INSTITUTE
OF COMMUNICATION,
INFORMATION
AND PERCEPTION
TECHNOLOGIES

Scuola Superiore
Sant'Anna

YANG Model and NETCONF Protocol for Control and Management of Elastic Optical Networks

M. Dallaglio[1], N. Sambo[1], J. Akhtar[1], F. Cugini[2], P. Castoldi[1]

1: Scuola Superiore Sant'Anna, Pisa, Italy
2: CNIT, Pisa, Italy

ORCHESTRA

OFC 2016

Introduction

- Relevant advances in the data and control plane
  - data plane:
    - ✓flexible transponders → configurable/adaptable rate, FEC, format, slice-ability …
    - ✓support of monitoring through Digital Signal Processing (pre-FEC BER, Q factor, etc.)
  - control plane:
    - ✓Software Defined Networking → to remotely set network devices, programming transmission characteristics (such as bit rate) and switching
- Management?
  - innovations have not followed these trends yet [a]:
    - issues related to the presence of network devices from different vendors
    - lack of standard solutions (e.g., for operation administration and maintainance – OAM)
- NETCONF based on YANG model is emerging as a standard SDN protocol providing both control (e.g., data plane device configuration) and management (e.g., access to monitoring information) functionalities

In this paper:

- we present and demonstrate a YANG model describing flexible transponders supporting monitoring functionalities
- experimental demonstration: connection setup and OAM through NETCONF and YANG

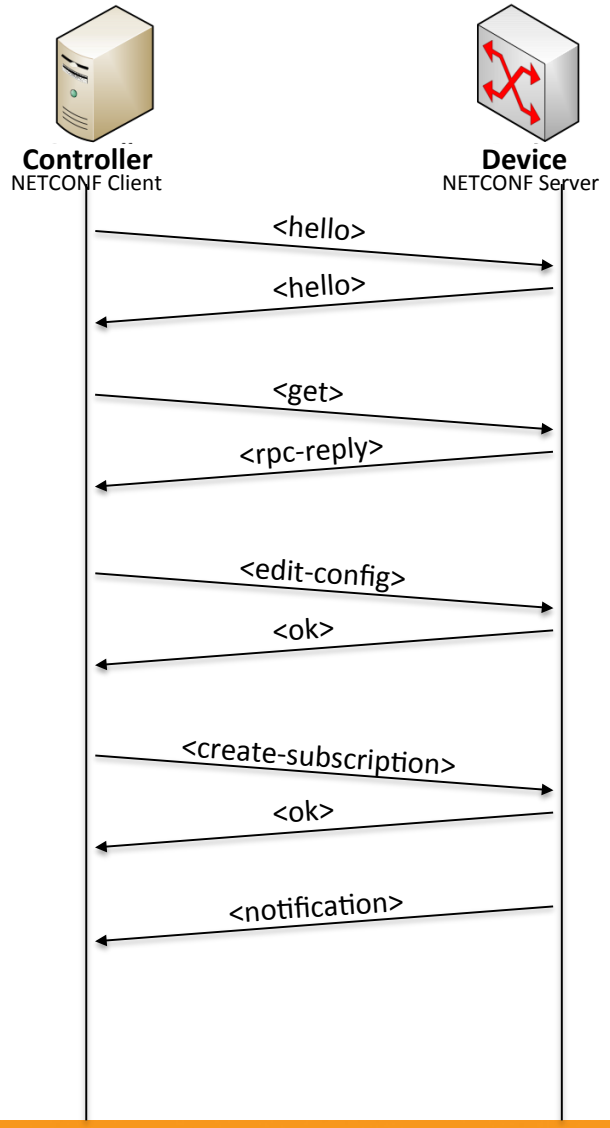
[a] A. Martinez, M. Yannuzzi, V. Lopez, D. Lopez, W. Ramirez, R. Serral-Gracia, X. Masip-Bruin, M. Maciejewski, and J. Altmann, "Network management challenges and trends in multi-layer and multi-vendor settings for carrier-grade networks," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 4, 2014.

NETCONF and YANG

- NETCONF: Standard protocol defined by the IETF [b]
- Provides mechanisms to install, manipulate, and delete states of network devices → it enables device configuration
- YANG is recommended to model and describe network devices into NETCONF messages [c]
  - ongoing work on YANG model for transponders, e.g. [d]
- NETCONF is indicated for management because it includes NOTIFICATION messages that can implement ALARMS upon monitoring

[b] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, "Network configuration protocol (NETCONF)," IETF RFC 6241, June 2011.
[c] M. Bjorklund, "YANG - a data modeling language for the network configuration protocol (NETCONF)," IETF RFC 6020.
[d] J. Vergara and et al., IETF draft-vergara-flexigrid-yang-02, March. 2016.

NETCONF messages



DISCOVERY

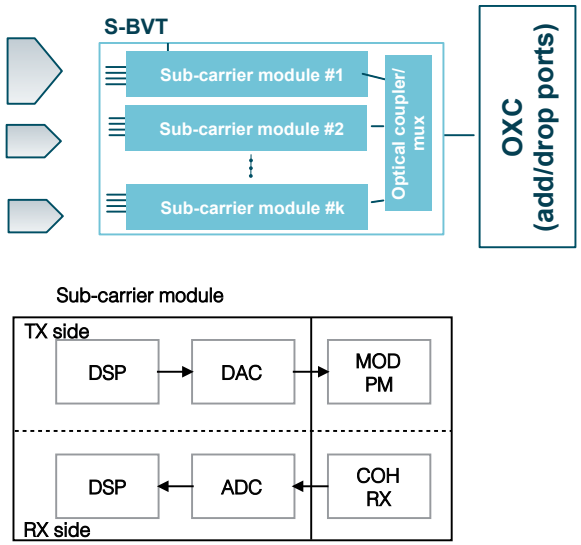
WHICH SUPPORTED CAPABILITIES?

DEVICE CONFIGURATION

SUBSCRIPTION TO SPECIFIC NOTIFICATIONS

NOTIFICATION THAT SUBSCRIBED EVENT OCCURRED

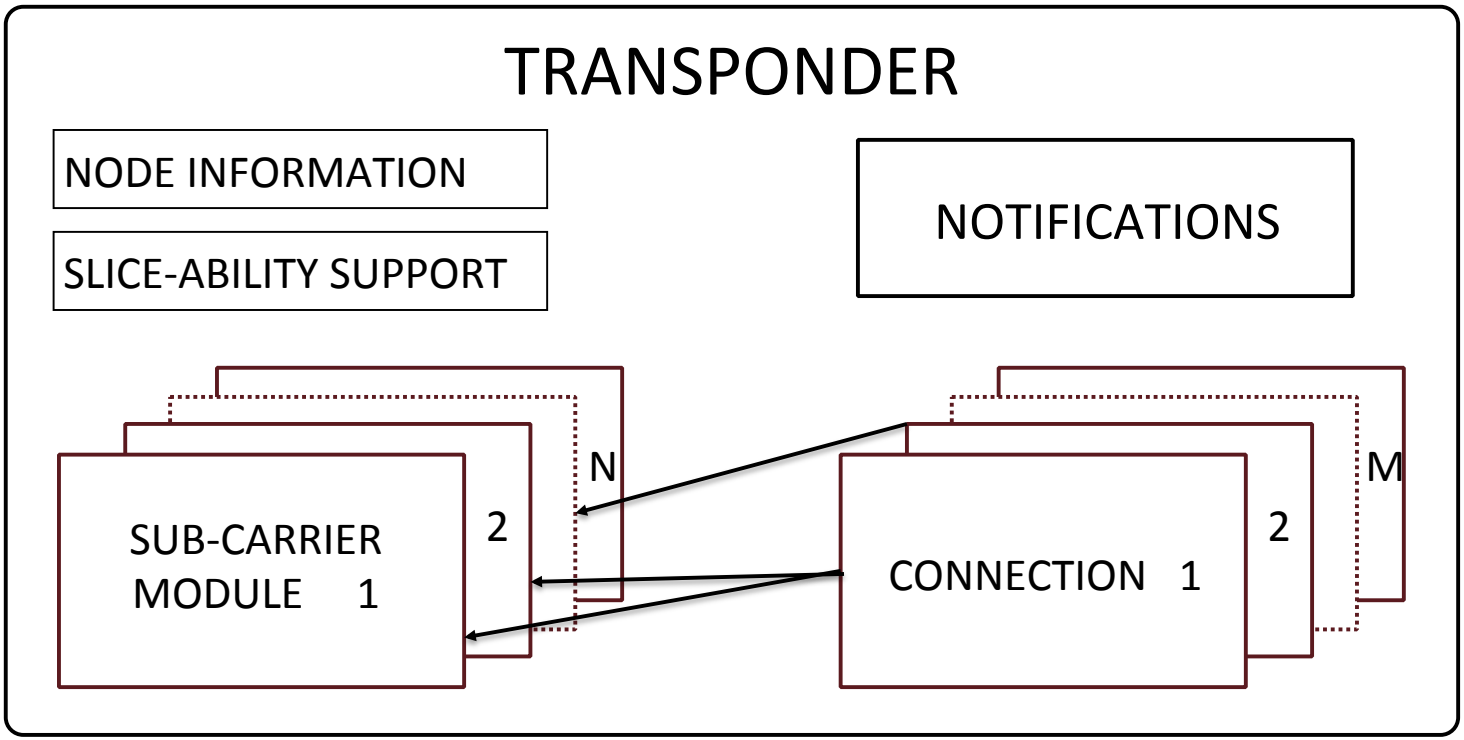
Reference sliceable transponder (S-BVT)



N. Sambo and et al., “Next generation sliceable bandwidth variable transponders,” Communications Magazine, IEEE, vol. 53, no. 2, pp. 163–171, Feb 2015.

TRANSPONDER YANG SCHEME

Following YANG standardization guidelines IETF [e,f] and OpenConfig working group [g], we propose:



[e] A. Bierman, "Guidelines for Authors and Reviewers of YANG Data Model Documents", IETF RFC 6087, 2011.
[f] R. Shakir, "Consistent Modeling of Operational State Data in YANG draft-openconfig-netmod-opstate-01", IETF Draft, 2015.
[g] http://www.openconfig.net

YANG CONFIG AND STATE DATA

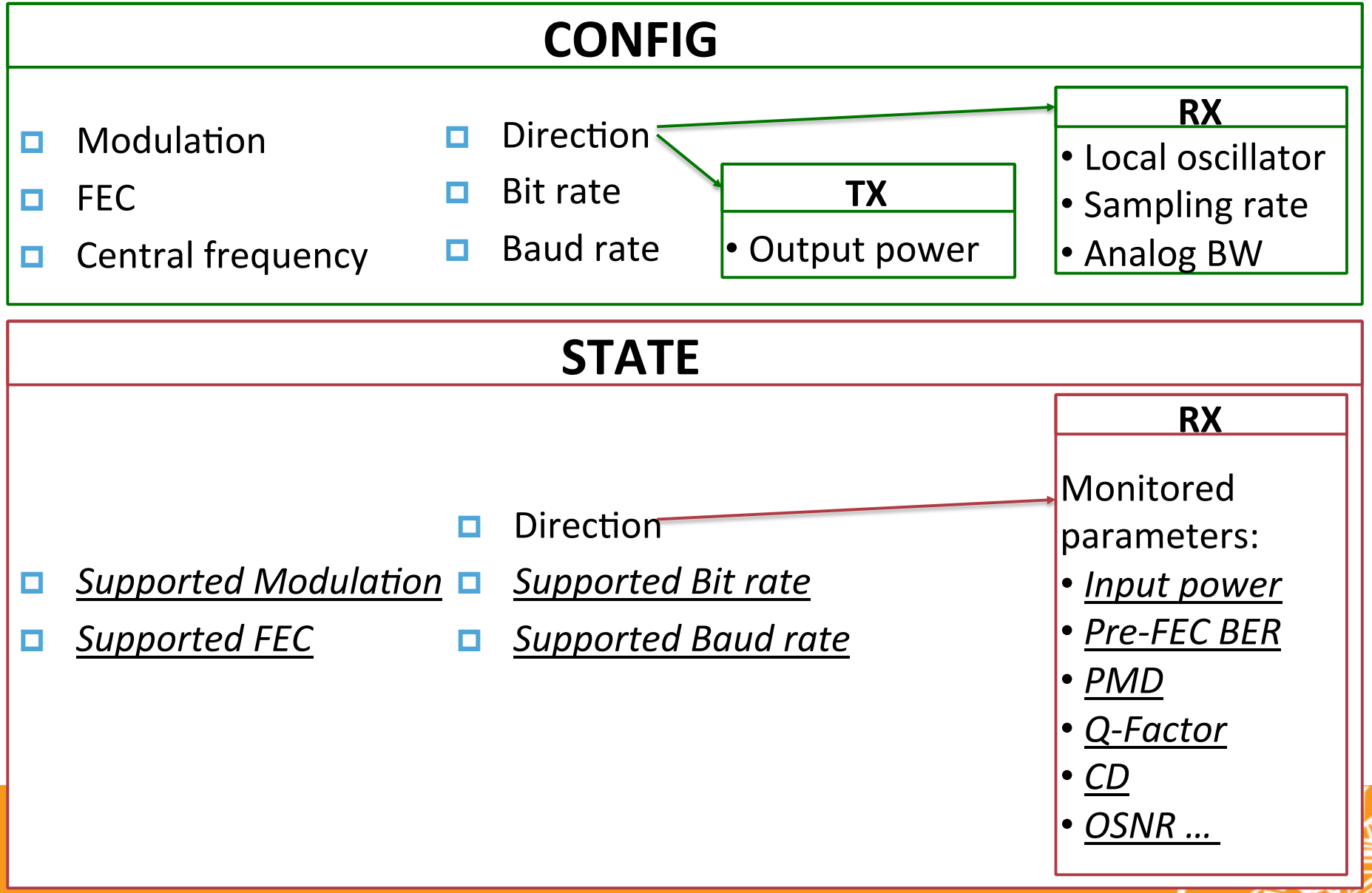
Configuration data

- Writable (NETCONF <edit-config>)
- Explicitly set by an external entity

State data

- Read only (NETCONF <get>)
- Parameters that cannot be set by an external entity
- List of parameters supported by the device

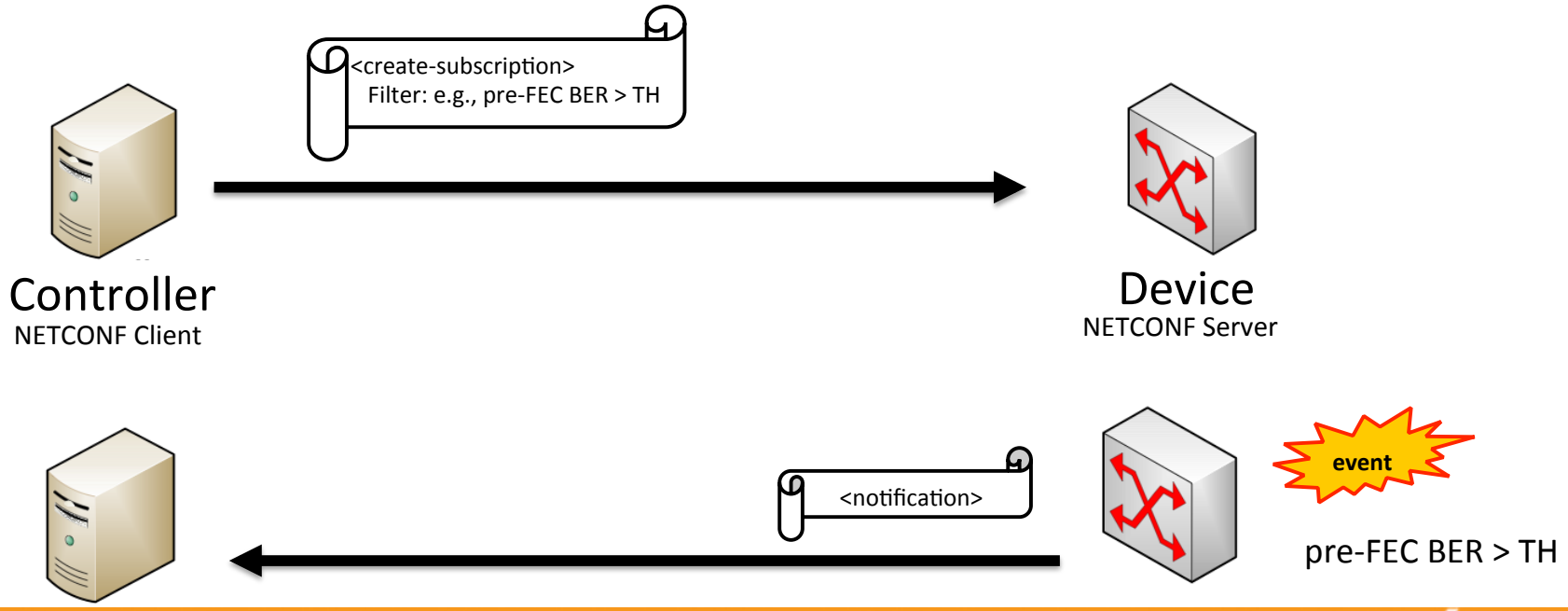
SUB-CARRIER MODULE



NOTIFICATION

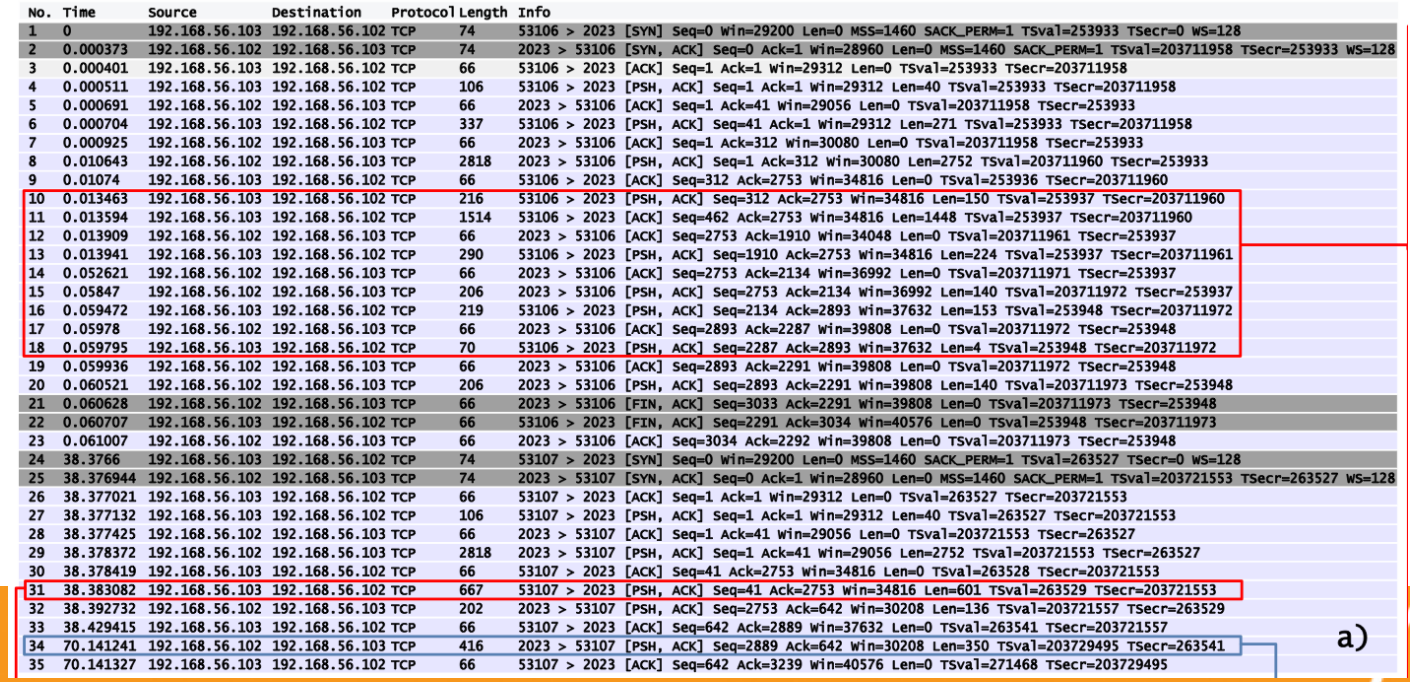
PRE-FEC BER CHANGE

- Sub-carrier ID
- pre-FEC BER above a threshold



Experimental demonstration

- The Controller based on PYTHON NETCONF client
- The device runs ConfD, a NETCONF server implementation made by Tail-f
- A 100 Gb/s connection request has been considered:
  - a baudrate of 28 Gbaud PM-QPSK supports 100 Gb/s net rate and 7% FEC
  - 31 Gbaud around 20% FEC
- The controller subscribes to the transponder Notification stream specifying, through a filter, that it is interested on pre-FEC BER exceeding 9×10^{-4} threshold.



No.	Time	Source	Destination	Protocol	Length	Info
1	0	192.168.56.103	192.168.56.102	TCP	74	53106 > 2023 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=253933 TSecr=0 WS=128
2	0.000373	192.168.56.102	192.168.56.103	TCP	74	2023 > 53106 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=203711958 TSecr=253933 WS=128
3	0.000401	192.168.56.103	192.168.56.102	TCP	66	53106 > 2023 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=253933 TSecr=203711958
4	0.000511	192.168.56.103	192.168.56.102	TCP	106	53106 > 2023 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=40 TSval=253933 TSecr=203711958
5	0.000691	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=1 Ack=41 Win=29056 Len=0 TSval=203711958 TSecr=253933
6	0.000704	192.168.56.103	192.168.56.102	TCP	337	53106 > 2023 [PSH, ACK] Seq=41 Ack=1 Win=29312 Len=271 TSval=253933 TSecr=203711958
7	0.000925	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=1 Ack=312 Win=30080 Len=0 TSval=203711958 TSecr=253933
8	0.010643	192.168.56.102	192.168.56.103	TCP	2818	2023 > 53106 [PSH, ACK] Seq=1 Ack=312 Win=30080 Len=2752 TSval=203711960 TSecr=253933
9	0.01074	192.168.56.103	192.168.56.102	TCP	66	53106 > 2023 [ACK] Seq=312 Ack=2753 Win=34816 Len=0 TSval=253936 TSecr=203711960
10	0.013463	192.168.56.103	192.168.56.102	TCP	216	53106 > 2023 [PSH, ACK] Seq=312 Ack=2753 Win=34816 Len=150 TSval=253937 TSecr=203711960
11	0.013594	192.168.56.103	192.168.56.102	TCP	1514	53106 > 2023 [ACK] Seq=462 Ack=2753 Win=34816 Len=1448 TSval=253937 TSecr=203711960
12	0.013909	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=2753 Ack=1910 Win=34048 Len=0 TSval=203711961 TSecr=253937
13	0.013941	192.168.56.103	192.168.56.102	TCP	290	53106 > 2023 [PSH, ACK] Seq=1910 Ack=2753 Win=34816 Len=224 TSval=253937 TSecr=203711961
14	0.052621	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=2753 Ack=2134 Win=36992 Len=0 TSval=203711971 TSecr=253937
15	0.05847	192.168.56.102	192.168.56.103	TCP	206	2023 > 53106 [PSH, ACK] Seq=2753 Ack=2134 Win=36992 Len=140 TSval=203711972 TSecr=253937
16	0.059472	192.168.56.103	192.168.56.102	TCP	219	53106 > 2023 [PSH, ACK] Seq=2134 Ack=2893 Win=37632 Len=153 TSval=253948 TSecr=203711972
17	0.05978	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=2893 Ack=2287 Win=39808 Len=0 TSval=203711972 TSecr=253948
18	0.059795	192.168.56.103	192.168.56.102	TCP	70	53106 > 2023 [PSH, ACK] Seq=2287 Ack=2893 Win=37632 Len=4 TSval=253948 TSecr=203711972
19	0.059936	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=2893 Ack=2291 Win=39808 Len=0 TSval=203711972 TSecr=253948
20	0.060521	192.168.56.102	192.168.56.103	TCP	206	2023 > 53106 [PSH, ACK] Seq=2893 Ack=2291 Win=39808 Len=140 TSval=203711973 TSecr=253948
21	0.060628	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [FIN, ACK] Seq=3033 Ack=2291 Win=39808 Len=0 TSval=203711973 TSecr=253948
22	0.060707	192.168.56.103	192.168.56.102	TCP	66	53106 > 2023 [FIN, ACK] Seq=2291 Ack=3034 Win=40576 Len=0 TSval=253948 TSecr=203711973
23	0.061007	192.168.56.102	192.168.56.103	TCP	66	2023 > 53106 [ACK] Seq=3034 Ack=2292 Win=39808 Len=0 TSval=203711973 TSecr=253948
24	38.3766	192.168.56.103	192.168.56.102	TCP	74	53107 > 2023 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=263527 TSecr=0 WS=128
25	38.376944	192.168.56.102	192.168.56.103	TCP	74	2023 > 53107 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=203721553 TSecr=263527 WS=128
26	38.377021	192.168.56.103	192.168.56.102	TCP	66	53107 > 2023 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=263527 TSecr=203721553
27	38.377132	192.168.56.103	192.168.56.102	TCP	106	53107 > 2023 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=40 TSval=263527 TSecr=203721553
28	38.377425	192.168.56.102	192.168.56.103	TCP	66	2023 > 53107 [ACK] Seq=1 Ack=41 Win=29056 Len=0 TSval=203721553 TSecr=263527
29	38.378372	192.168.56.102	192.168.56.103	TCP	2818	2023 > 53107 [PSH, ACK] Seq=1 Ack=41 Win=29056 Len=2752 TSval=203721553 TSecr=263527
30	38.378419	192.168.56.103	192.168.56.102	TCP	66	53107 > 2023 [ACK] Seq=41 Ack=2753 Win=34816 Len=0 TSval=263528 TSecr=203721553
31	38.383082	192.168.56.103	192.168.56.102	TCP	667	53107 > 2023 [PSH, ACK] Seq=41 Ack=2753 Win=34816 Len=601 TSval=263529 TSecr=203721553
32	38.392732	192.168.56.102	192.168.56.103	TCP	202	2023 > 53107 [PSH, ACK] Seq=2753 Ack=642 Win=30208 Len=136 TSval=203721557 TSecr=263529
33	38.429415	192.168.56.103	192.168.56.102	TCP	66	53107 > 2023 [ACK] Seq=642 Ack=2889 Win=37632 Len=0 TSval=263541 TSecr=203721557
34	70.141241	192.168.56.102	192.168.56.103	TCP	416	2023 > 53107 [PSH, ACK] Seq=2889 Ack=642 Win=30208 Len=350 TSval=203729495 TSecr=263541
35	70.141327	192.168.56.103	192.168.56.102	TCP	66	53107 > 2023 [ACK] Seq=642 Ack=3239 Win=40576 Len=0 TSval=271468 TSecr=203729495

a)

Edt-config: connection setup

```
<?xml version="1.0" encoding="UTF-8"?>
                <rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     message-id="1">
<edit-config xmlns:nc='urn:ietf:params:xml:ns:netconf:base:1.0'>
<target><running/></target><config>
<transponder xmlns="http://sssup.it/transponder"
             xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <subcarrier-module>
        <subcarrier-id>2</subcarrier-id>
        <config>
            <direction>TX</direction>
            <bit-rate>112</bit-rate>
            <baud-rate>28</baud-rate>
            <modulation xmlns:mf="http://sssup.it/modulation-formats">
                mf:pm-qpsk
            </modulation>
            <fec-in-use>
                <name xmlns:fec="http://sssup.it/fec-types">
                    fec:ldpc
                </name>
                <rate>
                    <message-length>14</message-length>
                    <block-length>15</block-length>
                </rate>
            </fec-in-use>
            <central-frequency>193100</central-frequency>
            <bandwidth>33.6</bandwidth>
            <transmitter>
                <output-power>0</output-power>
            </transmitter>
        </config>
    </subcarrier-module>
    <connections>
        <connection nc:operation="create">
            <connection-id>2</connection-id>
            <config>
                <connection-id>2</connection-id>
                <transmission-scheme>NWDM</transmission-scheme>
                <subcarrier>
                    <subcarrier-id>2</subcarrier-id>
                </subcarrier>
                <frequency-slot>
                    <n>0</n>
                    <m>3</m>
                </frequency-slot>
            </config>
        </connection>
    </connections>
</transponder>
</config></edit-config>
</rpc>
```

b)

Subscription to pre-FEC BER increase

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <create-subscription xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <stream>transponder</stream>
    <filter type="xpath" xmlns:tran="http://sssup.it/transponder"
    .select="/tran:pre-fec-ber-change[tran:pre-fec-ber>=0.0009]"/>
  </create-subscription>
</rpc>
]]>]]>
```

c)

Notification of BER increase

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1"><ok/></rpc-reply>]]>]]>
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"><eventTime>2015-10-12T13:11:58.306596+00:00</eventTime>
<pre-fec-ber-change xmlns='http://sssup.it/transponder'>
  <subcarrier-module-id>1</subcarrier-module-id>
  <pre-fec-ber>0.00096</pre-fec-ber>
</pre-fec-ber-change>
</notification>]]>]]>
```

d)

Conclusions

- This paper presented a YANG model for transponders with monitoring capabilities, slice-ability, and variable:
  - rate
  - baudrate
  - FEC
  - format
- Experiments have shown connection set up and management
  - OAM: the mechanism of Notification messages can be easily exploited to implement alarms

ACK: The work has been supported by the ORCHESTRA project.

TRANSPONDER YANG

```
module transponder {
 namespace "http://sssup.it/transponder";
 prefix tran;

  import modulation-formats {
    prefix mdfrms;
  }

  import fec-types {
    prefix fec;
  }

  import ietf-yang-types {
    prefix yang;
  }

  organization
   "Scuola Superiore Sant'Anna Network and Services Laboratory";

  description
   "This module contains a YANG definitions for configuring Optical Transponder.";

  revision 2015-09-15 {
    description "Initial Revision.";
    reference   "TBD";
  }

  typedef transmission-type {
    description "The transmission method";
    type enumeration {
      enum NWDM;
      enum O-OFDM;
      enum TFP;//Time-frequency packing
    }
  }
```

```
  typedef direction-type {
    description "Indicates the direction";
```

TRANSPONDER YANG

```
typedef direction-type {
  description "Indicates the direction";
  type enumeration {
    enum TX;
    enum RX;
  }
}

typedef bit-rate-type {
  type decimal64 {
    fraction-digits 3;
    range "0..max";
  }
  units "Gb/s";
}

typedef baud-rate-type {
  type decimal64 {
    fraction-digits 3;
    range "0..max";
  }
  units "Gbaud";
}

typedef modulation-type {
  type identityref {
    base mdfrms:modulation-format;
  }
}

typedef fec-type {
  type identityref {
    base fec:fec-type;
  }
}
```

TRANSPONDER YANG

```
typedef frequency-ghz-type {
  type decimal64 {
    fraction-digits 8;
    range "0..max";
  }
  units "GHz";
}


grouping fec-config {
  description "Configuration data for forward error correction";

  container fec-in-use {
    description "FEC in use";
    presence "Enables FEC";

    leaf name {
      type fec-type;
    }

    container rate {
      description
      "The code rate is given by message-length/block-length";
      leaf message-length {
        type int16 {
          range "1..max";
        }
      }
      leaf block-length {
        type int16 {
          range "1..max";
        }
      }
      must "block-length >= message-length" {
          error-message "block-length must be greater or equal to message-length";
      }
    }//container rate

  }//container fec-in-use
}//grouping fec-config
```

TRANSPONDER YANG

```
grouping fec-state {
  description "Operational state data for forward error correction";
  container supported-fec {
    description "List of supported FEC schemes";
    leaf-list fec {
      type fec-type;
    }
  }//supported
}//grouping fec-state



grouping transmitter-config {
  description "Configuration data for the transmitter";
  leaf output-power {
    description "launch power at the transmitter";
    type int16;
    units "dBm";
  }
}//grouping transmitter-config

grouping transmitter-state {
  description "Operational state data for the transmitter";
}



grouping receiver-config {
  description "Configuration data for the receiver";

  leaf local-oscillator {
    type frequency-ghz-type;
  }

  leaf sampling-rate {
    description "Minimum hardware requirements in terms of sampling rate";
    type uint32;
    units "GS/s";
  }
```

TRANSPONDER YANG

```
    leaf analog-bw {
      description "Minimum hardware requirements in terms of analog bandwidth";
      type frequency-ghz-type;
    }

  }//grouping receiver-config

  grouping receiver-state {
    description "Operational state data for the receiver";

    leaf input-power {
      description "per-channel received optical power at the receiver";
      type int16;
      units "dBm";
    }

    leaf pre-fec-ber {
      description
      "Pre-FEC Bit Error Rate.";
      type decimal64 {
        fraction-digits 18;
        range "0..max";
      }
    }

    leaf sample-variance {
      type decimal64 {
        fraction-digits 18;
        range "0..max";
      }
      reference
        "http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7035536";
    }
```

TRANSPONDER YANG

```
    leaf pmd {
      description
      "Polarization Mode Dispersion.";
      type decimal64 {
        fraction-digits 8;
        range "0..max";
      }
      units "ps/(km)^0.5";
    }

    leaf cd {
      description
      "Chromatic Dispersion.";
      type decimal64 {
        fraction-digits 5;
      }
      units "ps/nm/km";
    }

    leaf q-factor{
      type decimal64 {
        fraction-digits 5;
      }
      units "dB";
    }
  }//grouping receiver-state


  grouping subcarrier-module-config {
    description "Configuration data for the optical subcarrier-module";

    leaf direction {
      description "Defines whether the subcarrier is received or transmitted";
      type direction-type;
    }

    leaf bit-rate {
      description "The bit-rate in use";
      type bit-rate-type;
    }
```

TRANSPONDER YANG

```
    leaf baud-rate {
      description "The baud-rate in use";
      type baud-rate-type;
    }

    leaf modulation {
      description "Modulation format in use";
      type modulation-type;
    }
    uses fec-config;
    leaf central-frequency {
      description
        "The central frequency of the subcarrier.";
      type frequency-ghz-type;
    }

    leaf bandwidth {
      description
        "The bandwidth occupied.";
      type frequency-ghz-type;
    }
  }//subcarrier-module-config

  grouping subcarrier-module-state {
    description "Operational state data for the optical subcarrier-module";
    container supported-bit-rates{
      description "List of supported bit-rates";
      leaf-list bit-rate {
        description "the bit rate value";
        type bit-rate-type;
      }
    }

    container supported-baud-rates {
      description "List of supported baud-rates";
      leaf-list baud-rate {
        description "the baud rate value";
        type baud-rate-type;
      }
    }
```

TRANSPONDER YANG

```
    container supported-modulations {
      description "List of supported modulation formats";
      leaf-list modulation {
        description "Name of the supported modulation";
        type modulation-type;
      }
    }
    uses fec-state;
  }//subcarrier-module-state

  grouping subcarrier-module {
    description "Top-level grouping for optical subcarrier-module";

    container config {
      description
        "Configuration data for subcarrier-module";
      uses subcarrier-module-config;

      container transmitter {
        when "../direction = 'TX'";
        uses transmitter-config;
      }

      container receiver {
        when "../direction = 'RX'";
        uses receiver-config;
      }
    }
    container state {
      config false;
      description
        "Operational state data for subcarrier-module";
      uses subcarrier-module-config;
      uses subcarrier-module-state;

      container transmitter {
        when "../direction = 'TX'";
        uses transmitter-config;
        uses transmitter-state;
      }
```

TRANSPONDER YANG

```
    container receiver {
      when "../direction = 'RX'";
      uses receiver-config;
      uses receiver-state;
    }
  }

}//subcarrier-module

grouping connection-config {
  description "Configuration data for a connection";
  leaf connection-id{
    type uint32;
  }

  leaf transmission-scheme {
    description "The scheme adopted for the transmission";
    type transmission-type;
  }

  list subcarrier {
    description "List of ids of the involved subcarriers";
    key "subcarrier-id";
    leaf subcarrier-id {
      type leafref {
        path "/tran:transponder/tran:subcarrier-module/tran:subcarrier-id";
      }
    }
  }

  container frequency-slot {
    description
      "The frequency range allocated to a slot
      within the flexible grid and unavailable to other slots.  A
      frequency slot is defined by its nominal central frequency and its
      slot width.";
    reference "draft-ietf-ccamp-flexi-grid-fwk-07";
```

TRANSPONDER YANG

```
    leaf nominal-central-frequency-granularity {
      description
        "It is the spacing between allowed nominal central frequencies.";
      type frequency-ghz-type;
      default 6.25;
    }//leaf nominal-central-frequency-granularity

    leaf slot-width-granularity {
      description "It is the minimum slot width.";
      type frequency-ghz-type;
      default 12.5;
    }//leaf slot-width-granularity

    leaf n {
      description
        "n gives the nominal central frequency (ncf) using the following formula:
        ncf = 193.1THz + n x nominal-central-frequency-granularity[THz].";
      type int16;
      mandatory true;
    }//leaf n
    leaf m {
      description
        "m gives the slot width. A slot width is constrained to be
        m x slot-width-granularity";
      type int16 {
        range "1..max";
      }
      mandatory true;
    }//leaf m
  }//container frequency-slot

  leaf source-address {
    description "The IP address of the source node";
    Type inet:ip-address;
  }//leaf source-address
  leaf destination-address {
    description "The IP address of the destination node";
    Type inet:ip-address;
  }//leaf source-address
}//grouping connection-config
```

TRANSPONDER YANG

```
grouping connection-state {
  description "Operational state data for a connection";
}//grouping connection-state

grouping connections {
  description "List of all connections served by the transponder";
  list connection {
    key "connection-id";
    leaf connection-id {
      description "references the configured connection-id";
      type leafref {
        path "../config/connection-id";
      }
    }
    container config {
      description "Configuration parameters for connection";
      uses connection-config;
    }

    container state {
      config false;
      description "State variables for connection";
      uses connection-config;
      uses connection-state;
    }
  }//list connection
}//grouping connections
```

TRANSPONDER YANG

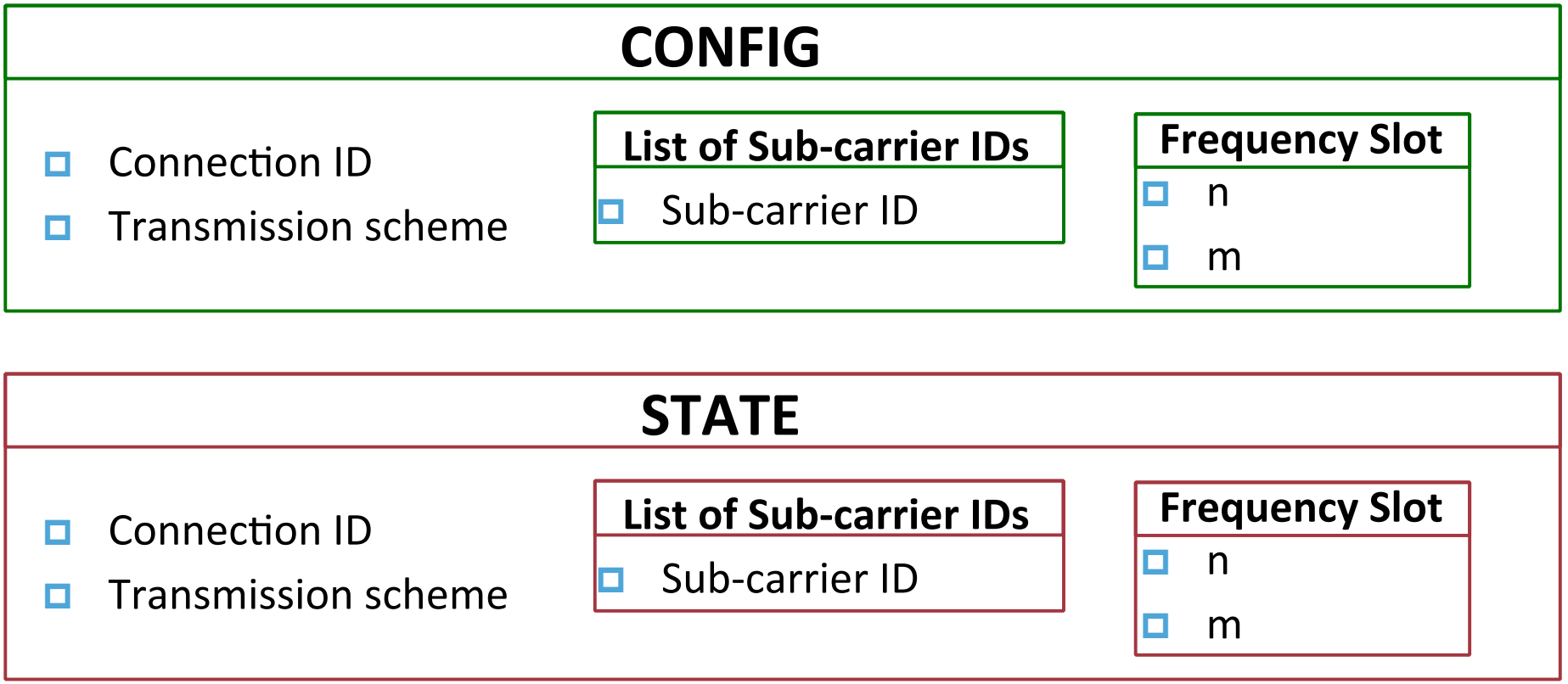
```
//---------- MAIN TREE ------------//
container transponder {
    list subcarrier-module {
      description
        "List of all the subcarrier modules installed in the transponder";
      key "subcarrier-id";
      leaf subcarrier-id {
        type uint32;
      }
      uses subcarrier-module;
    }

    leaf slice-ability-support {
      when "count(../subcarrier-module) > 1";
      type boolean;
      config false;
      description "Determines if the transponder is slice-able.";
    }


    leaf node-id {
      description "ID of the node where the transponder is installed";
      type uint16;
    }

    leaf add-drop-id {
      description "Add/drop ID inside the node";
      type uint16;
    }

    container connections {
      uses connections;
    }
}
```

TRANSPONDER YANG

```
  //------------- NOTIFICATIONS ----------------//
  notification pre-fec-ber-change {
    leaf subcarrier-module-id {
      description
        "An existing subcarrier-module in the list";
      type leafref {
        path "/tran:transponder/tran:subcarrier-module/tran:subcarrier-id";
      }
      mandatory true;
    }
    leaf pre-fec-ber {
      type leafref {
        path "/transponder/subcarrier-module[subcarrier-id=current()/../subcarrier-module-id]/state/receiver/pre-fec-ber";
      }
      mandatory true;
    }
  }
  notification pmd-change {
    leaf subcarrier-module-id {
      description
        "An existing subcarrier-module in the list";
      type leafref {
        path "/tran:transponder/tran:subcarrier-module/tran:subcarrier-id";
      }
      mandatory true;
    }
    leaf pmd {
      type leafref {
        path "/transponder/subcarrier-module[subcarrier-id=current()/../subcarrier-module-id]/state/receiver/pmd";
      }
      mandatory true;
    }
  }
}//module transponder
```

email: nicola.sambo@sssup.it

CONNECTION


CONFIG
Connection ID
Transmission scheme
List of Sub-carrier IDs
Sub-carrier ID
Frequency Slot
n
m
STATE
Connection ID
Transmission scheme
List of Sub-carrier IDs
Sub-carrier ID
Frequency Slot
n
m

CONNECTION Example

```
<connection>
    <connection-id>1</connection-id>
    <config>
        <connection-id>1</connection-id>
        <transmission-scheme>
         NWDM
        </transmission-scheme>
        <subcarrier>
            <subcarrier-id>1</subcarrier-id>
            <subcarrier-id>2</subcarrier-id>
        </subcarrier>
        <frequency-slot>
            <n>0</n>
            <m>6</m>
        </frequency-slot>
    </config>
    <state>
        <connection-id>1</connection-id>
        <transmission-scheme>
            NWDM
        </transmission-scheme>
        <subcarrier>
            <subcarrier-id>1</subcarrier-id>
            <subcarrier-id>2</subcarrier-id>
        </subcarrier>
        <frequency-slot>
            <n>0</n>
            <m>6</m>
        </frequency-slot>
    </state>
</connection>
```