IEEE 802.1Q YANG Bridge Port Interface Model in Support of 802.1AX, 802.1X, etc.

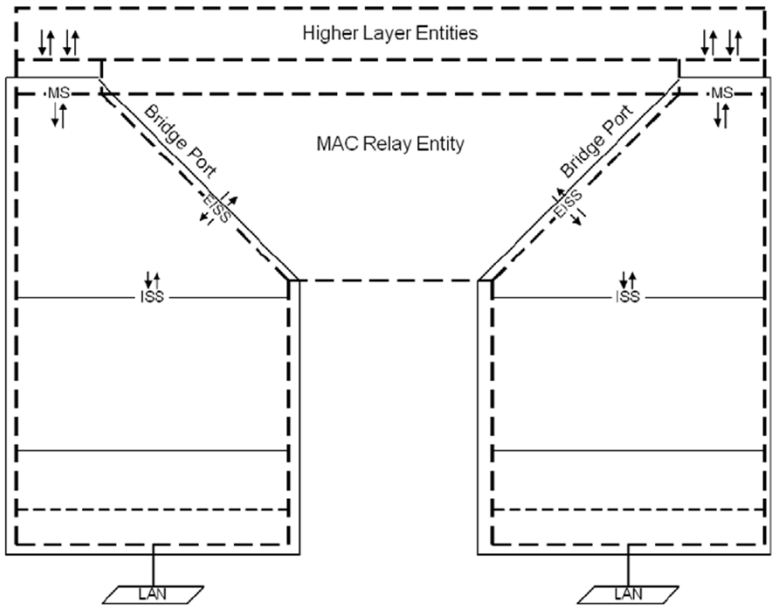
Marc Holness
Version 0.53
10 Aug 2016

IEEE 802.1Q Bridge Port

- Each Bridge Port is associated with one Interface, and in most situations, each Bridge Port is associated with a different interface
- However, there are situations in which multiple Bridge Ports are associated with the same interface
  - For example, several Bridge Ports can each correspond one-to-one with several Ethernet private lines (or SDH virtual circuits) but all on the same Interface
  - Or multiple Bridge Ports can each correspond to a single internal LAN (I-LAN) port
- Alternatively, there is the Link Aggregation (IEEE Std 802.1AX) case where there are many physical Ports for one Bridge Port

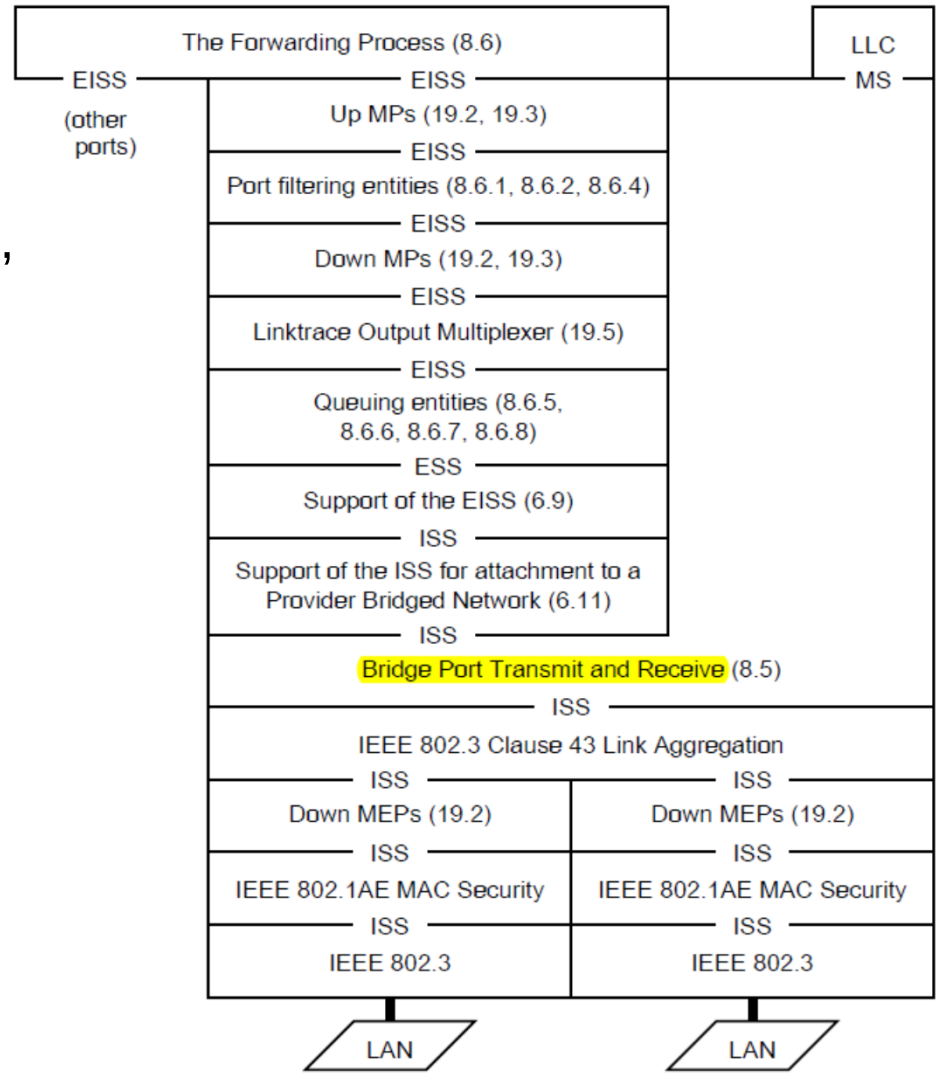
IEEE 802.1Q Bridge Port

- The MAC Relay Entity handles the media access method-independent functions of relaying frames among Bridge Ports. It uses the EISS (6.8, 6.9) provided by each Bridge Port
- Each Bridge Port also functions as an end station and shall provide the MAC Service to an LLC Entity that operates LLC Type 1 procedures to support protocol identification, multiplexing, and demultiplexing, for PDU transmission and reception by the Spanning Tree Protocol Entity and other higher layer entities

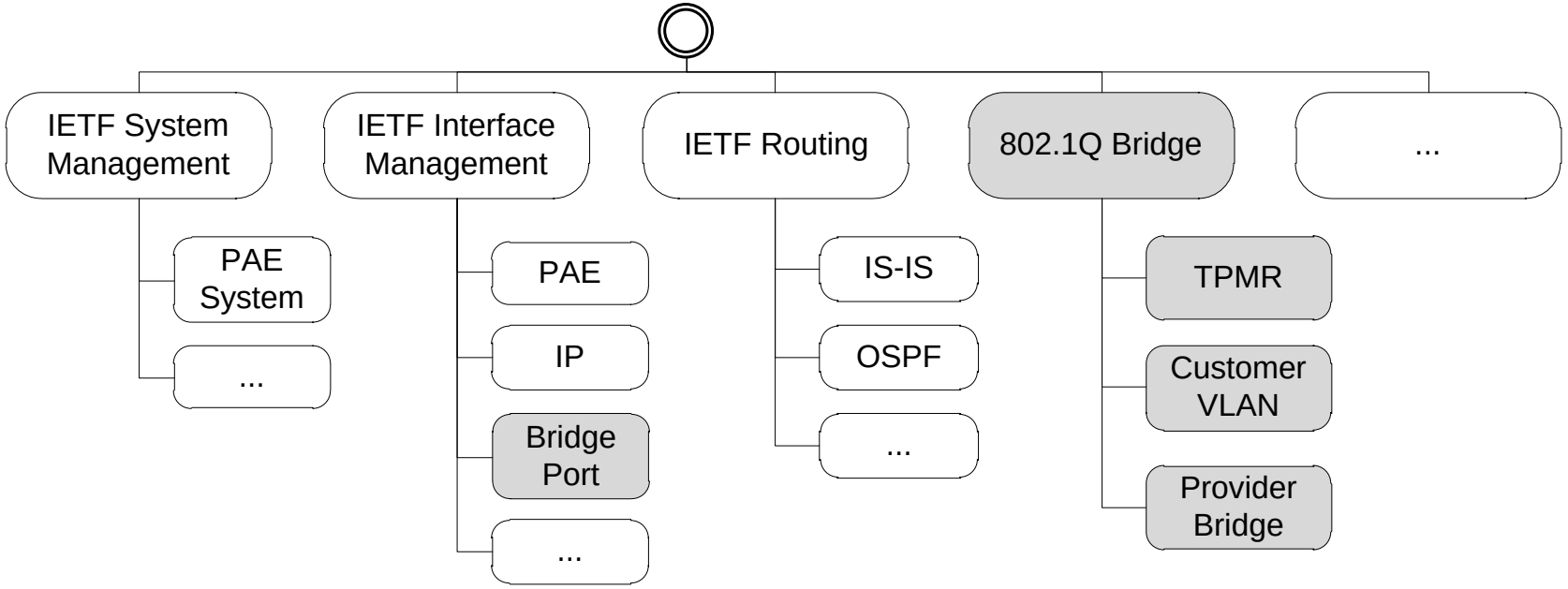


CFM Maintenance Point Placement

- CFM entities are specified as shims that make use of and provide the ISS or EISSS at SAPs within the network
- The relationships among MPs, and between the MPs and the other entities in a Bridge, are configurable

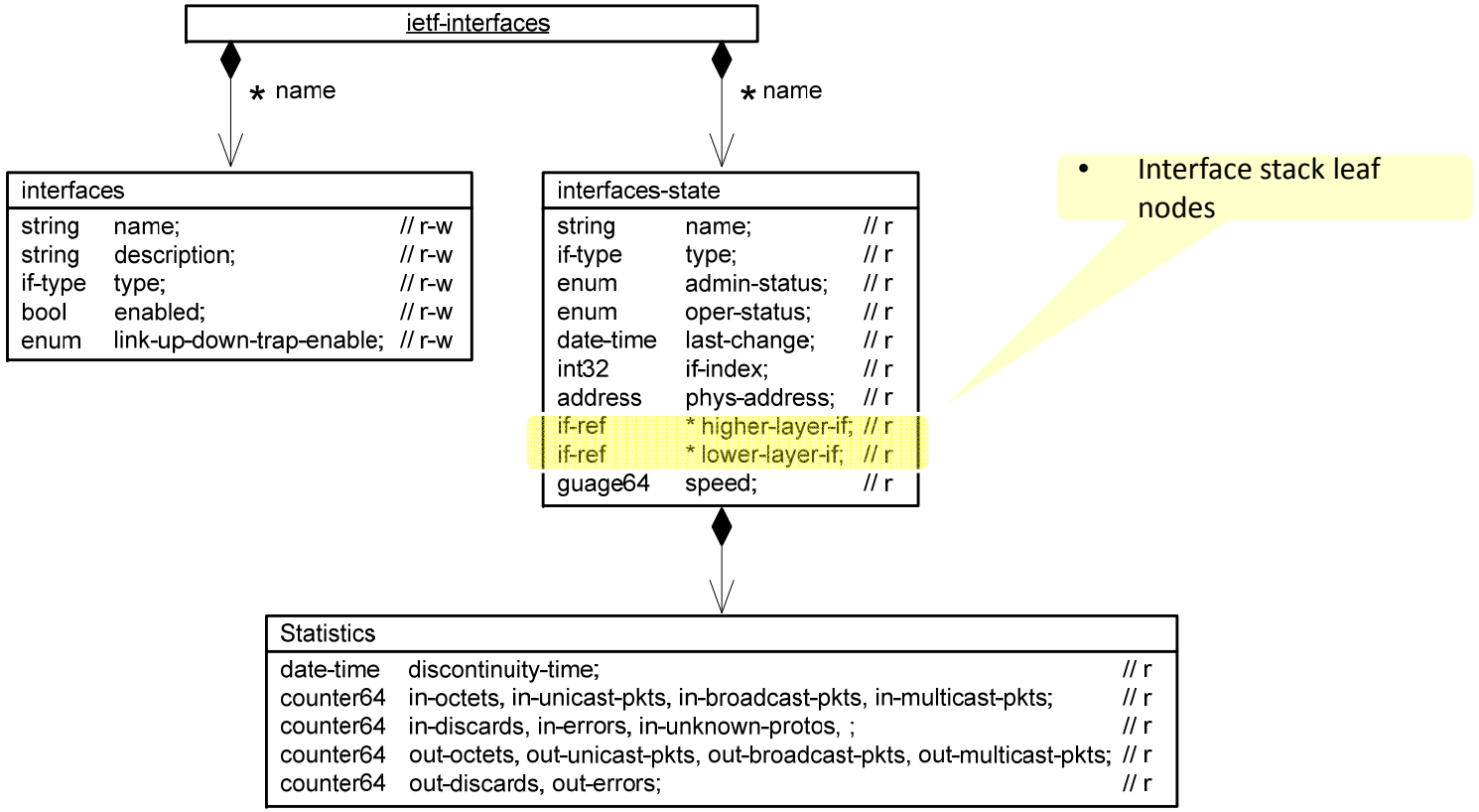


IEEE 802.1 Objects Within YANG Object Hierarchy

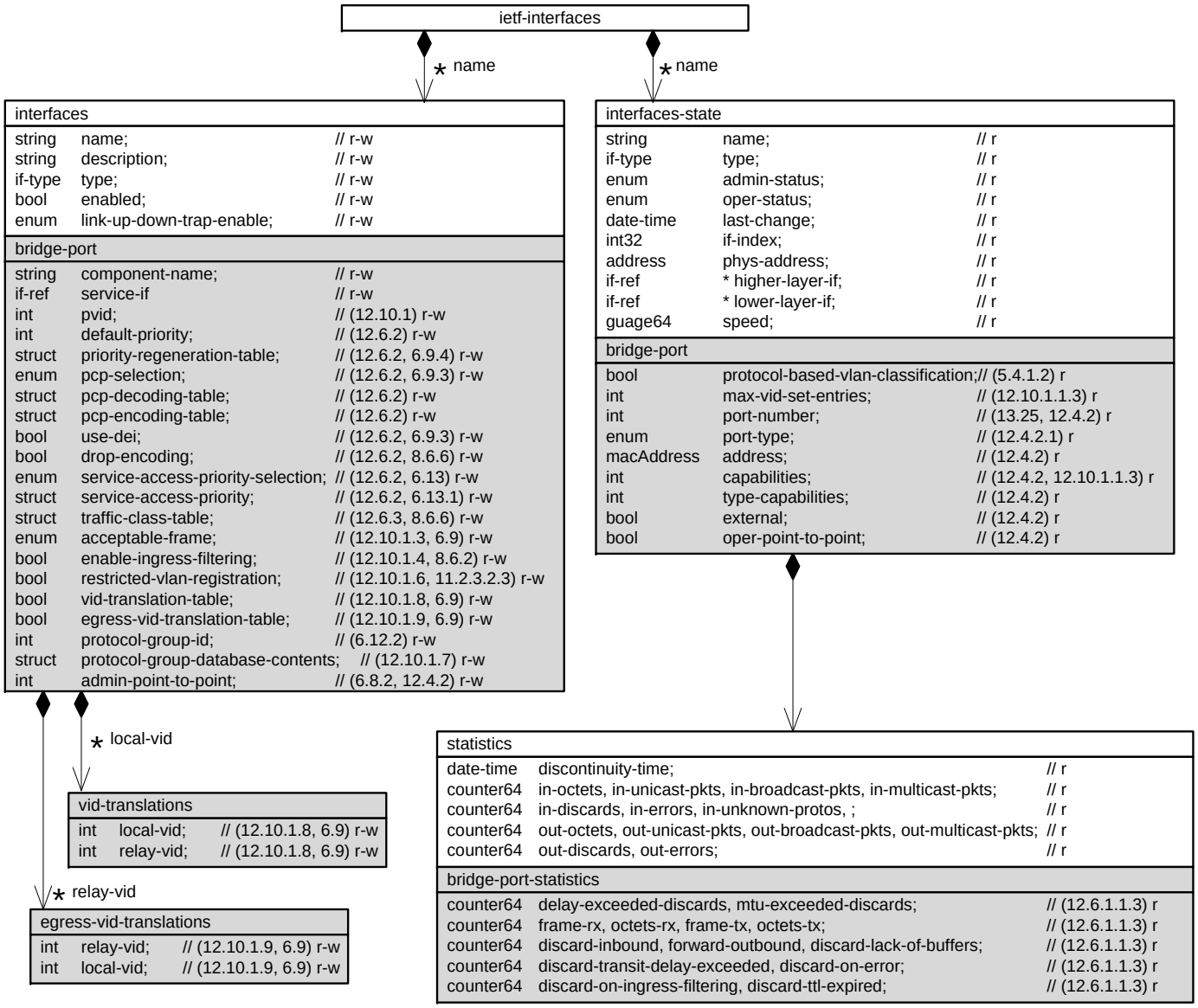


IETF Interface Management Model

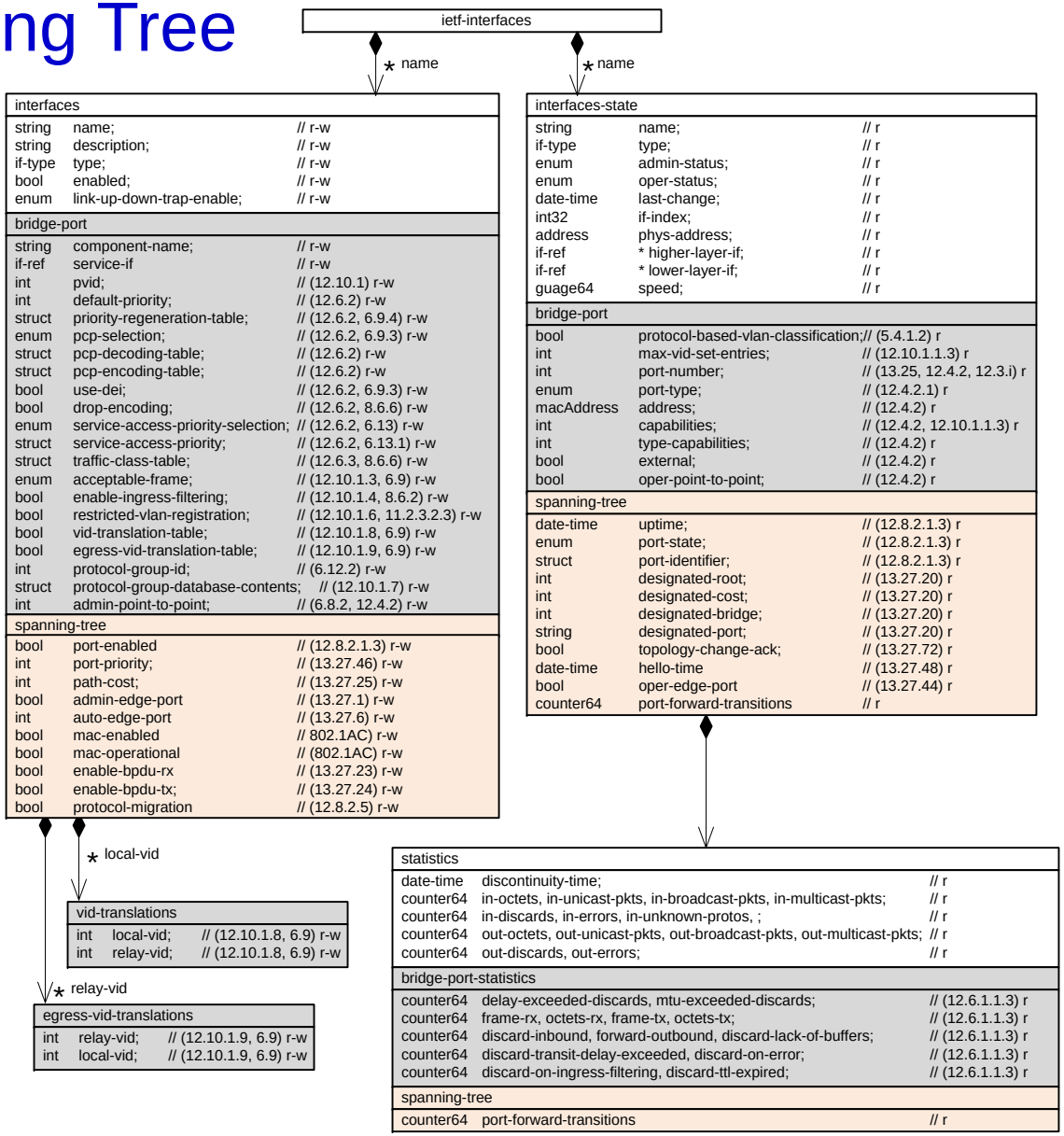
- IETF Interface Management Model (RFC 7223) can be represented using UML as shown below



The Bridge Port Interface Model

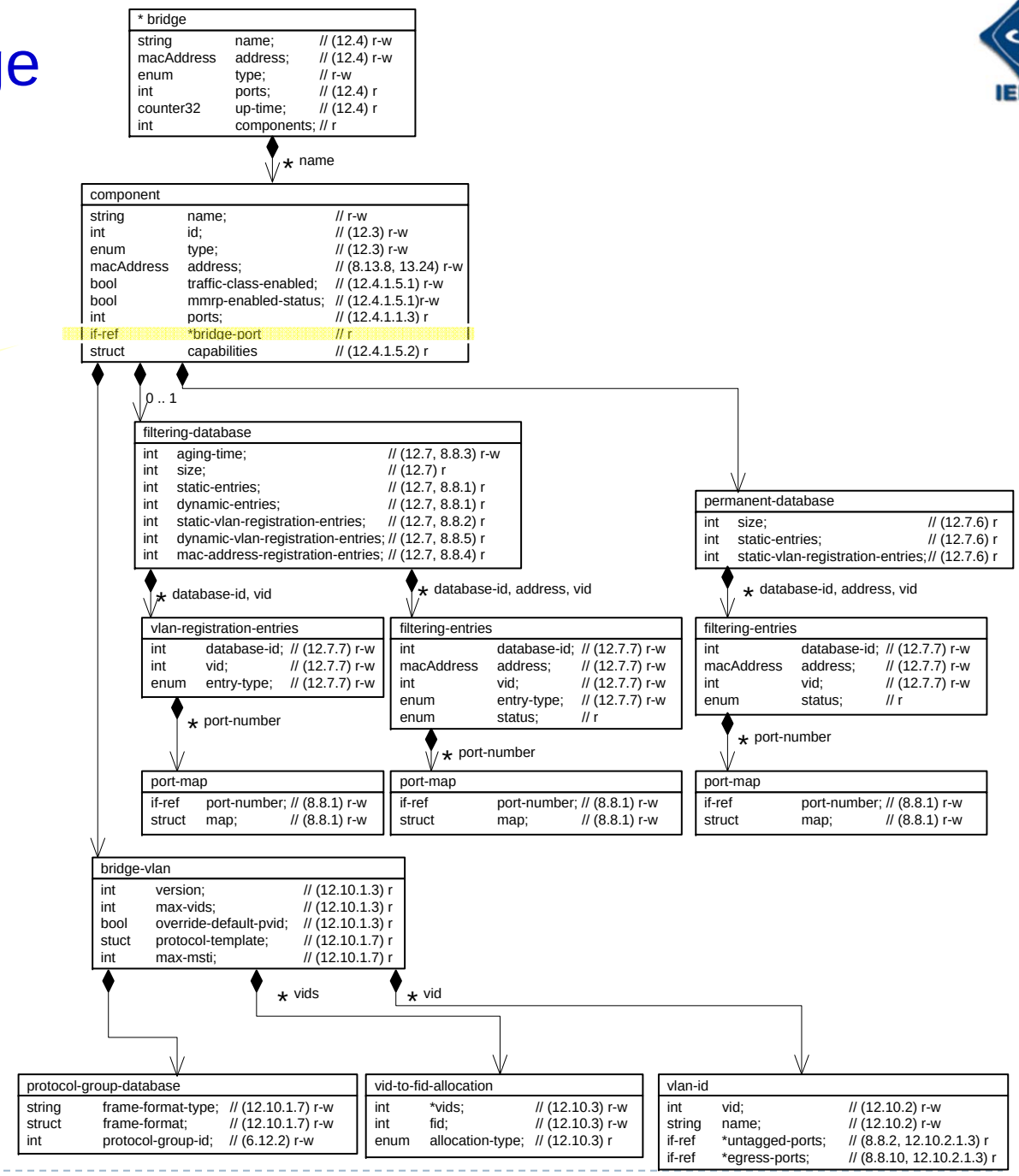


The Bridge Port Interface Model — Spanning Tree



The Generic Bridge YANG Model

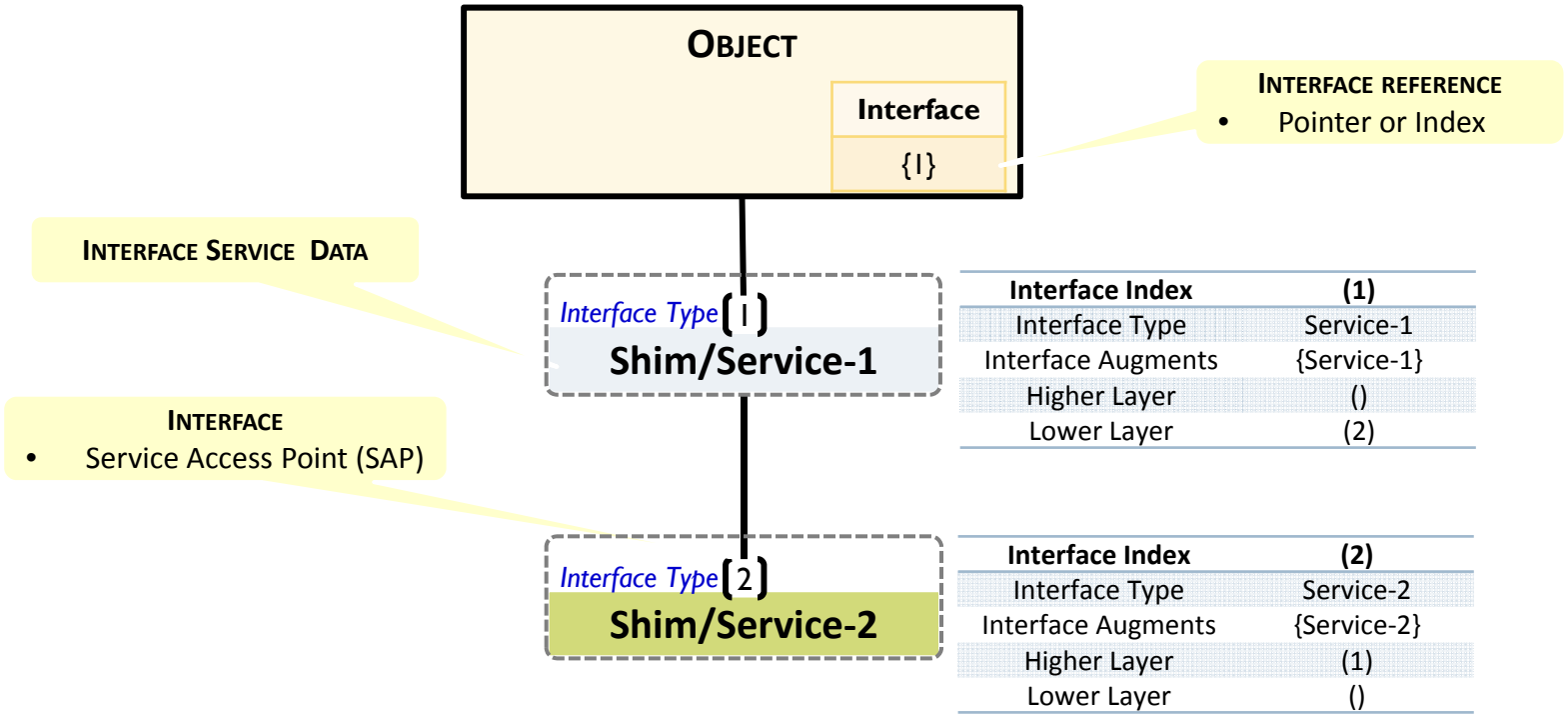
The list of Bridge Ports



Bridge Port Interface Stack Models

Interface Stack Diagram Representation

- A SAP is an abstraction and does not necessarily correspond to any concrete realization within a system
- The entities that support a particular SAP compose an interface stack
- Each YANG Interface definition contains an interface stack table

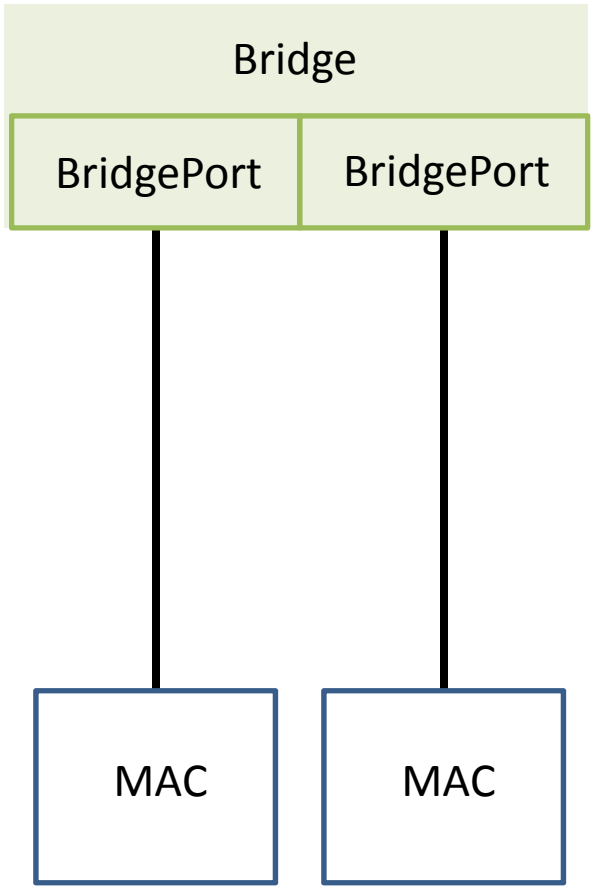


Interface Index	(1)
Interface Type	Service-1
Interface Augments	{Service-1}
Higher Layer	()
Lower Layer	(2)

Interface Index	(2)
Interface Type	Service-2
Interface Augments	{Service-2}
Higher Layer	(1)
Lower Layer	()

Basic Bridge

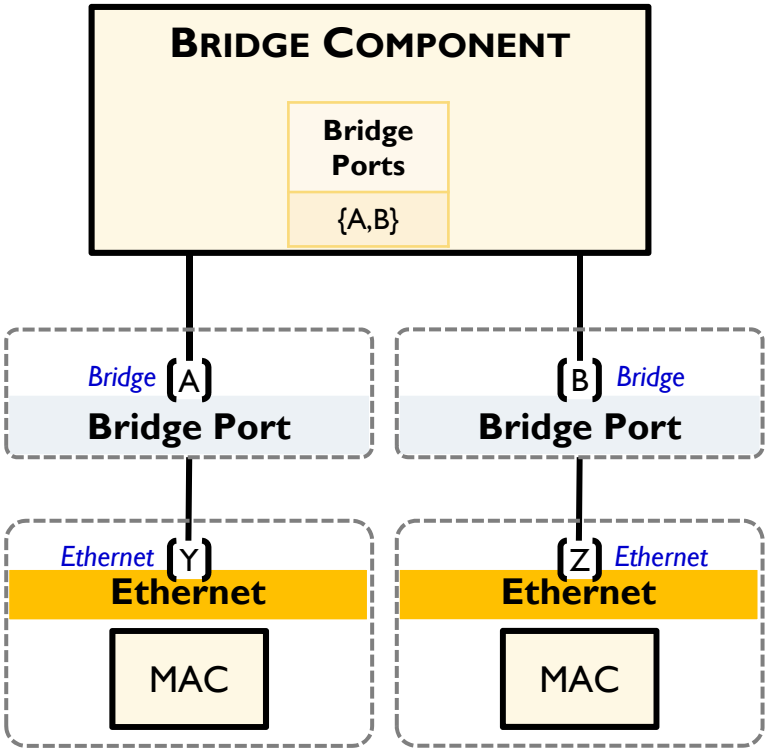
- Bridge Ports are associated with Bridge components



Basic Bridge (Port) Models

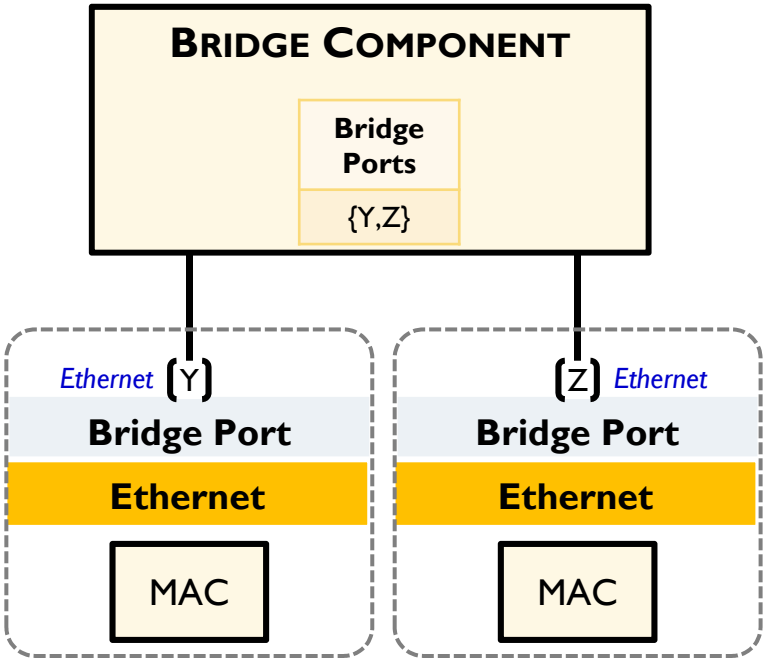
Model-1

- Bridge Ports are assigned to an Interface that is independent of the underlying MAC (or service)
- Bridge Port data and MAC data attributes are associated with separate Interfaces



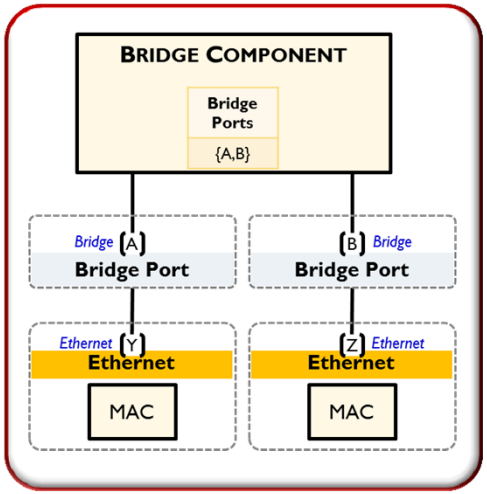
Model-2

- Bridge Ports are underlying MAC (or service) share the same Interface
- Bridge Port and MAC specific data attributes associated with same Interface



Model-1 Bridge Port — NETCONF Example

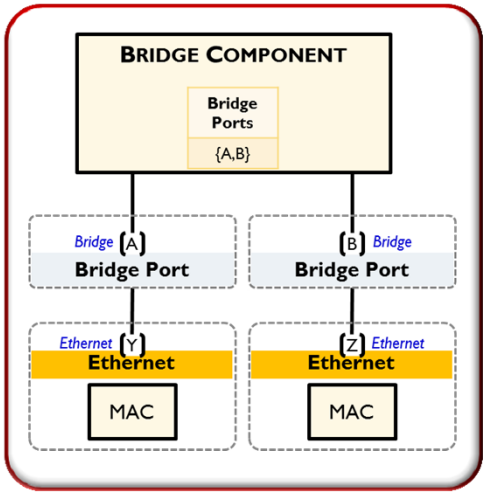
```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <bridges xc:operation="create">
          <bridge>
            <name>bridge-name</name>
            <component>
              <name>cvlan-comp</name>
                :
            </component>
              :
          </bridge>
          :
        </bridges>
```



Model-1 Bridge Port — NETCONF Example

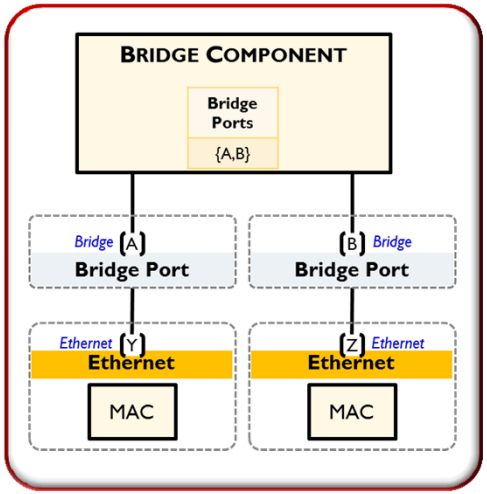
```
<interface xc:operation="create">
  <name>if-Y</name>
  <type>ethernetCsmacd</type>
  <ethernet-interface-attributes>
    :
  </ethernet-interface-attributes>
    :
</interface>
```

```
<interface xc:operation="create">
  <name>if-A</name>
  <type>bridge</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
    <service-if>if-Y</service-if>
    :
  </bridge-port>
    :
</interface>
```



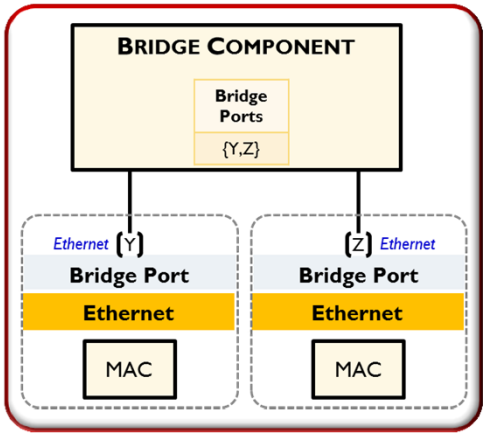
Model-1 Bridge Port — NETCONF Example

```
<interface xc:operation="create">
  <name>if-Z</name>
  <type>ethernetCsmacd</type>
  <ethernet-interface-attributes>
    :
  </ethernet-interface-attributes>
    :
</interface>

<interface xc:operation="create">
  <name>if-B</name>
  <type>bridge</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
    <service-if>if-Z</service-if>
    :
  </bridge-port>
    :
</interface>

    </top>
  </config>
</edit-config>
</rpc>
```



Model-2 Bridge Port — NETCONF Example

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <bridges xc:operation="create">
          <bridge>
            <name>bridge-name</name>
            <component>
              <name>cvlan-comp</name>
                :
            </component>
              :
          </bridge>
           :
        </bridges>
```

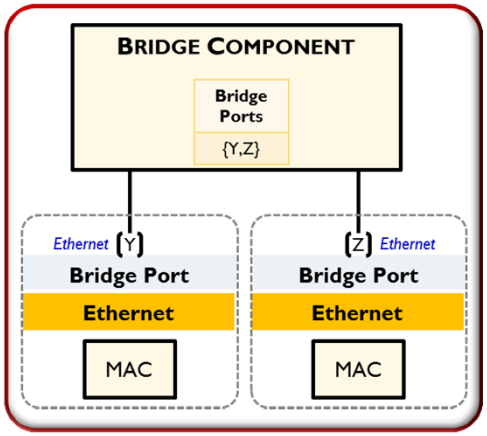


Model-2 Bridge Port — NETCONF Example

```
<interface xc:operation="create">
  <name>if-Y</name>
  <type>ethernetCsmacd</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
      :
  </bridge-port>
  <ethernet-interface-attributes>
    :
  </ethernet-interface-attributes>
    :
</interface>
```

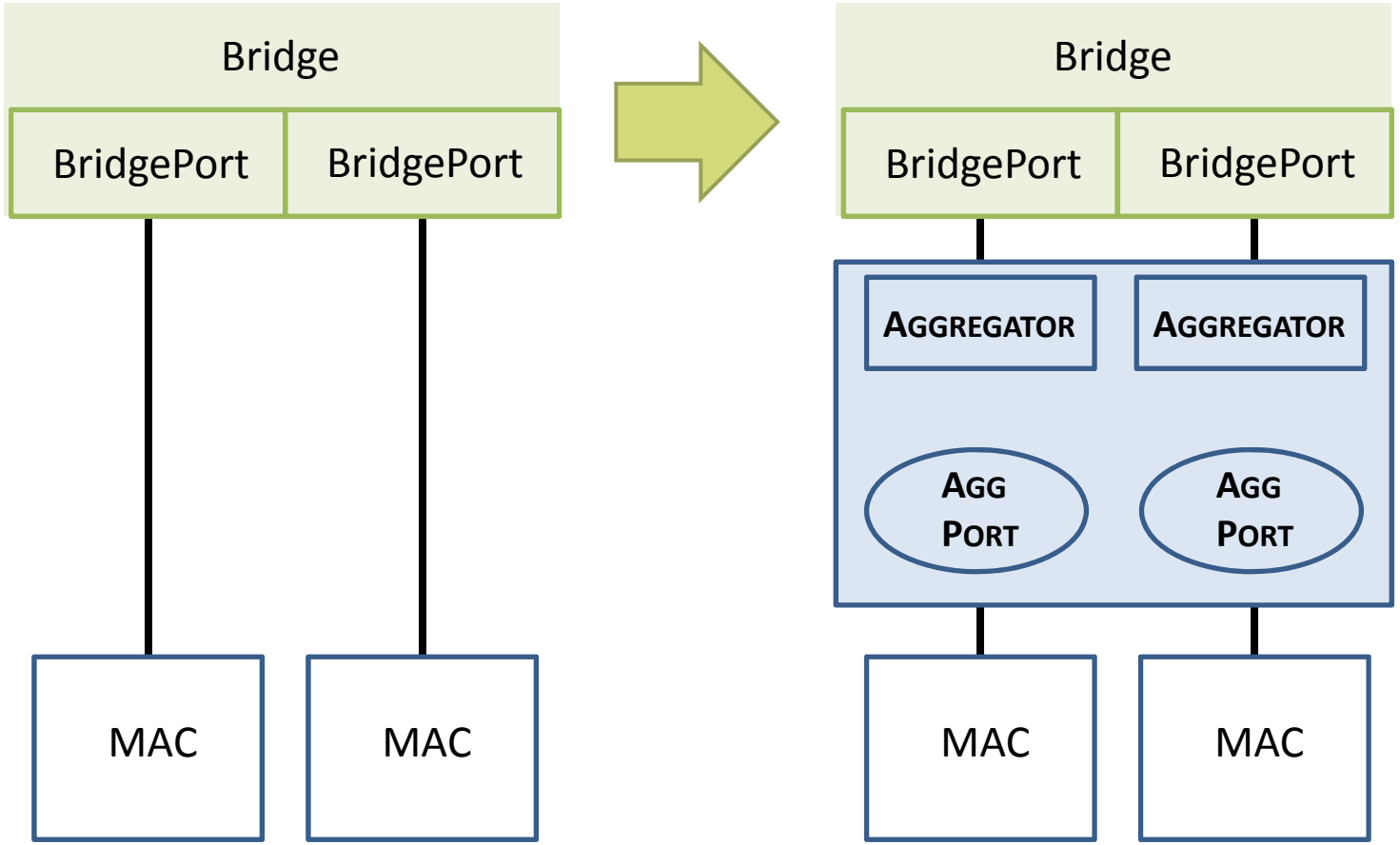
```
<interface xc:operation="create">
  <name>if-Z</name>
  <type>ethernetCsmacd</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
      :
  </bridge-port>
  <ethernet-interface-attributes>
    :
  </ethernet-interface-attributes>
    :
</interface>
```

```
    </top>
  </config>
</edit-config>
</rpc>
```

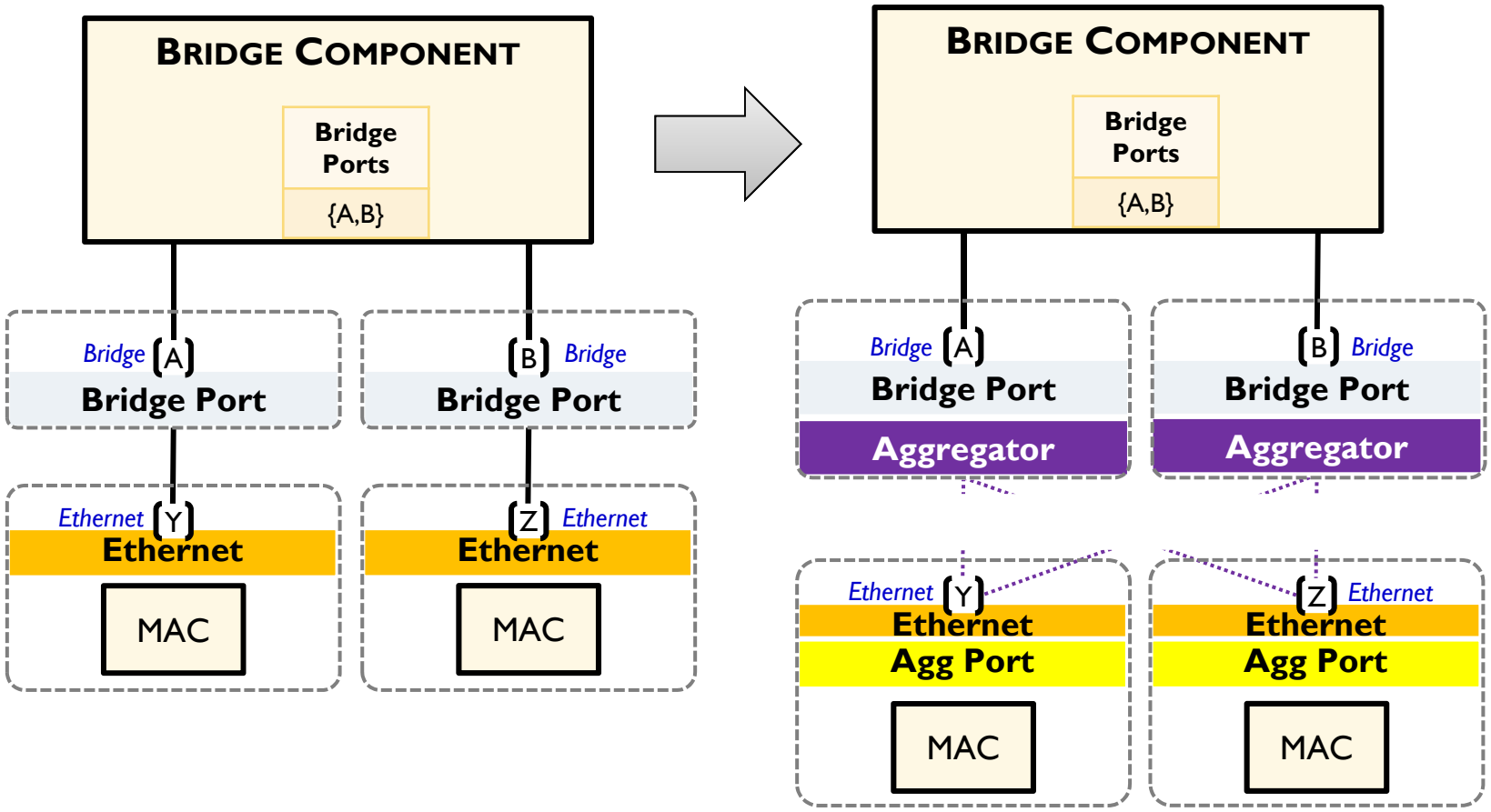


Basic Bridge with Link Aggregation

- "Agg Port" specific configuration gets applied
- "Aggregator" specific configuration gets applied
- System then utilizes LACP signaling to tie together aggregation members with the Aggregator



Evolution of Bridge Port Model-1 — Link Aggregation

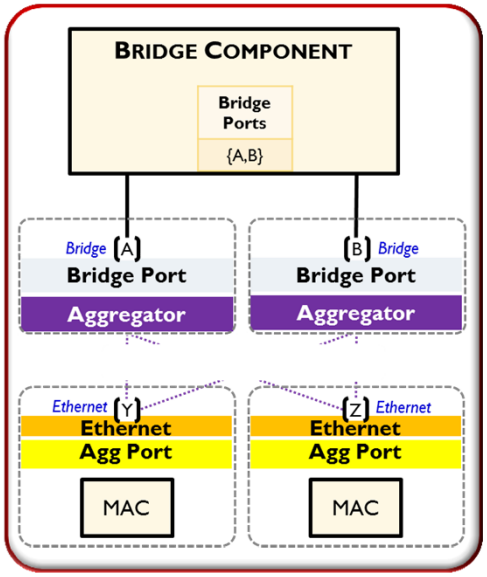


Evolution of Bridge Port Model-1 — Link Aggregation

- The Bridge Port Interfaces (A) and (B) are extended to include AGGREGATOR specific configuration and operational data
- Interface (Y) and (Z) will be extended to include AGGREGATION PORT specific configuration and operational data
  - From a YANG perspective, this is an augmentation
- LACP operation determines which AGGREGATION PORT points to which AGGREGATOR

Model-1 Bridge Port — NETCONF Example

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <interface xc:operation="merge">
          <name>if-Y</name>
          <agg-port-interface-attributes>
            :
          </agg-port-interface-attributes>
        </interface>

        <interface xc:operation="merge">
          <name>if-Z</name>
          <agg-port-interface-attributes>
            :
          </agg-port-interface-attributes>
        </interface>
```

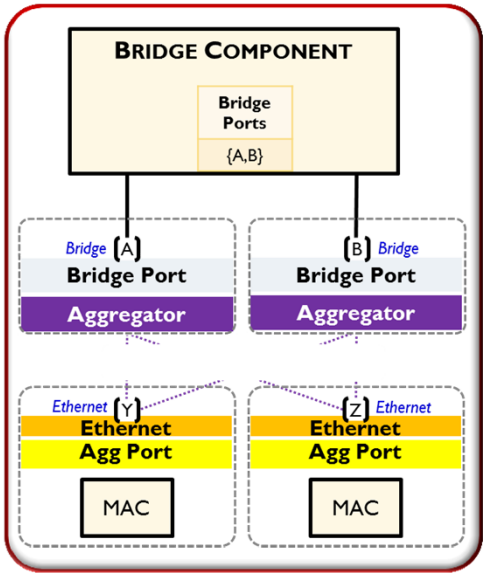


Model-1 Bridge Port — NETCONF Example

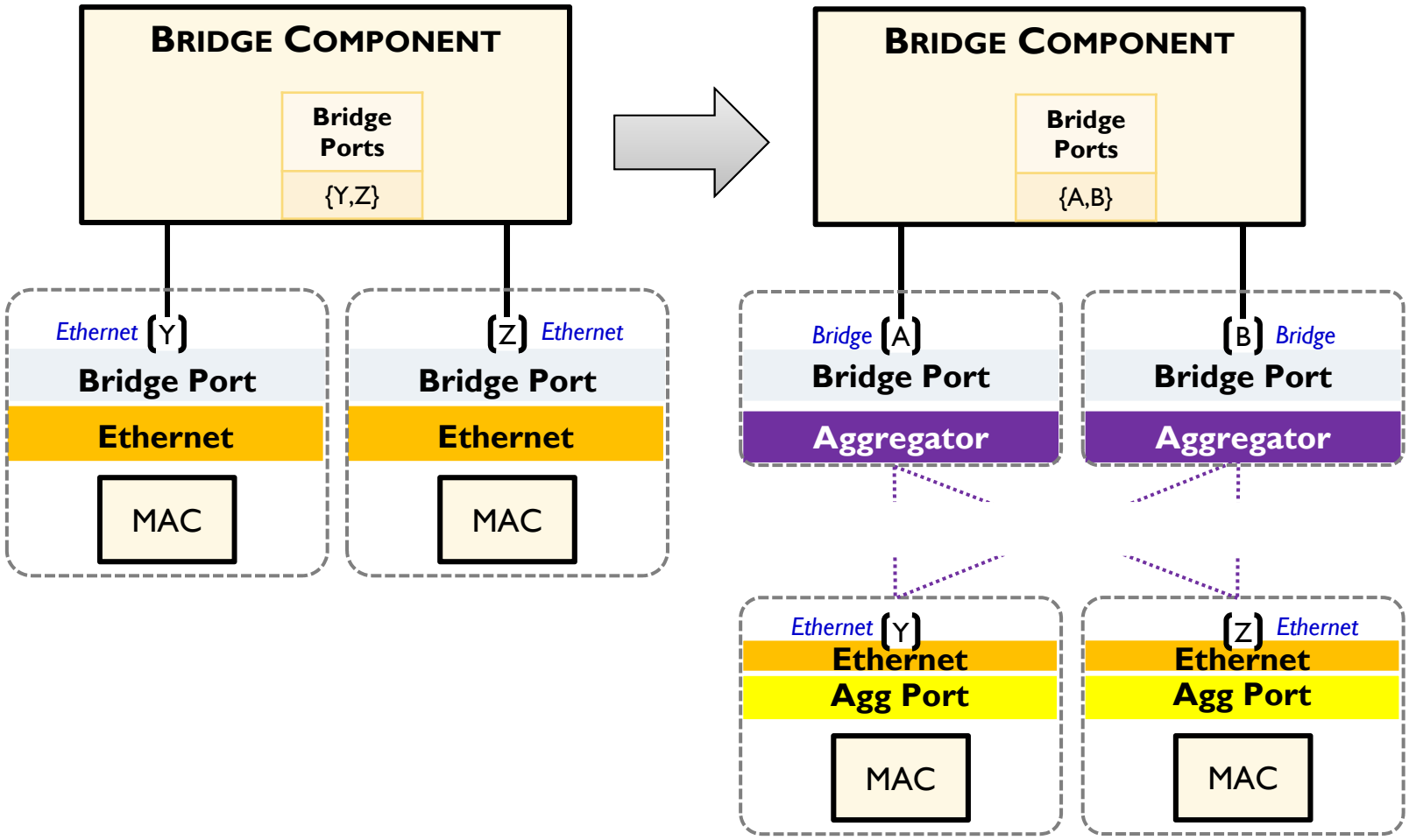
```
<interface xc:operation="merge">
  <name>if-A</name>
  <aggregator-interface-attributes>
    :
  </aggregator-interface-attributes>
</interface>
```

```
<interface xc:operation="merge">
  <name>if-B</name>
  <aggregator-interface-attributes>
    :
  </aggregator-interface-attributes>
</interface>
```

```
    </top>
  </config>
</edit-config>
</rpc>
```



Evolution of Bridge Port Model-2 — Link Aggregation

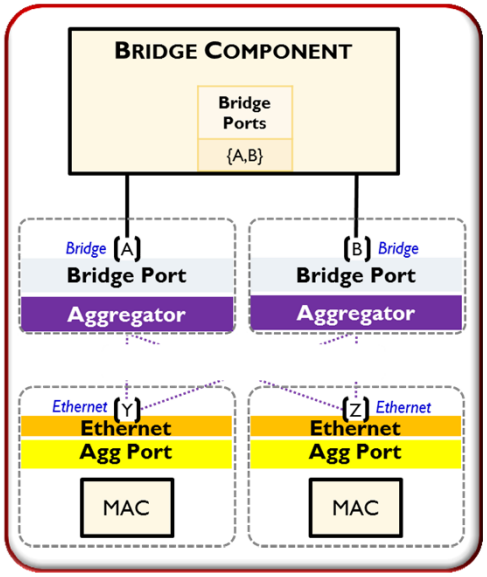


Evolution of Bridge Port Model-2 — Link Aggregation

- Bridge Port data is removed from Interfaces (Y) and (Z)
  - Interface (Y) and (Z) will be extended to include AGGREGATION PORT specific configuration and operational data
- Interface (A) and (B) are created
  - Interface (A) and (B) will be extended to include AGGREGATOR specific and BRIDGE PORT configuration and operational data
- LACP operation determines which AGGREGATION PORT points to which AGGREGATOR
- **NOTE**: Moving (i.e., deleting and recreating) the Bridge Port [configuration and operational] data from Interfaces (Y) and (Z), to newly created Interfaces (A) and (B) may result in loss of operational state information

Model-2 Bridge Port — NETCONF Example

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <interface>
          <name>if-Y</name>
          <bridge-port xc:operation="delete">
            <port-number>Y</port-number>
          </bridge-port>
          <agg-port-interface-attributes xc:operation="merge">
            :
          </agg-port-interface-attributes>
        </interface>

        <interface>
          <name>if-Z</name>
          <bridge-port xc:operation="delete">
            <port-number>Z</port-number>
          </bridge-port>
          <agg-port-interface-attributes xc:operation="merge">
            :
          </agg-port-interface-attributes>
        </interface>
```



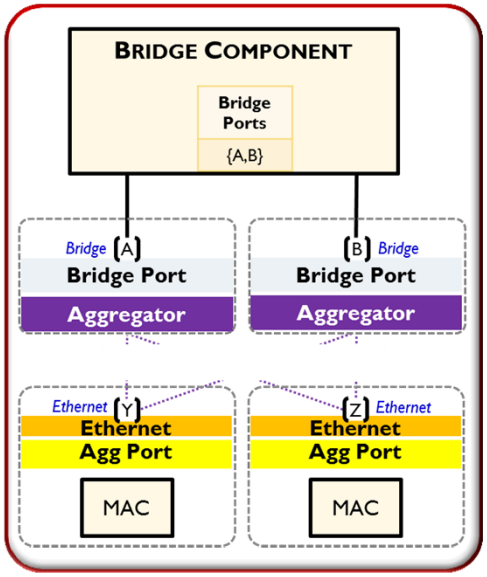
Model-2 Bridge Port — NETCONF Example

```
<interface xc:operation="create">
  <name>if-A</name>
  <type>ieee8023adLag</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
      :
  </bridge-port>
  <aggregator-interface-attributes>
    :
  </aggregator-interface-attributes>
</interface>
```

```
<interface xc:operation="create">
  <name>if-B</name>
  <type>ieee8023adLag</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
      :
  </bridge-port>
  <aggregator-interface-attributes>
    :
  </aggregator-interface-attributes>
</interface>
```

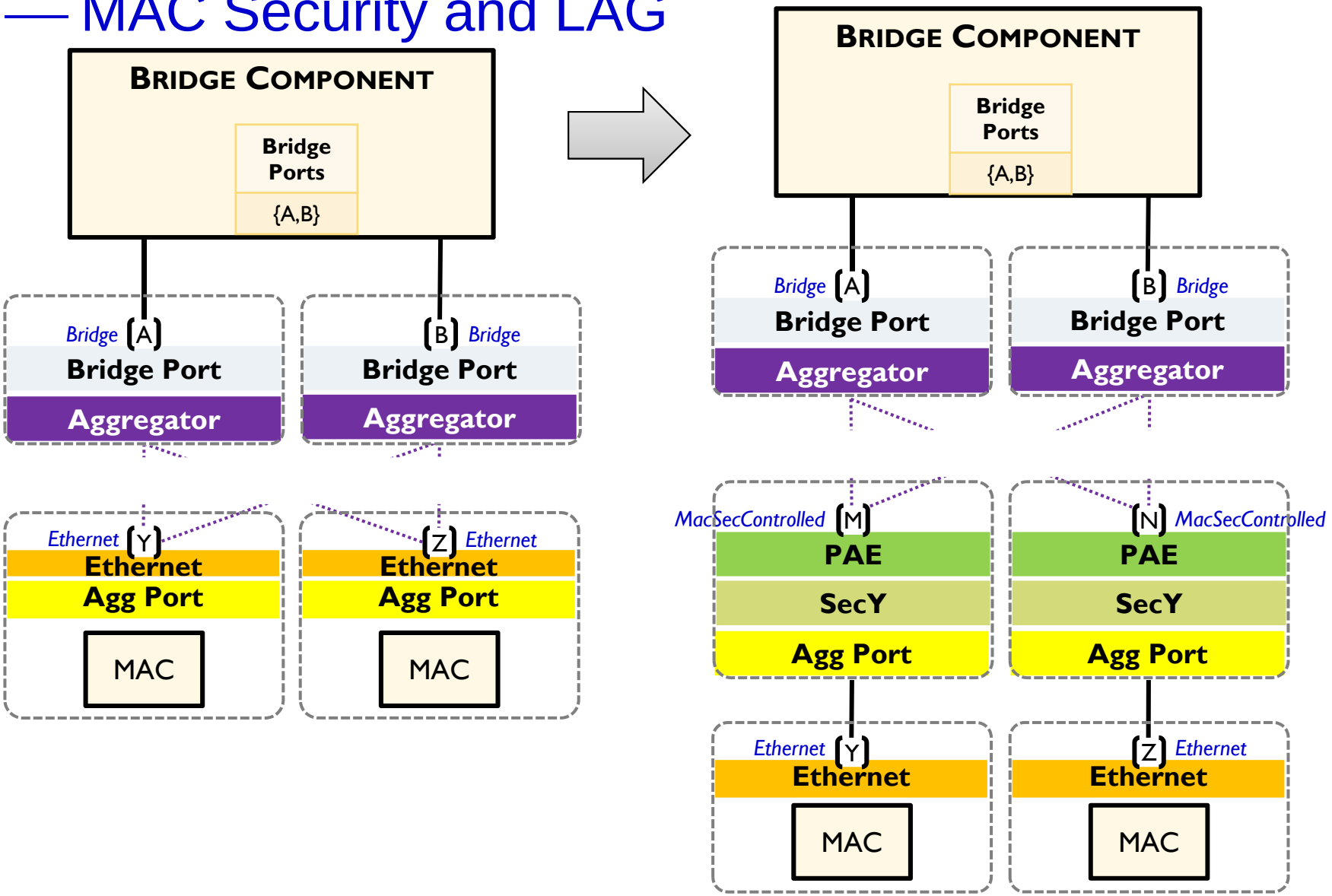
```
    </top>
  </config>
</edit-config>
</rpc>
```

NOTE: Bridge port configuration and operational data attributes removed from Interface (Y) needs to be [somehow] preserved and set on Interface (A)

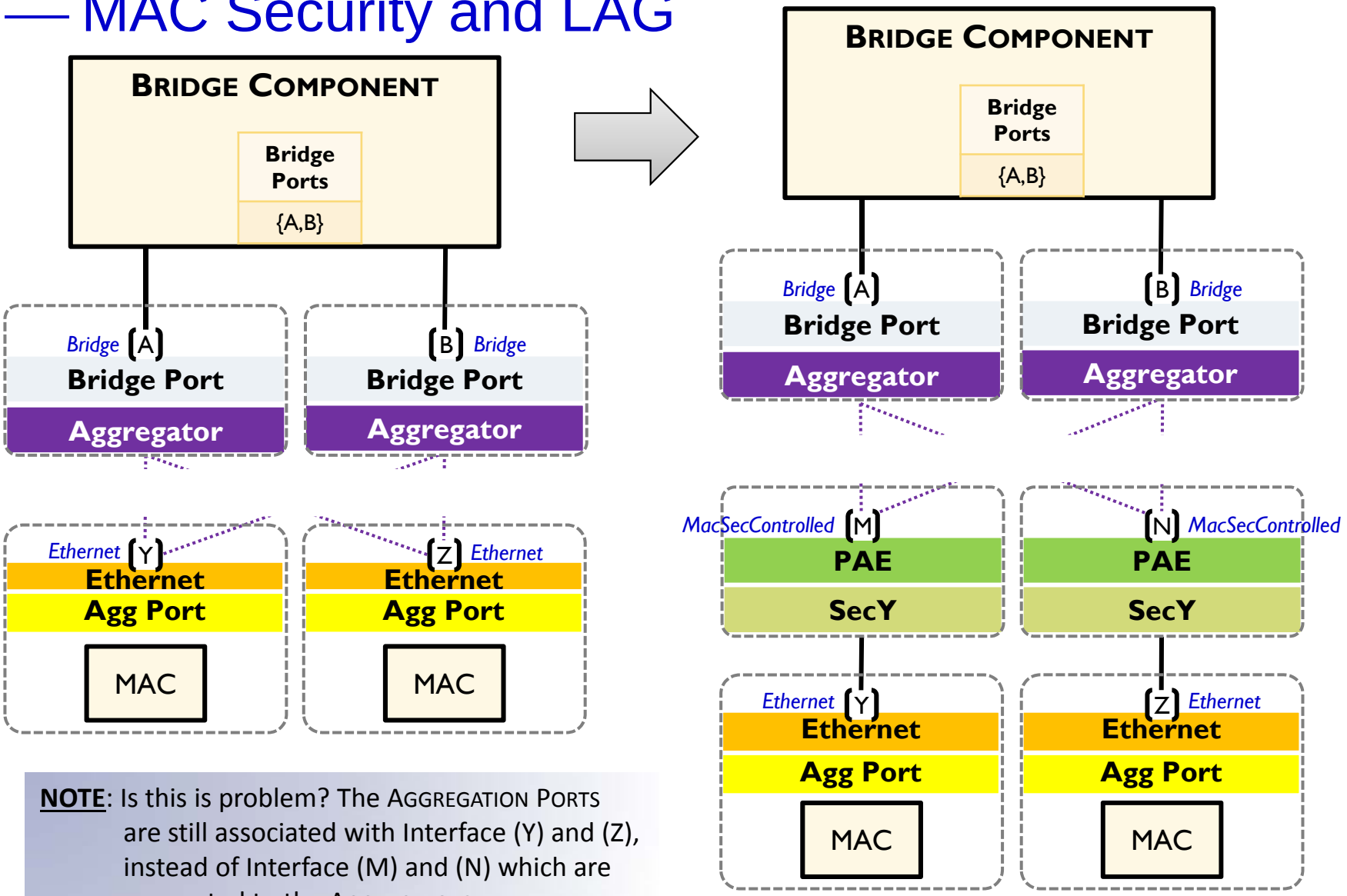


NOTE: Bridge port configuration and operational data attributes removed from Interface (Z) needs to be [somehow] preserved and set on Interface (B)

Evolution of Bridge Port Model — MAC Security and LAG



Evolution of Bridge Port Model — MAC Security and LAG

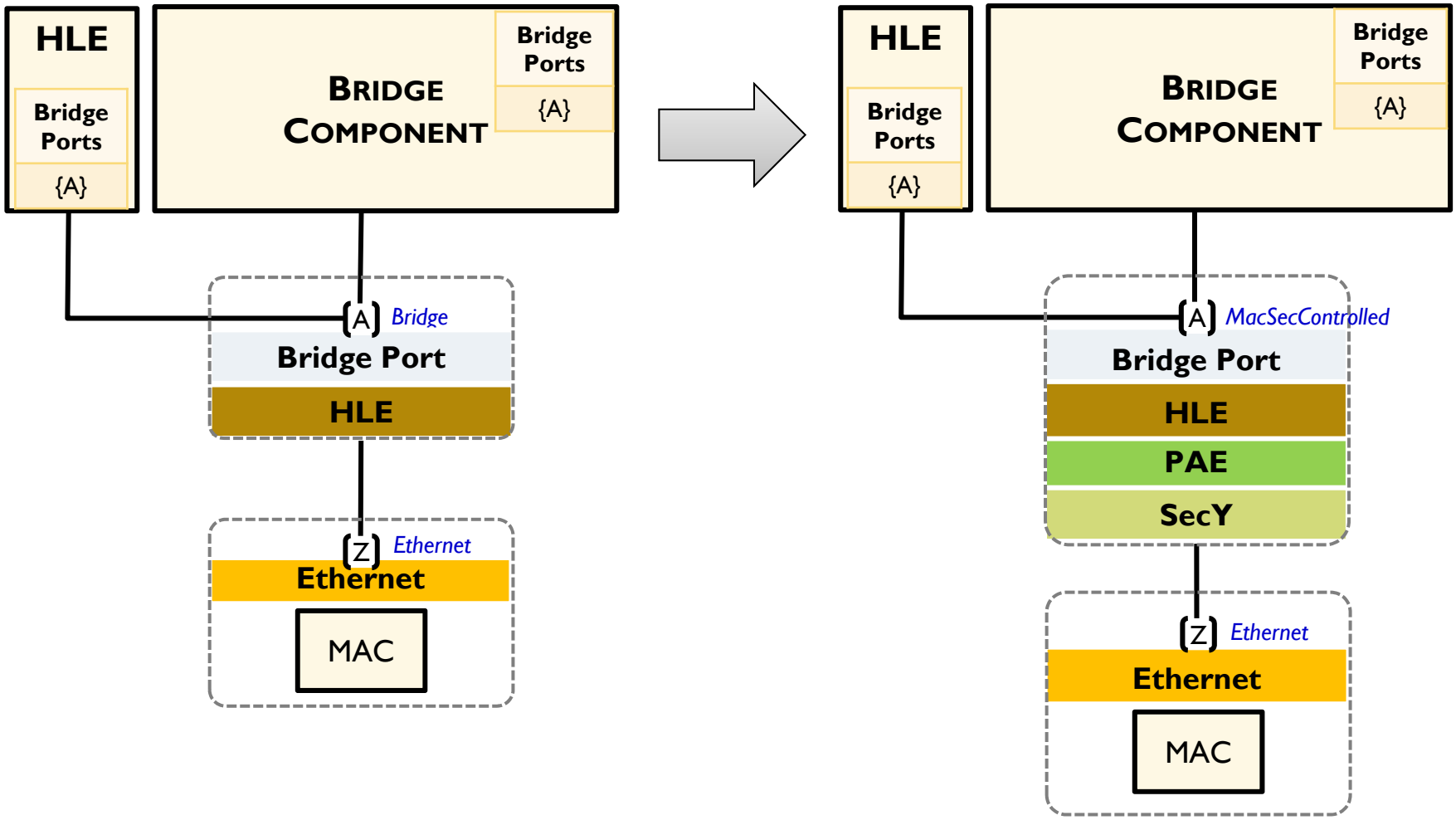


NOTE: Is this is problem? The AGGREGATION PORTS are still associated with Interface (Y) and (Z), instead of Interface (M) and (N) which are connected to the AGGREGATORS.

Evolution of Bridge Port Model — MAC Security and LAG

- A PAE (service shim) is inserted in the Interface stack
  - The PAE/SECY/PAC is associated with Interface (M) and (N)

Evolution of Bridge Port Model-1 — MACSec and HLE

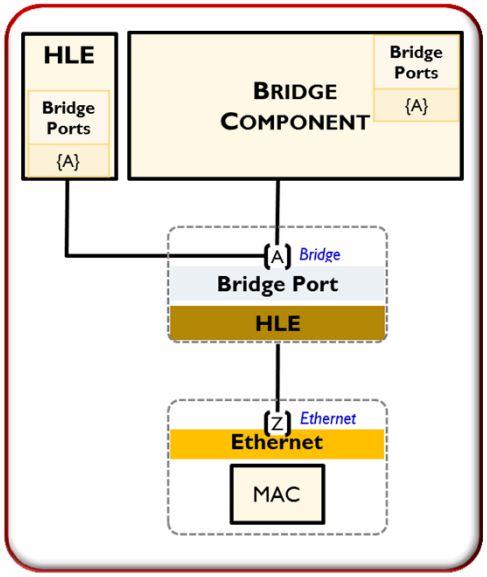


Evolution of Bridge Port Model-1 — MACSec and HLE

- Higher Layer Entity (HLE) references Bridge Port Interface (e.g., (A))
- HLE specific configuration and operational data extends Bridge Port Interface (A) data
- PAE MACSec configuration and operational data extends Bridge Port Interface (A)

Model-1 Bridge Port — NETCONF Example

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <bridges xc:operation="create">
          <bridge>
            <name>bridge-name</name>
            <component>
              <name>cvlan-comp</name>
                :
            </component>
              :
          </bridge>
          :
        </bridges>

        <HLE xc:operation="create">
          :
        </HLE>
```

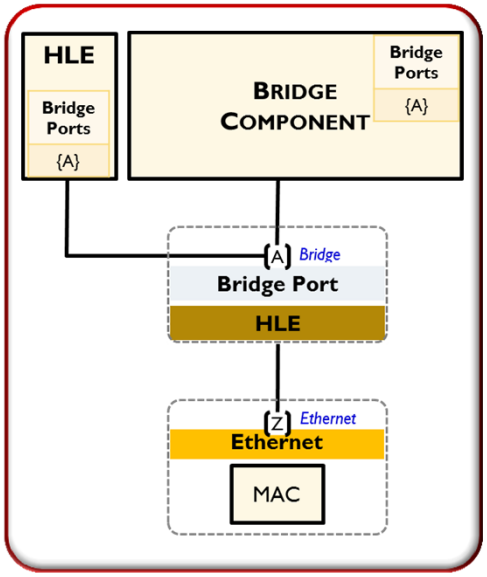


Model-1 Bridge Port — NETCONF Example

```
<interface xc:operation="create">
  <name>if-Z</name>
  <type>ethernetCsmacd</type>
  <ethernet-interface-attributes>
    :
  </ethernet-interface-attributes>
    :
</interface>
```

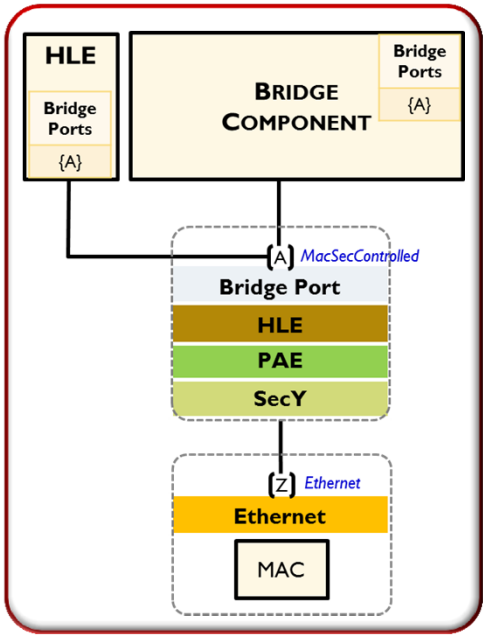
```
<interface xc:operation="create">
  <name>if-A</name>
  <type>bridge</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
    <service-if>if-Z</service-if>
    :
  </bridge-port>
  <HLE-interface-attributes>
    :
  </HLE-interface-attributes>
    :
</interface>
```

```
    </top>
  </config>
</edit-config>
</rpc>
```

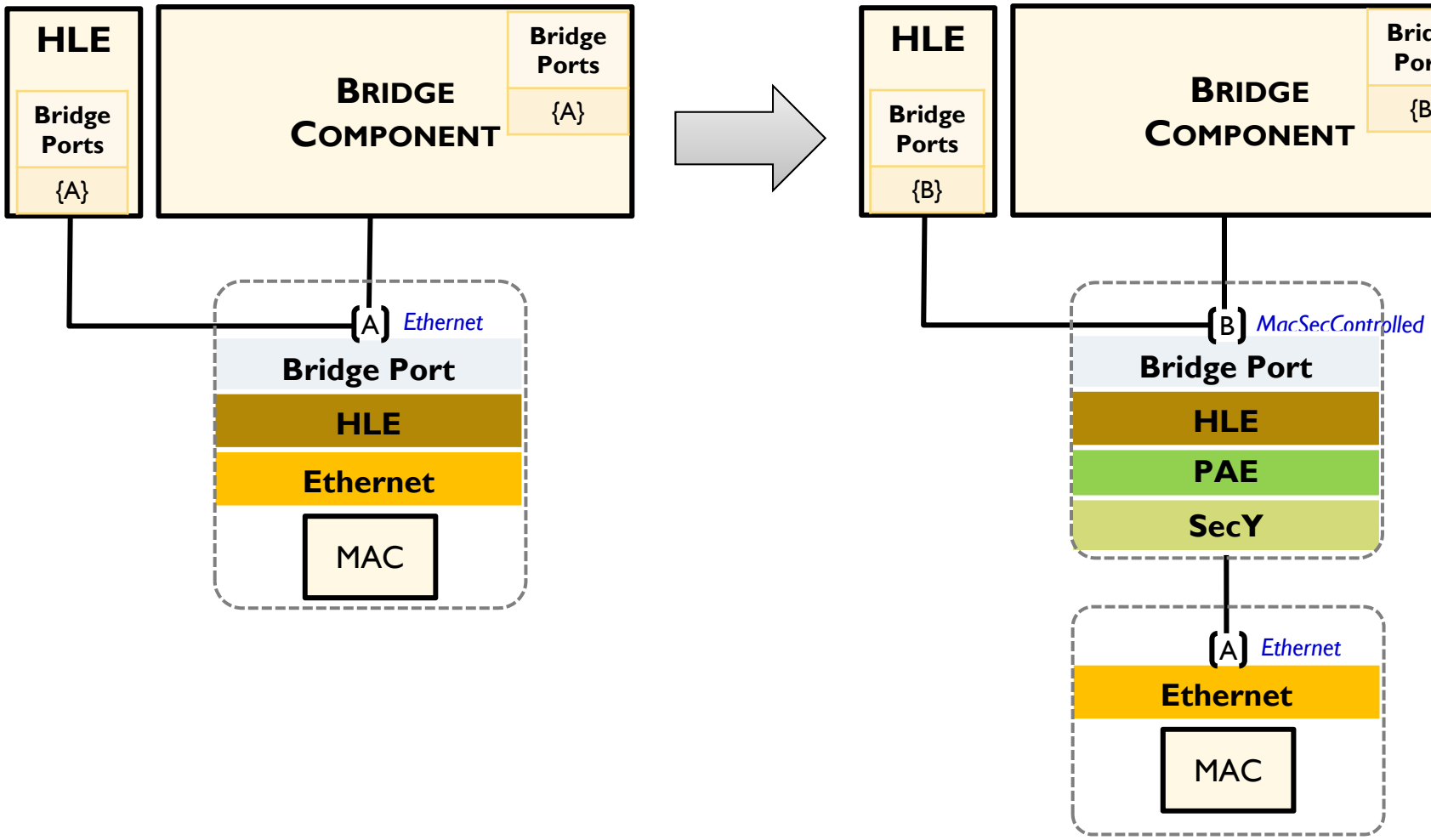


Model-1 Bridge Port — NETCONF Example

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <interface>
          <name>if-A</name>
          <pae>
            :
          </pae>
        </interface>

      </top>
    </config>
  </edit-config>
</rpc>
```



Evolution of Bridge Port Model-2 — MACSec and HLE



Evolution of Bridge Port Model-2 — MACSec and HLE

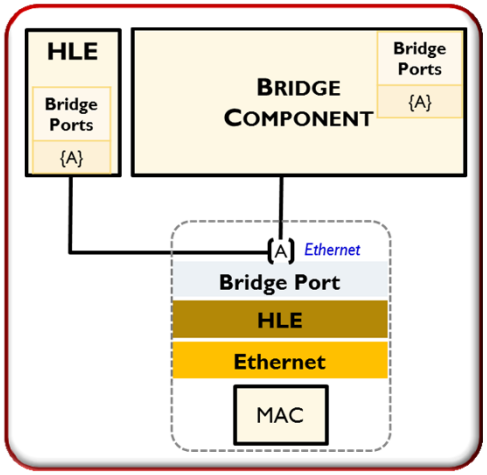
- The Bridge Port and HLE data is removed from Interface (A)
- Interface (B) is created
  - The Bridge Port and HLE data attributes are added to Interface (B)
  - MACSec configuration (and operational) data is added to Interface (B)

- The Bridge Port pointers within the Component and HLE needs to change

- **NOTE**: Moving (i.e., deleting and recreating) the Bridge Port and HLE (e.g., Spanning Tree) [configuration and operational] data from Interfaces (A), to newly created Interfaces (B) may result in loss of operational state information

Model-2 Bridge Port — NETCONF Example

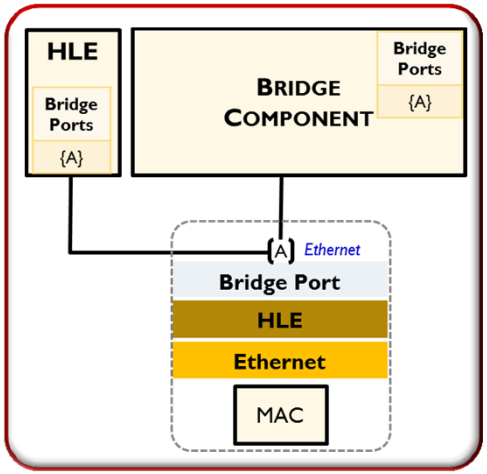
```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <bridges xc:operation="create">
          <bridge>
            <name>bridge-name</name>
            <component>
              <name>cvlan-comp</name>
                :
            </component>
              :
          </bridge>
           :
        </bridges>

        <HLE xc:operation="create">
          :
        </HLE>
```



Model-2 Bridge Port — NETCONF Example

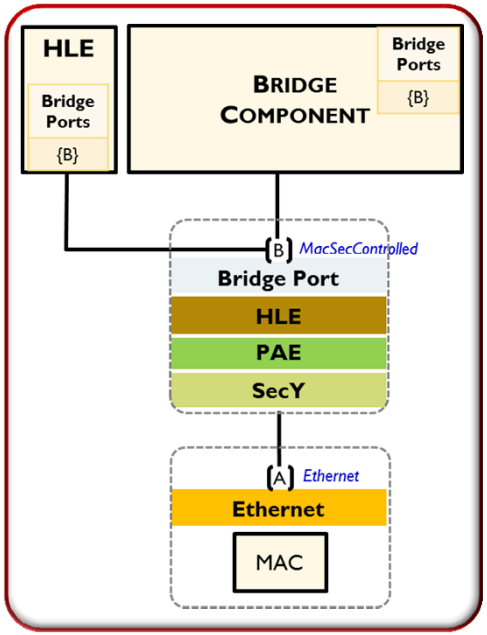
```
<interface xc:operation="create">
  <name>if-A</name>
  <type>ethernetCsmacd</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
      :
  </bridge-port>
  <HLE-interface-attributes>
    :
  </HLE-interface-attributes>
  <ethernet-interface-attributes>
    :
  </ethernet-interface-attributes>
      :
</interface>
```

```
    </top>
  </config>
</edit-config>
</rpc>
```



Model-2 Bridge Port — NETCONF Example

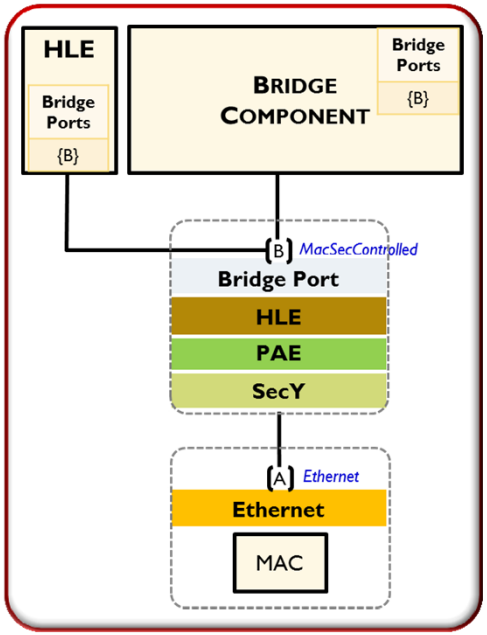
```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <top xmlns="http://example.com/schema/1.2/config">

        <interface>
          <name>if-A</name>
          <bridge-port xc:operation="delete">
            <port-number>A</port-number>
          </bridge-port>
          <HLE-interface-attributes xc:operation="delete">
            :
          </HLE-interface-attributes>
        </interface>
```



Model-2 Bridge Port — NETCONF Examp

```
<interface xc:operation="create">
  <name>if-B</name>
  <type>ethernetCsmacd</type>
  <bridge-port>
    <component-name>cvlan-comp</component-name>
      :
  </bridge-port>
  <HLE-interface-attributes>
    :
  </HLE-interface-attributes>
  <pae>
    :
  </pae>
</interface>

    </top>
  </config>
</edit-config>
</rpc>
```

NOTE: Bridge port configuration and operational data attributes removed from Interface (A) needs to be [somehow] preserved and set on Interface (B)



NOTE: PAE (Port Access Entity) configuration and operational data attributes removed from Interface (A) needs to be [somehow] preserved and set on Interface (B)

Conclusions/Observations and Recommendations

Conclusions/Observations

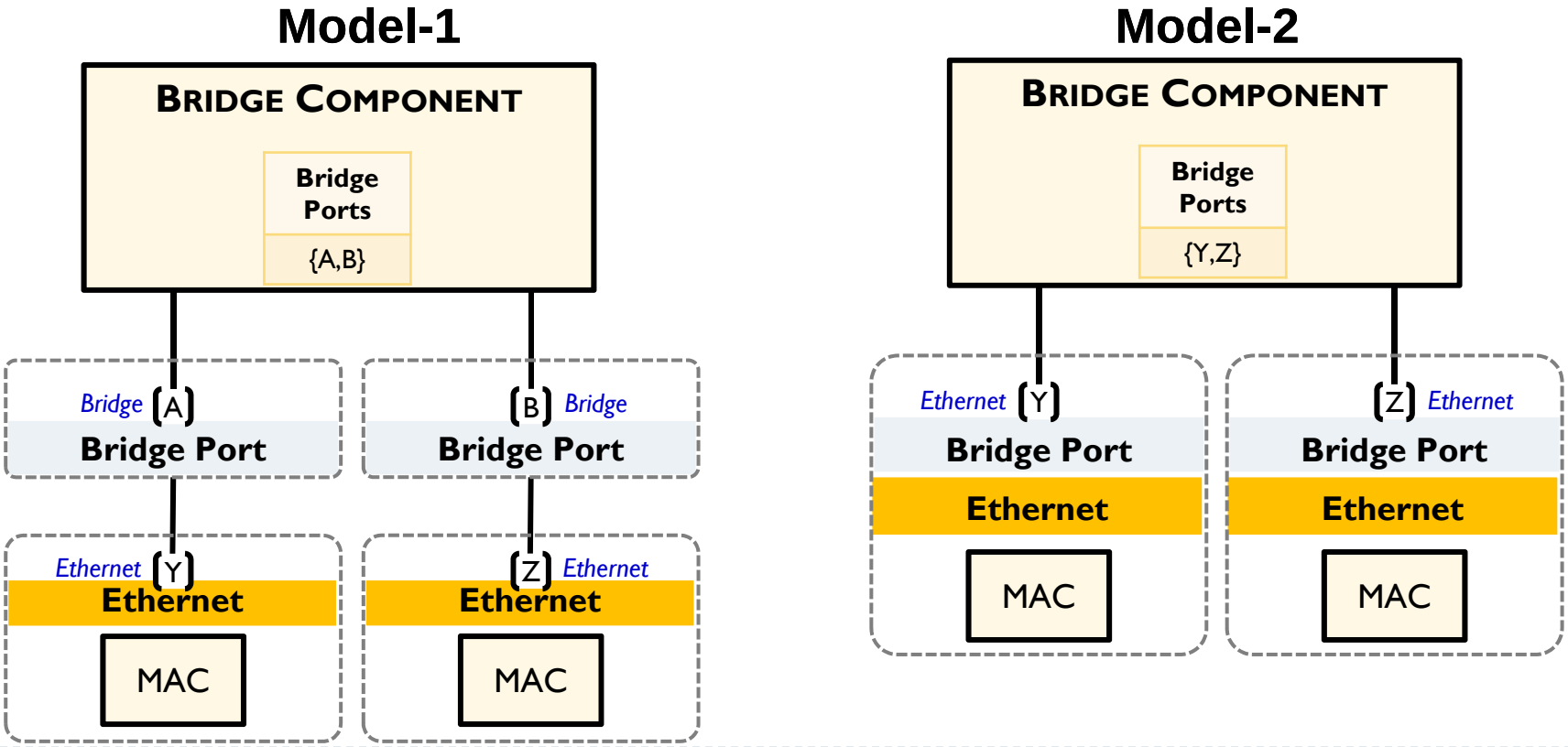
- The Bridge Port provides an interface to both the MAC Relay Entity as well as higher layer entities (e.g., spanning tree, IP router, end stations, etc.)
  - Consequently, [configuration and operational] data associated HLEs as well as the MAC relay entity are associated with a Bridge Port

- Additionally, other 802.1Q-2014 features contributes to the [configuration and operational] data associated with a Bridge Port
  - Congestion Notification (CN), for congestion aware end stations or Bridge components
  - Stream Reservation Protocol (SRP)
  - Edge Control Protocol (ECP)

- Consequently, care should be taken to preserve Bridge Port (and HLE) [configuration and operational data when protocol entities are introduced within the Bridge Port (interface stack)

Recommendation

- Both models (Model-1 and Model-2) have pros and cons, and can work
  - It is a matter of perspective and the device that the model may be realized on
- The Bridge Port YANG model (used by the 802.1Q Bridge) will be developed to accommodate both Model-1 and Model-2



Backup Material

Bridge Port Interface

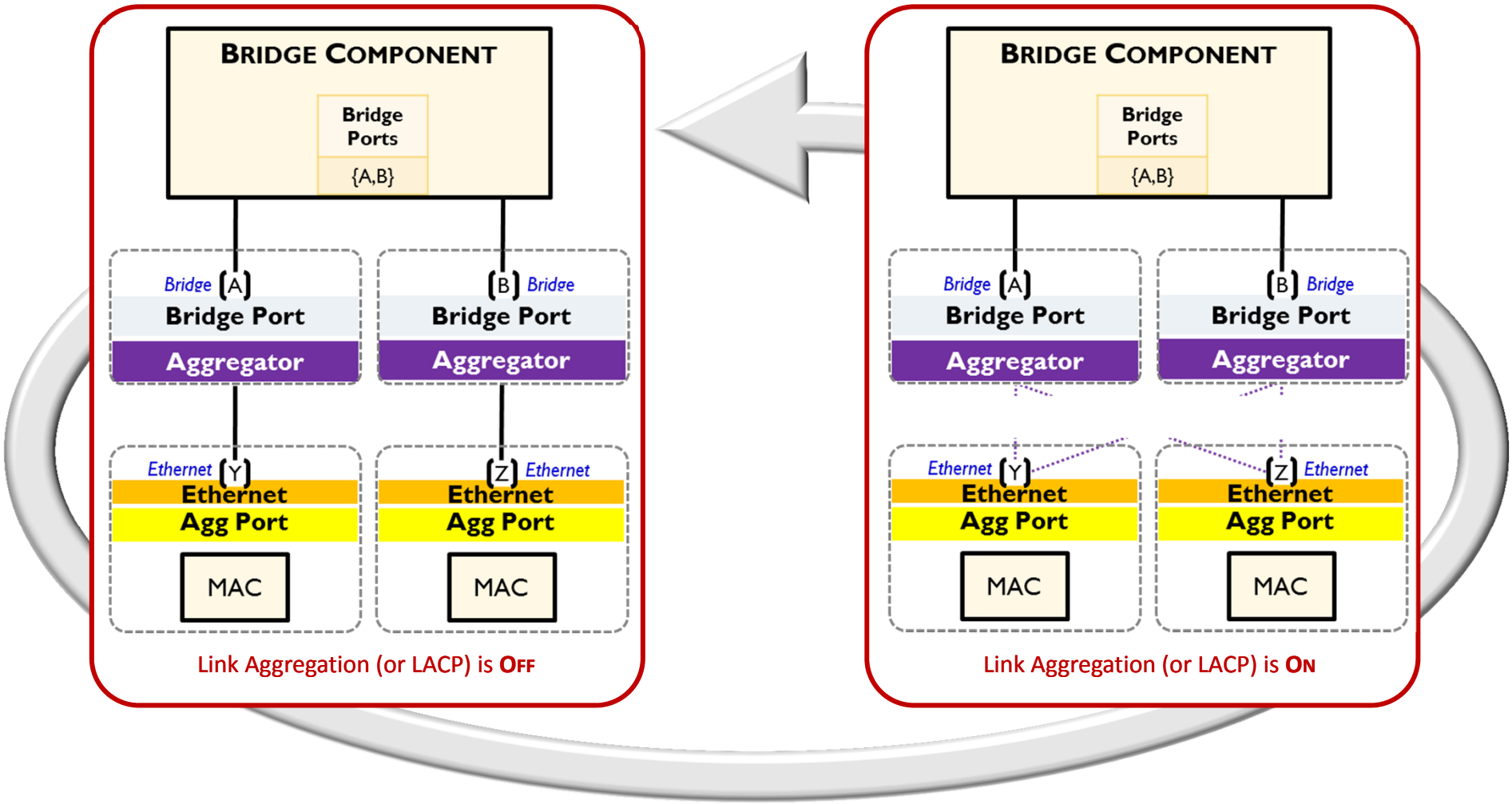
YANG Model Enhancement

```
augment "/if:interfaces/if:interface" {
  when "/if:type = 'ianaif:bridge' or
        /if:type = 'ianaif:ethernetCsmacd' or
        /if:type = 'ianaif:ieee8023adLag'" {
    description
      "Applies when a Bridge Port, or an underlying Ethernet MAC or LAG.";
    }
  description
    "Augment the Interface model with the Bridge Port.";
  container bridge-port {
    description
      "Bridge Port is an extension of the IETF Interfaces model (RFC7223).";
    :
    :
  }

augment "/if:interfaces-state/if:interface" {
  when "/if:type = 'ianaif:bridge' or
        /if:type = 'ianaif:ethernetCsmacd' or
        /if:type = 'ianaif:ieee8023adLag'" {
    description
      "Applies when a Bridge Port, or an underlying Ethernet MAC or LAG.";
    }
  description
    "Augment the Interface model with the Bridge Port.";
  container bridge-port-state {
    description
      "Bridge Port is an extension of the IETF Interfaces model (RFC7223).";
    :
    :
  }
```

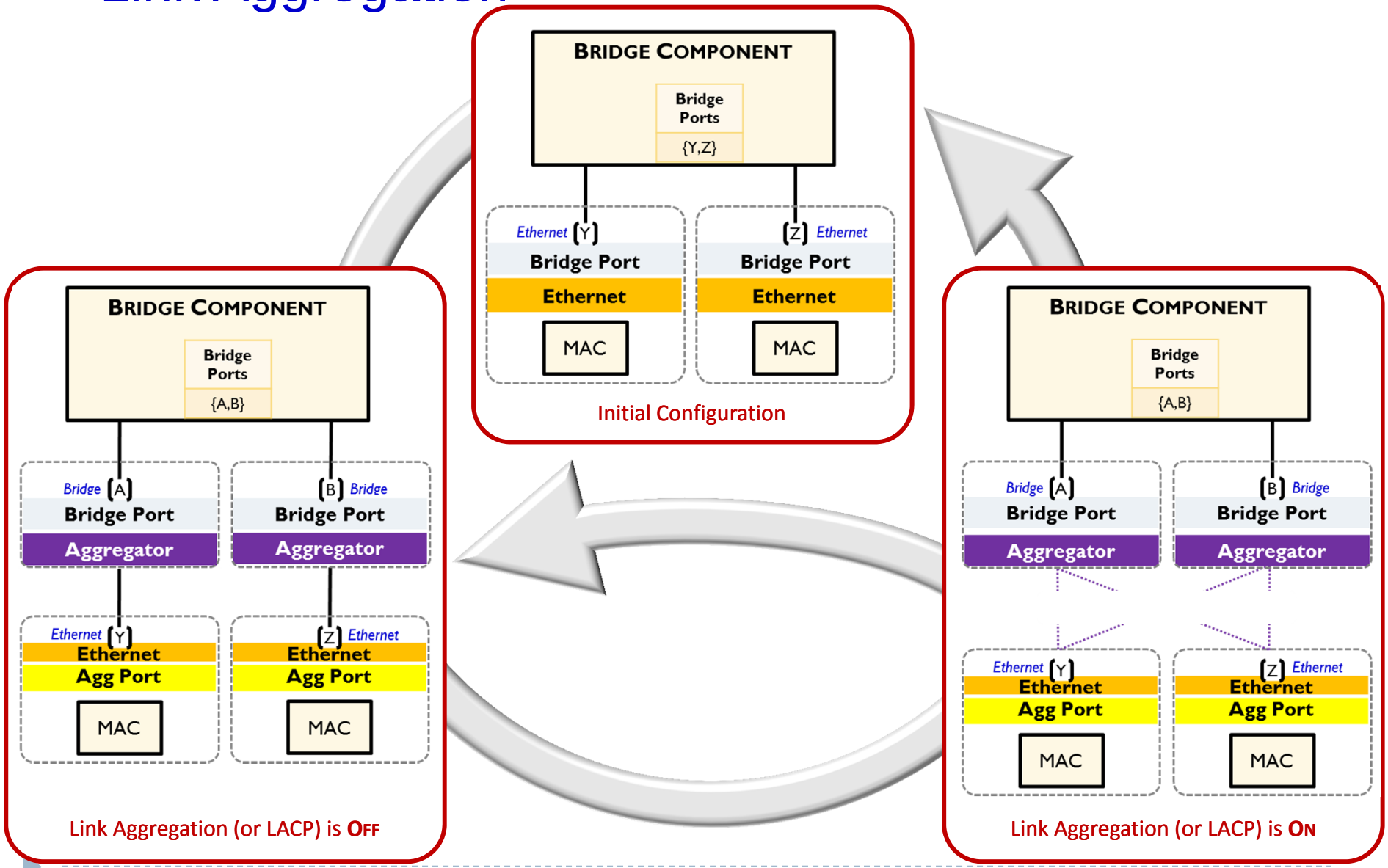
YANG Model Excerpt

```
feature simple-bridge-port {
   description
     "A simple bridge port allows underlying (MAC) layers to share
     the same Interface as the Bridge Port.";
 }

 feature flexible-bridge-port {
   description
     "A flexible bridge port supports an Interface that is a Bridge
     Port to be a separate Interface from the underlying (MAC)
     layer.";
 }
```

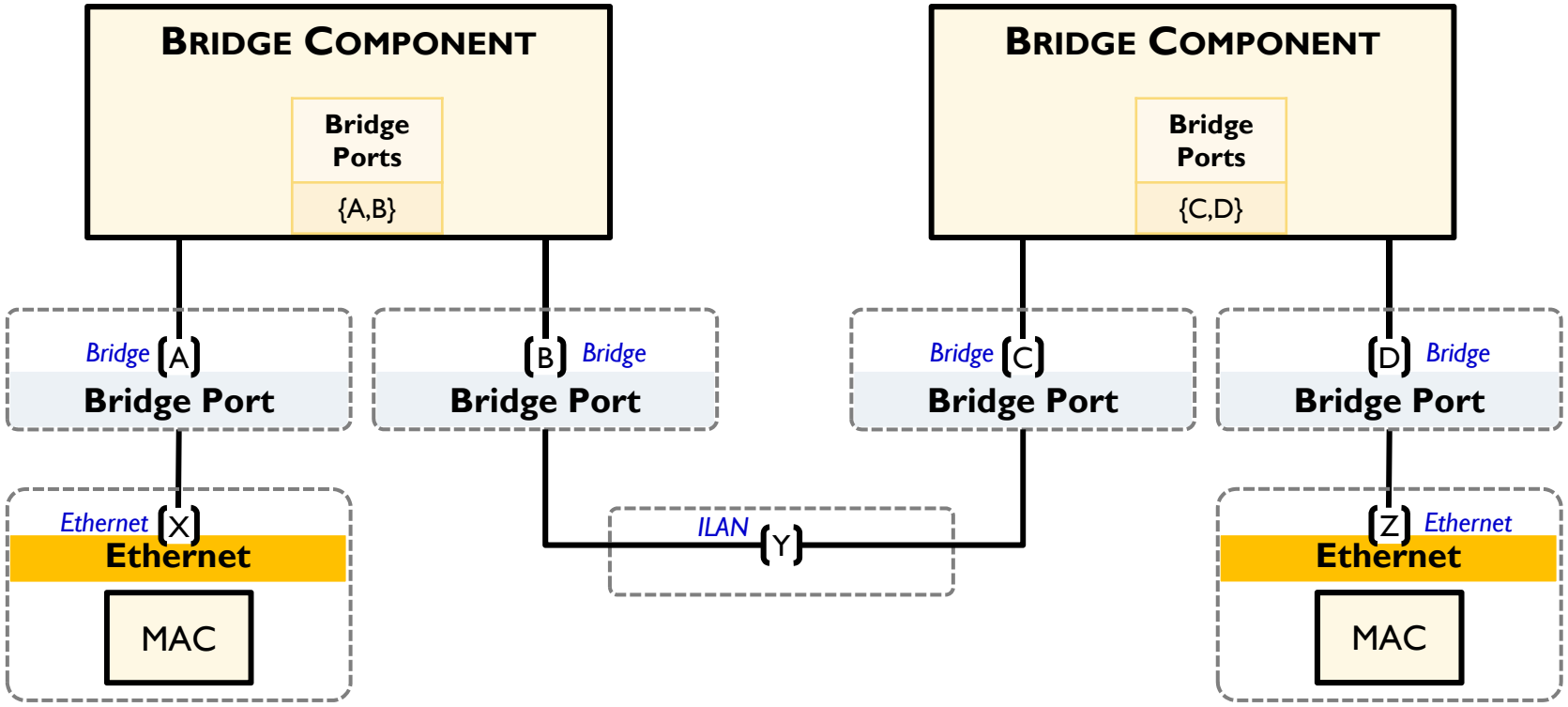
Evolution of Bridge Port Model-1 — Link Aggregation



Evolution of Bridge Port Model-2 — Link Aggregation



Evolution of Bridge Port Model-1 — Internal LAN



Evolution of Bridge Port Model-1 — Internal LAN