

EFM OAM Loopback

**Ben Brown – AMCC
Kevin Daines – World Wide Packets
Don Pannell – Marvell
Al Braga – UNH IOL**

Changes to Discovery

- Add 2 new LOCAL_STATE & FAR_END_STATE bits
- New state encoding
 - UNSTABLE
 - STABLE
 - LOOPBACK_STABLE
 - LOOPBACK_UNSTABLE
 - LOOPBACK_COMPLETE
- Use modified Discovery state machine to move from Loopback_xxx states to STABLE
 - Still needs to be drawn

Startup

- **Device A queues Loopback Control OAMPDU with non-zero Loopback Quantum value**
- **Device B responds with Status OAMPDU with new LOCAL_STATE value of LOOPBACK_STABLE**
- **Device A waits for Device B change of state, resending the Loopback Control OAMPDU if necessary**
- **When Device A state is STABLE and Device B state is LOOPBACK_STABLE, Device A's MAC Client initiates loopback data**
- **Device B uses this state combination to loop data frames**
- **Device A uses this state combination to drop data frames**

Timeout Finish

- **Device A keeps a copy of Device B's Loopback Quantum timer**
- **This counter is loaded and begins counting upon generation of Loopback Control OAMPDU**
- **Device A's counter will expire before Device B's counter**
- **Device A's MAC Client stops sending loopback data when its copy timer expires**
- **Device B reflects a state change upon its timer expiration**
- **Device B sends a Status OAMPDU with LOCAL_STATE = LOOPBACK_UNSTABLE**
- **Device B uses this state combination to drop data frames**
- **Device A uses this state combination to drop data frames**

Loopback Timer Extension

- To extend Device B's loopback quantum timer, Device A queues additional Loopback Control OAMPDUs with non-zero Loopback Quantum value
- These can be lost with no knowledge of either side
- It is recommended that these are sent more often than absolutely necessary but there are still no guarantees
- If all extension frames are lost, Device B can still timeout
- Device B sends a Status OAMPDU with LOCAL_STATE = LOOPBACK_UNSTABLE
- Device A's MAC Client stops sending loopback data

Loopback Control Finish

- **Device A's MAC Client stops sending loopback data when it desires to end the loopback test**
- **Device A queues Loopback Control OAMPDU with zero Loopback Quantum value**
- **Device B responds with Status OAMPDU with new LOCAL_STATE value of LOOPBACK_UNSTABLE**
- **Device A waits for Device B change of state, resending the Loopback Control OAMPDU if necessary**

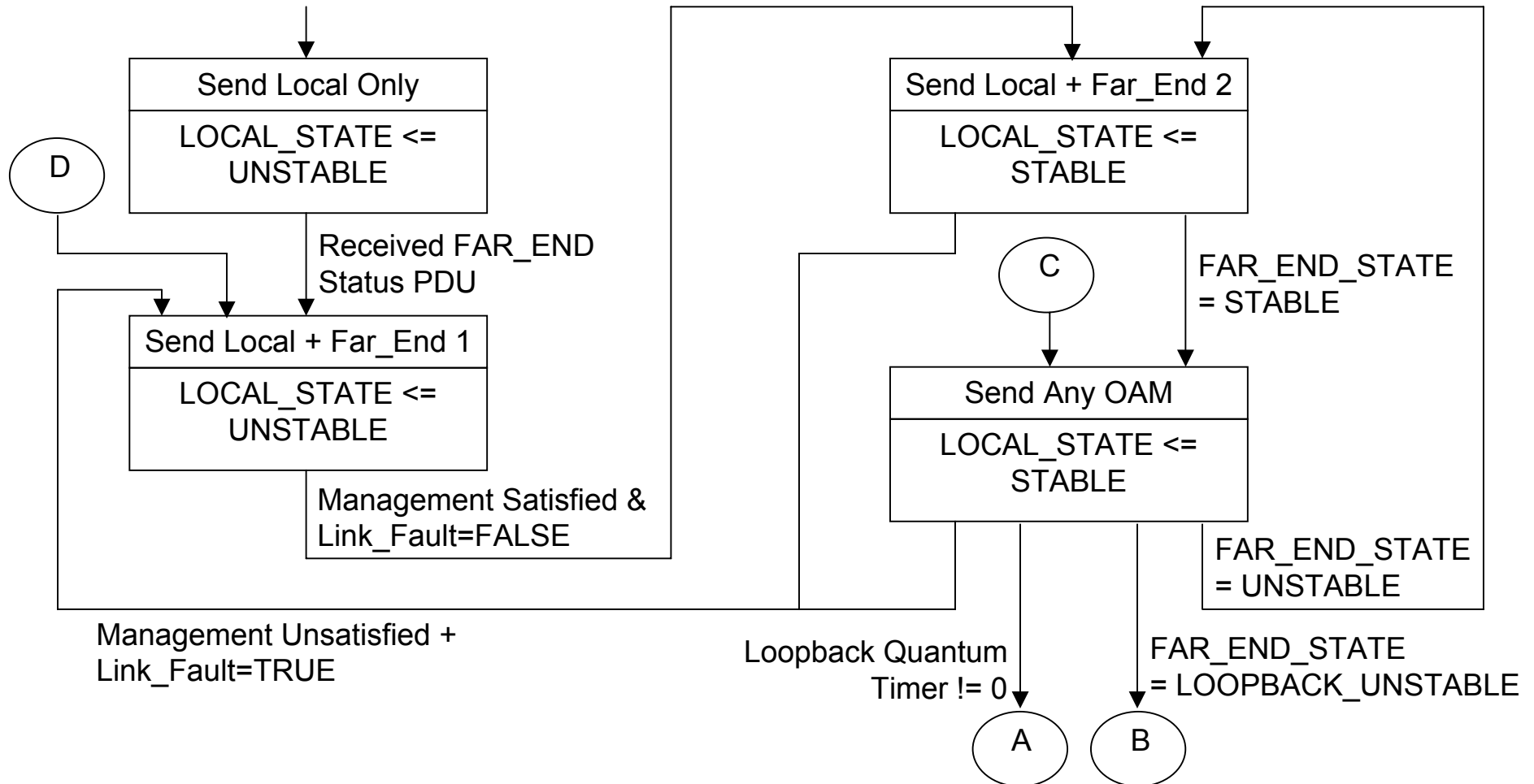
Loopback Conclusion

- When Device A state is **STABLE** and Device B state is **LOOPBACK_UNSTABLE**, Device A changes state to **LOOPBACK_COMPLETE**
- When Device A state is **LOOPBACK_COMPLETE** and Device B state is **LOOPBACK_UNSTABLE**, Device B changes state to **STABLE**
- When Device A state is **LOOPBACK_COMPLETE** and Device B state is **STABLE**, Device A changes state to **STABLE**
- Until both devices are **STABLE**, both Device A and Device B use these state combinations to drop data frames

OAM Discovery State Machine

Page 1

Reset + Restart + Link Loss Timer Expired



OAM Discovery State Machine

Page 2

