FUJITSU

shaping tomorrow with you

Open ROADM API Overview
MSA v4

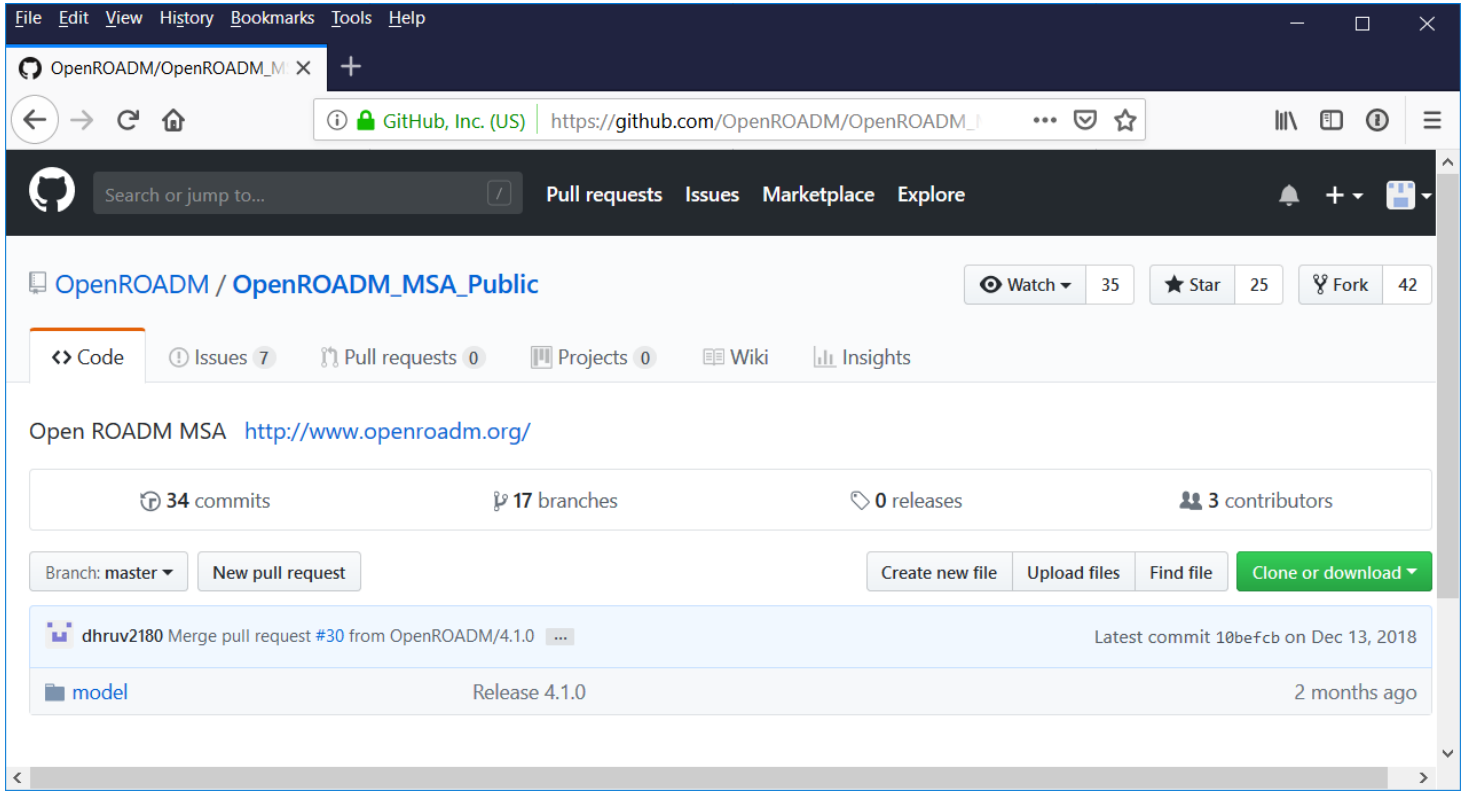
February 24, 2019

Fred Gruman
Fujitsu Network Communications, Inc

Open ROADM

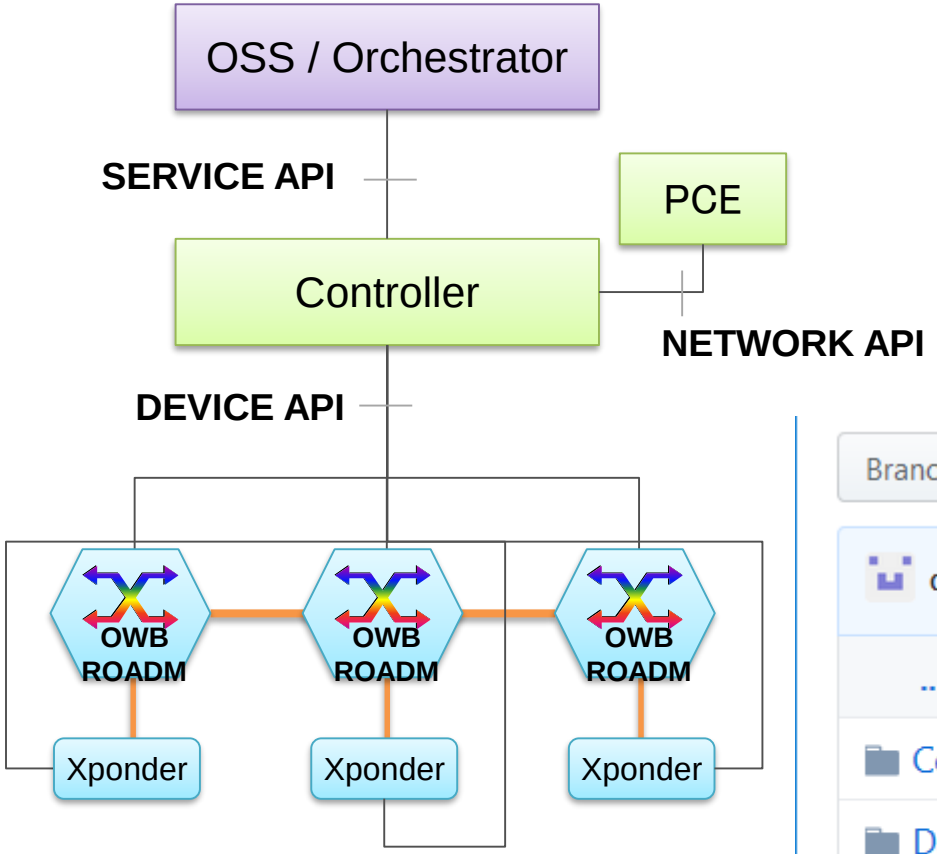
Open ROADM API Model

- Open ROADM API model is available on GitHub
  - https://github.com/OpenROADM/OpenROADM_MSA_Public
- Current version is Version 4.1.0
- This document discusses v4.1.0

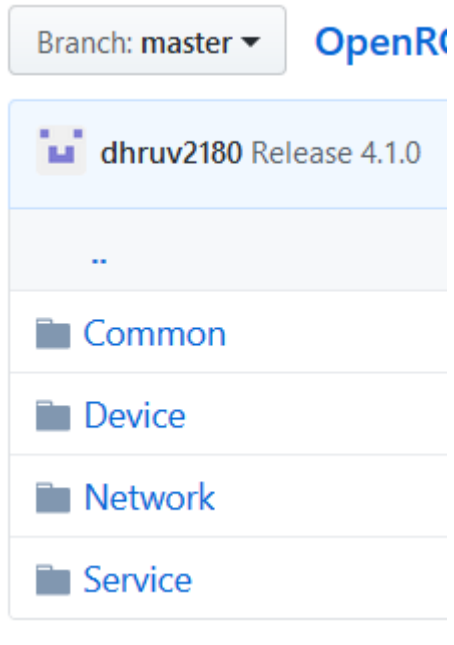


Open ROADM Model Organization

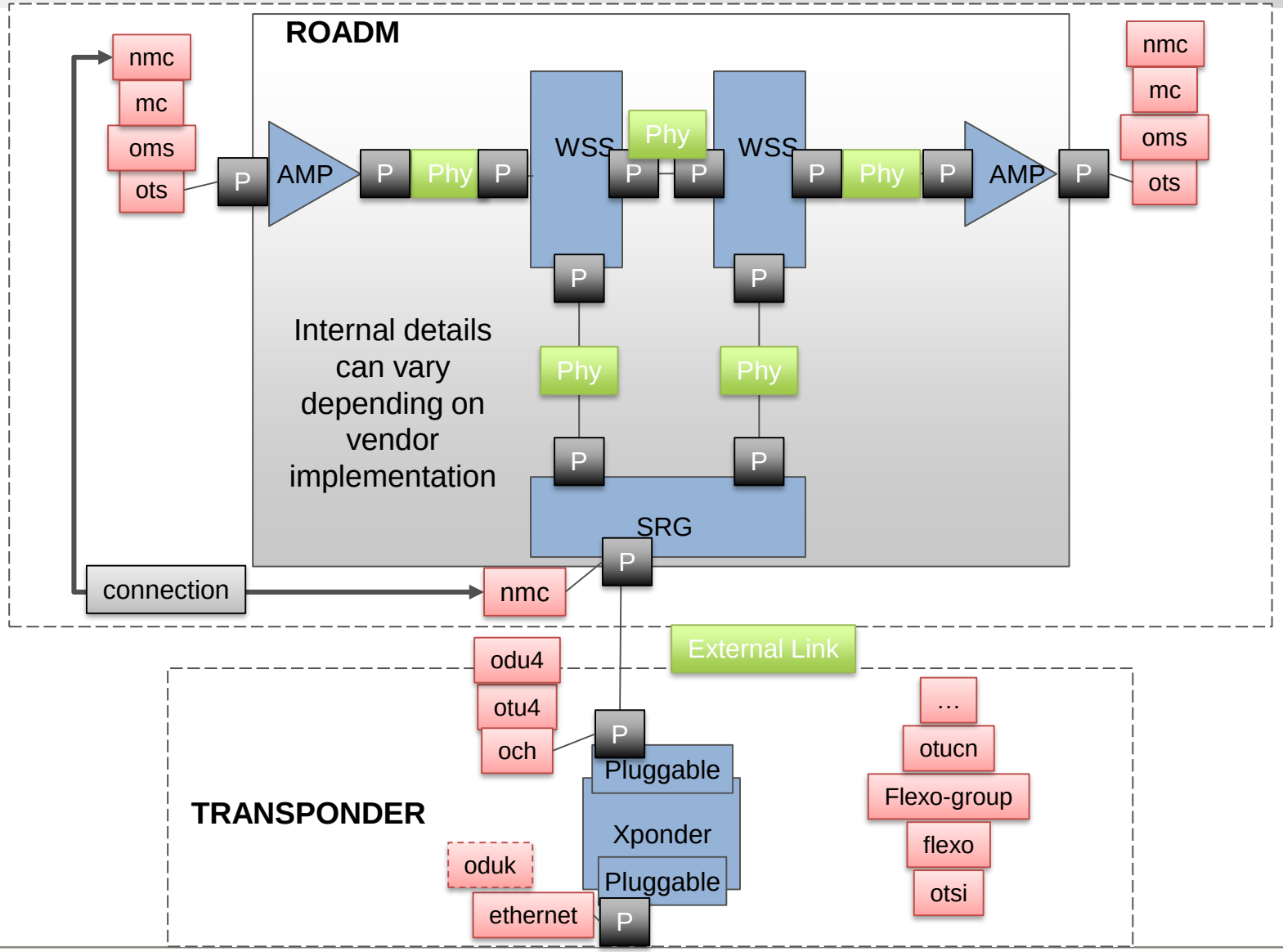
- **Device Model**
  - **Interface to Open ROADM devices**
  - **Also basis for the inventory database**
- Network Model
  - Abstract network model for routing
- Service Model
  - Optical service creation, modification, and deletion
- **Common Models**
  - **Models that apply to both the Open ROADM device and controller**
  - **Alarms, PM, TCA**
  - **Common type definitions**



External pluggables also supported but not fully validated



Open ROADM Object Model



Note: MC and NMC replace OCH from v1.2.1 on ROADM

Circuit Pack

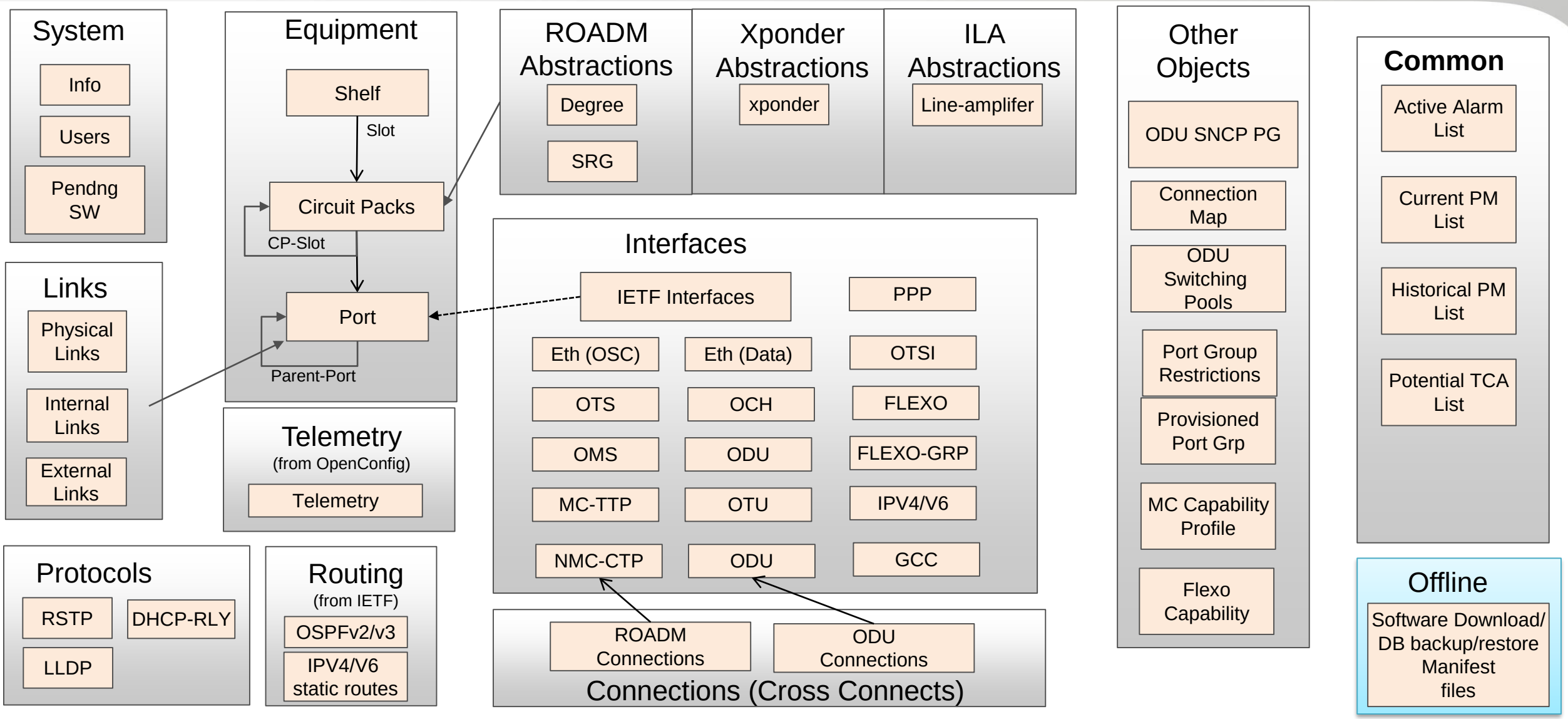
PORT

Physical or External Link

Interfaces

ROADM-Connection

Device Model Objects



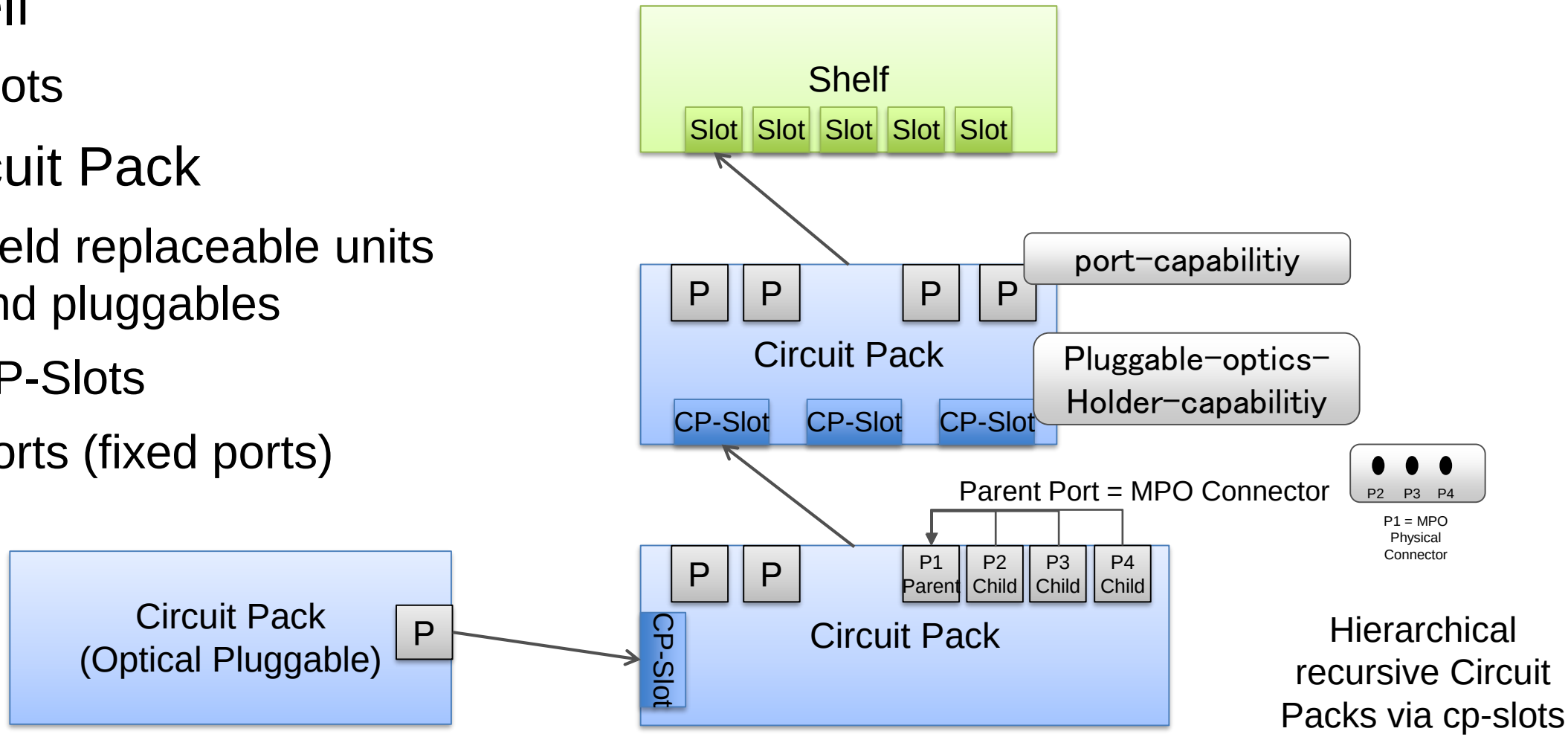
Information Container

- Info container contains system level attributes
- Node-id, clli (office location), vendor/model/serial-id for the node, IP address info (IPv4 or IPv6), software version/build, openroadm-version, current time, and location information
  - Also contains some capabilities related to the maximum number of degrees and SRGs, and maximum number of historical PM bins for 15m and 24h granularity

```
+--rw info
  |  +--rw node-id?
  |  +--rw node-number?
  |  +--rw node-type
  |  +--rw clli?
  |  +--ro vendor
  |  +--ro model
  |  +--ro serial-id
  |  +--rw ipAddress?
  |  +--rw prefix-length?
  |  +--rw defaultGateway?
  |  +--ro source?
  |  +--ro current-ipAddress?
  |  +--ro current-prefix-length?
  |  +--ro current-defaultGateway?
  |  +--ro macAddress?
  |  +--ro softwareVersion?
  |  +--ro software-build?
  |  +--ro openroadm-version?
  |  +--rw template?
  |  +--ro current-datetime?
  |  +--rw lifecycle-state?
  |  +--rw geoLocation
  |  |  +--rw latitude?
  |  |  +--rw longitude?
  |  +--ro max-degrees?
  |  +--ro max-srgs?
  |  +--ro max-num-bin-15min-historical-pm?
  |  +--ro max-num-bin-24hour-historical-pm?
```

Equipment Model

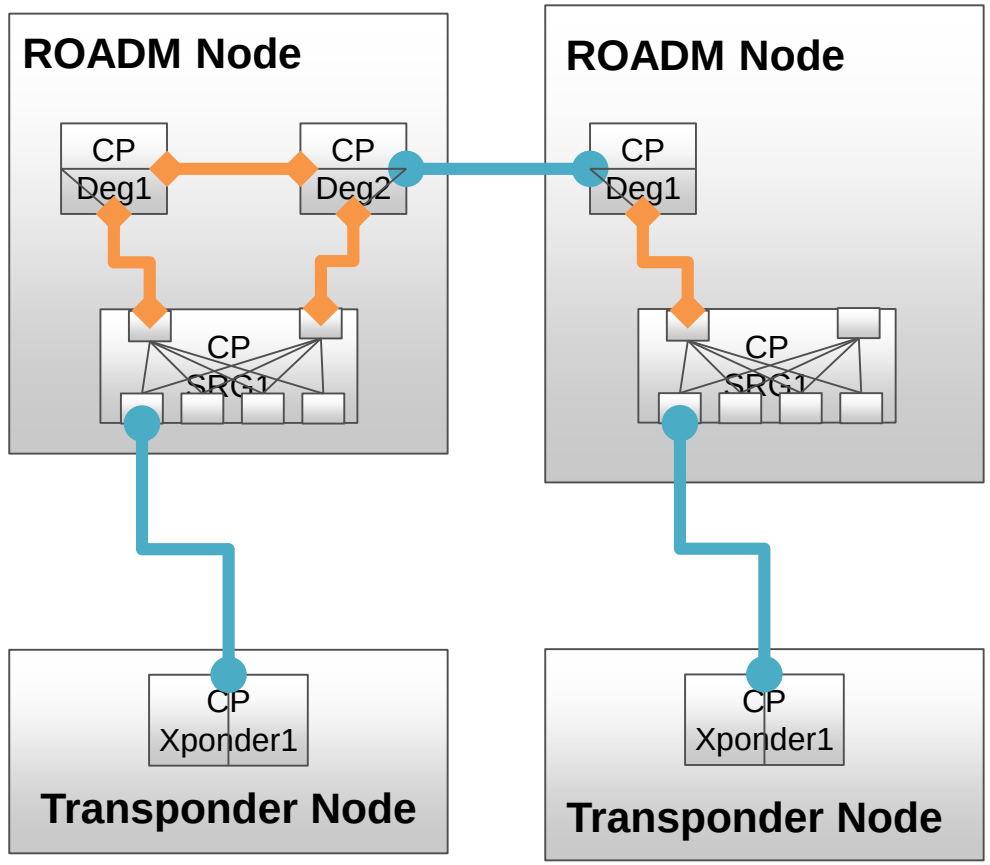
- Shelf
  - Slots
- Circuit Pack
  - Field replaceable units and pluggables
  - CP-Slots
  - Ports (fixed ports)



Hierarchical recursive Circuit Packs via cp-slots

Links

- Links connect circuit-pack ports
  - Links are unidirectional
- 3 types of Links modeled in the device
  - *Physical Link* – Fiber or electrical cable between two circuit-packs
  - *External Link* – Fiber between a circuit-pack/port on one NE and a port on another NE
  - *Internal Link* – internal connectivity on the data path within a circuit-pack (or group of circuit packs if connected via backplane)



WDM Abstractions – Degree

- Degree
  - ROADM degree (connects to degree on another ROADM NE)
  - Includes pre/post-Amplifier and WSS
  - Lifecycle state tracks the lifecycle of the degree (planned, deployed, maintenance, etc.)
  - Circuit-packs list identifies all circuit packs that make up the degree
  - Connection-ports indicate the line side ports (e.g., location of TTP)
  - OTDR port identifies the OTDR circuit-pack and port for this degree
  - MC capabilities provide the flexgrid capability information

```
 +--rw degree* [degree-number]
|  +--rw degree-number
|  +--rw lifecycle-state?
|  +--ro max-wavelengths
|  +--rw circuit-packs* [index]
|  |  +--rw index
|  |  +--rw circuit-pack-name
|  +--rw connection-ports* [index]
|  |  +--rw index
|  |  +--rw circuit-pack-name
|  |  +--rw port-name
|  +--rw otdr-port
|  |  +--rw circuit-pack-name?
|  |  +--rw port-name?
|  +--ro mc-capabilities*
```

WDM Abstractions – SRG

- Shared Risk Group (SRG)
  - Add/Drop unit(s) for CD or CDC technology
  - For CD, an SRG group is a “contention group”, e.g., a wavelength can only be dropped once within the SRG
  - Lifecycle state tracks the lifecycle of the SRG
  - Maximum and currently provisioned number of add-drop-ports for the SRG
  - Wavelength-duplication determines if SRG is CD or CDC
  - Circuit-packs list identifies all circuit packs that make up the SRG
  - MC capabilities provide the flexgrid capability information

```
+--rw shared-risk-group* [srg-number]
|  +--ro max-add-drop-ports
|  +--ro current-provisioned-add-drop-ports
|  +--rw srg-number
|  +--rw lifecycle-state?
|  +--ro wavelength-duplication
|  +--rw circuit-packs* [index]
|  |  +--rw index
|  |  +--rw circuit-pack-name
|  +--ro mc-capabilities*
```

WDM Abstractions – Xponder

- Xponder
  - Identifies a logical xponder
  - Xponder would be a transponder, muxponder, regenerator or OTN switch/switchponder (xpdr-type)
  - Lifecycle state tracks the lifecycle of the xponder
  - Recolor only applies to regenerators and indicates that wavelength recoloring is supported
  - Xpdr-port list contains the list of ports associated with the xponder
    - Eqpt-srg-id indicates SRG risks associated with the ports

```
 +--rw xponder* [xpdr-number]
|  +--rw xpdr-number
|  +--rw xpdr-type
|  +--rw lifecycle-state?
|  +--ro recolor?
|  +--rw xpdr-port* [index]
|     +--rw index
|     +--rw circuit-pack-name
|     +--rw port-name
|     +--rw eqpt-srg-id?
```

Xponder Summary

FUJITSU

xponder	Xpdr-type	Port-Qualifier	Connectivity	Cross Connect?
Transponder	Tpdr	Xpdr-client/network	Connectivity-map	No
Muxponder	Mpdr	Switch-client/network	Switching Pool	Yes
Regenerator	Regen-uni	Xpdr-client/network	Connectivity-map	No
	Regen	Switch-client/network	Switching Pool	Yes
OTN Switch / Switchponder	Switch	Switch-client/network	Switching Pool	Yes

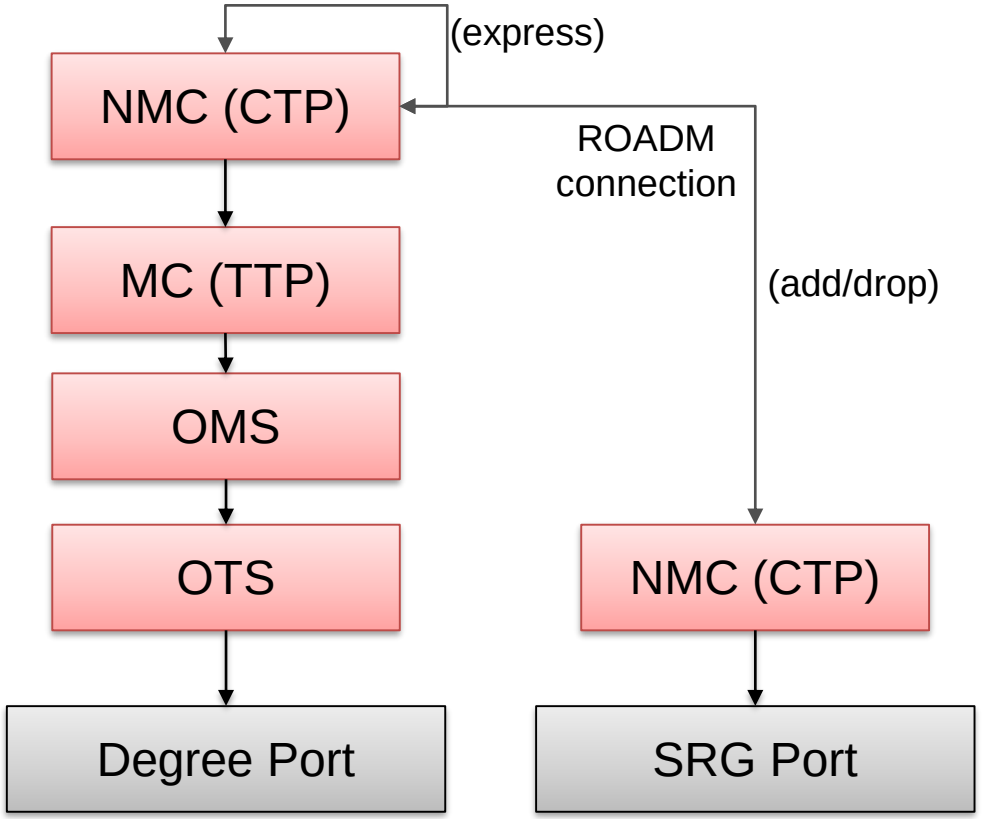
WDM Abstractions – ILA

- Line Amplifier (ILA)
  - Represents an ILA
  - Control is provided via control-mode, target-gain target-tilt, and egress-average-channel-power
  - Capabilities provided by amp-type, amp-gain-range
  - Out-voa-att provides a RO value of the attenuator
  - Lifecycle state tracks the lifecycle of the line-amp
  - Circuit-packs list identifies all circuit packs that make up the ILA
  - Line port list identifies the line side ports
  - OSC port identifies the OSC pluggable circuit-pack and port (specific to direction)
  - OTDR port identifies the OTDR circuit-pack and port for this degree

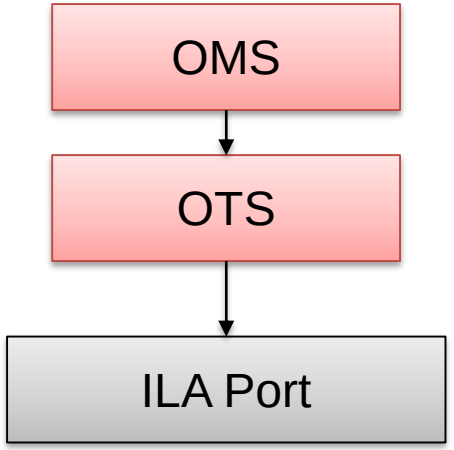
```
+--rw line-amplifier* [amp-number]
|  +--rw amp-number
|  +--ro amp-type
|  +--rw control-mode?
|  +--ro amp-gain-range?
|  +--rw target-gain?
|  +--rw target-tilt?
|  +--rw egress-average-channel-power?
|  +--ro out-voa-att?
|  +--ro partner-amp?
|  +--rw ila-direction-label?
|  +--rw lifecycle-state?
|  +--rw circuit-pack* [index]
|  |  +--rw index
|  |  +--rw circuit-pack-name
|  +--rw line-port* [port-direction]
|  |  +--rw port-direction
|  |  +--rw tx-instance-port-direction-label?
|  |  +--rw rx-instance-port-direction-label?
|  |  +--rw circuit-pack-name
|  |  +--rw port-name
|  +--rw osc-port* [port-direction]
|  |  +--rw port-direction
|  |  +--rw circuit-pack-name
|  |  +--rw port-name
|  +--rw otdr-port* [otdr-direction]
|     +--rw otdr-direction
|     +--rw circuit-pack-name
|     +--rw port-name
```

Interfaces

- Interfaces represents the facilities on the device



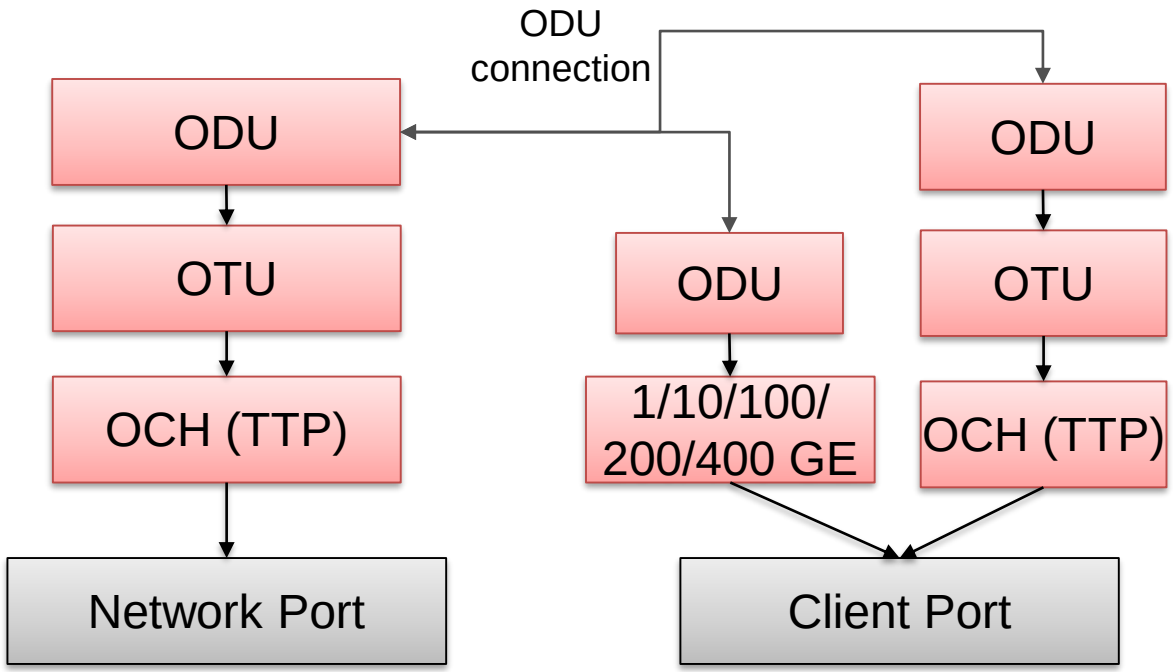
ROADM Interfaces



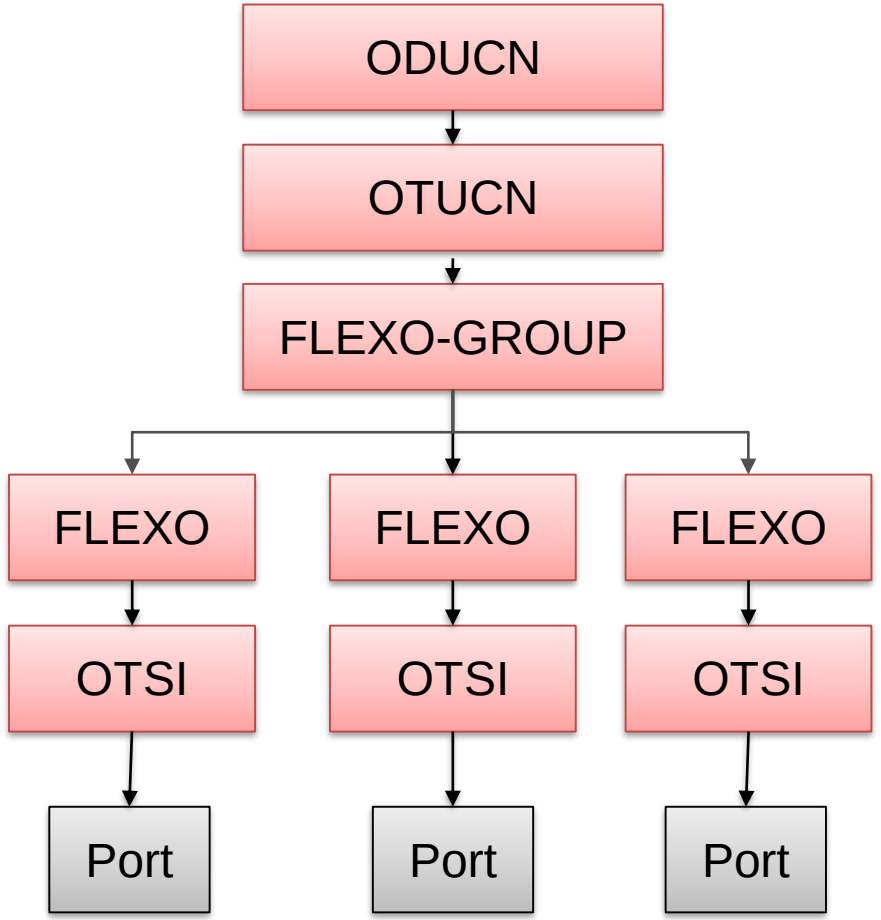
ILA Interfaces

Interfaces

- Interfaces represents the facilities on the device



Xponder Interfaces



Other Interfaces

- IPV4
- IPV6
- GCC
- PPP

Connectivity matrices – Connection Map

- Connection maps are used with ROADMs, transponders and unidirectional regenerators
  - Used to indicate the port-to-port connectivity between external ports of the ROADM or xponder
  - Connection map represents uni-directional connectivity between ports
- Connection map provides the semi-static connectivity matrix for the ROADM node
  - Describes what degrees and SRGs can be interconnected by a roadm-connection
  - Updated as degrees and SRGs are commissioned and physical links provisioned
- Connection map provides the semi-static connectivity matrix for transponders and unidirectional regenerators
  - The connectivity is fixed and no cross connects are required

```
  +--ro connection-map* [connection-map-
number]
|  +--ro connection-map-number
|  +--ro source
|  |  +--ro circuit-pack-name
|  |  +--ro port-name
|  +--ro destination* [circuit-pack-name port-
name]
|     +--ro circuit-pack-name
|     +--ro port-name
```

Connectivity matrices – Switching Pool

- Switching pools provide the ODU connectivity between external ports of an OTN switch, muxponder or bi-directional regenerator
  - Switching pools represents bi-directional connectivity between ports
- Represents both blocking and non-blocking connectivity
  - Each set of ports within a non-blocking-list are not blocking
  - Multiple sets of non-blocking-list entries are blocking and provide an interconnect BW constraint

```
+--ro odu-switching-pools* [switching-
pool-number]
   +--ro switching-pool-number
   +--ro switching-pool-type?
   +--ro odu-connection-direction-
capabilities?
   +--ro non-blocking-list* [nbl-number]
      +--ro nbl-number
      +--ro interconnect-bandwidth-unit?
      +--ro interconnect-bandwidth?
      +--ro port-list* [circuit-pack-name
port-name]
      |  +--ro circuit-pack-name
      |  +--ro port-name
      +--ro pluggable-optics-holder-list*
[circuit-pack-name slot-name]
         +--ro circuit-pack-name
         +--ro slot-name
```

WDM Cross Connects

- Roadm-connections are the cross connects at the NMC (CTP) level
  - Unidirectional
  - Connects NMC interfaces on MC/OMS and NMC on SRG add/drop port
  - Supports Express, Add and Drop traffic
  - Also controls the optical control mode for the channel

```
+--rw roadm-connections* [connection-name]
     |  +--rw connection-name
     |  +--rw opticalControlMode?
     |  +--rw target-output-power?
     |  +--rw source
     |  |  +--rw src-if
     |  +--rw destination
     |     +--rw dst-if
```

ODU Cross Connects

- ODU-connections are the cross connects at the ODU (CTP) level
  - Models support both bidirectional and unidirectional cross connects
  - Vendors must support bidirectional cross connects; support for unidirectional cross connects is optional (defined in the switching pool advertisement)
  - Connects ODU interfaces on client and/or network ports (ODU CTPs)
    - Supports Client – Client, Client – Network, Network – Network
  - Used by OTN switches/switchponder, muxponder and regenerators
  - Not used on transponders or uni-directional regenerators

```
+--rw odu-connection* [connection-name]
       |  +--rw connection-name
       |  +--rw direction?
       |  +--rw source
       |  |  +--rw src-if
          |  +--rw destination
          |     +--rw dst-if
```

Protection Groups

- ODU-SNCP protection group
  - Level: Supports line and path protection
  - Prot-type: supports 1+1
  - Switching-direction: unidirectional or bidirectional switching
  - Modes: SNC/Ne, SNC/Ns, SNC/S, SNC/I, CL-SNCG/I

```
+--rw protection-grps
      |  +--rw org-openroadm-prot-otn-linear-aps:odu-sncp-pg* [name]
      |     +--rw org-openroadm-prot-otn-linear-aps:name
      |     +--rw org-openroadm-prot-otn-linear-aps:level
      |     +--rw org-openroadm-prot-otn-linear-aps:prot-type?
      |     +--rw org-openroadm-prot-otn-linear-aps:switching-direction?
      |     +--rw org-openroadm-prot-otn-linear-aps:revertive?
      |     +--rw org-openroadm-prot-otn-linear-aps:mode
      |     +--rw org-openroadm-prot-otn-linear-aps:protection-trigger-level?
      |     +--rw org-openroadm-prot-otn-linear-aps:wait-to-restore?
      |     +--rw org-openroadm-prot-otn-linear-aps:holdoff-timer
      |     |  +--rw org-openroadm-prot-otn-linear-aps:holdoff?
      |     |  +--rw org-openroadm-prot-otn-linear-aps:holdoff-multiplier?
      |     +--rw org-openroadm-prot-otn-linear-aps:working-if
      |     +--rw org-openroadm-prot-otn-linear-aps:pg-interfaces*
      |     +--ro org-openroadm-prot-otn-linear-aps:active-if?
```

Protocols

- LLDP
  - Used for link discovery over the OSC (WDM layer)
- RSTP (not shown)
  - Used for communication via L2 DCN
- IPV4/IPV6-DHCP-RELAY
  - Used for remote transponder connectivity via GCC0 for ZTP

```
+--rw protocols
      |  +--rw org-openroadm-lldp:lldp
      |  |  +--rw org-openroadm-lldp:global-config
      |  |  |  +--rw org-openroadm-lldp:adminStatus?
      |  |  |  +--rw org-openroadm-lldp:msgTxInterval?
      |  |  |  +--rw org-openroadm-lldp:msgTxHoldMultiplier?
      |  |  +--rw org-openroadm-lldp:port-config* [ifName]
      |  |  |  +--rw org-openroadm-lldp:ifName
      |  |  |  +--rw org-openroadm-lldp:adminStatus?
      |  |  +--ro org-openroadm-lldp:nbr-list
      |  |     +--ro org-openroadm-lldp:if-name* [ifName]
      |  |        +--ro org-openroadm-lldp:ifName
      |  |        +--ro org-openroadm-lldp:remoteSysName?
      |  |        +--ro org-openroadm-lldp:remoteMgmtAddressSubType?
      |  |        +--ro org-openroadm-lldp:remoteMgmtAddress?
      |  |        +--ro org-openroadm-lldp:remotePortIdSubType?
      |  |        +--ro org-openroadm-lldp:remotePortId?
      |  |        +--ro org-openroadm-lldp:remoteChassisIdSubType?
      |  |        +--ro org-openroadm-lldp:remoteChassisId?
```

Flexgrid capabilities

- Media Channel (MC) capabilities can be announced against any port
  - Can be used to define small ROADM, AWG constraints, etc.

```
+--ro mc-capability-profile* [profile-name]
|  +--ro profile-name
|  +--ro center-freq-granularity?
|  +--ro min-edge-freq?
|  +--ro max-edge-freq?
|  +--ro slot-width-granularity?
|  +--ro min-slots?
|  +--ro max-slots?
```

- Profile-based solution
  - Define a profile
  - Port's mc-capabilities would reference the profile name

```
 Port Object:
+--ro mc-capabilities*          -> /org-openroadm-
device/mc-capability-profile/profile-name
```

Slot and Port Capabilities Announcements

FUJITSU

- A series of capabilities are defined against the slot and port
  - Circuit-pack cp-slots pluggable-optics-holder-capability (only applies to slots that can host pluggables)
  - Circuit-pack port port-capabilities
- Both capabilities define:
  - Interface hierarchy supported via the if-cap-type
  - If the port can host flexo-group, then it points to the flexogroup-capability-group-id
  - Protection capability
  - OTN specific capabilities such as protection capability, delay measurement, TCM, muxponder mapping restrictions, and ODU muxing hierarchy

Switching Pool Announcement

- Switching pool applies to OTN (ODU) cross connect fabrics
  - Bidirectional connectivity announcement
  - Also supports ability to identify blocking
- Modeled as a series of non-blocking lists (full mesh connectivity) interconnected via a fixed, blocking bandwidth
- Non-blocking list can contain ports (pluggable ports) or slots (cp-slots that will hold pluggables)

```
+--ro odu-switching-pools* [switching-pool-number]
   +--ro switching-pool-number
   +--ro switching-pool-type?
   +--ro odu-connection-direction-capabilities?
   +--ro non-blocking-list* [nbl-number]
      +--ro nbl-number
      +--ro interconnect-bandwidth-unit?
      +--ro interconnect-bandwidth?         uit32
      +--ro port-list* [circuit-pack-name port-name]
      |  +--ro circuit-pack-name
      |  +--ro port-name              -
      +--ro pluggable-optics-holder-list* [circuit-
pack-name slot-name]
         +--ro circuit-pack-name
         +--ro slot-name
```

Switching Pool Example

- Blocking example : Partially blocking fabric interconnect between 2 non-blocking elements
  - $[(P1, P2, \ldots, P10)_{500G}, (P11, P12, \ldots Pn)_{500G}]$

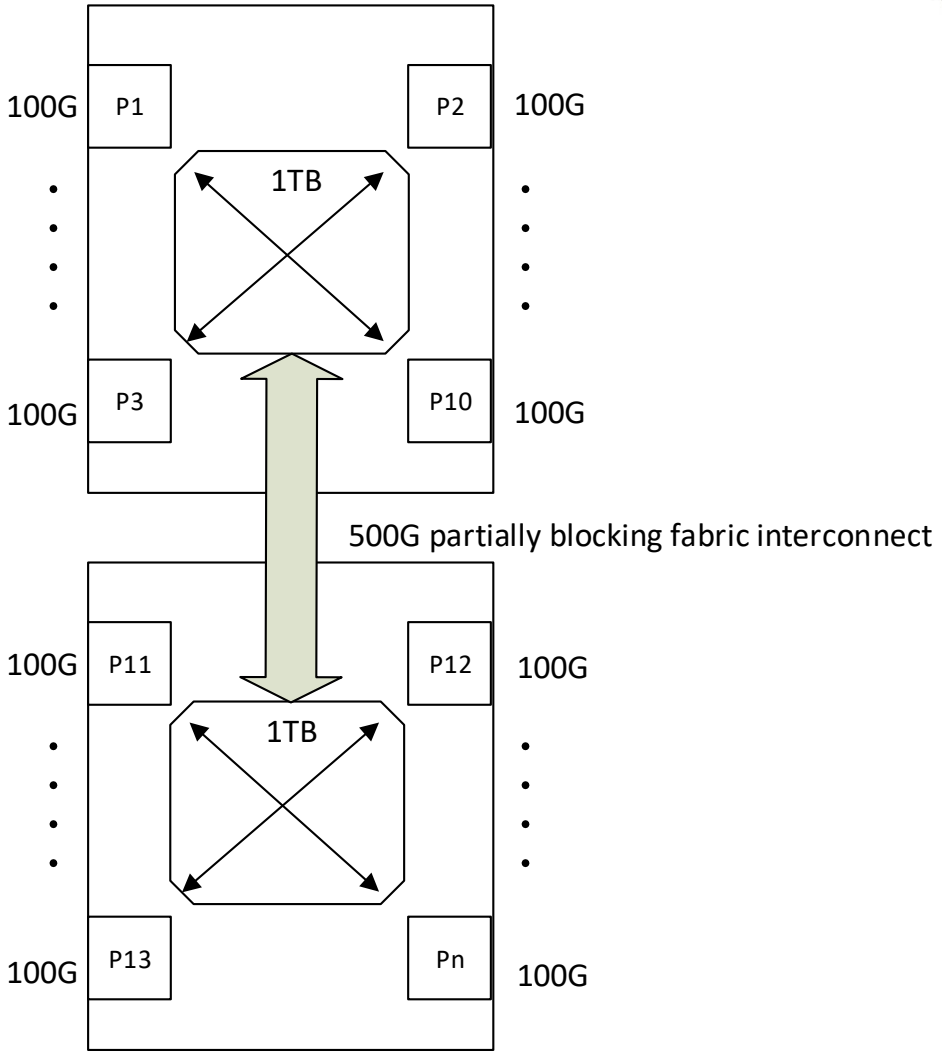
Non-blocking list #1

Non-blocking list #2

Interconnect BW = 500G

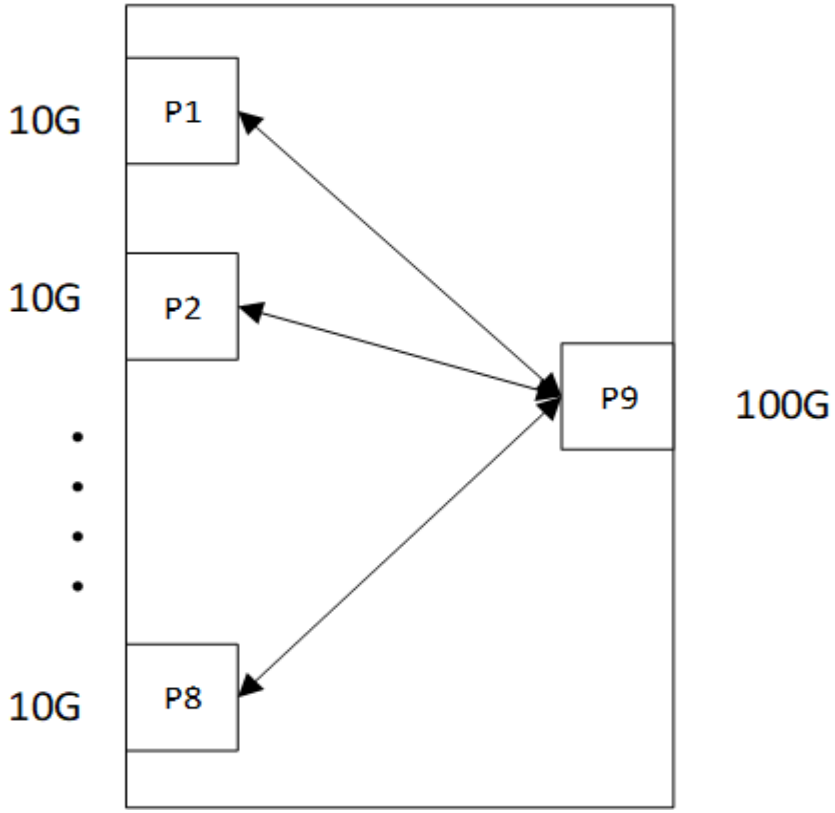
Uses multiple non-blocking-lists - one for each 1TB non-blocking element

Example has 500Gb fabric interconnect-bandwidth between non-blocking elements



Switching Pool Example - Muxponder

- Muxponder1: Simple Example
  - $[(P1, P9)_{0G}, (P2, P9)_{0G} \ldots (P8, P9)_{0G}]$




```
odu-switching-pools
  switching-pool-number              1
  switching-pool-type                blocking
  non-blocking-list
    {nbl-number              1
     interconnect-bandwidth-unit   0
     interconnect-bandwidth        0
     port-list [{CP1, P1}, {CP-NWK, P9}]
    },
    {nbl-number              2
     interconnect-bandwidth-unit   0
     interconnect-bandwidth        0
     port-list [{CP2, P2}, {CP-NWK, P9}]
    },
    …,
    {nbl-number              8
     interconnect-bandwidth-unit   0
     interconnect-bandwidth        0
     port-list [{CP10, P8}, {CP-NWK, P9}]
    }
```

Users

- Default user name/password = openroadm/openroadm

- Only one role is defined on the NE:
  - Group = ‘sudo’
  - Full access
  - Controller is expected to manage user roles

```
+--rw users
      +--rw user* [name]
      +--rw name
      +--rw password?
      +--rw group?

COMMON:
 rpcs:
    +---x chg-password
       +---w input
       |  +---w currentPassword
       |  +---w newPassword
       |  +---w newPasswordConfirm
       +--ro output
          +--ro status
          +--ro status-message?
```

Software Download

- Driven by manifest file
  - Provides vendor specific sequencing of commands to support a software download
  - Provides the sequence of commands that the controller will send to the NE for software download
  - Indicates if the commands are synchronous or asynchronous
- Software download primitives:
  - Transfer: transfers a file from an SFTP server to the NE (transfer, delete-file)
  - Stage: stage the software within the device (sw-stage)
  - Activate: Activate the new software. This will typically cause the NE to reboot and may put the NE into an optional validation phase (sw-activate)
  - Commit: commit the new software load if the vendor supports a validation phase (cancel-validation-timer)

Database Backup/Restore

FUJITSU

- Also supported by manifest file similar to software download
- Database backup primitives:
  - Backup: backup the database into a file on the device (db-backup)
  - File transfer: transfer database file to offboard SFTP server (transfer)
- Database restore primitives
  - File transfer: transfer database file from SFTP server to the device (transfer)
  - Restore: restore the database file onto the device (similar to sw-stage) (db-restore)
  - Activate: activate a restored database (db-activate). Typically will require a reboot and may enter an optional validation phase
  - Commit: commits the new database if the vendor supports a validation phase (cancel-rollback-timer)
- TBD: support of config data retrieval and replay as a mechanism for configuration backup

- Re-initialize databae to factory default is also supported via the database-init RPC

Software and Database Load Information

- The pending-sw and database-info structures are used to indicate software and database information
- Pending-sw is used during the software download process
  - Only populated when the software download is in progress (e.g., in the staging or validation phase)
  - Staging – provides the sw-version that is being staged
  - Validation phase – provides the activation time and the remaining time of the validation timer
- Database-info provides information about the current database
  - Last-restored-time would give information about when the database was created or last restored (whichever is later)
  - Rollback-timer and activation-date-time is used during the validation phase of a database restore

```
+--ro pending-sw
   +--ro sw-version?
   +--ro sw-validation-timer?
   +--ro activation-date-time?

+--ro database-info
   +--ro last-restored-time?
   +--ro rollback-timer?
   +--ro activation-date-time?
```

Firmware

- Vendors can report the firmware as either “features” or “components”

- Features is meant to be descriptive
  - Describes a feature that is enabled by the firmware
- Component is meant to relay firmware version information per hardware component

- Firmware is automatically updated if non service affecting
- Firmware is loaded on a cold reset or power cycle when service affecting
- Note: firmware may be delayed upgrading during software download validation phase
- Can also use the fw-update command to update firmware explicitly
  - Also used to force reload a firmware, if supported by the vendor

- Boot: indicates that the firmware is boot-loader or similar
  - Not protected by dual banks
  - Does not automatically get upgraded
  - Requires explicit use of fw-update command

- New firmware is available when:
  - Activated = false for feature type firmware
  - Current-version does not equal version-to-apply for component type firmware

```
+--ro software-load-version?
+--ro circuit-pack-features*
|  +--ro feature
|     +--ro description?
|     +--ro boot?
|     +--ro activated?
+--ro circuit-pack-components*
|  +--ro component
|     +--ro name?
|     +--ro boot?
|     +--ro current-version?
|     +--ro version-to-apply?
```

Zero Touch Provisioning

Streaming Telemetry

- Uses the OpenConfig telemetry model
- Model supports dial-out and dial-in telemetry
  - But still working on agreement about streaming telemetry formats and transport layers

Syslog

DCN

- Layer 2 DCN in the ROADM network
- Extend to remote transponders using OSPF/DHCP relay

Common: Alarms & PM

FUJITSU

- Alarm & PM models are in the common directory
  - Applies to both NE and controller
- Major containers:
  - Active-alarm-list
  - Alarm-notification
  - Current-Pm-list
  - Historical-Pm-list
  - Potential-tca-list
  - Tca-notification
  - RPC: collect-historical-pm-file with historical-pm-collect-result notification
  - RPC: clear-pm, chg-password
- Manifest files (description only – provided to controller offline)
  - Sw-manifest, db-backup-manfiest, db-restore-manifest

- Note: Alarm and PM YANG models went through a significant refactoring in v2.0 to improve the indexing and retrieval/filtering performance
  - Refactoring of potential-tca-list and tca-notification is pending as of R4.1.0

FUJITSU

shaping tomorrow with you