

# YANG derived-from-of-self example (updated)

yangsters-smansfield-yang-derived-example-0323

# Example of YANG derived-from-or-self()

- This is an example of what the derived-from-or-self() YANG function does related to augment statements
- ieee802-small-interface.yang
  - Example of creating an “interface”
  - Has two derived identities to use for the example
- ieee802-small-bridge-interface.yang
  - Example that augment small interface and provides an example of a “bridge port”
- ieee802-small-car-interface.yang
  - Example that augments small bridge interface
  - Derives a new type from “small-ethernet-type”

# YANG

```
module: ieee802-small-interface
+--rw small-interfaces
+--rw small-interface* [name]
+--rw name                string
+--rw type?                identityref
+--rw small-bridge:ethernet-port-strict
| +--rw small-bridge:ethernet-port-strict-leaf-example? string
| +--rw car-interface:car-strict-example
|   +--rw car-interface:car-strict-leaf-1? string
+--rw small-bridge:ethernet-port-open
+--rw small-bridge:ethernet-port-open-leaf-example? string
+--rw car-interface:car-derived-from-example
+--rw car-interface:car-derived-leaf-1? string
```



ieee802-small-interface.yang



ieee802-small-bridge-interface.yang



ieee802-small-car-interface.yang

# When statements used

- **Strict Example**

```
augment "/small-interface:small-interfaces/small-interface:small-interface/small-bridge:ethernet-port-strict" {  
  when "../small-interface:type = 'small-interface:small-ethernet-type' " {
```

- **Derived From Example**

```
augment "/small-interface:small-interfaces/small-interface:small-interface/small-bridge:ethernet-port-open" {  
  when "../small-interface:type = derived-from-or-self(../small-interface:type, 'small-interface:small-ethernet-type')" {
```

# Instance data that works

```
{  
  "name": "Car1",  
  "type": "ieee802-small-car-interface:car-interface-type",  
  "ieee802-small-bridge-interface:ethernet-port-open": {  
    "ethernet-port-open-leaf-example": "car example 1",  
    "ieee802-small-car-interface:car-derived-from-example": {  
      "car-derived-leaf-1": "car interface example leaf"  
    }  
  }  
},
```



e-ok.json

This works because of the use of  
derived-from-or-self()

# Instance data that doesn't work

```
{
  "name": "Fail Case 1 - This one should fail",
  "type": "ieee802-small-car-interface:car-interface-type",
  "ieee802-small-bridge-interface:ethernet-port-strict": {
    "ethernet-port-strict-leaf-example": "added attribute
to ethernet-port-strict",
    "ieee802-small-car-interface:car-strict-example": {
      "ieee802-small-car-interface:car-strict-leaf-1": "car
interface example leaf"
    }
  }
}
```



e-error.json

This identity is derived from another base type so this example fails

```
scott@Cosima:~/rws/derived/small$ expect e-error.in
spawn /usr/local/bin/yanglint
> load ieee802-small-interface
> load ieee802-small-bridge-interface
> load ieee802-small-car-interface
> data -t config -f json e-error.json
libyang[0]: When condition "small-interface:type = 'small-
interface:small-ethernet-type' " not satisfied. (path: Data
location "/ieee802-small-interface:small-interfaces/small-
interface[name='Fail Case 1 - This one should fail']/ieee802-
small-bridge-interface:ethernet-port-strict".)
YANGLINT[E]: Failed to parse input data file "e-error.json".
> quit
```

# What this means

- If an IEEE module augments IETF interface and does not use derived-from-or-self
  - Any module that augments the IEEE module will only be able to use the Interface types that are specifically listed in the “when” condition.
- IEEE 802.1Q bridge has relaxed this restriction to allow by using derived-from-or-self on bridge-port.
  - So for example the type can be ianaif:ethernetCsmacd or anything that derives from that.

## ieee802-dot1q-bridge.yang

```
***snip***
augment "/if:interfaces/if:interface" {
  when
    "derived-from-or-self(if:type,'ianaif:bridge') or "+
    "derived-from-or-self(if:type,'ianaif:ethernetCsmacd') or "+
    "derived-from-or-self(if:type,'ianaif:ieee8023adLag') or "+
    "derived-from-or-self(if:type,'ianaif:ilan')" {
    description
      "Applies when a Bridge interface.";
  }
  description
    "Augment the interface model with the Bridge Port";
  container bridge-port {
    description
  }
}
***snip***
```