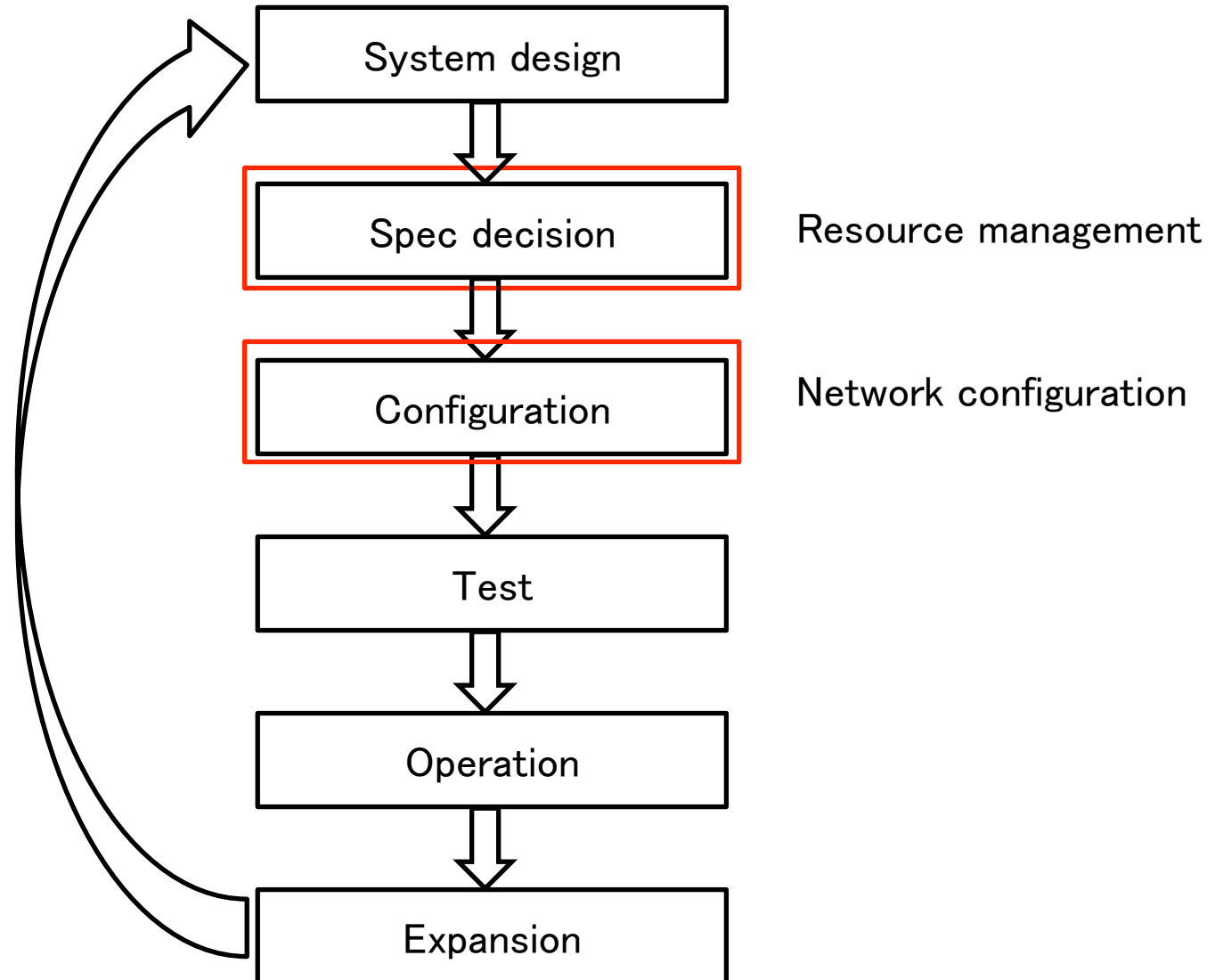IIJ
Internet Initiative Japan

NSAC for datacenter using NETCONF
（Network System Automatic Control）

Ray Atarashi

ray@iijlab.net

Motivation

- ■ Automatic management system for network system in a Data Center/Cloud

- ■ Operator need to manage computer servers and network equipments

- ■ We need to configure the system automatically to manage mega datacenter in low cost

- ■ Evaluation of NETCONF product by AlaxalA network

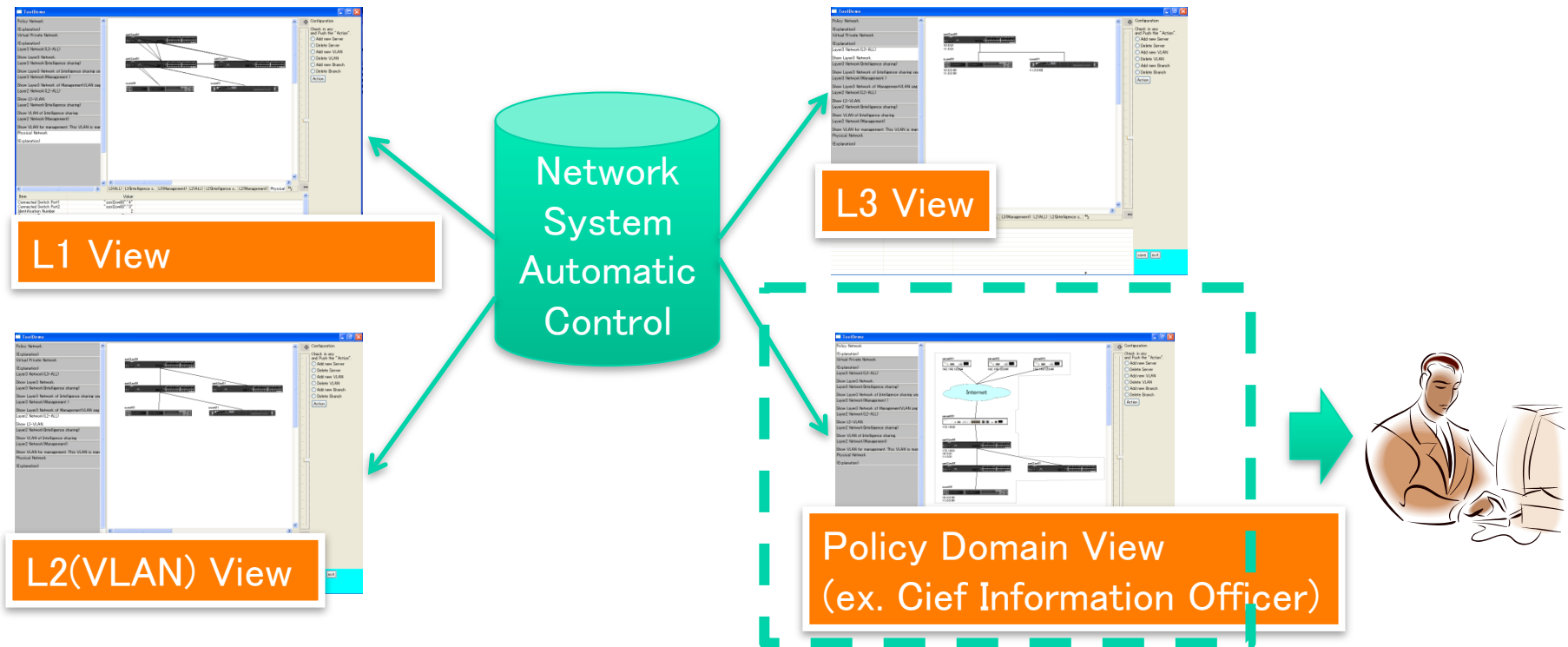 - ◆ AlaxalA provides Java API to use NETCONF

The management model in Data Center

```
        ┌─────────────────────┐
        │    System design    │
        └─────────────────────┘
                  │
                  ▼
        ┌─────────────────────┐
        │    Spec decision    │   Resource management
        └─────────────────────┘
                  │
                  ▼
        ┌─────────────────────┐
        │    Configuration    │   Network configuration
        └─────────────────────┘
                  │
                  ▼
        ┌─────────────────────┐
        │        Test         │
        └─────────────────────┘
                  │
                  ▼
        ┌─────────────────────┐
        │     Operation       │
        └─────────────────────┘
                  │
                  ▼
        ┌─────────────────────┐
        │     Expansion       │
        └─────────────────────┘
```

What we did

- **Defined VLAN datamodel by XML Schema**

- **Describe the system topology using XML**

- Built database for topology, network resource, configuration, and scenario

- Implement GUI system to show a network system
  - Multi view GUI for each layer

- Defined a few scenario
  - Add/delete server
  - Add/delete VLAN

- Programming language: Java

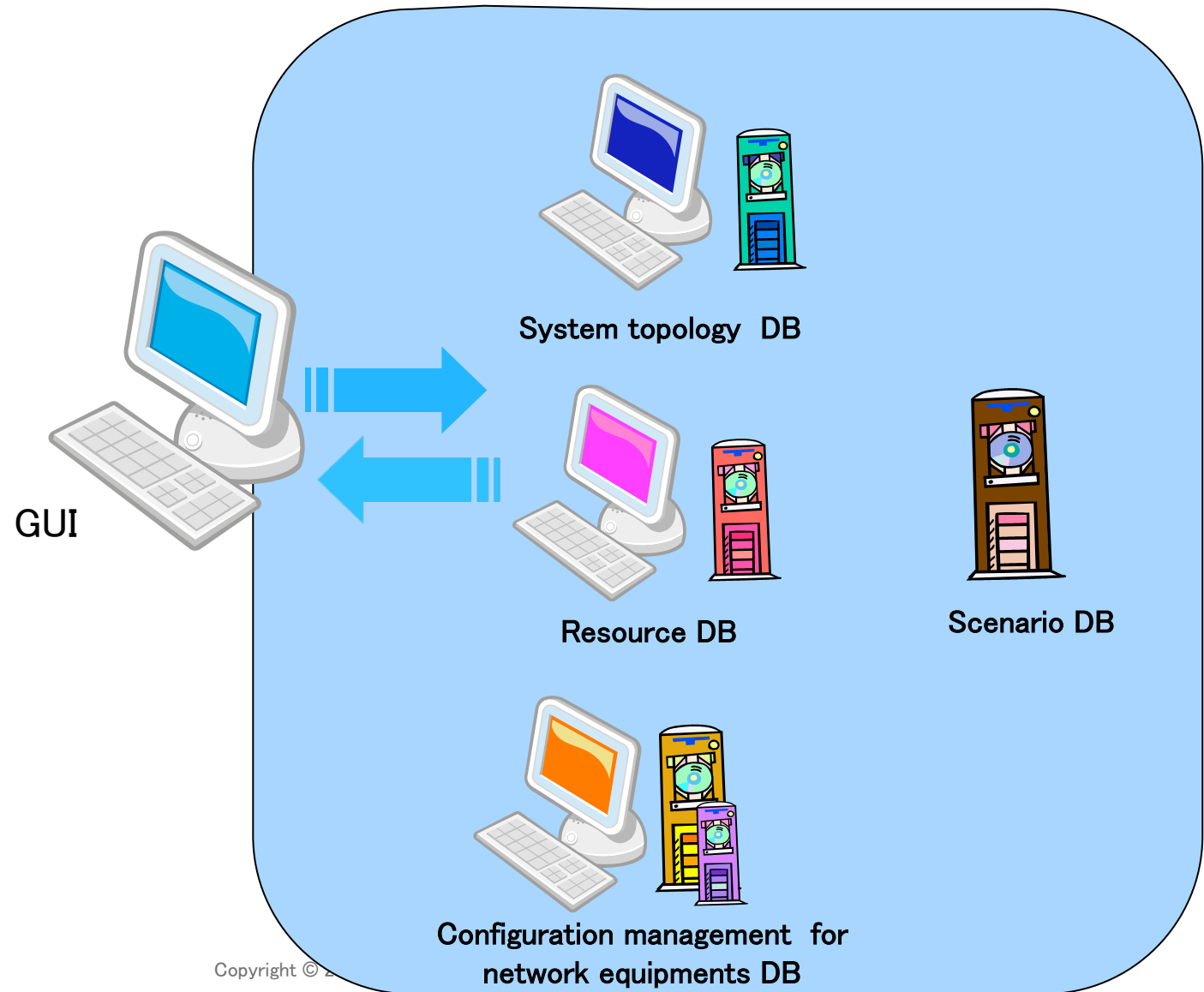- Assumption: all hardware is already mounted and installed in the datacenter. We can change the system resource by configuration.

Multi view

L1 View

L2(VLAN) View

Network System Automatic Control

L3 View

Policy Domain View
(ex. Cief Information Officer)

VLAN model（XML Schema）

```
<xsd:element name="tagged_vlan_domain" maxOccurs="unbounded">
   <xsd:complexType>
  <xsd:sequence>
   <xsd:element name="tagged_vlan_domain_id" type="xsd:string" />
   <xsd:element name="tagged_vlan" maxOccurs="4096">
    <xsd:complexType>
    <xsd:sequence>
     <xsd:element name="vlan_id">
      <xsd:simpleType>
       <xsd:restriction base="xsd:integer">
        <xsd:minInclusive value="1" />
        <xsd:maxInclusive value="4096" />
       </xsd:restriction>
      </xsd:simpleType>
     </xsd:element>
     <xsd:element name="vlan_name" type="xsd:string" />
     <xsd:element name="network_interface_list">
```

Network System Automatic Control（NSAC）

SI/operators

GUI

System topology DB

Resource DB

Scenario DB

Configuration management for
network equipments DB

Add New Web Server

NSAC

Server Add

Add new
Web Server
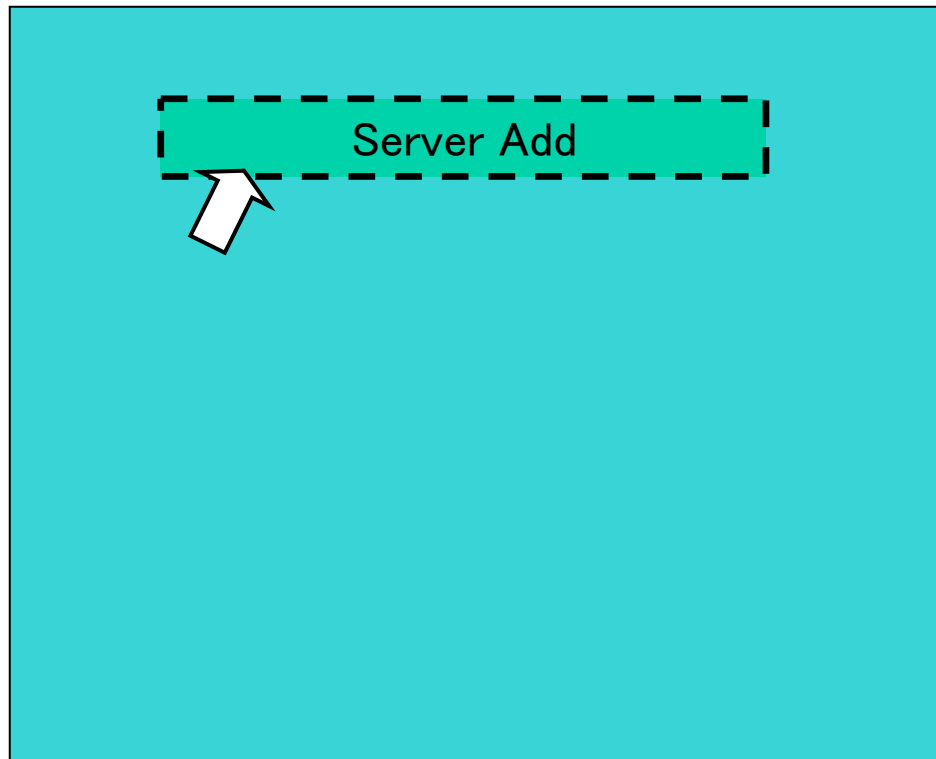
SI

Automatic Resource assign

・IP address (Service) request
・IP address (monitoring) request
・NIC I/F（Service）request
・NIC I/F（monitoring）request

SI/operators

System topology DB

For the Service
・V−LAN ID
・IP address
　　10.0.0.2

For the monitoring
・V−LAN ID
・IPアドレス
　　11.0.0.2

NIC I/F
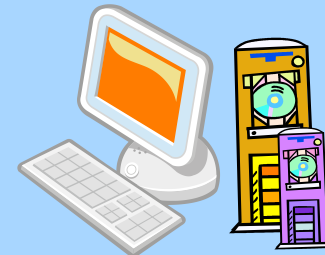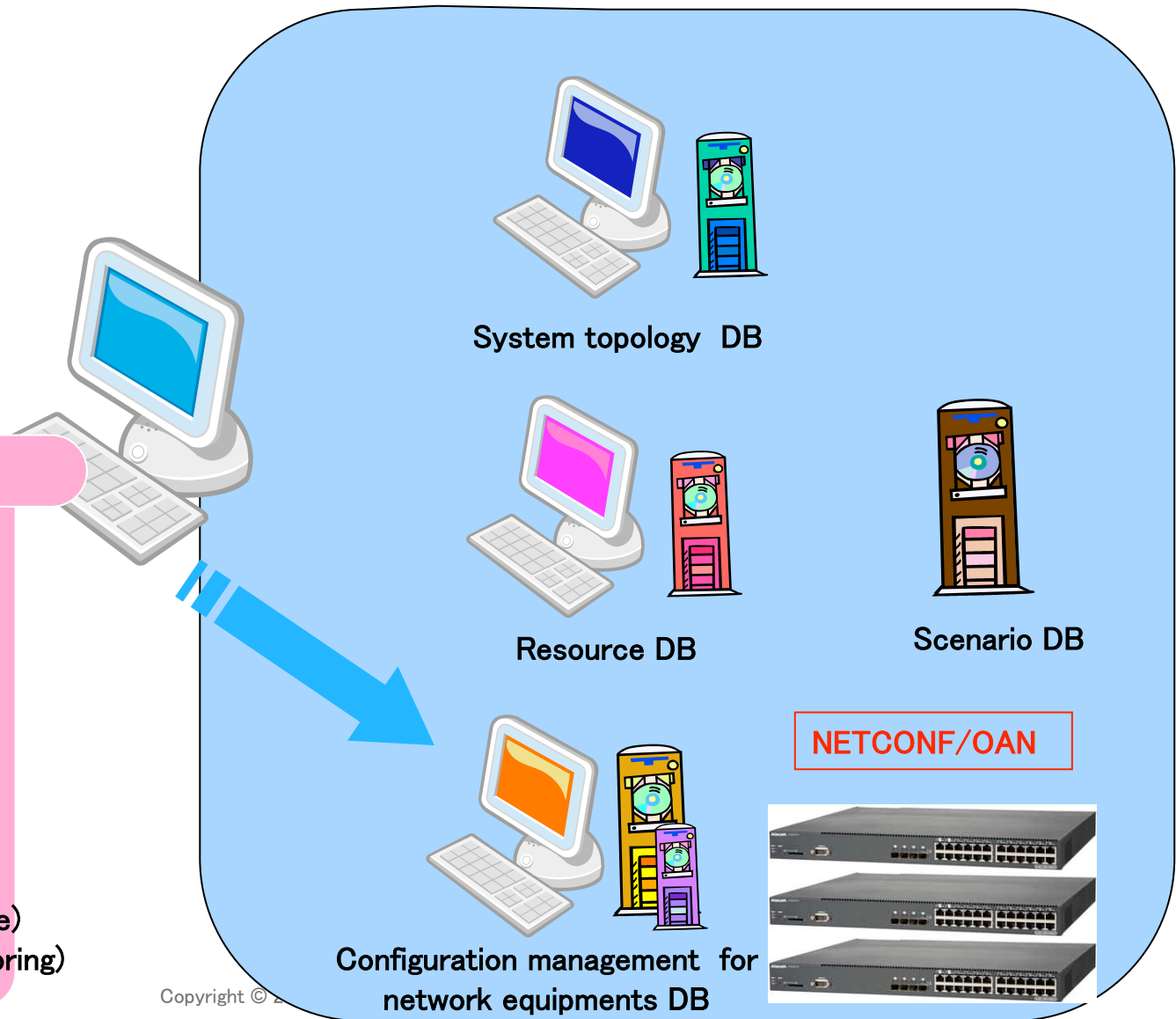・Snsl2sw01:02 (for the service)
・Snsl2sw01:12 (for the monitoring)
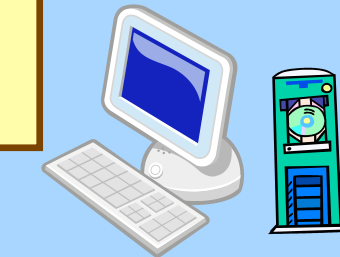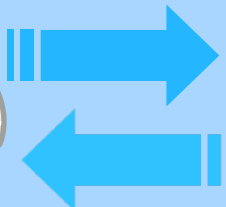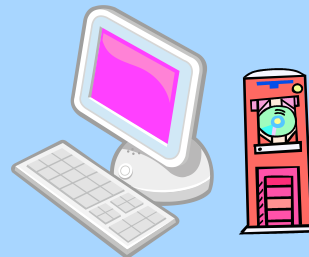
Resource DB

Scenario DB

Configuration management for
network equipments DB

Automatic configuration using NETCONF/OAN

System topology DB

Resource DB

Scenario DB

NETCONF/OAN

Configuration management for network equipments DB

For the Service
・V-LAN ID
・IP address
　　10.0.0.2

For the monitoring
・V-LAN ID
・IPアドレス
　　11.0.0.2

NIC I/F
・Snsl2sw01:02 (for the service)
・Snsl2sw01:12 (for the monitoring)

Prototype System

select
layer of system

select
operating scenario

view selected system
each layer

view properties of equipment

Lessons learned

- **It was important to decide use cases to implement NSAC.**
    - ◆ To define datamodel
    - ◆ To design the databases
- **Defining datamodel/meta-model is important for implementation of automatic management system.**
    - ◆ AlaxalA SW provided VLAN datamodel. NSAC used the datamodel. However, we defined our datamodel/meta-model for the system. Because for the system, we didn't use all VLAN function, we needed to deal with relation other computer.

Lessons leaned (cont.)

■ We had to consider system overview at the first, then designed the components for 4 levels

  ◆ Protocol level (NETCONF)

  ◆ Datamodel level

  ◆ Implementation level

  ◆ System level

■ NSAC was for tight coupled system. The Internet is loose coupled system. Requirement for each system is very different.

| Tight | ⟷ | Loose |

VLAN in datacenter   Wide area router network
                                  by BGP