

BGP BFD Strict Mode

draft-ietf-idr-bgp-bfd-strict-mode-13

Mercia Zhang, Acee Lindem, **Jeffrey Haas**, Albert Fu

What is BFD

- Bidirectional Forwarding Detection (BFD) provides a light-weight protocol that can be leveraged by client protocols to quickly detect connectivity faults between two systems.
Protocols, like BGP, that use second-level timers can use BFD for faster failure detection, or to assist in scaling faster timers.
- RFC 5882 describes the generic procedure for using BFD with network components. In general, “when BFD goes down, take the protocol down”.

What's the problem?

- RFC 5882 isn't specific about when BFD is used with a network component.
 - Some BGP implementations would start BFD prior to BGP becoming Established - perhaps before the first BGP packet is sent.
 - Some BGP implementations would start BFD only after BGP reached Established.
 - These different implementation behaviors could cause interoperability issues, including deadlocks.
- Some use cases for BFD would also require the BFD session to be stable for "long enough" before letting BGP send routes. This is a BGP route stability consideration.

What's BGP BFD Strict Mode?

- BFD strict mode is BGP capability, when negotiated between two BGP speakers, that determines when BFD should be started vs. BGP and when BGP should move to Established based on BFD session status.
- Tersely:
 - BGP should start its state machine without waiting for BFD to be Up
 - Implementations, might want to start the BFD session prior to starting the BGP FSM for faster BFD session setup vs. the dependent BGP session.
 - BGP doesn't send its final KEEPALIVE message to move to Established until BFD is Up.

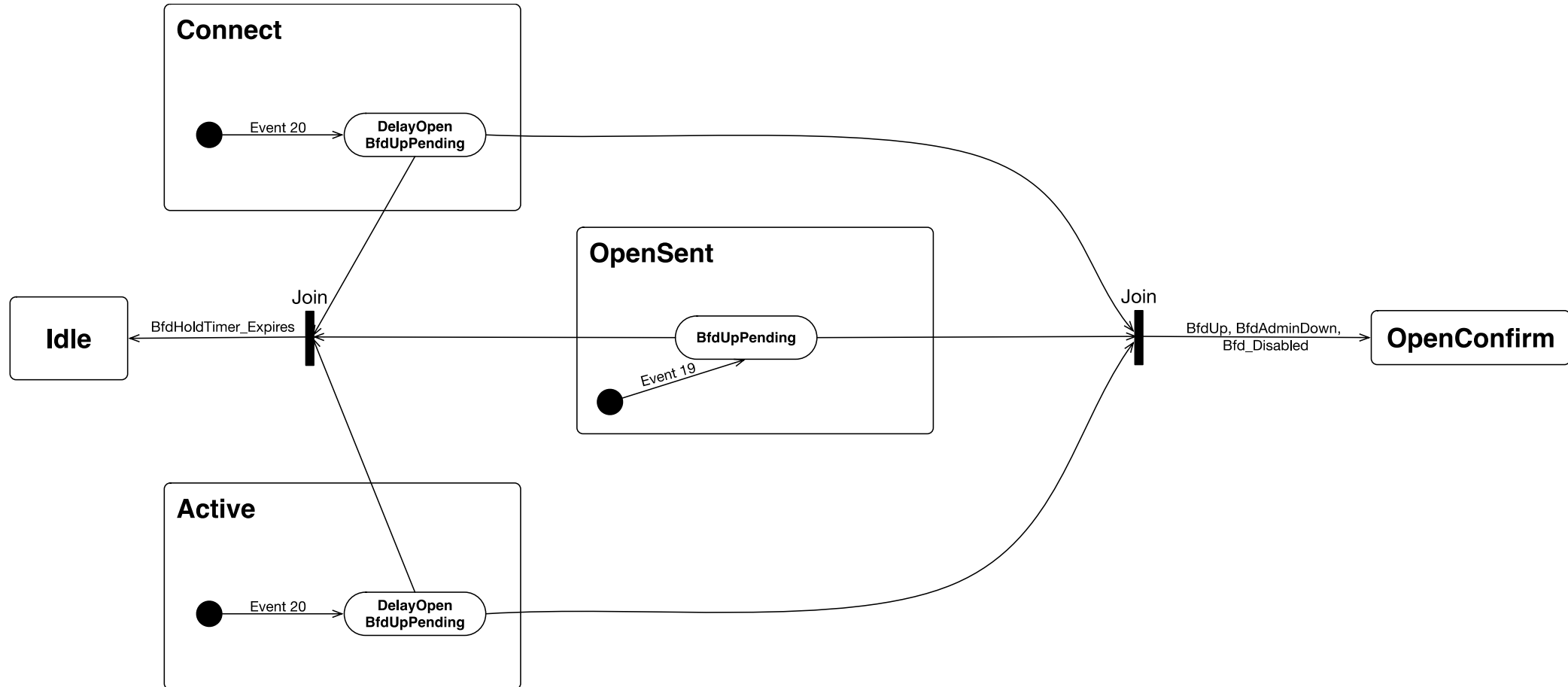
What's changed in version -13

- In versions prior to -13, the text basically said upon entering the OpenConfirm state, the implementation should not send its keepalive until BFD goes Up.
- Ketan Talalulikar suggested a deep audit of the BGP FSM behaviors for correctness. That audit resulted in:
 - We want to stay in the states leading up to OpenConfirm and control advancement to OpenConfirm depending on the BFD session state.
 - This resulted in new sub-states for BGP.
 - (This also fed a broader conversation about maintaining the BGP FSM.)
 - We also documented the BFD going Down impacts for BFD.

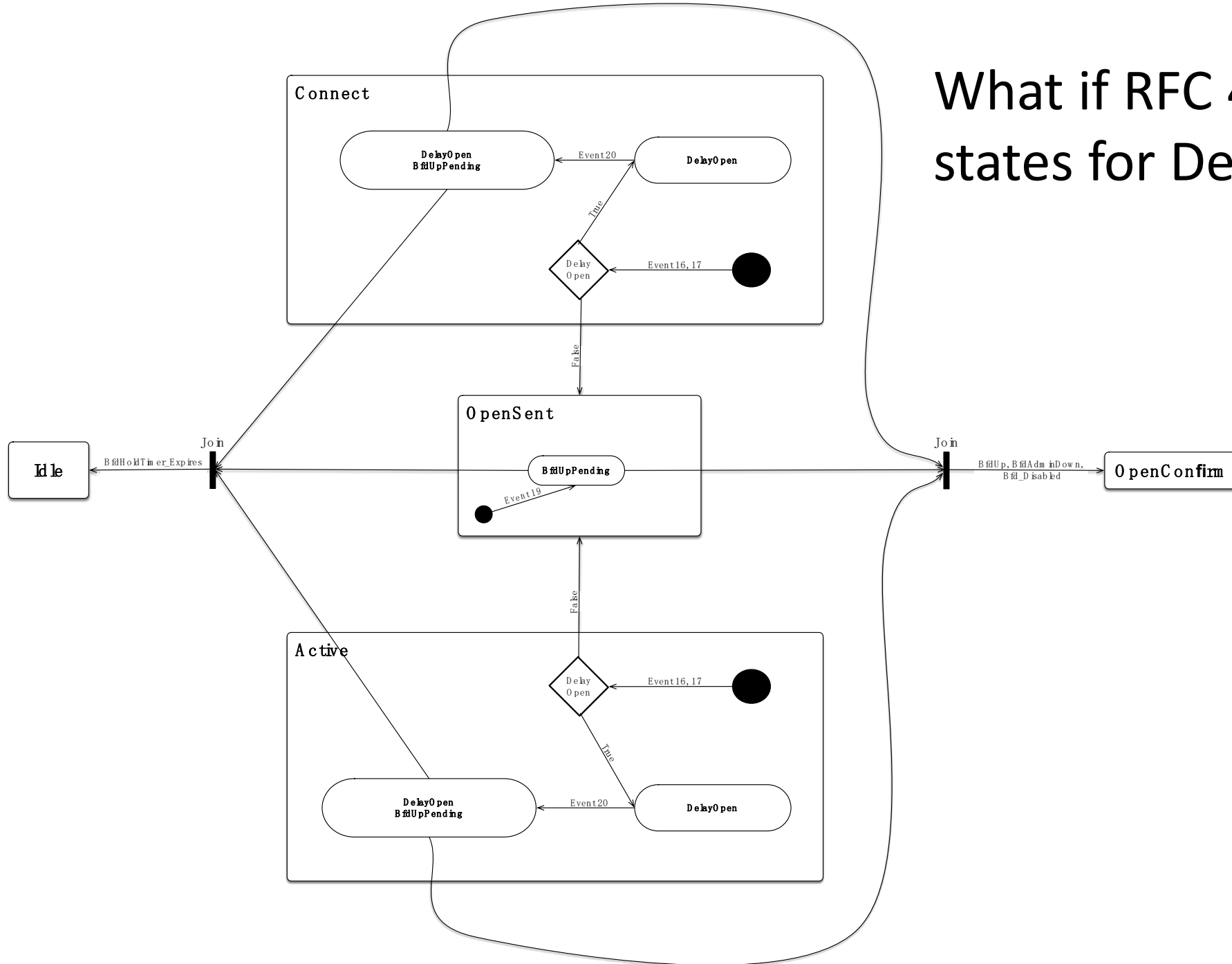
Document status

- We have early chair feedback on the current text. This is a prelude to the shepherd's report for WGLC.
- The implementors will need to tweak how their implementations report the states.
- Likely ready for WGLC at IETF 121.

UML for representing BGP BFD Strict sub-states



What if RFC 4271 BGP had sub-states for Delay Open?



Questions?

- Note, more general FSM discussion will be done in the BGP-QUIC presentation...