



# ONAP PNF Plug and Play for Casablanca (R3, 4Q 2018)







- ONAP and PNF Plug and Play for 5G RAN
- 5G Use Case Team



# PNF Plug and Play CASABLANCA IMPROVEMENTS OVERVIEW

- ONAP and PNF Plug and Play for 5G RAN
- 5G Use Case Team

# Casablanca R3 Summary Overview

TOPIC	ICON	DESCRIPTION
<b>PNF Registration Handler (PRH) Improvements</b>		New VES Event domain for PNF registration with corresponding support in VES collector, DMaaP and PRH.
<b>SO Workflow enhancements</b>		Introduction of dedicated 5G use case work-flow
<b>Service Configuration Improvement</b>		Service configuration improvements from APP-C/SDN-R to PNF after PNF registration to PRH
<b>Security Enhancements</b>		Authentication, Certificates, User name & password and intra-ONAP security.
<b>Modeling enhancements</b>		Modeling enhancements to support 5G PNF in ONAP. Inheritance, and PNF characteristics for sharing. Focusing on PNF connectivity. PNF-SDK.
<b>PNF Onboarding / Package</b>		Defining <i>PNF Onboarding Package</i> . Extending framework to work with PNFs. Defining PNF Package framework.

# PRH Enhancements

## DESCRIPTION

*PNF Registration Handler (PRH) Enhancements.*

### **(1) New VES Event domain for PNF Registration –**

Create new VES event domain for PNF registration with corresponding support in VES collector, DMaaP and PRH. In Beijing the PnP Use case used the “*other*” domain to register VES events. For Casablanca, we propose using a dedicated domain “*pnfRegistration*”. New DMaaP topic corresponding to new domain.

**(2) VES EXTENSIONS** - As a result, VES collector and VES agent content will change with field updates using the new domain. Extensions for pnfRegistration fields. Corresponding VES Schema change.

**(3) DU SIMULATOR UPDATE** – The registration VES event used by the DU Simulator (or actual PNF) will need to update its JSON payload to match the changes above.

**(4) PRH & AAF INTEGRATION** – PRH needs to interface to AAF for Intra-ONAP security. PRH needs to use certificates and tokens for authentication to be allowed to exchange events. PRH needs this authorization to publish to DMaaP topics.

**(5) REVIEW LIBRARIES** – comply to Linux foundation requirement for licensing & security vulnerabilities. Risk estimation.

**(6) DEMO** – Demo of PRH w/ PNF PnP in 5G Use Case



## PROJECTS

PNF Registration Handler, DMaaP, DCAE

# PNF PnP: SO Workflow Enhancements

## DESCRIPTION

### (1) SO WORKFLOW ENHANCEMENTS –

Dedicated 5G BTS Workflow in SO. (If not model-driven would need special 5G BTS workflow)

### (2) PNF WORKFLOW – Extensions to Beijing SO Workflow (part of VCPE workflow).

Developed in Beijing not tested or integrated.

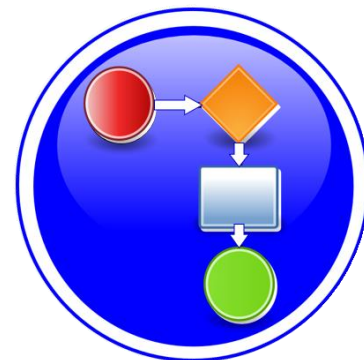
### (3) MODEL DRIVEN – SO not yet model driven.

Need to solve vis-à-vis a SO work flow specific to service & resource use case.

### (4) UPDATE PNF WORKFLOW – needs to be officially tested, accepted (in Casablanca).

### (5) SDN-R TO SO INTERACTION –SO calls SDN-R (Generic API call vs REST call)

### (6) DFX (Design for Excellence) – Resilience, Performance, Scalability, Stability, Multi-site.



## PROJECTS

SO

# SERVICE CONFIGURATION ENHANCEMENT

## DESCRIPTION

- (1) Service configuration Enhancements** to ONAP Controller to PNF *service configuration* exchange with PNF.
- (2) PROTOCOL DEFINITION** – Better definition around the Protocols supported (and/or support more protocols). What ONAP controller supports what PNF and what protocols are supported. NetConf, Ansible, Chef.
- (3) Configuration Extensions** – New parameters needed for Casablanca use cases. Vid script to pushing data, ID config, ID where data comes from. Generic configuration support.
- (4) PNF PnP Config** – Finishing PNF PnP by sending down config data.
- (5) (OPTIONAL) Vendor Data** – Extensions to Service Configuration with Vendor specific configuration data can be developed (vendor dependent).



## PROJECTS

ONAP Controller (SDN-R), VID, SO,  
SDC/CDT

# SECURITY ENHANCEMENTS

## DESCRIPTION

- (1) PNF AUTHENTICATION** – DCAE must authenticate the HTTP/TLS connection from the PNF. DCAE to integrate the HTTP and TLS authentication functionality by AAF.
- (2) VENDOR CERTIFICATES** – Handling Vendor Certificates for TLS/SSH for PNFs.
- (3) USER NAME & PASSWORDS** – Provisioning. DCAE & PNF management of *User Name & Passwords*.
- (4) SECURITY BETWEEN COMPONENTS** – DMaaP & PRH to authenticate w/ other ONAP components.



## PROJECTS

PNF Registration Handler, DCAE, AAF,  
ONAP Controller, DMaaP

# PNF PnP: MODELING ENHANCEMENTS

## DESCRIPTION

- (1) PNF MODELING** – Modeling enhancements to support 5G PNF in ONAP. Model Inheritance definitions for PNF. SDC modeling improvements from Beijing PnP use case.
- (2) PNF SHARING** – SDC model updates for PNF characteristics focusing on PNF inter-connectivity.
- (3) PNF-SDK** – SDK provided from Vendors. This will help modeling the Physical “Box” (PNF) and network functions.
- (4) CDT ENHANCEMENTS** - Improving CDT to handle complex config templates, multiple templates per PNF, identify different sources for template data, integrating CDT into SDC, expanding CDT usage to other controllers.



## PROJECTS

SDC, CDT, PNF-SDK



# PNF ONBOARDING / PNF PACKAGE

## DESCRIPTION

PNF Onboarding and PNF Package

**(1) PNF PACKAGE DEFINITION** – Defining *PNF Onboarding Package*. Extending framework to work with PNFs. Defining PNF Package framework.

- A. **PNF ARTIFACTS DEFINITION** – Vendor specific/provided artifacts to add to the (new PNF) package.
- B. **PNF ARTIFACTS DISTRIBUTION**



PROJECTS:  
SDC, APP-C



# PNF Plug and Play IN CASABLANCA (R3)

- ONAP and PNF Plug and Play for 5G RAN
- 5G Use Case Team

# PNF Plug and Play Stages

Design Time

A



## PNF Modeling

Resources Definition/Services Definition  
SDC: PNF (physical element) Modeling  
Distribution of types



B



## PNF Instance Declaration

PNF Infrastructure Service Declaration  
First part of PNF instantiation  
DCAE & AAI Entry with PNF ID (e.g. MAC address)



C



## PNF Boot-strapping

PNF Powers up and Boot-straps  
PNF performs a "Plug and Play" procedure  
Equipment vendor proprietary steps

D



## PNF Contacts ONAP

PNF connects to ONAP via a Registration Event  
PNF Registration Handler (PRH) processes the event  
Generic (not vendor proprietary)

E



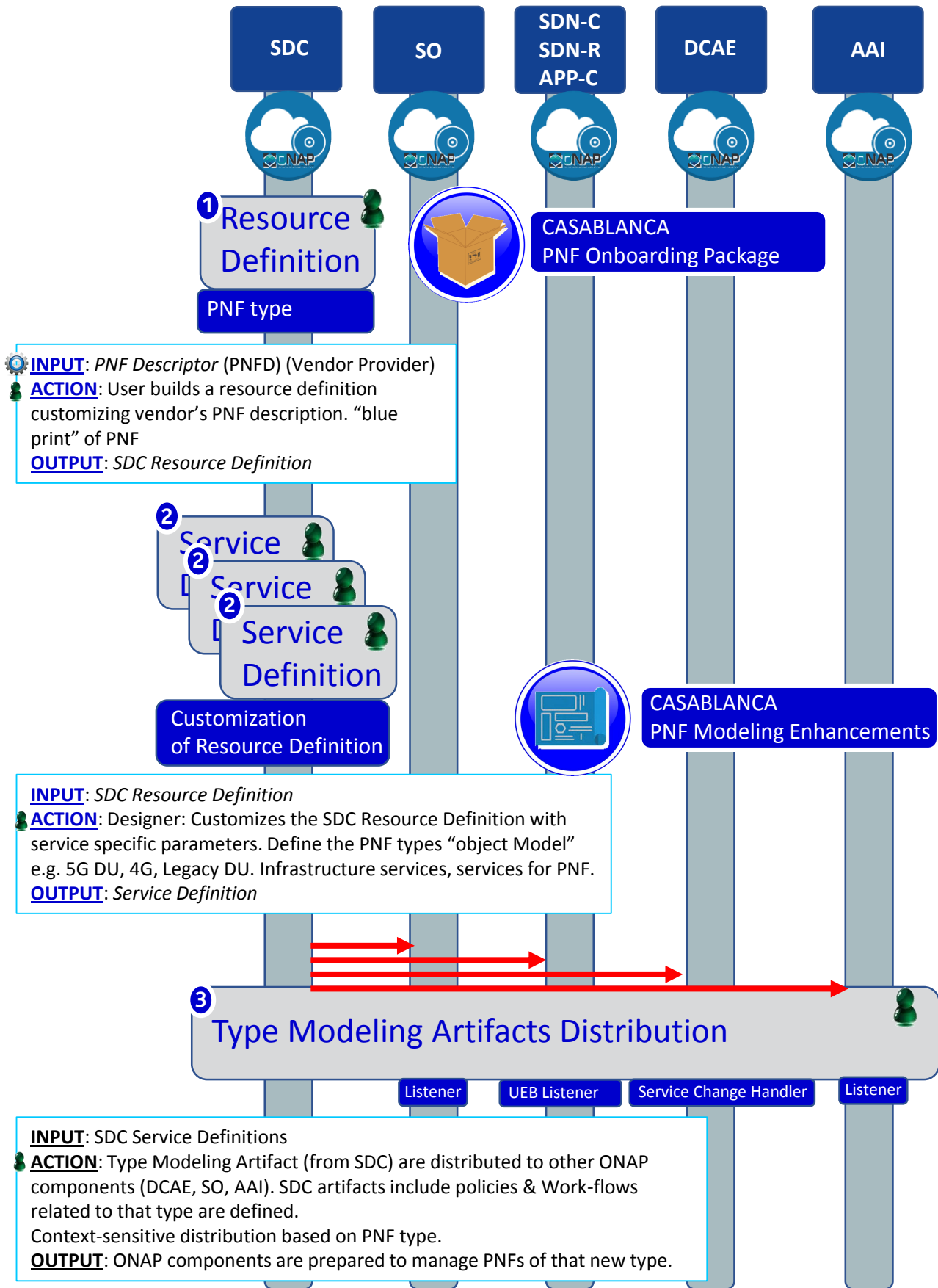
## PNF Activation

Connection points configured  
Second part of PNF service instantiation  
PNF configured and ready to provide service

Run-Time (Instances)

ACTORS	DESCRIPTION
PNF	<b>PHYSICAL NETWORK FUNCTION (PNF)</b> – A Network Function that is hosted on a physical hardware device, not in the cloud infrastructure.
DHCP	<b>DYNAMIC HOST CONFIGURATION PROTOCOL (DHCP)</b> – Protocol to assign IP addresses to a network element (NE). The IP address can be dynamically assigned or static based on MAC address of PNF.
SEGW	<b>SECURITY GATEWAY</b> – Used to set up IPSec tunnels to protects against unsecured traffic entering an internal network of a operator; used by enterprises to protect their users from accessing and being infected by malicious traffic.
CA/RA	<b>CERTIFICATE AUTHORITY / REGISTRATION AUTHORITY</b> – Used to generate a service provider certificate for the PNF.
Initial EM	<b>INITIAL EM</b> – Provides basic configuration and software download services to the PNF. This might be a equipment vendor specific solution. Also, responsible for identifying a PNF.
SDN-C	<b>SOFTWARE DEFINED NETWORK CONTROLLER (SDN-C)</b> – A controller for Layer 0 to 3 devices. Manages transport and network connections.
DCA&E	<b>DATA COLLECTION, ANALYTICS AND EVENTS (DCAE)</b> – Gathers performance, usage, and configuration data from the managed environment. Collect, store data and provides a basis for analytics within ONAP. For PNF Plug and Play can potentially perform analytics on the Plug and Play process, statistics, logs.
A&AI	<b>ACTIVE &amp; AVAILABLE INVENTORY</b> – The PNF is identified as available inventory and tracked through a key which is the PNF ID. When onboarded the PNF gets an entry in A&AI and can then be tracked, requested, and seen by the ONAP components for service requests or other queries.
SO	<b>SERVICE ORCHESTRATOR</b> – Serves as a mediator and coordinator of service requests.
APP-C	<b>APPLICATION CONTROLLER (APP-C)</b> - A controller for Layer 4 to 7 applications. Manages the life cycle of virtual applications, virtual network functions (VNFs), and components. APP-C manages the 5G DU & 4G DU.
PNF Registration Handler	<b>PNF Infrastructure Manager</b> – A microservice in DCAE used during PNF Plug-n-Play to process the PNF Registration event.

# Design Time (ONAP)



# PNF PnP: MODELING ENHANCEMENTS

## DESCRIPTION

- (1) PNF MODELING** – Modeling enhancements to support 5G PNF in ONAP. Model Inheritance definitions for PNF. SDC modeling improvements from Beijing PnP use case.
- (2) PNF SHARING** – SDC model updates for PNF characteristics focusing on PNF inter-connectivity. DCAE-DS Micro-service modeling.
- (3) PNF-SDK** – SDK provided from Vendors. This will help modeling the Physical “Box” (PNF) and network functions.
- (4) CDT ENHANCEMENTS** - Improving CDT to handle complex config templates, multiple templates per PNF, identify different sources for template data, integrating CDT into SDC, expanding CDT usage to other controllers.



## PROJECTS

SDC, CDT

# PNF ONBOARDING / PNF PACKAGE

## DESCRIPTION

PNF Onboarding and PNF Package

**(1) PNF PACKAGE DEFINITION** – Defining *PNF Onboarding Package*. Extending framework to work with PNFs. Defining Package framework.

A. **PNF ARTIFACTS DEFINITION** – Vendor specific/provided artifacts to add to the (new PNF) package.

B. **PNF ARTIFACTS DISTRIBUTION**

**TOSCA Meta data** - main service template, TOSCA template.

**Artifact Package**. Separated by types of artifacts. Separations by folder for different types. Place artifacts in categories. Anyone can choose which artifacts to receive.

**Definitions** – Specifies definitions such as CM, FM and PM definitions

**(1) Protocols Supported** – PNF package. CM Protocol is in PNF onboarding package. (Chef, Ansible, NetConf)

**(2) Controller** – What is the PNF controller



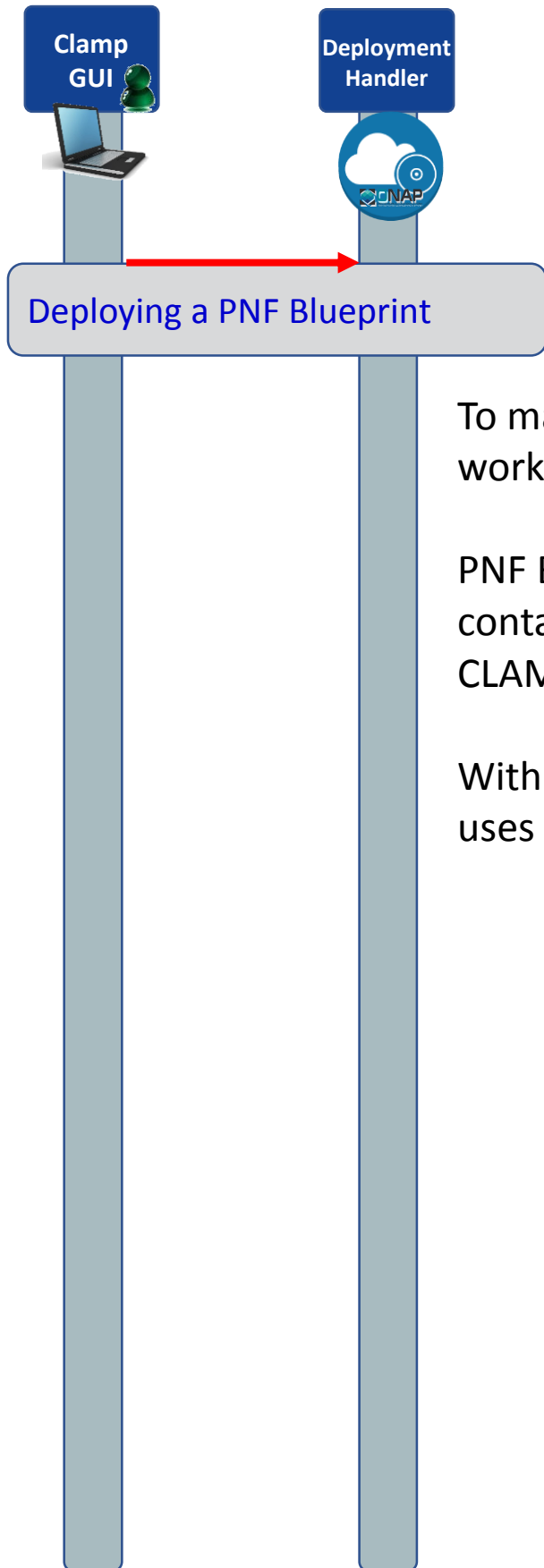
PROJECTS:

SDC, APP-C

STEP	DESCRIPTION
1	<p><b>RESOURCE DECLARATION</b> – A user on the VID performs a Resource Declaration. This uses the Service definition created in SDC. The user on the VID can define known information about the PNF. The user can (optional) provide the following information</p> <p><b>PNF RESOURCE Definition</b></p> <ul style="list-style-type: none"> <li><b>Resource Type</b> – Type of Resource. NEW type: PNF (pre-defined in SDC)</li> <li><b>NAME</b> – Name of the PNF type</li> <li><b>CATEGORY</b> – e.g. Infrastructure</li> <li><b>TAGS</b> – User-definable tags (default name of the PNF)</li> <li><b>DESCRIPTION</b> – Textual description</li> <li><b>CONTACT ID</b> – Designer (user of ONAP)</li> <li><b>VENDOR</b> – PNF Vendor (e.g. Nokia)</li> <li><b>VENDOR RELEASE</b> – Vendor release</li> <li><b>VENDOR MODEL NUMBER</b> – PNF Model value (link to A&amp;AI)</li> <li><b>EVENTS</b> – Monitoring Event definitions. Define design-time templates. CLAMP (runtime monitoring), DCAD (design time design template attach to VNF). Define templates &amp; attach them.</li> </ul> <p>Note: The user may provide whatever information in the above fields they know.  Note: Consumer vs Enterprise deployments. Consumer systems pre-registered, distributed throughout a region. For a consumer deployment you might not know the MAC address/Serial number (PND IF) until the PNF connects to ONAP.</p>
2	<p><b>SERVICE Definition (uses a PNF)</b></p> <ul style="list-style-type: none"> <li><b>NAME</b> – Name of the Service (mandatory)</li> <li><b>CATEGORY</b> – e.g. Network L1...L4, VOIP call Control, Mobility</li> <li><b>TAGS</b> – User-definable tags (default name of the PNF)</li> <li><b>DESCRIPTION</b> – Textual description of service (mandatory)</li> <li><b>CONTACT ID</b> – Designer (user of ONAP) (mandatory)</li> <li><b>PROJECT CODE</b> – ID (mandatory)</li> <li><b>Ecomp-Generated Naming</b> – Name</li> <li><b>Naming Policy</b> – Policy to be used to assign a name to a service by SO/SDNC</li> <li><b>SERVICE TYPE</b> – Type of service</li> <li><b>SERVICE ROLE</b> – The Role of this service.</li> <li><b>ENVIRONMENTAL CONTEXT</b> – distributed environments</li> <li><b>Specific Service(?)</b> – PNF, allotted resource from a CU Service</li> </ul> <p>The “basic” model are extended. Inherit (OO) from existing model. Vendor takes standard node types and creates their own extension.  CDT (Configuration Design Tool) (GUI) to build artifacts to be used by APP-C (Tosca models) for a configure Template.</p>
3	<p><b>DISTRIBUTION</b> – Event Monitoring Templates distributed. (?)</p>



# Deployment Handler (VES Collector)



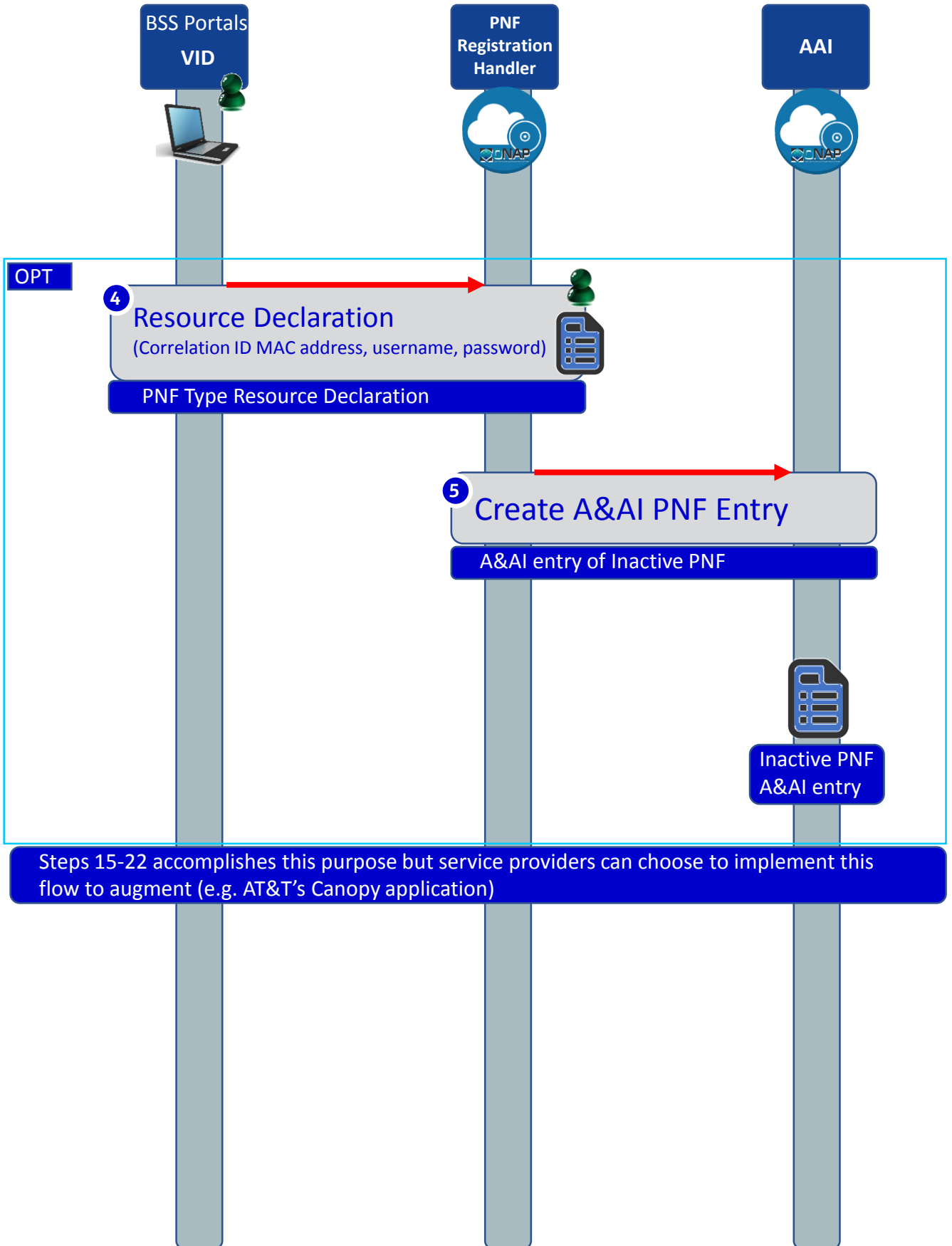
To make a VES Collector in DCAE work properly

PNF Blueprint will be docker container. May be deployed from CLAMP or Cloudify Orchestrator.

Within VES Collector schema uses "Adder" (existing)

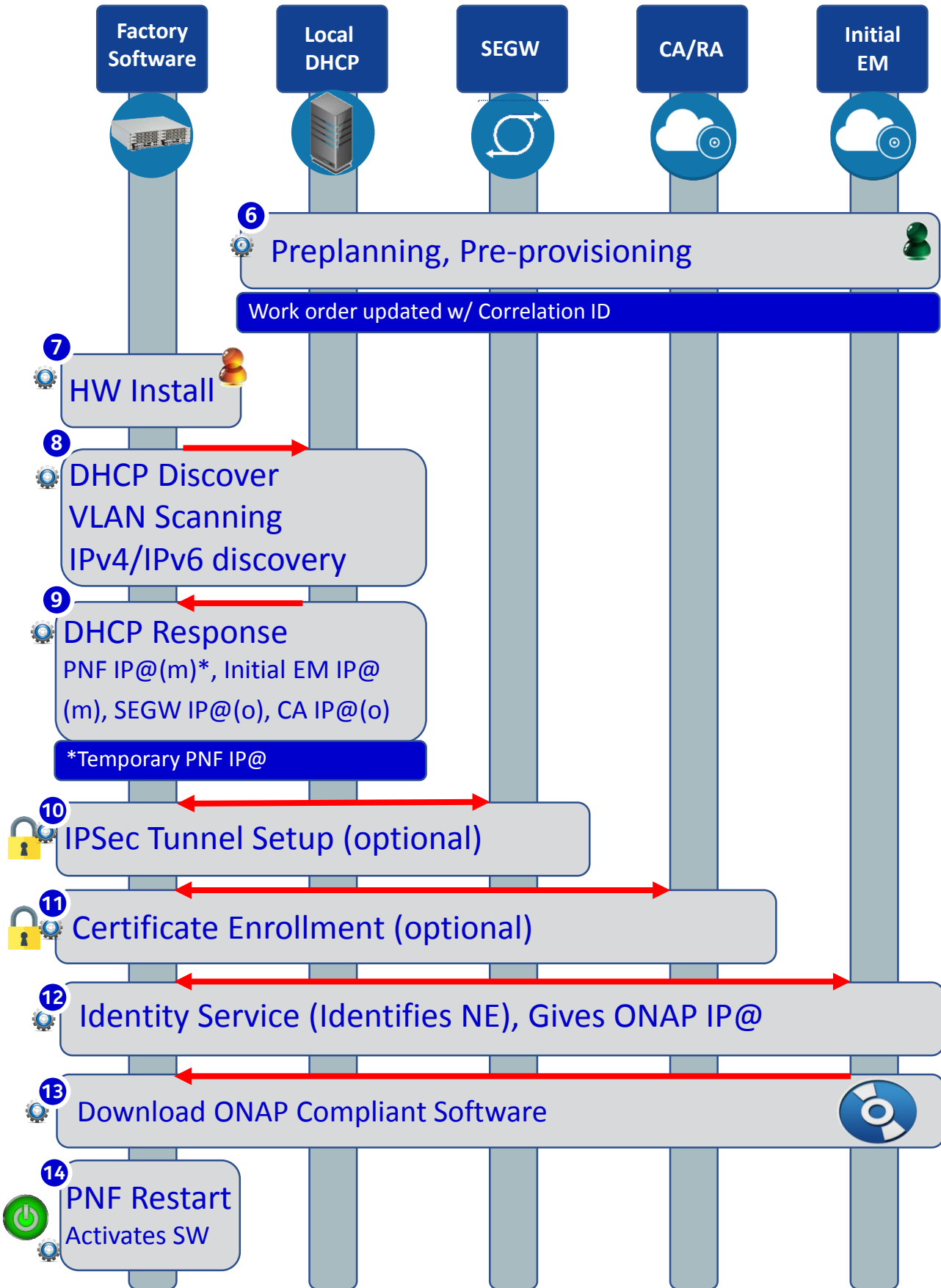
STEP	DESCRIPTION
1	<p><b>DEPLOYMENT HANDLER PNF BLUEPRINT</b> – The Deployment handler is necessary to use, to give it the PNF blueprint so that later on in the Plug and Play flow, DCAE will recognize the PNF. The PNF Blueprint will be docker container. The PNF blueprint May be deployed from CLAMP or Cloudify Orchestrator to DCAE. Within VES Collector schema uses “Adder” (existing).</p>










# PNF Resource Declaration



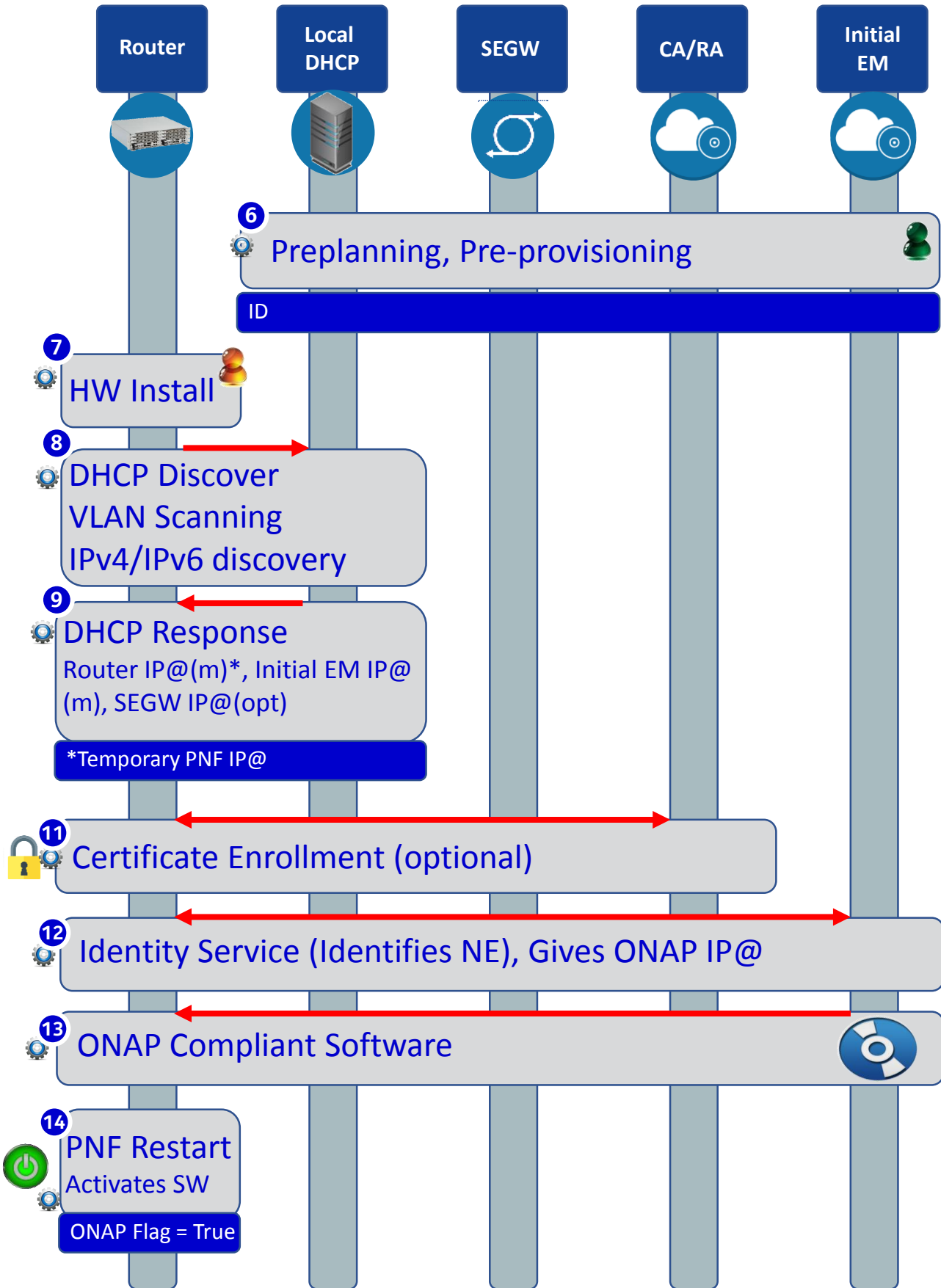
STEP	DESCRIPTION
4	<p><b>RESOURCE DECLARATION</b> – A user on the VID performs a Resource Declaration. This uses the Service definition created in SDC. The user on the VID can define known information about the PNF. The user can (optional) provide the following information</p> <p><b>PNF RESOURCE Definition</b></p> <ul style="list-style-type: none"> <li><b>Resource Type</b> – Type of Resource. NEW type: PNF (pre-defined in SDC)</li> <li><b>NAME</b> – Name of the PNF type</li> <li><b>CATEGORY</b> – e.g. Infrastructure</li> <li><b>TAGS</b> – User-definable tags (default name of the PNF)</li> <li><b>DESCRIPTION</b> – Textual description</li> <li><b>CONTACT ID</b> – Designer (user of ONAP)</li> <li><b>VENDOR</b> – PNF Vendor (e.g. Nokia)</li> <li><b>VENDOR RELEASE</b> – Vendor release</li> <li><b>VENDOR MODEL NUMBER</b> – PNF Model value (link to A&amp;AI)</li> <li><b>EVENTS</b> – Monitoring Event definitions. Define design-time templates. .</li> </ul>
5	<p><b>PNF Registration Handler (PRH) creates an A&amp;AI Entry:</b></p> <p>The PNF entry in AAI is created by the PRH with the PNF ID (or correlation ID) address. After this step, the PNF is considered to be created but inactive within ONAP. It can then be later used in the second part of service instantiation. See Step 28.</p>









# PNF Plug and Play Steps (for 5G DU)



STEP	DESCRIPTION (Plug and Play Vendor steps for Infrastructure components)
6 	<p><b>PRE-PLANNING, PRE-PROVISIONING</b> – There is data which is programmed into the system for the PNF Plug and Play operation. The user programs the local DHCP IP address(@), the Security Gateway IP@, the CA/RA certificate information, the management plane IP address (the ONAP IP@), the software service IP@ for use by the PNF during the onboarding process.</p> <p>Note: <i>The CU is instantiated ahead of time with the expected DUs that it should be connected to (that is outside the scope of this flow).</i></p> <p>Note: The user name &amp; password which the PNF needs to know to contact and get through the vAAA server before it contacts ONAP.</p>
7 	<p><b>HW INSTALL</b> – The physical hardware is installed at the site. Site licensing, real estate contacts, zoning, and physical hardware of the PNF is installed by technicians. Power, backhaul, and antennas are installed and connected.</p>
8 	<p><b>INITIAL NETWORK ACCESS</b> – A DHCP Discover procedure is executed when the PNF powers on, VLAN Scanning is performed, and IPv4/IPv6 discovery is done. The DHCP Discover message exchange provides an entryway into the network and is designed as an procedure for a network element to be able to find connection to the network from “scratch”. VLAN Scanning and IPv4 vs IPv6 discovery is done as well.</p>
9 	<p><b>DHCP RESPONSE</b> – The DHCP response returns a PNF IP address, the initial EM IP address, Security Gateway IP address (optional), and certificate authority IP address (opt). It is possible the PNF IP address is a temporary IP address used for initial connectivity purposes, and that a permanent PNF IP address will be granted later.</p>
10 	<p><b>IPSEC TUNNEL</b> – An IP Sec Tunnel is established which uses cryptography to provides a secure connection. IPSec has two security services: Authentication header and an encapsulating security payload with tunnel and transport modes.</p>
11 	<p><b>CERTIFICATE ENROLLMENT</b> – The process where the PNF gets a service provider certificate from the Certificate authority. The certificate is then used to authenticate and verify the PNF.</p>
12 	<p><b>IDENTITY SERVICE</b> – The identity service is there to identify the PNF. It also returns the ONAP (DCAE) IP address.</p>
13 	<p><b>ONAP COMPLIANT SOFTWARE</b> – The PNF contacts the initial EM and downloads the ONAP Bootstrap software. This is a software package that is meant to perform the remaining steps of PNF registration and activation onto ONAP</p>
14 	<p><b>PNF RESET</b> – The PNF is reset so that the downloaded ONAP Bootstrap software becomes activated and is then ready to continue to PNF registration</p>

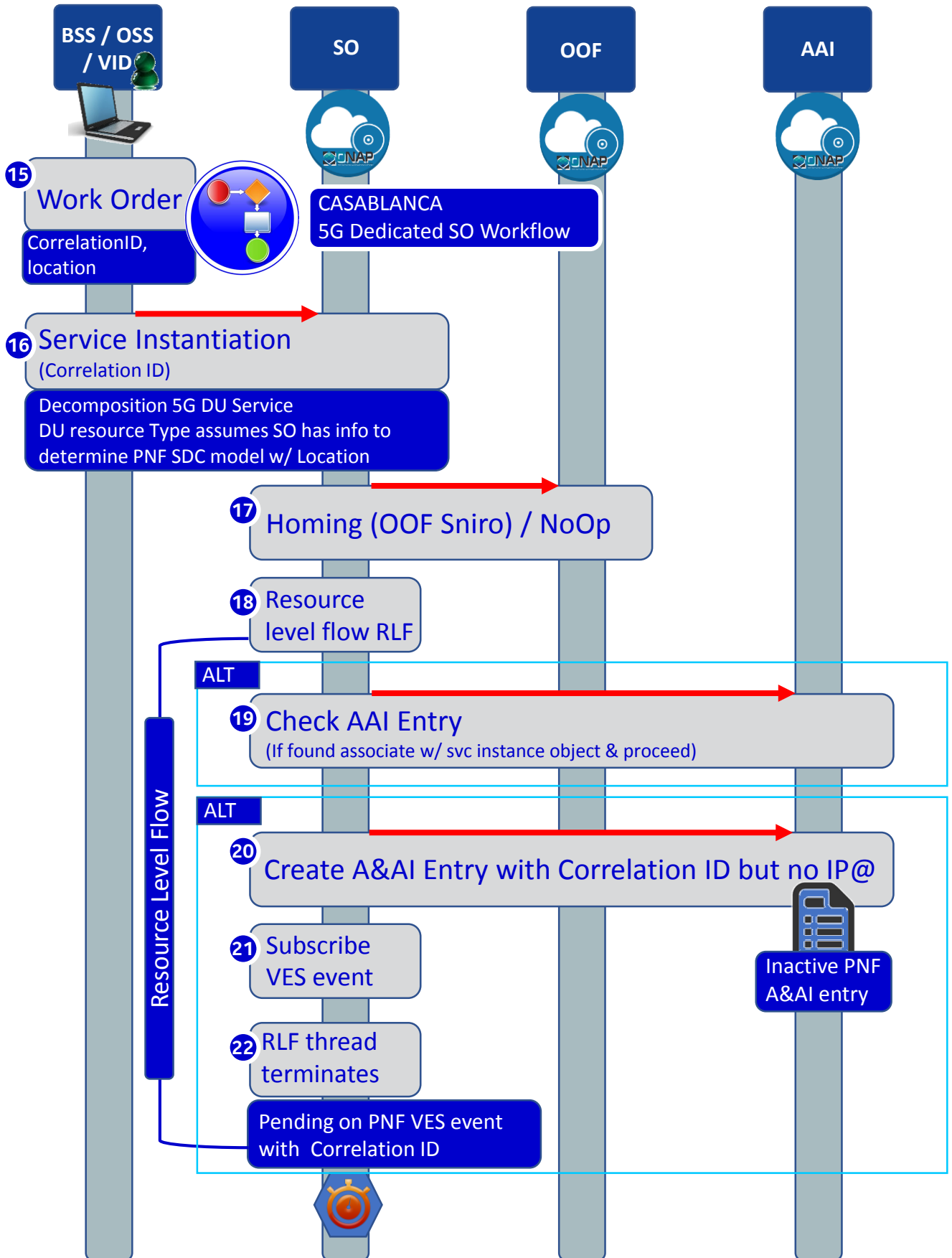
# PNF Bootstrapping Steps (for Routers)



STEP	DESCRIPTION (Plug and Play Vendor steps for Infrastructure components)
6 	<p><b>PRE-PLANNING, PRE-PROVISIONING</b> – There is data which is programmed into the system for the PNF Plug and Play operation. The user programs the local DHCP IP address(@), the Security Gateway IP@, the CA/RA certificate information, the management plane IP address (the ONAP IP@), the software service IP@ for use by the PNF during the onboarding process.</p> <p>Note: <i>The CU is instantiated ahead of time with the expected DUs that it should be connected to (that is outside the scope of this flow).</i></p> <p>Note: The user name &amp; password which the PNF needs to know to contact and get through the vAAA server before it contacts ONAP.</p>
7 	<p><b>HW INSTALL</b> – The physical hardware is installed at the site. Site licensing, real estate contacts, zoning, and physical hardware of the PNF is installed by technicians. Power, backhaul, and antennas are installed and connected.</p>
8 	<p><b>INITIAL NETWORK ACCESS</b> – A DHCP Discover procedure is executed when the PNF powers on, VLAN Scanning is performed, and IPv4/IPv6 discovery is done. The DHCP Discover message exchange provides an entryway into the network and is designed as an procedure for a network element to be able to find connection to the network from “scratch”. VLAN Scanning and IPv4 vs IPv6 discovery is done as well.</p>
9 	<p><b>DHCP RESPONSE</b> – The DHCP response returns a PNF IP address, the initial EM IP address, Security Gateway IP address (optional), and certificate authority IP address (opt). It is possible the PNF IP address is a temporary IP address used for initial connectivity purposes, and that a permanent PNF IP address will be granted later.</p>
11 	<p><b>CERTIFICATE ENROLLMENT</b> – The process where the PNF gets a service provider certificate from the Certificate authority. The certificate is then used to authenticate and verify the PNF.</p>
12 	<p><b>IDENTITY SERVICE</b> – The identity service is there to identify the PNF. It also returns the ONAP (DCAE) IP address.</p>
13 	<p><b>ONAP COMPLIANT SOFTWARE</b> – The PNF contacts the initial EM and downloads the ONAP Bootstrap software. This is a software package that is meant to perform the remaining steps of PNF registration and activation onto ONAP</p>
14 	<p><b>PNF RESET</b> – The PNF is reset so that the downloaded ONAP Bootstrap software becomes activated and is then ready to continue to PNF registration</p>



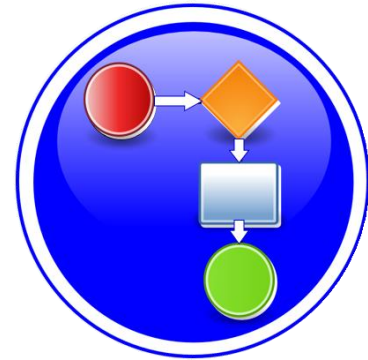
# Service Instantiation Process (Part 1)



# PNF PnP: SO Enhancements


## DESCRIPTION

- (1) PNF WORKFLOW** – Coded in Beijing but needs yet to be integrated, tested, and accepted. Note: One PNF workflow (SO) that has two entry points: (1) instantiate & wait (Step 16) (2) Update (Step 34). One BPMN.
- (2) SO TO SDN-R INTERACTION** – SO calls SDN-R (Generic API call vs REST call (?)). How should SO recognize which controller to use. List of operations.
- (3) PNF WORKFLOW ENHANCEMENTS** – Investigate the use of a Generic PNF workflow (will it work for 5G PNF). Investigate if enhancements (customizations) are needed to the PNF Workflow.
- (4) MODEL DRIVEN** – SO not yet model driven. Need to solve vis-à-vis a SO work flow specific to service & resource use case. *Controller type* to be used w/ PNF should be modeled in SDC.
- (5) DFX (Design for Excellence)** – Resilience, Performance, Scalability, Stability, Multi-tenants (multi-tenants access). Multi-site (single instances of SO).

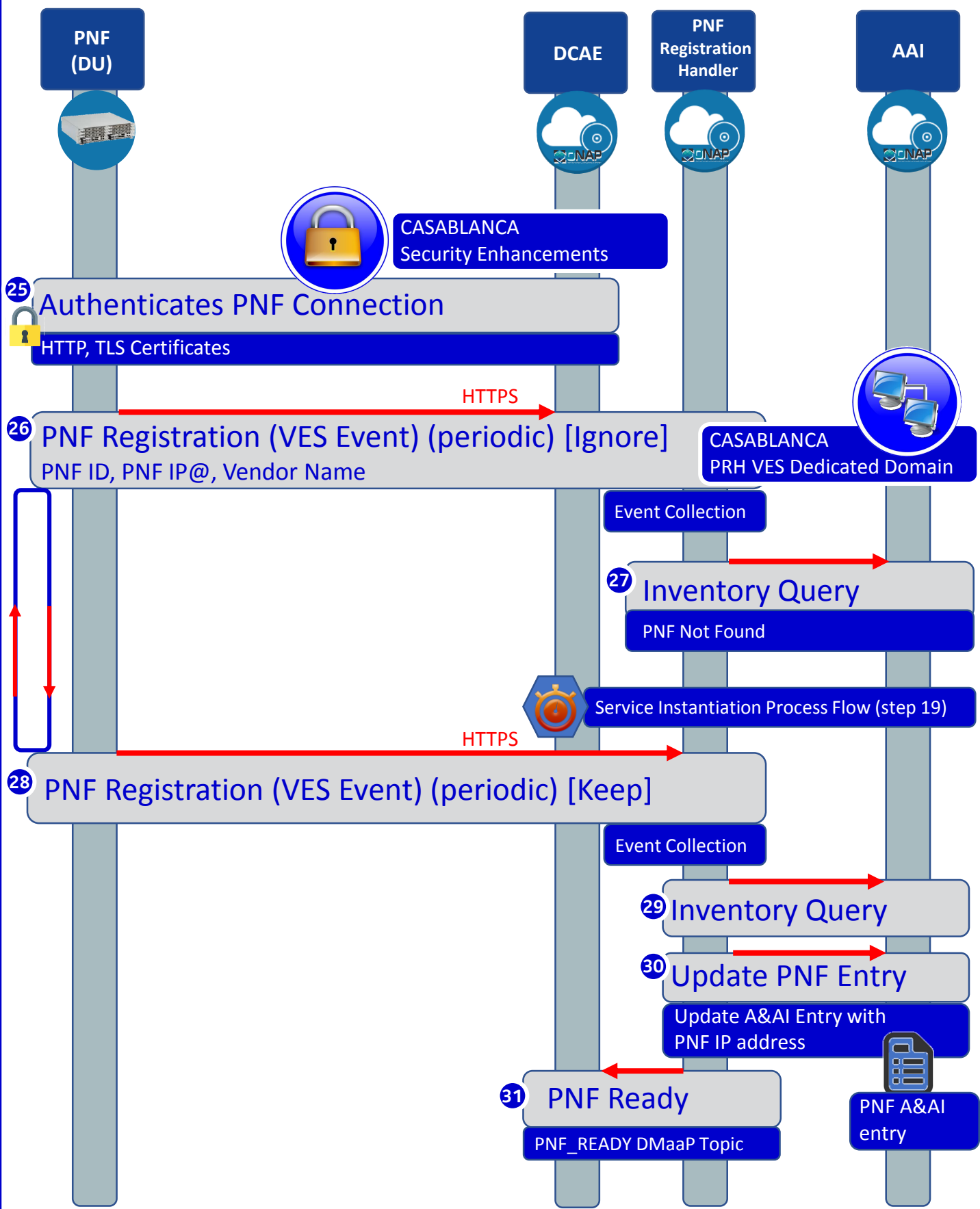


## PROJECTS

SO, SDN-R, SDC, AAI

#	DESCRIPTION
15	<b>WORK ORDER</b> – The work order (AT&T work order process) determines which PNF to use for this Service/work order. BSS is told the correlation ID. Typically, the PNF will be known before the PNF comes on-line. Orchestrated with a equipment order (to vendor) and location value for the PNF.
16	<b>SERVICE INSTANTIATION</b> – The user on the VID creates a service instantiation providing a correlation ID. The service is decomposed for a 5G DU PNF. The DU resource types assumes that there is enough information to determine the PNF SDC model. The configuration parameter is provided manually (as part of service instantiation data).
17	<b>HOMING</b> – The SO instantiation is homed with the OOF. Dependencies stated on PNF. The homing latency constraints are based on CPE address (location). As part of the service definition a latency needs to be less than [x]. This is a service constraint. Homing dependencies should come from SDC or VID. PNF homes to a CU. [FUTURE] Homing identifies the place where a VNF is instantiated. For PNF there is no “cloud” resources needed (CU); ONAP instantiations/data-centers want to define which ONAP instantiation takes care of a PNF [FUTURE].
18	<b>RESOURCE LEVEL FLOW (RLF)</b> - The resource level flow thread starts. This thread is responsible for carrying out the creation of an A&AI entry in the following steps (steps 18 through 21).
19	<b>CHECK A&amp;AI ENTRY</b> – The RLF thread in SO checks the A&AI entry for the PNF. If SO discovers that there is an A&AI entry with both the correlationID and the PNF IP@ then it can continue. If found it can associate it with the service instance.
20	<b>CREATE A&amp;AI ENTRY</b> – A&AI entry created by SO for PNF using the available information and the correlation ID. This is done in anticipation of the PnP PNF VES event.
21	<b>CREATE DMaaP TOPIC LISTENER</b> – The RLF thread (process) subscribes to the DMaaP Topic that will complete the service instantiation. It allows ONAP to intercept the VES event that will eventually come from the PNF when it reaches a point in the PNF Plug and Play process that it is ready to contact ONAP. The RLF specific resource thread indicates that it cares about the VES event with this correlation ID. Essentially it activates a “listener” of the PNF VES event. DCAE is the normal VES event Listener which creates a DMaaP Topic. SO saves the state information looks and sees if it is one of the DMaaP topics it is waiting for.
22	<b>RLF THREAD TERMINATES</b> – The Resource Level Flow (RLF) thread in SO terminates. When the VES event is received at a later point in time, it can be processed accordingly. Additionally, these steps 15-22 prepare ONAP with the pre-requisite information so that when the VES event comes from the PNF it will not be discarded. This is denoted by stopwatch icon  . At a later step in the PNF Plug and Play this thread becomes relevant again at the other stopwatch icon. The RLF thread/process stops processing and wait for an asynchronous event (to avoid a long running event). Writes a process, kamunda handler for an event that rehydrates it. (this reuses the SO rainy day handling)

# PNF Registration Steps



# pnfRegistration VES Event

Field	Type	Required?	Description
version	number	Yes	Version of the event header (currently: 3.0) <b>3.0</b>
eventName	string	Yes	<b>pnfRegistration_vendor_pnfName</b> where <i>pnfName</i> is specified by the vendor and is a PNF type; e.g. <b>pnfRegistration_Nokia_5gDu</b>
domain	string	Yes	Event domain enumeration: 'fault', 'heartbeat', 'measurementsForVfScaling', 'mobileFlow', 'other', 'sipSignaling', 'stateChange', 'syslog', 'thresholdCrossingAlert', 'voiceQuality', ' <b>pnfRegistration</b> '
eventId	string	Yes	Event key that is unique to the event source <b>registration_yyyyyyyy</b> where <i>yyyyyyyy</i> is an integer starting at 0 and incremented by 1 for every pnfRegistration event sent by this PNF
eventType	string	No	<b>pnfRegistration</b>
nfcNamingCode	string	No	Network function component type: 3 characters (aligned with vfc naming standards) <b>Not used</b>
nfNamingCode	string	No	Network function type: 4 characters (aligned with vnf naming standards) <b>Not used</b>
sourceId	string	No	UUID identifying the entity experiencing the event issue (note: the AT&T internal enrichment process shall ensure that this field is populated) <b>Not used</b>
sourceName	string	Yes	Name of the entity experiencing the event issue <b>PNFid (unique PNF instance ID = PNF correlation ID = pnf-name stored in AAI ; e.g. NOK6061ZW3)</b>
reportingEntityId	string	No	UUID identifying the entity reporting the event, for example an OAM VM (note: the AT&T internal enrichment process shall ensure that this field is populated) <b>Not used</b>
reportingEntityName	string	Yes	Name of the entity reporting the event, for example, an EMS name. May be the same as the sourceName. For synthetic events generated by DCAE, it is the name of the app generating the event. <b>PNFid (unique PNF instance ID = PNF correlation ID = pnf-name stored in AAI ; e.g. NOK6061ZW3)</b>
priority	string	Yes	Processing priority enumeration: 'High', 'Medium', 'Normal', 'Low' <b>Normal</b>
startEpochMicrosec	number	Yes	the earliest unix time aka epoch time associated with the event from any component--as microseconds elapsed since 1 Jan 1970 not including leap seconds <b>current time</b>
lastEpochMicrosec	number	Yes	the latest unix time aka epoch time associated with the event from any component--as microseconds elapsed since 1 Jan 1970 not including leap seconds <b>current time</b>
sequence	integer	Yes	Ordering of events communicated by an event source instance (or 0 if not needed) <b>0</b>
internalHeaderFields	internalHeaderFields	No	Fields (not supplied by event sources) that the VES Event Listener service can use to enrich the event if needed for efficient internal processing. This is an empty object which is intended to be defined separately by each provider implementing the VES Event Listener. Empty <b>Not used</b>

# pnfRegistration VES Event

Field	Type	Required?	Description
pnfRegistrationFieldsVersion	number	Yes	Version of the pnfRegistrationFields block (currently: 1.0)
serialNumber	string	Yes	TS 32.692 serialNumber = serial number of the unit; e.g. 6061ZW3
vendorName	string	Yes	TS 32.692 vendorName = name of manufacturer; e.g. Nokia. Maps to AAI equip-vendor.
oamV4IpAddress	string	No <sup>1</sup>	IPv4 m-plane IP address to be used by the manager to contact the PNF. Maps to AAI ipaddress-v4-oam.
oamV6IpAddress	string	No <sup>1</sup>	IPv6 m-plane IP address to be used by the manager to contact the PNF. Maps to AAI ipaddress-v6-oam.
macAddress	string	No	MAC address of unit.
unitFamily	string	No	TS 32.692 vendorUnitFamilyType = general type of HW unit; e.g. BBU.
unitType	string	No	TS 32.692 vendorUnitTypeNumber = vendor name for the unit; e.g. Airscale. Maps to AAI equip-type.
modelName	string	No	TS 32.692 versionNumber = version of the unit from vendor; e.g. AJ02. Maps to AAI equip-model.
softwareVersion	string	No	TS 32.692 swName = active SW running on the unit; e.g. 5gDUv18.05.201.
manufactureDate	string	No	TS 32.692 dateOfManufacture = manufacture date of the unit in ISO 8601 format; 2016-04-23.
lastServiceDate	string	No	TS 32.692 dateOfLastService = date of last service in ISO 8601 format; e.g. 2017-02-15.
additionalFields	hashMap	No	Additional registration fields if needed, provided as key-value pairs.

# PRH Enhancements

*PNF Registration Handler (PRH) Enhancements.*

**(1) New VES Event domain for PNF Registration** – Create new VES event domain *pnfRegistration* for PNF registration with corresponding support in VES collector, DMaaP and PRH. PnP Use case was using the “*other*” domain to register VES events. For Casablanca, we propose using a dedicated domain. VES separation of events. VES agent update in PNF.

**(2) VES EXTENSIONS** - As a result, VES collector and VES agent content will change with field updates using the new domain. Extensions for PNF registration fields. Corresponding VES Schema change (VES message 6.0 standard)

**(3) REQUIREMENTS PROJECT** – Add requirements: for all equipment providers their PNFs have to use this event.

**(4) PNF REGISTRATION EXCHANGE UPDATE** – The registration VES event used by the DU Simulator (or actual PNF) will need to update its JSON payload to match the changes above.

**(5) AAF & PRH** – Intra-ONAP Security. PRH integration with AAF for security.

**(6) TOPIC CREATION** – Once DMaaP is secured, creation of new *pnfRegistration* DMaaP Topic. Topic needs to be pre-provisioned. *pnfReady* event.

**(7) FAILURE HANDLING** – Case of when registration fails. When & how is PNF notified when registration fails. (1) send response didn't find AAI keep sending (2) if not notified failure/success of AAI entry how to avoid multiple PNF ready to SO for same PNF. (3) AAI unavailable ONAP components not running (4) PNF sending erroneous info. (5) Failures on (a) PNF registration (b) DMaaP PNF Ready (c) AAI (d) Platform

**(8) SO INTEGRATION** – SO WF Integration.



## PROJECTS

PNF Registration Handler, DMaaP, DCAE

# SECURITY ENHANCEMENTS

## HTTP

- ONAP to eliminate username and password for HTTPS.

## TLS

- DCAE authenticates Vendor or Service Provider X.509v3 certificate for TLS connection.
- ONAP User is able to install Vendor Root CA certificate as trust anchor in ONAP.
- ONAP supports 3 level chain (Root CA, Sub CA and End-Entity certificates).
- Vendor is able to install ONAP Root CA as trust anchor on NF for mutual TLS authentication.

## CA

- Provide CA services to NFs in ONAP for development and testing; enrollment, authentication.
- Support CMPv2 for NF for certificate enrollment and end-entity key renewal in ONAP for development and testing.
- Support integration with Service Provider CA/PKI in ONAP.

## SSH/SFTP


- Create RSA public/private key pair for ONAP Controllers and File Collector for SSH/SFTP access into the NF for CM/PM.
- Provision SSH username/public key on NF for Ansible or NetConf access from ONAP Controllers for CM and SFTP.
- Client shouldn't accept username / password for SSH authentication.



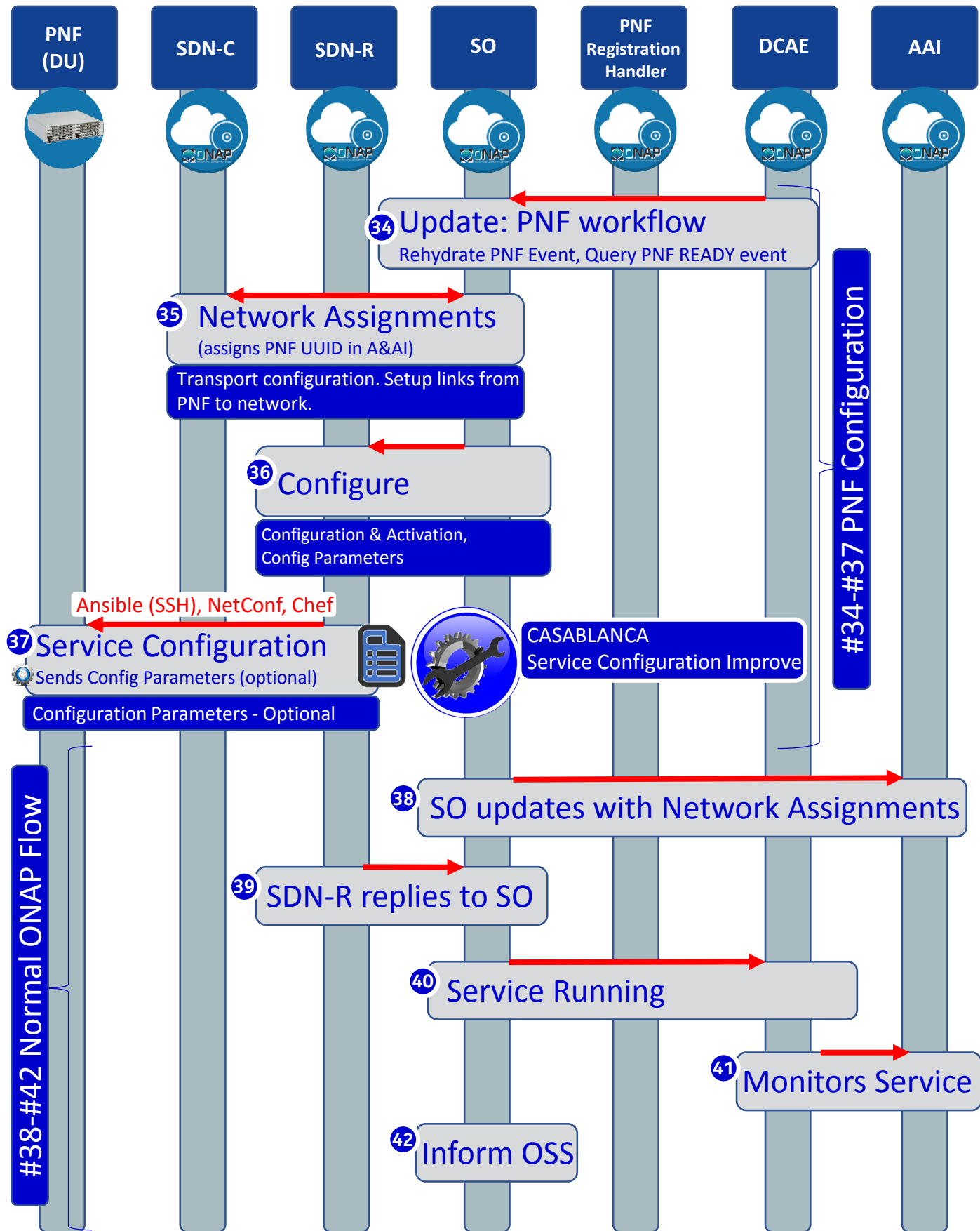
## PROJECTS

PNF Registration Handler, DCAE, AAF,  
ONAP Controller, DMaaP



STEP	DESCRIPTION
23	<b>DHCP Request</b> –ONAP Onboarding S/W performs a DHCP procedure with vDHCP
24	<b>DHCP Response</b> – DHCP response returns a ONAP IP address
25	<b>Authenticate PNF</b> – The PNF is authenticated through a vAAA.
26 & 28	<p><b>PNF REGISTRATION</b> – The PNF <b>periodically</b> generates a <b>VES Event (json schema extend for discovery event)</b> to DCAE which is the “triggering” event that tells ONAP that the PNF is trying to register. This event contains the Correlation ID (PNF ID), which will serve as an identifying key within A&amp;AI to seek for that particular PNF instance. The event also contains the PNF OAM IP address and the vendor name. PNF sends the Event over an HTTPS connection which may be authenticated with a username and password. The Event is “standardized” and is the same for all hardware (PNF) irrespective of equipment vendor, thus there needs to be ONAP compliant software. The PNF must natively support or have an adapter to be ONAP capable.</p> <p>Note [FUTURE]: A (new) “<i>PNF Infrastructure Manager</i>” may evolve in the future to include other management functions, as there may be a need for an entity that owns the interactions with device interactions w.r.t. ONAP. Management of devices vs containers. I/F manages &amp; consumes services. VIM/Multi-Cloud. Multi-VIM PNF plugin. Multi-VIM would call the PNF infrastructure manager.</p>
27 & 29	<b>INVENTORY QUERY</b> – PNF Registration Handler performs an inventory Query to A&AI using the Correlation ID (based on the PNF ID) as the key. The AAI instance for this PNF ID must have already been created. If it has, then this is a valid, expected PNF. If not, then this is not a valid or not found a response is given to the PNF. In Step 32, PNF A&AI entry is found.
30	<b>UPDATE PNF ENTRY IN AAI</b> – The PNF entry in AAI is updated with the PNF IP address. After this step, the PNF is considered to be active in ONAP and becomes available as an network element to fulfill service requests.
31	<b>PNF READY</b> - PRH publishes a PNF Ready event on the DMaaP bus to which SO subscribes. SO receives the PNF Ready event, determines that it is an event it is waiting for and rehydrates the appropriate RLF to restart the Service Instantiation
(15-22)	<b>SERVICE INSTANTIATION PROCESS (PART 1)</b> – Steps 15-22, the Service Instantiation Process (part 1) occurs in parallel to these steps. When that process reaches the pending point (denoted by  ) it rejoins the flow here. PNF previously declared.
32	<b>PNF REGISTRATION RESPONSE</b> - There is discussion whether in the failure cases of PNF registration if a response back from ONAP should be sent. Under consideration for Casablanca release (R3)

# PNF Activation Steps (ONAP)



# SERVICE CONFIGURATION ENHANCEMENT


## DESCRIPTION

- (1) Service configuration Enhancements** to ONAP Controller to PNF *service configuration* exchange with PNF.
- (2) PROTOCOL DEFINITION** – Framework for how SDN-R will support NetConf, Ansible, Chef. All 3 are expected to be supported with PNFs. How is the protocol agreement done?
- (3) Configuration Extensions** – New parameters needed for Casablanca use cases. Vid script to pushing data, ID config, ID where data comes from. Generic configuration support.
- (4) PNF PnP Config** – Finishing PNF PnP by sending down config data.
- (5) (OPTIONAL) Vendor Data** – Extensions to Service Configuration with Vendor specific configuration data can be developed (vendor dependent).

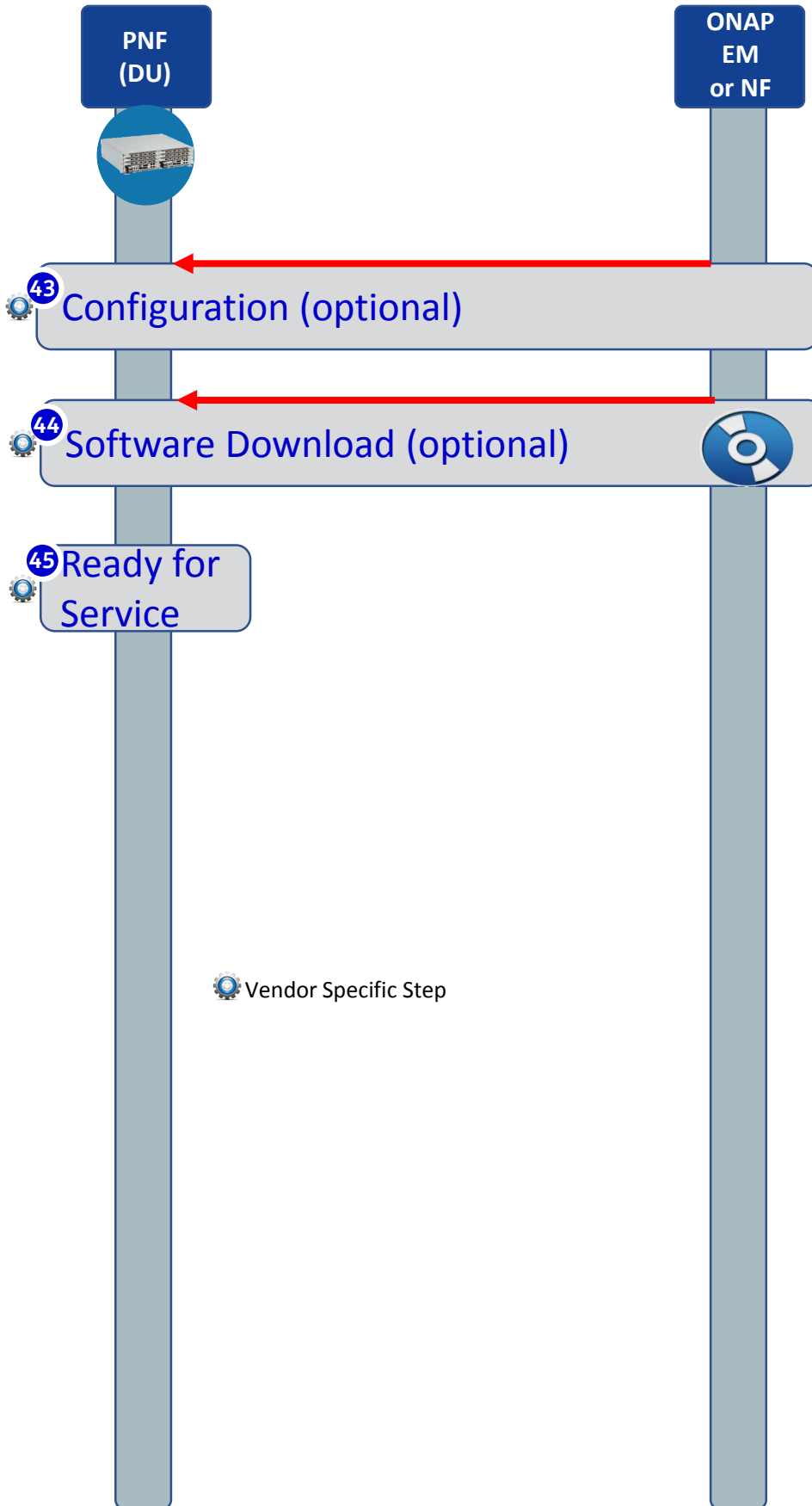





## PROJECTS

ONAP Controller (SDN-R), VID, SO,  
SDC/CDT

STEP	DESCRIPTION
34	<b>SO NOTIFIED</b> - PNF Infrastructure Manager Notifies SO. SO listens to the DMaaP hook. A trigger. Wait for PNF onboarded. Calls SDN-C.
35	<b>NETWORK ASSIGNMENTS</b> – SDN-C assigns an IP Address for PNF. The IP @ assigned to the PNF is drawn either from the DHCP server, IP address Pool, or a Static IP@. Managing physical/virtual links to PNF. Between CU & DU. Transport connectivity setup. SDNC makes assignments, the resource model have external/internal connection points (named). For each point, attributes say L1/L2 connection. If L3 who assigns the IP address. Each point, SDNC knows if and what to assign. Set either through SDNC (L0-L3) or APP-C (L4-L7). Driven by TOSCA Model.
36	<b>ACTIVATE</b> – Configuration & Activation of the PNF Depends on the resource type. The controller requires input data based on PNF type. Either VF-C or APP-C orchestrate with SO. The IP@ is retrieved SDNC instance manager for PNF and the DHCP server may be updated. Pass on configuration parameter(s). (Future) retrieve VNF a configuration parameter.
37	 <b>SERVICE CONFIGURATION</b> – APP-C calls Ansible to configure PNF’s (Configuration Parameter) NetConf messages from SDN-C to PNF. (1) <b>Configuration Parameter (optional)</b> – SDN-R/APP-C gives the Controller IP@ to the DU. In R3, SDN-R/APP-C may pass configuration parameter(s) to the 5G DU, this will also give a configuration parameter (e.g. CU IP@). (2) <b>OAM IP@ (optional)</b> - The permanent OAM IP address is given to the PNF. The IP address may come from vAAA, or drawn local pool of IP addresses. SDN-C performs the IP address selection. SDN-C knows if a permanent IP address should be assigned. (3) <b>Transport configuration (optional)</b> – Transport configuration is given to the PNF. (4) <b>Location (optional)</b> – the Location configuration may be given to the PNF. (5) <b>Software Version (optional)</b> – In Casablanca it could be specified a Software Version. <a href="http://onap.readthedocs.io/en/latest/submodules/appc.git/docs/APPC%20LCM%20API%20Guide/APPC%20LCM%20API%20Guide.html">http://onap.readthedocs.io/en/latest/submodules/appc.git/docs/APPC%20LCM%20API%20Guide/APPC%20LCM%20API%20Guide.html</a>
38	<b>SDN-C Updates A&amp;AI</b> – SO updates A&AI with Network Assignments (from step 35)
39	<b>SDN-C replies to SO</b> – SDN-C replies to SO are the service configuration step.
40	<b>Service Running</b> – SO publishes a “Service running” event to which DCAE subscribes.
41	<b>Monitors Service</b> - DCAE reads A&AI entry and sets up monitoring for the new service. DCAE publishes “Service monitored” event to which SO subscribes. For monitoring events, the DU will be managed by the CU (in the FUTURE an M-Plane will be setup to ONAP to DU
42	<b>OSS Inform</b> - SO responds to User/BSS/OSS that the service is active.

# PNF Final Download & Activation (Vendor Specific)



STEP	DESCRIPTION
43 	<b>CONFIGURATION (Optional)</b> – The configuration information is downloaded from [ONAP, EMS, or another PNF] to the DU. This information provides operational configurations and settings which are vital for service. They would be pre-provisioned and allow the PNF to operate with specified configurations, optimizations, RF settings, connectivity, and L1/L2 algorithmic settings.
44 	<b>SOFTWARE DOWNLOAD (Optional)</b> – Software download from [ONAP, EMS, or another PNF] to the DU. This software download is optional and is specific to a vendor.
45 	<b>DU RESTART</b> – The 5G PNF (DU) is ready to provide service using the configuration provided to it. Typically, a test call is performed to verify service is working end-to-end.

# VNF vs PNF Comparison

TOPIC	VNF	PNF
<b>Concept</b>	Application fulfills the role of a network function.	It is a network element, a physical entity, which can implement the role of a network function.
<b>Physical Characteristic</b>	Application without dedicated hardware; Virtualized applications require specific capabilities; Run on different vendor servers. SRIOV, Inter-DPDK. Hardware capabilities.	Has an actual physical asset that is deployed and associated directly with the PNF.
<b>On-boarding</b>	To onboard a VNF is to “bring it into ONAP” i.e. the VNF images, component VNF-C provide descriptors of these NFs. Deployment model, # components, functions. Configuration parameters. VNF is not tied or optimized for a specific hardware, only requiring perhaps some capability to be supported.	For PNF provide the descriptors. Only provide the meta-data. PNF S/W specifically optimized to run on dedicated hardware. (Now) Not the software image. (Future) ONAP will provide the software image repository.
<b>Plug and Play</b>	The model triggers the orchestration.	(See this slide package for PNF Plug and Play) at the end of PnP the PNF can provide service.
<b>Characteristics</b>	5G CU could be a VNF since there is no need to have an association to a physical environment.	5G DU must be PNF. PNFs are Elements which may need to interact with the physical environment. PNF is “High-Touch” technology. E.g. Emit radio waves in a geographical area.
<b>Configurability &amp; Deployment</b>	Easily adaptable to functions that you expect. E.g. Packet gateway to reconfigure as different NFs. Services easily create instances reconfigures including deployments (for different applications). Use a different instances of the VNF to provide a new service. For a VNF you can easily “delete” and “create” a new VNF to perform a new function. Configured dynamically.	PNF has a “fixed” set of capabilities but can’t easily reconfigure it. One PNF in multiple services. Different capabilities exposed by the PNF. Reuse the same PNF with different services configuration. For a PNF you would not “destroy” a PNF but rather re-configure it. Can be configured dynamically.
<b>ONAP Interaction</b>	ONAP is started with VNF. VNF is “deployed” on-demand. Control from the ONAP perspective when a deployment of a VNF happens. DCAE – same Configure – Chef, Ansible	PNF do not “deploy” application. Do not use multi-VIM. Only “configure” the application, the PNF is deployed. A technician goes to site and “deploys” a PNF. DCAE – same Configure – Implementation of PNF client. Communication protocol, Client
<b>Design Time Modeling</b>	Model VNF. Templates. Onboarded before. In Run-time. Make sure properly identify specific PNF instance already deployed. Vs a dynamically created instances. VNF instances could be created & instantiated dynamically. SDC may assumed instantiation of network function.	PNF cannot be instantiated, a PNF is only instantiated when it “powers up” and connects to ONAP. Service Orchestration. PNF is instantiated by nature of a PNF installation & commission procedure.
<b>Service Orchestration</b>	VNF cloud, #VM resources consumption, define components implement different functions. Where & What will be deployed.	Physical location, pre-provisioned capabilities, performance monitoring. Components installed. RUs for specific functions.
<b>Resources</b>	VNF dynamically assigned resources.	PNF statically associated (hardware) resources.
<b>Capacity</b>	VNF Capacity can be dynamically changed	PNF is static (number of cells supported)



# PNF Plug and Play Project Impacts

- ONAP and PNF Plug and Play for 5G RAN
- 5G Use Case Team



# PROJECT IMPACTS FROM ONBOARDING

ONAP Project	IMPACT
<b>SDC/ Modeling</b>	Modeling Project - (No License management impact – See <b>Futures</b> Section) [See follow-on Slides “MODELING”]
<b>VNF-SDK (PNF-SDK) Validation</b>	<p><b><u>PNF PACKAGE DEFINITION</u></b></p> <p>PNF packages similar to VNF packages. PNF Descriptors, artifacts. In PNF not doing deployment process in SDC. Only PNF configuration. Model a PNF. Onboard PNFs (create templates service configuration). Orchestrate a service on a PNF. Service provisioning. Life cycle: Template/service orchestrated. For PNF every PNF vendor makes this package. Need specific PNF properties. Image details. ONAP updates the image.</p> <p><b><u>VALIDATION OF A SDK PACKAGE</u></b></p> <p>VNF-SDK (validation, Package definition, verification tool) – package compliant. Allows creation/validation of packages. PNF-SDK <i>validates</i> the package.</p>
<b>WORK FLOW (SDC)</b>	Create work-flow for PNF (Srini) SDC Impacts related to work-flow. ACTION: Sample Work-flow for PNF.
<b>MONITORING (SDC)</b>	Monitoring definitions – SDC has a side monitoring template designer. Way to define monitoring alarms etc; In AT&T there is a project; IN ONAP code is there finalizing code; pluggable modeler for monitoring. DCAE as part of onboarding specify what VES template. DCAE-DS [Design Studio] define microservices for monitoring. How is PNF monitored & correlated. If [x] goes down how is this correlated. SDC would define the Modeling what needs to be monitored and how they would correlated with other events from other NE. Thresholds. [Baby step to process get an alarm from PNF, YAML file describes fault VES event, Fault meta-data; alarms generate]. Alarms raises are documented in SDC. Upload an “Artifact” file (Alarm Dictionary / Fault Meta-data / YAML, YANG). Vendor Specific. Demo and separate discussion.  How monitored – processing in DCAE-DS (Design Studio time). Based on design time data DCAE is done in.

# DCAE-DS IMPACTS

**OVERVIEW** - DCAE-DS generates the **templates** for monitoring the models. DCAE-DS is model-driven. It specifies which monitoring microservice are utilized in monitoring a specific service model. Cloudify blueprints specify the requirements on micro-service and are configured by a user. Configurations are distributed to components who subscribe to that specific type of artifact.

**DCAE-DS TEMPLATES** – monitoring templates composition of micro-service to be used (open/closed loop). The templates are *Cloudify Blueprints*. E.g. Micro-Service collectors, analytics, monitoring. VES collectors, holmes. A micro-service that is part of a monitoring flow that a designer can design that can be reused for difference service models. A building block represented by TOSCA models. First needs to be represented by development team. Monitoring template certified.

**DCAE-DS GUI** - DCAE-DS is a pluggable designer in SDC provides a GUI to the user that selects/composes the micro-services, or use predefined templates, for specific flows. E.g. SNMP type of flow or different protocol. User can configure different micro-services according to requirements to the model.

**PNF PLUG AND PLAY** – Cloudify Blueprints (for a [1:X] service) has (UUID, Micro-service values, Properties, service specific policies). What are we trying to Monitor? SDC Service-Package attached to VF-Level.

# MODELING IMPACTS

Notes:

1) **EXTERNALS** - Not trying to model the internals of PNFs. What is exposed by the box is what is modeled.

2) **INTERRELATIONS** - Focus on relations of PNFs/VNFs. Interworking between PNFs/VNFs.

3) **VISIBILITY** - CP/UP visibility

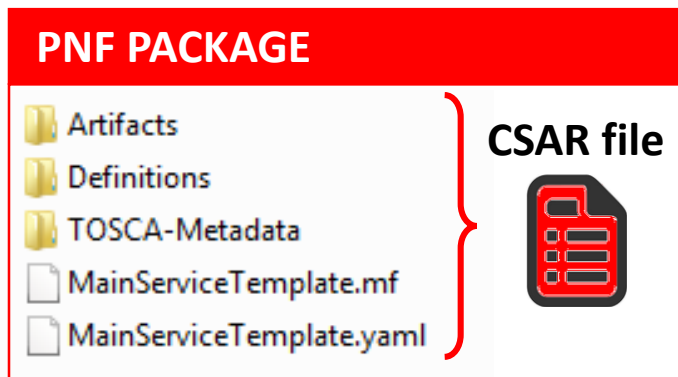
Not M-Plane (as this is 3GPP standardized)

4) **MODELING ANALYSIS** - Modeling activity to assess PNF, and check SDC model is sufficient to cover Casa use cases if additional parameters need to be added (e.g. relations between other NFs). Expanding the “Release 0 model” for Casa. PNF type vs PNF instance. Design-time vs Run-time model.

# PNF PACKAGE

PNF has no onboarding package.

Just model the PNF from the modeling screens.



CSAR – decompile info stored in SDC model.

In VNF flow. Onboard the VNF. VNF cataloged as a version to be used. Check-in/check-out. After onboarding can add more artifacts and certify the VNF. A “building block” to be used in different services. Generic, the structure will be the same. E.g. 2000 ports vs 10 ports. “Ports”. How to comm w/ PNF what to do w/ PNF. Specific work-flow or configuration. PNF & VNF similar. SO will orchestrate, already exists in ecosystem.

Modeling of the Service. E.g. Connection point what will connect to the PNF.

ARTIFACTS

DEFINITIONS

PNF – 5G Base Stations

**Backhaul Ports** – PNF & VNF and want to communicate. In a VNF can describe a port a TOSCA.

Model onboarded understand what can connect to what. CP connections. Can see they can connect. Model needs to capture info for modeling parts representing connections. Model allow someone designing service to connections. Or requirements from VNF/PNF from the model. A virtual link. One VNF & PNF connected via virtual network/link.

PNF Work-flows – initialization, triggered when connecting to PNF. Configuration/registration that needs to be done. DNS pre-loaded. Location. Policies attached to PNF, High volume # of PNF deployments, port-allocation. *Capabilities. Triggered by orchestrator as part of the instantiation.*

PNF Policies -

Tilt – (Antenna Tilt - RF) – not related to PNF / VNF communicate.

Software Version

Modeling Project, VNF-SDK (validation, Package definition, verification tool) – package compliant

# PNF PACKAGE

## CONFIGURATION DEFS

Configuration  
Schema

(Vendor provided)



YAML Definitions

(Vendor provided)



## ALARM DEFINITIONS

Alarm Dictionary

(Vendor provided)



YAML Definitions

(Vendor provided)



PNF Alarm #22

#22 S/W problem xyz

VES = fault fields; "fault" domain publish DMaaP  
Clamp, Analytic, Holmes – [#22 event]

Dictionary looks up #22 ->

## NF PACKAGE

Artifacts

Definitions

TOSCA-Metadata

MainServiceTemplate.mf

MainServiceTemplate.yaml

CSAR file



Need to define a common "template" that all vendors  
Provide dictionaries for. DCAE-DS responsible for  
reading these files. Closed Loop. Vendor describe  
alarms being sent. Definition what alarms correlated  
to (collecting, triggers). CLAMP performs analysis,  
design of who uses it is SDC. Holmes monitoring  
micro-services (to receive alarms).

Communication STANDARDS we want to support?  
[Data Format]

## MEASUREMENT DEFS

Measurement  
Dictionary

(Vendor provided)



Measurement  
Schema

(Vendor provided)

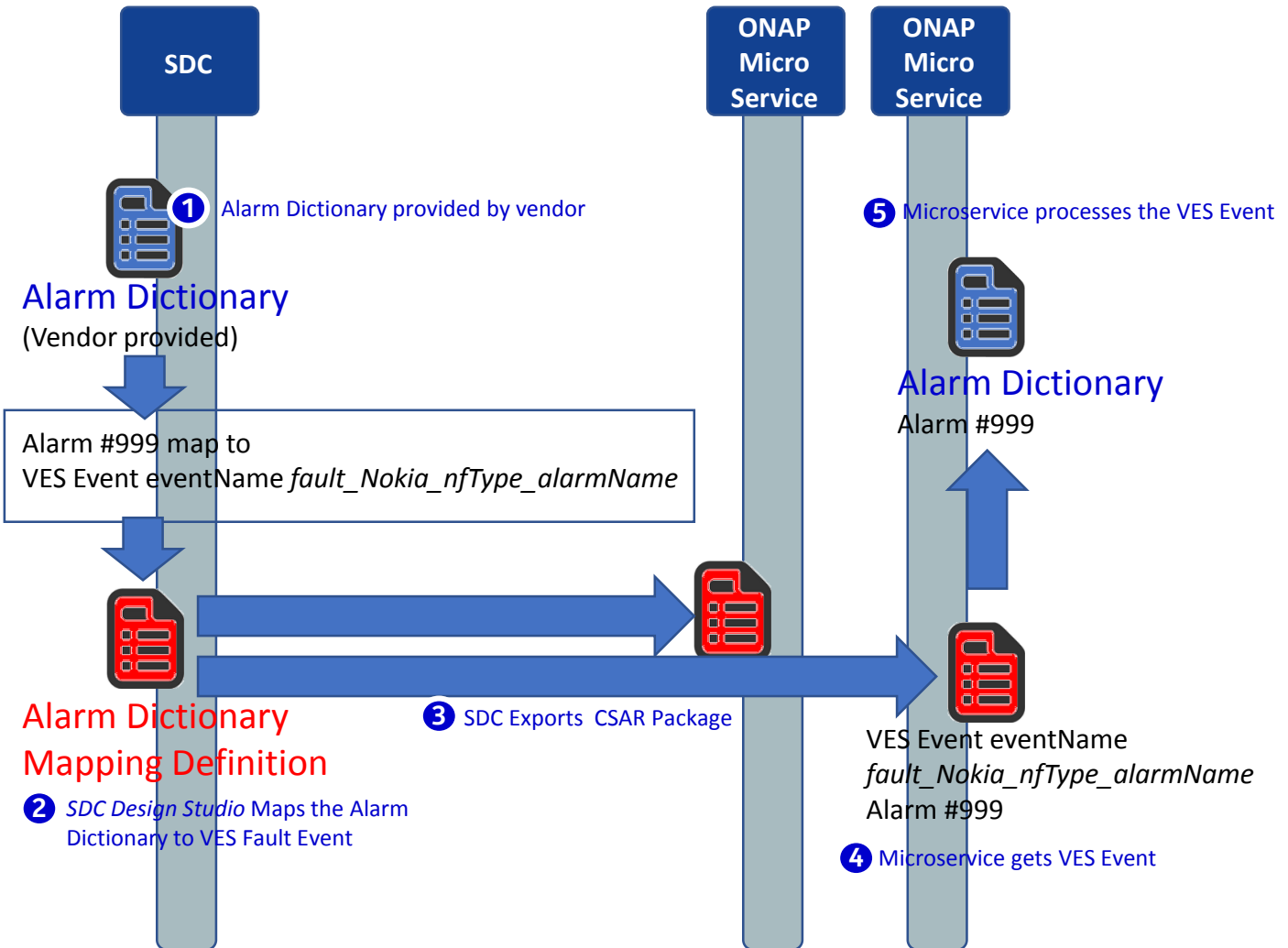


YAML Definitions

(Vendor provided)



# Alarm Dictionary Usage



## 1 ALARM DICTIONARY

Alarm Dictionary defines all alarms/faults published by xNF (x=V or P)  
Based on 3GPP TS32.111, ETSI, and VES document (v6.0)

## 2 SDC DESIGN STUDIO MAPS VES FAULT Event to Alarm Dictionary

SDC Design studio does mapping of Alarm dictionary entries to VES Fault Events to produce a "mapping" definition in the CSAR package.

## 3 SDC DISTRIBUTES DEFINITIONS

SDC creates a definition of how to start-up service. SDC exports the *CSAR package* with the VES Fault to Alarm dictionary mapping definitions.

### DEFINITIONS & ARTIFACTS (CSAR PACKAGE)

1. VES-Alarm mapping definition passed by SDC (CSAR Package) to ONAP components
2. DCAE micro-service gets CSAR package

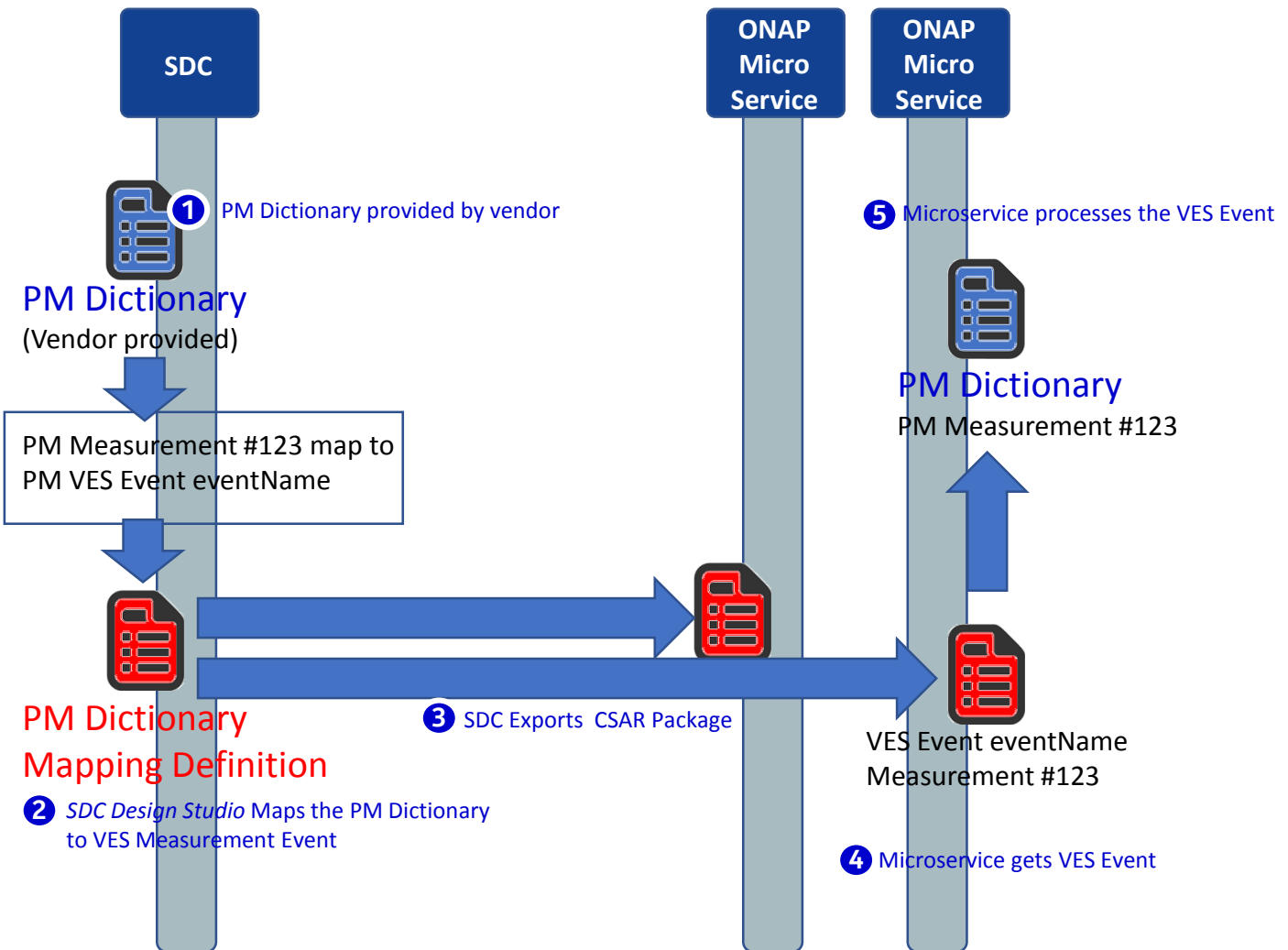
## 4 MICRO SERVICE GETS FAULT EVENT

Microservice has subscribed to fault domain DMaaP Topic and receives the VES Fault Event from xNF in run-time.

## 5 MICRO SERVICE PROCESSES EVENT

Microservice processes the VES Event using the Alarm Dictionary

# PM Dictionary Usage



## 1 PM DICTIONARY

PM Dictionary defines all measurements published by xNF (x=V or P)  
Based on 3GPP TS32.111, ETSI, and VES document (v6.0)

## 2 SDC DESIGN STUDIO MAPS VES FAULT Event to PM Dictionary

SDC Design studio does mapping of PM dictionary entries to VES Fault Events to produce a “mapping” definition in the CSAR package.

## 3 SDC DISTRIBUTES DEFINITIONS

SDC creates a definition of how to start-up service. SDC exports the *CSAR package* with the VES Fault to PM dictionary mapping definitions.

### DEFINITIONS & ARTIFACTS (CSAR PACKAGE)

1. VES meas mapping definition passed by SDC (CSAR Package) to ONAP components
2. DCAE micro-service gets CSAR package

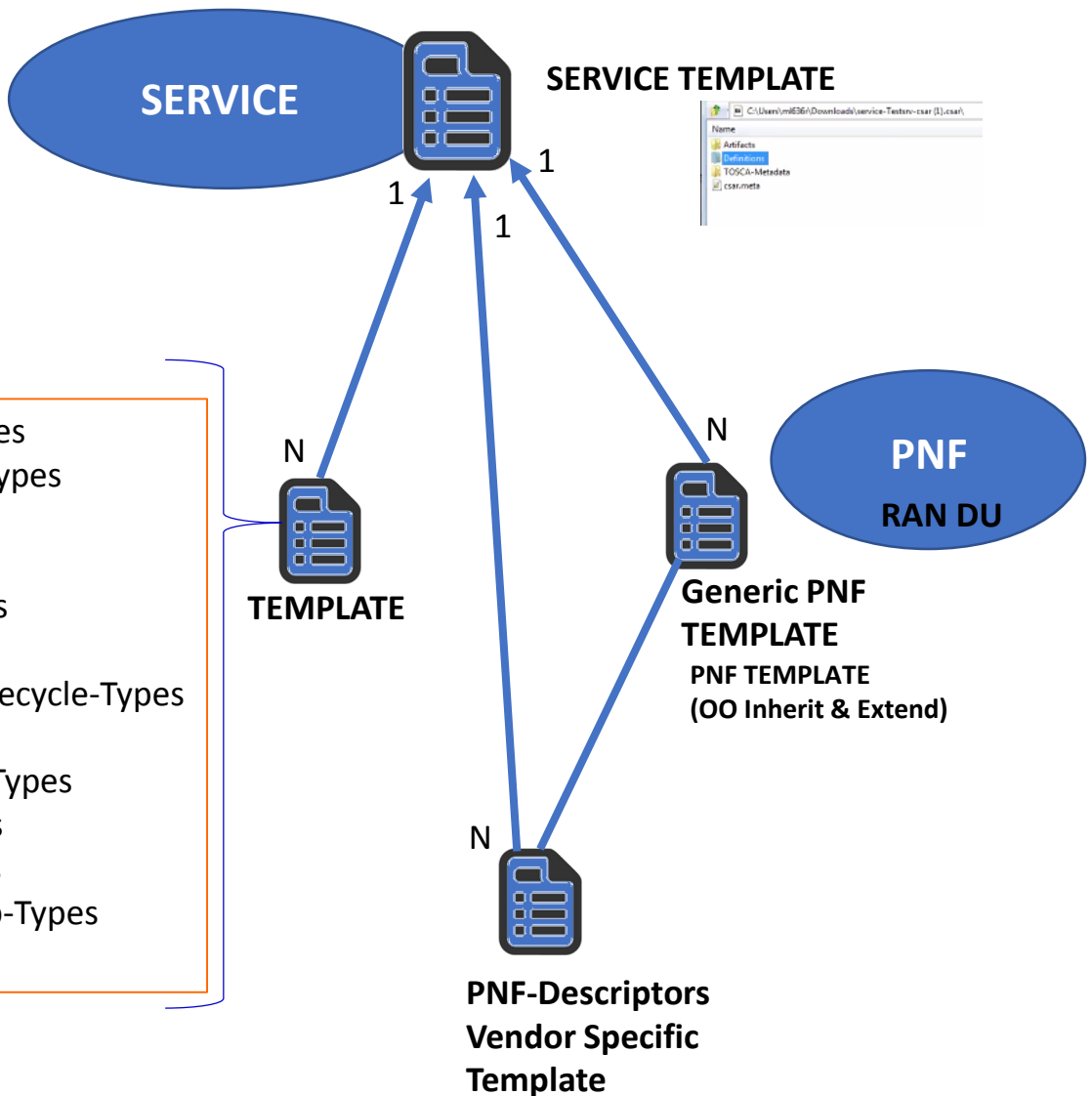
## 4 MICRO SERVICE GETS MEASUREMENT EVENT

Microservice has subscribed to fault domain DMaaP Topic and receives the VES Fault Event from xNF in run-time.

## 5 MICRO SERVICE PROCESSES EVENT

Microservice processes the VES Event using the PM Measurements Dictionary

# SDC MODELING (Design Time)



Name	Type
artifact-type	ST
capability-type	ST
category	ST
category	ST
group-type	ST
heat-type	ST
interface-lifecycle-type	ST
nf-v-type	ST
normative-type	ST
onap-type	ST
policy-type	ST
relationship-type	ST
users	ST

```

sdccatalog-be > src > main > resources > import > toasca > heat-types > Generic_PNF
Open New folder
Name Date modified Type Size
Generic_PNF 9/10/2017 12:49 JSON File
Generic_PNF 9/10/2017 12:49 YAML File
Generic_PNF 9/10/2017 12:40 Compressed (zip...)
tosca_definitions_version: toasca_simple_yaml_1_0_0
nodes_types:
- org.openecomp.resource.abstract.nodes.PNF:
  derived from: toasca.nodes.Root
  properties:
  nf_function:
    type: string
  nf_role:
    type: string
  nf_type:
    type: string
  
```

Controller Type  
[“common fields”] ...  
Vendor-field1



## PNFD Definition in ETSI-NFV-IFA014v242

Attribute	Qualifier	Cardinality	Content	Description
pnfdId	M	1	Identifier	Identifier of this Pnfd information element. It uniquely identifies the PNFD.
functionDescription	M	1	String	Describes the PNF function
provider	M	1	String	Identifies the provider of the PNFD.
version	M	1	Version	Identifies the version of the PNFD.
pnfdInvariantId	M	1	Identifier	Identifies a PNFD in a version independent manner. This attribute is invariant across versions of PNFD.
name	M	1	String	Provides the human readable name of the PNFD.
pnfExtCp	M	1..N	PnfExtCpd	Specifies the characteristics of one or more connection points where to connect the PNF to a VL. See clause 6.6.4.
security	M	0..1	SecurityParameters	Provides a signature to prevent tampering.
geographicalLocationInfo	M	0..1	Not specified	It provides information about the geographical location (e.g. geographic coordinates or address of the building, etc.) of the PNF. The cardinality 0 is used when the location is unknown.

## Basic Content of PNF template **PNF-D (DESCRIPTOR)**

Contents	Description
<b>pnfdId</b>	Identifier of this Pnfd information element. It uniquely identifies the PNFD.
<b>provider</b>	Identifies the provider of the PNFD.
<b>PNFD version</b>	Identifies the version of the PNFD.
<b>pnfdInvariantId</b>	<del>Identifies a PNFD in a version independent manner. This attribute is invariant across versions of PNFD. (pnfdInvariantId would be inside the meta-data in ONAP)</del>
<b>name</b>	Provides the human readable name of the PNFD.
<b>security</b>	Provides a signature to prevent tampering.
<b>pnfInformation</b>	Describes the PNF information
<b>pnfSoftwareVersion</b>	Software Version supported PNFD.

# VERSION MODELING

## Diagram of Software Version Management for a PNF



### PNF-Descriptor (version)

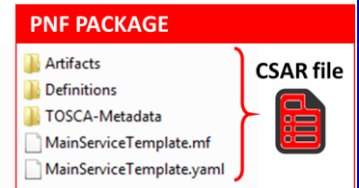
*Vendor Provided*

PNFD version = 6



### PNF-Package (version)

Minimal PNFPackage version = 7.1  
(SDC Versions the Package)



### PNF Software (version)

[Detected Software, Expected Software]

DETECTED PNF S/W [Part 1] 2.1 [Part 2] 2.2 (A&AI)

ONAP EXPECTED PNF S/W 3.0 (modeling)



### PNF Hardware (version)

Hardware Version

Firmware Version

Product Model version

# PNF MODELING Information (IN SDC)

Contents	Description
<b>pnfId*</b>	Identifier of this Pnf information element. CORRELATIONID (A&AI)
<b>pnfType (template)*</b>	Type of Resource. NEW type: PNF (pre-defined in SDC)
<b>Category*</b>	PNF category, e.g. infrastructure
<b>Vendor (template)*</b>	Identifies the vendor of the PNF.
<b>Name*</b>	Provides the human readable name of the PNF.
<b>vendorrelease *</b>	Vendor release
<b>vendormodelNumber*</b>	PNF Model value (link to A&AI)
<b>functionDescription*</b>	Describes the PNF function
<b>pnfExtConnPt (modelling def. of connection pt not a template)</b>	Specifies the characteristics of one or more connection points where to connect the PNF to a VL. Align ETSI SOL-001
<b>contactId (metadata)</b>	Designer (user of ONAP)
<b>pnfSWversionList</b>	The <b>EXPECTED</b> software to be supported by the PNF.
<b>PackageVersion</b>	The version of the PNF Package.

```

mdu_templates:
  - type: org.onap.sdc.resource.cp_extCP
  metadata:
    - resourceName: 9d8223e-33c0-476b-acc0-57af52b0e4
    - uri: cdf79ee-5da-4488-ba23-2912380a4f
    - customAnnotationId: 76a3d0e-30ef-937e-bd6e-11b0e009a1
    - version: 1.0
    - name: ExtCP
    - description: The ExtCP Connection Point base type all other CP derive from
    - type: CP
    - category: Generic
    - subcategory: Network Element
    - resourceVendor: 32T (Tosca)
    - resourceVendorRelease: 1.0.0,extCP
    - resourceVendorModelNumber:
  properties:
    - name: requires
      - mdu_required:
    - name: is_required: false
    - extCP_enabled:
    - mdu_generated_naming: true
  
```

## • Content of PNF software version List

Contents		Description
softwareList	<b>description</b>	Describes the main feature of the this software version
	<b>swVersion</b>	Software version

**\*Already supported in Beijing**

NAME

Service Level Package

Template, TOSCA level template representing some item  
On-boarded VNF want instance of that virtual firewall  
Place inside all the orchestrator needs

Name	Size	Packed Size	Modified	Created
artifacts.yml	3 098	1 169	2018-03-19 09:55	
capabilities.yml	8 306	1 748	2018-03-19 09:55	
data.yml	45 549	6 222	2018-03-19 09:55	
groups.yml	3 527	1 257	2018-03-19 09:55	
interfaces.yml	1 019	486	2018-03-19 09:55	
nodes.yml	130 477	11 667	2018-03-19 09:55	
policies.yml	3 631	861	2018-03-19 09:55	
relationships.yml	5 360	1 560	2018-03-19 09:55	
resource-Estneutroncp-template.yml	3 064	1 060	2018-03-19 09:55	
resource-Neutronnet-template.yml	2 223	846	2018-03-19 09:55	
resource-Neutronport-template.yml	3 145	1 074	2018-03-19 09:55	
resource-Vfw-template-interface.yml	246 202	7 771	2018-03-19 09:55	
resource-Vfw-template.yml	70 134	2 255	2018-03-19 09:55	
resource-VfwComputeNodesHeatVfw-template.yml	786	394	2018-03-19 09:55	
resource-VfwComputeNodesHeatVpg-template.yml	786	394	2018-03-19 09:55	
resource-VfwComputeNodesHeatVsn-template.yml	786	394	2018-03-19 09:55	
resource-VfwNodesVfwcvfc-template-interface.yml	31 443	3 862	2018-03-19 09:55	
resource-VfwNodesVpgcvfc-template-interface.yml	31 443	3 862	2018-03-19 09:55	
resource-VfwNodesVsnvfc-template-interface.yml	31 443	3 862	2018-03-19 09:55	
resource-VfwNodesVsnvfc-template.yml	24 733	1 763	2018-03-19 09:55	
resource-Testsv-template-interface.yml	25 508	3 553	2018-03-19 09:55	
resource-Testsv-template.yml	243 220	7 143	2018-03-19 09:55	
service-Testsv-template.yml	63 014	5 295	2018-03-19 09:55	

```

topology_template: 00
node_templates: 00
  vfw 0: 00
    type: org.openstack.resource.vf.vfw 00
    metadata: 00
      invariantUUID: fa9f72b3-d656-4e08-ae5e-9482ab165d5c 00
      UUID: 0a9e817b-4c98-455a-95d4-71e24713f73d 00
      customizationUUID: -E7432b08-ed4a-43fa-b179-40d2c5ff1b1e 00
      version: '0.2' 00
      name: vfw 00
      description: saad 00
      type: VF 00
      category: Network L4+ 00
      subcategory: Common Network Resources 00
      resourceVendor: example 00
      resourceVendorRelease: '1.0' 00
      resourceVendorModelNumber: '' 00
    properties: 00
      nf_naming: 00
      scomp_generated_naming: true 00
      multi_stage_design: false 00
      availability_zone_max_count: 1 00
      vfw_flavor_name: 4 GB General Purpose v1 00
      public_net_id: 00000000-0000-0000-0000-000000000000 00
      vfw_image_name: Ubuntu 14.04 LTS (Trusty Tahr) (PVHVM) 00
    capabilities: 00
      network.incoming.bytes.rate.vpg.vpg_private_1.port: 00
    properties: 00
      unit: B/s 00
      description: Average rate of incoming bytes 00
      type: Gauge 00
      category: network 00
      disk.read.bytes.vpg: 00

```

# PROJECT IMPACTS FROM ONBOARDING

ONAP Project	IMPACT
<b>Management Design</b>	(After modeling, and instantiation) “in the middle” Policy Design changes? (check w/ Policy Project) CDT (APPC) - Networking Design – SDN-C design tool for networking design. Creates Yang files. Config Design
<b>VID</b>	

# PNF REGISTRATION HANDLER IMPACTS

## PNF Registration Handler (PRH) Project Impacts




# ACTIVE INVENTORY (A&AI) IMPACTS

## ACTIVE & AVAILABLE INVENTORY (A&AI) PROJECT IMPACTS

**New A&AI PNF Parameters**

**PNF GEOLOCATION** - provides information about the geographical location (e.g. geographic coordinates or address of the building, etc.) of the PNF.  
Latitude/Longitude

**Software Version**

**DETECTED SOFTWARE PNF VERSION(S)** – during Run-Time when the PNF registers with ONAP it can report its (list) of PNF Software that is currently has installed. And this could be kept track of in the A&AI entry for that PNF. Entry will also have *Active/Passive*

### • Content of PNF software version List

Contents		Description
softwareList	description	Describes the main feature of the this software version
	swVersion	Software version

**PNF [#1:CU/#2:ONAP] CLOUD HOME (CLOUD SERVER LOCATION)** – PNF is served by some regional ONAP cloud servers. Serves in “Rehome” PNF. CLLI Code (specifies location, street address, CloudID, physical server is deployed). [Potentially a list of locations]

**Manager IP Address** – provides an additional IP address for the BTS that is vendor-specific and relevant to the OAM management of the BTS.



### PNF A&AI ENTRY (From Beijing)

Register PNF Service – may need new registration information in AAA

- PNF has a “*pnf-name*” = Key in AAI. Could change it to the ID. E.g. “Name” = “abcd””ID#””#Code” (automated, NF naming code); *equip-type* (PNF Type). *equip-vendor* (optional); *equip-model* (optional); *pnf-id* (PNF ID) (Step 33) adds *ipaddress-v4-oam*; *ipaddress-v6-oam* This is the “manager IP Address” which for a DU might be a CU IP address. ; (FYI/ *ipaddress-v4-loopback-0*).

*mac-address* & *serial-number*,

*PNF:: proxy IP address*



# DCAE IMPACTS

## DCAE PROJECT IMPACTS

**Security  
Validation**

Authenticate Connection

**MONITORING**

Monitoring definitions – SDC has a side monitoring template designer. Way to define monitoring alarms etc; In AT&T there is a project; IN ONAP code is there finalizing code; pluggable modeler for monitoring. DCAE as part of onboarding specify what VES template. DCAD define microservices for monitoring. How is PNF monitored & correlated. If [x] goes down how is this correlated. SDC would define the Modeling what needs to be monitored and how they would correlated with other events from other NE. Thresholds. [Baby step to process get an alarm from PNF, YAML file describes fault VES event, Fault meta-data; alarms generate].

# SERVICE ORCHESTRATOR (SO) IMPACTS

## SERVICE ORCHESTRATOR (SO) PROJECT IMPACTS

**WORK FLOW  
(SO  
executor)**

Create work-flow for PNF (Srini)  
For configuration for SO



# SDN-R (Controller) IMPACTS [APP-C]

## SDN-R APP-C PROJECT IMPACTS



CC-SDK (SDN-R)

Artifacts from CC-SDK, APP-C

Mobility controller

Artifacts “borrowed” will become part of CC-SDK

Available for all controller persona.

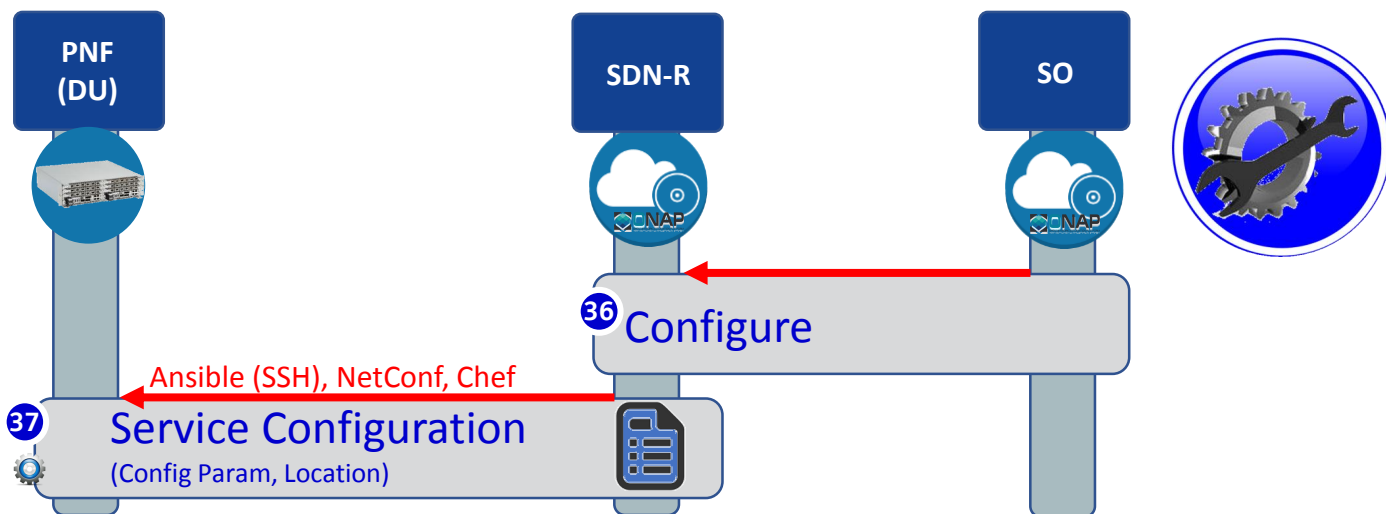
QUESTION: moving towards CC-SDK & common foundation is there anything

On top of that planned for SDN-R.

Some thing specific to wireless, ... perhaps they do not belong in CC-SDK.

Things borrowed from CC-SDK and libraries that are common.

# SDN-R w/ PnP in R3



SDN-R is ONAP controller derived from CC-SDK that combines functionality of APP-C & SDN-C and includes wireless artifacts.

SDN-R FUNCTIONS	DESCRIPTION
PROTOCOLS SUPPORTED	Ansible, Chef, and NetConf (all 3 will be supported)
PNF SUPPORTED	(Radio RAN Wireless) PNF Other PNFs?
PNF DATA SUPPORTED	<p>(1) <b>Configuration Parameter (optional)</b> – SDN-R/APP-C gives the Controller IP@ to the DU. In R3, SDN-R/APP-C may pass configuration parameter(s) to the 5G DU, this will also give a configuration parameter (e.g. CU IP@).</p> <p>(2) <b>OAM IP@ (optional)</b> - The permanent OAM IP address is given to the PNF. The IP address assigned from <b>SDN-C</b> may come from the vAAA, or it may draw from a local pool of IP addresses. <b>SDN-C</b> performs the IP address selection. It knows if a permanent IP address should be assigned to the PNF. Note: this IP@ assignment <i>optional</i>.</p> <p>(3) <b>Transport configuration (optional)</b> – Transport configuration is given to the PNF.</p> <p>(4) <b>Location (optional)</b> – the Location configuration may be given to the PNF.</p> <p>(5) <b>Software Version (optional)</b> – The expected PNF Software Version(s).</p>

# SDN-R w/ PnP in R3

SDN-R FUNCTIONS	DESCRIPTION
<b>PROTOCOL SPECIFIED</b>	<p>PNF onboarding package defines of PNF (protocol). PNF info model (SDC) will have (1) protocol (2) controller:</p> <ol style="list-style-type: none"><li><b>(1) <u>Protocols Supported</u></b> – PNF package. CM Protocol is in PNF onboarding package.</li><li><b>(2) <u>Controller type</u></b> – defined in design time. Controller type is specified in design time. Design time designer specifies which controller is used.</li></ol> <p>Distributed in PNF Modeling artifacts</p>
<b>MODEL DRIVEN DESIGN</b>	<p>APP-C, SDN-C, SDN-R, VF-C as possible controllers. Modeling which controllers will be used by the PNF. Also needs to standardize what it expects from controller. Things controller needs to support.</p>
<b>VENDOR SPECIFIC ADAPTOR</b>	<p>Customized DG (directed graph) element ODL (node in graph database) added with Java. Substitute a value and add fields in a template. SDN-R is following DG abstraction. If there is a vendor specific plug-in in the service model it can be represented as a node in DG and the interaction will be captured in that model.</p>
<b>UPDATING PNF &amp; CONTROLLER RELATION INFO</b>	<p>Updating the controller/protocol (“lifecycle management”).</p>
<b>ERROR LEGS (ERROR FLOWS)</b>	<p>Errors when SDN-R interacts w/ a PNF</p>

# SDN-R w/ PnP in R3

SDN-R is ONAP controller derived from CC-SDK that combines functionality of APP-C & SDN-C and includes wireless artifacts.

SDN-R is a sub-project of SDN-C.

*“Is SDN-R is a sub-project of SDN-C. If not what is it?”*

SDN-R = APP-C for RAN. “ how is SDN-R common to APP-C “

ISSUE:

Concern that SDN-R replicates functions (e.g. SDN-C does IP assignments).

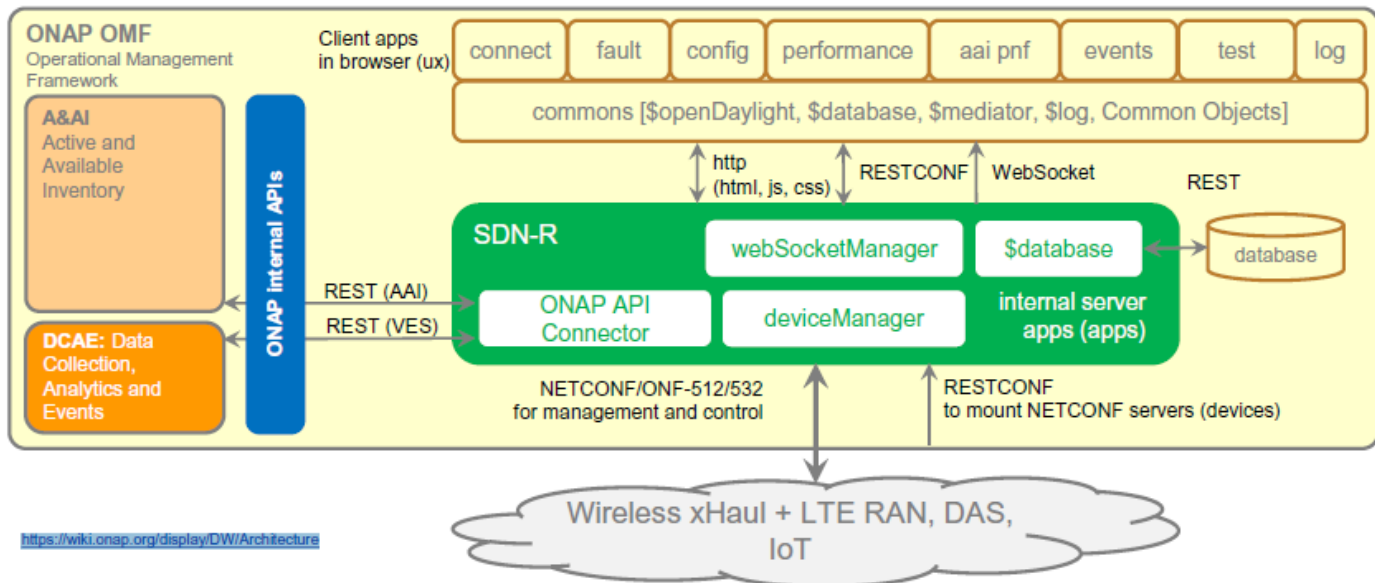
SDN-C configures/creates Networks.

SDC would need to be aware of another SDN-controller, new logic to pass DGs.

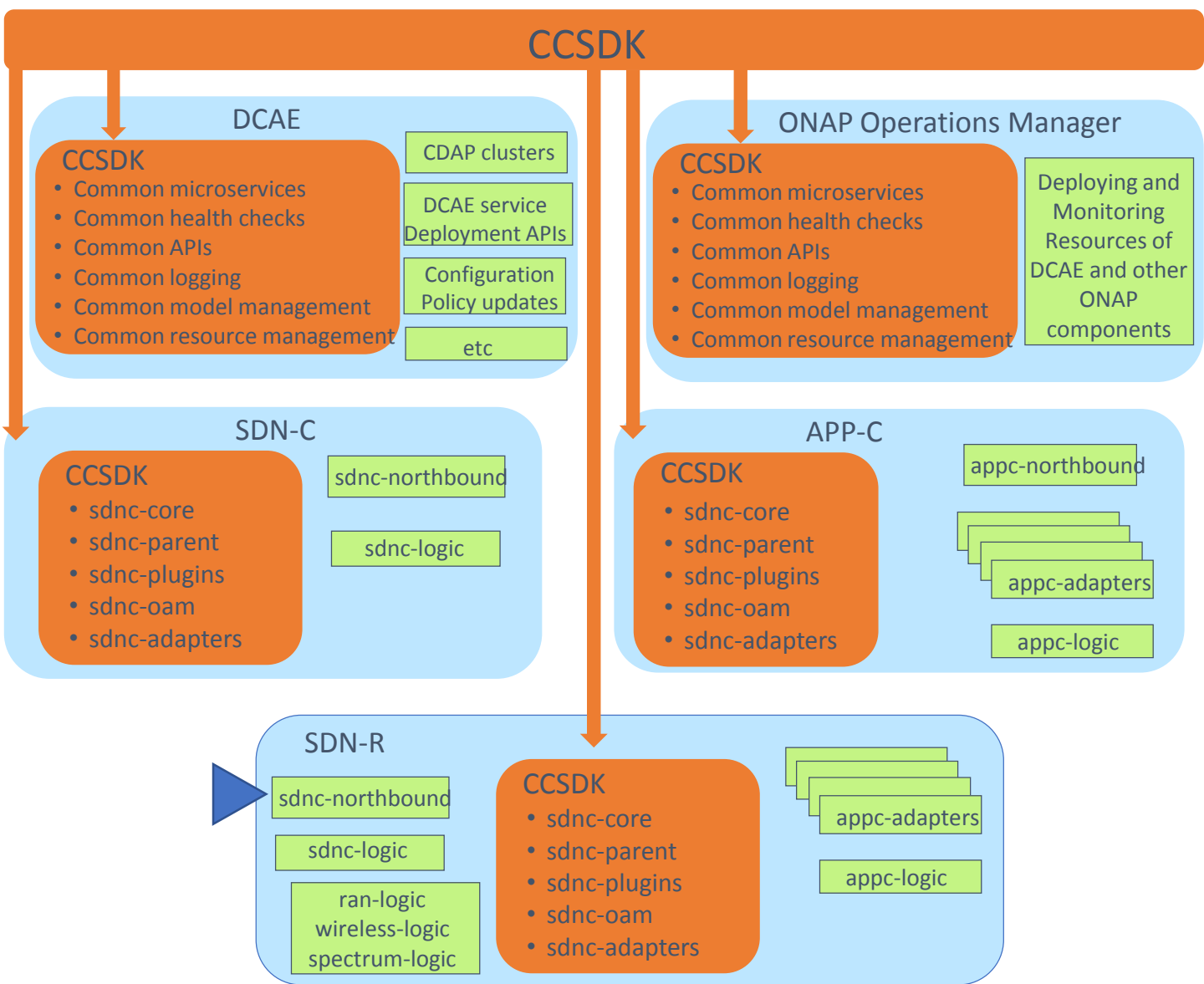
ISSUE:

“SDN” implies a networking controller vs an PNF “App” controller

SDN-R app controller for PNF/VNFs vis a vis LCM.



# CC-SDK based Mobility Network Function Controller (MNFC) for Mobility/Wireless



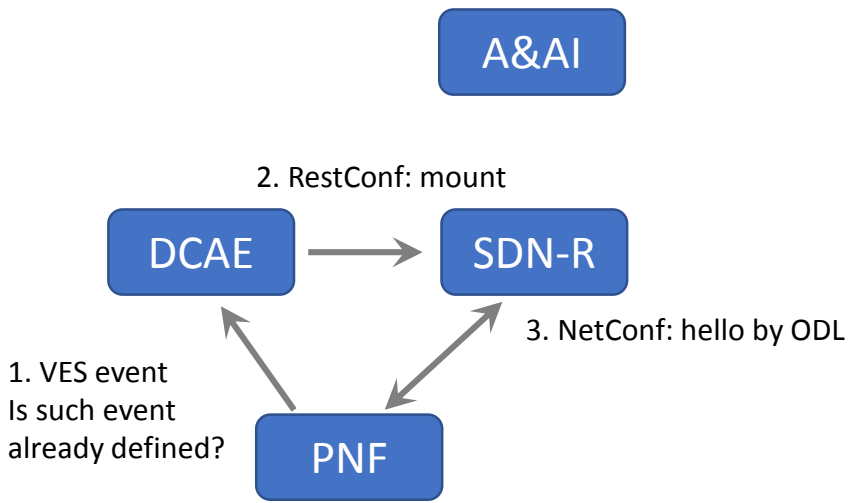
## SDN-R

- Evolving to become the one ONAP Controller for Mobility/Wireless PNF/VNF
- Sub-project in SDN-C

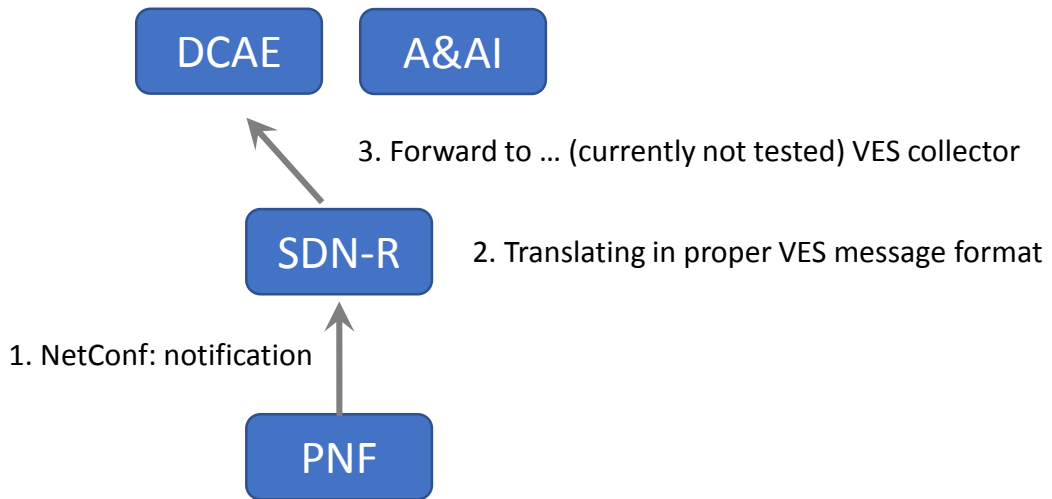
SDN-R intended to be a stand-alone controller  
Only organizationally is SDN-R a sub-project of SDN-C

SDN-R uses SDN-C Northbound (verify - ?)

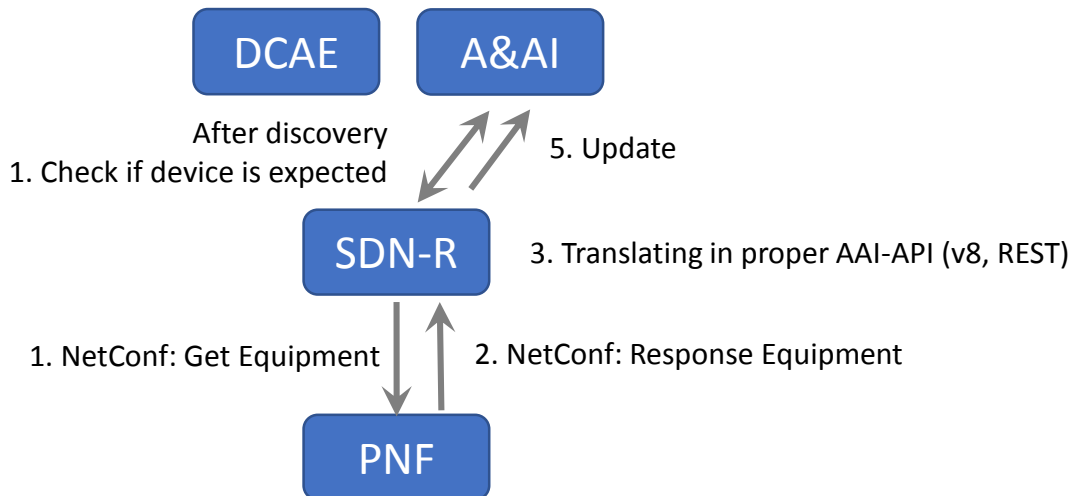
SDN-R Device Discovery



SDN-R Event Flow



SDN-R A&AI Flow







# PNF Plug and Play INTEGRATION & SHOWCASING

- ONAP and PNF Plug and Play for 5G RAN
- 5G Use Case Team

# INTEGRATION TEAM IMPACTS

## INTEGRATION TEAM IMPACTS

### SHOWCASING

**DEMO CONTENT** – [Closed] What will be **showcased**? Test send registration event. Instantiation Pt1/Pt2 (Update A&AI). PRH service request (VES event & DMaaP topic). APP-C Ansible configure. PNF sends event, receives ansible event. Start with R1 use case (VoLTE / vCPE) upgraded w/ mechanisms of R2. DU Emulator (for registration).

### ACTUAL PNF, DU Emulator

**DU EMULATOR** – [Closed] Nokia will provide a simple DU emulator [Mar 16 2018]. Capable of sending PNF registration request (Step 26, 28). Capable of receive configuration command from APP-C via Ansible (Step 37) and recording result of successful reception (or failure) in a log.

### Test Case Description (Demo Document) [Open Point]

(1) **Demo Document** – Document describing demo (use case group?) <https://wiki.onap.org/display/DW/vCPE+Use+Case+Tutorial%3A+Design+and+Deploy+based+on+ONAP+Amsterdam+Release>  
 (2) **Test case description** – Documenting the test case. Start with VoltE and modify it.

### LOGISTICS (Open Point)

(1) **DEMO ANNOUNCEMENT** – “come see this demo” – Integration ONAP Wiki. Video. Schedule ONAP Wiki Events Calendar. Integration project. (SME = x).  
 (2) **LOGISTICS** – *Contacts* Helen Chen, Yang Xu. Intel Windriver Lab, T-Lab (AT&T Lab, Bedminster), and/or use CMCC Lab. ONAP “Open Lab”. ONAP installed in a lab. Demo shown via conference. Virtual Demo (dial in).  
 (3) **RELEASE MANAGER** – Coordinate w/ Joe Gildas. Joe Gildas release manager (status of project). M4 Voting.

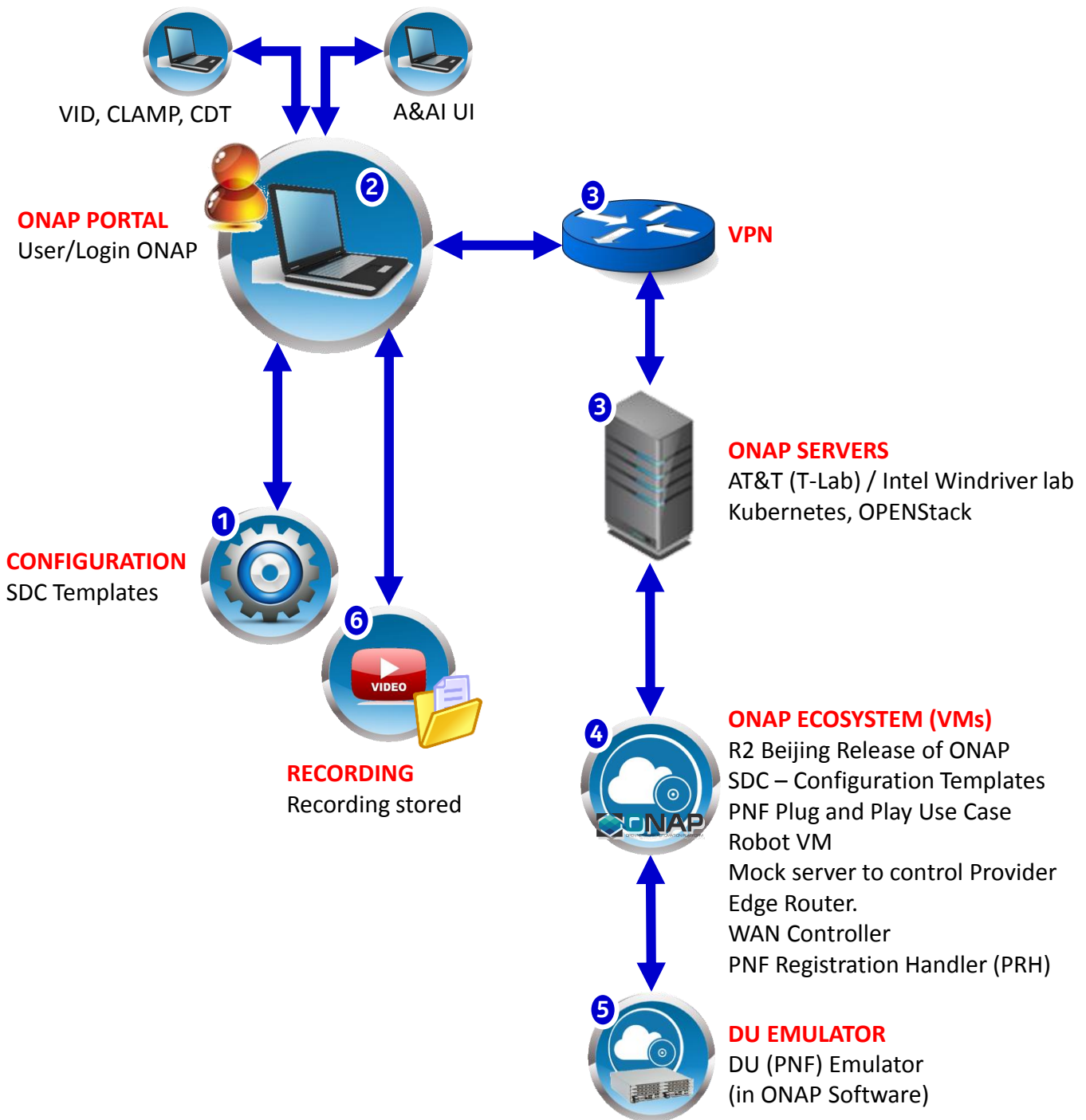
### AUTHENTICAT ION

**AUTHENTICATION** – Authentication is optional, and for R2 Demo we will not have any authentication. Could be considered for Casablanca (R3). Amsterdam authentication flow not useful for PNFs. Not use second vDHCP & vAAA.

### CONTROLLER CONFIGURATI ON

**CONFIGURATION**: IP address (from DHCP). Username & Password. Configured in controller (manually added to ONAP). VES configuration (do not need username & password to be configured on VES)

# DEMO ARCHITECTURE



STEP	DESCRIPTION
1	<p><b>CONFIGURATION</b> – SDC Design templates need to be configured in preparation of the Demo &amp; Showcasing. The appropriate PNF and VNF designs are designed, validated and exported to the other components of ONAP.</p>
2	<p><b>PORTAL &amp; GUI</b> – The ONAP Portal is used to access the appropriate GUIs and interfaces necessary to operate and configure ONAP. The VID and CLMAP GUI are used to configure the appropriate information for the showcase. The A&amp;AI GUI are used to show the created entries in A&amp;AI for the PNF.</p>
3	<p><b>VPN</b> – The appropriate security gateways and/or VPN networks are configured. The appropriate security setups are authenticated against. Passwords and IP addresses that are needed to connect to the ONAP servers go through this VPN or security gateway (as appropriate)</p> <p><b>ONAP SERVERS</b> – The ONAP Servers that host the ONAP ecosystem are connected to</p>
4	<p><b>ONAP ECOSYSTEM (VM)</b> – The ONAP ecosystem has all of the appropriate Virtual Machines necessary to operate. This has the R2 Beijing Release of ONAP. All of the PNF Plug and Play Use Case updated software is available in the ONAP Ecosystem. The Robot VM, Mock server to control Provider Edge Router, and WAN Controller are established.</p> <p>The PNF Registration Handler a new DCAE plug-in are available in this ONAP ecosystem which is used to register the PNF into ONAP.</p>
5	<p><b>DU EMULATOR</b> – The DU Emulator is used to emulate the responses and connection to ONAP. The DU emulator establishes a VES HTTPS connection with the PRH during the registration process and receives a APP-C response back via ansible.</p>
6	<p><b>RECORDING</b> – A recording is made of the showcase and demo for later upload/download and playback for demonstration purposes.</p>



# PNF Plug and Play ROADMAP (After Casablanca)

- ONAP and PNF Plug and Play for 5G RAN
- 5G Use Case Team

# FUTURE FUNCTIONS

ONAP Project	IMPACT
<b>Licensing Management</b>	Licensing Impact – meta-data managing licensing. PNF (DU#55) – Service on DU (5G) [feature xyz] Licensing Management (FUTURE)
<b>(VNF 5G Flow)</b>	[in Casablanca] a use case for VNF & PNF bringup.



# APPENDIX & Meeting Notes



DISCUSSION MAY 03, 2018

SDN-R supports all 3

CC-SDK controller

Should support all 3

How does that get implemented

PNF agnostic/future-proof

PNFs will support one

ADAPTORS/MEDIATORS

Adaptor – No adaptor

PNF talks directly to ONAP

Adap/transl/mediators

For embedded base.

5G “going native”

PNF – Descriptor

Protocols supports

PNF Package defines

The Meta-models Tbsupported

mm microwave PNF unchanged



ID PNF

PNF #62

SDN-C  
UUID

PNF #62

A&AI

Mac Address  
IP Address

Correlation ID  
(UUID)  
PNF #62

Gets Assigned a UUID

Just like any VNFID

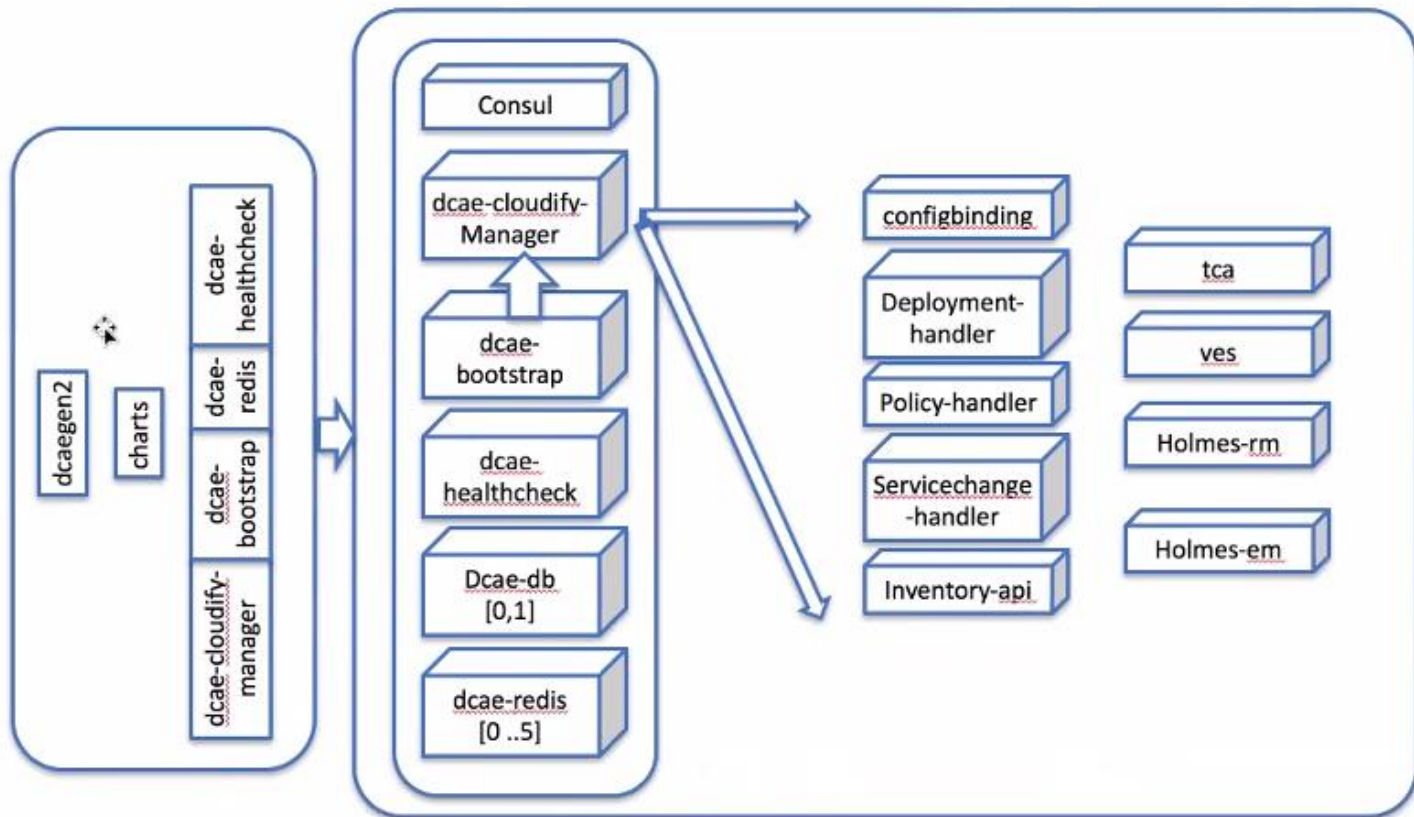
From then on processes of ONAP

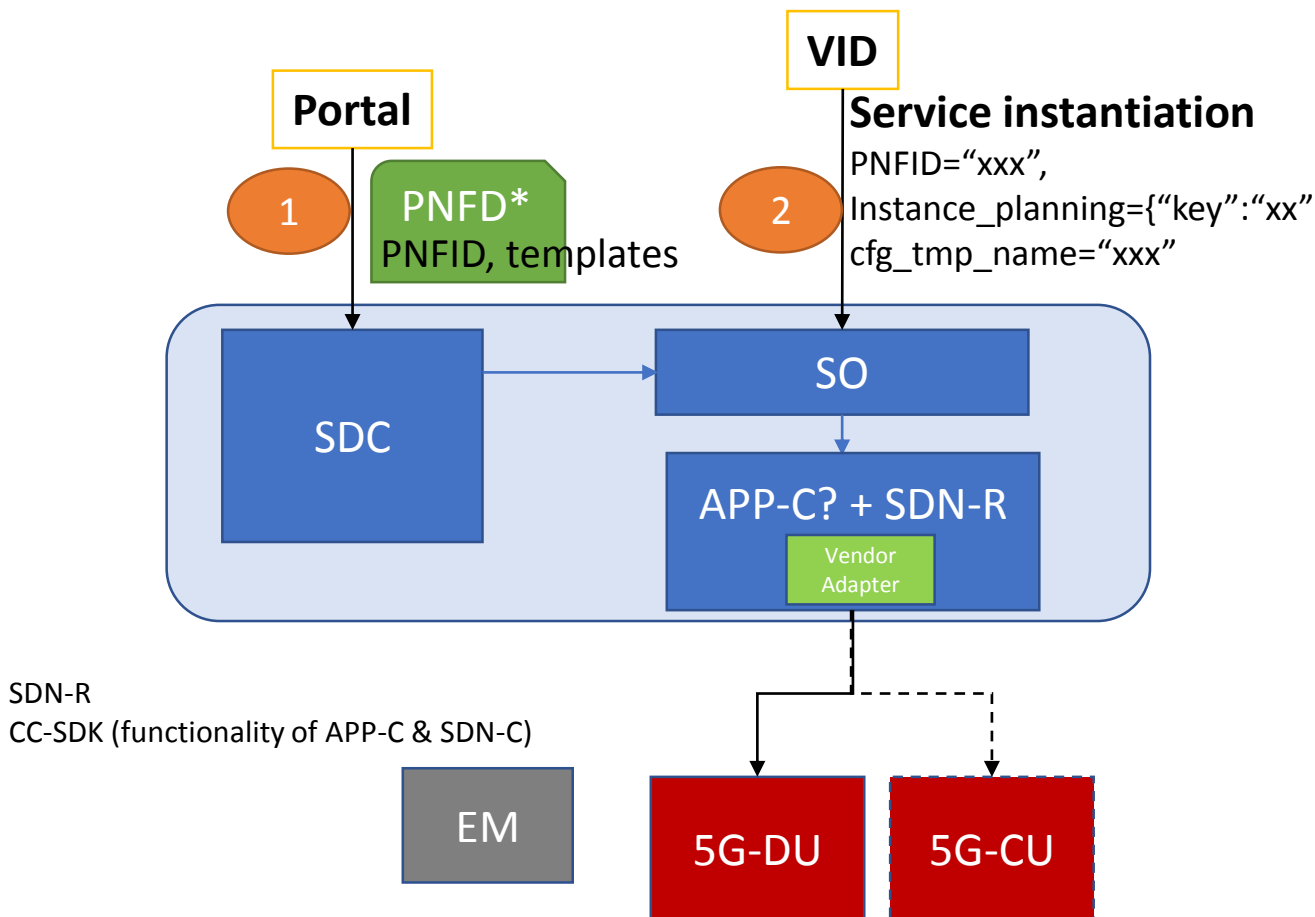
Work on UUID

CorrelationID is no longer used.

UUID assigned by SDN-C (rules how UUIDs assigned)

# DCAE Kubernetes Deployment



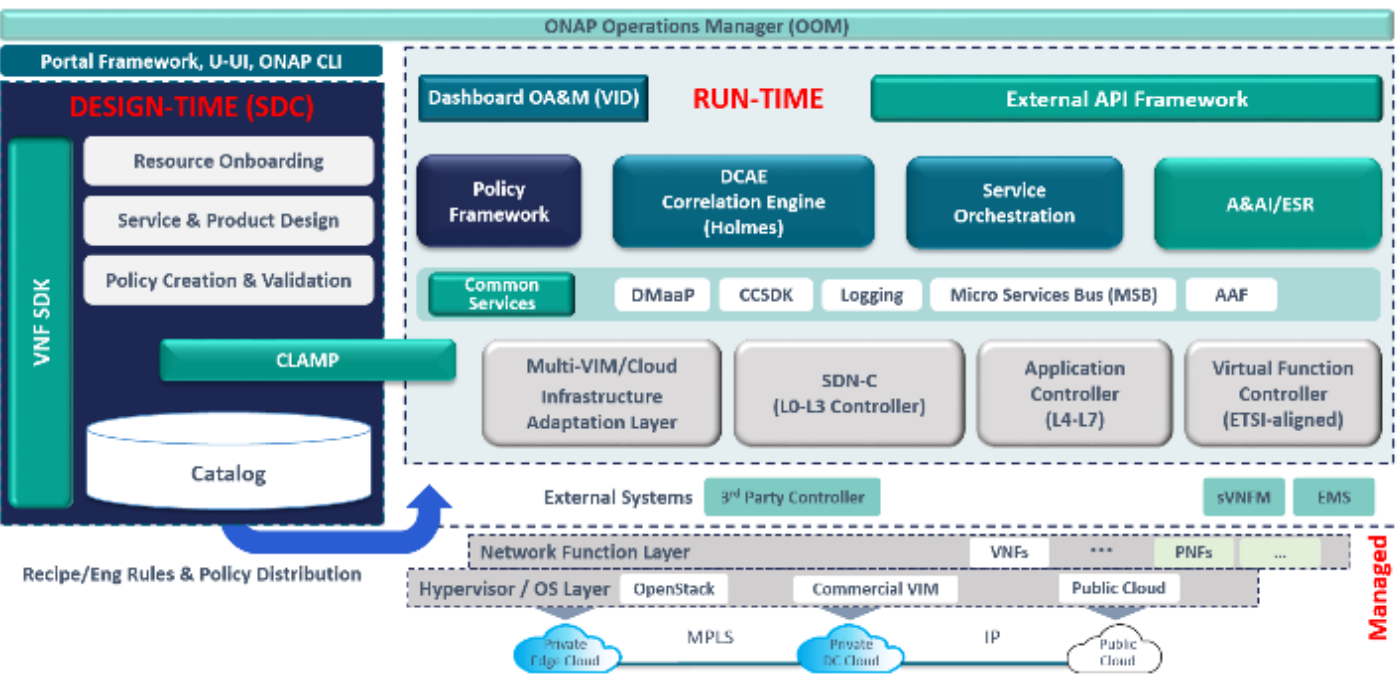
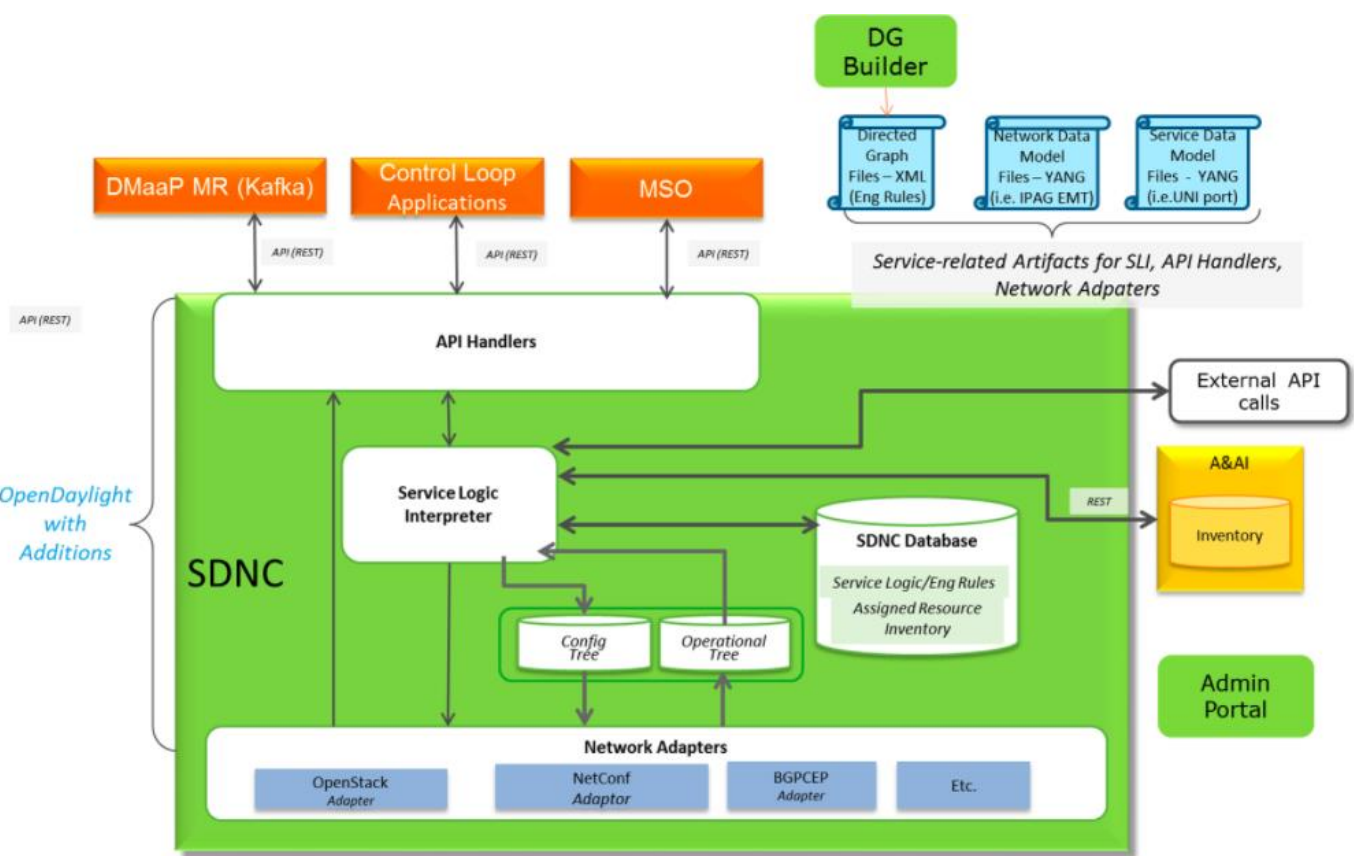


## Design time

- (1) Onboarding PNFD
  - PNFID
  - Templates, including PNF/VNF templates,
  - Configuration templates

## Run time

- (2) service instantiation
- (3) generate configuration data
- (4) configure PNF/VNF via EM
- (5) CU-DU inter-connect



Design onboarding/ Policy/API...	Closed Loop Design and performance	Instantiation portal	Automation rules execution	Data Collection Analytics and Events	Real time Inventory	Camunda- based Orchestration	VIM adaptation	SDN Controller	Application Controller (VNFs/PNFs)
<b>SDC</b>	<b>CLAMP</b>	<b>VID</b>	<b>Policy</b>	<b>DCAE</b>	<b>A&amp;AI</b>	<b>SO</b>	<b>Multi VIM</b>	<b>SDN-C</b>	<b>APP-C</b>











