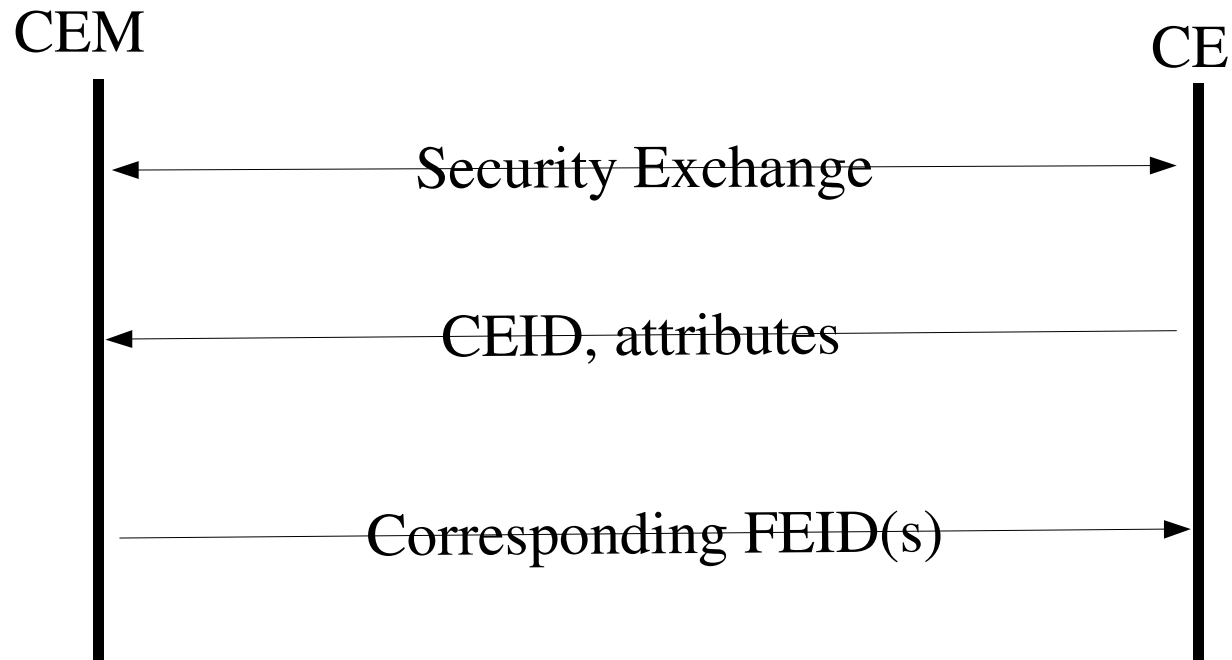
This reference point could be simple, static and based on Fes reading config files

Ff Reference point



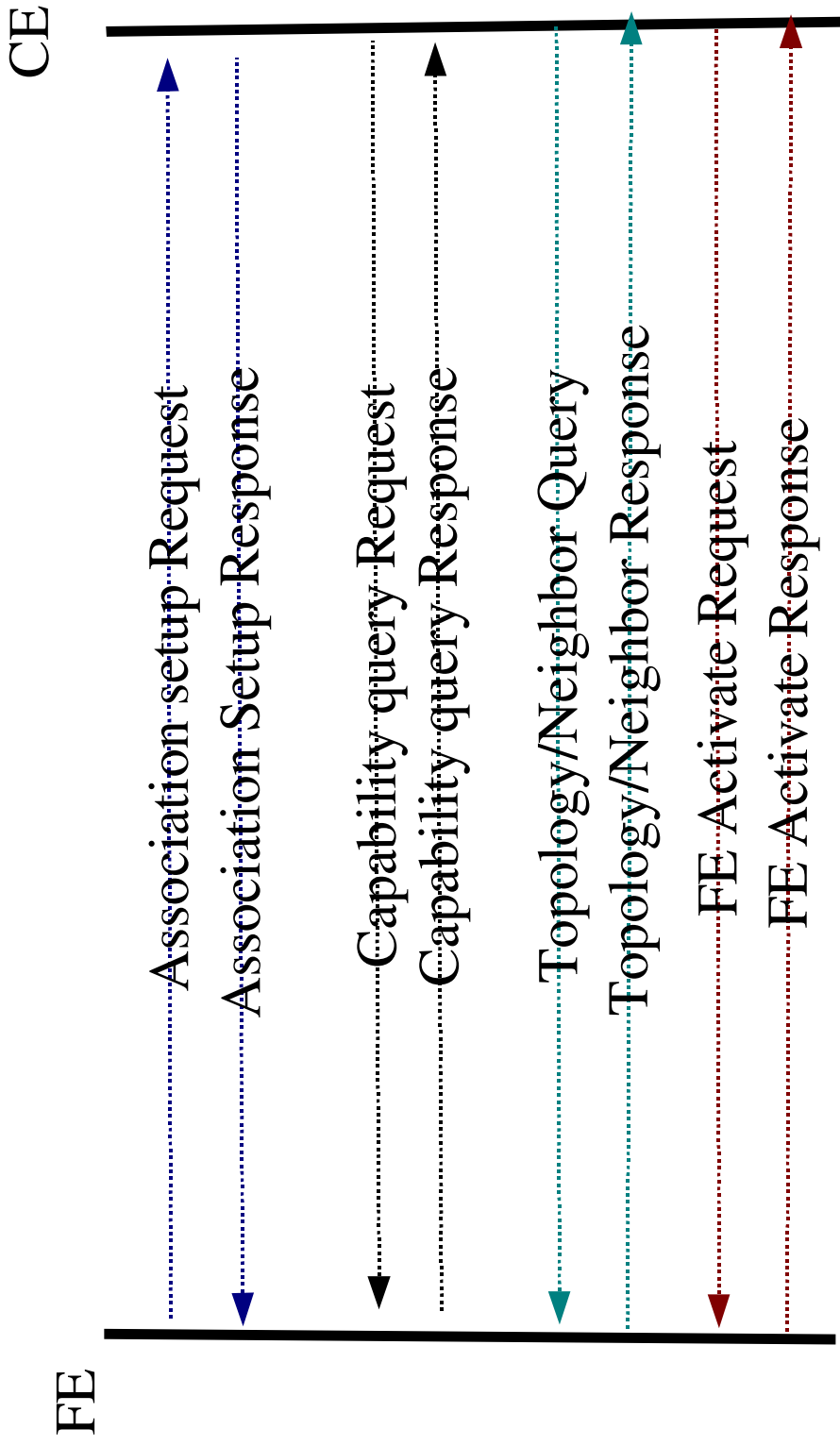
This reference point could be simple, static and based on FE reading config files

Fc Reference point

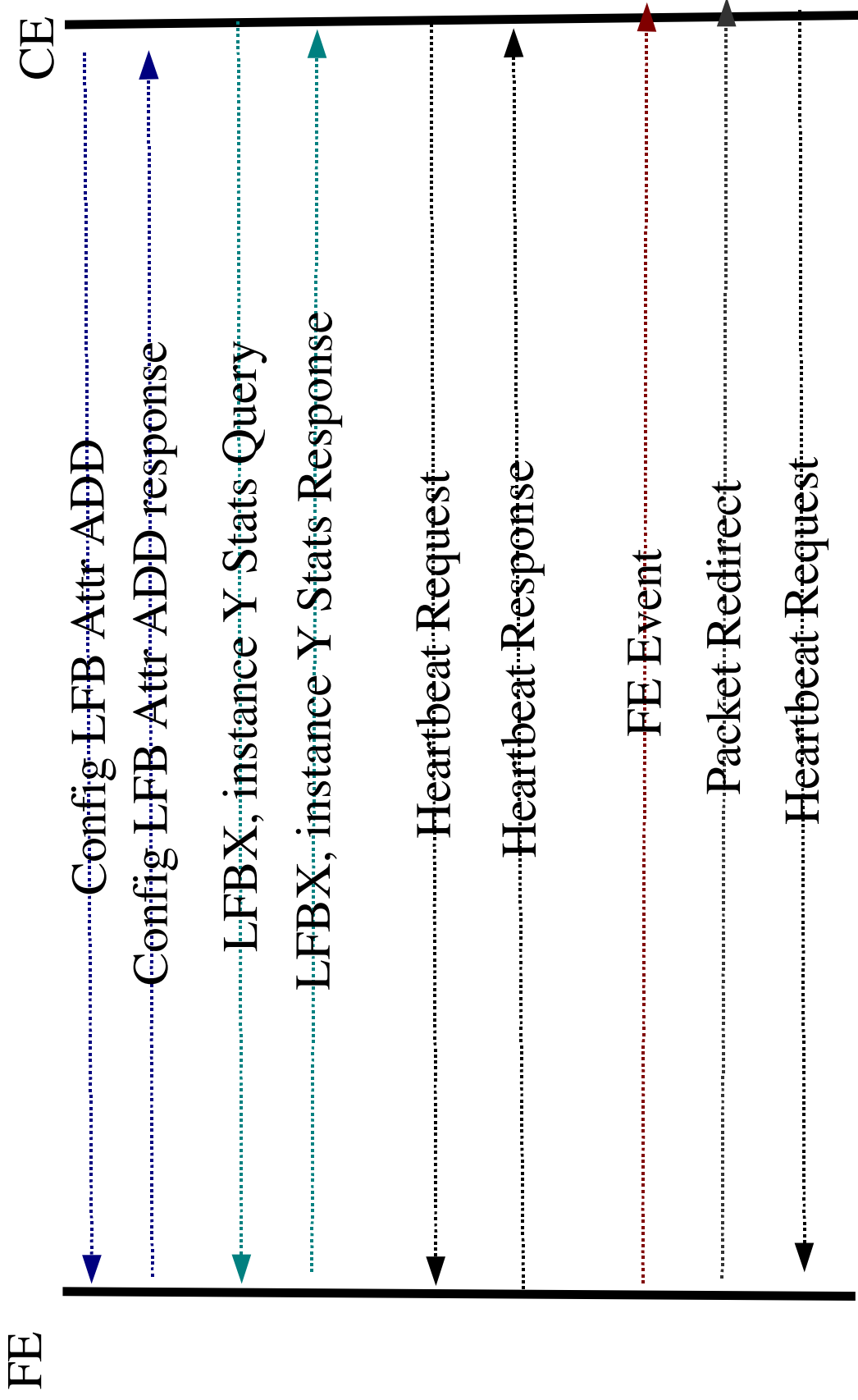


This reference point could be simple, static and based on FE reading config files

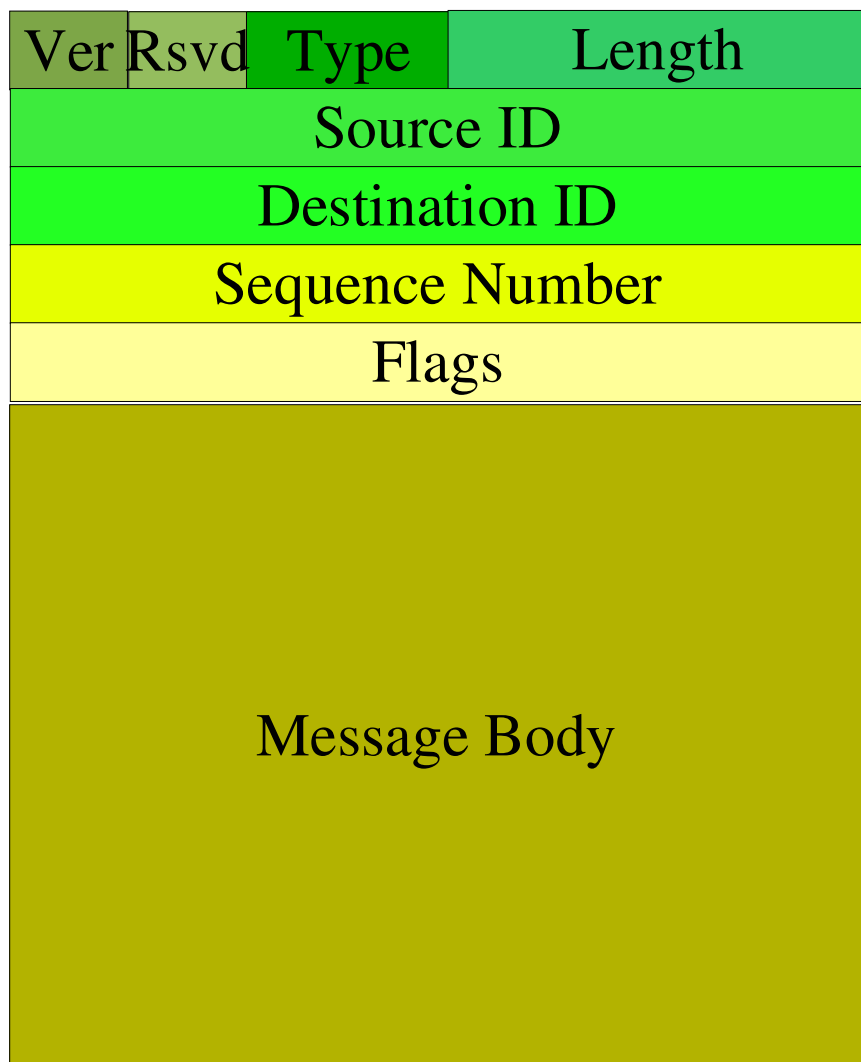
Sample Scenario: FE initialization



Sample Scenario: FE established



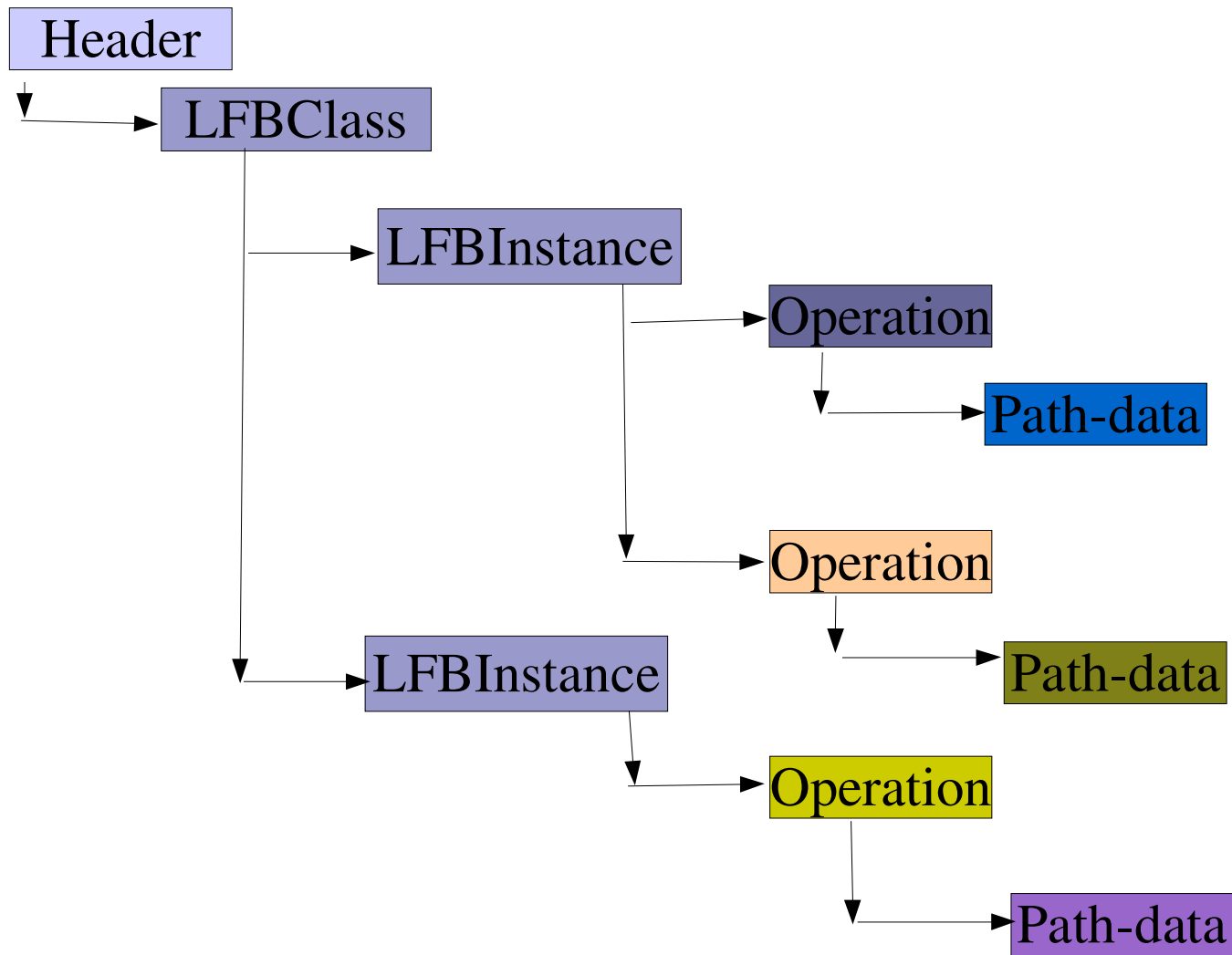
PL Message Layout



- Type defines content of message
- Length DWORDS includes header + message body
- Source ID: The ID of source CE/FE
- Sequence Number of message
- Flags that further define the content in the body
 - Example ACK requested
- Message Body: Constitutes of multiple TLVs nested or in the same level

Source/Destination ID field breakdown

Batching: Multiple LFB Instances



Open Issues

- If possible (and time permits)
 - bring a description made by Weiming and one collected by Jamal and open discussions on them
 - Bring data packing description and point out issues