The Automatic On-boarding System Used in China Mobile's NFV/SDN Network

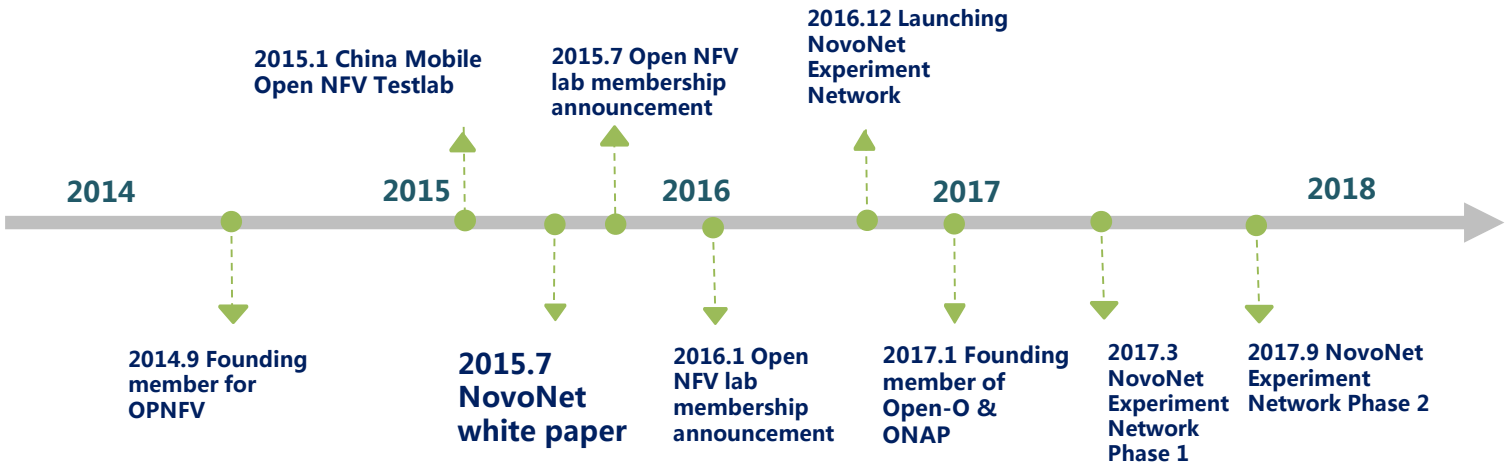
Fan Yamei, China Mobile, fanyamei@chinamobile.com
Fu Qiao, China Mobile, fuqiao@chinamobile.com

Index

- Best Practice: Telecom Integrated Cloud(TIC) in China Mobile
- Automatic On-Boarding system:
  reducing TIC integration from weeks to minutes
- Demo

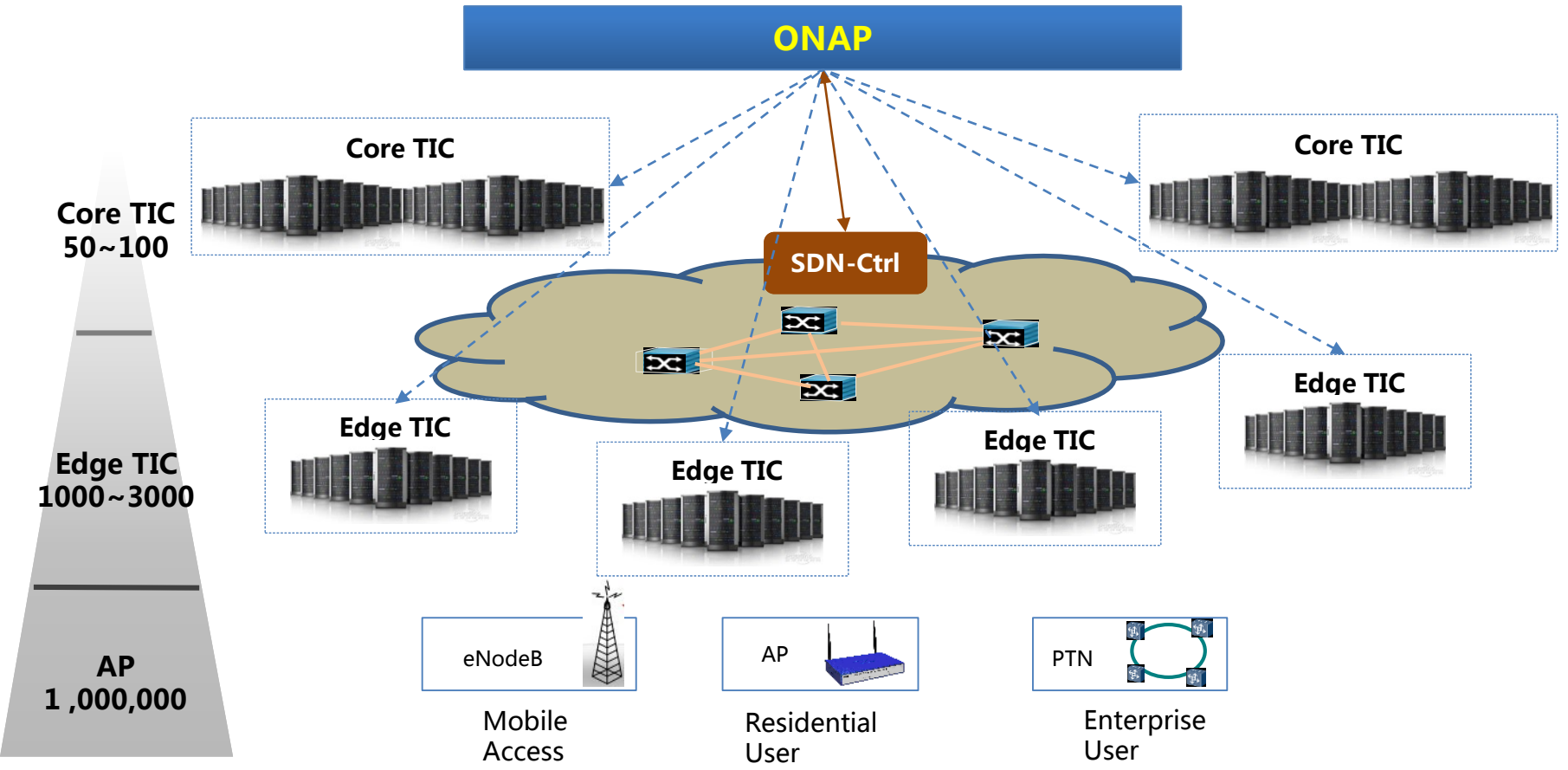
China Mobile’s Trip to NFV/SDN

- China Mobile announced its future network as “NovoNet”
- Based on NFV/SDN, NovoNet is a new network with shared virtual resources and agile network structure. NovoNet provides network users with scalable resources and open API to utilize
- China Mobile began its vIMS trail in 4 provinces since 2015, NB-IoT trail in 6 provinces in 2016
- China Mobile began its NovoNet Experiment network since 2016. This is a grand new network built for experiments and testing of NFV/SDN. The network currently includes 4 provinces. Multiple services, including NB-IoT, vBRAS, sCPE, E-BoD are tested in this network



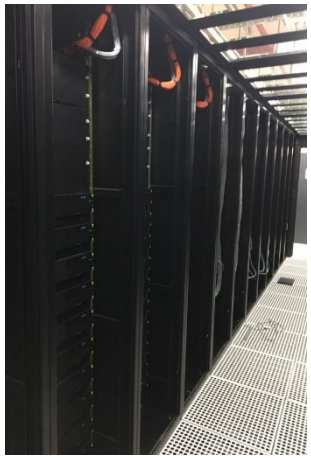
China Mobile Future Network Architecture

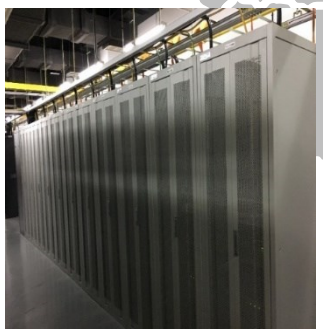
- China Mobile structures its NFV Cloud into TICs (Telecom Integrated Cloud)
- TIC is a standard unit, with unified hardware and software infrastructure. Multiple VNF services are deployed in TICs
- TICs are categorized into Core TIC and Edge TIC , in which core TIC mainly supports control plane services, and Edge TIC is for forwarding plane services



China Mobile NovoNet Experiment Network

- Since 2016, China Mobile is building its Novonet Experiment Network in 4 Provinces.
- 7 DC sites are included in the network, 15 TICs are constructed
- VNFs include vEPC for NB-IoT, vBRAS, vCPE, and E-BoD
- 9 virtual infrastructure vendors, 5 VNF vendors, 3 orchestrator vendors, 4 SDN vendors, are included in the network
- In Phase 1, most of the integration and testing work are done manually, which cost us more than 9 months to construct the 15 TICs. In phase 2&3, we are going to utilize the automatic system so as to significantly reduce complexity and man’ s labor

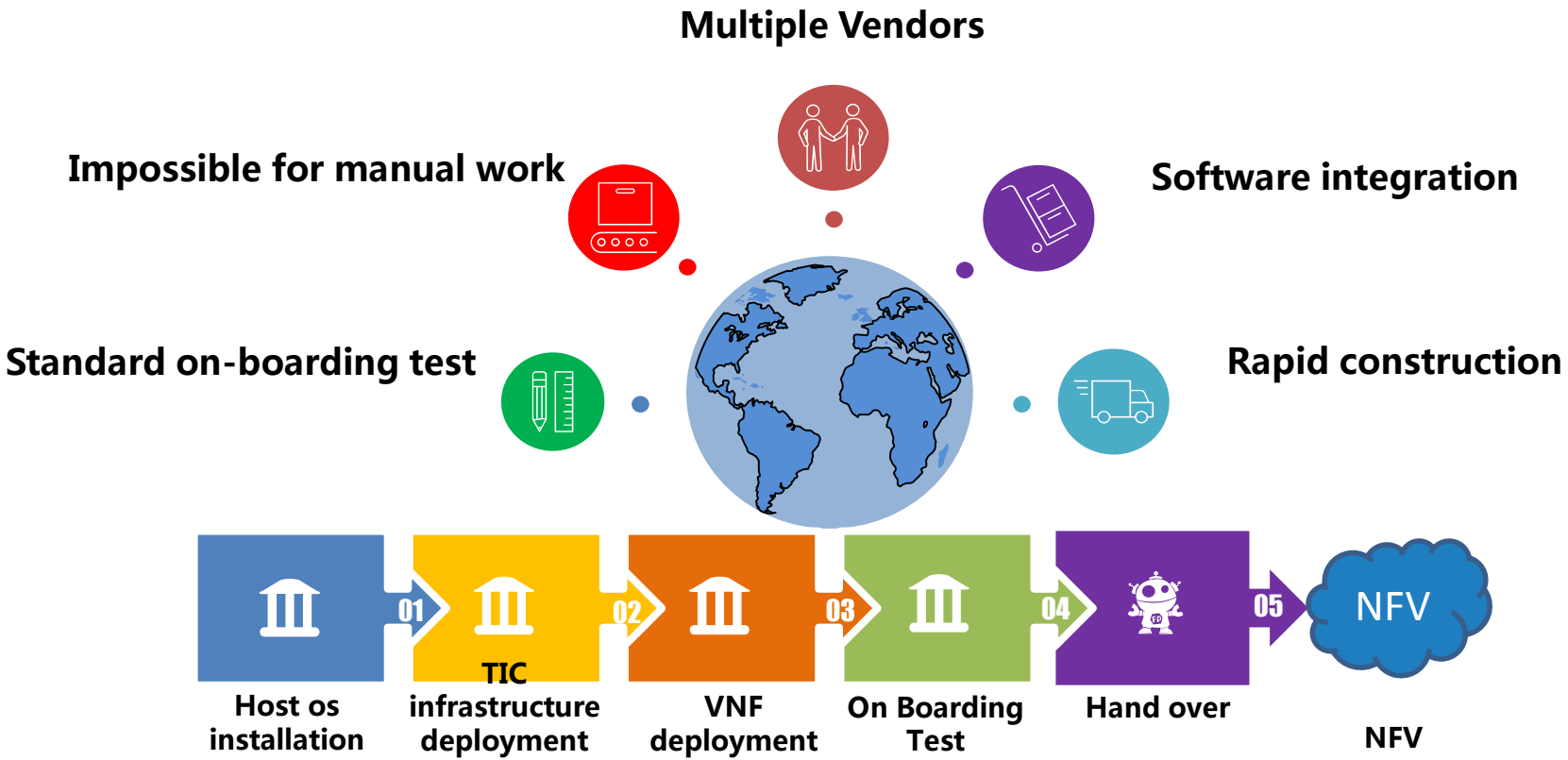




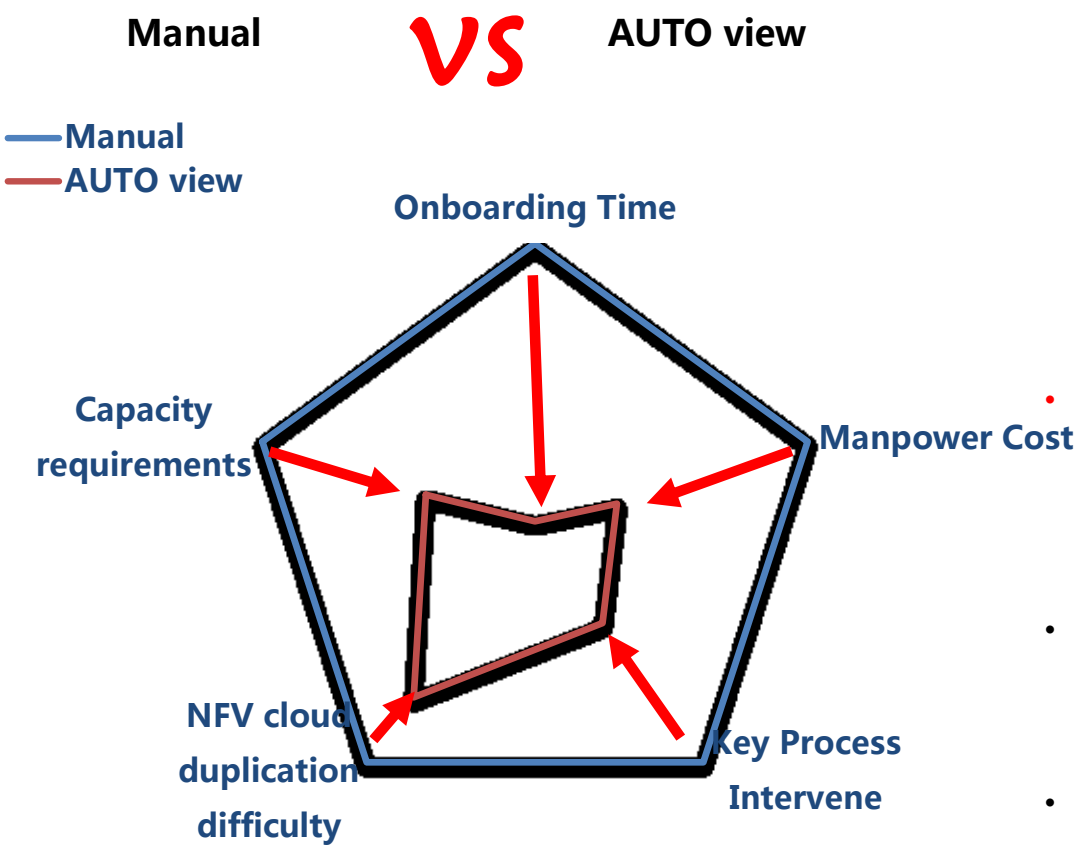
Automatic On-boarding system: Integration Difficulties

Automatic On-boarding system is crucial for future network integration and large scale deployment

- The integration of TIC includes components from multiple vendors with complicated process. Manual integration leads to huge workload, and is almost impossible to repeat.
- Different from traditional network made of bunch of hardware, future software infrastructure on-boarding of TIC requires functional, performance and availability testing before launching
- An automatic integration and on-boarding system provide standard process and tools for integration and on-boarding testing for multiple vendors. Making the whole TIC easy to copy and re-integrated



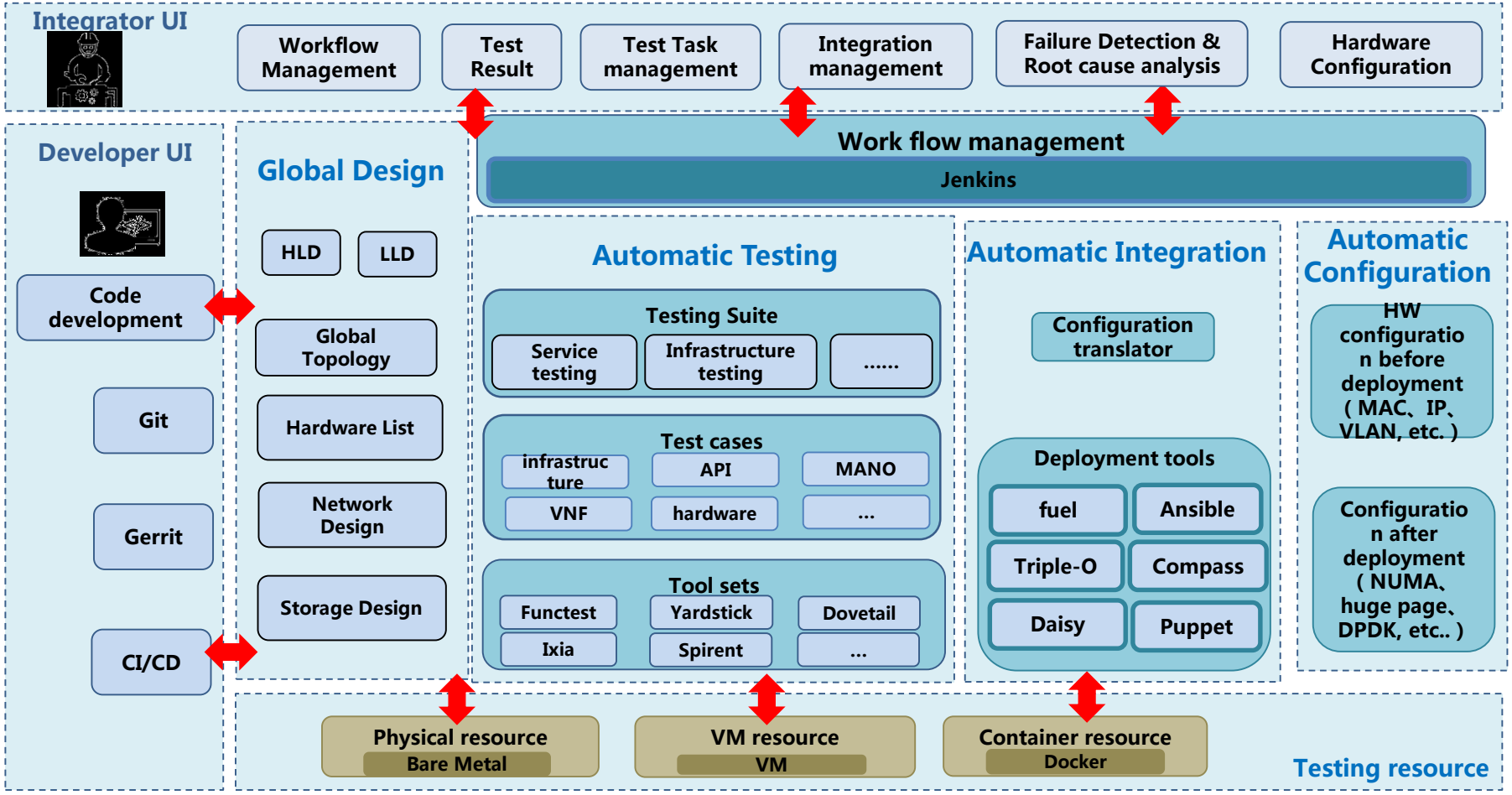
Automatic On-boarding system: Weeks to Minutes



- Short the NFV Integration work into hours form days:
  - Experience sharing: When we manually do the test on NFV infrastructure, the time costs more than 7 days * 2 person. If the automatic testing is on, we only use 2 hours.
- **Reduce the manpower cost:** The automatic configuration deployment and testing can help reduce the on-site persons to deeply reduce the cost of money.
- **Reduce the capacity requirements** for the field engineer. With easier deployment and convenient problem location.
- To achieve a certain degree of unmanned intervention
- TIC can be **easily copied** and the experience can be recorded efficiency.

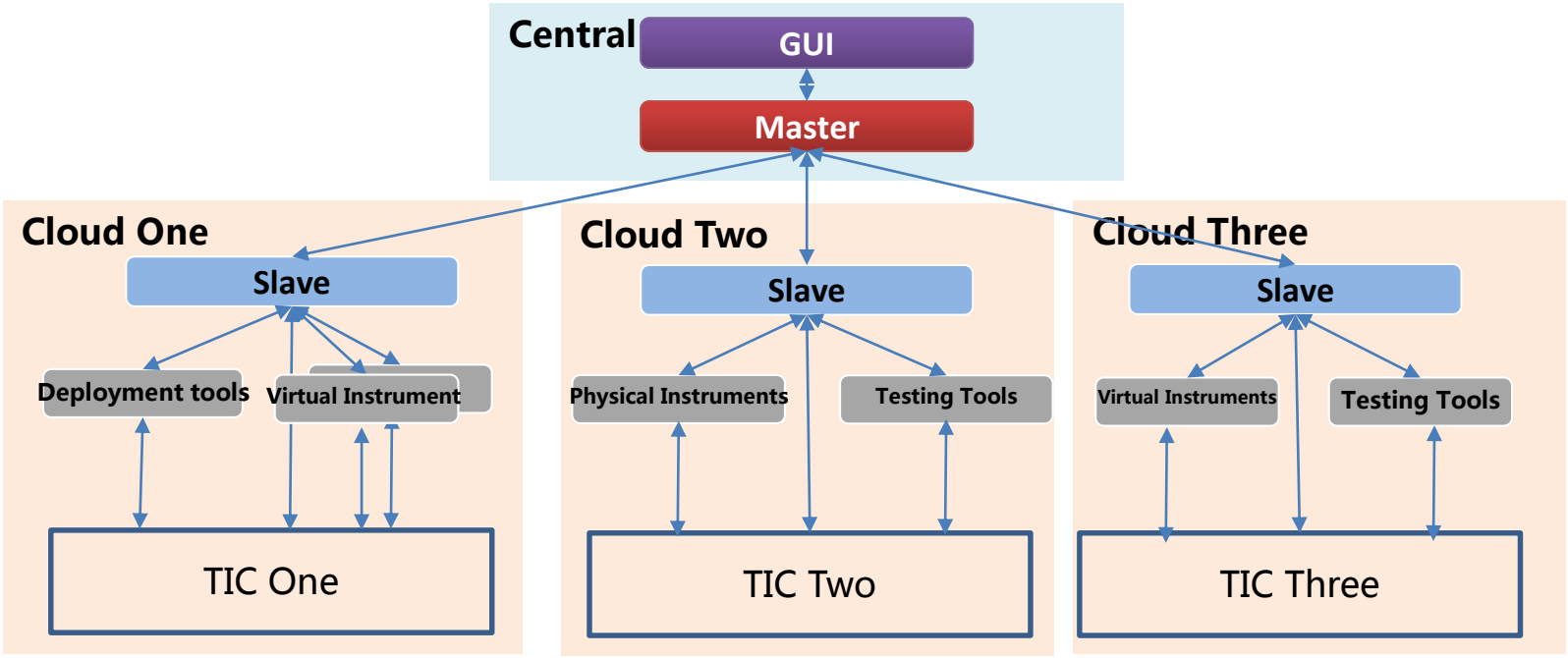
System overview

- Provide agile selection of automatic on-boarding tools, including integration, testing and configuration
- Provide pipeline mechanism for work flow management
- Easy to use for integrators from End Users



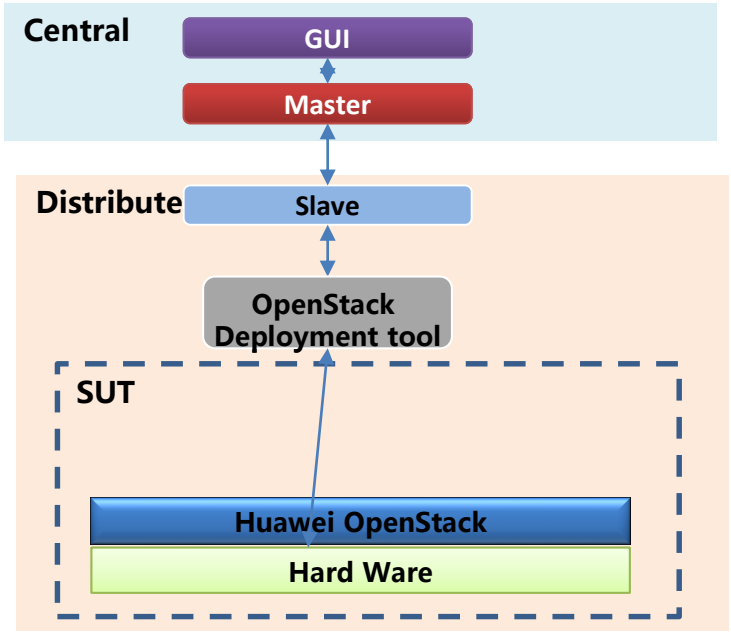
Deployment Architecture of AUTO

- Two layers architecture is designed for the easier tasks management and distribution.
- It includes one Master and several slaves.
- All of the integration and testing tasks will be managed by the Master to distribute the tasks to slaves



1. Auto integration – Architecture & Demo

Preparation; Set VT-d/SR-IoV/IPMI/Switches VLAN and network design.



1. **Environment cleaning**
2. **Issue the parameters to Deployment tool and trigger automation OpenStack deployment task**
3. **Wirh the automation scripts, trigger the FushionSphere deployment.**
4. **Deployment complete and run basic function tests.**
   - **Create Basic Resource(e.g. Image, Network, Flavor, etc.)**
   - **Create a VM**
   - **Wait and ssh login to the vm and exec basic cmd**
5. **Login and verify the OpenStack.**

Components

- Jenkins & Registry tools
- Installer(FushionSphere, etc.)

Features

- Support multiple Installers
- Transfer adjustable parameters to deployment env.
- Unified network and env configuration
- Jenkins + Ansible + python driven

Problem Met Yet

Each vendor has their own deployment tools. At present, we have integrated commercial OpenStack automation installation, vendors like HUAWEI, ZTE, WINDRIVER, and so on. We have found some problems and difficulties in the trail.

1. Installation tools' difference make the unified configuration more difficult. Under this challenge, we want to define a **configuration document** to help the VIM installation, such as role distribution and CPU model setting and so on. **PDF and IDF defined by pharos may help with this.**

2. **After virtualization layer deployment,** how to implement the automation of virtualization layer later configuration, such as configuring large page memory, tenant, project, DPDK, NUMA, affinity, SR-IoV, subnet and so on. It is also a key point in ATUO.

HLD(High Level Design) & LLD(Low Level Design)

中国移动 China Mobile

Integration project Design

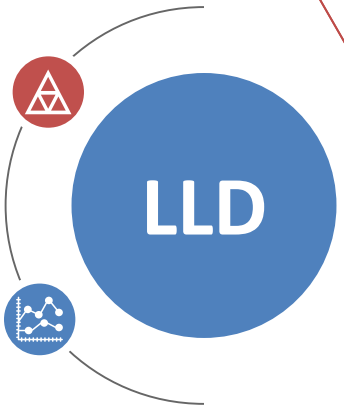
Resource Pool Planning
Network Planning
Disaster tolerance in different places
Security Planning

Construction Planning

Data Center reconstruction
Servers and Switches' connection
Power Testing

Version Management

Software version management
Patch Management
License Management



"It is very important for future deployment of NFV

OPNFV On-going discussion about PDF/IDF in Pharos:
To help the lab owners can deploy any installer using the same configuration file. It is defined in PDF(POD Descriptor File) and IDF(Installer Descriptor File)

Hardware Setting

Before deployment

Server Setting: BIOS/PXE/Hard Disk
Switch Setting: VLAN/router/security strategy/port
Storage Setting: RAID/Storage resource pool

Unified Configuration File

Deploying

Role distribution/
CPU model/……

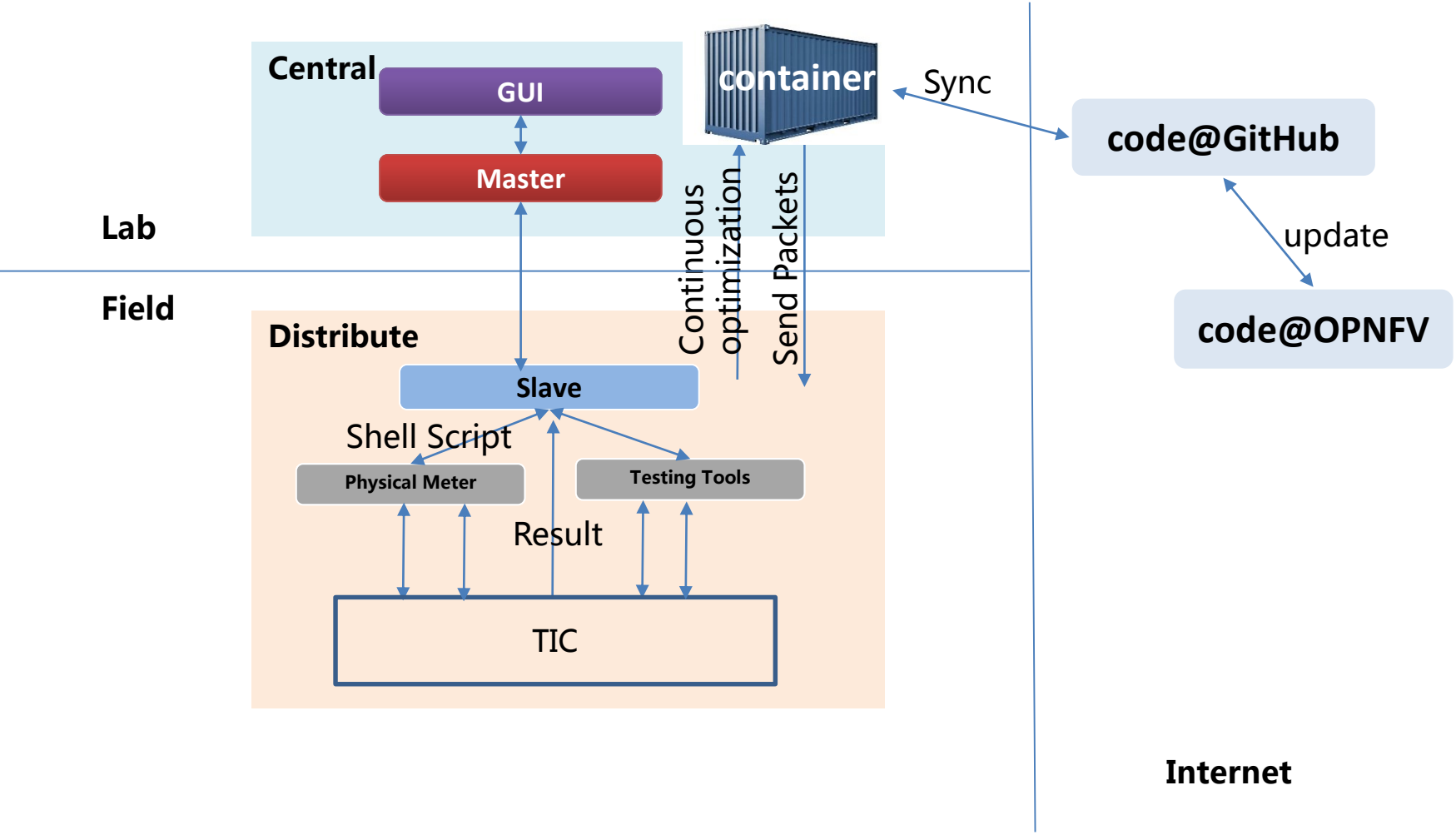
Server and VIM Setting

After deployment

NTP/DHCP server setting
VIM setting: large page memory, tenant, project, DPDK, NUMA, affinity, SR-IoV, subnet

2. Lab Testing Code Maintain & Easier Migration

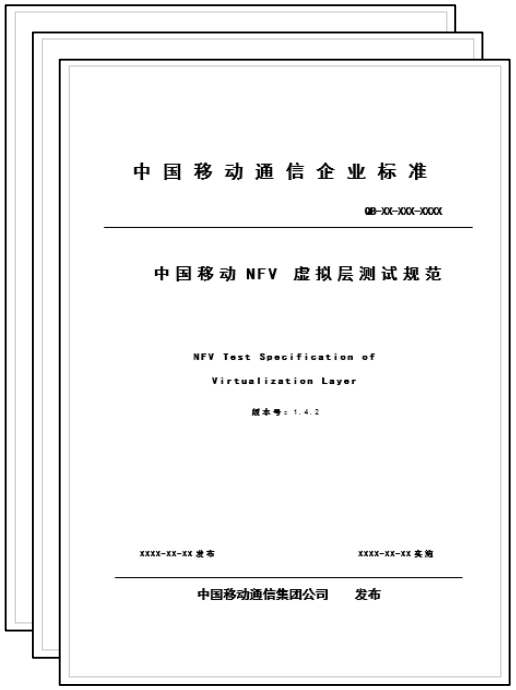
- Continuously downloading updated codes from OPNFV
- More than 4 TICs are included into the automatic system for integration and on-boarding testing
- Finish development of more than 300 test cases on infrastructure and services



Auto NFVi Testing Reference

- **China Mobile NFVi Specification: Translate into machine language one by one**
  - Developed by China Mobile
  - **200+** test cases
  - functional, performance, and availability
- **Framework: Yardstick in OPNFV & Ansible**
  - Easier code migration: packet all the NFVi test cases into docker
  - Quite rich and useful Rest API, Contexts and scenarios.
  - Support Plug-in, can integrate other tools

China Mobile
NFVI testing specification



Translate into

Utilizing Yardstick framwork

Machine Language

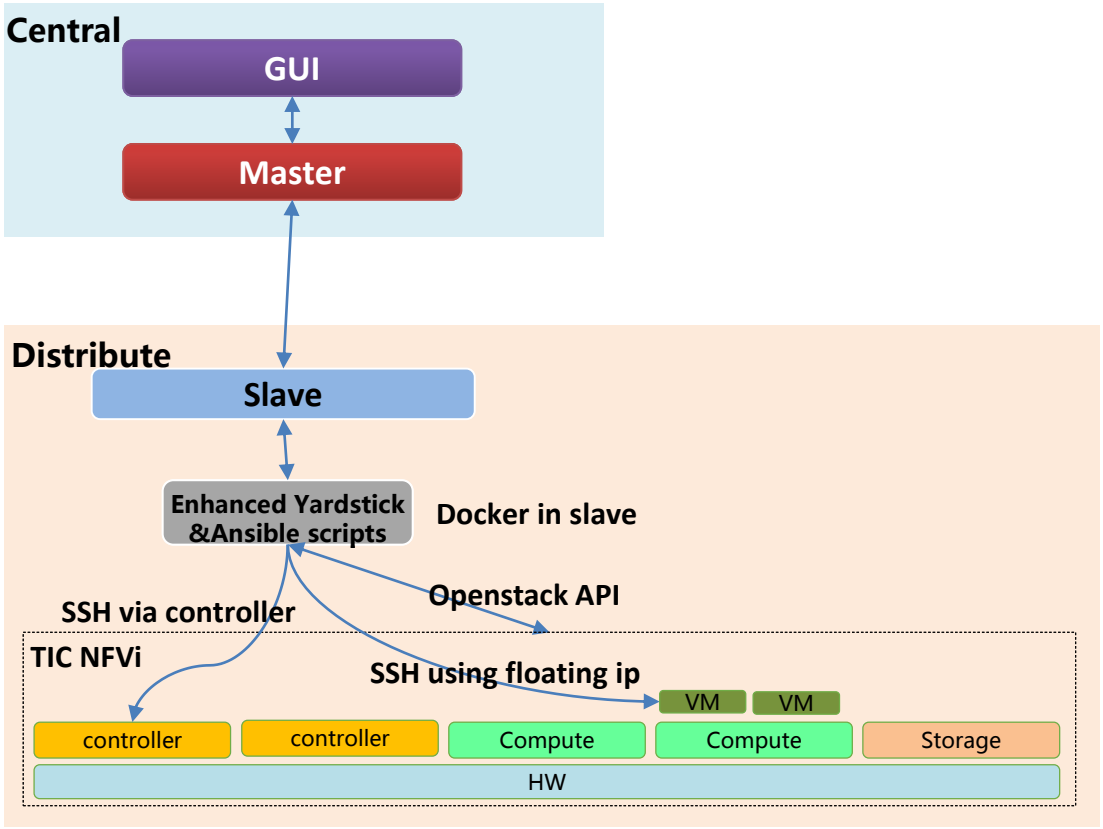
NFVI test cases

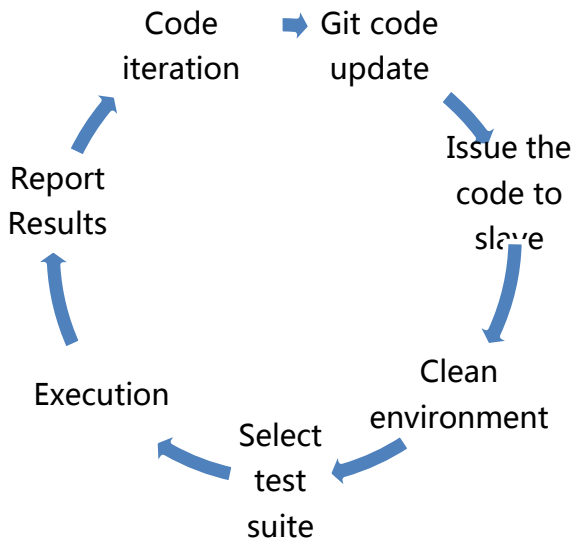
The auto test cases are based on the NFVI testing specification in China Mobile. More test cases will be included like availability and performance.

- **Functional testing:**
  - Hypervisor
    - Virtual compute, including NUMA, hyper-threading, over allocation switch off, etc.
    - Virtua storage, including over allocation switch off, share volume, etc.
    - Virtua network, VLAN-transparent, DPDK & SR-IoV collocating
  - VIM
    - Virtual resource management
    - Lifecycle management
    - Resource orchestration
    - On-line maintenance
    - Restful API
- **Performance testing:**
  - Network forwarding performance (with/without DPDK/SR-IOV)
  - Concurrent operation testing
- **Availability testing:**
  - OpenStack Controller HA
  - VM HA: VM can self-recover when failure happens in Host os or servers.
  - Network HA, including management network, storage network, service network
  - Storage HA
  - SDN controller and device HA

Auto NFVi testing – Detailed Architecture & Demo

- Verify the NFVi





Execution Steps:

1. Git code update
2. Issue the codes to slave
3. Clean environment (some VMs flavors that left by failure test cases
4. Select test suite
5. Execute shell scripts
6. Report the results to GUI
7. Code iteration

Problems Met Yet

1. **Many vendors have strengthen their own products**. For example, some commercial OpenStack enhance the network security with encryption of NIC. When testing these products, we must do more adaption work to adapt the test cases to commercial environment.

2. We can not use ssh to **connect to the compute node directly**, we use the controller node to transfer to the compute node to finish the operation of hypervisor.

3. In some of the **high availability test cases**, we need to simulate destructive effects that work on environment. Some can not be recovered. We are trying to make the destructive effects have lower influence on environment. For example, the NIC damaging case, we simulate it with a NIC down command to the sever.

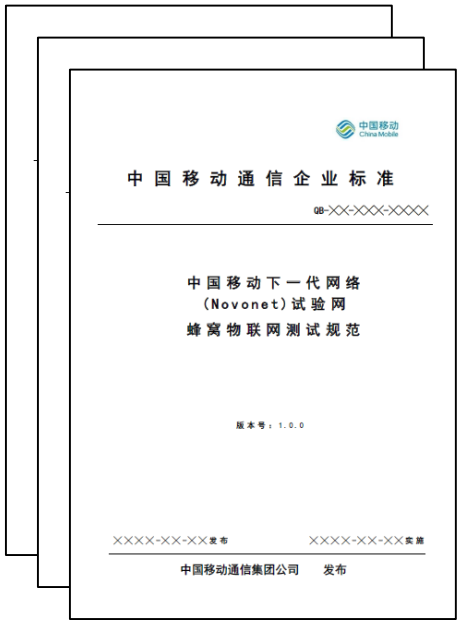
4. We all know that at the beginning of Yardstick Project, the aim is to test **the virtualization layer with deployment of VNFs.** We believe that it is very important to bear the VNF to do the virtualization function and performance testing such as NIC forwarding performance, CPU and memory loss.

3. Auto VNF Testing Reference

The VNF test cases are based on the NB-IoT testing specification in China Mobile.

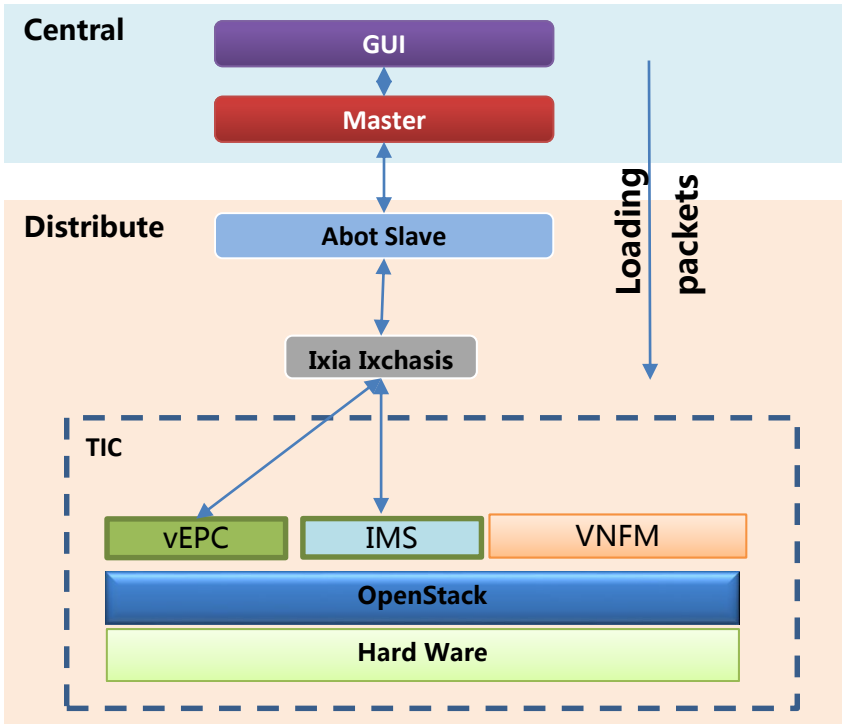
We are now cooperating with Ixia and Rebeca to cover all the EPC function test cases that are more than 30.

China Mobile
EPC(NB-IoT) testing specification



- **Functional testing:**
  - Mobility Management
    - IPv4/IPv6 attach or detach requirement, syc location update, etc.
    - Period TAU/idle TAU/Data Download or upload
    - Call Data Service(for upload & download)
    - Called Data Service(for upload & download)
- **Cloud Management testing:**
    - VNF lifecycle management, EMS managing the VNF instance, VNF upload, etc.
- **The influence of virtualization to the VNF**
    - The influence of NUMA, core binding, SR-IOV and so on
    - The influence of VM’ s software update, scale-up/scale-down
    - The influence of VM failure, etc.
- **The influence of Cloud Management to the VNF**
    - The update of cloud management platform
    - Hypervisor failure
    - Hypervisor service network failure
    - Hypervisor storage network failure
    - And so on

Auto VNF Testing——Detailed Architecture

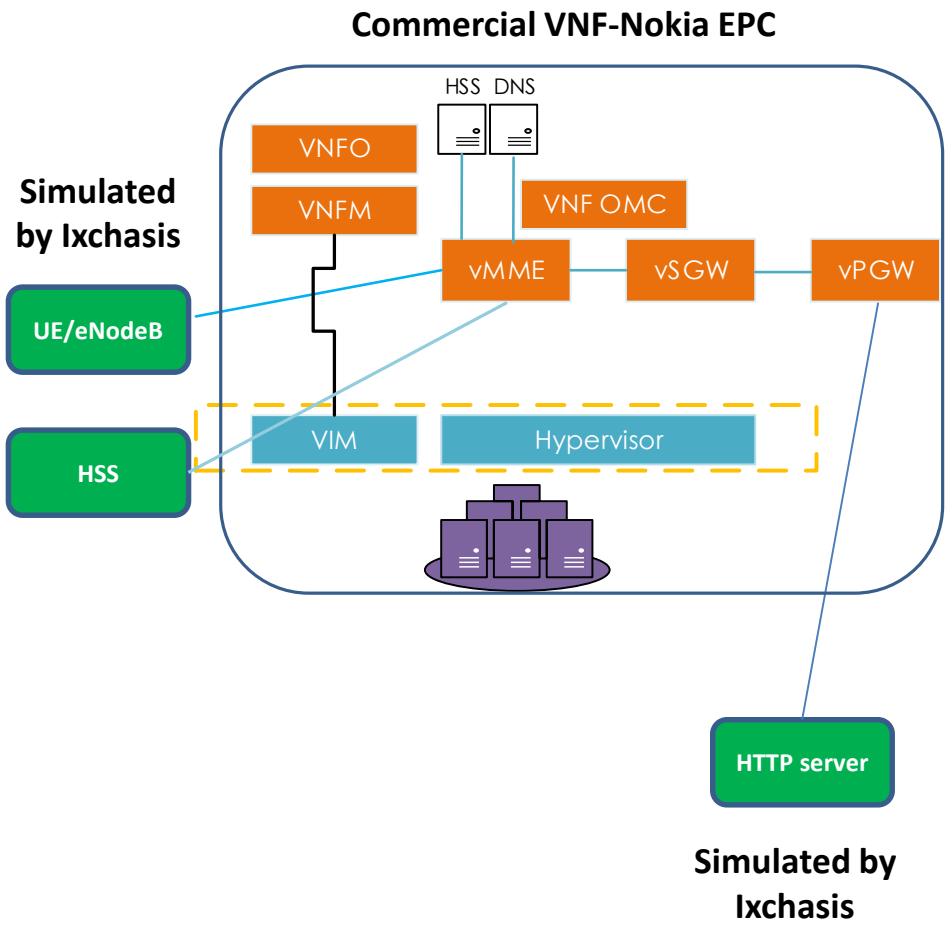


Execution Steps:

1. Management center will send the packets to the distribute slave. The packets are consisted of installation, setting, test scripts and so on. Some configurable parameters will also be involved.
2. Slave will make the Ixchasis set and then Ixchasis will send traffic to VNFs.
3. The results will be sent to GUI

VNF	EPC(NB-IoT)	IMS
[redacted]		
The criteria for the test result	Users can attach successfully Users can access to the Internet ……	Can make a voice call ……

vEPC Testing Strategy & Demo Cases



Test cases showed:

- NB users attach and detach with IPv4/IPv6
- Idle TAU
- Period TAU
- Upload, download data

One simple EPC case:

1. Simulate 1000 UEs attach to IPv4 APN using IMSI (or withGUIT)
2. Set the attach speed to 20 per second
3. After 60s, UEs begin to detach with speed 20 per second

The adjustable parameters :

1. eNodeB/HSS IP address/Gateway/Netmask
2. MME IP address
3. TAC
4. National code
5. Area code

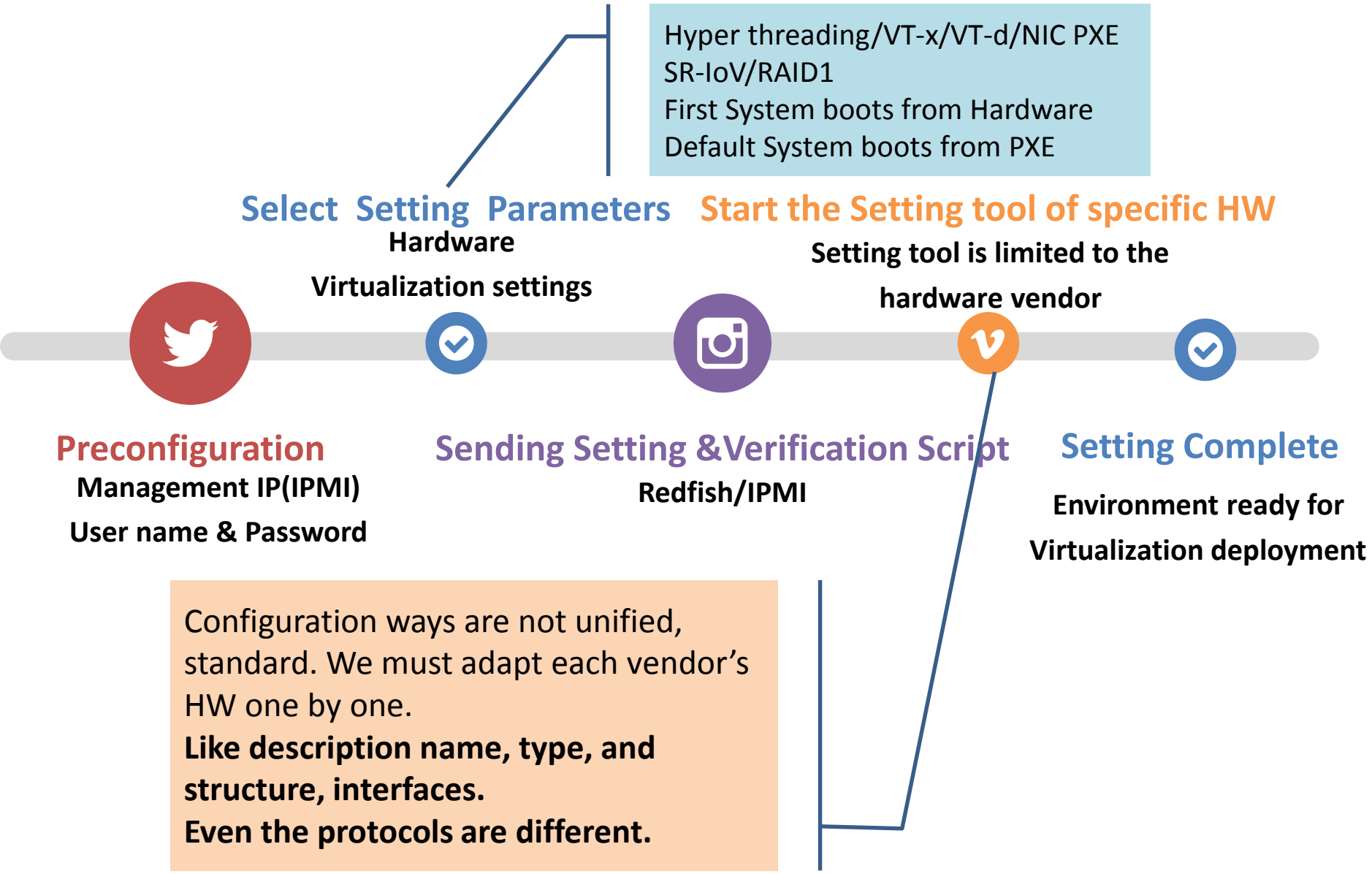
Problems Met Yet

Some of the procedures that we can not step by.

1. Make the whole procedure **in a more friendly way** to be easier for the field engineer to deploy the test tools and testing cases into their OpenStack(or other VIM) environment? For example, **Once click, all done.**

2. **Some of the test cases that are not have been covered**
   - I. Some test cases that are related to **NFVO & VNFM**
   - II. Some test cases that are related to NFVi’ s high availability. That depends much on **NFVi’ s** failure cause. (For example, Storage network or compute node failure)
   - **III. SR-IoV**’ s effects on VNF’ s performance
   - IV. ……

4. BIOS Setting Exploration



Problem Met

We have done several servers' automation BIOS setting verification (Huawei/H3C/ ……).

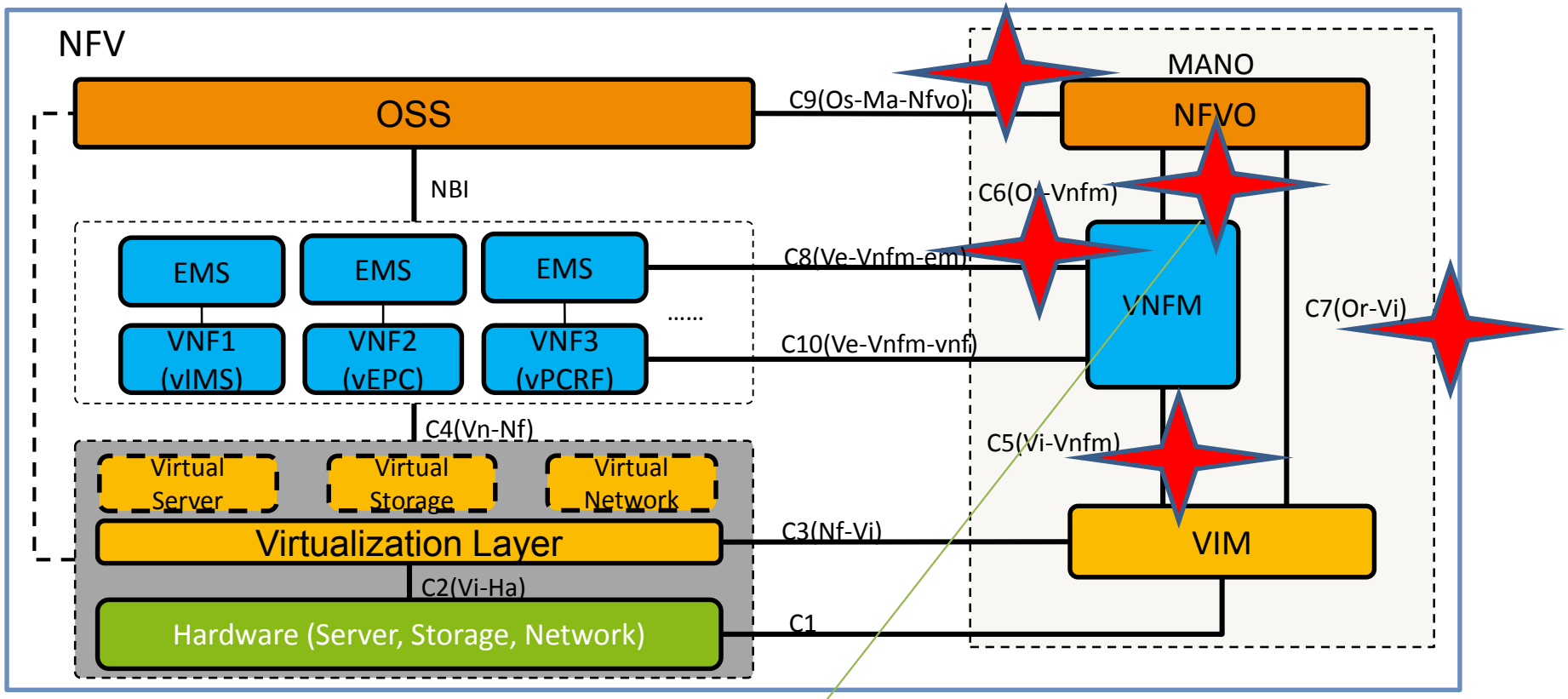
We find that:

1. Unifying all configuration documents is very important. Like using one protocol.

2. Switch automation setting is on research. We may choose TCL script to realize the automation setting of switches to set the VLAN, router and so on.

3. Dedicated storage device is not decided yet.

Open Questions ? ——MANO Testing



MANO testing is immature.
We are involved and exploring a better way

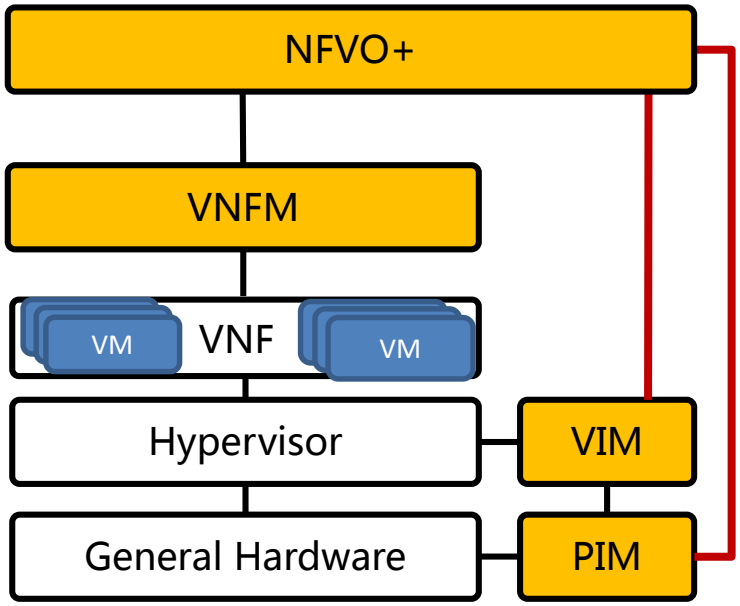
OpenStack Tempest:

C5(Vi-Vnfm)/C7(Or-Vi)

NFV TST Working Group: TST010

Black box/Testing the API operations/SOL API conformance tests

Open Questions?——Root cause Analysis



There are many types of errors.

MANO Error NFVO/VNFM/VIM/PIM

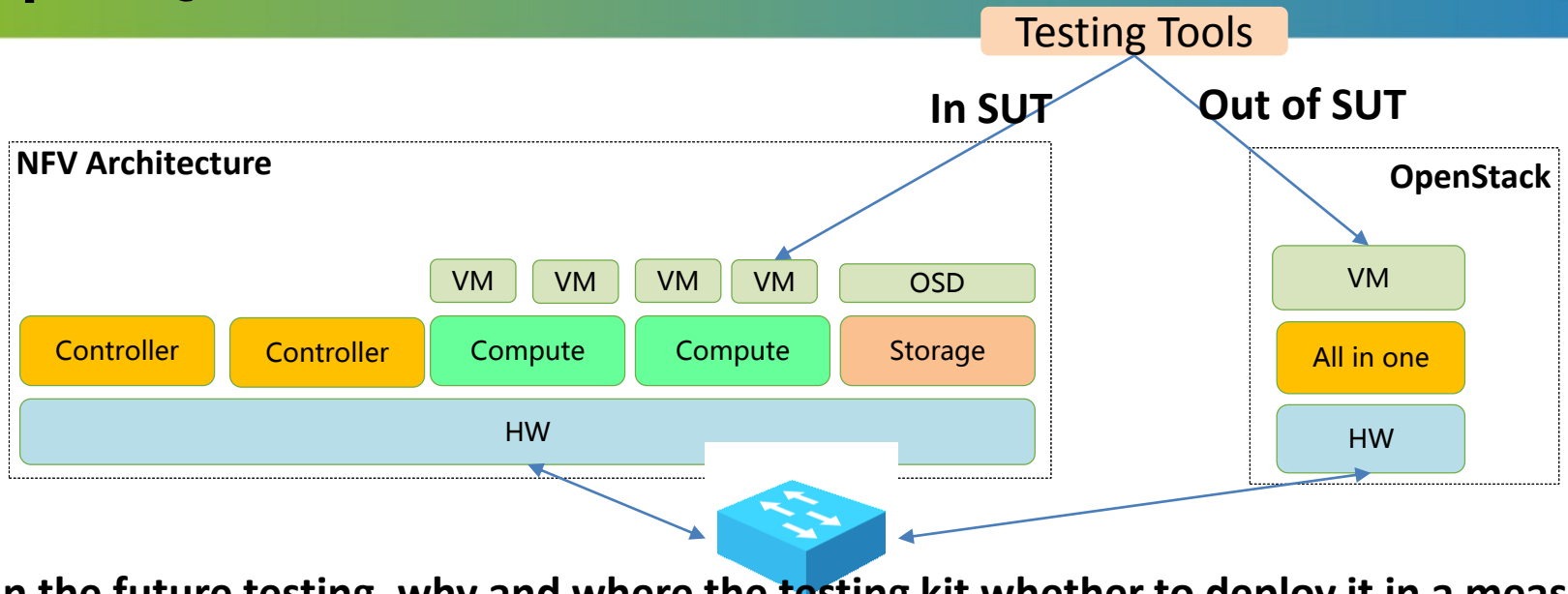
VNF Error VNFC/LB/etc

Hypervisor Error Virtual Machine/Virtual Software/etc

Hardware Error Compute /Storage/Network/etc

- Root cause analysis is very important to solve the fault and error in short time.
- Quick response and the analysis to the warning and errors are basic guarantee to reach carrier degree high reliability. There are several ways to monitor the error.
  1. Heartbeat detection mechanism
  2. Watch dog
  3. active/ standby changeover
  4. ……

Open Questions?——In or out of SUT



In the future testing, why and where the testing kit whether to deploy it in a measured environment(SUT) or out of the environment(out of SUT).

	In SUT	Out of SUT
Advantage	the environment is already prepared. We have been already familiar with the environment and the problems will be solved more quickly.	the test tools will not influence the performance of SUT
Disadvantage	When test the NFVi performance, it will influence the basic parameters of SUT.	Need to deploy anther environment and prepare a new machine. The test result will influence by switch performance.

Next Steps

- **General:**
  - Verify and utilize the system in lab to help TIC on-boarding.
  - Improve the GUI function make it more friendly and useful.
  - Solve the problems and open questions.
- **Auto testing:**
  - Finish development of NFVI performance testing and availability testing cases
  - Integrate more service test cases, including vEPC, vIMS, vCPE, and etc.
  - Improve test result presentation
- **Auto integration:**
  - Integrate more installers
  - Provide generic configuration file and translator

THANK YOU!