

IETF YANG models for VLAN interface classification

[draft-wilton-netmod-intf-vlan-yang](#)

Robert Wilton (Cisco)

rwilton@cisco.com

Presentation Objectives

- To explain what this IETF model is trying to achieve
- To justify why this is being standardized in IETF
- To hopefully get agreement from IEEE 802.1 that it is acceptable for IETF to standardize this model ...

Without this model a lot of standard IETF protocols are unconfigurable via YANG, or unusable with 802.1Q VLAN tagged traffic!

The draft is currently blocked in IETF at the equivalent of the PAR phase – IETF doesn't want to antagonise IEEE 802.1 by starting a project regarded as conflicting with IEEE.

What is an interface?

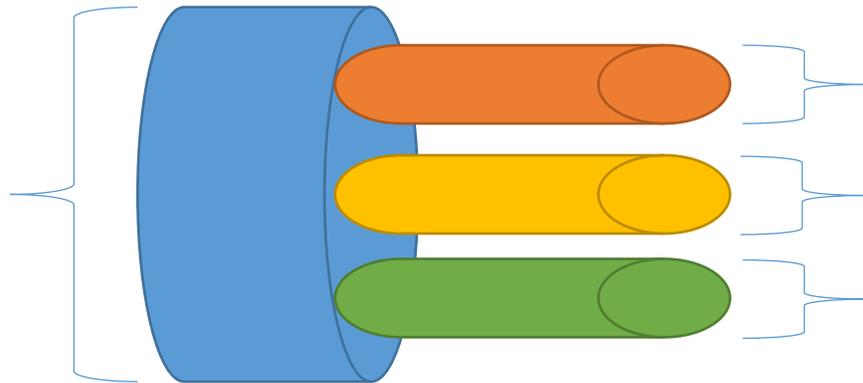
Based on the IFMIB & ietf-interfaces.yang, an interface can be defined as a generic container with the following characteristics:

- **It identifies a stream of network traffic** (potentially at any layer)
- **It is an anchor point to apply features and protocol forwarding configuration on that stream of traffic**
- It has an interface type (IANA defines many different flavours)
- It has a default set of traffic statistics
- It can be enabled/disabled via configuration

What is a child interface? Why is it needed? (1)

- Sometimes it is necessary to apply features or forwarding rules to a subset of traffic on an interface
- This can be modelled cleanly using child-interfaces

Parent interface
E.g.
Phy Ethernet,
LAG,
Pseudo-wire,
...



Child interfaces

Traffic forwarded by IP in VRF x

Traffic forwarded by IP in VRF y

Traffic forwarded via VPLS instance

...

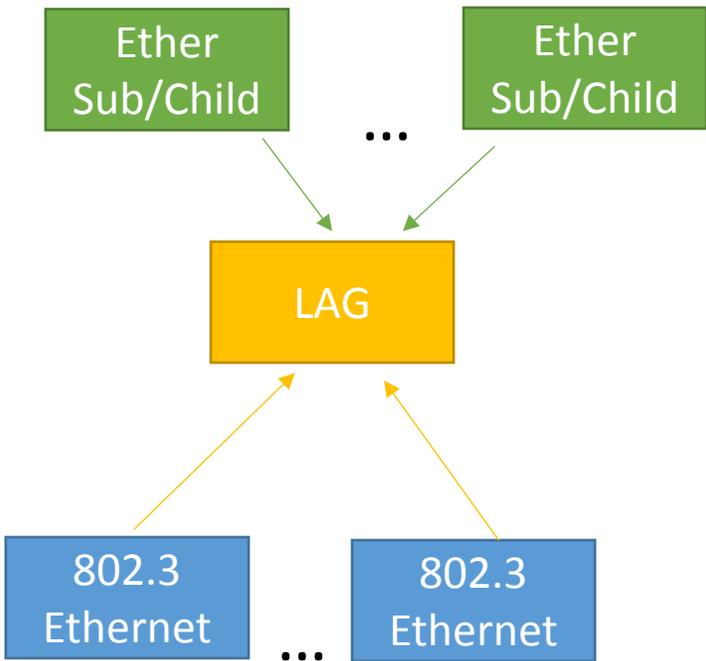
What is a child interface? Why is it needed? (2)

- A child interface is a type of interface that allows separate feature and forwarding rules to be applied to a subset of the traffic of any parent interface.
 - Feature/forwarding YANG schema paths are the same as for any other type of interface!
 - Lots of different parent interface types can have child interfaces: Physical Ethernet, LAG, Pseudo-wire Head end, Bridge virtual interface, Frame Relay, ATM, ...
- A parent interface can have many child interfaces if required
- Multiple layers of interface hierarchy are possible
- Classification rules are used to demux traffic from a parent interface to its children interfaces
 - For interfaces with Ethernet framing, VLAN Ids are often used as the demux classification
- Forwarding is configured via other IETF protocol YANG models (IPv4, IPv6, MPLS, Pseudowires, VPLS, EVPN, L2TPv3)

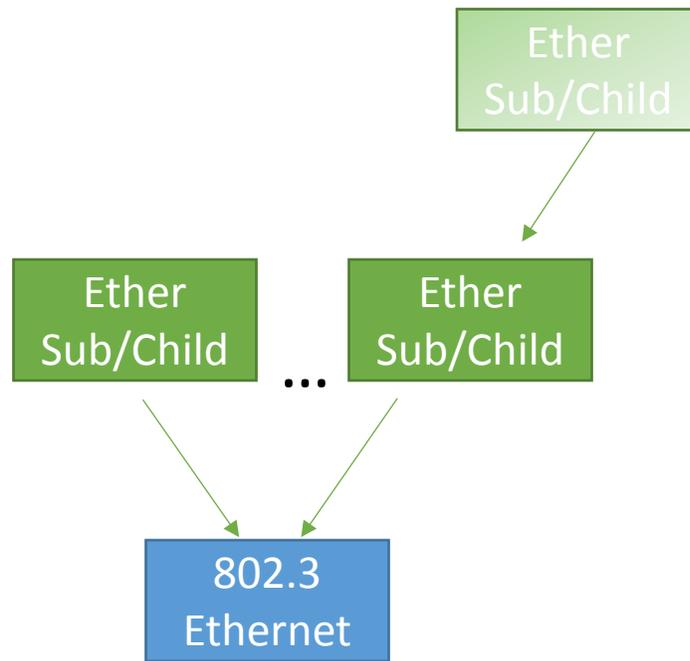
What is a child interface? (3)

Abstract relationship between interfaces

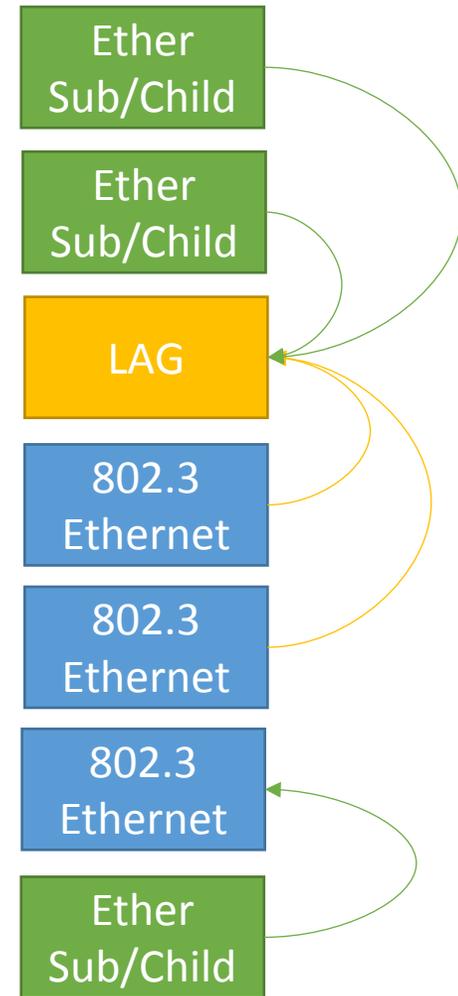
YANG representation



LAG example



Ethernet example



What isn't a child-interface?

- It is not just traffic associated with a particular single VLAN Id
It is far more flexible, matching on sets of tags, or other fields.
- It has no predefined forwarding semantics
Forwarding models can just refer to an if:interface without caring about what type of interface it is, or whether it is a parent or child. IETF forwarding models work with VLANs with no extra changes required.
- It is not an alternative way of configuring an IEEE 802.1Q bridge
Mostly devices that implement child interfaces don't implement IEEE 802.1Q bridging, although there is no reason why they can't coexist on the same device.
- A LAG member is not a child interface
A different relationship is required because there are N LAG members to one LAG, and no explicit demux is required.

Why standardize? Why in IETF?

- Many vendors have proprietary configuration constructs similar to what is being proposed.
- No existing standards exist for this technology in any standards body because historically the end user configuration has not been standardized.
- However, there is now a strong market demand for automation via standard YANG models.
- Many IETF forwarding YANG models are incomplete without being able to interop with VLAN tagged traffic.
- IETF is the right place because IETF owns the generic interface model, this is just an extension, and VLANs are one example.

Clearing up other areas of confusion

Q. Doesn't ietf-interfaces already model this?

No, it has read-only leaves for operational state, separate configuration leaves are required (as per RFC 7223 section 3.3)

Q. Is the long term plan for IEEE to adopt ownership of this model?

No, this model covers different technology/protocols to those standardized in IEEE.

Q. Does this model allow different VLAN traffic on LAG member interfaces?

No, this model can't be used for LAG membership.

Q. Will this model redefine VLAN types?

No, it will be updated to use the IEEE defined types where possible - we may request that some additional ones get added/defined by IEEE.

Issues raised from last TSN call

(1) Agreeing the name of these “child-interfaces” objects:

- Calling this sub-interfaces seems to cause some confusion, or have some undesirable historical baggage.
- Calling these VLAN interfaces also causes confusion with existing usage.
- My current preferred choices are calling them either:
Ether sub-interface or Ether child interface

(2) Does IEEE 802.1 plan on developing a host YANG model for interop with 802.1Q bridges?

- Marc Holness indicates that there are no current plans.

Open issues (2)

(3) Should there be restrictions on what VLAN tags the IETF model should be allowed to match on?

- E.g. if double tagged, could only allow matching traffic on outer S-VLAN, inner C-VLAN.
- This ensures that the double tagged traffic interops cleanly with 802.1Q bridges
- Broadband forum isn't keen on this restriction – they would prefer greater flexibility. As well as 802.1Q bridging, VLAN tags are also being used as a generic tagging mechanism to keep multiple traffic streams on an Ethernet interface separate.

Next steps?

- Does anyone object to this work proceeding in IETF as a WG document?
- Otherwise what is required to unblock it:
 - A formal liaison from IETF to IEEE?
- I'm attending the interim until the end of Thursday, so if you have views on this then please talk to me, or send me an email (rwilton@cisco.com)
- I will continue to incorporate feedback and evolve the model - on the hope/assumption that it will eventually get unblocked 😊

Backup slide

- More generic picture of how child interface can be used to configure different forwarding services.

