A Treatise On ForCES
ForCES: Functional scope

- Network Element (NE)
  - Packet Processing Entity
    - Constitutes CEs and FEs
    - Multiple CEs to FE for HA
    - CE/FE Physical or virtual

- NE components distributed
  - Local within box/rack/room
  - Geographically distributed
    - Within rooms/buildings
    - Across the internet
Southbound Approaches

*LFB = Logical Functional Block
Architecture In A Nutshell

- A protocol (The *Verbs*)
  - A modular transport for the protocol
- A data model (The *nouns*)
  - Logical Functional Block constructs
- Combine the above and you have a language
  - [<verb> <noun> [args]]+
    - Few verbs but infinite possibilities of nouns
General Control Example

- Dump the LFB1 table
  GET /FE/3/LFB1/1/table1
- Set row 3 of LFB1 table
  SET /FE/3/LFB1/2/table1/row3 \{a,b,c,d\}
- Batch-update row 11 and 4, col 3 of the LFB1 table
  SET /FE/3/LFB1/1/
    table1/row11/col3 \{y\}
    table1/row4/col3 \{z\}
- Update struct foobar to foo=3, bar=4
  SET /FE/3/LFB2/2/foobar \{3,4\}
- Retrieve foobar/foo
  GET /FE/3/LFB2/2/foobar/foo
LFB Topology Service Chain

Controller

ForCES

LFB1 \(\rightarrow\) LFB2 \(\rightarrow\) LFB3 \(\rightarrow\) LFB5 \(\rightarrow\) LFB4 \(\rightarrow\) LFB6

App \app\ App

P \rightarrow P+M \rightarrow P+M \rightarrow P+M \rightarrow P+M
Distributed Service Chain
(new Recharter Work)

Controller

App  app  App

ForCES

LFB1  LFB2  LFB3  LFB4  LFB5  LFB6

IFE  FEB  P+M  P+M  P+M  P+M

P  P  P+M  P+M  P+M  P+M

Mojatatu Networks
Opening The Network Box™
Protocol Framework

• A Protocol Layer
  - Base ForCES semantics and encapsulation
  - Standardized in RFC 5810

• A Transport Module Layer
  - Depends on underlying media or transport
  - One is standardized (RFC 5811)
    • Expect others to be defined in the future
Protocol Semantics

- Simple Verbs
- Transactional capability (2PC)
- Various Execution modes
- Scalability via batching and command pipeline
- Security
- Traffic Sensitive Heartbeating
- High Availability
SCTP TML

- **Strict Priority Scheduling**
  - **HP**: Strictly reliable channel
    - Configs and Queries issued by CE
    - Response by the FE
  - **MP**: Semi-reliable
    - Event notifications from FE
  - **LP**: Unreliable channel
    - Packet Redirects and HBs
Data Modeling

• Based on XML (RFC 5812)
• Respect for backward and forward compat
  - Let the CE deal with disparate versions
• Main constructs
  - Datatype: definitions used by LFBs
  - LFB Classes: Basic packet processing entity
    - Components: Control entities a CE is aware of
    - Capabilities: define LFB capabilities
    - Events: define events that an LFB can generate
Data Modeling

- Data types: *atomic* and *compound*
- Components have properties
  - ACL (Read/Write permissions)
  - Arrays, Strings, octetstrings, events
    - Events can be subscribed/unsubscribed
- Tables/arrays are accessed by index or by key
- Aliases (symlinks) and optional components
- Read RFC 5812 for more details
LFB model Example

Datatype definition

MyLFB

Components

Capabilities

Events

Example:

```
<dataTypeDef>
  <name>foobartype</name>
  <synopsis>Describes The foobar</synopsis>
  <struct>
    <component componentID="1">
      <name>foo</name>
      <synopsis>that foo</synopsis>
      <typeRef>uint32</typeRef>
    </component>
    <component componentID="2">
      <name>bar</name>
      <synopsis>the bar</synopsis>
      <typeRef>uint32</typeRef>
    </component>
  </struct>
</dataTypeDef>

<component componentID="1" access="read-write">
  <name>foobar</name>
  <synopsis>The Foo and Bar thingy</synopsis>
  <typeRef>foobartype</typeRef>
</component>
```
Development of LFB/Apps

- University of Patras DSL toolkit
- Sample LFB and CE/North App
One day in the life of an FE

FEM

Here's your Config. Go!

FE

Security Exchange (Authentication/Authorization)
FE attempts to Associate and is allowed To join the NE

Capability exchange
FE tells CE what LFBs, Versions, how they can be Inter-connected etc

(assuming modifiable)
CE instantiates different LFB classes

CE creates different Service Topologies

CE

Mojatatu Networks
Opening The Network Box™
One day In The Life OF An FE

Heres your Config. Go!

Config from CE to FE
Unsolicited events and Packet redirects from FE

Config from CE to FE
Unsolicited events and Packet redirects from FE

Di-association
Failure or either side could Terminate the association
Have Model, Will travel

- Initial ForCES intent: southbound datapath control
- Experience: Semantics are good enough for
  - Northbound interface
  - Managing config
  - General purpose encoding and transport of data
    - Not unlike protobuf or thrift
Current Status

- Architecture pretty mature
  - Several implementations exist
  - Two interop meetings in the last 3-4 years
- WG just got rechartered
How To Get Involved

- Read the RFCs
  - https://datatracker.ietf.org/wg/forces/
- Join the Mailing list
  - https://www.ietf.org/mailman/listinfo/forces
- Help us spread the word
  - Tell the ONF about the existence of ForCES
- To contact me: hadi@mojatatu.com
References

- All ForCES documents: https://datatracker.ietf.org/wg/forces/

- A good high level paper

- DSL Paper:
References

Implementations


- Kyota Hattori, Hiroki Date, Kentaro Ogawa, Takahiro Kobayashi, Kenichi Higuchi, Hideaki Iwata, Michihiro Aoki, Shinichiro Chaki. "FDTN Architecture: Functionally Distributed Transport Networking Architecture", SIGCOMM 2009, Demos 1 8, Aug 2009


- Mojatatu Networks (commercial support)