TeraFlow SDN by ETSI

FLEX-SCALE – Optical SDN controller

Nicola Sambo, Andrea Sgambelluri

Scuola Superiore Sant'Anna (Italy), CNIT (Italy)

October 2023

Optical SDN controller

TeraFlow SDN

Objective: control an optical network including:

- Parallel fibers
- Multi-band (e.g., S+C+L)

Optical SDN controller

Objective: control an optical network including:

- Parallel fibers
- Multi-band (e.g., S+C+L)

Definitions:

- Hereafter **band** will be named **window**
- For **band** we consider **a wide portion of spectrum** that
  - can include more channels
  - can be an entire window (e.g., S) or a portion of a window

Optical SDN controller

Objective: control an optical network including:

- Parallel fibers
- Multi-window (e.g., S+C+L)

Supporting:

- Wavelength switching
- Band switching

Definitions:

- Hereafter **band** will be named **window**
- For **band** we consider **a wide portion of spectrum** that
  - can include more channels
  - can be an entire window (e.g., S) or a portion of a window

Optical SDN controller

Objective: control an optical network including:

- Parallel fibers
- Multi-window (e.g., S+C+L)

Supporting:

- Wavelength switching
- Band switching

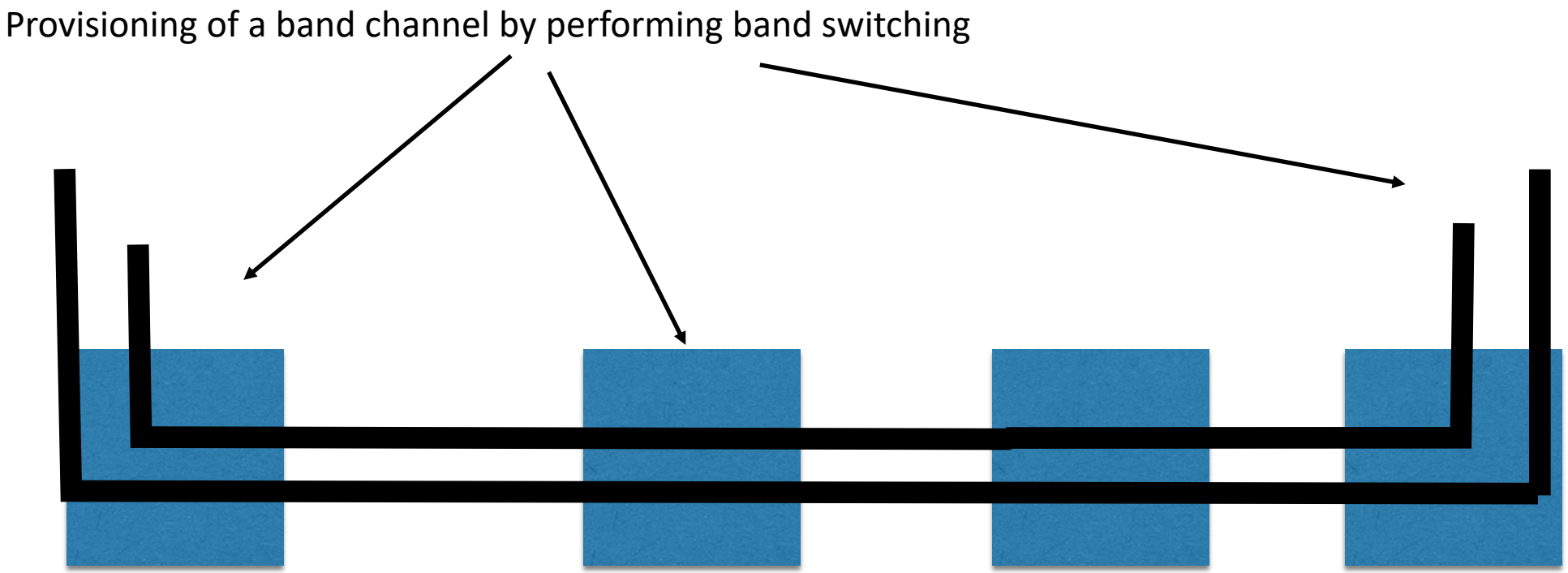
Definitions:

- Hereafter **band** will be named **window**
- For **band** we consider **a wide portion of spectrum** that
  - can include more channels
  - can be an entire window (e.g., S) or a portion of a window

Based on service request, the Optical controller:

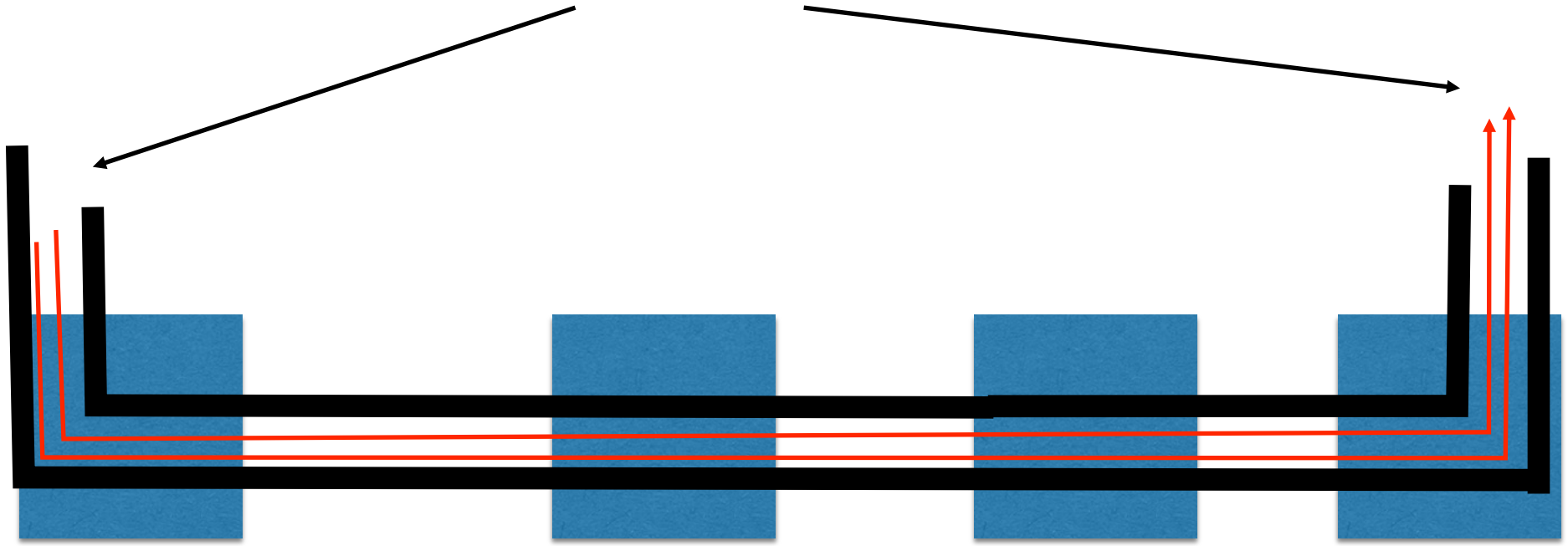
- Performs resource allocation
- Triggers device configuration based on NETCONF/YANG

Example of band switching (1/4)



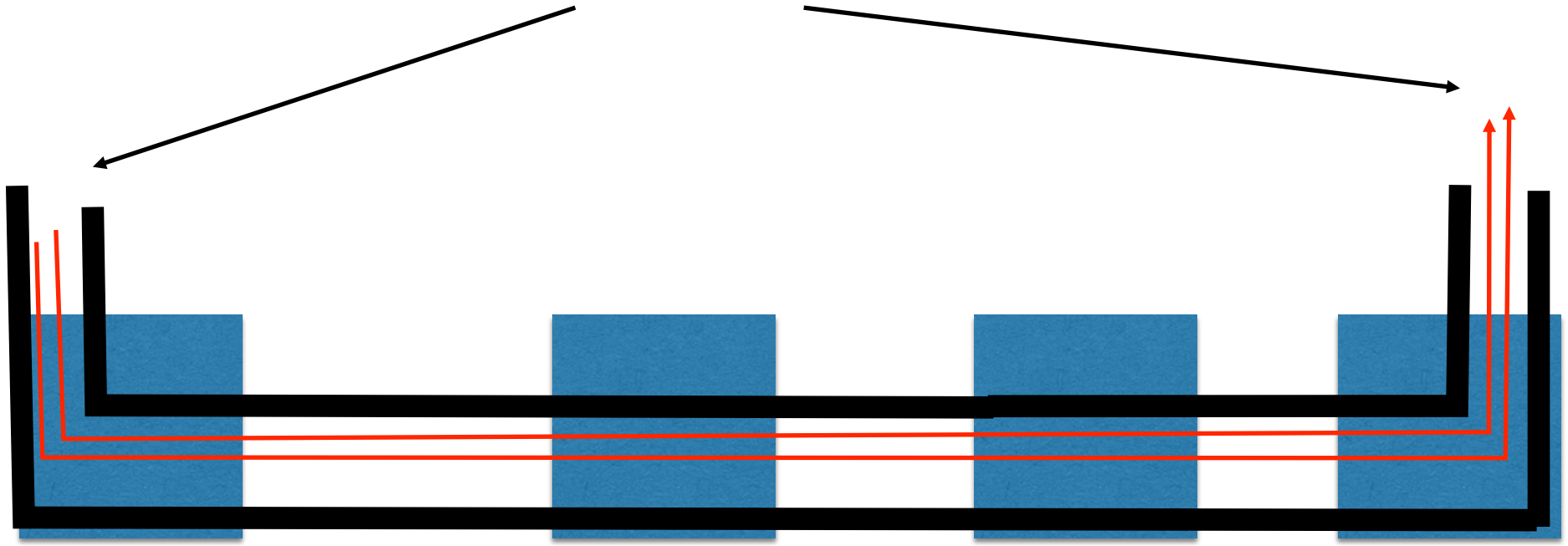
Example of band switching (2/4)

Provisioning of optical wavelengths channels by adding/dropping them within the band channel at the edges

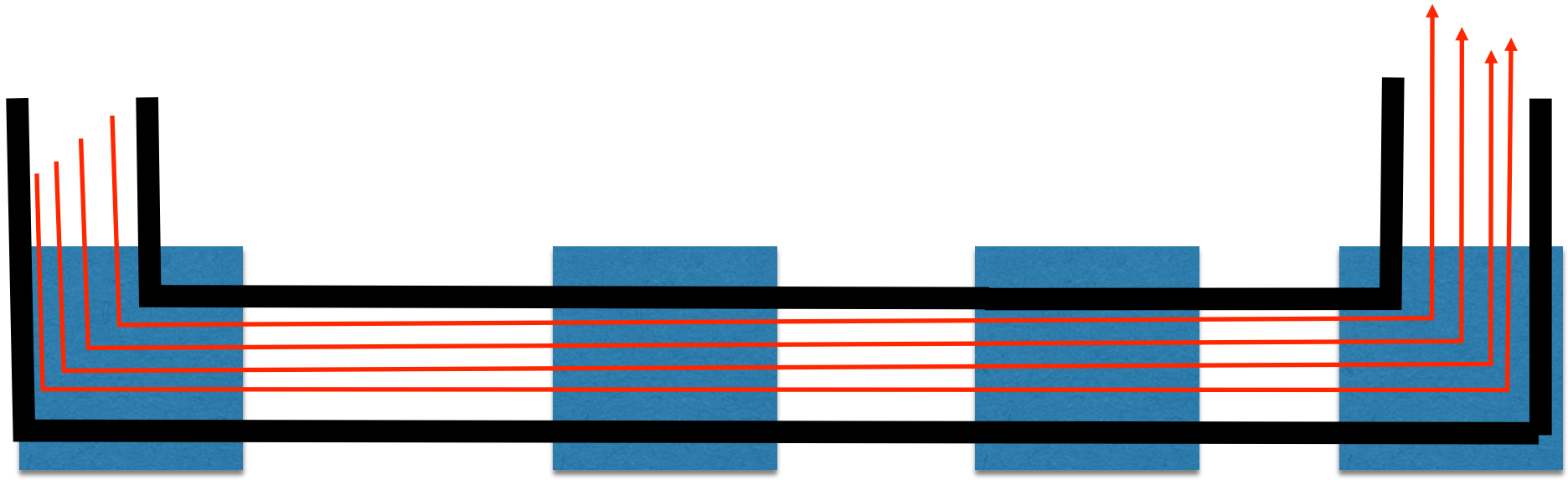


Example of band switching (3/4)

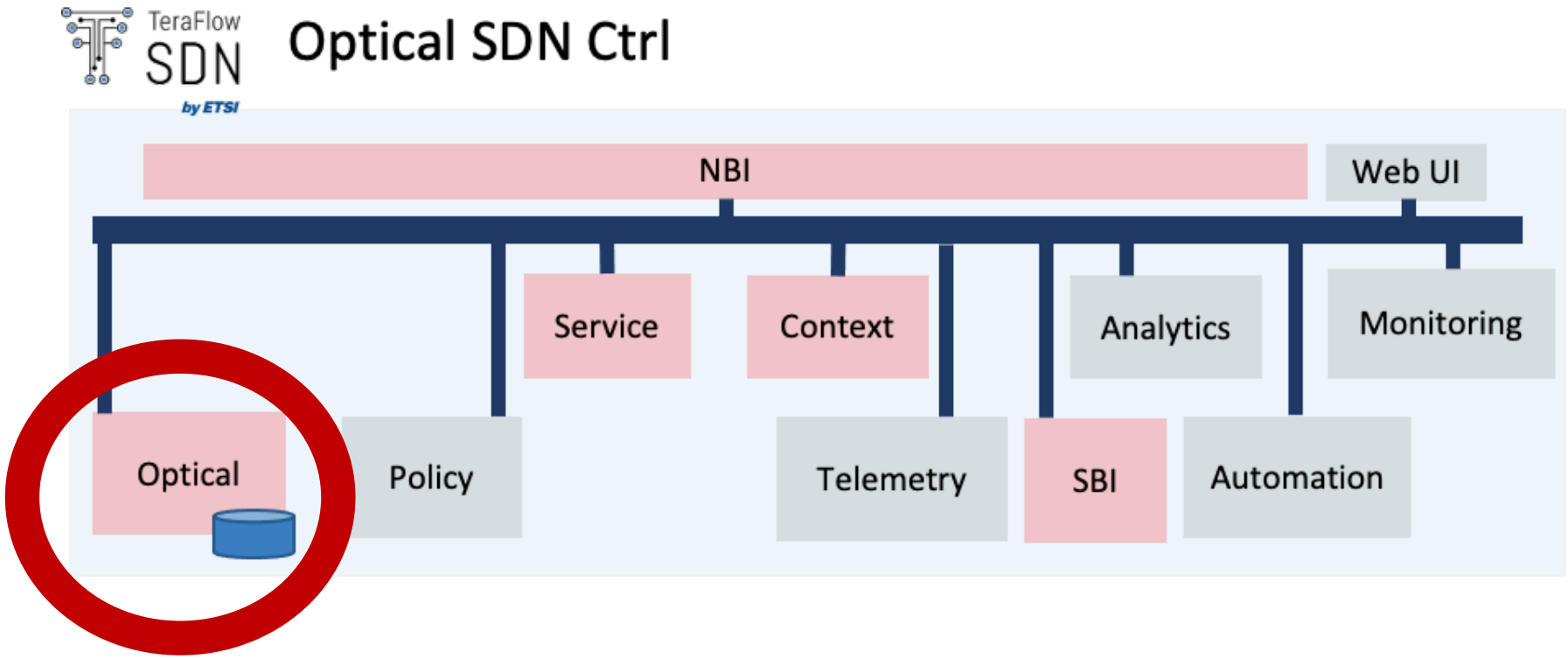
Possibility to reconfigure the band increasing the bandwidth to accommodate more channels



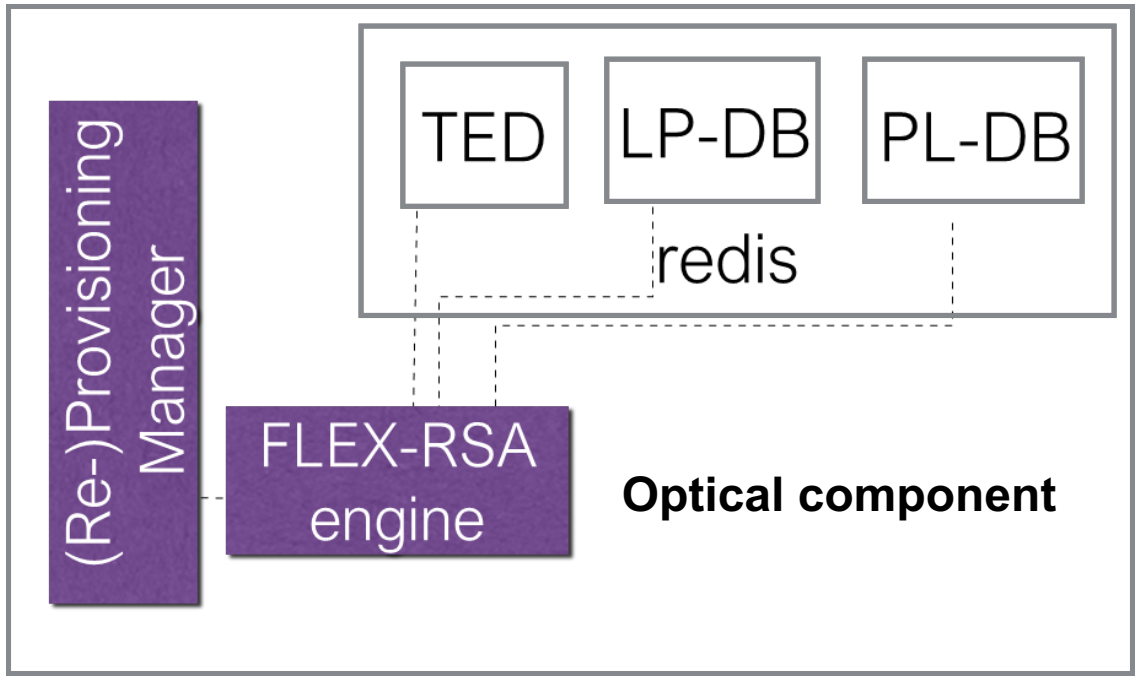
Example of band switching (4/4)



Optical controller



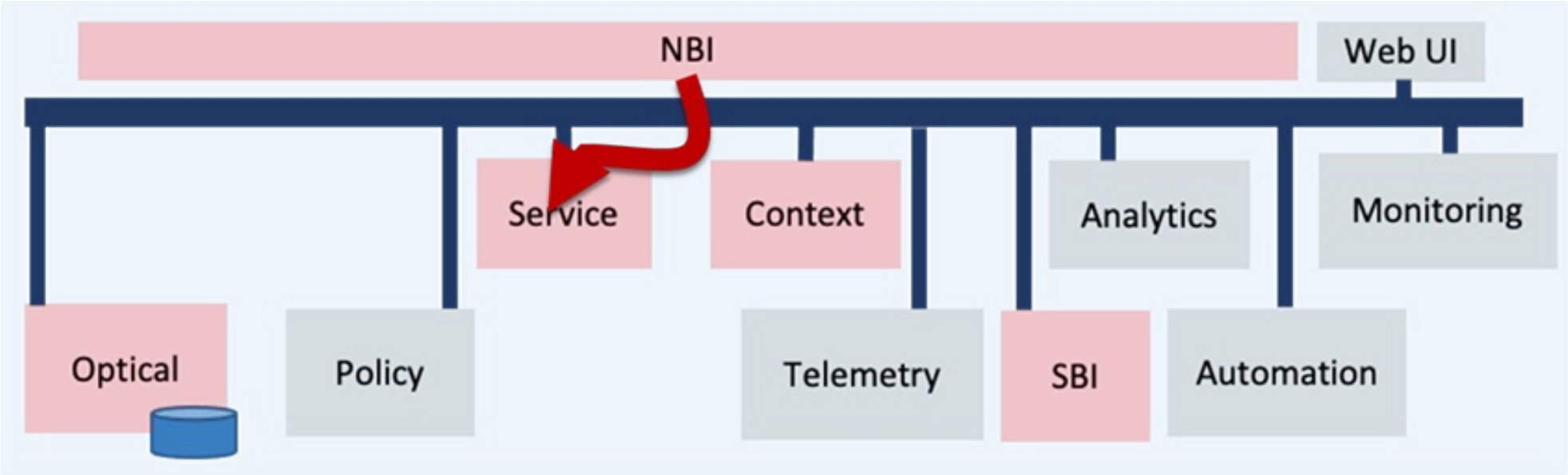
Optical component



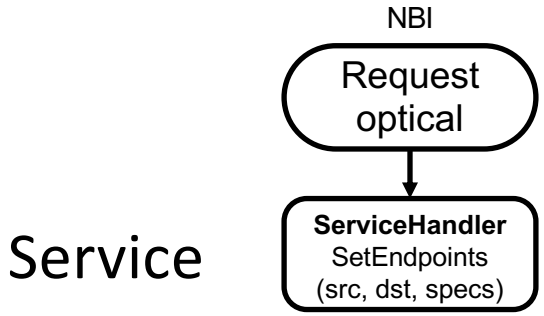
- **TED:** traffic engineering database
- **LP-DB:** label switched path database
- **PL-DB:** physical layer database

- **(Re-)Provisioning Manager:** triggers resource allocation for
  - Provisioning
  - Re-optimization
  - Recovery
- **FLEX-RSA** (routing and spectrum assignment) engine allocating resources:
  - Route
  - Performs quality of transmission estimation
  - Selects the Operational Mode (combination of symbol rate, FEC, modulation format → line rate)
  - Selects the number of channels to meet rate requirements
  - Portion of spectrum
  - Accounts for parallel fibers and band swithing

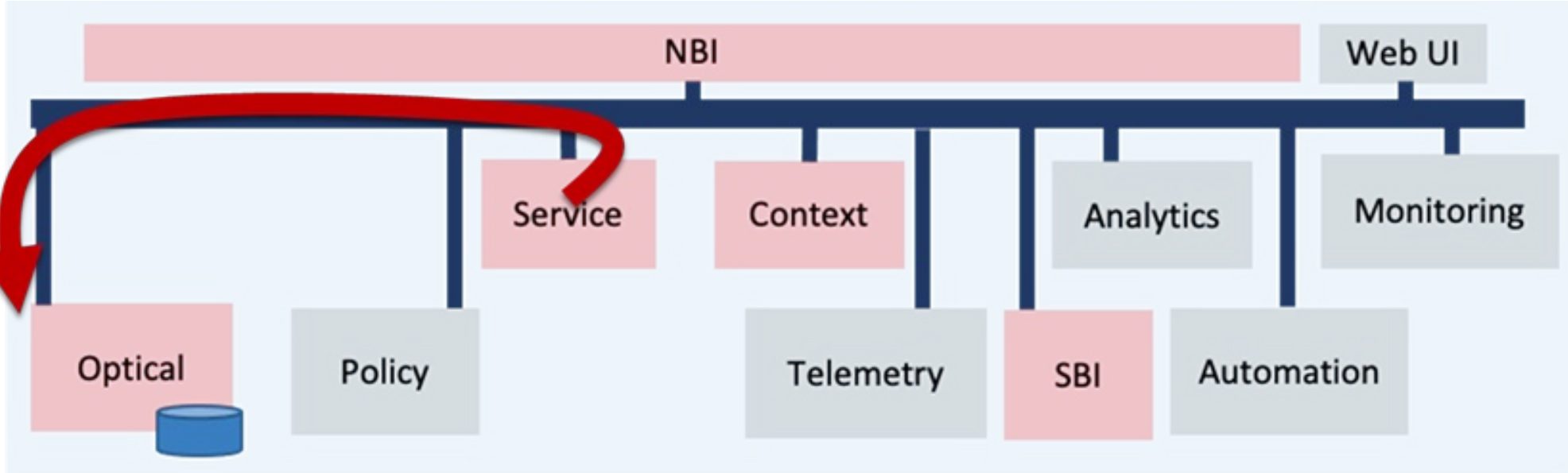
Optical controller workflow



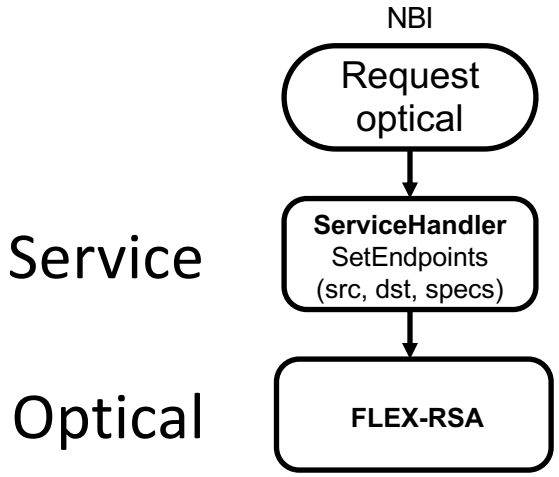
Optical controller workflow



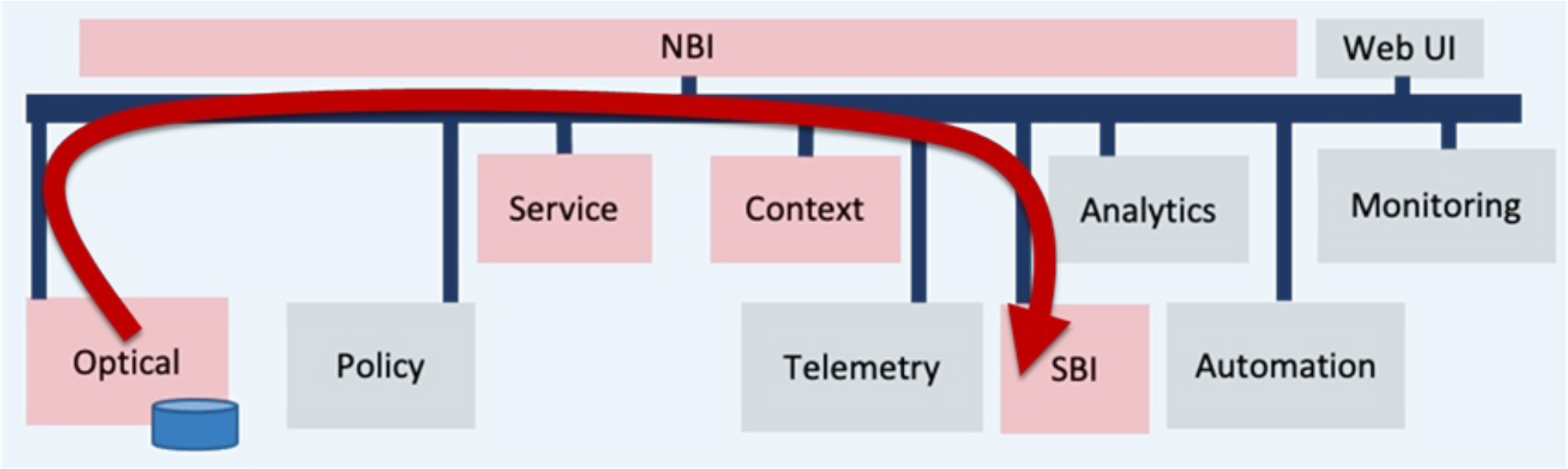
Optical controller workflow



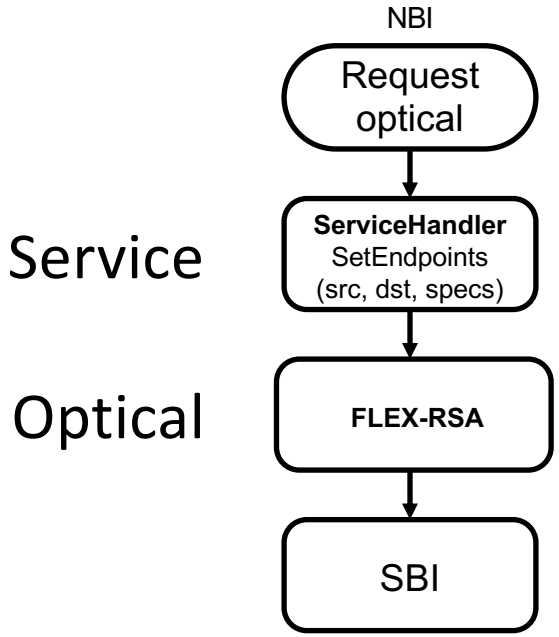
Optical controller workflow



Optical controller workflow



Optical controller workflow



The **output of FLEX-RSA** is then **translated** into the **devices** to be configured (e.g., nodes and transceivers) together with their **proper configuration parameters** values and propagated to the SBI

SBI based on NETCONF plugin

YANG model for transceiver

TeraFlow SDN

```
+rw terminal-device
  +--rw config
  +--ro state
  +--rw logical-channels
  |  +--rw channel* [index]
  |     +--rw index
  |     +--rw config
  |     |  +--rw index
  |     |  +--rw description
  |     |  +--rw admin-state
  |     |  +--rw rate-class
  |     |  +--rw trib-protocol
  |     |  +--rw logical-channel-type
  |     +--ro state
  |     |  +--rw index
  |     |  +--rw description
  |     |  +--rw admin-state
  |     |  +--rw rate-class
  |     |  +--rw trib-protocol
  |     |  +--rw logical-channel-type
  |     |  +--ro link-state
  |     +--rw otn
  |     |  +--ro state
  |     |     +--ro pre-fec-ber
  |     |     +--ro post-fec-ber
  |     |     +--ro q-value
  |     |     +--ro esnr
  |     +--rw logical-channel-assignments
  |        +--rw assignment* [index]
  |           +--rw index
  |           +--rw config
  |           +--ro state
+rw components
  +--rw component* [name]
     +--rw config
     |  +--rw frequency
     |  +--rw target-output-power
     |  +--rw operational-mode
     +--ro state
        +--ro output-power
        +--ro input-power
        +--ro laser-bias-current
        +--ro chromatic-dispersion
        +--ro polarization-mode-dispersion
        +--ro second-order-polarization-mode-dispersion
        +--ro polarization-dependent-loss
```

OpenConfig

```
<components xmlns="http://openconfig.net/yang/platform">
  <component>
    <name>channel-1</name>
    <optical-channel xmlns="http://openconfig.net/yang/terminal-device">
      <config>
        <frequency>193900000</frequency>
        <target-output-power>0</target-output-power>
        <operational-mode>3</operational-mode>
      </config>
    </optical-channel>
  </component>
</components>
```

YANG model for node

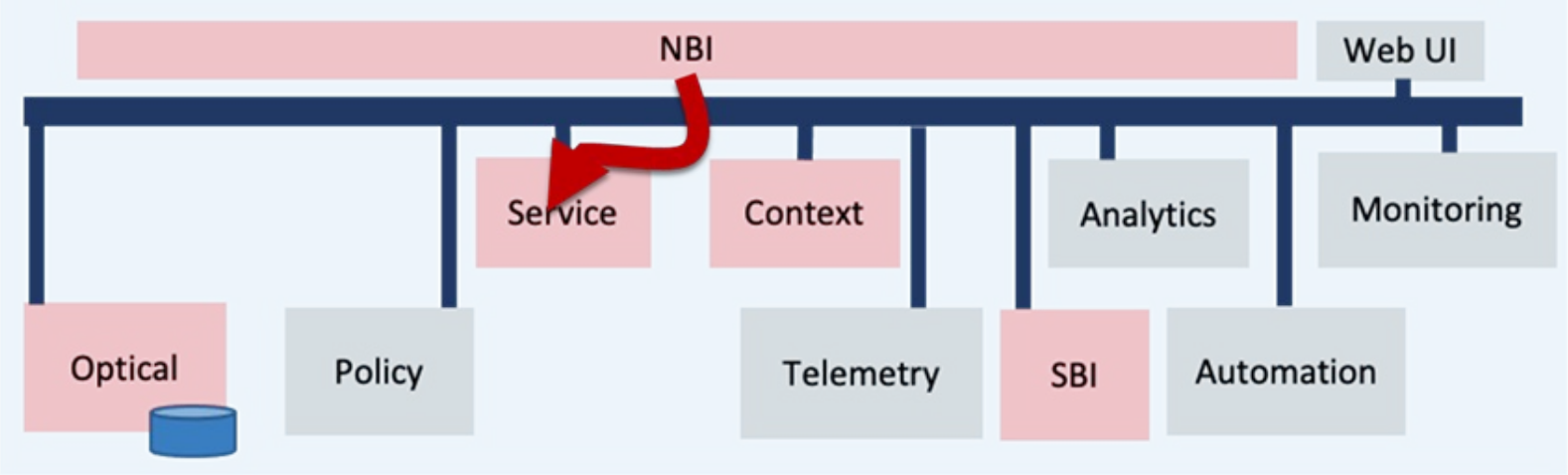
```
module: openconfig-wavelength-router
  +--rw wavelength-router
     +--rw media-channels
        |  +--rw channel* [index]
        |     +--rw index                    -> ../config/index
        |     +--rw config
        |     |  +--rw index?                     uint32
        |     |  +--rw name?                      string
        |     |  +--rw lower-frequency?           oc-opt-types:frequency-type
        |     |  +--rw upper-frequency?           oc-opt-types:frequency-type
        |     |  +--rw admin-status?              oc-opt-types:admin-state-type
        |     |  +--rw super-channel?             boolean
        |     |  +--rw super-channel-parent?      -> ../../../channel/config/index
        |     |  +--rw ase-control-mode?          identityref
        |     |  +--rw ase-injection-mode?        enumeration
        |     |  +--rw ase-injection-threshold?   decimal64
        |     |  +--rw ase-injection-delta?       decimal64
        |     |  +--rw attenuation-control-mode?  identityref
        |     +--ro state
        |     |  +--ro index?                     uint32
        |     |  +--ro name?                      string
        |     |  +--ro lower-frequency?           oc-opt-types:frequency-type
        |     |  +--ro upper-frequency?           oc-opt-types:frequency-type
        |     |  +--ro admin-status?              oc-opt-types:admin-state-type
        |     |  +--ro super-channel?             boolean
        |     |  +--ro super-channel-parent?      -> ../../../channel/config/index
        |     |  +--ro ase-control-mode?          identityref
        |     |  +--ro ase-injection-mode?        enumeration
        |     |  +--ro ase-injection-threshold?   decimal64
        |     |  +--ro ase-injection-delta?       decimal64
        |     |  +--ro attenuation-control-mode?  identityref
        |     |  +--ro oper-status?               enumeration
        |     |  +--ro ase-status?                enumeration
        |     +--rw source
        |     |  +--rw config
        |     |  |  +--rw port-name?   -> /oc-platform:components/component/name
        |     |  +--ro state
        |     |     +--ro port-name?   -> /oc-platform:components/component/name
        |     +--rw dest
        |     |  +--rw config
        |     |  |  +--rw port-name?   -> /oc-platform:components/component/name
        |     |  +--ro state
        |     |     +--ro port-name?   -> /oc-platform:components/component/name
        |     +--rw spectrum-power-profile
        |        +--rw distribution* [lower-frequency upper-frequency]
        |           +--rw lower-frequency    -> ../config/lower-frequency
        |           +--rw upper-frequency    -> ../config/upper-frequency
        |           +--rw config
        |           |  +--rw lower-frequency?   oc-opt-types:frequency-type
        |           |  +--rw upper-frequency?   oc-opt-types:frequency-type
        |           |  +--rw target-power?      decimal64
        |           +--ro state
        |              +--ro lower-frequency?   oc-opt-types:frequency-type
        |              +--ro upper-frequency?   oc-opt-types:frequency-type
        |              +--ro target-power?      decimal64
```

```
+--rw optical-bands
   +--rw optical-band* [index]
      +--rw index?    -> ../config/index
      +--rw config
      |  +--rw index?             uint32
      |  +--rw name?              string
      |  +--rw lower-frequency?   oc-opt-types:frequency-type
      |  +--rw upper-frequency?   oc-opt-types:frequency-type
      |  +--rw admin-status?      oc-opt-types:admin-state-type
      |  +--rw fiber-parent?      -> ../../../../fibers/fiber/config/index
      +--ro state
      |  +--ro index?             uint32
      |  +--ro name?              string
      |  +--ro lower-frequency?   oc-opt-types:frequency-type
      |  +--ro upper-frequency?   oc-opt-types:frequency-type
      |  +--ro admin-status?      oc-opt-types:admin-state-type
      |  +--ro fiber-parent?      -> ../../../../fibers/fiber/config/index
      |  +--ro oper-status?       enumeration
      +--rw source
      |  +--rw config
      |  |  +--rw port-name?   -> /oc-platform:components/component/name
      |  +--ro state
      |     +--ro port-name?   -> /oc-platform:components/component/name
      +--rw dest
         +--rw config
         |  +--rw port-name?   -> /oc-platform:components/component/name
         +--ro state
            +--ro port-name?   -> /oc-platform:components/component/name
```

Thanks to Michael Enrico (HUBER+SUHNER Polatis)
Reference: FLEX-SCALE D5.1, «Deliverable D5.1 – Preliminary design of sustainable transport SDN control enablers for 6G», 2023

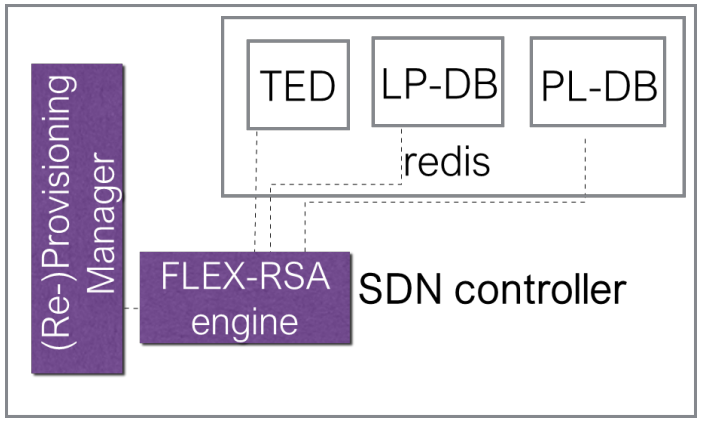
Impact on TeraFlow

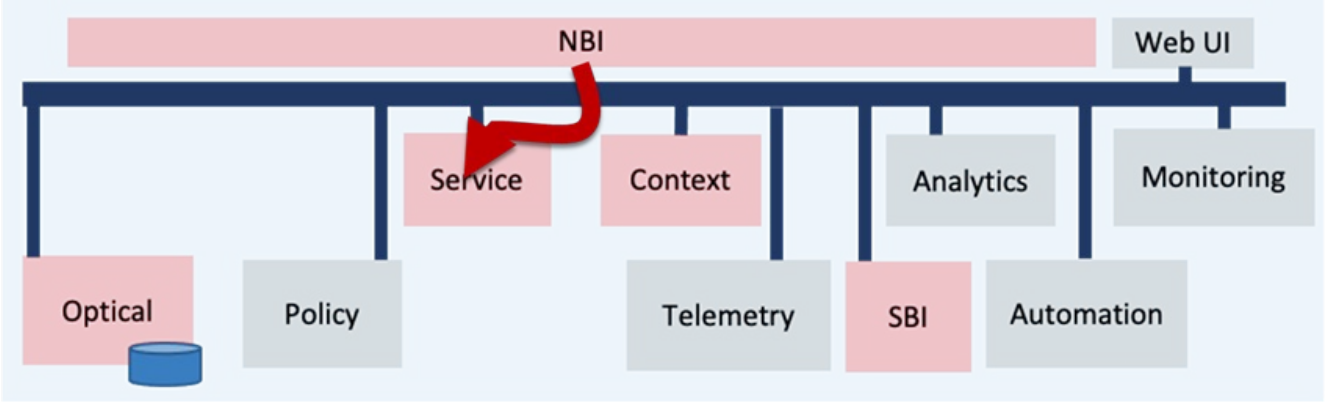
- NBI: extended to accept a request for optical service
- Service component: extended to trigger the optical component
- Optical component: introduced
- SBI: extended to control SDM/multi-band optical network supporting band switching



Status

- NBI: in progress (JSON, REST)
- Service component: in progress
- Optical component: first implementation handling parallel fibers and multi-band
- SBI: in progress





TeraFlow SDN by ETSI

FLEX-SCALE

n.sambo@santannapisa.it