

# ONAP NF Modeling in SDC for Casablanca (R3, 4Q 2018)

- NF (VNF/PNF) Modeling for R3
- SDC Project

# Modeling ONAP Links

PAGE	LINK
<b>Use case proposal: 5G- RAN deployment, Slicing, SON</b>	<a href="https://wiki.onap.org/display/DW/Use+case+proposal%3A+5G-+RAN+deployment%2C+Slicing%2C+SON">https://wiki.onap.org/display/DW/Use+case+proposal%3A+5G-+RAN+deployment%2C+Slicing%2C+SON</a>
<b>5G Functional Requirements Tracking</b>	<a href="https://wiki.onap.org/display/DW/5G+Functional+Requirements+Tracking">https://wiki.onap.org/display/DW/5G+Functional+Requirements+Tracking</a>
<b>Casablanca Release Requirements</b>	<a href="https://wiki.onap.org/display/DW/Casablanca+Release+Requirements">https://wiki.onap.org/display/DW/Casablanca+Release+Requirements</a>
<b>NFModeling- SDC_R324Jul2018v1</b>	<a href="https://wiki.onap.org/display/DW/Casablanca">https://wiki.onap.org/display/DW/Casablanca</a>
<b>Service Design &amp; Creation (SDC) Portal Page</b>	<a href="https://wiki.onap.org/display/DW/Service+Design+and+Creation+%28SDC%29+Portal">https://wiki.onap.org/display/DW/Service+Design+and+Creation+%28SDC%29+Portal</a>

# 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. Updated NF Model to include software\_versions. Integration of software\_versions
- (2) PNF SHARING** – SDC model updates for PNF characteristics focusing on PNF inter-connectivity. (Dublin)
- (3) PNF-SDK** – SDK provided from Vendors. This will help modeling the Physical “Box” (PNF) and network functions. (Dublin)
- (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. (Dublin)



## 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 PNF Package framework.

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



PROJECTS:  
SDC, APP-C



# TERMS, CONCEPTS & PRINCIPLES

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

# MODELING PRINCIPLES



## Planner Personnel

Optimization  
Network Planning



## Provider Personnel

Operators  
Technicians



## Vendor Personnel

Technician  
Developers  
Product Support

## SERVICE MODEL

Services

Application Data

Operational Operator

Functional Aspects

Run-Time

ONAP Service vs ETSI/OPENO/3GPP Service

Orchestrating ONAP component & resources

NF Interconnectivity, chaining, relations

## RESOURCE MODEL

Physical Resources

Application Data

Operational Operator

Physical Aspects

Run-Time

Data/Information model

VNF and PNF resources

## (ONAP) PLATFORM MODEL

NF Interconnectivity

ONAP Platform-level information

Design-time Operator

Templates

Meta-data

SDC Design Studio & Catalog

ONAP Components (SO, A&AI, APPC/SDNC etc)

Inventory, Events



## ONAP Personnel

"Roles" (Operator  
Governor  
Designer  
Administrator  
Tester)

# INFORMATION HANDLING PRINCIPLES

## DESIGN TIME INFORMATION

- Design-time Operator
- NF Interconnectivity
- ONAP platform & component information
- Templates
- Meta-data, TOSCA semantics (policies, cap/req)
- Design Time Model
- Control Loop / Closed loop model

## RUN-TIME INFORMATION / APPLICATION DATA

- Run-Time Attributes, Run-Time Model
- Accessing NF instance for ONAP components
- Instance of a NF
- Application Data
- Operational Operator
- Functional Aspects
- Orchestrating ONAP component & resources
- States & modes

## CONFIGURATION INFORMATION

- Service Parameters
- Run-Time parameters
- Functional Configuration
- Data Consistency & Data Validation

Aug 21, 2018

(NK Shankar)  
PCI Discussion

If ONAP needs to know about PCI to  
Perform control loop functions, ONAP needs to know  
As part of a process flow.

(Arash Hekmat)

5G Architecture

Configuration kept in SDNC/SDNR

Source of 5G configuration information.

Control designer studio feature to **CDT** in Casa/R3

Adding XML or JSON template for configuration to CDT

CDT would move to SDC Design Studio (additional cap)

Consume model/create artifacts.

When you have config you config a port IP@,

Resource Resolution.

Dictionary of resources.

Parameters are tagged (\$, #) resolve @ Run time.

As resource resolution **SDNC** > templates > data dictionary

Resolves parameters to configure NF >

Protocol to write config (Ansible, Chef, Netconf)

UI Data dictionary/parameters to resolve at run-time



# VERSION CONCEPT

## Diagram of Software Version Management for a PNF



### PNF-Descriptor (version)

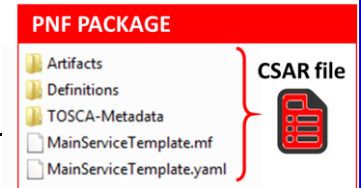
*Vendor Provided*

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** – [Partition1 “Active”] 12345

[Part 2] 67890 [Recovery Partition] 00010

**ONAP EXPECTED PNF S/W** 3.0 (modeling)



### PNF Hardware (version)

Hardware Version

Firmware Version

Product Model version

Vendor Release – VID match PNF avail in system

S/W version management – Use casa

Troubleshoot

Network Analysis

Correlation Version

Error Checking

Modeling informational

Network Planning

OPENStack – Image Repository in Glance. In VNF service designer request different version of S/W than is one in PNF itself

# ASSETS MANAGED (WIKI)

**Resource:** a fundamental capability, implemented either entirely in software, or as software that interacts with a hardware device. Each Resource is a combination of one or more Virtual Function Components (VFCs), along with all the information necessary to instantiate, update, delete, and manage the Resource. A Resource also includes license-related information. There are three kinds of Resource:

- Infrastructure (the Cloud resources, e.g., Compute, Storage)
- Network (network connectivity functions & elements);  
example: a Virtual Network Function (VNF)
- Application (features and capabilities of a software application); example: a load-balancing function

**Service:** a well formed object comprising one or more Resources. Service Designers create Services from Resources, and include all of the information about the Service needed to instantiate, update, delete, and manage the Service

**Product:** includes one or more Services packaged with commercialization attributes for customer ordering, billing, and issue resolution. Products are created by Product Managers, and can have one or more "category" attributes assigned by Product Strategists.

**Offer:** bundling of Products with specific Marketing configurations for selling to customers

# SDC COMPONENTS (WIKI)

There are four major components of SDC:

The **Catalog** is the repository for assets at the Resource, Service and Product levels. Assets are added to the Catalog using the Design Studio.

The **Design Studio** is used to create, modify, and add Resource, Service, and Product definitions in the Catalog.

The **Certification Studio**, available in a future release, is used to test new assets at all levels. It will be used for sandbox experimentation, and will include support for automated testing.

The **Distribution Studio** is used to deploy certified assets. From the Distribution studio, new Product assets, including their underlying Resources and Services, are deployed into lab environments for testing purposes, and into production after certification is complete. In a future release, there will be a way to export Product information to external Business Support Systems for customer ordering and billing.

# SDC COMPONENTS (WIKI)

## 1. Resource Model

(The one, which is defined in SDC, and defines those relevant resource parameters, which characterize services running on top of that resources, or allows these resources to bring relations to other resources in a service definition) Possibly, this is as well an interaction model – how different VNFs interact with each other, what relations are they building?

## 2. Inventory Model

(The one, which defines, which configuration/instance parameters are stored per resource/service instance – e.g. concrete IPs or Serial Numbers, that are assigned to concrete instances)

## 3. Configuration Model

(The one, which defines, which configuration parameters are required/exposed as application parameters to e.g. controllers)

## 4. Event “model”

(How the events, that we`re generating look like – what are their structures/elements/ what is the meta-data that is used around them?)

A side-effect of this one is “interaction model” – so which actions are we taking, when we discover, that there is something wrong with the xNF based on this model contents.

# VNF vs PNF Comparison

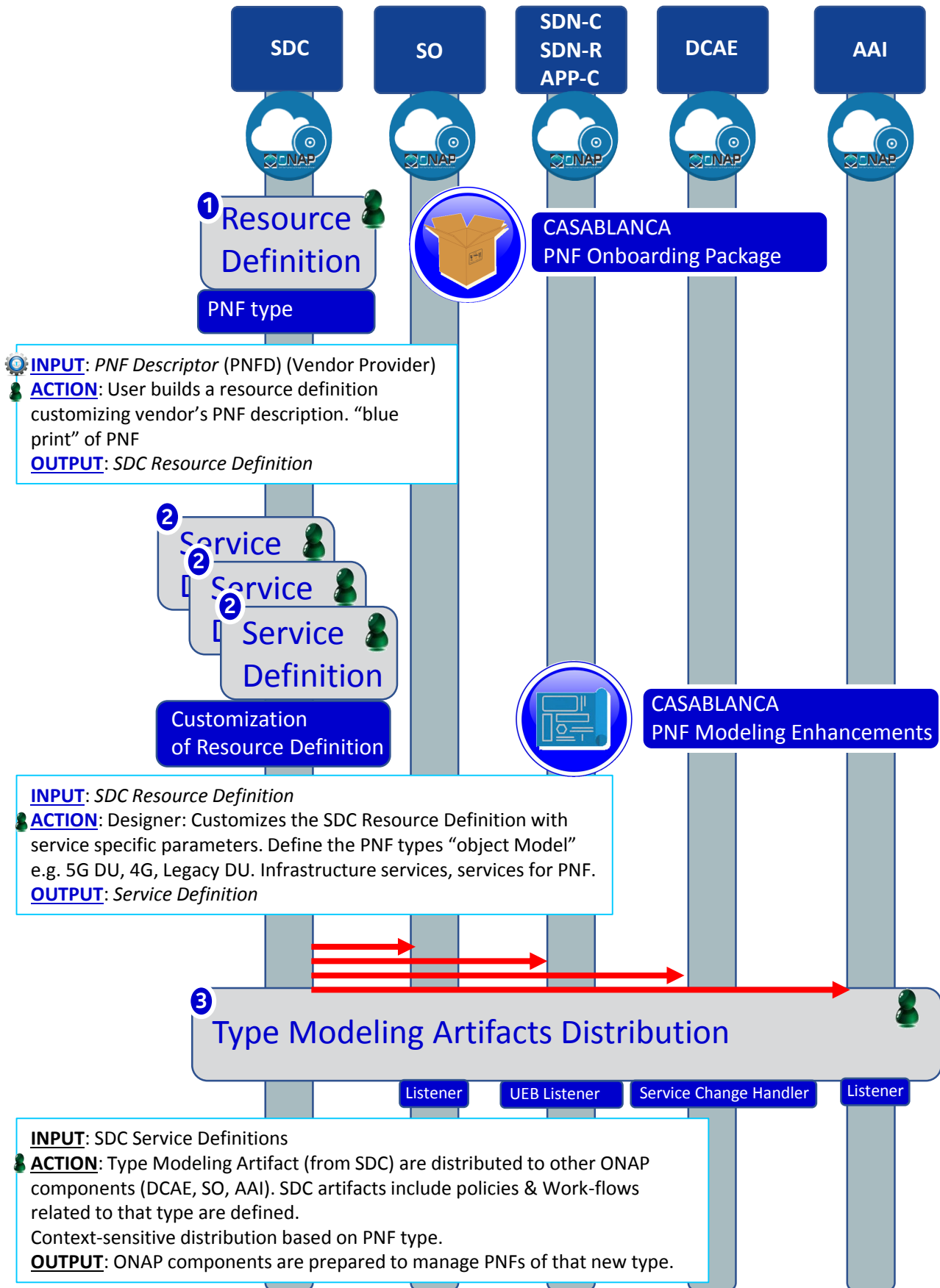
TOPIC	VNF	PNF
<b>Concept</b>	Application fulfills the role of a network function.	It is a network element, a physical entity, which can implement the role of a network function.
<b>Physical Characteristic</b>	Application without dedicated hardware; Virtualized applications require specific capabilities; 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)



# NF Modeling in SDC IN CASABLANCA (R3)

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

# Design Time (ONAP)



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>



# 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. Updated NF Model to include software\_versions
- (2) PNF SHARING** – SDC model updates for PNF characteristics focusing on PNF inter-connectivity. (Dublin)
- (3) PNF-SDK** – SDK provided from Vendors. This will help modeling the Physical “Box” (PNF) and network functions. (Dublin)
- (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. (Dublin)



## 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. (Dublin)
- 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



# NF SDC & Modeling Project Impacts Overview

- ONAP and SDC NF Modeling for 5G RAN
- 5G SDC Project

# 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. vCPE, PnP (Sub) W/F are supported in R3. 5G RAN W/F R4.
<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. (No open point for Casablanca R3)

# 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.

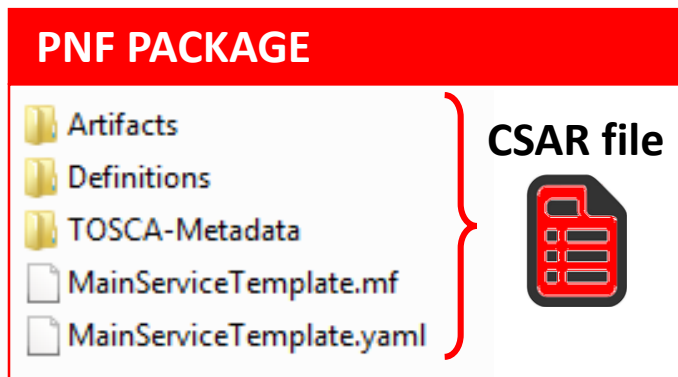
 Suggested VES Event Entry - Fault' Domain Datatypes  
For Alarm Dictionary Index (in **Dublin**)

Alarm Index	Dictionary	number	No	Alarm Dictionary Index, (since optional if left blank would mean dictionary is not used)
----------------	------------	--------	----	--

# 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.*

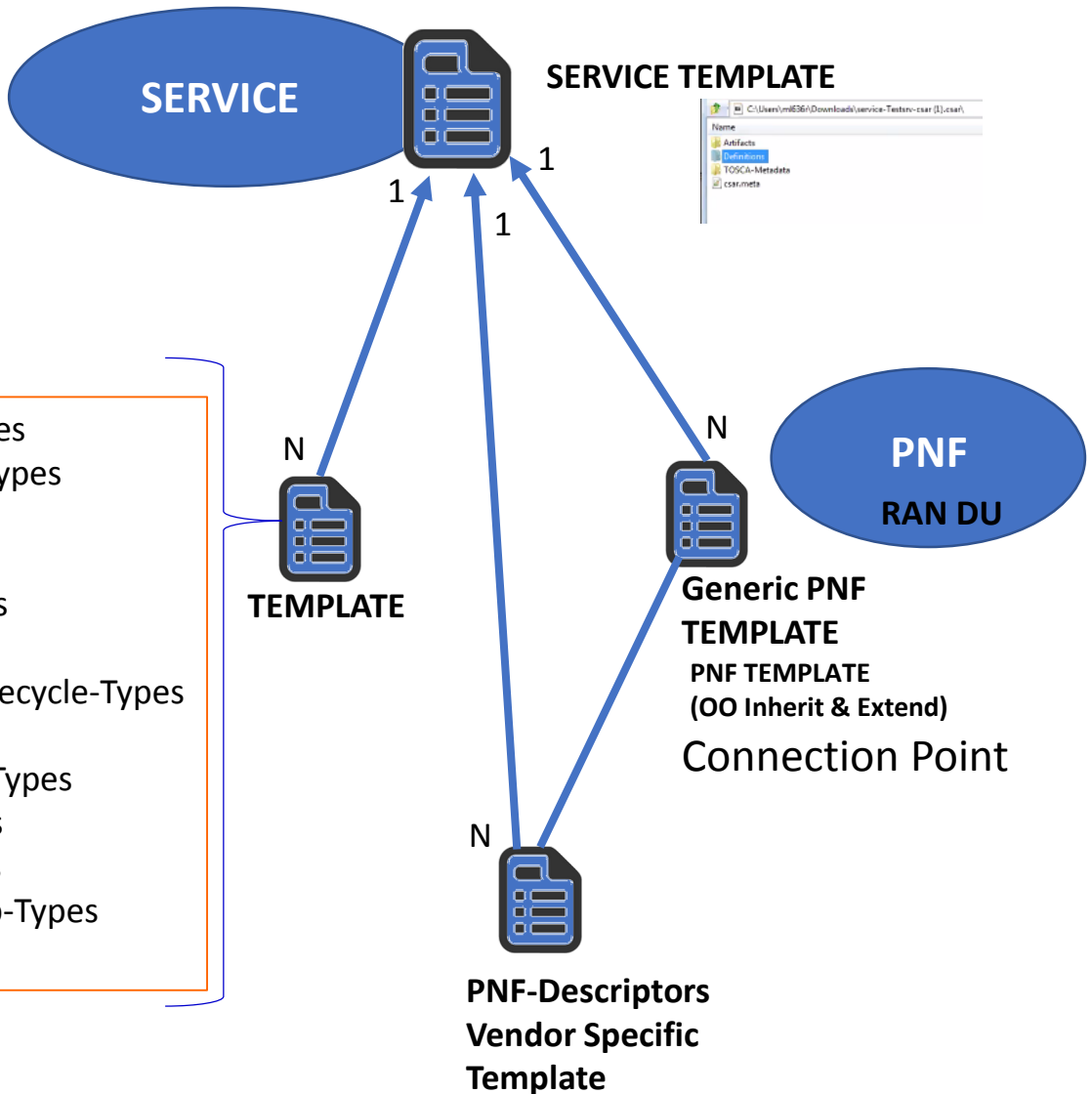
PNF Policies -

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

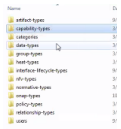
Software Version

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

# SDC MODELING (Design Time) – Casa R3



- Artifact-Types
- Capability-Types
- Categories
- Data-Types
- Group-Types
- Heat-Types
- Interface-Lifecycle-Types
- NFV-Types
- Normative-Types
- ONAP-Types
- Policy-Types
- Relationship-Types
- Users



```

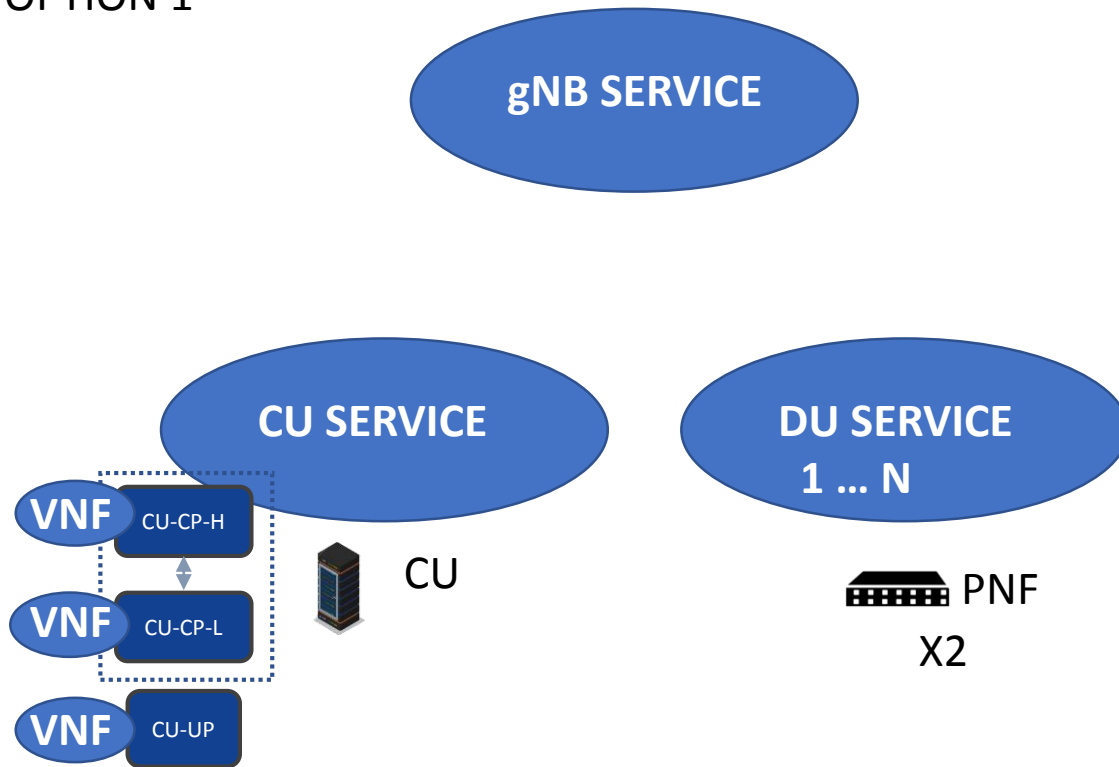
sdccatalog-be> src> main> resources> import> tosca> heat-types> Generic_PNF
Open New folder
Name Date modified Type Size
Generic_PNF 9/10/2017 12:49 JSON File
Generic_PNF 9/10/2017 12:49 YAML File
Generic_PNF 9/10/2017 12:49 Compressed (zip...)
tosca-definitions-version: tosca-simple-yaml-1.0.0
nodes:
  - org.openecomp.resource.abstract.nodes.PNF:
    derived from: tosca.nodes.Root
    properties:
      nf_function:
        type: string
      nf_role:
        type: string
      nf_type:
        type: string
  
```

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

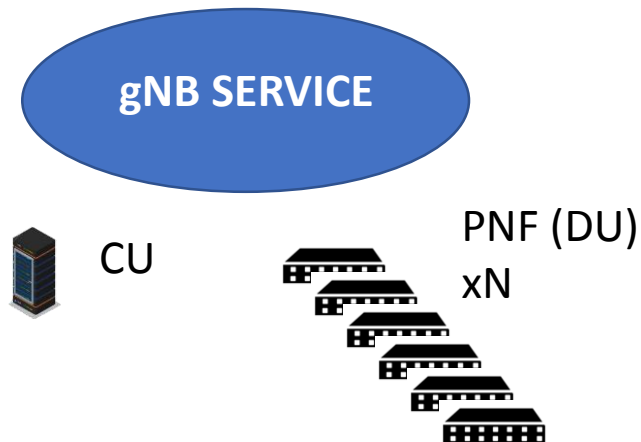


# gNB Model (R4)

## OPTION 1



## OPTION 2





# NFD DESCRIPTOR (PNFD, VNFD)

- ONAP and SDC NF Modeling for 5G RAN
- 5G SDC Project

# PNF Descriptor

## 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.

*From Potential PNF template for PNF S/W management & change management (Lixiang,YaoguangWang, ChangMing Bai Hwawei)*

## 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.

# VNF Information Model

Red items are proposed to be deleted.

gold indicates an add or a change

«Experimental»



Vnf

- «Experimental» + vnfInstanceId: Identifier [1]
- «Experimental» + vnfInstanceName: String [1..\*]
- «Faulty» + vnfProductName: String [0..1]
- «Faulty» + description: <Undefined> [0..1]
- «Faulty» + vnfProvider: <Undefined> [1]
- «Faulty» + vnfId: Identifier [1]
- «Faulty» + vnfVersion: String [1]
- «Faulty» + vnfSoftwareVersion: String [1]
- «Experimental» + onboardedVnfPkgInfoId: Identifier [1]
- «Faulty» + availabilityZone: <Undefined> [1]
- «Experimental» + operationalStatus: OperationalStatus [0..1]
- «Experimental» + orchestrationStatus: OrchestrationStatus [1]
- «Faulty» + oamIpv4Address: <Undefined> [0..1]
- «Faulty» + oamIpv6Address: <Undefined> [0..1]
- «Faulty» + instantiatedVnfInfo: <Undefined> [0..1]
- «Faulty» + inMaint: Boolean [0..1]
- «Faulty» + isClosedLoopDisabled: Boolean [0..1]
- «Faulty» + encryptedAccessFlag: Boolean [0..1]
- «Faulty» + vnfConfigurableProperty: <Undefined> [0..1]
- «Experimental» + nfNamingCode: String [1]
- «Experimental» + vnfNamingPolicyId: String [1]
- «Experimental» + vnfHomingPolicyId: String [1]
- «Experimental» + nfType: String [1]
- «Experimental» + nfFunction: String [1]
- «Experimental» + nfRole: String [1]
- «Experimental» + closedLoopStatus: ClosedLoopStatus [1]

«Experimental»



Vnfc

- «Experimental» + vnfcInstanceId: Identifier [1]
- «Experimental» + nfcNamingCode: String [0..1]
- «Faulty» + description: <Undefined> [0..1]
- «Faulty» + vduId: Identifier [1]
- «Faulty» + I3InterfaceIpv4AddressList: <Undefined> [\*]
- «Faulty» + I3InterfaceIpv6AddressList: <Undefined> [\*]
- «Faulty» + vnfcState: <Undefined> [0..1]
- «Faulty» + inMaint: Boolean [0..1]
- «Faulty» + isClosedLoopDisabled: Boolean [0..1]
- «Experimental» + vnfcInstanceName: String [1..\*]
- «Experimental» + vnfcNamingPolicyId: String [1]
- «Experimental» + nfcFunction: String [1]
- «Experimental» + operationalStatus: OperationalStatus [1]
- «Experimental» + orchestrationStatus: OrchestrationStatus [1]
- «Experimental» + closedLoopStatus: ClosedLoopStatus [1]

«Enumeration»

«Experimental»

ProvStatus

- «Experimental» PROVISIONED
- «Experimental» PREPROVISIONED
- «Experimental» CAPPED

«Enumeration»

«Experimental»

OperationalStatus

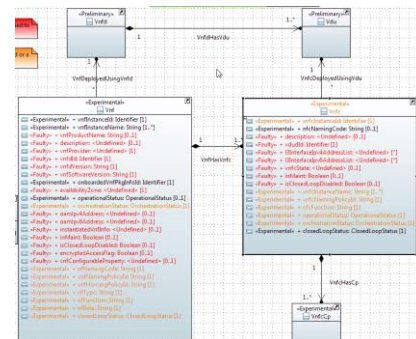
- «Experimental» IN-SERVICE-PATH
- «Experimental» OUT-OF-SERVICE-PATH

«Enumeration»

«Experimental»

OrchestrationStatus

- INVENTORIED
- ASSIGNED
- CREATED
- ACTIVE





# NF MODEL

- ONAP and SDC NF Modeling for 5G RAN
- 5G SDC Project

# SDC PNF MODEL (R3 Casablanca)

Contents	Description
<b>pnfId*</b>	Identifier of this Pnf information element. CORRELATIONID (A&AI). <b>ACTION: Discuss further</b>
<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. MANDATORY
<b>Name*</b>	Provides the human readable name of the PNF.
<b>vendorrelease *</b>	Vendor release. MANDATORY
<b>vendormodelNumber*</b>	PNF Model value (link to A&AI)
<b>functionDescription*</b>	Describes the PNF function
<b>software_versions</b>	The <b>EXPECTED</b> software to be supported by the PNF. (see TOPIC: SWVersionList)

**Sept 4 ACTION** – SDC working with integration team that will use software\_versions. Ben will ask on the Thursday (SW Upgrade Use Case Call)

**ACTION** – Coordinate Modeling Calls? (Tues) “Merge” every once in a while to Sync up. Contact: Deng Hui

**\*Already supported in Beijing**

# TOPIC: SWVERSIONLIST (R3)

July 31, 2018 Discussion about **software\_versions**

## TOPIC:

### **software\_versions in the PNF Model (in Casablanca R3)**

Problem Statement: How will it be defined in SDC

Objective want to have a list of S/W versions

## SOLUTION

Will be a **property (STRING)**

REASON: Meta-Data can't have lists so modeled as a Property.

**TOSCA model** has different sections

Notes: Vendor/resource version as META-DATA for NF

If this is a property has different set of validations

Properties are model information

Inputs to set properties.

Meta-Data (section of TOSCA model of PNF)

Constraints can be imposed upon Properties

An enhancement on "meta-data" which you can impose

Proper / valid values upon the Properties.

New DATATYPES would need to go through Modeling Sub-committee

## Discussion

Linda Horn (Nokia) "don't we only need ONE Expected S/W version?"

Li Xiang (CMCC) "we need a list"

DESCRIPTION for software\_versions – to highlight features in a SW version.

e.g. the set of Services the S/W is targeted for.

### • Content of PNF software version List

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

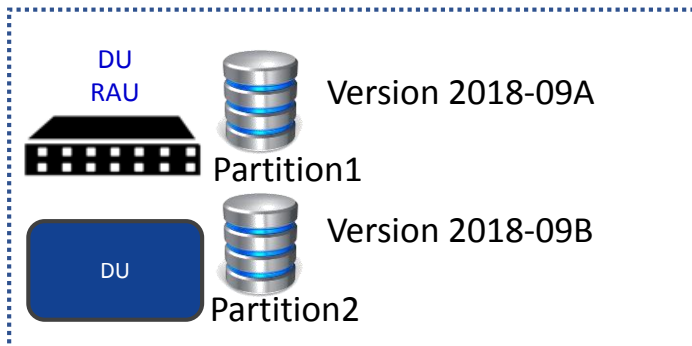


# TOPIC: Use Cases for SWVERSIONLIST (R3)

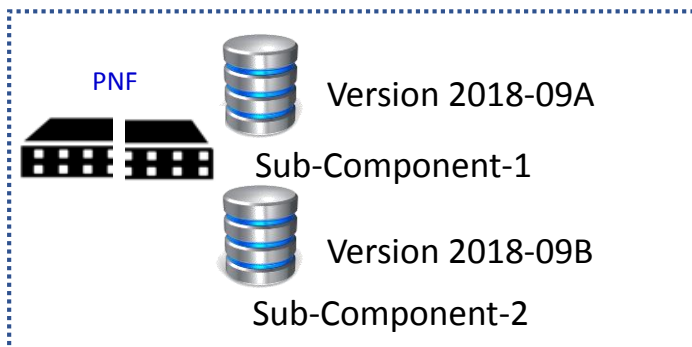
Use Cases that use Software Version LIST	Description
Troubleshooting	Troubleshooting software defects and tracking & isolating problematic software
Network Analysis	Metrics, Data-mining, analysis on PNF and VNF based on software developments.
Correlation Version	KPI, Performance, and operational analysis based on software version and regional correlation.
Error Checking	Mis=deployed software, error correction and analysis.
Modeling informational	Modeling software versions and feature planning
Network Planning	Regional and area planning for software deployment and rollout.
Software Management & Inventory	Feature roll-out and tracking, software management and inventory tracking
Software Upgrade (Network wide)	Service Provider wants to upgrade a network (or fallback a network) to a new software version. Region #1 – v1; Region #2 – v2.

# TOPIC: Use Cases for SWVERSIONLIST (R3)

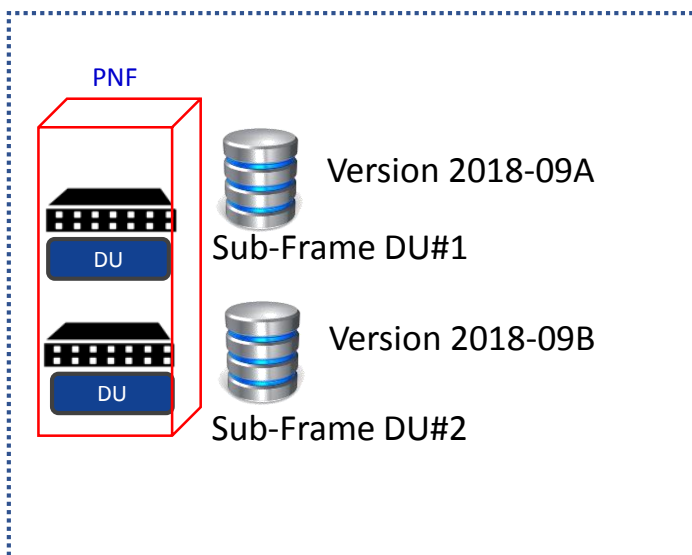
## DRIVE PARTITIONS



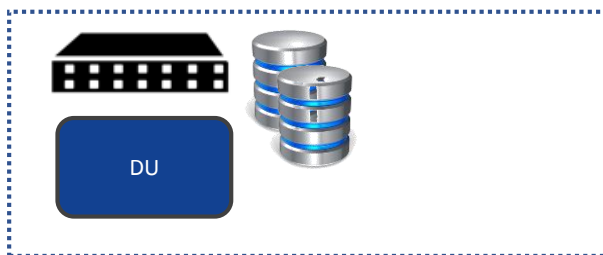
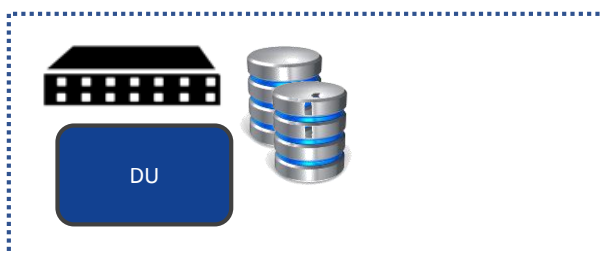
## SUBCOMPONENTS



## TANDEM CHASSIS CONFIGURATIONS



## MULTI-PNF DAISY CHAIN DEPLOYMENT



# TOPIC: Use Cases for SWVERSIONLIST (R3)

Use Cases that use Software Version LIST	Description
Software Upgrade (Network wide)	Service Provider wants to upgrade a network (or fallback a network) to a new software version. Region #1 – v1; Region #2 – v2.

Area #1

Modelled for NF  
Software Version  
2018-09A

Area #2

Modelled for NF  
Software Version  
2018-09A

A planner is responsible for an ONAP deployment that covers a set of areas, in this case #1 and #2.

For this particular use case/application (of this parameters), a class of PNFs is allowed to have a set of Software versions in this case 2018-09A and 2018-09B.

# SDC PNF MODEL (R4) Dublin+

Contents	Description
<b>pnfid*</b>	Identifier of this Pnf information element. CORRELATIONID (A&AI). <b>ACTION: Discuss further</b>
<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. MANDATORY
<b>Name*</b>	Provides the human readable name of the PNF.
<b>vendorrelease *</b>	Vendor release. MANDATORY
<b>vendormodelName*</b>	PNF Model value (link to A&AI)
<b>functionDescription*</b>	Describes the PNF function
<b>software_versions (opt)</b>	The <b>EXPECTED</b> software to be supported by the PNF. (see TOPIC: SWVersionList)
<b>pnfExtConnPt (modelling def. of connection pt not a template)</b>	<div style="display: flex; align-items: center;"> <p>Specifies the characteristics of one or more connection points where to connect the PNF to a VL. Align ETSI SOL-001. <b>ML: connection pt model in TOSCA TEMPLATE not as properties.</b></p> </div> <div style="font-family: monospace; font-size: 0.8em; margin-top: 5px;"> <pre> node_template {   sdccp: {     type: org.onap.sdc.resource.cp.sdccp     metadata {       ...       lowerCaseMID: 94b225e-33c9-476b-acc0-537ef63b644       MID: c0b795ee-5cb4-468b-ba3b-3842830a481       outOfBandMID: 76a36d01-30ef-4370-b66c-11b0e039a1       version: '1'       name: SDCCP       description: The SDCCP Connection Point base type all other CP derive from       category: {         subcategory: Network Element       }       resourceVendor: AT&amp;T (Tosca)       resourceVendorRelease: 1.0.0       resourceVendorReleaseNumber: 0       properties {         ...         ml_required: false         ml_optional: true         ml_generated_name: true       }     }   } } </pre> </div>
<b>contactId (metadata)</b>	Designer (user of ONAP) ML: Need for this, Audit/tracking, User creates audit log not associated with model itself. 19-Jul – REMOVE THIS PARAMETER
<b>PackageVersion</b>	The version of the PNF Package. 19-Jul – Not going to onboard the PNF, model it from scratch. Define in SDC.
<b>NF Controller</b>	Controller for PNF (APP-C, SDN-R, SDN-C, VF-C)

# TOPIC: R3 NF YAML DEFINITIONS

**August 7, 2018**

## **PROBLEM STATEMENT:**

We need to Store a YAML registration event in the SDC Catalog.

Note: The YAML registration event is necessary to validate emitted by PNF Is expected

Emitting what it is supposed to be emitting.

## **SOLUTION (Casablanca R3)**

Manually uploaded to different systems

If no monitoring defined, can define information manually

## **PnP FLOW (updated Wiki)**

[Added Note & PNP-1310] in Wiki

<https://wiki.onap.org/display/DW/5G+-+PNF+Plug+and+Play>

## **LONG TERM SOLUTION**

PNF Onboarding – Packages

(See roadmap section)

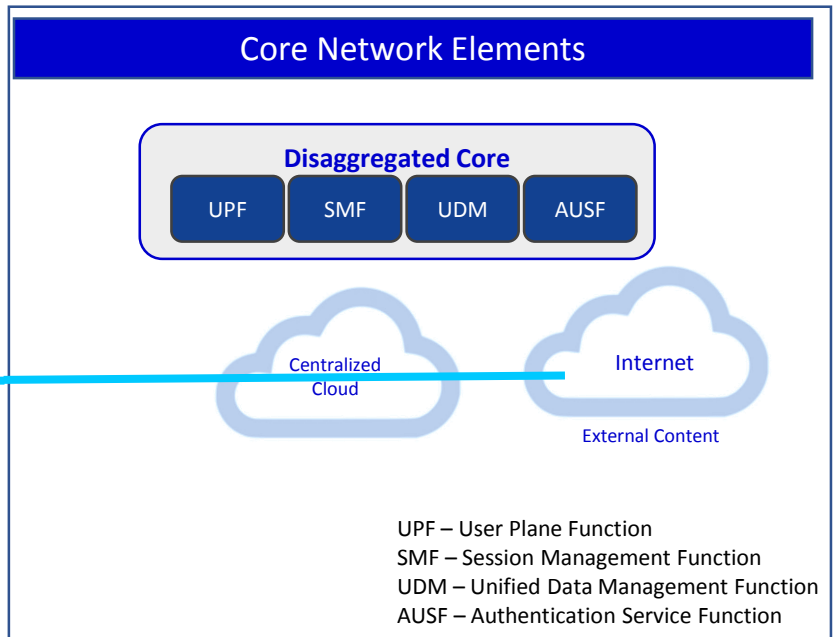
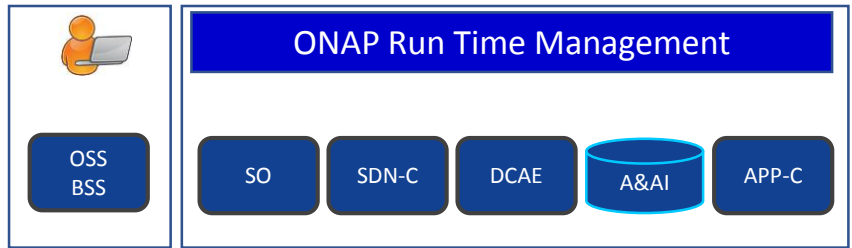


# RAN CONCEPT

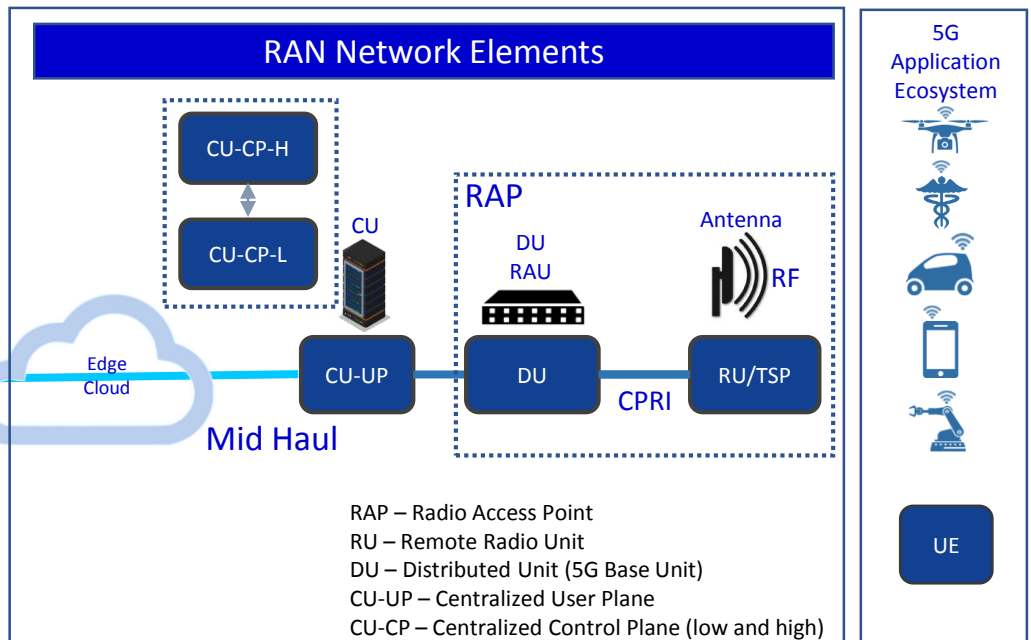
- ONAP and SDC NF Modeling for 5G RAN
- 5G SDC Project

# 5G RAN NETWORK ARCHITECTURE

SO – Service Orchestrator  
 SDN-C – Service Design Network Controller  
 DCA&E – Data Collection Analytics & Events  
 A&AI – Available & Active Inventory  
 APP-C – Application Control



Back Haul



# 5G RAN NETWORK ARCHITECTURE

ONAP  
Management – Design Time

ONAP  
Management - Runtime

RAN Network Slice Segment  
(NSSIs)

Transport Slice Segments  
(NSSIs)

Core Slice Segments (NSSIs)

RAN Abstraction Model

Transport Abstraction  
Model

Core Abstraction Model



ONAP Run Time Management

OSS  
BSS

SO

SDN-C

DCAE

A&AI

APP-C

Core Network Elements

Disaggregated Core

UPF

SMF

UDM

AUSF

Centralized  
Cloud

Internet

External Content

Back Haul

RAN Network Elements

CU-CP-H

CU-CP-L

CU

RAP

DU  
RAU

Antenna  
RF

Edge  
Cloud

CU-UP

DU

CPRI

RU/TSP

Mid Haul

5G  
Application  
Ecosystem



UE





# Associating a Controller for a NF

- ONAP and PNF Plug and Play for 5G RAN for Dublin R4
- 5G Use Case Team

# NF Controller (R3 Casablanca)

## **PROBLEM STATEMENT**

Associating the ONAP Platform Controller (APP-C, SDN-C, VF-C) for a NF

## **OBJECTIVES (Long-Term Goal)**

As automated as possible

Using discovery if possible

Flexible operator could design PersonaA for PNF1, PersonaB for PNF2

## **RESULT**

e.g. SO knows which API to use for NF controller

LCM policy engine, DCAE, Change management

## **NF**

OTN PNF (CCVPN), Router PNFs, 5G DU RAN are PNFs are relevant

## **SOLUTION (R3 Casablanca)**

SDN-C, Hard-Code controller to PNF.

“Hard code” Generic-API

# NF Controller Concepts

## ONAP Deployment

### ONAP Platform Controller (Run Time)

VF-C

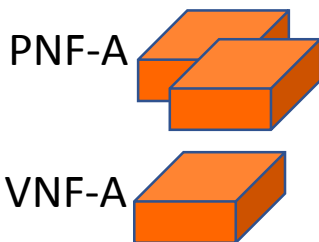
SDN-C  
SDN-R

APP-C

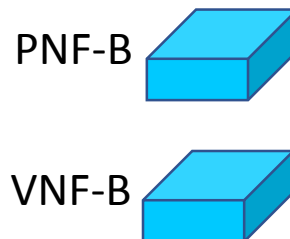
[New/Future]  
X controller

### Technology Domain

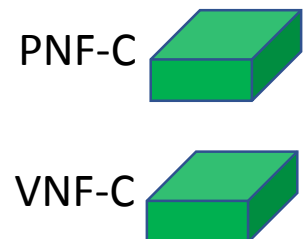
#### Wireless RAN



#### Optical



#### IoT



### PROBLEM DESCRIPTION

Pictured above are three different kinds of PNFs. In orange are wireless (RAN) base stations, such as 5G DU units and their corresponding 5G VNFs. For Optical, there are SOTN PNFs for example as used in the CCVPN use case. Then pictured in green are IoT PNFs. These might include things like smart home units, smart doorbells and the like.

Each of these PNFs fall into a domain category, Wireless, Optical, IoT. These categories are just example categories. There will be many other divisions.

Each of these categories of PNFs & VNFs will have attending Controllers.

For any service provider, (w/ a mix of different vendor NFs, they will have the same Controller)

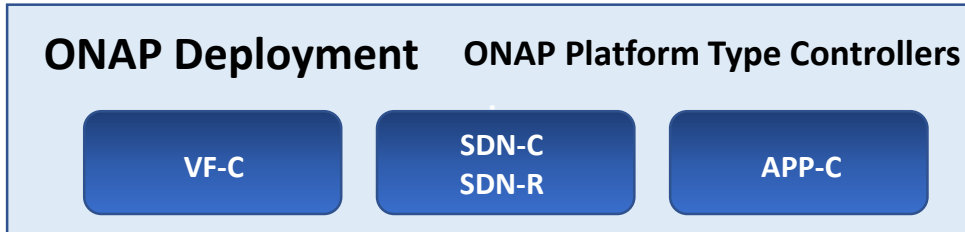
# NF Controller Concepts

## ONAP PLATFORM CONTROLLERS (*Persona*)

(SDN-C (SDN-R), VF-C, APP-C, xyz-C)

### DESCRIPTION

ONAP Platform-Type controllers are SDN-C, SDN-R, VF-C and APP-C. These are specific types of ONAP projects that are controllers to NFs.



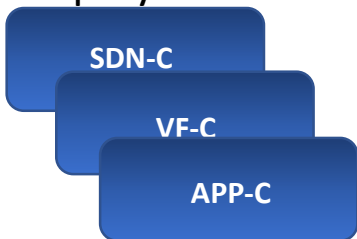
## Regional CONTROLLER (*Instances*)

Regional Deployment (instances) of Controllers

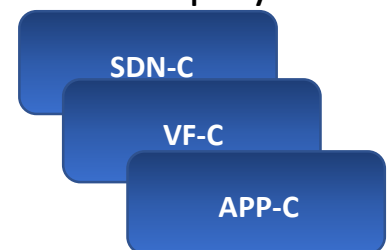
### DESCRIPTION

Regional Controllers are specific instances of ONAP platform-type controllers deployed to a particular region or responsible for a particular region. For example SDN-C deployment #1 responsible for the western part of a country, and SDN-C deployment #2 responsible for the eastern part of a country

ONAP Deployment #1



ONAP Deployment #2

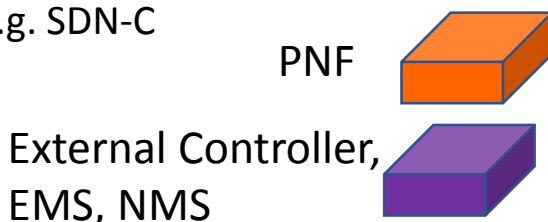


## External (to ONAP) CONTROLLERS

(OSS): EMS, NMS, Vendor proprietary controllers, etc

### DESCRIPTION

ONAP External Controllers that reside outside of ONAP that perform management functions with the PNF and VNFs. Incl. Vendor-proprietary controllers e.g. SDN-C



# NF Controller PROPOSALS (Goal)

## PROPOSAL #1

NF Model (SDC Design Studio)

### DESCRIPTION

To have the Controller as an attribute as a NF model is specified in the NF Model. Differences between PNFs & VNFs. The PNF has a req for a physical device, VNF does not. Both NFs need controllers.

Objection to model in SDC is that the VID user may not know the controller. The model designer & The Network Engineering should know.

Problem #1 - Requires designer to know the controller

Problem #2 - How is this managed (a hard-coded list) e.g. a 3<sup>rd</sup> party External Controller

## PROPOSAL #2

Policy Driven

### DESCRIPTION

A policy is designed which has the Controller used by the NF.

Problem #1 – That's not what the policy function does

## PROPOSAL #3

Table Driven Association

### DESCRIPTION

Table-Driven Look-up solution based on NF function type. For example a controller may support a particular technology domain (wireless/wireline/optical). Controller support domain and auto-populates the tables. Could be a GUI in SDC (a run-time catalog table). The Table could be onboarded. Design-time field. The PNF needs to have a "Technology" domain (a user or designer). Specific images (S/W loads) to specific Controllers.

1 "ONAP platform type Controller" SDN-C (SDN-R) VF-C APP-C

2. Domain Controller - Controller-Instances (regional dependent)

ONAP deployment [controller] – Domain Controller – ONAP Controller

OTN PNF = "optical" domain = controller-z

OTN PNF w/ S/W load 1.1.1.2 = controller-X

OTN PNF w/ S/W load 1.1.1.3 = controller-y

Scale, US/Europe, W-E coast. REGIONAL

3. Vendor / External Controller

Question – who defines the "Domain". Defined by Service Provider.

# NF Controller PROPOSALS (Long-term Goal)

## Design Time

Tech Domain	PNF	ONAP Platform Controller	API version/ variation/ name Protocol
Wireless	E// 5G DU	SDN-C	SDN-C v1.1 Generic API
	Nokia 5G DU	SDN-C	SDN-C v1.2
Wireless Subdomain 1	Xyz 5G DU	VF-C	VF-C v2
Wireline	Wireline PNF	APP-C	
Optical	OTN PNF#2	SDN-C	

SO Recipe different controller per flow  
Lifecycle Management vs CM  
SSH key to NF to authenticate controller  
Data structure/Platform-Application Data

# NF Controller PROPOSALS (Goal)

## DESIGN-TIME

SDC – Design Studio

Onboard a resource (type, role, function, [tech domain])

Deduce tech domain? From type-role-function?

Operator specifies the Technology Domain of the NF

Operator specified the Technology domains

(or possibly Techdomain is deduced from type-role-function)

Service Provider defines the possible Technology Domains

Assign every NE and Service to a Technology Domain

TD1 = SDN-C TD2 = VF-C manually modeled, table created.

TechDomain to ONAPPlatformController

SDC – Model “mapping” > Catalog

## RUN-TIME

As a NF registers, the (managing ONAP entity e.g. SO for PNF) would look up the TD > OPC mapping)

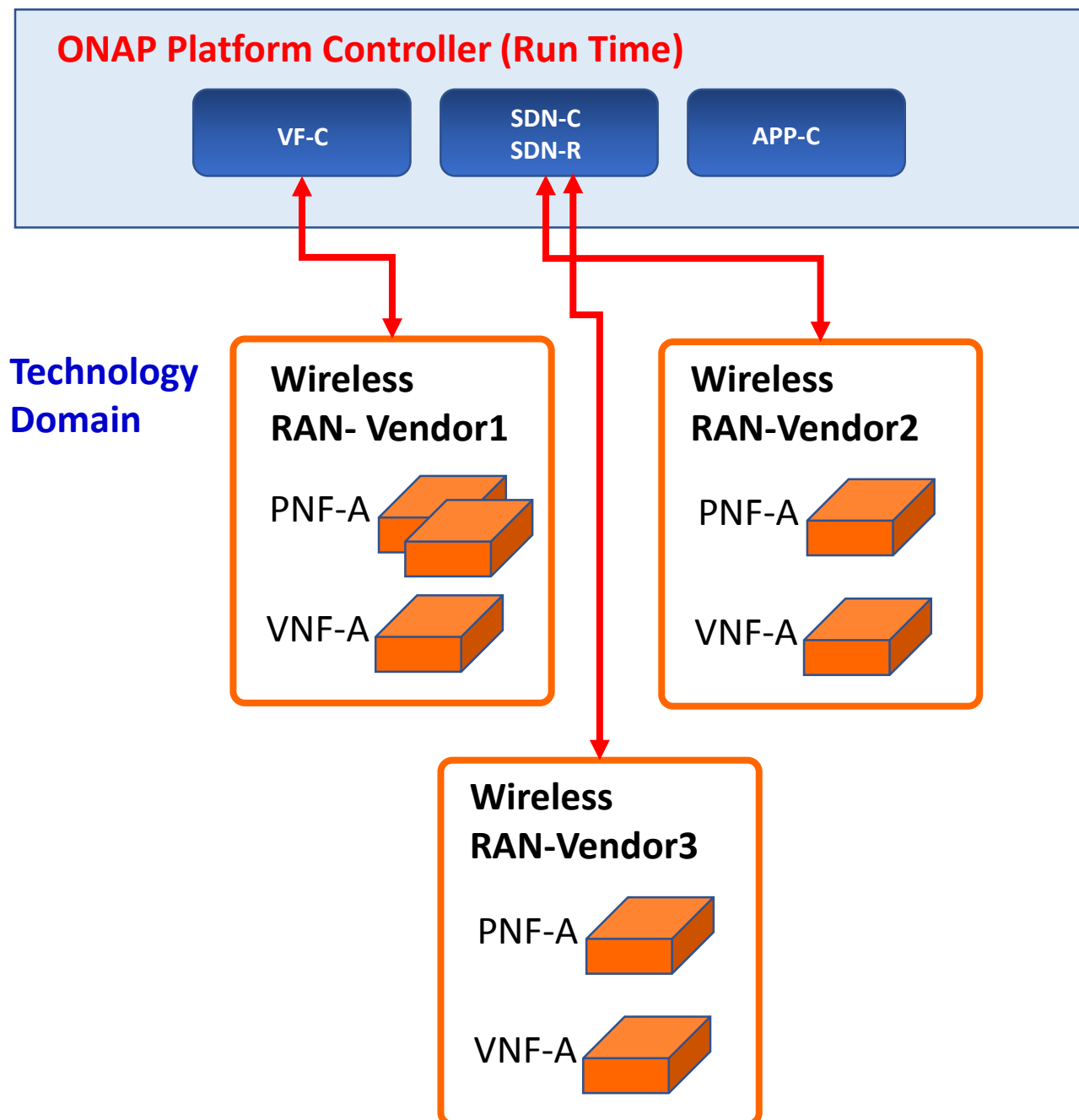
3 Technology Controllers (Wireless, Enterprise, Other)

When generate controller it populates the table.

Dynamically populate table

# Who Chooses the Controller?

Who Choose the ONAP controller type for the NF?





# NF Controller - Notes

Identifying the NF controller

For VNF is part of the Call Flow

VNF gets orchestrated through recipe & DG & Yang models  
(assumption is SDNC is the controller)

VNF can have own domain controller

DG pass control VN adaptor to 3<sup>rd</sup> party controller

PNF controller to be discovered as part of the PnP Flow

Provision PNF manually specify the Controller

SO passes to APPC service instance

SO pulls service info

PNF (CU) must be configured first

CU configuration process (could identify the ONAP controller)

PNF (routers, access pts, RAN 5G DU, CU)

- SDN-C, VF-C, x-controllers
- ONAP SO needs to know what API & Controller for PNF.
-

# NF Controller - Notes

NOTES: July 17<sup>th</sup> Modeling Discussion

Seshu

We have talking of a TOSCA event-based flavor given

To the WF entity-based;

SO Trying to adapt – The NF controller modeled-

User on client side selects the controller (also has problems)

If not the UI, the modeler who can understand this is the case

Information that something is a missing point – trying to

Can we have a understanding in des-time; for the controller

SDN-C, APP-C, VF-C, GNF-C etc

Resources compatible w/ this resource type

A designer driven

Designer may not know

PNF PNP does not use OOF

This is a RECURRENT problem (also encountered in Scaling use case)

Alex Vul

PLACEMENT POLICY – POLICY FRAMEWORK

Design times generic, bind specific VNF

Map to OOF.

Mechanism exists.

Log des; phys infrastructure; binding between 2 (by OOF)

Heirarchical orchestration

Policy design by a Human Operator, designs the policy of PNF.

Designer doesn't know controller;

Chaker AlHakim

Restate the problem

Add an attribute in A&AI

Register a SERVICE. Add service in A&AI.

Best way to register the service controller is providing.

Creating the service don't know physical/virtual resources.

Srini Vellanki

Workflow designer

SERVICE MODEL – which controller to use



## A&AI (Reference)

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

# ACTIVE INVENTORY (A&AI) R3/Casa

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

### New A&AI PNF Parameters

~~PNF GEOLOCATION~~ - geographical location (e.g. coordinates or address of the building, etc.). Latitude/Longitude. **THIS ALREADY EXISTS VIA ASSOCIATION TO THE "COMPLEX" OBJECT.** The Complex Object represents a BUILDING or location with geographical information. The AAI PNF will have a UML association to the Complex object.

### Software Version



**DETECTED SOFTWARE PNF VERSION(S) – swVersionList.**

**R3 Casablanca** – This will be a list of software versions. In Run-Time when PNF registers with ONAP it can report its (list) of PNF Software that is currently has installed. This will be tracked in A&AI entry for that PNF. Entry will also an indication that one is *Active*.

**R4 Dublin** – proposing software Object which could be linked by SDC and A&AI.

**swVersion [1...x] (Array)**

```
{  
    swVersion (String)  
    activeSw (Boolean)  
}
```

### Homing



**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]

OOF determine the homing of a NF. Anything you home is determined in the context of a deployment. (Homing) Policy used as a f(service). Data center might have been divided into cloud regions. Service VMME running in NE area (distances, regions, tenants where to instantiate PNF). AAI has **COMPLEX node**. “Physical Location ID” (8 char CLLI code, lat/long = geolocation info of data center. Cloud region doesn’t span data centers.

### Manager IP Address



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

**SUGGESTION (from Christina A&AI PTL) model the NMS as a PNF itself (and the NMS will have parameters to represent itself; and then PNF can be associated with it).**

# S/W Image Repository / R3 Casa

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

**S/W Image  
Repository**



**S/W Image Repository** – Where the S/W is located.  
Configuration Parameter.

# PNF A&AI Entry

PARAMETER	DESCRIPTION
<b>PNF-NAME</b>	<p><i>pnf-name</i> is the Key in AAI. <i>pnf-name</i> is the first three letters of the Vendor and PNF serial number. This is a unique identifier for the PNF instance. It is also called the Correlation ID.</p> <p>Note: MAC address &amp; serial number are unique per vendors; not across vendors, so the Vendor name is added to insure uniqueness.</p>
<b>EQUIP-TYPE</b>	The <i>equip-type</i> parameter gives the type of the PNF.
<b>EQUIP-VENDOR</b>	The <i>equip-vendor</i> is an optional parameter which indicates the vendor for the PNF. For example, Nokia or Ericsson.
<b>EQUIP-MODEL</b>	The <i>equip-model</i> is an optional parameter which indicates the model of the PNF.
<b>PNF-ID</b>	UUID = Service provider assigned number from network planner.
<b>MANAGER IP ADDRESS</b>	adds <i>ipaddress-v4-oam</i> ; <i>ipaddress-v6-oam</i> This is the “manager IP Address” which for a DU might be a CU IP address; (FYI/ <i>ipaddress-v4-loopback-0</i> ).
<b>MAC ADDRESS</b>	This is the MAC address of the PNF. This is a service field.
<b>SERIAL NUMBER</b>	This is the serial number of the PNF. This is a service field.
<b>PROXY IP ADDRESS</b>	This field contains the <i>proxy IP address</i> for the PNF.



# PNF Plug and Play ROADMAP (After Casablanca)

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

# TOPIC: SWVERSIONLIST (R4+)

## TOPIC:

### Adding Description to software\_versions (after Casablanca R4+)

Problem Statement: Adding Description to SWVersionList

Problem Statement: Add new **DATATYPE**

(going through the modeling subcommittee)

## SOLUTION

### • Content of PNF software version List

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

Want a DATA STRUCTURE in the TOSCA MODEL for the NF Model

## Discussion



New Data Type for modeling PNF Software Versions

## Software (object)

FIELD	TYPE	DESCRIPTION
software-id	String	Index key for this object
swDescription	String	Descriptive text of the software
swVersion	String	The version of this software
swVendor	String	Vendor for this software
swVersionLabel	String	Semantic label for the software

# PNF PACKAGE (R4)

## ALARM DEFINITIONS

Alarm Dictionary  
(Vendor provided)

YAML Definitions  
(Vendor provided)



## MEASUREMENT DEFS

Measurement Dictionary  
(Vendor provided)

YAML Definitions  
(Vendor provided)

Measurement Schema  
(Vendor provided)



## CONFIGURATION DEFS

Configuration Schema  
(Vendor provided)



YAML Definitions  
(Vendor provided)



## EVENT DEFS

Events

YAML Definitions  
(pnfRegistration, HVVs)



## PLAYBOOK DEFS

NF Ext Controller Playbooks



## WORK-FLOWS

Work-Flows

## POLICY

Identity Controller

**UNDER DISCUSSION**



## 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]

PNF Alarm #22

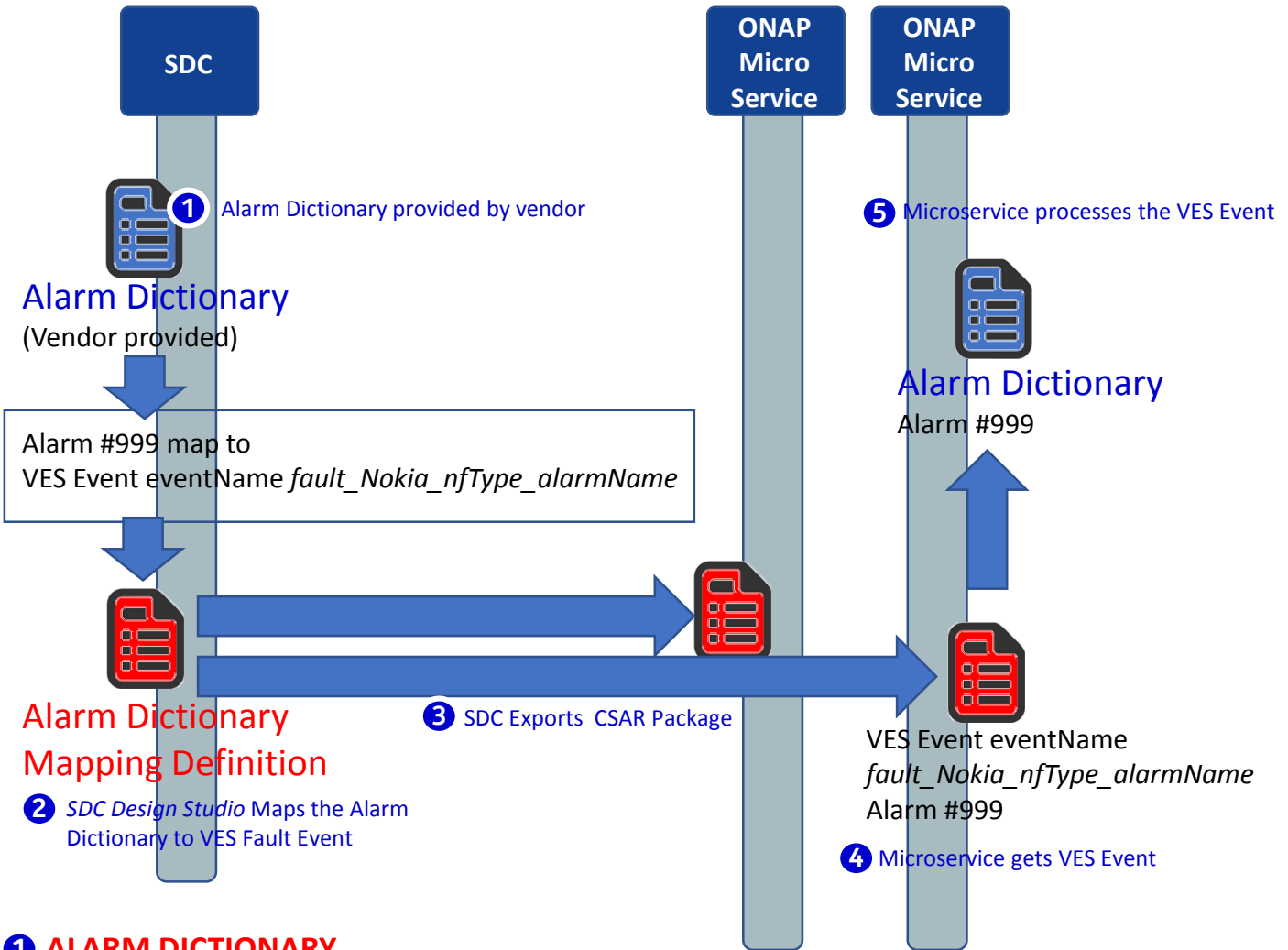
#22 S/W problem xyz

VES = fault fields; "fault" domain publish DMaaP

Clamp, Analytic, Holmes – [#22 event]

Dictionary looks up #22 ->

# 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. (RATHER than mS doing the mapping because it is more model driven)

## 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

# Alarm Dictionary Usage

## ALARM DICTIONARY PURPOSE


- (1) DICTIONARY** - it allows for a readily accessible body of the entire set of alarms & faults that are managed by a PNF. It would allow for an operator to see all of the alarms & faults of a PNF without having to wait for individual alarms & faults to arrive in ONAP.
- (2) Analytics facilitator** – A dictionary would allow for a variety of vendor specific (or vendor agnostic) analytics applications to be developed. There are a variety of fields in the Alarm Dictionary that would facilitate such analytics capabilities as correlation, escalation, isolation, recovery actions, self-healing, and life cycle management functions.
- (3) GENERAL ANALYTICS** – The strength of ONAP is the potential ability to coordinate information from multiple sources, different vendors, and disparate types of NFs. A dictionary can form the foundation for generalized analytics that are vendor agnostic.

## FAULT DICTIONARY PURPOSE

- (1) FAULTS vs ALARMS** - Fault can be a condition encountered in run-time that does not necessarily create a customer-facing alarm. An alarm is intended to result in a visual notification to a service provider to take action. An analogy would be the “Check engine” light in your car which would correspond to an Alarm. A solenoid, a carburetor, or distributor fault all might lead to a “Check engine” light. A driver (service provider) may not be able to directly act on the specific fault (or indeed care about the fault); but when the “check engine” light went on would know to take some action (go to the service station).

# NF ALARM DICTIONARY FIELDS (Template)



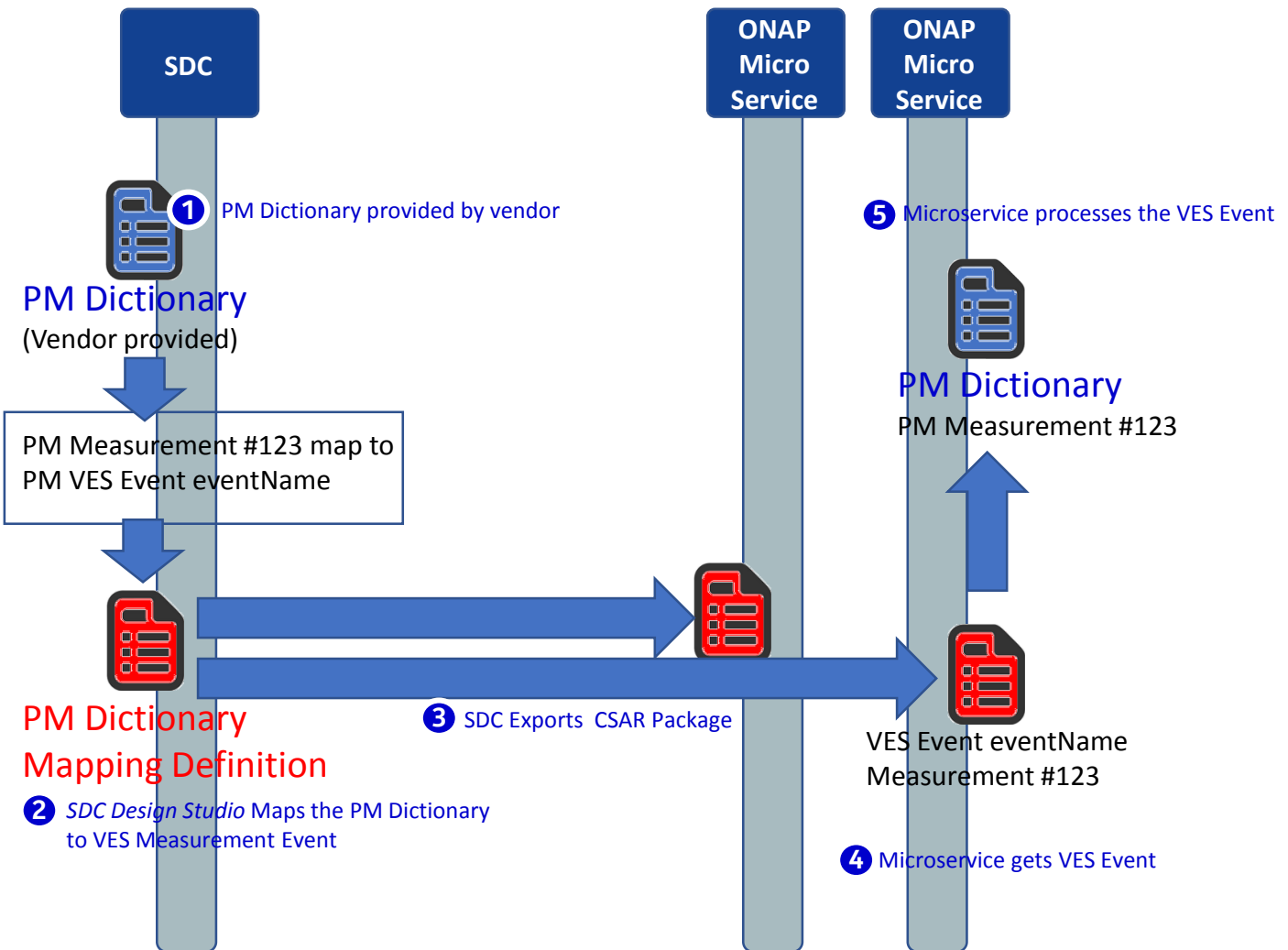
ALARM FIELD	DESCRIPTION
<b>Alarm Dictionary Index</b> 	<p>Gives the Identifier for the alarm. This is also the Identifier that is used in the VES event so it can be used to associate the event with the definition entry.</p> <p>The VES Event <i>EventID</i> would encode the Alarm number which will correspond to the Alarm Index.</p> <p>EXAMPLE: 12345</p>
<b>Alarm Name</b>	<p>Alarm Name which will be used in the Event Name. <b>Note</b> <i>this maps to the <a href="#">alarmCondition</a> in the VES Fault Event in faultevent fields.</i></p> <p>EXAMPLE: Synchronization Lost</p>
<b>Event Type</b>	<p>Indicates the type of alarm. The types are: Communications Alarm, Processing Error Alarm, Environmental Alarm, Quality of Service Alarm, Equipment Alarm, Integrity Violation, Operational Violation, Physical Violation, Security Service Violation, Mechanism Violation, or Time Domain Violation. <b>Note</b> <i>this maps to the <a href="#">eventCategory</a> in the VES Fault Event in faultevent fields.</i></p> <p>EXAMPLE: Quality of Service Alarm</p>
<b>Meaning of Alarm</b>	<p>Provides a descriptive meaning of the alarm condition. This is intended to be read by an operator to give an idea of what happened.</p> <p>EXAMPLE: Synchronization has been lost</p>
<b>Effect of Alarm</b>	<p>Provides a description of the consequence of the alarm condition. When this alarm condition occurs. This is intended to be read by an operator to give a sense of the effects, consequences, and other impacted areas of the system.</p> <p>EXAMPLE: Loss in Quality of Service</p>
<b>Managed Object(s)</b>	<p>Managed object (MO) associated with this Alarm. <b>Note</b> <i>this maps to the <a href="#">eventSourceType</a> in the VES Fault Event in faultevent fields.</i></p> <p>EXAMPLE: Clock (MO)</p>
<b>Probable Cause</b>	<p>Provides the probable cause qualifier for the alarm. Probable causes are found in 3GPP TS 32.111 Annex B drawn from ITU-T M.3100 and from ITU-T Recommendation X.721, X.733, and X.736</p> <p>EXAMPLE: lossOfSynchronisation</p>
<b>Probable Cause Number</b>	<p>Probable Cause Number the numeric value associated with the Probable Cause</p> <p>EXAMPLE: 76</p>
<b>Specific Problem</b>	<p>(Optional) It provides further qualification on the alarm than probable Cause. This attribute value shall be single-value and of simple type such as integer or string. Defined in ITU-T Recommendation X.733 Clause 8.1.2.2. <b>Note</b> <i>this is the 3GPP Specific problem not be confused with the <a href="#">specificProblem</a> field of the VES Fault Event in faultevent fields.</i></p>
<b>Proposed Repair Actions</b>	<p>It indicates instructions for proposed repair actions. These are defined in ITU-T Recommendation X.733 clause 8.1.2.12.</p> <p>EXAMPLE: Reset the BTS, ONAP Controller does x</p>
<b>Clearing Type</b>	<p>Indicates whether the alarm is automatically or manually cleared</p> <p>EXAMPLE: Automatic</p>
<b>Additional Text</b>	<p>This field contain further information on the alarm. This attribute provides <i>vendor specific</i> alarm information. A specific condition for this optional population is when an alarm presented by the EM has different values of perceived severity, and / or alarm type.</p> <p>EXAMPLE: Specific data 10</p>
<b>Associated Fault(s)</b>	<p>Indicates the associated faults that triggered this alarm. List of fault(s) associated with the alarm cross indexed against a vendor provided fault information.</p> <p>EXAMPLE: Fault 99999</p>

# NF FAULT DICTIONARY FIELDS (Template)



DEFINITION FIELD	DESCRIPTION
<b>Fault Id</b>	Gives the Identifier for the alarm. This is also the Identifier that is used in the VES event so it can be used to associate the event with the definition entry. EXAMPLE: 99999
<b>Fault Name</b>	Alarm Name which will be used in the Event Name. <b>Note</b> <i>this maps to the alarmCondition in the VES Fault Event in faultevent fields.</i> EXAMPLE: Loss of Synchronization
<b>Fault Description</b>	Provides a descriptive meaning of the alarm condition. This is intended to be read by an operator to give an idea of what happened. EXAMPLE: Synchronization due to PTP IEEE1588 Failure
<b>Managed Object(s)</b>	Managed object (MO) associated with this Alarm. <b>Note</b> <i>this maps to the eventSourceType in the VES Fault Event in faultevent fields.</i> EXAMPLE: Clock (MO)
<b>Effect of Fault</b>	Provides a description of the consequence of the alarm condition. When this alarm condition occurs. This is intended to be read by an operator to give a sense of the effects, consequences, and other impacted areas of the system. EXAMPLE: Loss of synchronization affect QoS
<b>Associated Alarm(s)</b>	Indicates the associated faults that triggered this alarm. List of fault(s) associated with the alarm cross indexed against a vendor provided fault information. EXAMPLE: 12345
<b>Proposed Repair Actions</b>	It indicates instructions for proposed repair actions. These are defined in ITU-T Recommendation X.733 clause 8.1.2.12. EXAMPLE: Reset BTS
<b>Additional Text</b>	This field contain further information on the alarm. This attribute provides <i>vendor specific</i> additional fault information.

# PM Dictionary Usage



## **1 PM DICTIONARY**

PM Dictionary defines all measurements published by xNF (x=V or P)  
Based on 3GPP TS32.503, ETSI, and VES document (v6.0) [has Cloud scaling counters]

## **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

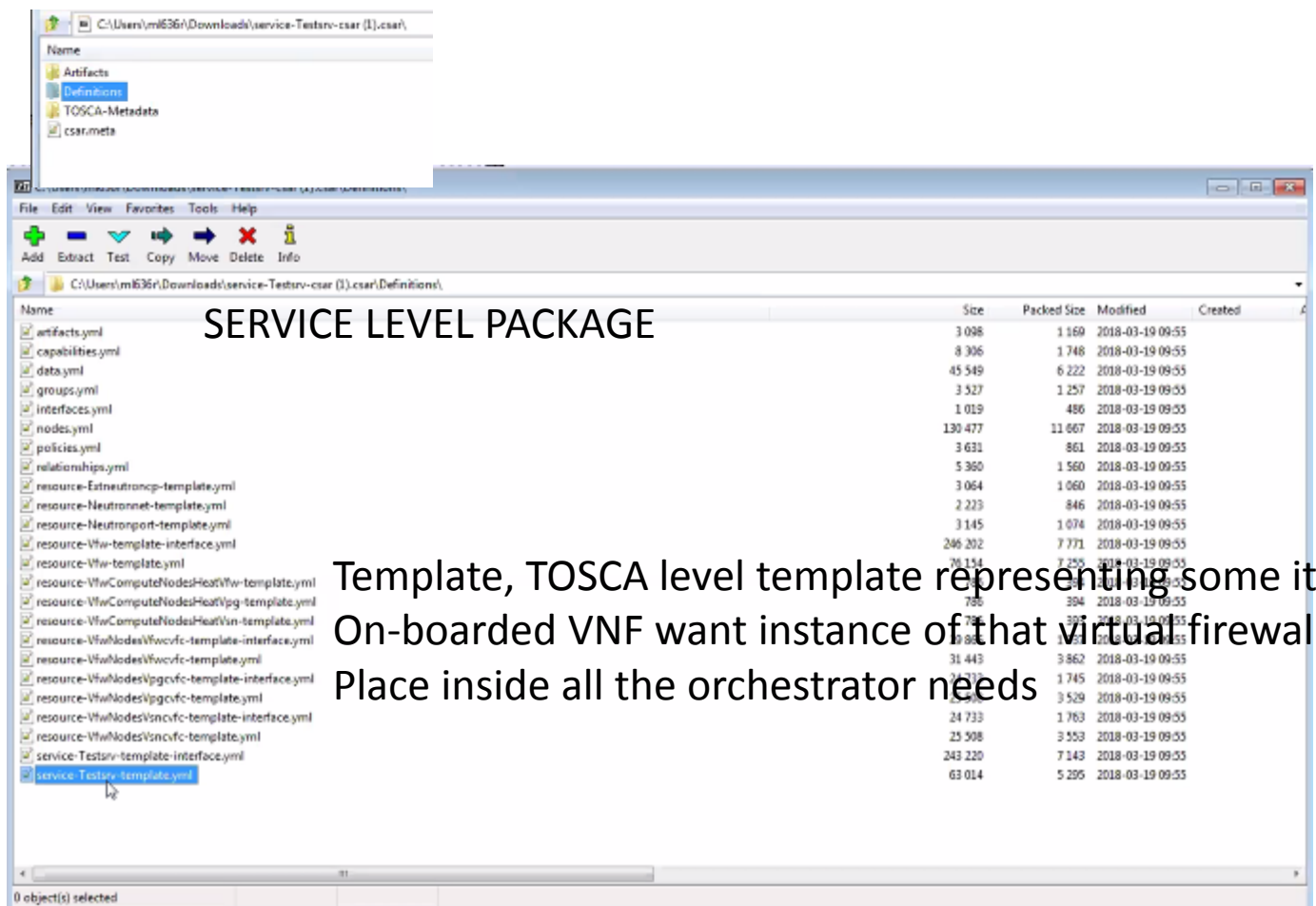
# PM Dictionary Usage





# APPENDIX & Meeting Notes





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

```

topology_template: 03
node_templates: 04
  - vfw 0: 05
    type: org.openecomp.resource.vf.vfw 06
    metadata: 07
      invariantUUID: fa9f72b3-d656-4e08-ae5e-9482ab165d5c 08
      UUID: 0a9e817b-4c98-455a-95d4-71e24713f73d 09
      customizationUUID: -E7432b08-ed4a-43fa-b179-40d2c5ff1b1e 10
      version: '0.2' 11
      name: vfw 12
      description: saad 13
      type: VF 14
      category: Network L4+ 15
      subcategory: Common Network Resources 16
      resourceVendor: example 17
      resourceVendorRelease: '1.0' 18
      resourceVendorModelNumber: '' 19
    properties: 20
      nf_naming: 21
      scomp_generated_naming: true 22
      multi_stage_design: false 23
      availability_zone_max_count: 1 24
      vfw_flavor_name: 4 GB General Purpose v1 25
      public_net_id: 00000000-0000-0000-0000-000000000000 26
      vfw_image_name: Ubuntu 14.04 LTS (Trusty Tahr) (PVHVM) 27
    capabilities: 28
      network.incoming.bytes.rate.vpg.vpg_private_1.port: 29
    properties: 30
      unit: B/s 31
      description: Average rate of incoming bytes 32
      type: Gauge 33
      category: network 34
    disk.read.bytes.vpg: 35
  
```

pnfExtConnPt (modelling def. of connection pt not a template)

Specifies the characteristics of one or more connection points where to connect the PNF to a VL. Align ETSI SOL-001. ML: connection pt model in TOSCA TEMPLATE not as properties.

The screenshot shows a TOSCA editor interface. On the left is a component palette with various types like 'ExtVL', 'Generic Neutron...', 'NeutronNet', 'NeutronPort', 'Port', 'subInterface', 'Vdu Cp', and 'Infrastructure'. The main workspace displays a network diagram with nodes 'Port 0' and 'Port 1' connected by lines. On the right, a metadata panel shows details for a connection point, including UUIDs, name, description, and vendor information.

```
file: resource=Examplepnf-template-interface.yml
resource-Port:
  file: resource-Port-template.yml
topology_template:
  inputs:
  nf_function:
    type: string
  nf_role:
    type: string
  nf_type:
    type: string
  node_templates:
  Port 0:
    type: toasca.nodes.network.Port
    metadata:
      invariantUUID: 941e0b80-67df-4866-95dd-3a841a2861a8
      UUID: 20da15c2-22c8-4cf8-a6ad-eb974d87f6c8
      customizationUUID: f5aae209-2d33-4651-9a12-0ac16d6b17cb
      version: '1.0'
      name: Port
      description: Represents a logical entity that associates between Compute and Network normative types.
      type: CP
      category: Generic
      subcategory: Network Elements
      resourceVendor: ATT (Tosca)
      resourceVendorRelease: 1.0.0.wd03
      resourceVendorModelNumber: ''
    properties:
      is_default: false
      order: 0
  substitution_mappings:
    node_type: org.openeo.comp.resource.pnf.Examplepnf
  capabilities:
  ...
topology_template:
  inputs:
  nf_function:
    type: string
  nf_role:
    type: string
  nf_type:
    type: string
  node_templates:
  Port 0:
    type: toasca.nodes.network.Port
    metadata:
      invariantUUID: 941e0b80-67df-4866-95dd-3a841a2861a8
      UUID: 20da15c2-22c8-4cf8-a6ad-eb974d87f6c8
      customizationUUID: f5aae209-2d33-4651-9a12-0ac16d6b17cb
      version: '1.0'
      name: Port
      description: Represents a logical entity that associates between Compute and Network normative types.
      type: CP
      category: Generic
      subcategory: Network Elements
      resourceVendor: ATT (Tosca)
      resourceVendorRelease: 1.0.0.wd03
      resourceVendorModelNumber: ''
    properties:
```

Metadata details from the right panel:

```
invariantUUID: 53f8b494-6e84-4f2f-9445-5793a603e2e6
UUID: 37f956eb-3c54-4d93-8ba9-f1638248e71c
name: examplepnf
description: '123123'
type: PNF
category: Allotted Resource
subcategory: Allotted Resource
resourceVendor: '12344'
resourceVendorRelease: '1243235'
resourceVendorModelNumber: '1232454'
```

The screenshot displays the ONAP Service Orchestrator interface. At the top, the logo for OPEN DAYLIGHT and ONAP are visible, along with the text "Service Orchestrator" and a "Logout (admin)" link. The main content area is titled "Design Phase" and "Create PNF via SO in A&AI". It features a form with the following fields:

- Name:** New-PNF
- Identifier:** New-PNF network unique identifier
- Equipment type:** Fancy equipment type
- Equipment model:** Best in class
- Equipment vendor:** ONAP SDN-R Community
- IPv4 address:** 10.10.10.10

A blue button labeled "+ Create PNF in Active and Available Inventory" is positioned below the form. At the bottom of the page, the text "ONAP SDN-R | ONF Wireless for OpenDaylight Carbon-SR1 - Build: 2018-06-18 13:43 UTC" is displayed. The left sidebar contains a navigation menu with items such as "Nodes", "Connect", "ONAP AAI", "ONAP DCAE", "ONAP SO", "pnf Fault", "pnf Config", "pnf PM Current", "pnf PM History", "pnf PM Link", "pnf Inventory", "pnf Topology", "pnf Mediator", and "Help". The top right corner shows system status: "Nodes: 1 | Alarm status: 4 | 0 | 0 | 2 | Sum: 6 | Help".

SDN-R in Open Daylight create a A&AI PNF entry

# software\_versions (SDC Demo)

Demo given by Michael Lando (Aug 21, 2018)

Demonstrating software\_versions in SDC Design Studio

## Properties Assignment

Properties Assignment interface showing a table of properties and a search bar.

Property Name	From Instance	Type	Value
nf_function		string	
software_versi...		list	
nf_role		string	
nf_type		string	

Buttons: Search, Declare, Discard, Save

Right panel: Composition, Property Structure, EXAMPLE, No data to display

## Properties Assignment

Properties Assignment interface with expanded 'software\_versions' property.

Property Name	Type	ES	Value
EXAMPLE 0			
<input type="checkbox"/> nf_function	string		
<input type="checkbox"/> software_versions	list	string	<input type="text" value="1"/> <input type="text" value="10"/>
<input type="checkbox"/> nf_role	string		
<input type="checkbox"/> nf_type	string		

Buttons: Search, Declare, Discard, Save

Right panel: Composition, Property Structure, SOFTWARE\_VERSIONS, No data to display

## TOSCA Artifacts

Name	Type	Version
Tosca Model	TOSCA_CSAR	0
Tosca Template	TOSCA_TEMPLATE	0

Name	Size	Packed Size	Modified	Created	Accessed	Attributes	Encrypted	Comment	CRC	Method	Charac
interface-Test-template.yml	1,592	698	2018-08-21 13:29				-		041C0007	Deflate	Descrip
service-Test-template-interface.yml	420	187	2018-08-21 13:29				-		C8014A49	Deflate	Descrip
resource-Example-template.yml	888	434	2018-08-21 13:29				-		A244C2B3	Deflate	Descrip
resource-Example-template-interface.yml	641	256	2018-08-21 13:29				-		18AFA005	Deflate	Descrip
relationships.yml	6108	1,446	2018-08-21 13:29				-		E3DBF59E	Deflate	Descrip
policies.yml	4,557	1,099	2018-08-21 13:29				-		E95763F5	Deflate	Descrip
nodes.yml	129,257	11,795	2018-08-21 13:29				-		AFF613F8	Deflate	Descrip
interfaces.yml	3,113	797	2018-08-21 13:29				-		3CE80215	Deflate	Descrip
groups.yml	4,883	1,502	2018-08-21 13:29				-		8E8B8096	Deflate	Descrip
data.yml	53,536	7,015	2018-08-21 13:29				-		A847239E	Deflate	Descrip
capabilities.yml	9,108	1,899	2018-08-21 13:29				-		B1880761	Deflate	Descrip
artifacts.yml	3,180	1,182	2018-08-21 13:29				-		8BF1789A	Deflate	Descrip
annotations.yml	1,025	550	2018-08-21 13:29				-		439CAD06	Deflate	Descrip

# software\_versions (SDC Demo)

```
27 - policies:
28   - file: policies.yml
29 - annotations:
30   - file: annotations.yml
31 - service-test-interface:
32   - file: service-test-template-interface.yml
33 - resource-example:
34   - file: resource-Example-template.yml
35 - resource-example-interface:
36   - file: resource-Example-template-interface.yml
37 topology_templates:
38   - node_templates:
39     - example 0:
40       - type: org.openecomp.resource.pnf.Example
41     - metadata:
42       - invariantUUID: 5718362c-ec6a-431b-afd7-f5ab63bf3039
43       - UUID: b553ba24-f006-494a-9608-d1642a3401e0
44       - customizationUUID: 634e2a1e-1fac-4bb3-aca3-35f3a0b00c5e
45       - version: '0.1'
46       - name: example
47       - description: '123'
48       - type: PNF
49       - category: Application I4+
50       - subcategory: Web Server
51       - resourceVendor: '123'
52       - resourceVendorRelease: '123'
53       - resourceVendorModelNumber: ''
54     - properties:
55       - software_versions:
56         - '11'
57         - '10'
58     - substitution_mappings:
59     - node_type: org.openecomp.service.Test
60
```

```
Search "dsasp" (3 hits in 1 file)
C:\Users\val636r\AppData\Local\Temp\asp27333\data\logs\BE\SDC\SDC-BE\error.log (3 hits)
Line 28255: 2018-08-21T12:17:49.551Z [main]SDC catalog||ExecuteRestRequest||INFO|0|Starting to consume topic for DMAP consumer with the next parameters DmapConsumerConfiguration [hosts=localhost:3905, consumerGroup=dc, con
Line 28256: 2018-08-21T12:17:49.552Z [main]SDC catalog||ExecuteRestRequest||INFO|0|Starting to consume topic for DMAP consumer with the next parameters DmapConsumerConfiguration [hosts=localhost:3905, consumerGroup=dc, con
Line 28256: 2018-08-21T12:17:49.562Z [main]SDC catalog||ExecuteRestRequest||ERROR|500|An error occurred upon consuming topic by Dmap consumer client.: null
Search "dsasp" (0 hits in 0 files)
```

HOME CATALOG ONBOARD DCAE-DS WORKFLOW

ADD	IMPORT	CIService7a73ba6893 V 1.0 Waiting For Distribution	CIService182e812e6eb9 V 1.0 Waiting For Distribution	CIService758b1bcd7cc8 V 1.0 Waiting For Distribution	Test V 0.1 In Design Check Out	CIService1f2a2c3e004c V 1.0 Waiting For Distribution
VF Certified	VF Certified	VF Certified	VF Certified	PNF In Design Check In		

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
2 tooca_definitions_version: tooca_simple_yaml_1_1
3 imports:
4 - nodes:
5   - file: nodes.yml
6 - datatypes:
7   - file: data.yml
8 - capabilities:
9   - file: capabilities.yml
10 - relationships:
11   - file: relationships.yml
12 - groups:
13   - file: groups.yml
14 - policies:
15   - file: policies.yml
16 - annotations:
17   - file: annotations.yml
18 - node_types:
19   - org.openecomp.resource.pnf.Example:
20     - derived from: org.openecomp.resource.abstract.nodes.PNF
21     - properties:
22       - pnf_function:
23         - type: string
24       - software_versions:
25         - type: List
26         - entry_schema:
27           - type: string
28       - pnf_role:
29         - type: string
30       - pnf_type:
31         - type: string
```

```
Search "dsasp" (3 hits in 1 file)
C:\Users\val636r\AppData\Local\Temp\asp27333\data\logs\BE\SDC\SDC-BE\error.log (3 hits)
Line 28255: 2018-08-21T12:17:49.551Z [main]SDC catalog||ExecuteRestRequest||INFO|0|Starting to consume topic for DMAP consumer with the next parameters DmapConsumerConfiguration [hosts=localhost:3905, consumerGroup=dc, con
Line 28256: 2018-08-21T12:17:49.552Z [main]SDC catalog||ExecuteRestRequest||INFO|0|Starting to consume topic for DMAP consumer with the next parameters DmapConsumerConfiguration [hosts=localhost:3905, consumerGroup=dc, con
Line 28256: 2018-08-21T12:17:49.562Z [main]SDC catalog||ExecuteRestRequest||ERROR|500|An error occurred upon consuming topic by Dmap consumer client.: null
Search "dsasp" (0 hits in 0 files)
```

# software\_versions (SDC Demo)

The dashboard features a top navigation bar with 'HOME', 'CATALOG', 'ONBOARD', 'DCAE-DS', and 'WORKFLOW'. A search bar is located in the top right. The main area contains a grid of service cards. On the left, there are 'ADD' and 'IMPORT' buttons. The cards are organized into two rows. The top row contains five cards with status 'S' (Waiting For Distribution or In Design Check Out). The bottom row contains five cards with status 'R' (Certified or In Design Check In). Each card displays a unique ID, version number, and status.

ID	Version	Status
CIService1e7a73a6893	V 1.0	Waiting For Distribution
CIService182e812e6eb9	V 1.0	Waiting For Distribution
CIService758b1bcd7cc8	V 1.0	Waiting For Distribution
Test	V 0.1	In Design Check Out
CIService1f2a2c3e004c	V 1.0	Waiting For Distribution
CIResVFOnboarded-Vvg-0cdf2638	V 1.0	Certified
CIResVFOnboarded-Base_vvg.F4533387	V 1.0	Certified
CIResVFOnboarded-ZteEpcMmeVf.3bc7dccc6	V 1.0	Certified
CIResVFOnboarded-Resource-ZteEpcMmeVf.Csa-3770081c	V 1.0	Certified
Example	V 0.1	In Design Check In

The 'General' configuration page shows fields for Name, Category, Contact ID, Project Code, Ecomp Generated Naming, Naming Policy, Service Type, Service Role, Environment Context, and Instantiation Type. A 'Save' button is in the top right. The 'Description' field contains the text '1234'. The 'Created' and 'Modified' dates are 08/21/2018, and the 'Author' is Carlos Santana.

**General**

Name: test

Category: Network L4+

Contact ID: cs0008

Project Code: 123214

Ecomp Generated Naming: True

Naming Policy:

Service Type:

Service Role:

Environment Context: General\_Revenue-Bearing

Instantiation Type: A-la-carte

Created: 08/21/2018, Carlos Santana

Modified: 08/21/2018

UUID: 79c65b4b-1ec1-4551-9b23-98bec38b1a36

Invariant UUID:

The 'Composition' page shows a dropdown menu with options: General, TOSCA Artifacts, Composition, Activity Log, Management Workflow, Network Call Flow, Deployment, Properties Assignment, and Monitoring. The 'Properties Assignment' option is highlighted. The main area shows a service card for 'example 0' with status 'R' and version 'V0.1'. The card is in 'IN DESIGN CHECK OUT' state. A 'Submit for Testing' button and a 'Check In' button are visible. The right sidebar shows 'GENERAL INFO' for the service, including Type, Resource Type, Version, Category, Sub Category, Creation Date, Author, Vendor Release, Vendor Model Number, and Contact ID.

ME > SERVICE: test > Composition >

V0.1 IN DESIGN CHECK OUT

Submit for Testing Check In

Service Flows: Hide all

example 0

GENERAL INFO

Type: RESOURCE

Resource Type: PNF

Version: 0.1

Category: Application L4+

Sub Category: Web Server

Creation Date: 08/21/2018

Author: Carlos Santana

Vendor Release: 123

Vendor Model Number:

Contact ID: cs0008

Description: 123

ADDITIONAL INFORMATION

TAGS

example

# software\_versions (SDC Demo)

V0.1 \* IN DESIGN CHECK OUT

Submit for Testing

Check In

## Properties Assignment

Properties		Inputs		Search		Declare	
Property Name	Type	ES	Value	Discard	Save		
<b>R EXAMPLE 0</b>							
<input type="checkbox"/> nf_function	string		<input type="text"/>				
<input type="checkbox"/> software_versions	list	string	<input type="text"/>			Add value to list	
			<input type="text" value="11"/>	<input type="text"/>	<input type="text"/>		
			<input type="text" value="10"/>	<input type="text"/>	<input type="text"/>		
<input type="checkbox"/> nf_role	string		<input type="text"/>				
<input type="checkbox"/> nf_type	string		<input type="text"/>				

**Composition** Property Structure

TEST

- R example 0**

## Properties Assignment

Properties		Inputs		Search	
Property Name	Type	ES	Value		
<b>R EXAMPLE 0</b>					
<input type="checkbox"/> nf_function	string		<input type="text"/>		
<input type="checkbox"/> software_versions	list	string	<input type="text"/>		
			<input type="text" value="11"/>		
			<input type="text" value="10"/>		
<input type="checkbox"/> nf_role	string		<input type="text"/>		
<input type="checkbox"/> nf_type	string		<input type="text"/>		