RIB – HEART OF I2RS

CHRIS L.

ED C.

JEFF T.

NITIN B.

SUSAN H.

TIM S.

STAGE 2 – AREAS OF CONCENTRATION

- Scope – What's in scope and what's not in scope
- Write data – What gets written to the RIB
- Events/Notifications – What gets channeled from I2RS Agent to Controller
- Scale – What's needed for scaling RIB interactions

- Stuff not covered
  - Read data – How the controller gets a handle on current state of RIB

SCOPE

- In Scope
    - Reading the RIB
    - Writing to the RIB

- Not in Scope
    - Programming services (Eg. L3VPN) and service chaining
    - Creating (dynamically or via config) and deleting RIBs

WRITE DATA – WHAT GETS WRITTEN INTO THE RIB

- Routes are programmed into the RIB
  - Unicast
  - Multicast
  - MPLS

- Route points to a next-hop
  - Backup next-hop programming capability MUST be supported
  - Backup next-hop MAY be combined with primary next-hop programming

- RIB manager does next-hop resolution
  - Eg. Next-hop specified by i2RS may specify egress point, but not the transport to reach that point

WHAT HAPPENS WHEN ROUTES ARE PROGRAMMED?

- Each route program results in a 2-part Return Code for the operation
  - Installed – Yes/No
  - Active – Yes/No
  - Reason (Eg. Not authorized)

- Route installed by i2RS client **NOT** exported to other RIBs if *Local-Only* flag is set.

5

ASYNC NOTIFICATIONS

- Async notifications are sent by i2RS agent to Controller on a RIB change
  - Installed – Yes/No   - MUST
  - Active – Yes/No   - MUST
  - Reason for notification   - MAY

- RIB change event examples
  - Active route is no longer active because of a better admin-distance route by protocol FOO
  - Inactive route became active as a side-effect of transport LSP coming UP

SCALE – LET'S GET REAL!

- *Systems typically break down on scale*

- Bulking APIs SHOULD be supported for RIB
    - Reads
    - Writes
    - Async notifications

- Per operation return-code MUST be included when bulking is used

ASSUMPTIONS

- i2RS needs to support *Capability Negotiation*
  - AF/SAFI that the i2rs agent supports

- Pub/sub model not a requirement for events and notifications

- Controller inserted bad routes is a problem of the controller

- Debugging & trouble-shooting should be part of a separate working team
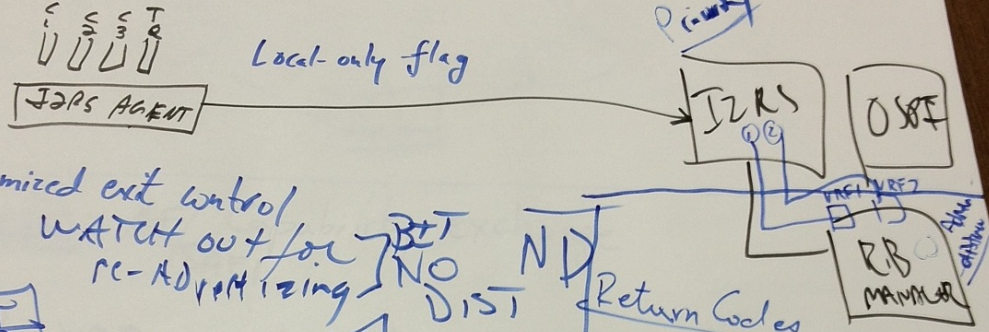
APPLYING THIS TO USE-CASES

- Each use-case can be converted into a route/next-hop programming on one or more network elements

- Data-model details will depend
    - On what the routes look like
    - What the next-hops look like
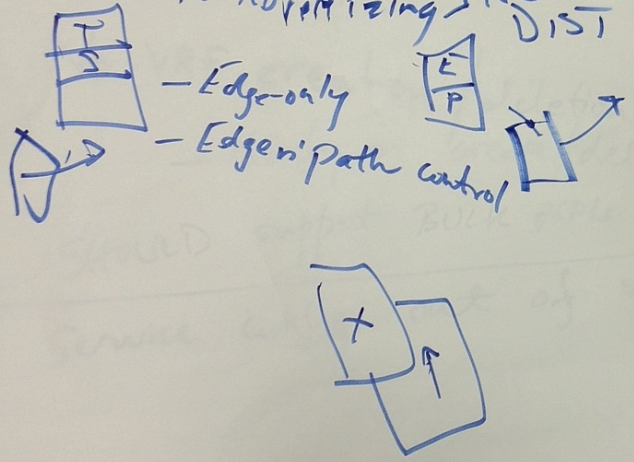   - **=> *What the use-case really wants***

THANK YOU!

APPENDIX: PICTURES OF BRAIN-STORMING

~~RIB~~ AF/ Capability exchange
/SAFI

No VRF creation /deletion
— Notify on create/delete

SHOULD support BULK APIs

Service config out of scope