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

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

Sept 14, 2018 version 6

Modeling ONAP Links

PAGE	LINK
Use case proposal: 5G- RAN deployment, Slicing, SON	https://wiki.onap.org/display/DW/Use+case+proposal%3A+5G-+RAN+deployment%2C+Slicing%2C+SON
5G Functional Requirements Tracking	https://wiki.onap.org/display/DW/5G+Functional+Requirements+Tracking
Casablanca Release Requirements	https://wiki.onap.org/display/DW/Casablanca+Release+Requirements
NFModeling-SDC_R324Jul2018v1	https://wiki.onap.org/display/DW/Casablanca
Service Design & Creation (SDC) Portal Page	https://wiki.onap.org/display/DW/Service+Design+and+Creation+%28SDC%29+Portal
Common about NFs (PNFs & VNFs and allotted NFs Info-model level "base")	https://wiki.onap.org/display/DW/Abstract+Base+IM+Class+for+Network+Function
SDC Artifacts	https://wiki.onap.org/display/DW/SDC%2Bsupported%2Bartifact%2Btypes
Working with SDC	https://wiki.onap.org/display/DW/Working+with+SDC
Micro-Services Onboarding Wiki	**https://wiki.onap.org/display/DW/MicroServices+Onboarding**

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

ONAP Personnel
"Roles" (Operator
Governor
Designer
Administrator
Tester)

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

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)

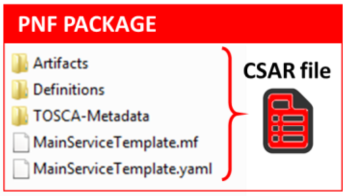
Vendor Provided

PNFD version = 6

PNF-Package (version)

Minimal PNFPackage version = 7.1

(SDC Versions the Package)



PNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

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); *e*xample: 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 structres/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.

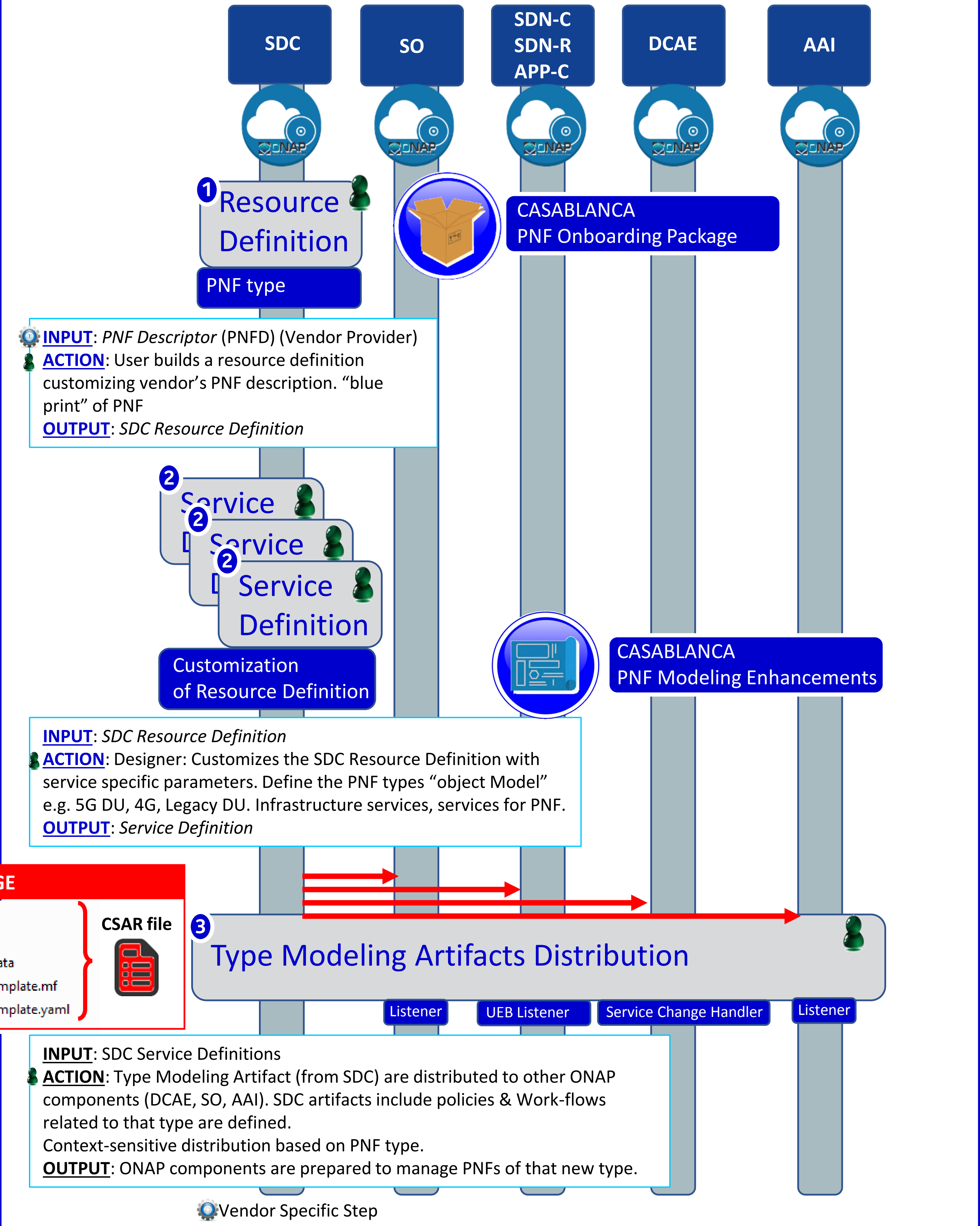
VNF vs PNF Comparison

TOPIC	VNF	PNF
Concept	Application fulfills the role of a network function.	It is a network element, a physical entity, which can implements the role of a network function.
Physical Characteristic	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.
On-boarding	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.
Plug and Play	The model triggers the orchestration.	(See this slide package for PNF Plug and Play) at the end of PnP the PNF can provide service.
Characteristics	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.
Configurability & Deployment	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.
ONAP Interaction	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
Design Time Modeling	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.
Service Orchestration	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.
Resources	VNF dynamically assigned resources.	PNF statically associated (hardware) resources.
Capacity	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)



INPUT: *PNF Descriptor* (PNFD) (Vendor Provider)
ACTION: User builds a resource definition customizing vendor’s PNF description. “blue print” of PNF
OUTPUT: *SDC Resource Definition*

INPUT: *SDC Resource Definition*
ACTION: Designer: Customizes the SDC Resource Definition with service specific parameters. Define the PNF types “object Model” e.g. 5G DU, 4G, Legacy DU. Infrastructure services, services for PNF.
OUTPUT: *Service Definition*

INPUT: SDC Service Definitions
ACTION: Type Modeling Artifact (from SDC) are distributed to other ONAP components (DCAE, SO, AAI). SDC artifacts include policies & Work-flows related to that type are defined.
Context-sensitive distribution based on PNF type.
OUTPUT: ONAP components are prepared to manage PNFs of that new type.

Vendor Specific Step

STEP	DESCRIPTION
1	**RESOURCE DECLARATION** – 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 **PNF RESOURCE Definition** **Resource Type** – Type of Resource. NEW type: PNF (pre-defined in SDC) **NAME** – Name of the PNF type **CATEGORY** – e.g. Infrastructure **TAGS** – User-definable tags (default name of the PNF) **DESCRIPTION** – Textual description **CONTACT ID** – Designer (user of ONAP) **VENDOR** – PNF Vendor (e.g. Nokia) **VENDOR RELEASE** – Vendor release **VENDOR MODEL NUMBER** – PNF Model value (link to A&AI) **EVENTS** – Monitoring Event definitions. Define design-time templates. CLAMP (runtime monitoring), DCAD (design time design template attach to VNF). Define templates & attach them. 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.
2	**SERVICE Definition (uses a PNF)** **NAME** – Name of the Service (mandatory) **CATEGORY** – e.g. Network L1…L4, VOIP call Control, Mobility **TAGS** – User-definable tags (default name of the PNF) **DESCRIPTION** – Textual description of service (mandatory) **CONTACT ID** – Designer (user of ONAP) (mandatory) **PROJECT CODE** – ID (mandatory) **Ecomp-Generated Naming** – Name **Naming Policy** – Policy to be used to assign a name to a service by SO/SDNC **SERVICE TYPE** – Type of service **SERVICE ROLE** – The Role of this service. **ENVIRONMENTAL CONTEXT** – distributed environments **Specific Service**(?) – PNF, allotted resource from a CU Service 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.
3	**DISTRIBUTION** – Event Monitoring Templates distributed. (?)

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
SDC/ Modeling	Modeling Project - (No License management impact – See **Futures** Section) [See follow-on Slides "MODELING"]
VNF-SDK (PNF-SDK) Validation	**PNF PACKAGE DEFINITION** 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. **VALIDATION OF A SDK PACKAGE** VNF-SDK (validation, Package definition, verification tool) – package compliant. Allows creation/validation of packages. PNF-SDK *validates* the package.
WORK FLOW (SDC)	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.
MONITORING (SDC)	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:

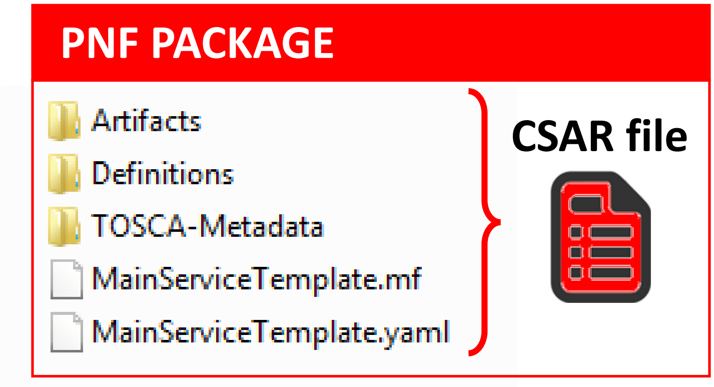
1) **EXTERNALS** - Not trying to model the internals of PNFs. What is exposed by the box is what is modeled.
2) **INTERRELATIONS** - Focus on relations of PNFs/VNFs. Interworking between PNFs/VNFs.
3) **VISIBILITY** - CP/UP visibility
Not M-Plane (as this is 3GPP standardized)
4) **MODELING ANALYSIS** - 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 Dictionary Index	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.



PNF PACKAGE

Artifacts
Definitions
TOSCA-Metadata
MainServiceTemplate.mf
MainServiceTemplate.yaml

CSAR file

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.

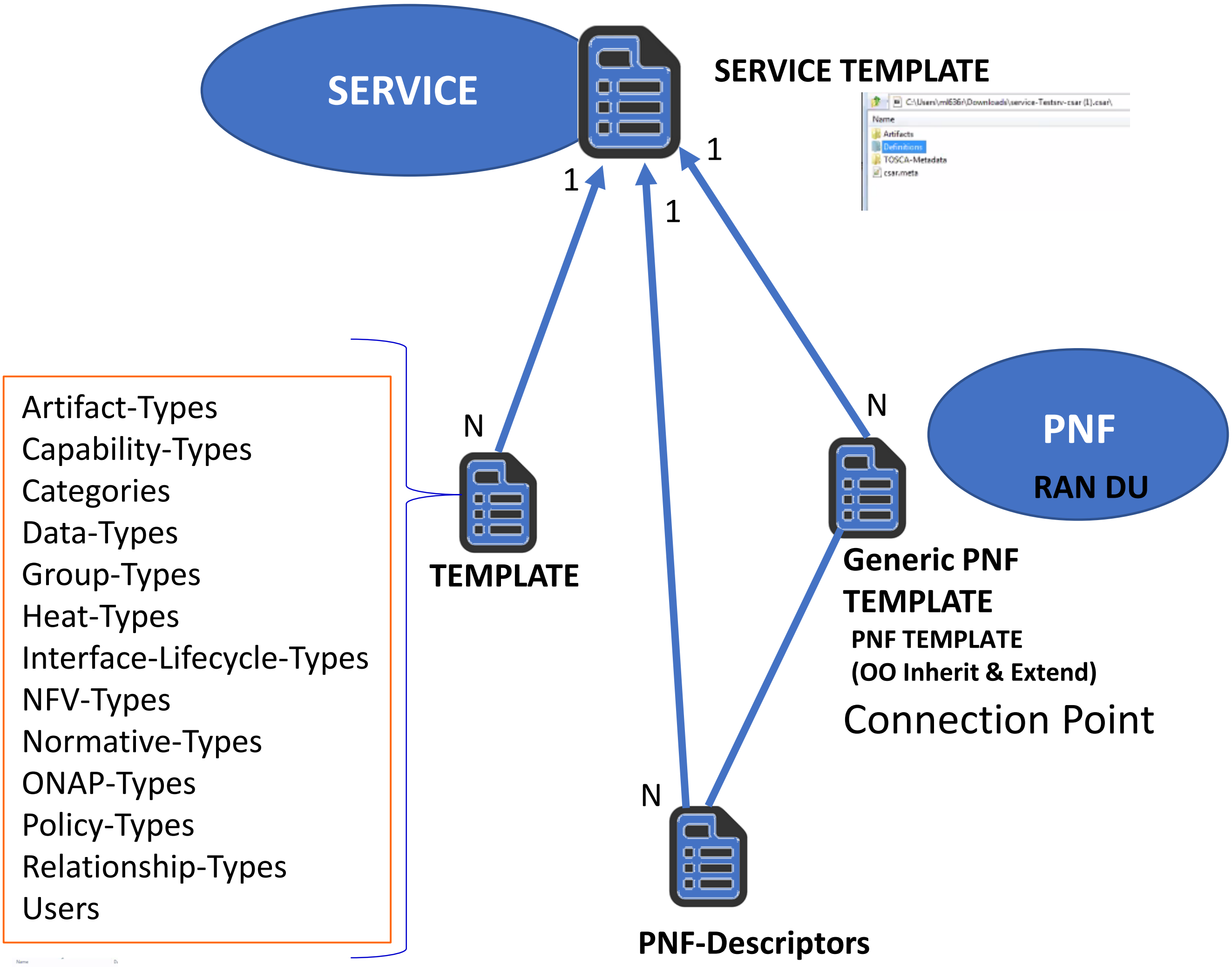
ARTIFACTS

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

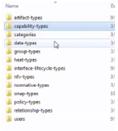
SDC MODELING (Design Time) – Casa R3



C:\Users\ml636r\Downloads\service-Testsrv-csar (1).csar\

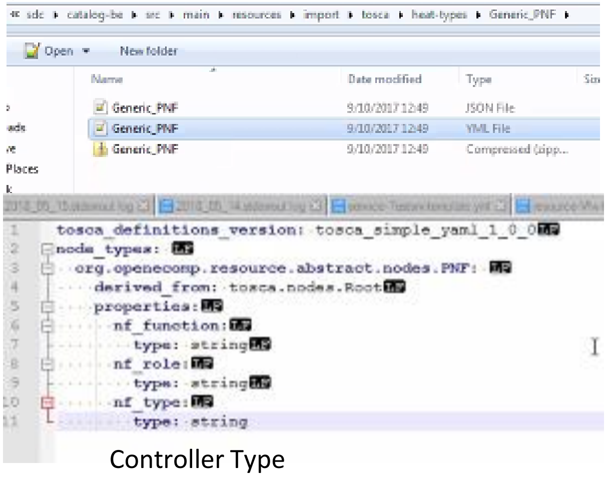
Name
Artifacts
Definitions
TOSCA-Metadata
csar.meta

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



Name	D
artifact-types	9/
capability-types	3/
categories	3/
data-types	3/
group-types	3/
heat-types	3/
interface-lifecycle-types	9/
nfv-types	3/
normative-types	3/
onap-types	10
policy-types	3/
relationship-types	3/
users	9/

PNF-Descriptors
Vendor Specific
Template



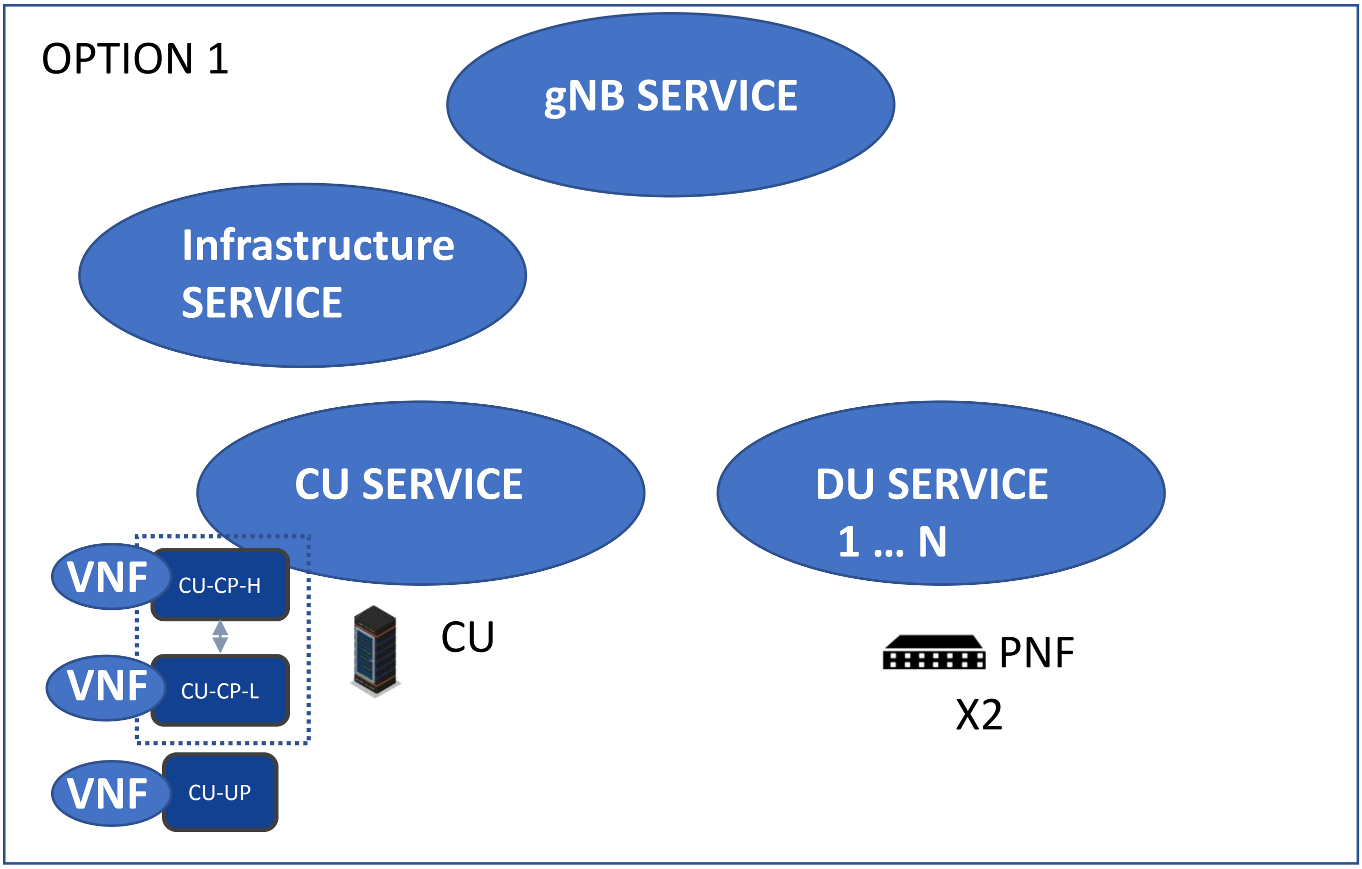
« sdc ▸ catalog-be ▸ src ▸ main ▸ resources ▸ import ▸ tosca ▸ heat-types ▸ Generic_PNF ▸

Name	Date modified	Type
Generic_PNF	9/10/2017 12:49	JSON File
Generic_PNF	9/10/2017 12:49	YML File
Generic_PNF	9/10/2017 12:49	Compressed (zipp...

```
tosca_definitions_version: tosca_simple_yaml_1_0_0
node_types:
  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

gNB SERVICE

Infrastructure SERVICE

CU SERVICE

DU SERVICE 1 … N

VNF CU-CP-H

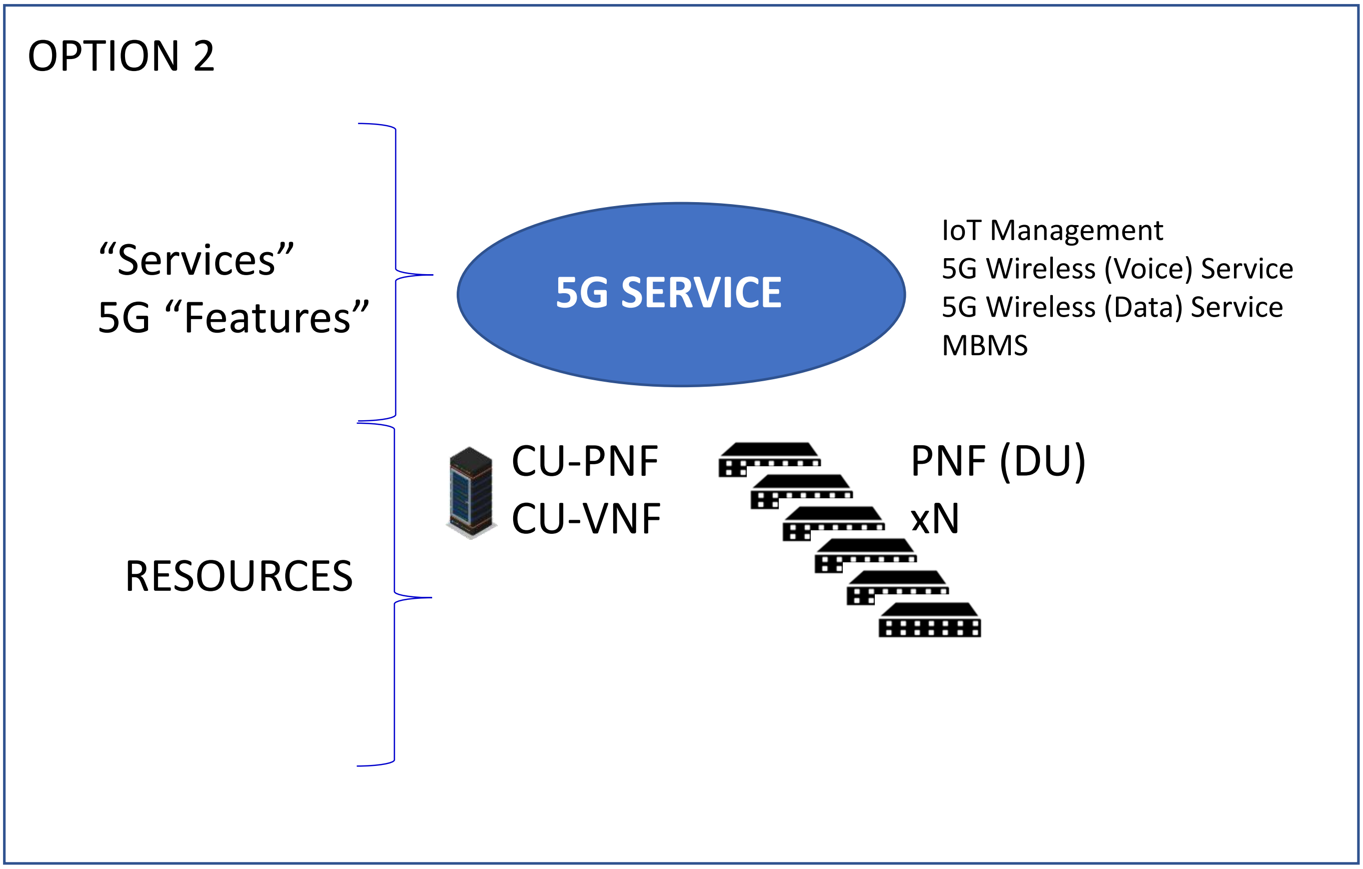
VNF CU-CP-L

VNF CU-UP

CU

PNF

X2



OPTION 2

"Services"
5G "Features"

5G SERVICE

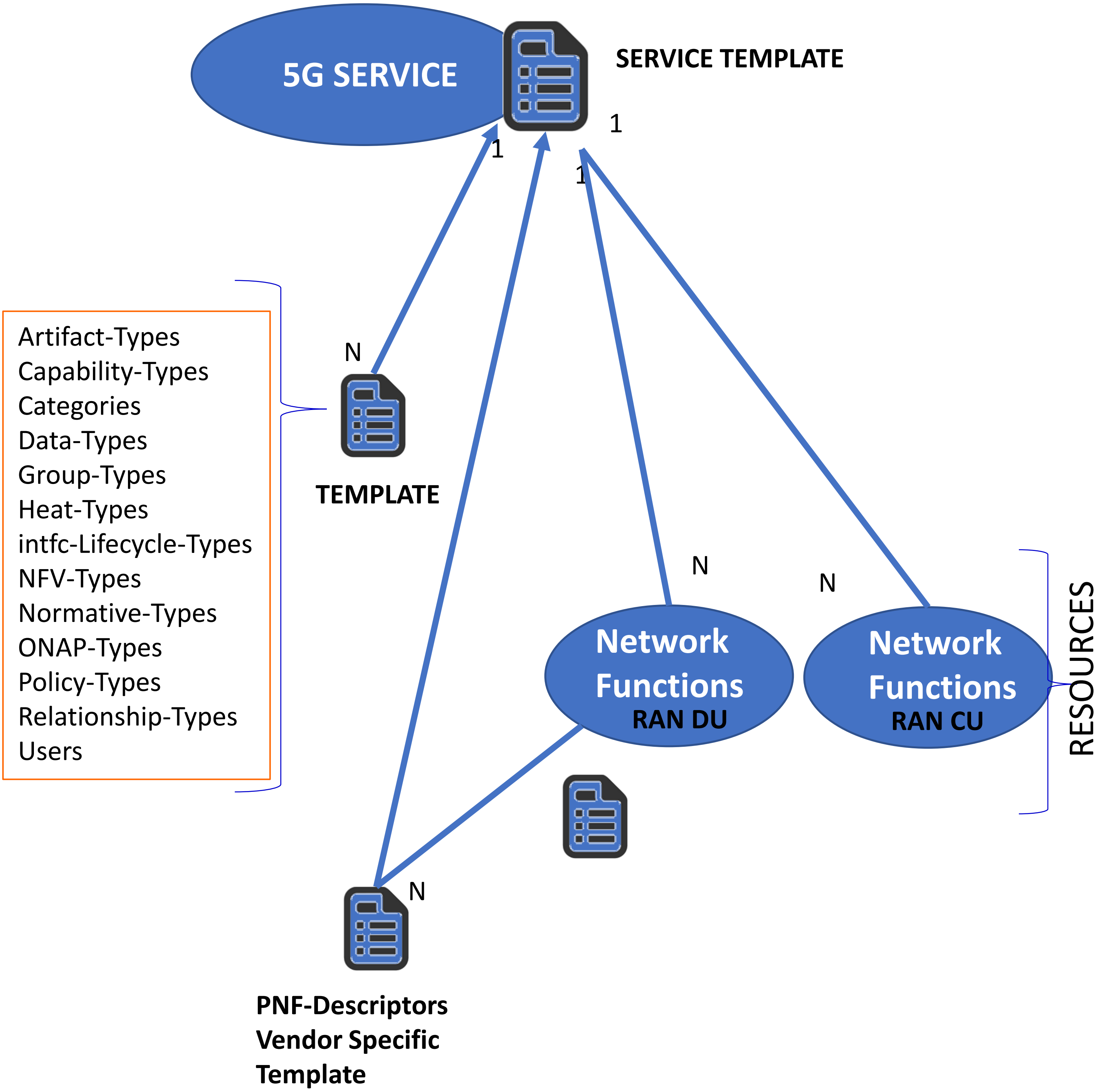
IoT Management
5G Wireless (Voice) Service
5G Wireless (Data) Service
MBMS

RESOURCES

CU-PNF
CU-VNF

PNF (DU)
xN

SDC MODELING (Design Time) – Casa R3



5G SERVICE

SERVICE TEMPLATE

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

TEMPLATE

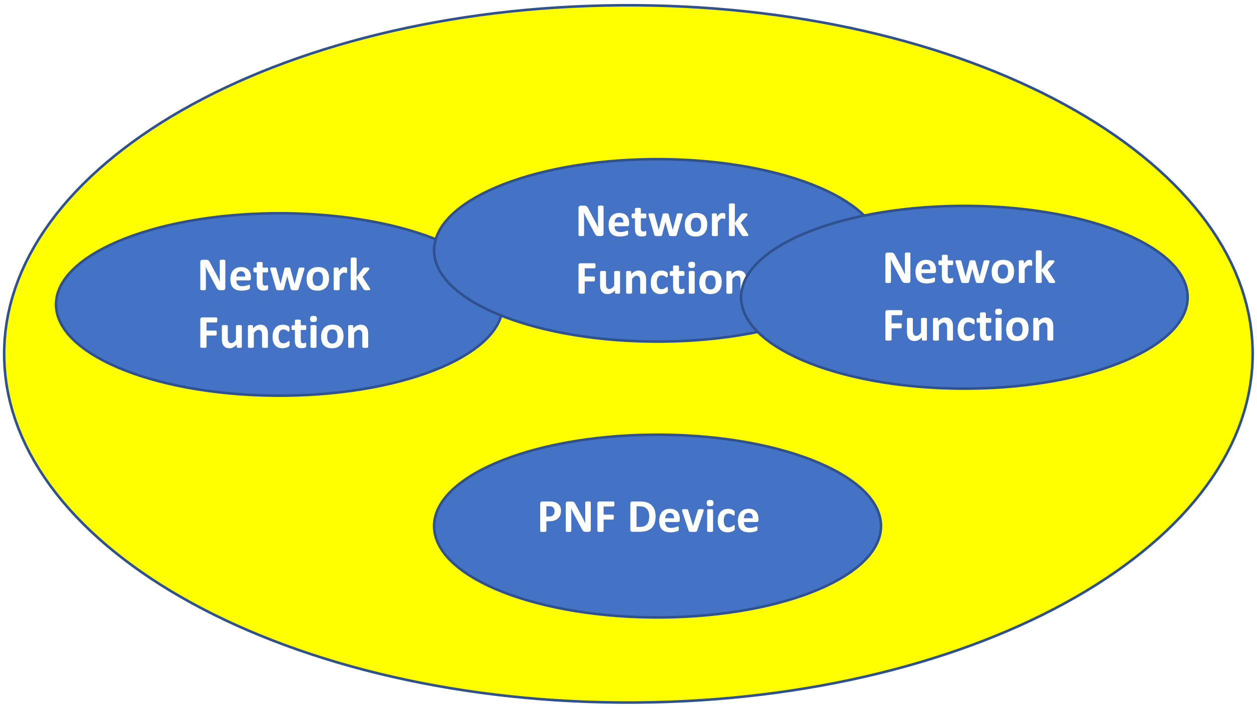
Network Functions
RAN DU

Network Functions
RAN CU

RESOURCES

PNF-Descriptors
Vendor Specific
Template

PNF DEVICE Node Type



PNF device located physically at a location
Network Function is a resource utilitizing PNFs or VNFs
Multiple NF w/ Licenses
SERVICES (UML) “has a” 1…N PNFs & VNFs.

(conceptually) What is the difference between a NF & a Service?
NF supports the creation of a service.
NF = RESOURCE

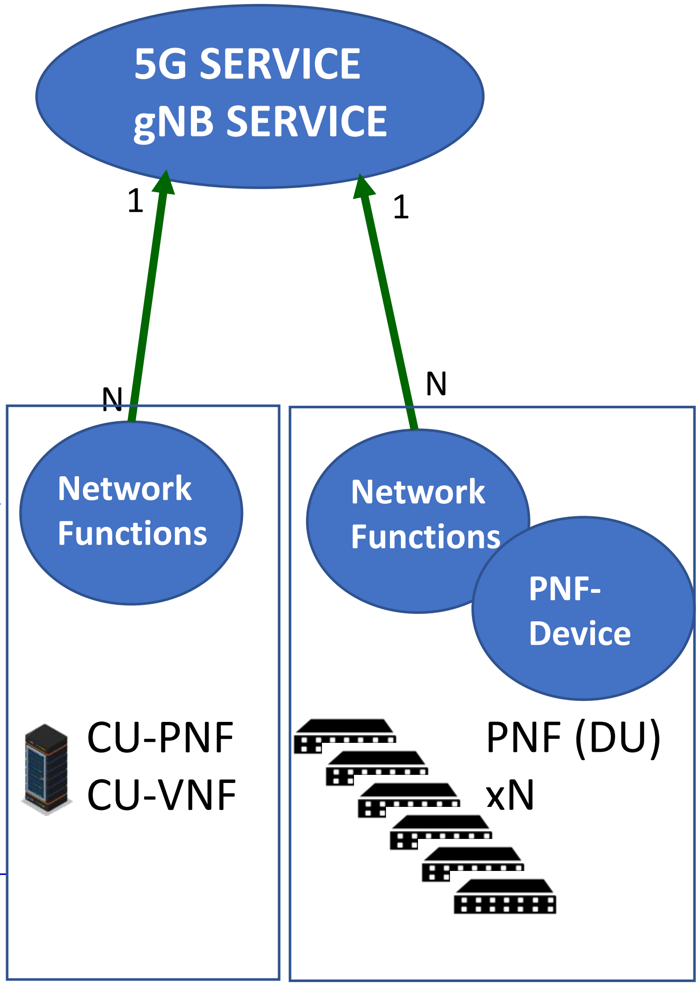
gNB Model (R4)

“Services”
5G “Features”

IoT Management
5G Wireless (Voice) Service
5G Wireless (Data) Service
MBMS xyz-features
(Licensing)
Infrastructure Service

Meta-
RESOURCES

Concrete-
RESOURCES
Catalog



5G SERVICE
gNB SERVICE

1

1

N

N

Network
Functions

Network
Functions

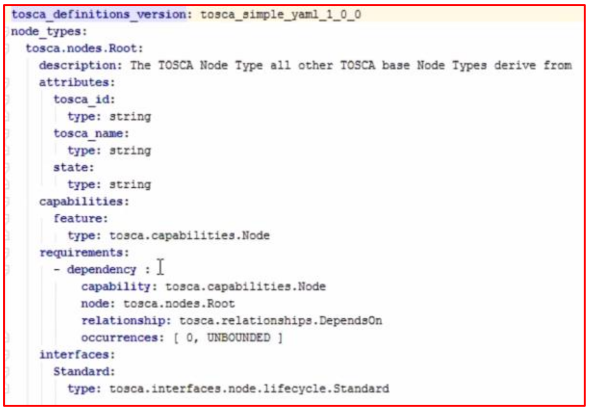
PNF-
Device

CU-PNF
CU-VNF

PNF (DU)
xN

During orchestration time
Find CU/DU-PNF resources catalog
Then Homing function
NF Global types derived DU/CU-NF
Implementations of CU/DUs

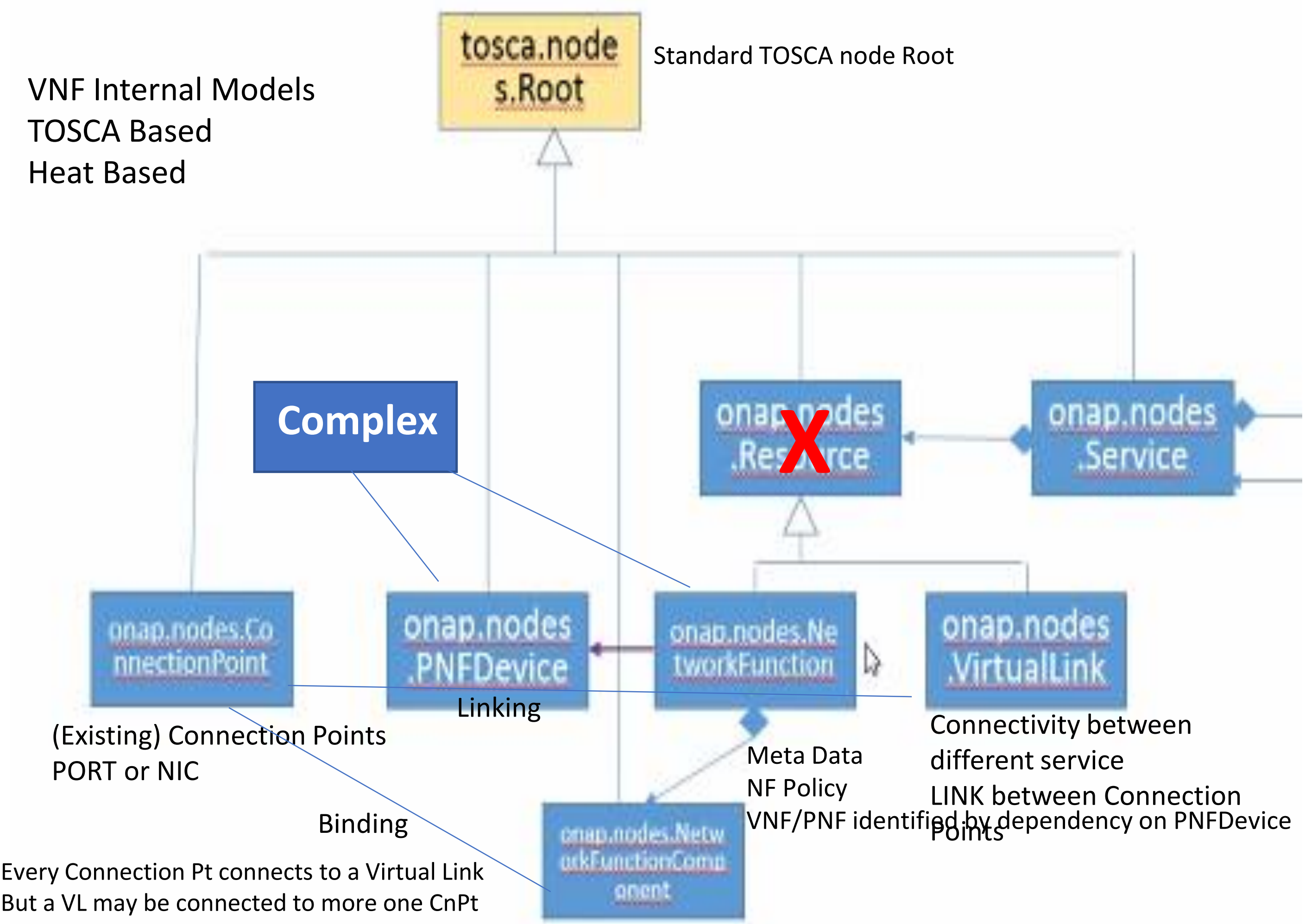
Associating CU-instances w/ specific [set of] DU instances



```
tosca_definitions_version: tosca_simple_yaml_1_0_0
node_types:
  tosca.nodes.Root:
    description: The TOSCA Node Type all other TOSCA base Node Types derive from
    attributes:
      tosca_id:
        type: string
      tosca_name:
        type: string
      state:
        type: string
    capabilities:
      feature:
        type: tosca.capabilities.Node
    requirements:
      - dependency :
          capability: tosca.capabilities.Node
          node: tosca.nodes.Root
          relationship: tosca.relationships.DependsOn
          occurrences: [ 0, UNBOUNDED ]
    interfaces:
      Standard:
        type: tosca.interfaces.node.lifecycle.Standard
```

Tosca.nodes.Root

DATA MODEL



Standard TOSCA node Root

VNF Internal Models
TOSCA Based
Heat Based

Linking

(Existing) Connection Points
PORT or NIC

Meta Data
NF Policy
VNF/PNF identified by dependency on PNFDevice

Connectivity between
different service
LINK between Connection
Points

Binding

Every Connection Pt connects to a Virtual Link
But a VL may be connected to more one CnPt
TRUNK MODE
10 VNICs off a NIC each VNIC is a CnPt
and each NIC is a CnPT
DOMAIN, CONTEXT Martin Fowler
VM ComputeHost has NICs

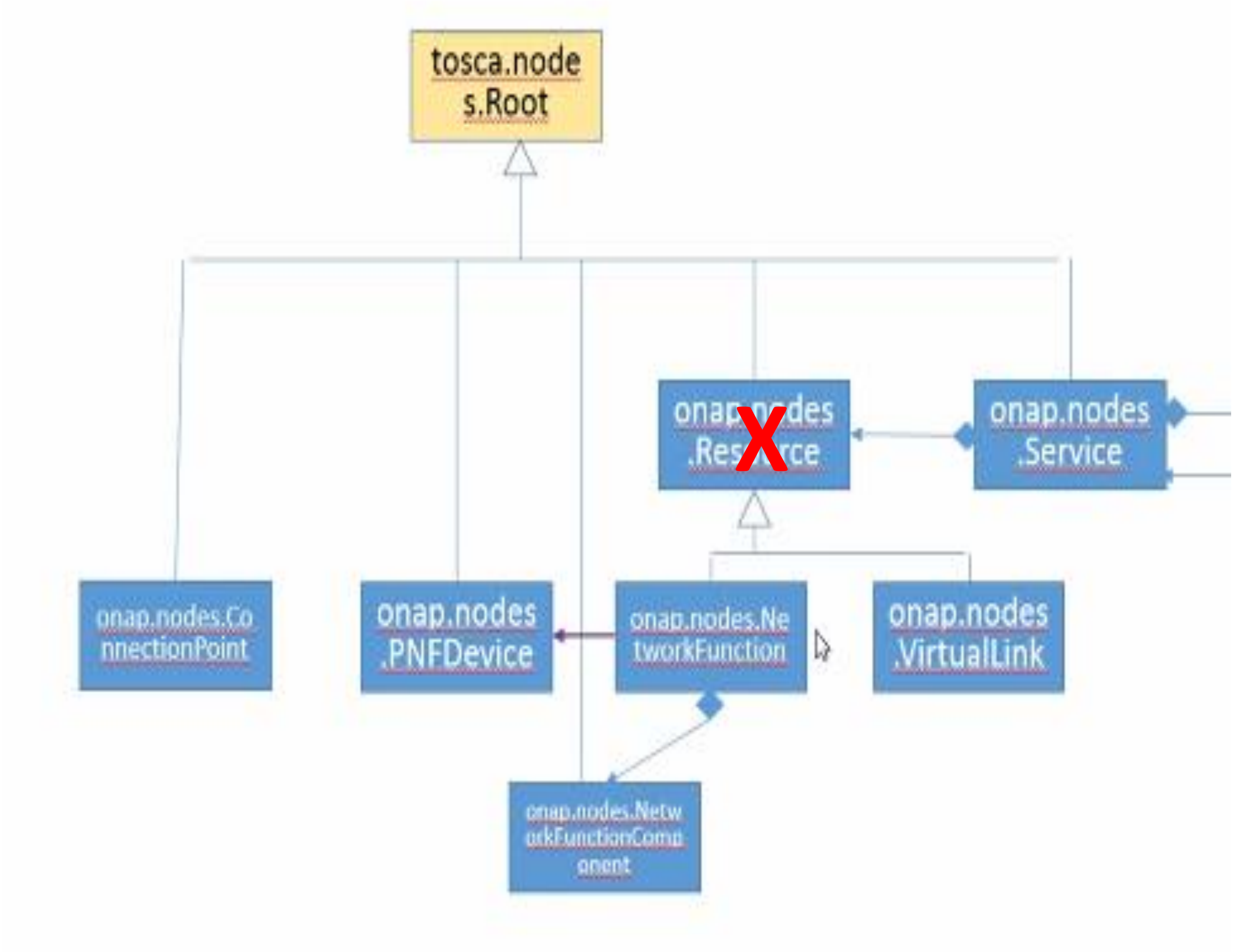
Component
e.g. Transcoder

SOFTWARE ACCELERATION

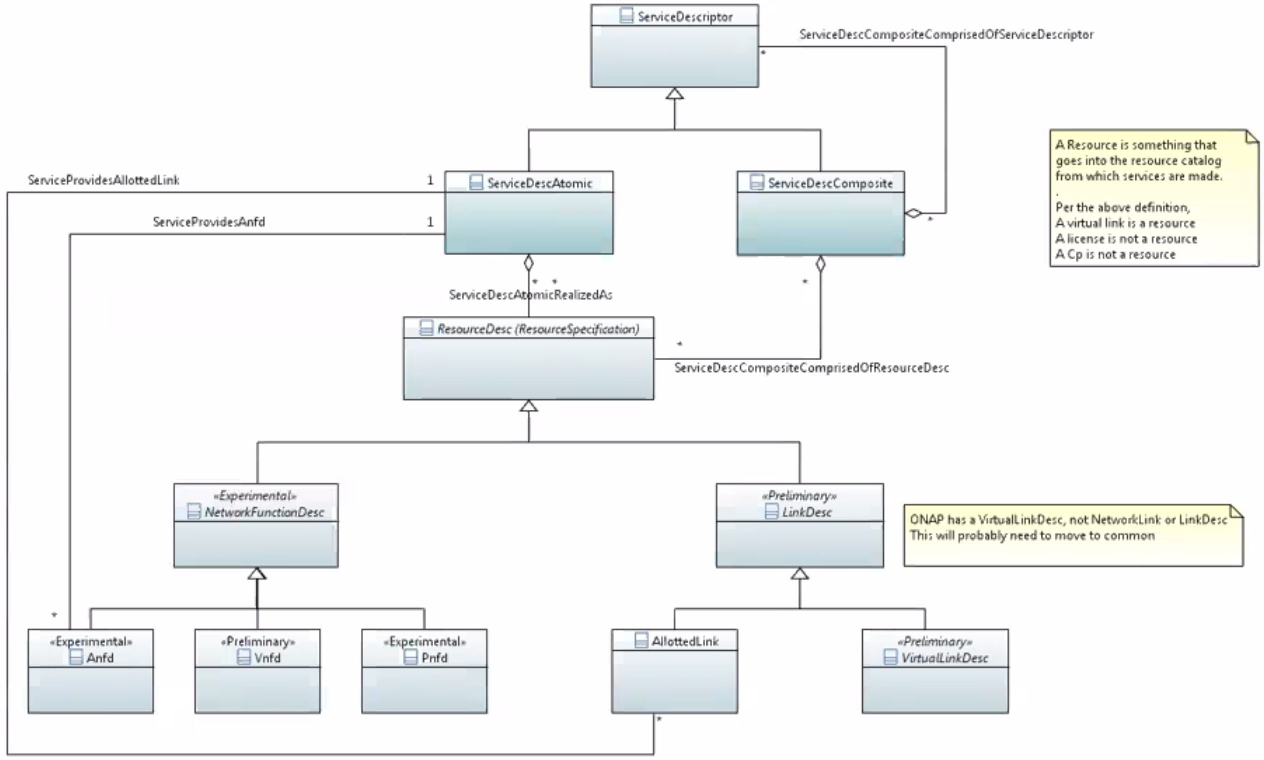
DPDK, SW Pinning. EPA (enhanced platform awareness)
Transcoder(s) is a NF w/ SBC [session boarded controller] based on through-put.
Session boarded controller DIAMETER, ROUTER
Typically done in H/W. Specialized NIC (bearer traffic) for Higher Bandwidth
could be done in S/W doesn’t make sense to do in S/W. Purpose-built H/W. Intel Telco FPGAs/ASIC.
session board controller, is a network device. [Fred Feisullin VZW]

https://wiki.onap.org/display/DW/Enhanced+Platform+Awareness+Capability

Anatoly Katzman (Monday 10AM data modeling)


tosca.nodes.Root
onap.nodes.Resource
X
onap.nodes.Service
onap.nodes.ConnectionPoint
onap.nodes.PNFDevice
onap.nodes.NetworkFunction
onap.nodes.VirtualLink
onap.nodes.NetworkFunctionComponent

Allotted Resource



A Resource is something that goes into the resource catalog from which services are made.
.
Per the above definition,
A virtual link is a resource
A license is not a resource
A Cp is not a resource

ONAP has a VirtualLinkDesc, not NetworkLink or LinkDesc
This will probably need to move to common

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)

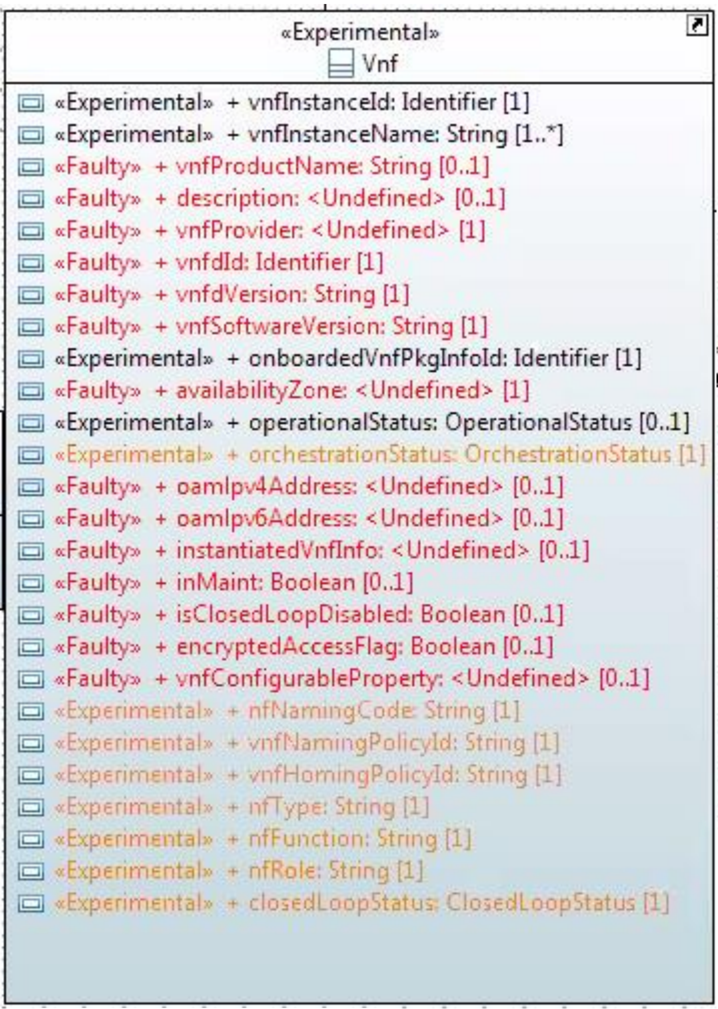
Basic Content of PNF template PNF-D (DESCRIPTOR)

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

VNF Information Model

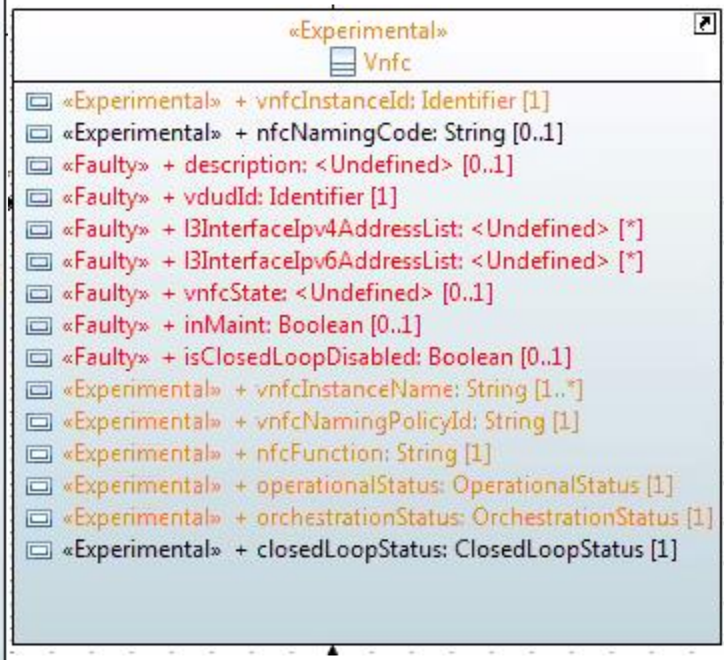
Red items are proposed to be deleted.

gold indicates an add or a change



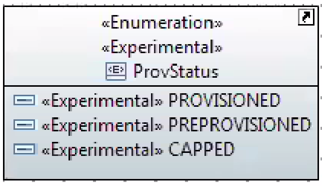
«Experimental»
Vnf

«Experimental» + vnfInstanceId: Identifier [1]
«Experimental» + vnfInstanceName: String [1..*]
«Faulty» + vnfProductName: String [0..1]
«Faulty» + description: <Undefined> [0..1]
«Faulty» + vnfProvider: <Undefined> [1]
«Faulty» + vnfdId: Identifier [1]
«Faulty» + vnfdVersion: 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]



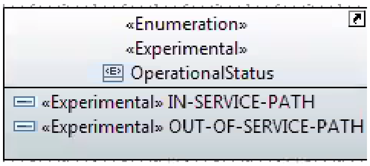
«Experimental»
Vnfc

«Experimental» + vnfcInstanceId: Identifier [1]
«Experimental» + nfcNamingCode: String [0..1]
«Faulty» + description: <Undefined> [0..1]
«Faulty» + vduId: Identifier [1]
«Faulty» + l3InterfaceIpv4AddressList: <Undefined> [*]
«Faulty» + l3InterfaceIpv6AddressList: <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]



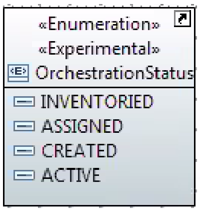
«Enumeration»
«Experimental»
ProvStatus

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



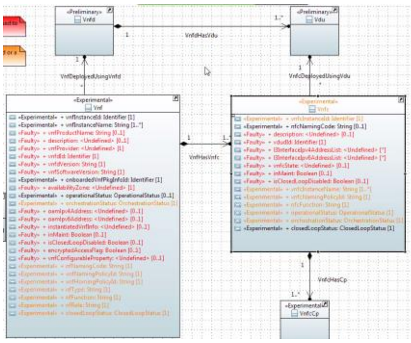
«Enumeration»
«Experimental»
OperationalStatus

«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
pnfId*	Identifier of this Pnf information element. CORRELATIONID (A&AI). ACTION: Discuss further
pnfType (template)*	Type of Resource. NEW type: PNF (pre-defined in SDC)
Category*	PNF category, e.g. infrastructure
Vendor (template)*	Identifies the vendor of the PNF. MANDATORY
Name*	Provides the human readable name of the PNF.
vendorrelease *	Vendor release. MANDATORY
vendormodelNumber*	PNF Model value (link to A&AI)
functionDescription*	Describes the PNF function
software_versions	The **EXPECTED** 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? (For Michael Lando) (Tues) “Merge” every once in a while to Sync up. Contact: Deng Hui

*Already supported in Beijing

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

TOPIC: SOFTWARE_VERSIONS (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

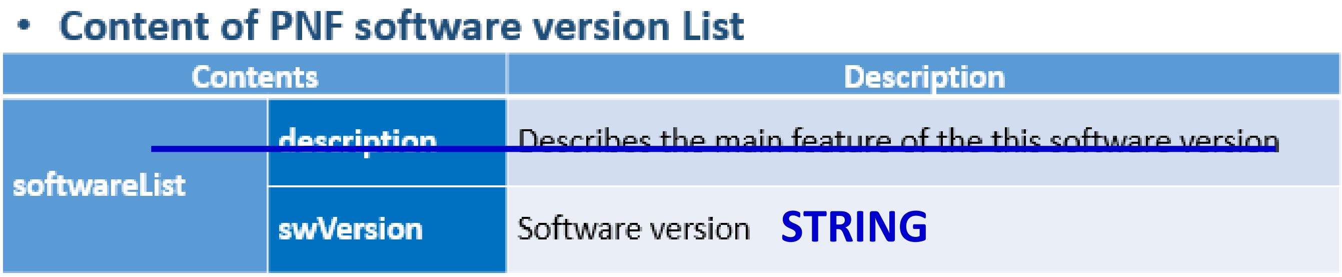
Discussion

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**

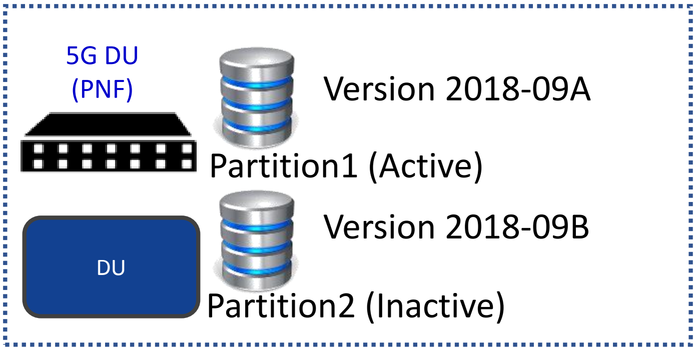
Contents		Description
softwareList	~~description~~	~~Describes the main feature of the this software version~~
	swVersion	Software version **STRING**

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



5G DU (PNF)

DU

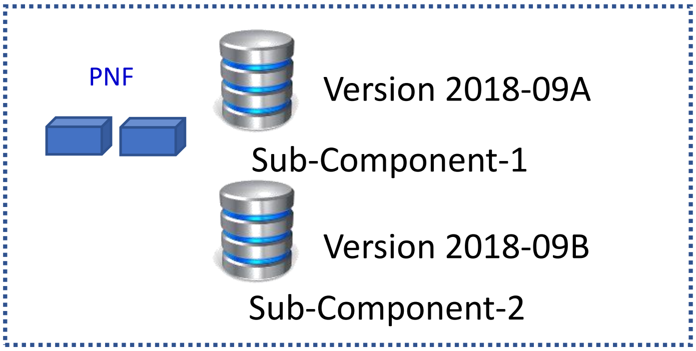
Version 2018-09A

Partition1 (Active)

Version 2018-09B

Partition2 (Inactive)

SUBCOMPONENTS (R4+)



PNF

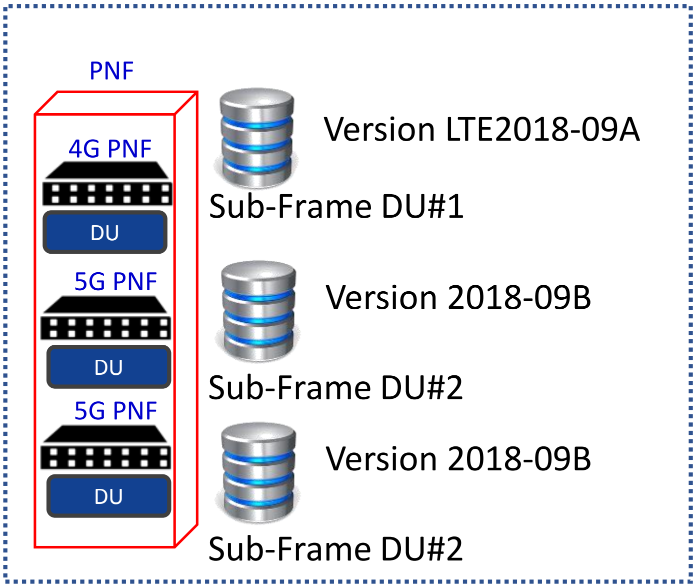
Version 2018-09A

Sub-Component-1

Version 2018-09B

Sub-Component-2

TANDEM CHASSIS CONFIGURATIONS



PNF

4G PNF

DU

5G PNF

DU

5G PNF

DU

Version LTE2018-09A

Sub-Frame DU#1

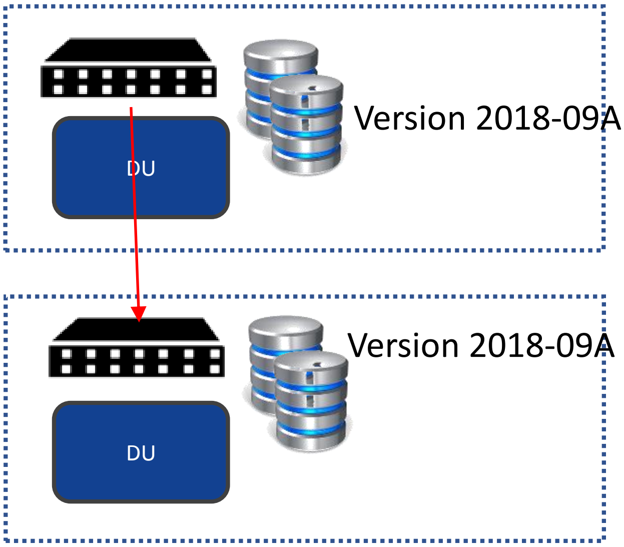
Version 2018-09B

Sub-Frame DU#2

Version 2018-09B

Sub-Frame DU#2

MULTI-PNF DAISY CHAIN DEPLOYMENT



DU

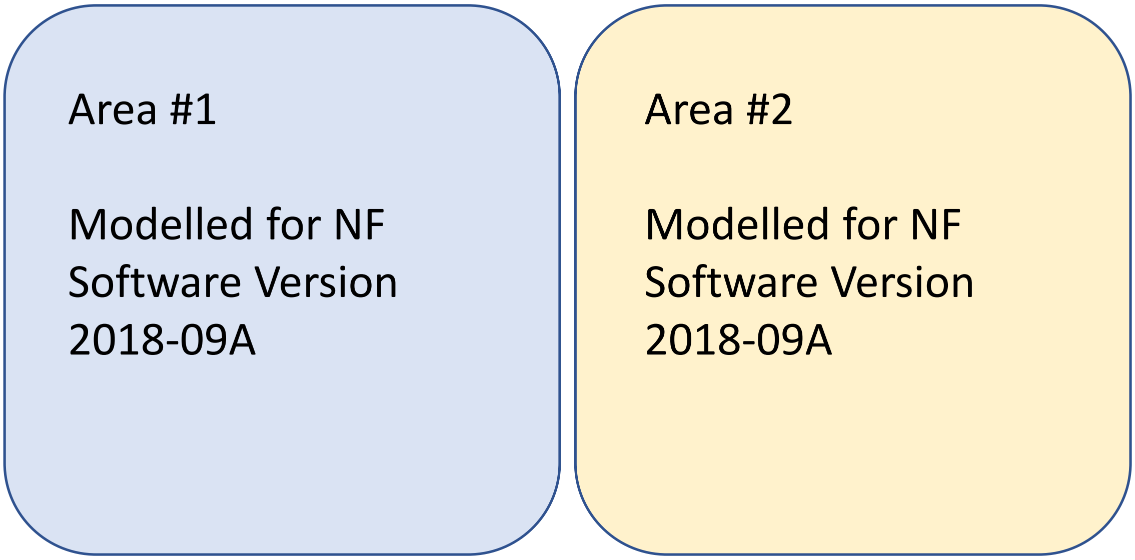
Version 2018-09A

DU

Version 2018-09A

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

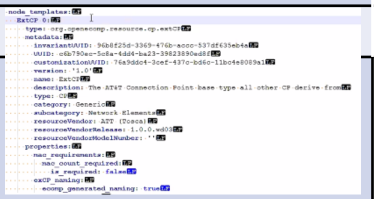
Modelled for NF Software Version 2018-09A

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
pnfId*	Identifier of this Pnf information element. CORRELATIONID (A&AI). ACTION: Discuss further
pnfType (template)*	Type of Resource. NEW type: PNF (pre-defined in SDC)
Category*	PNF category, e.g. infrastructure
Vendor (template)*	Identifies the vendor of the PNF. MANDATORY
Name*	Provides the human readable name of the PNF.
vendorrelease *	Vendor release. MANDATORY
vendormodelNumber*	PNF Model value (link to A&AI)
functionDescription*	Describes the PNF function
software_versions (opt)	The **EXPECTED** software to be supported by the PNF. (see TOPIC: SWVersionList) Introduced in R3/Casas
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.** 
contactId (metadata)	Designer (user of ONAP) ML: Need for this, Audit/tracking, User creates audit log not associated with model itself. 19-Jul – REMOVE THIS PARAMETER
PackageVersion	The version of the PNF Package. 19-Jul – Not going to onboard the PNF, model it from scratch. Define in SDC.
NF Controller	Controller for PNF (APP-C, SDN-R, SDN-C, VF-C)
Technology Domain	Technology Domain

```
node_templates:
  ExtCP 0:
    type: org.openecomp.resource.cp.extCP
    metadata:
      invariantUUID: 96b8f25d-3369-476b-accc-537df635eb4a
      UUID: c6b790ec-5c8a-4dd4-ba23-39823890ed8f
      customizationUUID: 76a9ddc4-3cef-437c-bd6c-11bc4e8089a1
      version: '1.0'
      name: ExtCP
      description: The AT&T Connection Point base type all other CP derive from
      type: CP
      category: Generic
      subcategory: Network Elements
      resourceVendor: ATT (Tosca)
      resourceVendorRelease: 1.0.0.wd03
      resourceVendorModelNumber: ''
    properties:
      mac_requirements:
        mac_count_required:
          is_required: false
      exCP_naming:
        ecomp_generated_naming: true
```

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

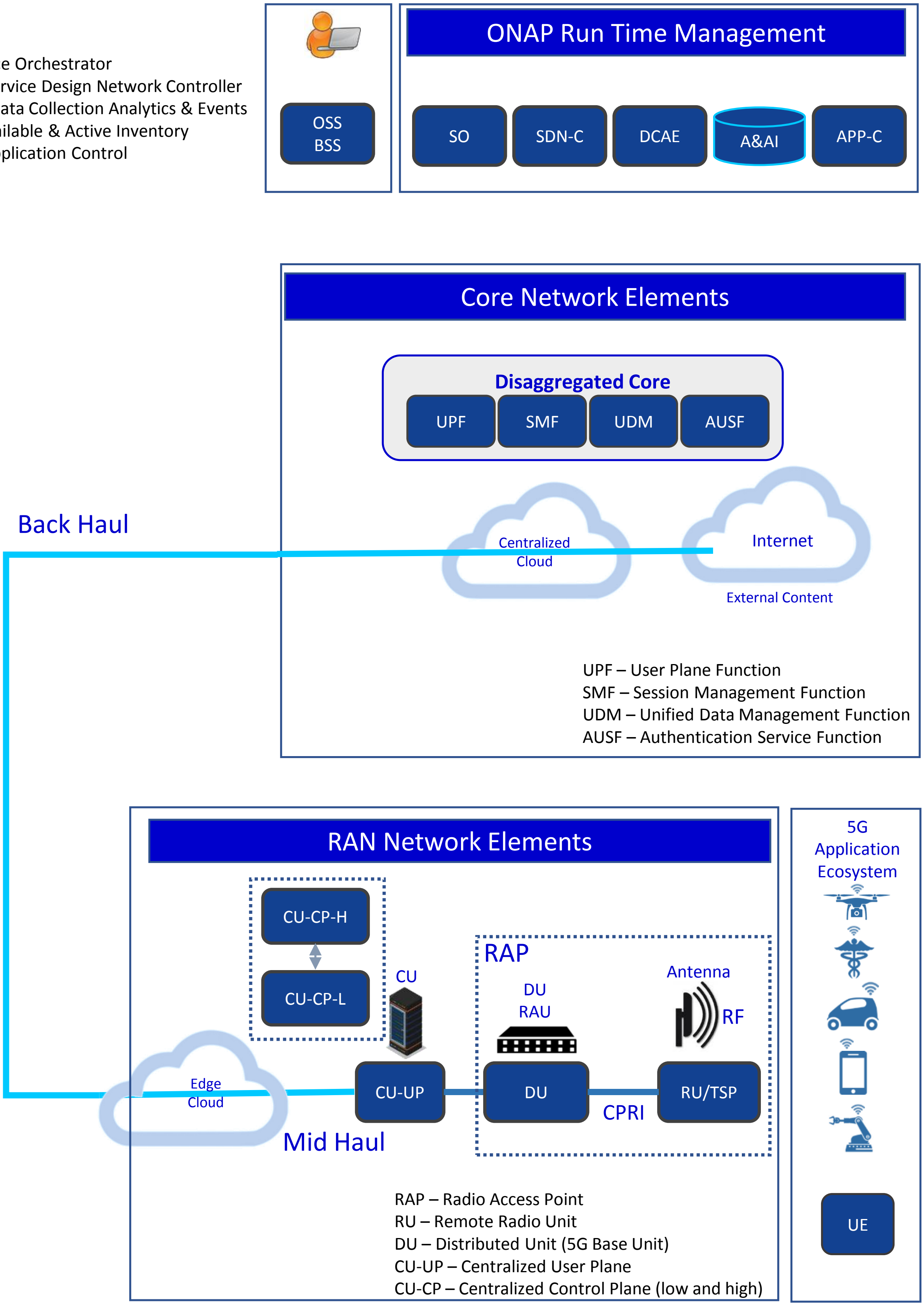
PnP FLOW (updated Wiki)
[Added Note & PNP-1310] in Wiki
https://wiki.onap.org/display/DW/5G+-+PNF+Plug+and+Play

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

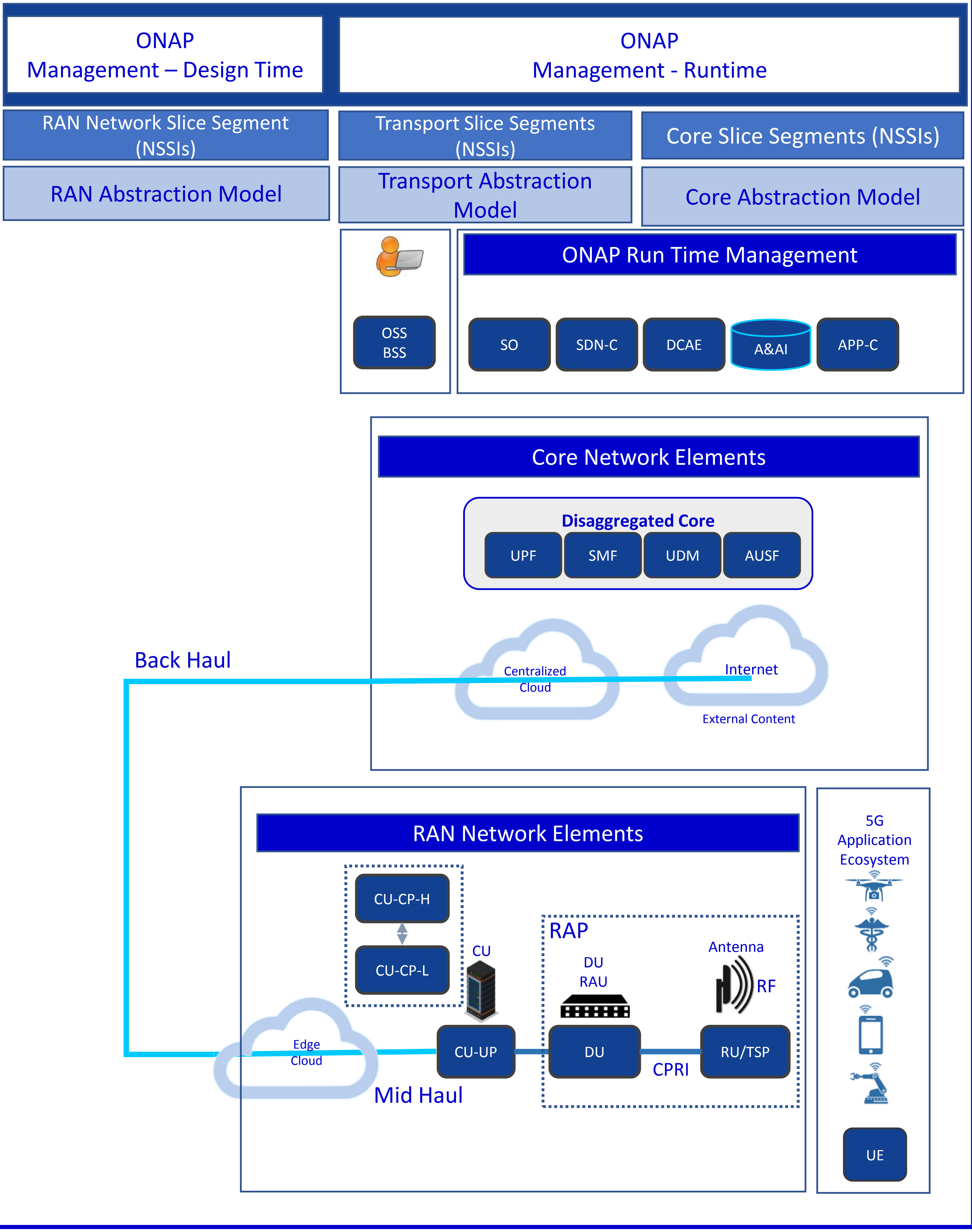


SO – Service Orchestrator
SDN-C – Service Design Network Controller
DCA&E – Data Collection Analytics & Events
A&AI – Available & Active Inventory
APP-C – Application Control

UPF – User Plane Function
SMF – Session Management Function
UDM – Unified Data Management Function
AUSF – Authentication Service Function

RAP – Radio Access Point
RU – Remote Radio Unit
DU – Distributed Unit (5G Base Unit)
CU-UP – Centralized User Plane
CU-CP – Centralized Control Plane (low and high)

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

OSS
BSS

ONAP Run Time Management

SO

SDN-C

DCAE

A&AI

APP-C

Core Network Elements

Disaggregated Core

UPF

SMF

UDM

AUSF

Back Haul

Centralized Cloud

Internet

External Content

RAN Network Elements

CU-CP-H

CU-CP-L

CU

RAP

DU
RAU

Antenna

RF

Edge Cloud

CU-UP

DU

RU/TSP

CPRI

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

RESULT

e.g. SO knows which API to use for NF controller
LCM policy engine, DCAE, Change management, S/W Download

NF

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

SOLUTION (R3 Casablanca)

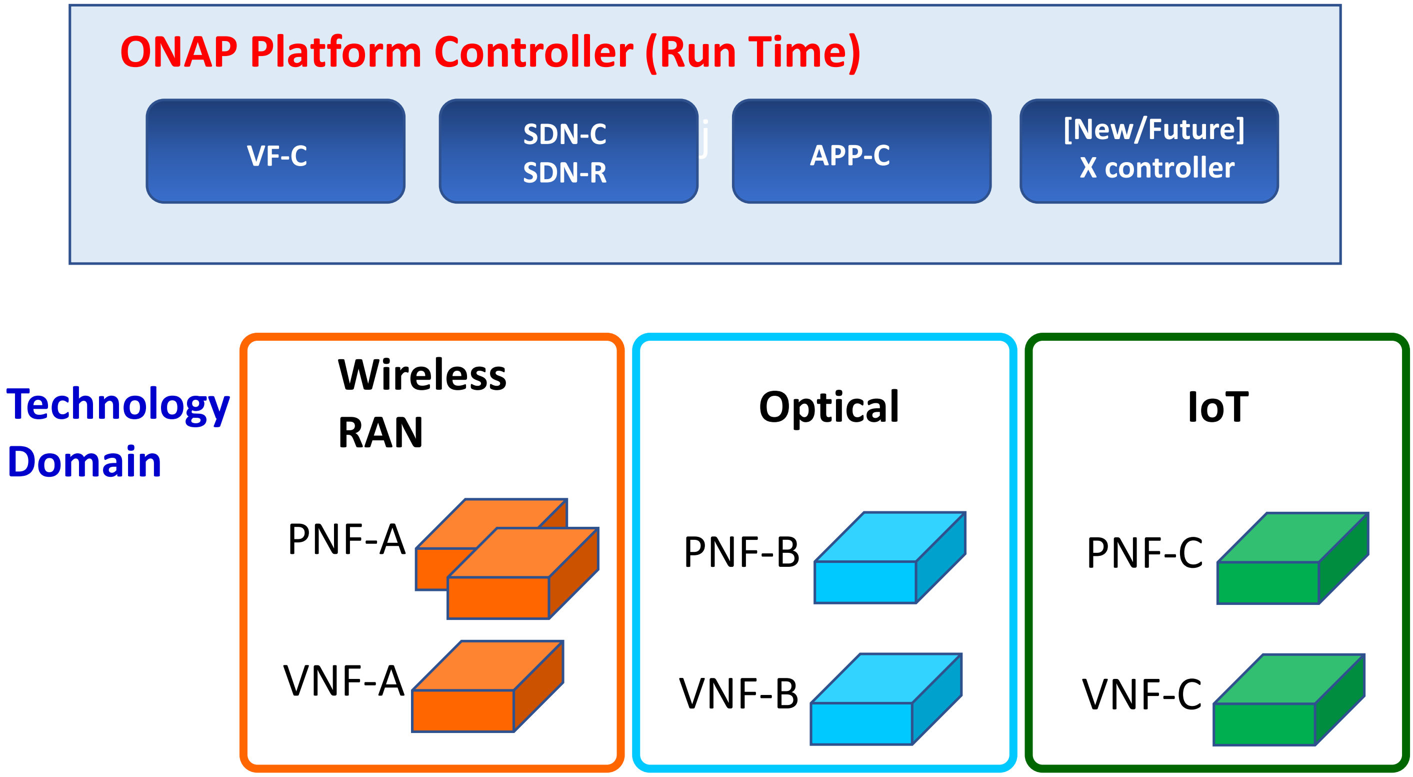
SDN-C, Hard-Code controller to PNF.
"*Hard code*" Generic-API (PNF Plug and Play UC)

OBJECTIVES (Long-Term Goal)

As automated as possible
Using discovery if possible
Flexible operator could design PersonaA for PNF1, PersonaB for PNF2
Take into consideration the capability & functions of the Controller
In general, would not want different controllers handling same NF instance.

NF Controller Concepts

ONAP Deployment



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)

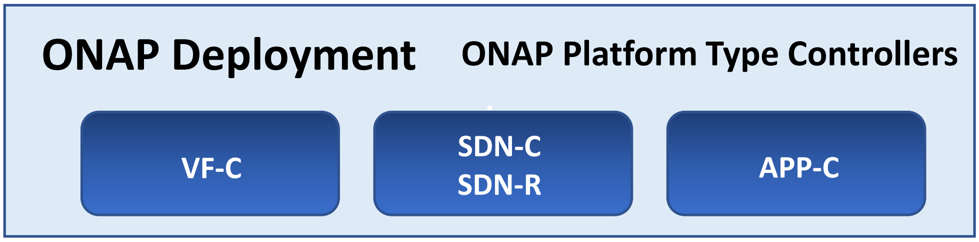
NF Controller Concepts

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.



ONAP Deployment

ONAP Platform Type Controllers

VF-C

SDN-C
SDN-R

APP-C

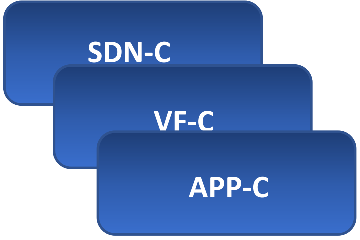
Regional CONTROLLER (Instances)

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

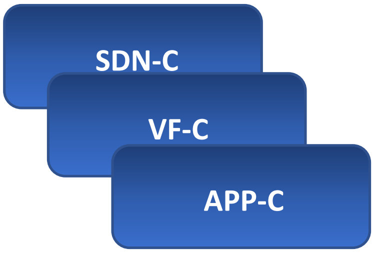


SDN-C

VF-C

APP-C

ONAP Deployment #2



SDN-C

VF-C

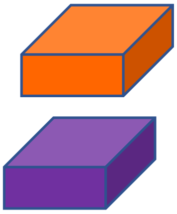
APP-C

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



PNF

External Controller,
EMS, NMS

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 3rd 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
Meta-Data of the Table (combined w/ idea #3)
Run-time Policy (to take up additions)

PROPOSAL #3

Dynamic Association (Table-Driven Approach)

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.

NOTES/IDEAS
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.

Dynamic Association Approach

Design Time / Model Level

Tech Domain	NF (PNF/VNF)	ONAP Platform Controller	Function (API)
Wireless	E// 5G DU	SDN-C	SDN-C Generic API
Wireless	Nokia 5G DU	SDN-C	SDN-C Generic API
Wireless	Huawei 5G DU	SDN-C	SDN-C Generic API
Wireless (Vendor xyz)	Xyz 5G DU	SDN-C	SDN-C
Wireless Subdomain 1	Xyz 5G DU	VF-C	VF-C
Wireline	Wireline PNF	APP-C	APP-C Assign > APP-C Modify Config Restart Stop/Start
Optical	OTN PNF#2	SDN-C	SDN-C

SO Recipe different controller per flow

Lifecycle Management vs CM

SSH key to NF to authenticate controller

Data structure/Platform-Application Data

Mechanism that SP to configure for their operation

Choose to have APP-C vs SDN-C to do something

Controllers configured to handle domain & function

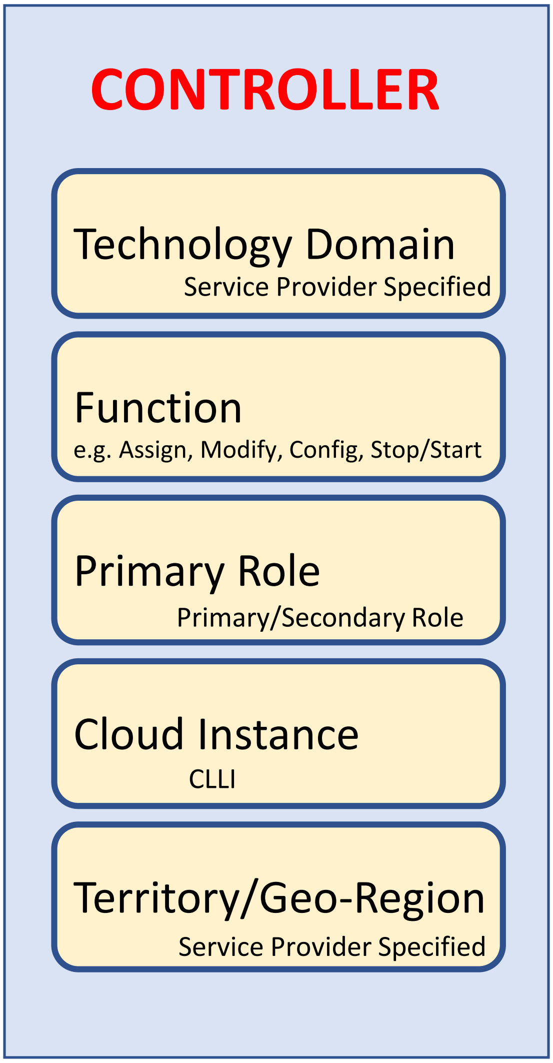
Controllers could publish to the table

Controller register to give provide input for auto-population

Controller when created can state its capabilities

Dynamic Association Approach

CONTROLLER REGISTERS IS ABILITIES



CONTROLLER

Technology Domain
Service Provider Specified

Function
e.g. Assign, Modify, Config, Stop/Start

Primary Role
Primary/Secondary Role

Cloud Instance
CLLI

Territory/Geo-Region
Service Provider Specified

In Run-time the controller can "register" with the *Dynamic Association Handler* which would specify its capabilities

Controller created specifies its abilities.

When you onboard a controller
It should self-identify the capabilities
It supports. SDNC might support modify config doesn't support stop/start LCM.

SO Work-flows. Not only at controller level but at different level (resource level).

SAS – A service comes up, dependencies / capabilities. Based on capabilities resolve dependencies & corresponding API used. PNP "on the fly" addition. 3rd party controller. OSDF/SAS Technology based.

Registry w/ Table-metaphor is dynamic. Do controller stuff in controllers. Bootup time register controller. Possible to Re-run registry (during runtime). Register/ Re-Register/ Deregister.

Table should be outside of SO or in Consul. E.g. Healthcheck where SO is not involved. Anyone should be able to do a look-up. CC-SDK.

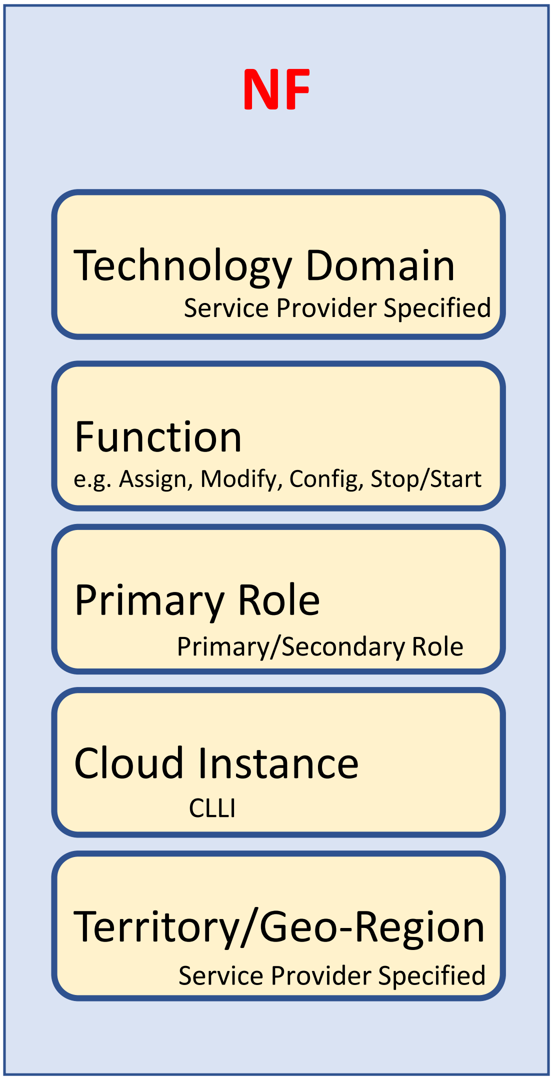
RUN TIME CATALOG (SDC created artifacts) vs CONSUL (Southbound of ONAP controller & capabilities)

VNF is a collection of VFCs. VNF one API endpoint. All VFCs are within same cloud instance. A service can be composed of VNFs from multiple vendors or the same.

Dynamic Association Approach

PNF Plug and Play

VNF Instantiation



NF

Technology Domain
Service Provider Specified

Function
e.g. Assign, Modify, Config, Stop/Start

Primary Role
Primary/Secondary Role

Cloud Instance
CLLI

Territory/Geo-Region
Service Provider Specified

A NF is a VNF or PNF that can interact with a Controller.

Corner Case: VNF w/ multiple VNF-Cs spanning multiple regions.

Service provider defines “Major” technology domains (e.g. Wireless), the Vendor can define “Minor” technology domain (e.g. 5G RAN)

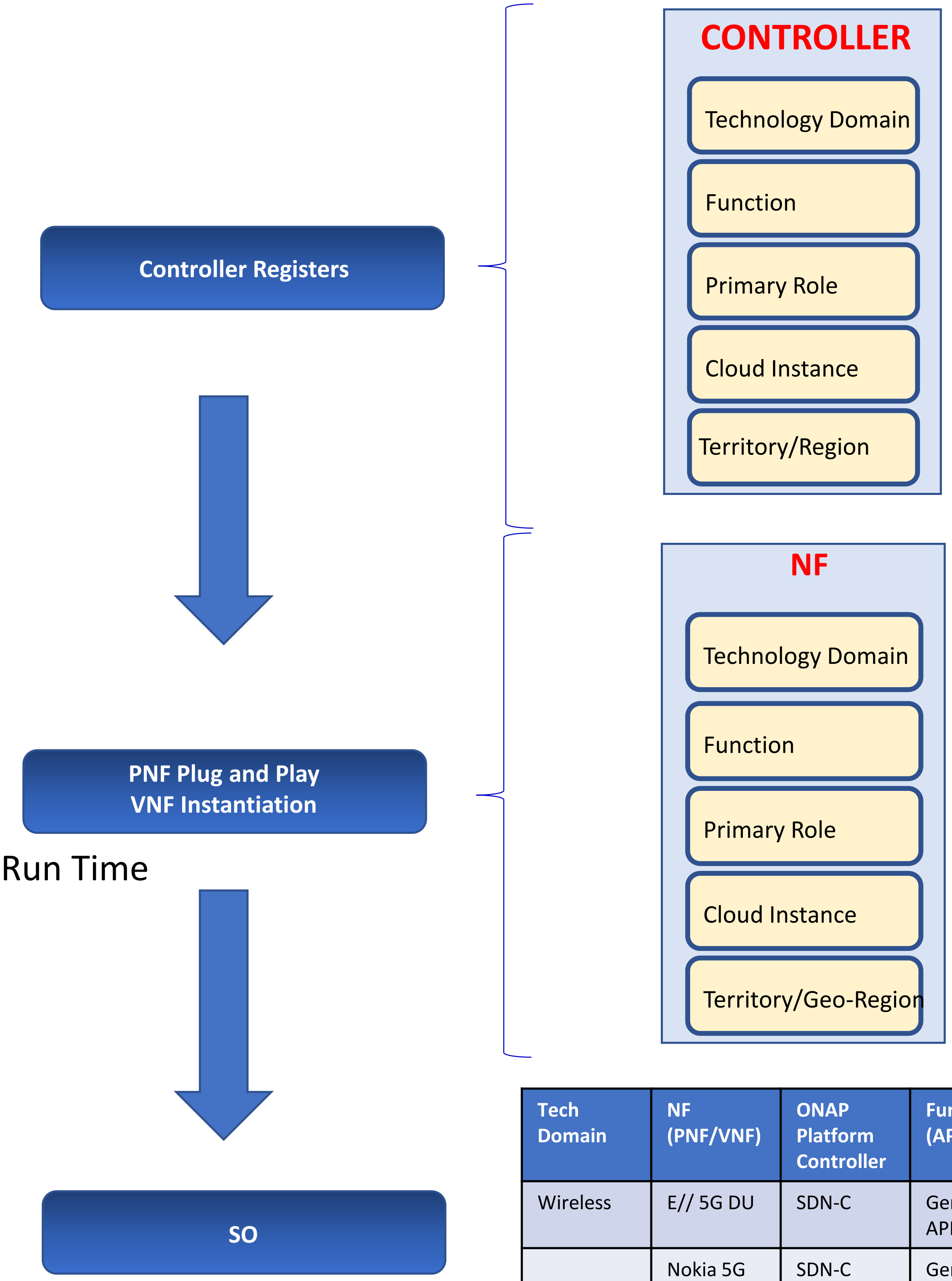
“Lookup” we have a NF … and its part of this xyz Domain.
DU in SDC (Model)
DU in A&AI (Instance)
SO looks up what controller should be used?
DU says “I’m in this tech domain”
Tech domain are “main lookup key”
Function are supporting attributes

SO would know the Cloud Region, Territory.
When it comes time to call a controller.
Via Homing & Complex-Object

QUERY into the “Table” returns Tech domain-Function – Primary – Cloud –Georegion

SDC, VID, OOF, A&AI dependencies.

Dynamic Association Flow



Controller Registers

PNF Plug and Play
VNF Instantiation

Run Time

SO

CONTROLLER

Technology Domain

Function

Primary Role

Cloud Instance

Territory/Region

NF

Technology Domain

Function

Primary Role

Cloud Instance

Territory/Geo-Region

Population Function
Run Time

Tech Domain	NF (PNF/VNF)	ONAP Platform Controller	Function (API)
Wireless	E// 5G DU	SDN-C	Generic API
	Nokia 5G DU	SDN-C	Generic API
	Huawei 5G DU	SDN-C	Generic API
Wireless	Xyz 5G DU	SDN-C	SDN-C

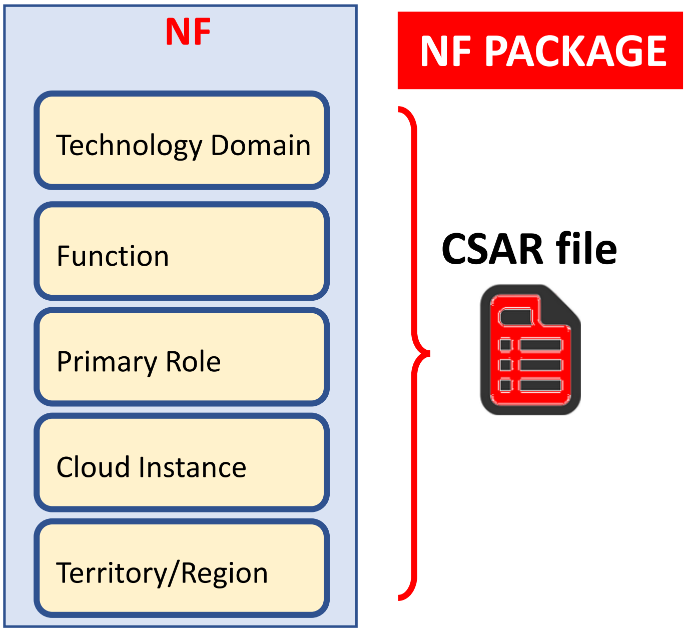
Dynamic Association – Design Time

SDC

STEP 1:
INSTANTIATE CONTROLLER (Design Time)
1a. VNF SDK describe attribute functionalities needed within the TOSCA model those capabilities are mapped into a controller based on the table.
1a. In the ONAP build, Empty Table is Created.
1a. SP provisioning/setup of ONAP instance/installation. i.e. SO, SDN-C, APP-C configuration
1b. SP Create instances of Controller
1c. Profile Controller (tech fun role instance)
1d. Filling in the Controller part of Table.

STEP 2:
NF MODEL (Design Time)
NF Model ingestion/onboarding
NF part of table updated (Vendor Class of NF)
CSAR > VNF SDK

Tech Domain	NF (PNF/VNF)	ONAP Platform Controller	Function (API)
Wireless	E// 5G DU	SDN-C	Generic API
Wireless	Nokia 5G DU	SDN-C	Generic API
	Huawei 5G DU	SDN-C	Generic API
Wireless	Xyz 5G DU	SDN-C	SDN-C



NF

Technology Domain

Function

Primary Role

Cloud Instance

Territory/Region

NF PACKAGE

CSAR file

STEP 1:

INSTANTIATE CONTROLLER (ONAP Installation)

WHO: Capacity/Network Planning Team (Service Provider), ONAP Installation Team (Service Provider)

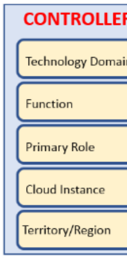
1a. The Table is Created with no value (Empty)

1b. SP provisioning/setup of ONAP instance/installation. i.e. SO, SDN-C, APP-C configuration

1c. SP Create instances of ONAP Platforms Controller (SDN-C, APP-C, VF-C)

1d. Onboard Profile Controller (tech fun role cloud instance loc)

1e. Filling in the Controller part of Table.



STEP 2 (DESIGN TIME ACTIVITIES)

WHO:

SDC Design Studio, CDT, DCAE-DS … etc

1a. Input to SDC TOSCA template from VNF-SDK (or manual) that specifies what the NF needs from a controller and NF properties

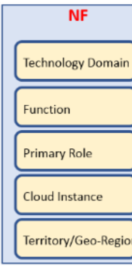
1b. Onboarding PNF-D, Defining Models & Artifacts, Updating SDC Catalog

1c. NF information put into the SDC artifacts/CSAR Package

1d. (optional) SDC could also update the Controller section of the table (by specifying controller information). Tool/Script to controller information if necessary (due to typos, adaptations for congestion, migrations, new technology domains, etc).

1e. SDC distributes Artifacts (CSAR) to ONAP components & listeners

1f. Process of ingesting the model (SDC artifacts), SO populates the table (NF part of the table "new", UPDATES the rest of the table from Optional updates [in step 1d.]).



STEP 3: (RUN TIME OPERATION)

1a. Components in ONAP (e.g. SO, policy) USE the table to find the appropriate controller & APIs for a NF.

1b. When policy's action require a controller look at the table.
When policy's action is to consult w/ SO it knows how to talk to SO

EXAMPLE: Message from NF, executing a Use Case. e.g. Threshold > NF > DCAE > Policy > Action > SO: Controller to interact w/ NF

DISCUSSION

NOTE: Once initial instantiation takes place & mapping, that mapping needs to be maintained for the lifecycle of the NF.

Q: Stability of Table? Will answers change?
A: For same data in will get same answer out.
A: Updates to the table can be made

Q: (Ben) case: conflict race condition (operator#1 updating / operator#2 updating table).
A: Transaction control
ONAP installation team updates. Should own data integrity of the table.
Maria DB transaction control.

Q: (Oskar) case: granularity of look-up, look-up based on a specific action.
Restart but can't do config modify.
For a particular instance of NF handled by one controller instances.
Cases w/ dynamic look-up hard to guarantee.
If you have granularity on operation-level get two answers depending on which operation you want to perform.
Locking in A&AI to block other things.
Operations lock out ONAP from doing changes while maintenance
Course locking/ Finer locking (future)
NF#12 two active Policies
Policy #1 Restart Policy #2 Config-Modify. Trigger at same time.
Table lookup > Get Controller Instances.

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

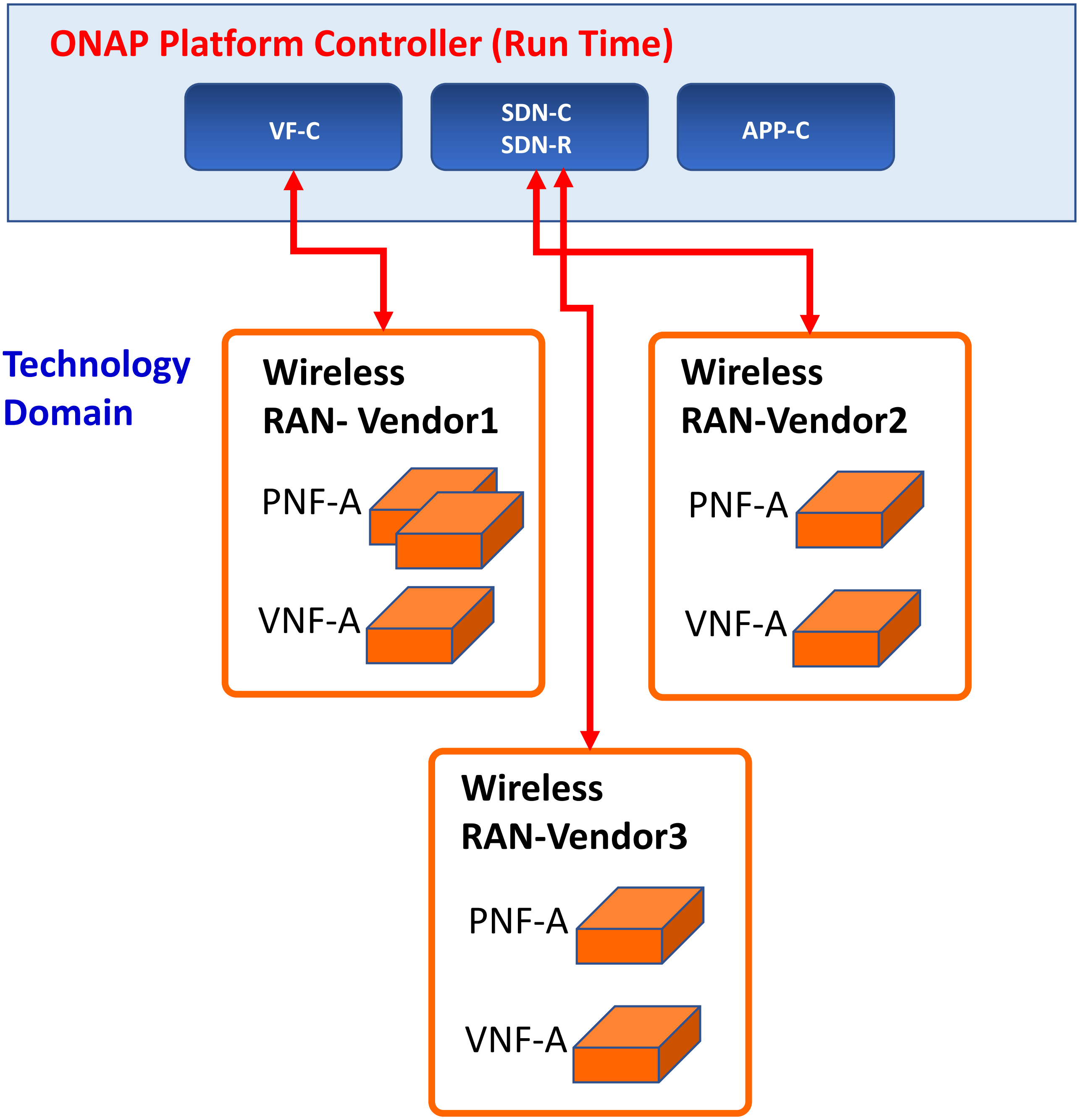
RUN-TIME

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?



ONAP Platform Controller (Run Time)

VF-C

SDN-C
SDN-R

APP-C

Technology Domain

Wireless
RAN- Vendor1

PNF-A

VNF-A

Wireless
RAN-Vendor2

PNF-A

VNF-A

Wireless
RAN-Vendor3

PNF-A

VNF-A

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 3rd 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 17th 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.
External Manager IP Address ⚠	**External 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) 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.

https://wiki.onap.org/pages/viewpage.action?pageId=25431491

Create a Complex Object

```
PUT /aai/v11/cloud-infrastructure/complexes/complex/clli2 HTTP/1.1
Host: <AAI_VM1_IP>:8443
X-TransactionId: 9999
X-FromAppId: jimmy-postman
Real-Time: true
Authorization: Basic QUFJOkFBSQ==
Content-Type: application/json
Accept: application/json
Cache-Control: no-cache
Postman-Token: 734b5a2e-2a89-1cd3-596d-d69904bcda0a

    {
      "physical-location-id": "clli2",
      "data-center-code": "example-data-center-code-val-6667",
      "complex-name": "clli2",
      "identity-url": "example-identity-url-val-28399",
      "physical-location-type": "example-physical-location-type-val-28399",
      "street1": "example-street1-700MountainAvenue",
      "street2": "example-street2-GlenSideRoad",
      "city": "example-city-MurrayHill",
      "state": "example-state-NewJersey",
      "postal-code": "example-postal-code-07974",
      "country": "example-country-UnitedStates",
      "region": "example-region-val-28399",
      "latitude": "111.1",
      "longitude": "234.2,
      "elevation": "example-elevation-538feet",
      "lata": "example-lata-val-28399"
    }
```

PNF A&AI Entry

PARAMETER	DESCRIPTION
PNF-NAME	*pnf-name* is the Key in AAI. pnf-name 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. Note: MAC address & serial number are unique per vendors; not across vendors, so the Vendor name is added to insure uniqueness.
EQUIP-TYPE	The *equip-type* parameter gives the type of the PNF.
EQUIP-VENDOR	The *equip-vendor* is an optional parameter which indicates the vendor for the PNF. For example, Nokia or Ericsson.
EQUIP-MODEL	The *equip-model* is an optional parameter which indicates the model of the PNF.
PNF-ID	UUID = Service provider assigned number from network planner.
MANAGER IP ADDRESS	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	This is the MAC address of the PNF. This is a service field.
SERIAL NUMBER	This is the serial number of the PNF. This is a service field.
PROXY IP ADDRESS	This field contains the *proxy IP address* 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:

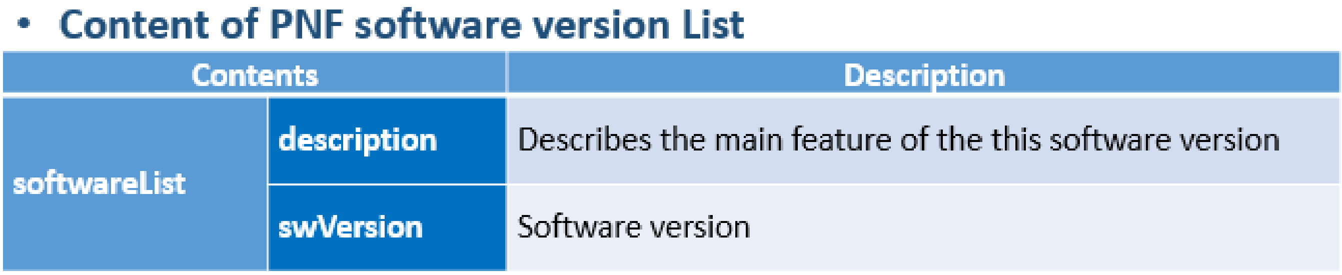
Adding Description to software_versions (after Casablanca R4+)

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

(SDC) software_versions data type (R4+)

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 DUBLIN)

ALARM DEFINITIONS

YAML Definitions
(Vendor provided)
Alarm Dictionary
& Sub-tending faults
[Static Info]

MEASUREMENT DEFS

YAML Definitions
(Vendor provided)
Meas. Dictionary

Measurement Schema
(Vendor provided)

CONFIGURATION DEFS

Configuration Schema
(Vendor provided)

YAML Definitions
(Vendor provided)

EVENT DEFS

Events
YAML Definitions
(pnfRegistration, HVVes)

PLAYBOOK DEFS

NF Ext Controller
Ansible Playbooks
(Ansible Configuration)

WORK-FLOWS

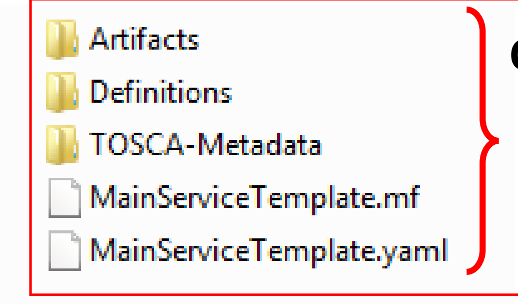
Work-Flows

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 ->

YAML NF REGISTRATION (R4 DUBLIN)

EVENT DEFS

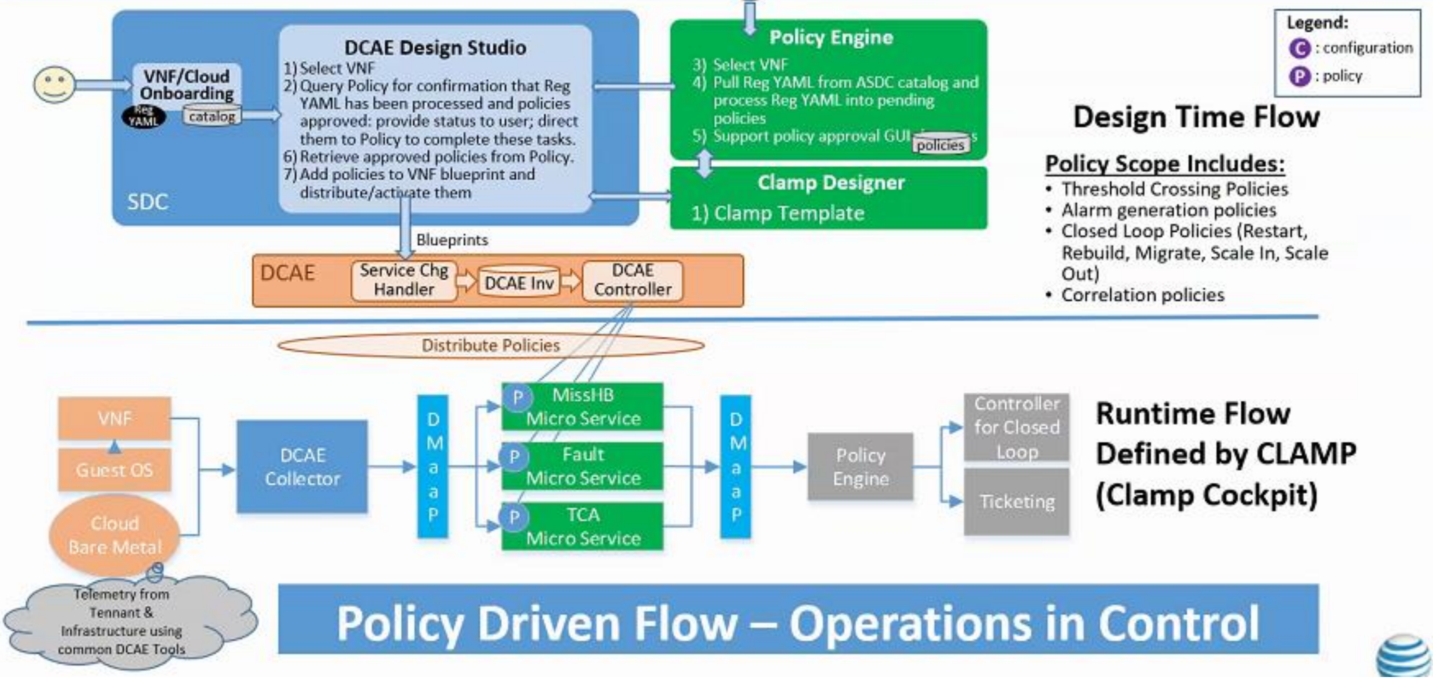
Events
YAML Definitions
(pnfRegistration, HVVes)

YAML Registration – All events a NF supports
Indicates how this NF send the data for that event
ACTION: (Sept 25) What is done w/ a VNF now? for VNF YAML Registration. Architecture Team (Chris Donley, Steve Terrill, Brian Freeman)
DCAE is consuming Artifacts / Registration YAML

1. **METADATA**: Map to Meta-data connected to Service to be instantiated for
2. **CONSUMER**: DCAE to connect between service & file. What DCAE component will read this information. Who is consuming the information. SERVICE CHANGE HANDLER is the DCAE "LISTENER" of SDC. Anyone can consume the SDC notifications. [Design Time: (now=R3/R4) person looking at YML designing the Policy, [future] policy to auto-read the YML and then a person looks at it, in YML format], go into policy, approved by people, then into tosca-file blueprint to DCAE to implement a policy)
3. **FORMAT**: Do they expect same format?
4. **MODIFICATIONS**: Does SDC need to do Modifications or processing? [FM] no additional mapping [PM]. Does DCAE-DS or DCAE need any modification to consume the data. [(R3/R4) Nothing needs to be modified. Use as a vendor artifact and is stored "as is". (future) Vendor may have recommendations. Requirements wanted meta-info stored in YML format. DS & Policy to have "round-trip" YML – interaction w/ policy results reflected back in YML format.] DS SDC GUI. YML loaded in the GUI. Policy-GUI "pending policies". Approved by designers. CLAMP (GUI) input parms for design template what policy is needed for template. DCAE-DS (GUI) [create re-usable service template]. Events are stored in a database. May have to create schema YMLs can be also be used for schema (to store). (1) what payload & data from xNF. (2) what to do with the data.
5. **DISTRIBUTION** - [YML vendor] > Policy GUI, CLAMP, DCAEDS GUI > DCAE Inventory > Cloudify Blueprint [policy database] > SDC distributes cloudify blueprint to policy controller. Controller distribute to micro-service. RUN TIME Clamp/Policy GUI run-time parm for that blueprint. DCAE Inventory pulls the blueprint. Update workflow want to change policy.
6. **SUBSCRIBING** - Subscribing to information. (1) new ARTIFACT TYPE can subscribe to that new type, DCAE makes new service, knows to download the new type, (2) all is packaged in the CSAR package.
7. **ARTIFACT TYPE**: Application that receives the artifact needs to know what it is. Will DCAE try one way to read, or another. *QUESTION do we need a new Artifact Type or can we reuse the DCAE_TOSCA, DCAE_JSON*.
8. **API** - SUPPORT API to retrieve to get whole Artifact or get part of it based on Event Name out of YML to show, PARSE or show portion.
9. **MERGED GUI** – Clamp, Policy, DCAE DS into the SDC Design Studio. These 3 GUI drive the work flow & processing YML registration files.

Control Loop Sub-Committee (Vijay/ Pam Dragosh)

ONAP Self-Serve Event Processing Policy Generation POC



Legend:
- C : configuration
- P : policy

Design Time Flow

Policy Scope Includes:
- Threshold Crossing Policies
- Alarm generation policies
- Closed Loop Policies (Restart, Rebuild, Migrate, Scale In, Scale Out)
- Correlation policies

DCAE Design Studio

1) Select VNF
2) Query Policy for confirmation that Reg YAML has been processed and policies approved: provide status to user; direct them to Policy to complete these tasks.
6) Retrieve approved policies from Policy.
7) Add policies to VNF blueprint and distribute/activate them

Policy Engine

3) Select VNF
4) Pull Reg YAML from ASDC catalog and process Reg YAML into pending policies
5) Support policy approval GUI policies s

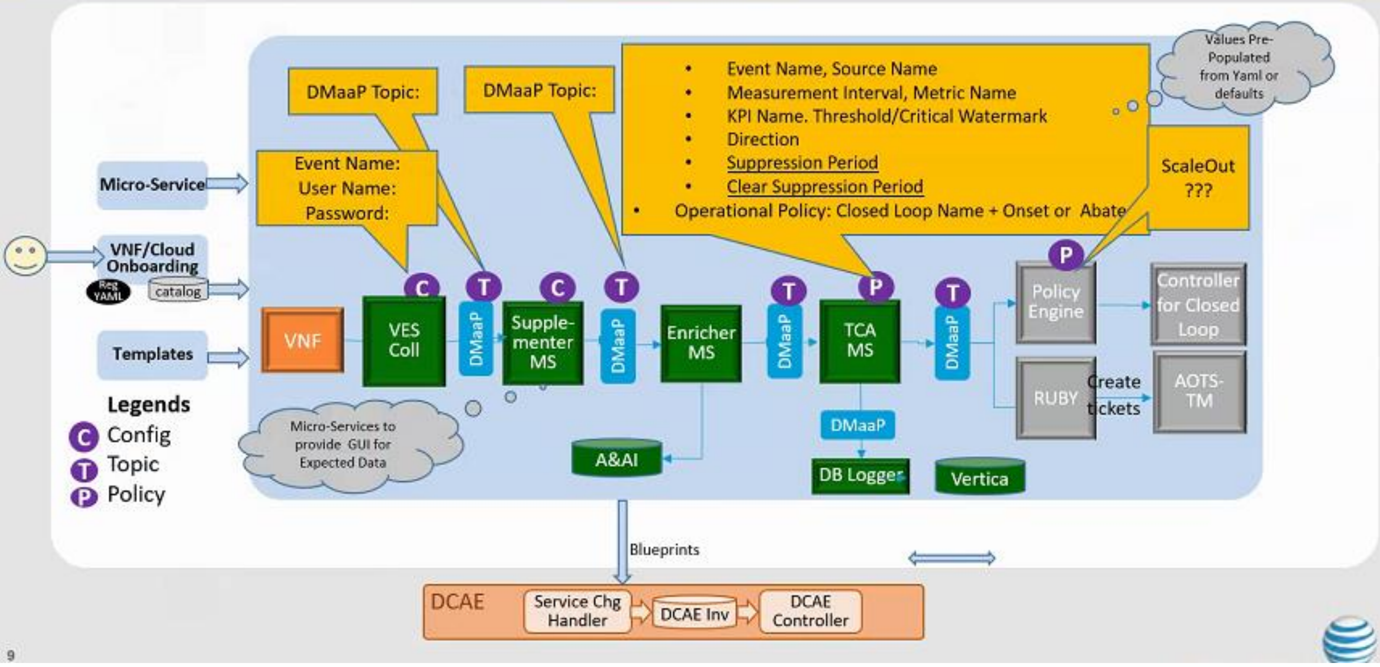
Clamp Designer

1) Clamp Template

Runtime Flow Defined by CLAMP (Clamp Cockpit)

Policy Driven Flow – Operations in Control

TCA Configuration (One for each TCA being watch and also for complex TCAs)



- Event Name, Source Name
- Measurement Interval, Metric Name
- KPI Name. Threshold/Critical Watermark
- Direction
- Suppression Period
- Clear Suppression Period
- Operational Policy: Closed Loop Name + Onset or Abate

Legends
- C Config
- T Topic
- P Policy

YML Registration File

SDC Catalog

Blueprint

Component Spec

SDC Design
Studio GUI
DCAE-DS
(SDC) GUI

Cloudify Blue-Print

DCAE Inventory

Policy
GUI

CLAMP
GUI

Policy Artifact
PolicyID

Policy Database

DESIGN TIME (Step 1 Policy & YML > Blueprint)

1. YML registration loaded SDC catalog (Database)
2. [person] Policy Blueprint SDC DS GUI
3. Policy Database/DCAE Inventory
4. Policy/CLAMP GUI are to modify or create the policy blueprint

DESIGN TIME (step 2)

1. CLAMP design configurations

RUN TIME

1. DCAE looks into DCAE Inventory gets the Policy Artifact

DCAE VES Collector

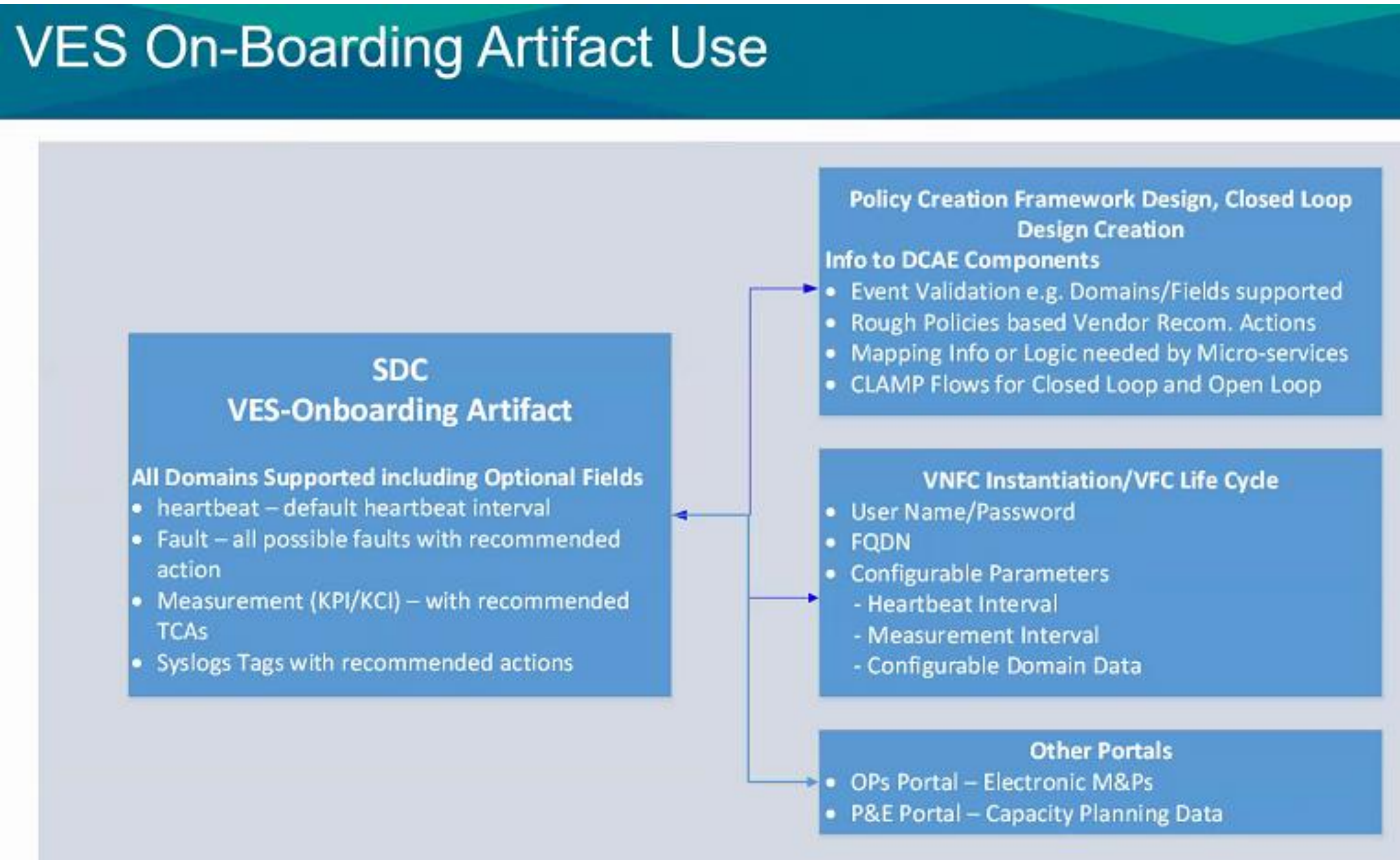
Micro-Service
Event Name
Source Name

https://wiki.onap.org/display/DW/MicroServices+Onboarding

Oct 10, 2018

Any Event that is a valid domain that is formatted properly will be accepted.
And DCAE collector puts it on the DMaaP bus
The Microservice will process (and use the YML registration file)
xNF artifact should be tied to the service
When the service is onboarded to platform
Should indicate what policy is support
These are the specific alarms that I perform analytics on.
Translate that into rules for the services.
Events, Policy flow is a run-time update.
In run-time can change the configuration for that service.

YML registration –
YML is never used in run-time
They are used to define policies that are used in run-time



VES On-Boarding Artifact Use

SDC
VES-Onboarding Artifact

All Domains Supported including Optional Fields
- heartbeat – default heartbeat interval
- Fault – all possible faults with recommended action
- Measurement (KPI/KCI) – with recommended TCAs
- Syslogs Tags with recommended actions

Policy Creation Framework Design, Closed Loop Design Creation
Info to DCAE Components
- Event Validation e.g. Domains/Fields supported
- Rough Policies based Vendor Recom. Actions
- Mapping Info or Logic needed by Micro-services
- CLAMP Flows for Closed Loop and Open Loop

VNFC Instantiation/VFC Life Cycle
- User Name/Password
- FQDN
- Configurable Parameters
  - Heartbeat Interval
  - Measurement Interval
  - Configurable Domain Data

Other Portals
- OPs Portal – Electronic M&Ps
- P&E Portal – Capacity Planning Data

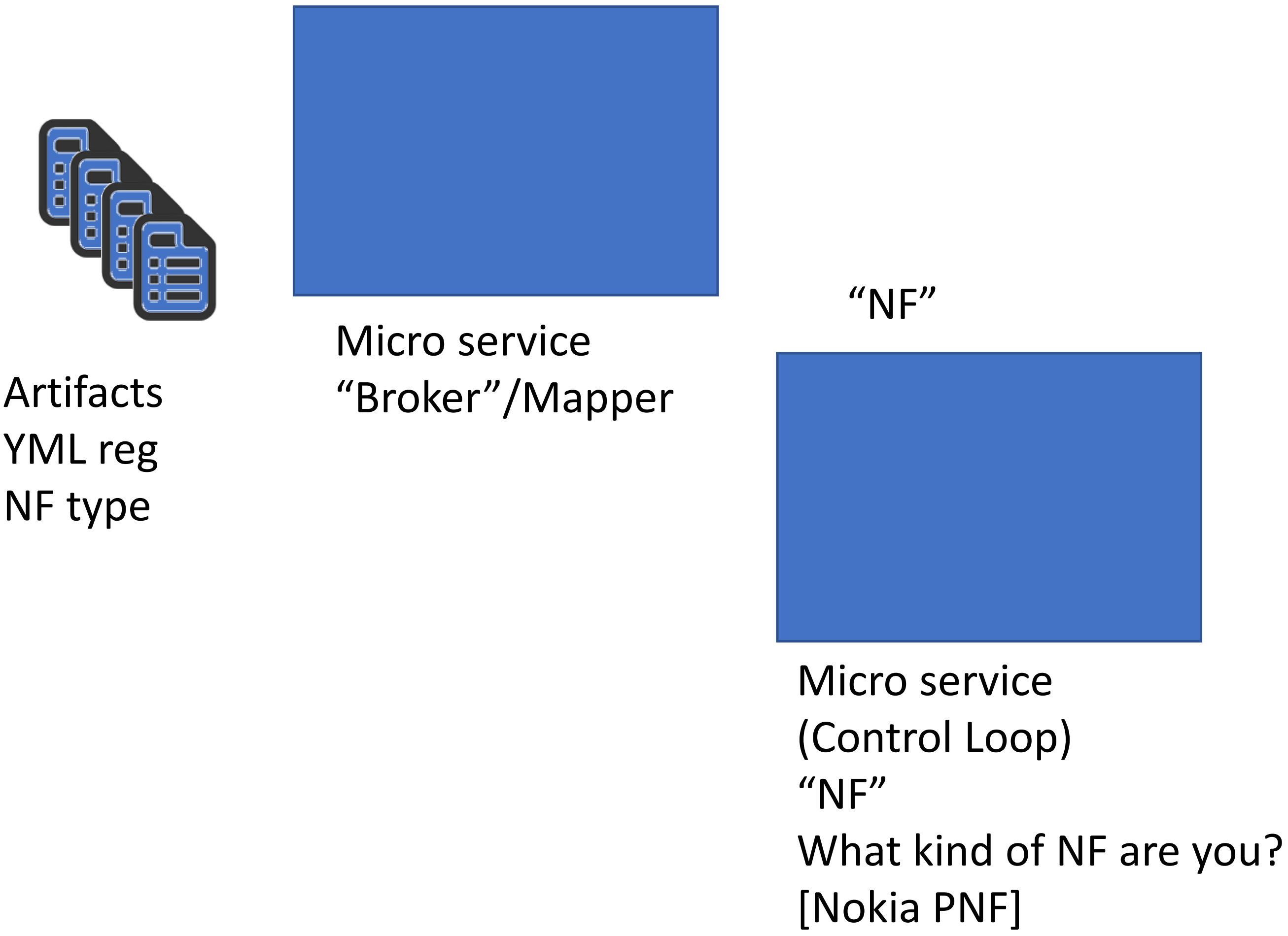
NOTES OCT 10, 2018 Control Loop Meeting
ATTENDEES - Pamela Dragosh, Martial Ngueko, Vijay VK, Vivian Ye, Adam Krysiak, Alex Shatov, Arash Hekmat, Liam Fallon, Lukasz Grech, Marco Platania, Xin Miao

SDC
Create a Service
Artifacts
Use VES registration to automatically indicate Control loops
Pick & choose
CLAMP & Policy to realize

Control loops – artifact
PNF CSAR
DCAE DS – cloudify blueprint, templates.
- tie together microservices,
- assign VES artifacts to specific control loop
- alarm that I want to use control loop
- at this point in template can specify which PNF types are valid
- match template to right VES artifact

For how microservice talk to each other
VES registration does that have to be part of DC

Service onboarding (generic)
Policy configuration for individual service
Tailored to control loop
Policy modeling for that service tied to the flow.
Support these VES things.
HELP TIE Artifacts.
QUESTION is there Separate Micro-service per PNF type?
Holmes have global instance
Rule-update.
CLAMP, POLICY, DCAE.
Support in automated way.



Artifacts
YML reg
NF type

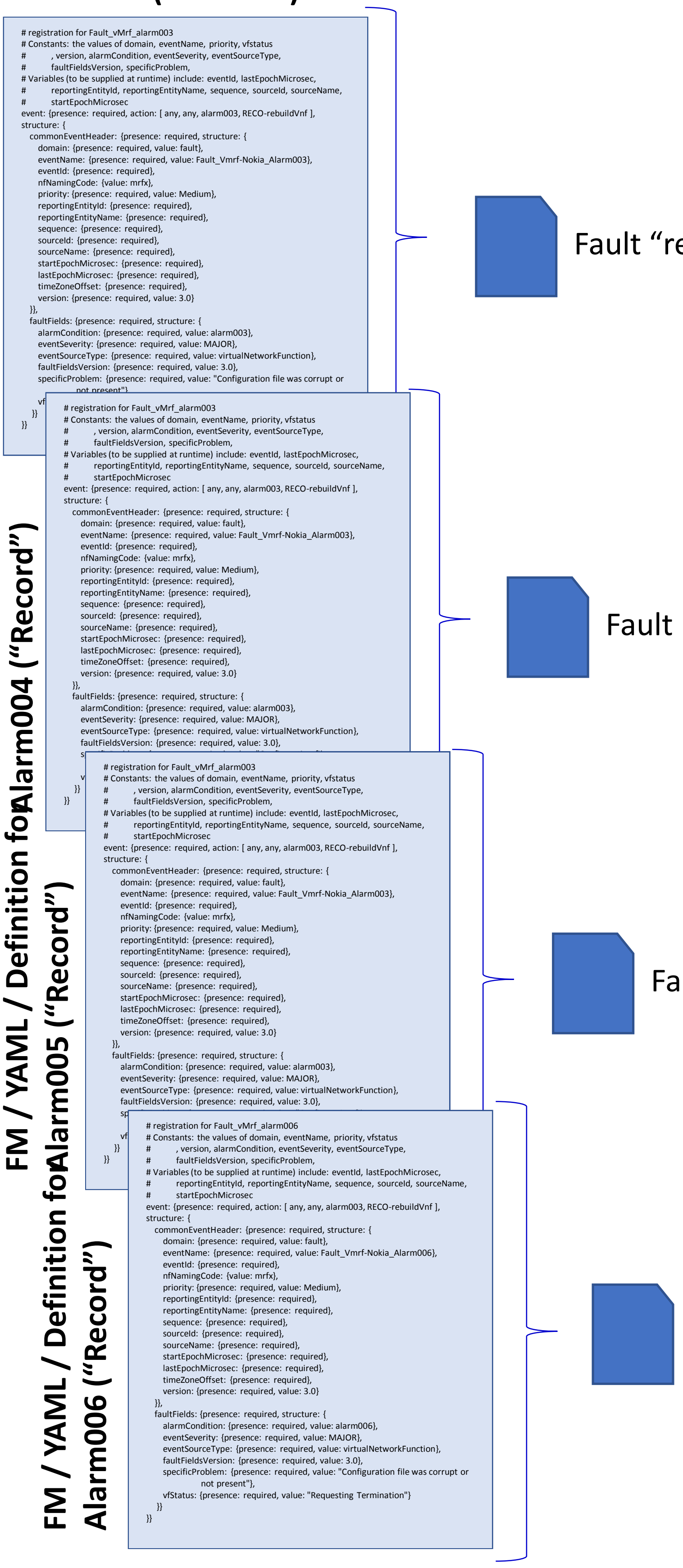
Micro service
“Broker”/Mapper

“NF”

Micro service
(Control Loop)
“NF”
What kind of NF are you?
[Nokia PNF]

FM / YAML / Definition for Alarm003 ("Record")

VES Registration Document



```
# registration for Fault_vMrf_alarm003
# Constants: the values of domain, eventName, priority, vfstatus
#         , version, alarmCondition, eventSeverity, eventSourceType,
#         faultFieldsVersion, specificProblem,
# Variables (to be supplied at runtime) include: eventId, lastEpochMicrosec,
#         reportingEntityId, reportingEntityName, sequence, sourceId, sourceName,
#         startEpochMicrosec
event: {presence: required, action: [ any, any, alarm003, RECO-rebuildVnf ],
structure: {
   commonEventHeader: {presence: required, structure: {
      domain: {presence: required, value: fault},
      eventName: {presence: required, value: Fault_Vmrf-Nokia_Alarm003},
      eventId: {presence: required},
      nfNamingCode: {value: mrfx},
      priority: {presence: required, value: Medium},
      reportingEntityId: {presence: required},
      reportingEntityName: {presence: required},
      sequence: {presence: required},
      sourceId: {presence: required},
      sourceName: {presence: required},
      startEpochMicrosec: {presence: required},
      lastEpochMicrosec: {presence: required},
      timeZoneOffset: {presence: required},
      version: {presence: required, value: 3.0}
   }},
   faultFields: {presence: required, structure: {
      alarmCondition: {presence: required, value: alarm003},
      eventSeverity: {presence: required, value: MAJOR},
      eventSourceType: {presence: required, value: virtualNetworkFunction},
      faultFieldsVersion: {presence: required, value: 3.0},
      specificProblem: {presence: required, value: "Configuration file was corrupt or
                 not present"},
      vf
   }}
}}
```

Fault "record" for Alarm #003

FM / YAML / Definition for Alarm004 ("Record")

```
# registration for Fault_vMrf_alarm003
# Constants: the values of domain, eventName, priority, vfstatus
#         , version, alarmCondition, eventSeverity, eventSourceType,
#         faultFieldsVersion, specificProblem,
# Variables (to be supplied at runtime) include: eventId, lastEpochMicrosec,
#         reportingEntityId, reportingEntityName, sequence, sourceId, sourceName,
#         startEpochMicrosec
event: {presence: required, action: [ any, any, alarm003, RECO-rebuildVnf ],
structure: {
   commonEventHeader: {presence: required, structure: {
      domain: {presence: required, value: fault},
      eventName: {presence: required, value: Fault_Vmrf-Nokia_Alarm003},
      eventId: {presence: required},
      nfNamingCode: {value: mrfx},
      priority: {presence: required, value: Medium},
      reportingEntityId: {presence: required},
      reportingEntityName: {presence: required},
      sequence: {presence: required},
      sourceId: {presence: required},
      sourceName: {presence: required},
      startEpochMicrosec: {presence: required},
      lastEpochMicrosec: {presence: required},
      timeZoneOffset: {presence: required},
      version: {presence: required, value: 3.0}
   }},
   faultFields: {presence: required, structure: {
      alarmCondition: {presence: required, value: alarm003},
      eventSeverity: {presence: required, value: MAJOR},
      eventSourceType: {presence: required, value: virtualNetworkFunction},
      faultFieldsVersion: {presence: required, value: 3.0},
      s
      v
   }}
}}
```

Fault record for Alarm #004

FM / YAML / Definition for Alarm005 ("Record")

```
# registration for Fault_vMrf_alarm003
# Constants: the values of domain, eventName, priority, vfstatus
#         , version, alarmCondition, eventSeverity, eventSourceType,
#         faultFieldsVersion, specificProblem,
# Variables (to be supplied at runtime) include: eventId, lastEpochMicrosec,
#         reportingEntityId, reportingEntityName, sequence, sourceId, sourceName,
#         startEpochMicrosec
event: {presence: required, action: [ any, any, alarm003, RECO-rebuildVnf ],
structure: {
   commonEventHeader: {presence: required, structure: {
      domain: {presence: required, value: fault},
      eventName: {presence: required, value: Fault_Vmrf-Nokia_Alarm003},
      eventId: {presence: required},
      nfNamingCode: {value: mrfx},
      priority: {presence: required, value: Medium},
      reportingEntityId: {presence: required},
      reportingEntityName: {presence: required},
      sequence: {presence: required},
      sourceId: {presence: required},
      sourceName: {presence: required},
      startEpochMicrosec: {presence: required},
      lastEpochMicrosec: {presence: required},
      timeZoneOffset: {presence: required},
      version: {presence: required, value: 3.0}
   }},
   faultFields: {presence: required, structure: {
      alarmCondition: {presence: required, value: alarm003},
      eventSeverity: {presence: required, value: MAJOR},
      eventSourceType: {presence: required, value: virtualNetworkFunction},
      faultFieldsVersion: {presence: required, value: 3.0},
      sp
      vf
   }}
}}
```

Fault record for Alarm #005

FM / YAML / Definition for Alarm006 ("Record")

```
# registration for Fault_vMrf_alarm006
# Constants: the values of domain, eventName, priority, vfstatus
#         , version, alarmCondition, eventSeverity, eventSourceType,
#         faultFieldsVersion, specificProblem,
# Variables (to be supplied at runtime) include: eventId, lastEpochMicrosec,
#         reportingEntityId, reportingEntityName, sequence, sourceId, sourceName,
#         startEpochMicrosec
event: {presence: required, action: [ any, any, alarm003, RECO-rebuildVnf ],
structure: {
   commonEventHeader: {presence: required, structure: {
      domain: {presence: required, value: fault},
      eventName: {presence: required, value: Fault_Vmrf-Nokia_Alarm006},
      eventId: {presence: required},
      nfNamingCode: {value: mrfx},
      priority: {presence: required, value: Medium},
      reportingEntityId: {presence: required},
      reportingEntityName: {presence: required},
      sequence: {presence: required},
      sourceId: {presence: required},
      sourceName: {presence: required},
      startEpochMicrosec: {presence: required},
      lastEpochMicrosec: {presence: required},
      timeZoneOffset: {presence: required},
      version: {presence: required, value: 3.0}
   }},
   faultFields: {presence: required, structure: {
      alarmCondition: {presence: required, value: alarm006},
      eventSeverity: {presence: required, value: MAJOR},
      eventSourceType: {presence: required, value: virtualNetworkFunction},
      faultFieldsVersion: {presence: required, value: 3.0},
      specificProblem: {presence: required, value: "Configuration file was corrupt or
                 not present"},
      vfStatus: {presence: required, value: "Requesting Termination"}
   }}
}}
```

Fault record for Alarm #006

🟦	Fault records	N
🟥	Heartbeat records	N
🟨	Measurement records	N
🟩	Mobile Flow Records	N
🟩	PnfRegistration records	1
🟦	SipSignaling records	N
🟪	State Change Records	N
🟧	SysLog records	N
🟦	TCA records	N
⬜	Voice Quality records	N
🟥	Other records	N

1

YAML REGISTRATION FILE (FM, Heartbeating, Meas, Syslog, Mobile Flow, Sip Signaling, etc)

Separate Registration event PER “NF type” (E//, Nokia, Huawei etc) each PNF has separate presentation onboarded PNF. Building block used in different services.

CONSUMERS: DCAE, Analytics Applications (?), Custom Plug-ins

- ‘Fault’ for the fault domain
- ‘Heartbeat’ for the heartbeat domain
- ‘Measurement’ for the measurements domain
- ‘MobileFlow’ for the mobileFlow domain
- ‘Other’ for the other domain
- ‘PnfReg’ for the pnfRegistration domain
- ‘SipSignaling’ for the sipSignaling domain
- ‘StateChange’ for the stateChange domain
- ‘Syslog’ for the syslog domain
- ‘Tca’ for the thresholdCrossingAlert domain
- ‘VoiceQuality’ for the voiceQuality domain

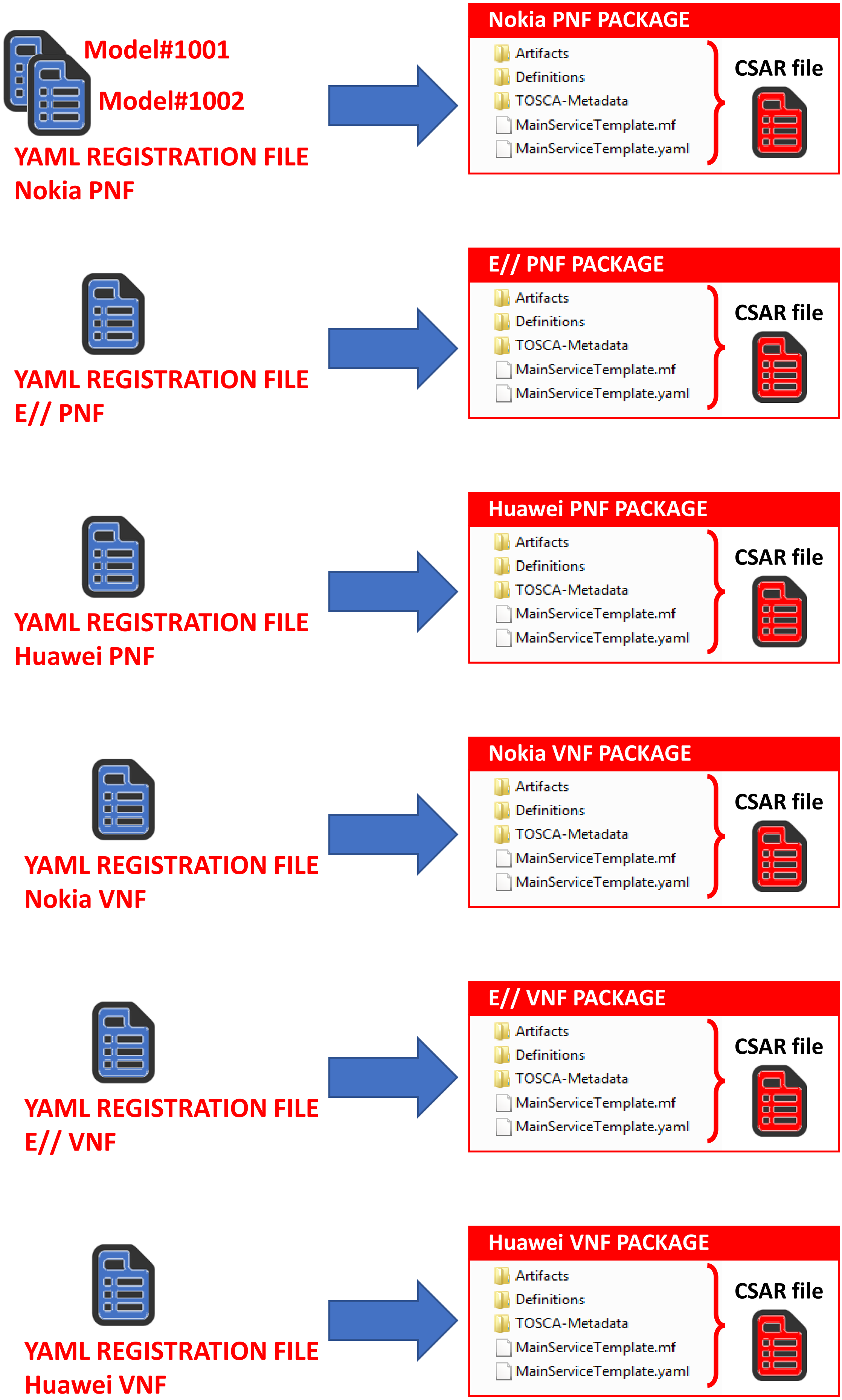
1. NORMALIZE – different definition for different NFs.
2. DCAE-DS – could be used to modify the YAML registration if necessary

I have PNF two different releases, in release 1 I may have a group of items In dictionary, in release 2 can I provide different one.

PNF Type	Release 1 (S/W version)	Release 2 (S/W version)
E// PNF – FM	FM “records” (YAML registration file) YAMLregfile-E//Rel1	FM “records” (YAML registration file) YAMLregfile-E//Rel2
E// PNF – PM	PM “records” (YAML registration file) YAMLregfile-E//Rel1	PM “records” (YAML registration file) YAMLregfile-E//Rel2
Nokia PNF – FM	FM “records” (YAML registration file) YAMLregfile-NokiaRel1	FM “records” (YAML registration file) YAMLregfile-NokiaRel1
Nokia PNF - PM	PM “records” (YAML registration file) YAMLregfile-NokiaRel1	PM “records” (YAML registration file) YAMLregfile-NokiaRel1

If just one alarm or one PM changed, could you provide the single “record” or the entire file again?

VNF FM/PM INFO
PROPOSAL
FM/PM info in the CSAR package
SOL 004
General Artifact shared & distributed
Similar to Heat File
No exact representation
For VNF no unified way to specify FM, PM
CM operation



Model#1001

Model#1002

YAML REGISTRATION FILE
Nokia PNF

Nokia PNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

YAML REGISTRATION FILE
E// PNF

E// PNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

YAML REGISTRATION FILE
Huawei PNF

Huawei PNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

YAML REGISTRATION FILE
Nokia VNF

Nokia VNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

YAML REGISTRATION FILE
E// VNF

E// VNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

YAML REGISTRATION FILE
Huawei VNF

Huawei VNF PACKAGE

- Artifacts
- Definitions
- TOSCA-Metadata
- MainServiceTemplate.mf
- MainServiceTemplate.yaml

CSAR file

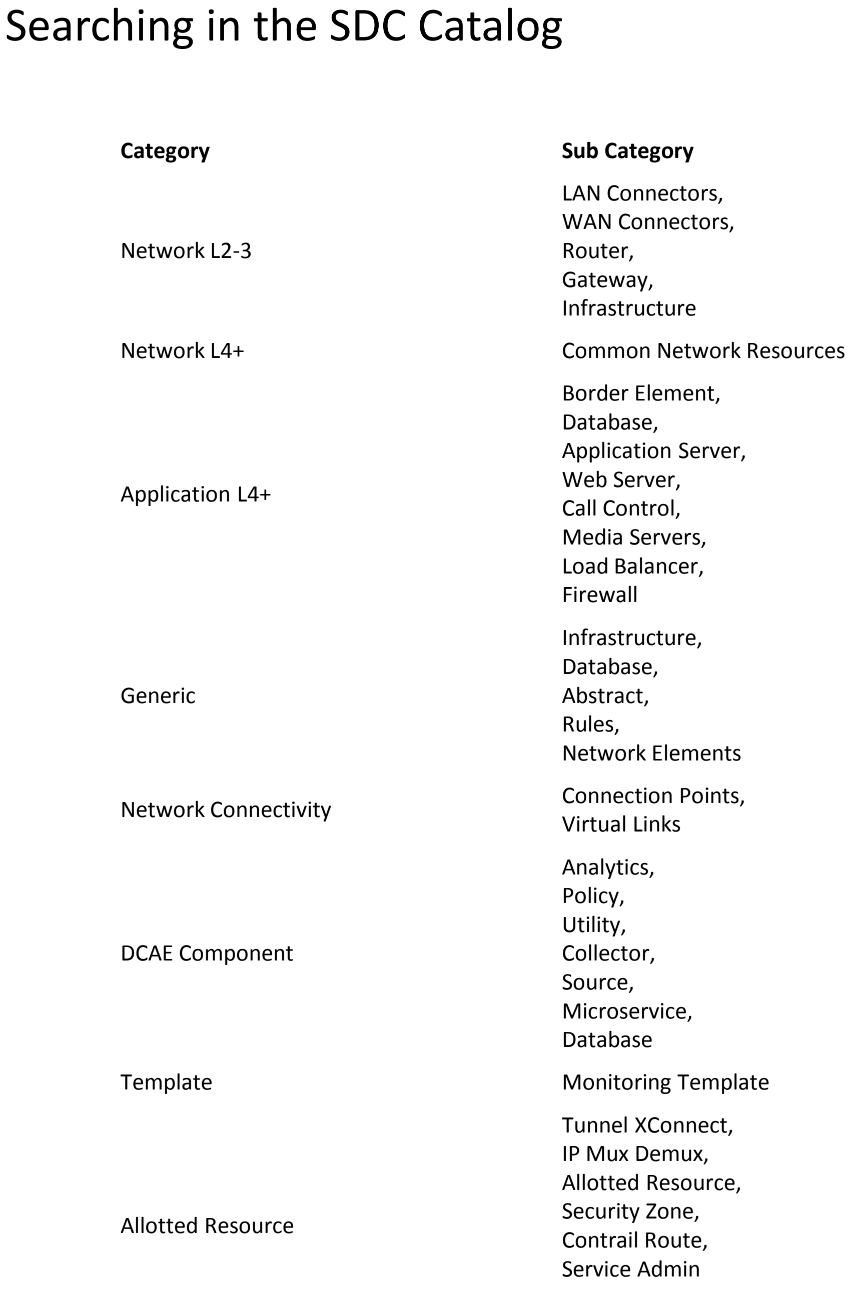
SDC Artifacts Type

https://wiki.onap.org/display/DW/SDC+supported+artifact+types

Artifact Type	Description	Valid on Component Types
HEAT	HEAT Base template (yaml format). Provided by vendor during on-boarding	VF
HEAT_VOL	HEAT template defining required volumes (yaml format). Provided by vendor during on-boarding	VF
HEAT_NET	HEAT template defining required networks (yaml format). Provided by vendor during on-boarding (deprecated)	VF
HEAT_ENV	HEAT Environment artifact (yaml format, extension ".env"). This artifact is accompanied to HEAT template and containing values to subset of the template's parameters. First version is provided by vendor, then SDC generates new content according to designer changes	VF / VF instance
HEAT_ARTIFACT	Supplementary artifact referenced in HEAT* template ("get_file"). Provided by vendor during on-boarding	VF
HEAT_NESTED	HEAT file referenced from another HEAT file (yaml format). Provided by vendor during on-boarding	VF
YANG_XML	YANG asset based XML	Service / VF / VFC
VNF_CATALOG	YANG asset based XML	Service / VF / VFC
MODEL_INVENTORY_PROFILE	Inventory Asset (XML format). As from 1610 it is generated by SDC during service certification	Service / VF / VFC
MODEL_INVENTORY_PROFILE	Inventory Asset Named Query Specification (XML format)	Service / VF / VFC
VF_LICENSE	VF License Artifact (XML format). Created during on-boarding of the VF	VF
VENDOR_LICENSE	Vendor License Artifact (XML format). Created during on-boarding of the VF	VF
APPC_CONFIG	configuration artifact	VF
VF_MODULES_META DATA	Json artifact that describes the vfModules of the resource. SDC generates this artifact for every VF instance during the service certification.	Component instance (e.g. VF in Service)
DCAE_TOSCA	TOSCA template used by DCAE (yaml format)	VF / VFCMT (= monitoring template)
DCAE_JSON	JSON description of the DCAE component / template	VF / VFCMT (= monitoring template)
PLAN	Workflow artifact type	Service / VF / VFC

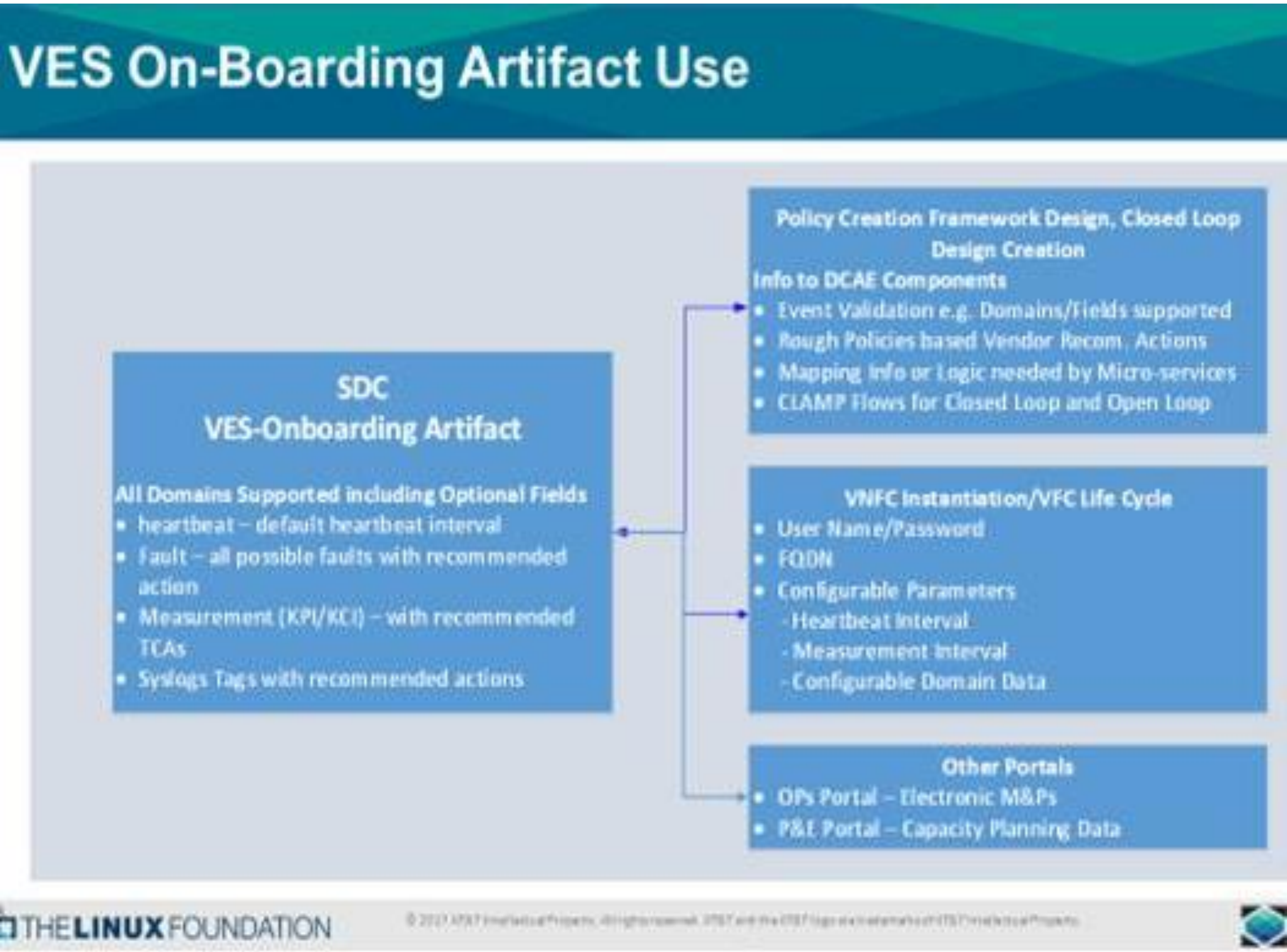
Resource / Service Categories

https://wiki.onap.org/pages/viewpage.action?pageId=38114398



Searching in the SDC Catalog

Category	Sub Category
Network L2-3	LAN Connectors, WAN Connectors, Router, Gateway, Infrastructure
Network L4+	Common Network Resources
Application L4+	Border Element, Database, Application Server, Web Server, Call Control, Media Servers, Load Balancer, Firewall
Generic	Infrastructure, Database, Abstract, Rules, Network Elements
Network Connectivity	Connection Points, Virtual Links
DCAE Component	Analytics, Policy, Utility, Collector, Source, Microservice, Database
Template	Monitoring Template
Allotted Resource	Tunnel XConnect, IP Mux Demux, Allotted Resource, Security Zone, Contrail Route, Service Admin



VES On-Boarding Artifact Use

SDC
VES-Onboarding Artifact

All Domains Supported including Optional Fields

- heartbeat – default heartbeat interval
- Fault – all possible faults with recommended action
- Measurement (KPI/KCI) – with recommended TCAs
- Syslogs Tags with recommended actions

Policy Creation Framework Design, Closed Loop Design Creation

Info to DCAE Components

- Event Validation e.g. Domains/Fields supported
- Rough Policies based Vendor Recom. Actions
- Mapping Info or Logic needed by Micro-services
- CLAMP Flows for Closed Loop and Open Loop

VNFC Instantiation/VFC Life Cycle

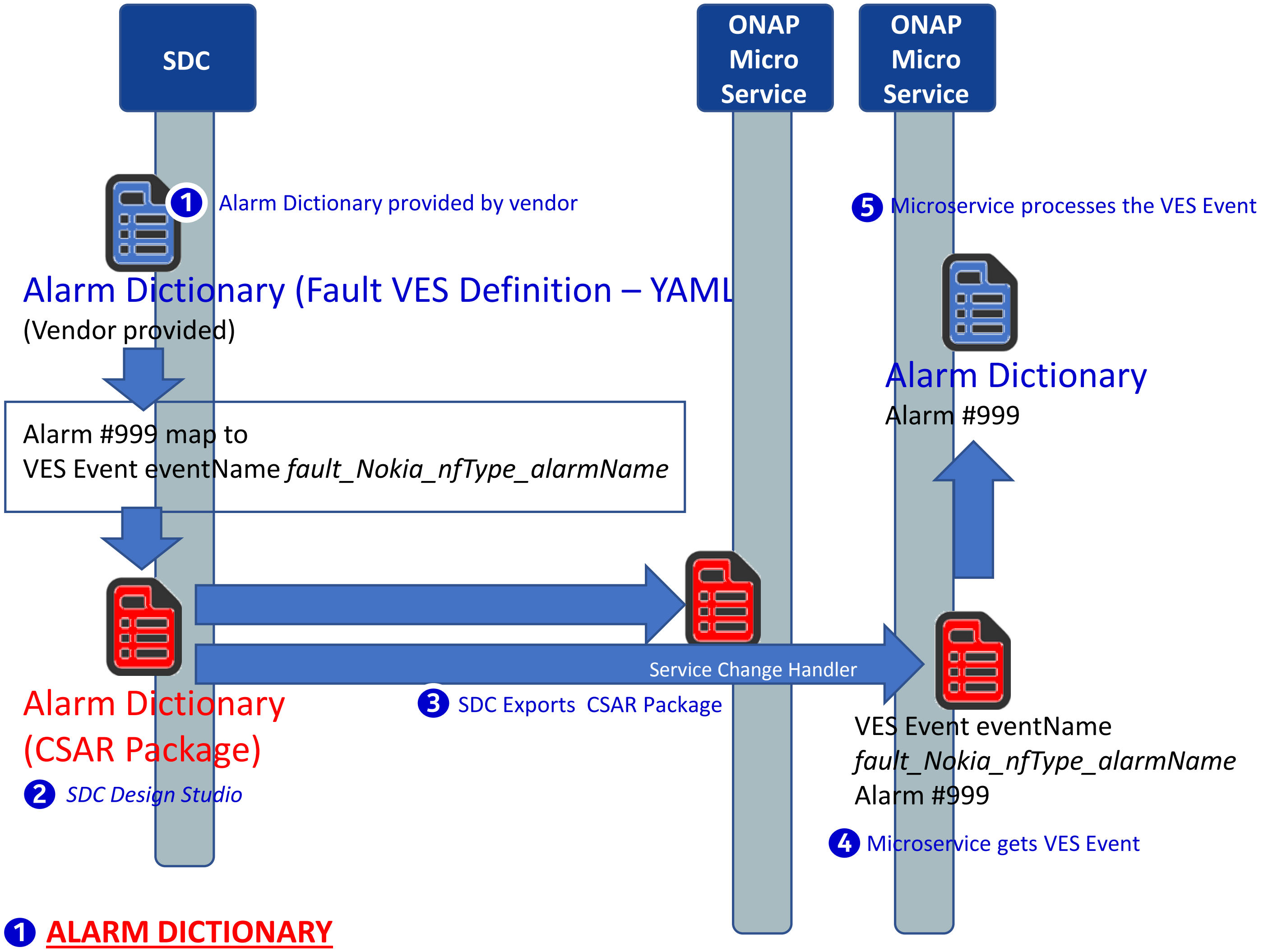
- User Name/Password
- FQDN
- Configurable Parameters
  - Heartbeat Interval
  - Measurement Interval
  - Configurable Domain Data

Other Portals

- OPs Portal – Electronic M&Ps
- P&E Portal – Capacity Planning Data

THE LINUX FOUNDATION

Alarm Dictionary Usage



SDC

ONAP Micro Service

ONAP Micro Service

1 Alarm Dictionary provided by vendor

Alarm Dictionary (Fault VES Definition – YAML

(Vendor provided)

Alarm #999 map to
VES Event eventName *fault_Nokia_nfType_alarmName*

Alarm Dictionary
(CSAR Package)

2 *SDC Design Studio*

Service Change Handler

3 SDC Exports CSAR Package

5 Microservice processes the VES Event

Alarm Dictionary
Alarm #999

VES Event eventName
fault_Nokia_nfType_alarmName
Alarm #999

4 Microservice gets VES Event

1 **ALARM DICTIONARY**

Fault VES “record” defines all alarms/faults published by NF (VNF or PNF)
See the VES Registration document

2 **SDC DESIGN STUDIO Packages & composes CSAR package**

SDC Makes 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
Policy/Action resultant from the alarm

Alarm Dictionary Usage

ALARM DICTIONARY (Fault YAML File) 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). FRTire-Fault FLTire-Fault RRTire-Fault RLTire-Fault -> TIRE LOW TIRE PRESSURE Alarm

NF ALARM DICTIONARY FIELDS (Template)

ALARM FIELD	DESCRIPTION
~~Alarm Dictionary Index~~	~~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.~~ ~~The VES Event *EventID* would encode the Alarm number which will correspond to the Alarm Index.~~ ~~EXAMPLE: 12345~~
Alarm Name	Alarm Name which will be used in the Event Name. **Note** *this maps to the alarmCondition in the VES Fault Event in faultevent fields.* EXAMPLE: Synchronization Lost
Event Type *	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. **Note** *this maps to the eventCategtory in the VES Fault Event in faultevent fields.* EXAMPLE: Quality of Service Alarm
Meaning of Alarm	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 has been lost
Effect of Alarm	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 in Quality of Service
Managed Object(s)	Managed object (MO) associated with this Alarm. **Note** *this maps to the eventSourceType in the VES Fault Event in faultevent fields.* EXAMPLE: Clock (MO)
Probable Cause *	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 EXAMPLE: lossOfSynchronisation
Probable Cause Number*	Probable Cause Number the numeric value associated with the Probable Cause EXAMPLE: 76
Specific Problem *	(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. **Note** *this is the 3GPP Specific problem not be confused with the specificProblem field of the VES Fault Event in faultevent fields.*
Proposed Repair Actions*	It indicates instructions for proposed repair actions. These are defined in ITU-T Recommendation X.733 clause 8.1.2.12. EXAMPLE: Action#12 [= Reset the PNF], ONAP Controller does x
Clearing Type	Indicates whether the alarm is automatically or manually cleared EXAMPLE: Automatic
Additional Text*	This field contain further information on the alarm. This attribute provides *vendor specific* 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. EXAMPLE: Specific data 10
~~Associated Fault(s)~~	~~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: Fault 99999~~

*3GPP TS32.111

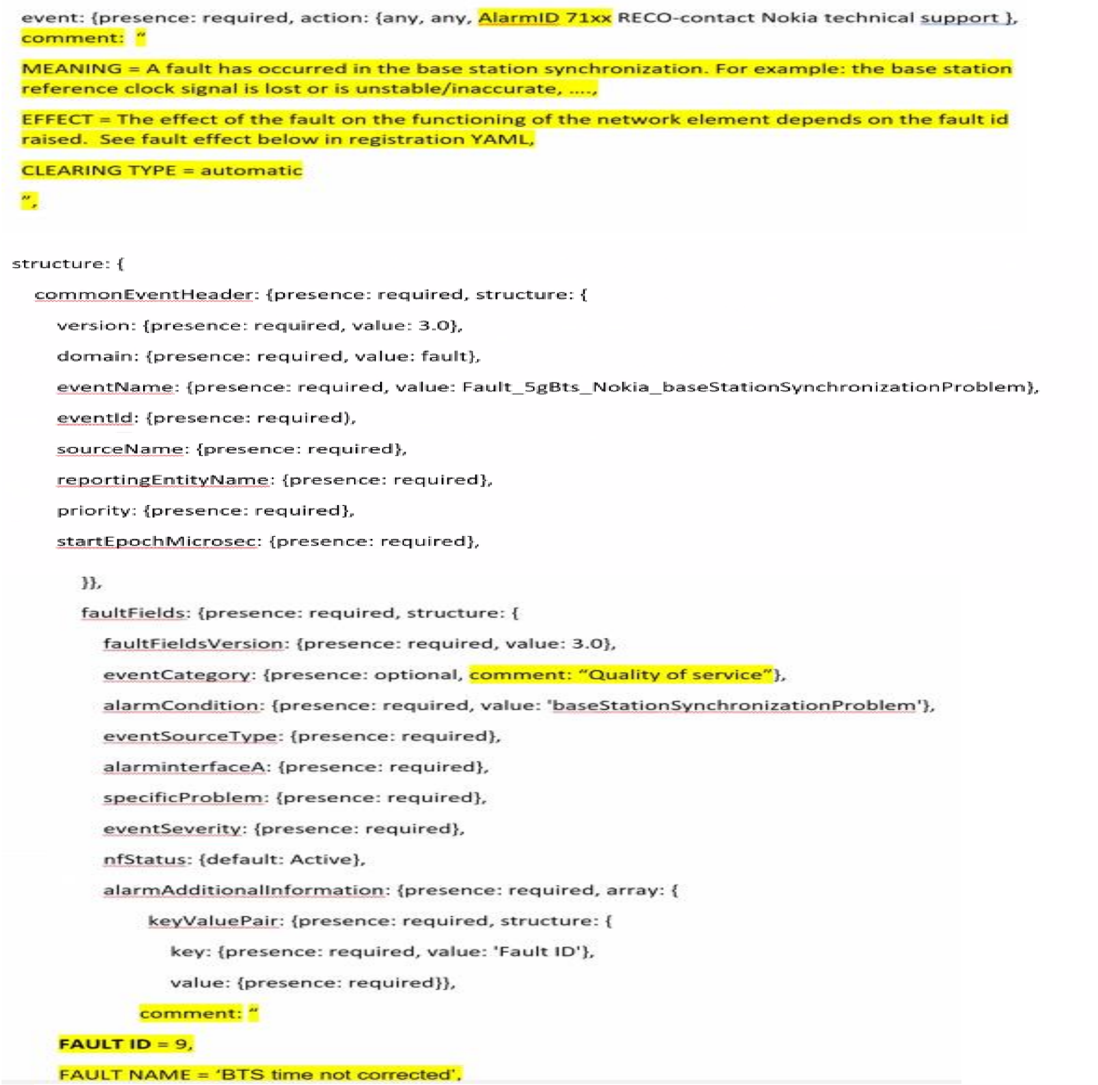
NF ALARM DICTIONARY FIELDS (Template)

ALARM FIELD	STANDARDS	VES FAULT EVENT
Alarm Name	None	*maps to alarmCondition in the VES Fault Event in faultevent fields*
Event Type *	3GPP TS32.111	*maps to eventCategtory in the VES Fault Event in faultevent fields*
Meaning of Alarm	N/A	N/A
Effect of Alarm	N/A	N/A
Managed Object(s)	N/A	*maps to eventSourceType in the VES Fault Event in faultevent fields*
Probable Cause *	3GPP TS 32.111 Annex B ITU-T M.3100 ITU-T Recommendation X.721, X.733, and X.736	N/A
Probable Cause Number *	3GPP TS 32.111 Annex B ITU-T M.3100 ITU-T Recommendation X.721, X.733, and X.736	N/A
Specific Problem *	3GPP TS32.111. ITU-T Recommendation X.733 Clause 8.1.2.2	N/A
Proposed Repair Actions*	3GPP TS32.111	N/A
Clearing Type	N/A	N/A
Additional Text	3GPP TS32.111	N/A

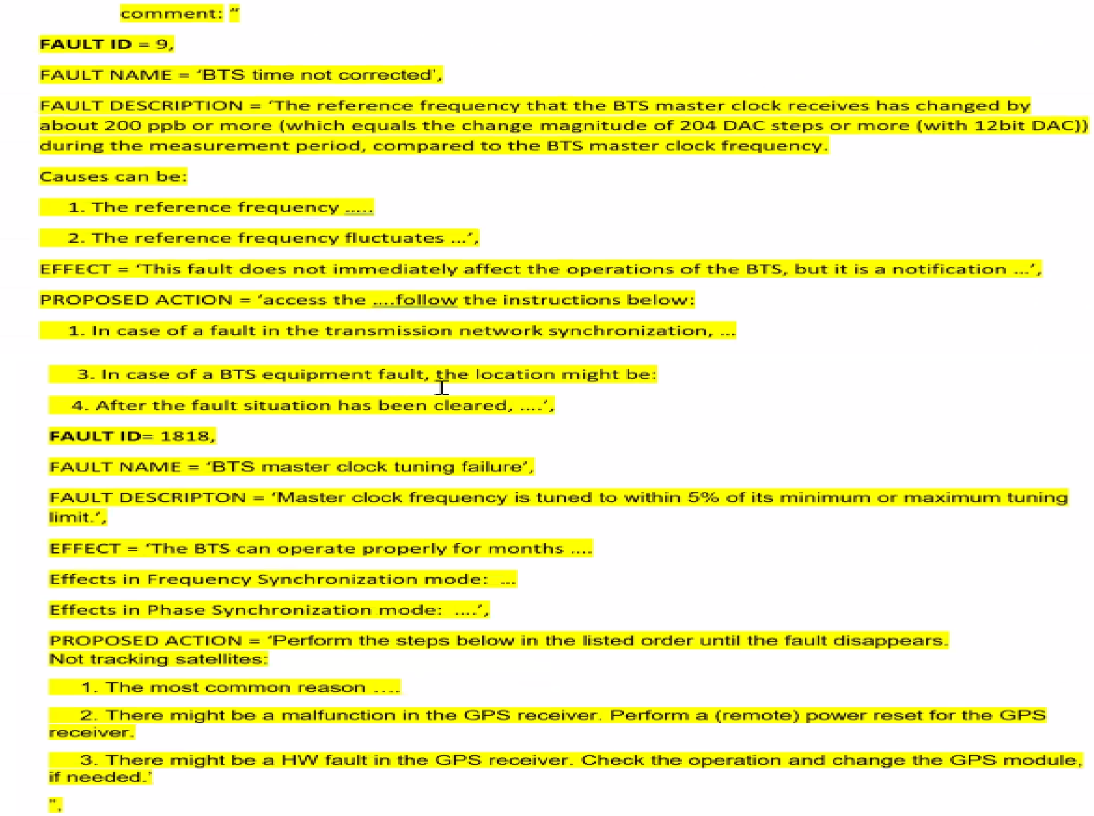
*3GPP TS32.111

NF FAULT DICTIONARY FIELDS (Template)

DEFINITION FIELD	DESCRIPTION
Fault Id	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
Fault Name	Alarm Name which will be used in the Event Name. **Note** *this maps to the alarmCondition in the VES Fault Event in faultevent fields.* EXAMPLE: Loss of Synchronization
Fault Description	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
Managed Object(s)	Managed object (MO) associated with this Alarm. **Note** *this maps to the eventSourceType in the VES Fault Event in faultevent fields.* EXAMPLE: Clock (MO)
Effect of Fault	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
Associated Alarm(s)	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
Proposed Repair Actions	It indicates instructions for proposed repair actions. These are defined in ITU-T Recommendation X.733 clause 8.1.2.12. EXAMPLE: Reset BTS
Additional Text	This field contain further information on the alarm. This attribute provides *vendor specific* additional fault information.

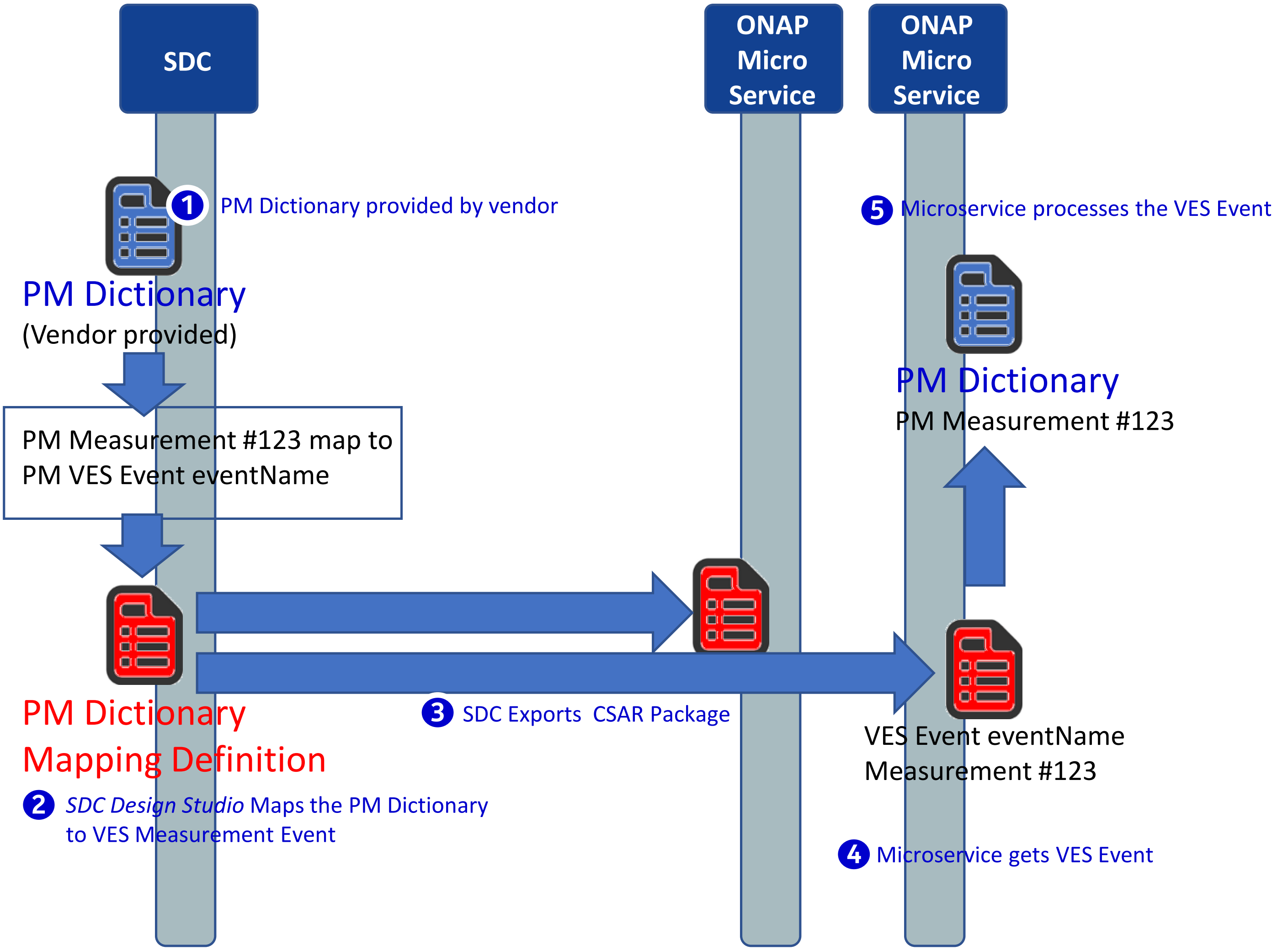


```
event: {presence: required, action: {any, any, AlarmID 71xx RECO-contact Nokia technical support },
comment: "
MEANING = A fault has occurred in the base station synchronization. For example: the base station
reference clock signal is lost or is unstable/inaccurate, ....,
EFFECT = The effect of the fault on the functioning of the network element depends on the fault id
raised.  See fault effect below in registration YAML,
CLEARING TYPE = automatic
",

structure: {
  commonEventHeader: {presence: required, structure: {
    version: {presence: required, value: 3.0},
    domain: {presence: required, value: fault},
    eventName: {presence: required, value: Fault_5gBts_Nokia_baseStationSynchronizationProblem},
    eventId: {presence: required},
    sourceName: {presence: required},
    reportingEntityName: {presence: required},
    priority: {presence: required},
    startEpochMicrosec: {presence: required},
      }},
      faultFields: {presence: required, structure: {
        faultFieldsVersion: {presence: required, value: 3.0},
        eventCategory: {presence: optional, comment: "Quality of service"},
        alarmCondition: {presence: required, value: 'baseStationSynchronizationProblem'},
        eventSourceType: {presence: required},
        alarminterfaceA: {presence: required},
        specificProblem: {presence: required},
        eventSeverity: {presence: required},
        nfStatus: {default: Active},
        alarmAdditionalInformation: {presence: required, array: {
            keyValuePair: {presence: required, structure: {
              key: {presence: required, value: 'Fault ID'},
              value: {presence: required}},
            comment: "
    FAULT ID = 9,
    FAULT NAME = 'BTS time not corrected',
```



```
           comment: "
     FAULT ID = 9,
     FAULT NAME = 'BTS time not corrected',
     FAULT DESCRIPTION = 'The reference frequency that the BTS master clock receives has changed by
     about 200 ppb or more (which equals the change magnitude of 204 DAC steps or more (with 12bit DAC))
     during the measurement period, compared to the BTS master clock frequency.
     Causes can be:
        1. The reference frequency .....
        2. The reference frequency fluctuates ...',
     EFFECT = 'This fault does not immediately affect the operations of the BTS, but it is a notification ...',
     PROPOSED ACTION = 'access the ....follow the instructions below:
        1. In case of a fault in the transmission network synchronization, ...

        3. In case of a BTS equipment fault, the location might be:
       4. After the fault situation has been cleared, ....',
     FAULT ID= 1818,
     FAULT NAME = 'BTS master clock tuning failure',
     FAULT DESCRIPTON = 'Master clock frequency is tuned to within 5% of its minimum or maximum tuning
     limit.',
     EFFECT = 'The BTS can operate properly for months ....
     Effects in Frequency Synchronization mode:  ...
     Effects in Phase Synchronization mode:  ....',
     PROPOSED ACTION = 'Perform the steps below in the listed order until the fault disappears.
     Not tracking satellites:
         1. The most common reason ....
         2. There might be a malfunction in the GPS receiver. Perform a (remote) power reset for the GPS
     receiver.
         3. There might be a HW fault in the GPS receiver. Check the operation and change the GPS module,
     if needed.'
     ",
```

PM Dictionary Usage


SDC
ONAP Micro Service
ONAP Micro Service
1 PM Dictionary provided by vendor
PM Dictionary
(Vendor provided)
PM Measurement #123 map to PM VES Event eventName
PM Dictionary Mapping Definition
2 SDC Design Studio Maps the PM Dictionary to VES Measurement Event
3 SDC Exports CSAR Package
5 Microservice processes the VES Event
PM Dictionary
PM Measurement #123
VES Event eventName Measurement #123
4 Microservice gets VES Event

❶ **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]

❷ **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.

❸ **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

❹ **MICRO SERVICE GETS MEASUREMENT EVENT**

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

❺ **MICRO SERVICE PROCESSES EVENT**

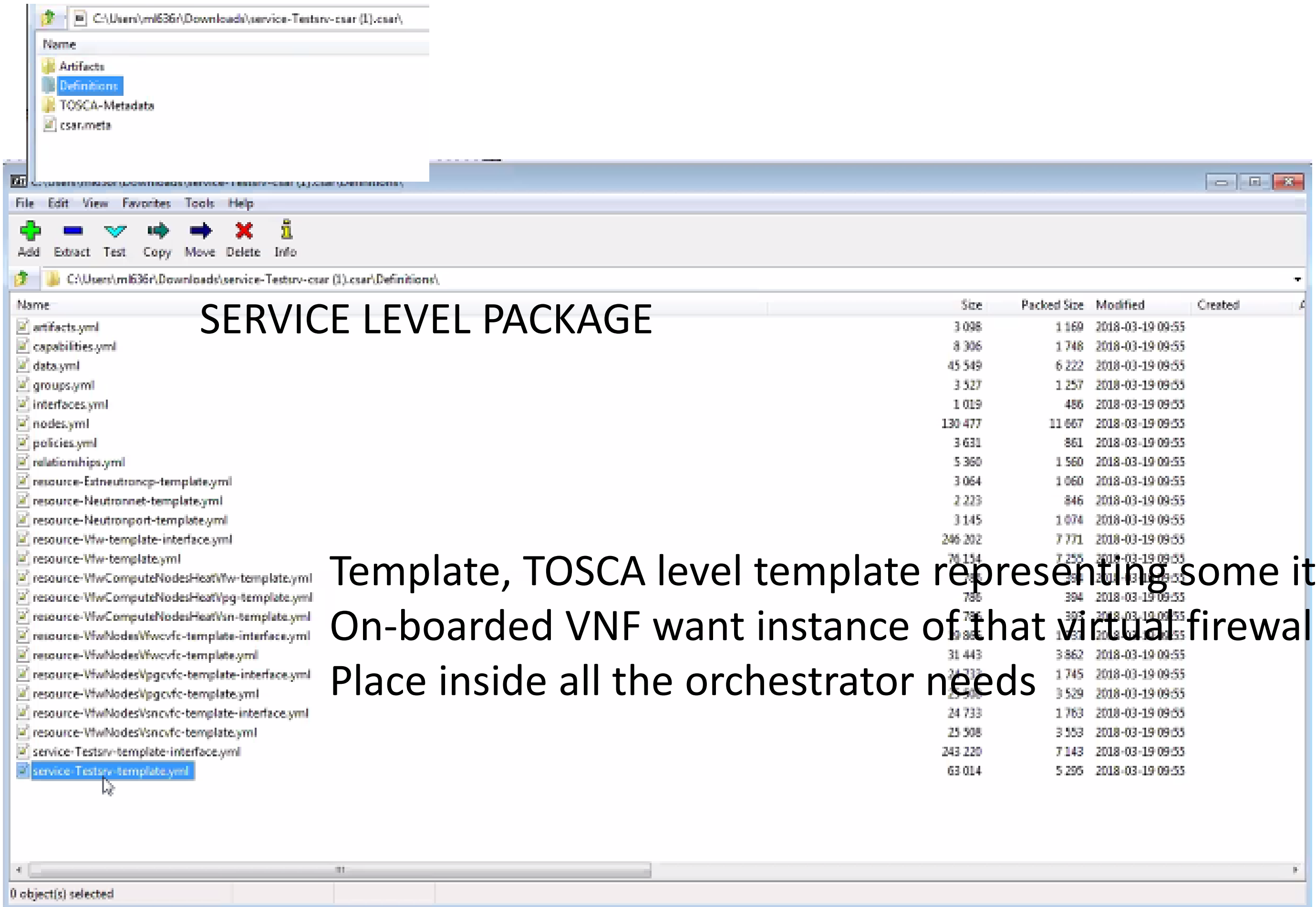
Microservice processes the VES Event using the PM Measurements Dictionary

PM Dictionary Usage

ONAP

OPEN NETWORK AUTOMATION PLATFORM

APPENDIX
& Meeting Notes



C:\Users\ml636r\Downloads\service-Testsrv-csar (1).csar\

Name
- Artifacts
- Definitions
- TOSCA-Metadata
- csar.meta

File Edit View Favorites Tools Help
Add Extract Test Copy Move Delete Info

C:\Users\ml636r\Downloads\service-Testsrv-csar (1).csar\Definitions\

SERVICE LEVEL PACKAGE

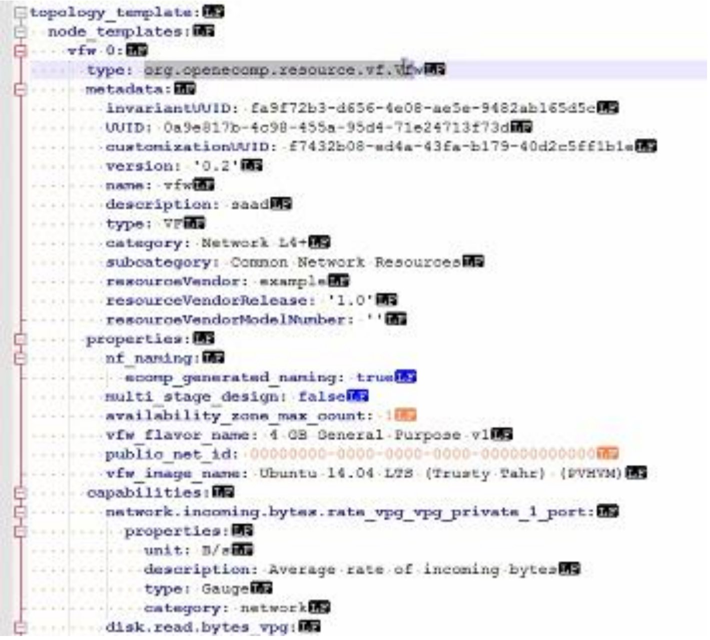
Name	Size	Packed Size	Modified	Created
artifacts.yml	3 098	1 169	2018-03-19 09:55	
capabilities.yml	8 306	1 748	2018-03-19 09:55	
data.yml	45 549	6 222	2018-03-19 09:55	
groups.yml	3 527	1 257	2018-03-19 09:55	
interfaces.yml	1 019	486	2018-03-19 09:55	
nodes.yml	130 477	11 667	2018-03-19 09:55	
policies.yml	3 631	861	2018-03-19 09:55	
relationships.yml	5 360	1 560	2018-03-19 09:55	
resource-Extneutroncp-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	78 134	7 255	2018-03-19 09:55	
resource-VfwComputeNodesHeatVfw-template.yml	[illegible]	394	2018-03-19 09:55	
resource-VfwComputeNodesHeatVpg-template.yml	786	394	2018-03-19 09:55	
resource-VfwComputeNodesHeatVsn-template.yml	786	393	2018-03-19 09:55	
resource-VfwNodesVfwcvfc-template-interface.yml	[illegible]	[illegible]	2018-03-19 09:55	
resource-VfwNodesVfwcvfc-template.yml	31 443	3 862	2018-03-19 09:55	
resource-VfwNodesVpgcvfc-template-interface.yml	24 732	1 745	2018-03-19 09:55	
resource-VfwNodesVpgcvfc-template.yml	[illegible]	3 529	2018-03-19 09:55	
resource-VfwNodesVsncvfc-template-interface.yml	24 733	1 763	2018-03-19 09:55	
resource-VfwNodesVsncvfc-template.yml	25 308	3 553	2018-03-19 09:55	
service-Testsrv-template-interface.yml	243 220	7 143	2018-03-19 09:55	
service-Testsrv-template.yml	63 014	5 295	2018-03-19 09:55	

Template, TOSCA level template representing some iter

On-boarded VNF want instance of that virtual firewall

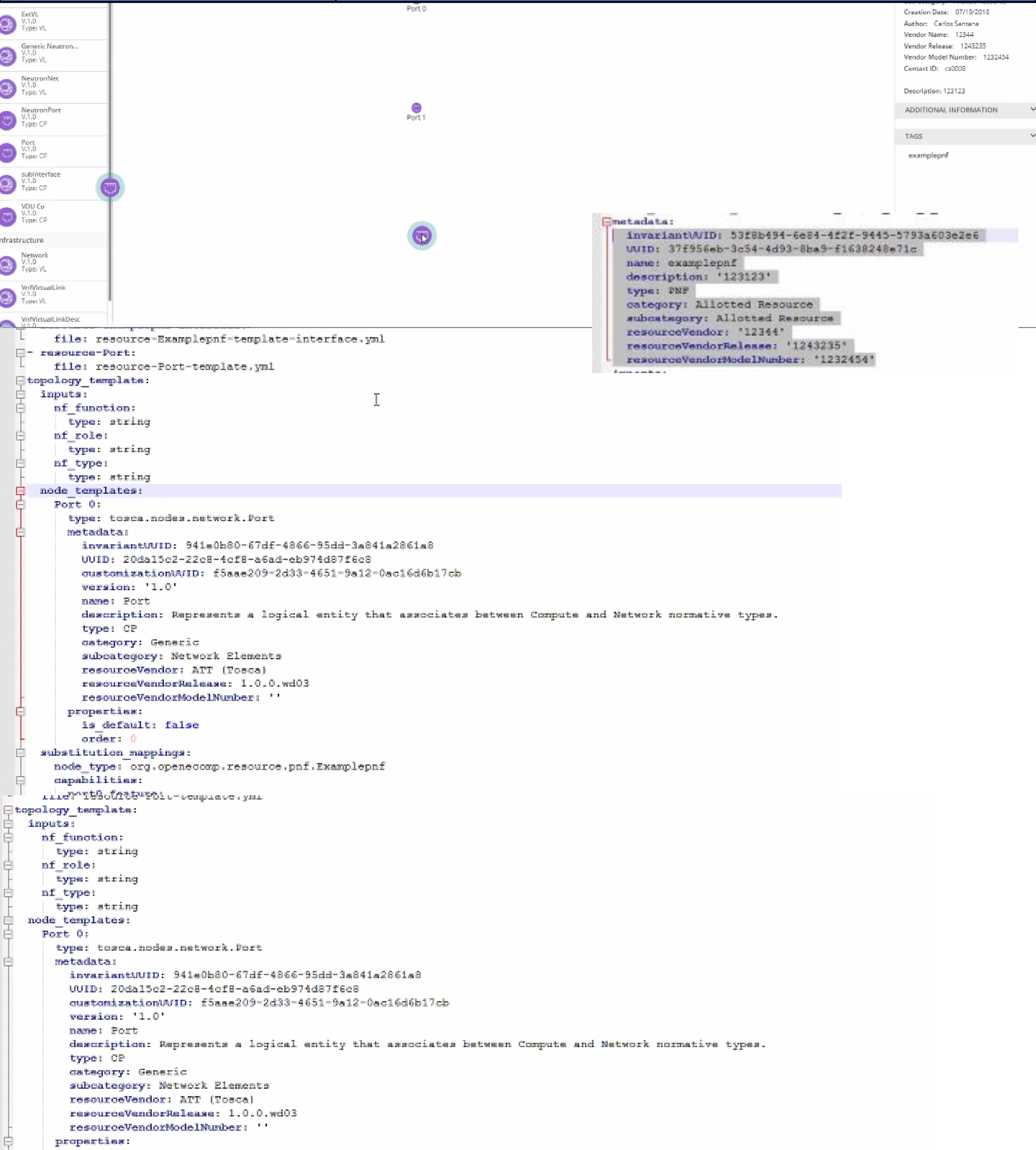
Place inside all the orchestrator needs

0 object(s) selected



```
topology_template:
  node_templates:
    vfw 0:
      type: org.openecomp.resource.vf.Vfw
      metadata:
        invariantUUID: fa9f72b3-d656-4e08-ae5e-9482ab165d5c
        UUID: 0a9e817b-4c98-455a-95d4-71e24713f73d
        customizationUUID: f7432b08-ed4a-43fa-b179-40d2c5ff1b1e
        version: '0.2'
        name: vfw
        description: saad
        type: VF
        category: Network L4+
        subcategory: Common Network Resources
        resourceVendor: example
        resourceVendorRelease: '1.0'
        resourceVendorModelNumber: ''
      properties:
        nf_naming:
          ecomp_generated_naming: true
        multi_stage_design: false
        availability_zone_max_count: 1
        vfw_flavor_name: 4 GB General Purpose v1
        public_net_id: 00000000-0000-0000-0000-000000000000
        vfw_image_name: Ubuntu 14.04 LTS (Trusty Tahr) (PVHVM)
      capabilities:
        network.incoming.bytes.rate_vpg_vpg_private_1_port:
          properties:
            unit: B/s
            description: Average rate of incoming bytes
            type: Gauge
            category: network
        disk.read.bytes_vpg:
```

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.



Creation Date: 07/19/2018
Author: Carlos Santana
Vendor Name: 12344
Vendor Release: 1243235
Vendor Model Number: 1232454
Contact ID: cs0008

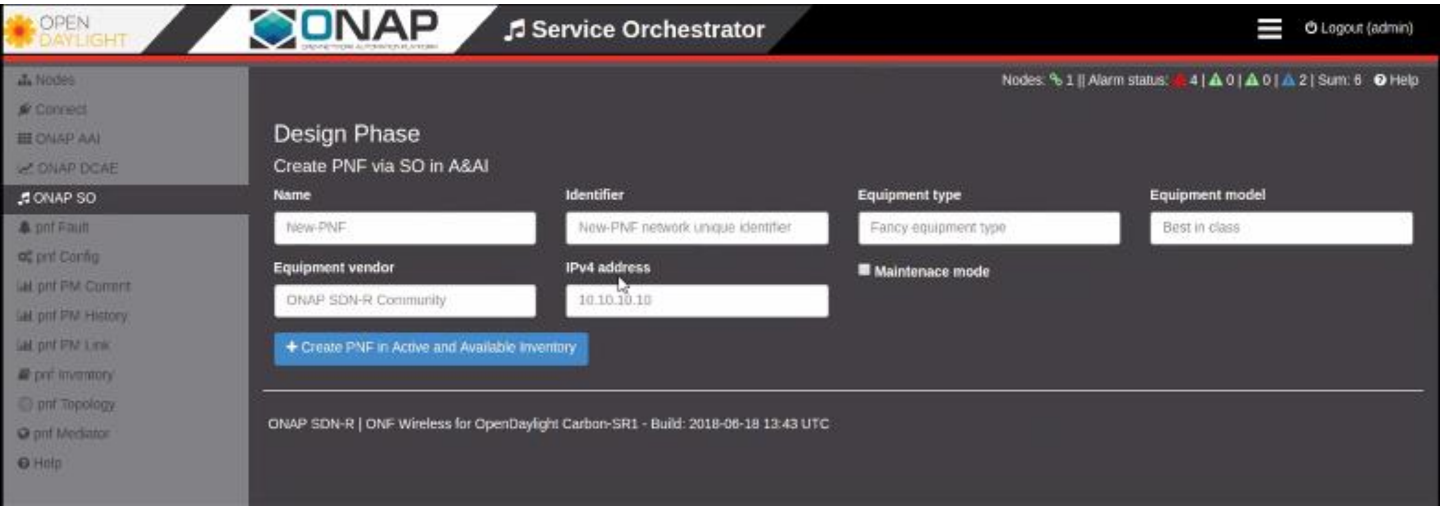
Description: 123123

examplepnf

```
metadata:
  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'
```

```
    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: tosca.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.openecomp.resource.pnf.Examplepnf
    capabilities:
```

```
topology_template:
  inputs:
    nf_function:
      type: string
    nf_role:
      type: string
    nf_type:
      type: string
  node_templates:
    Port 0:
      type: tosca.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:
```



Design Phase

Create PNF via SO in A&AI

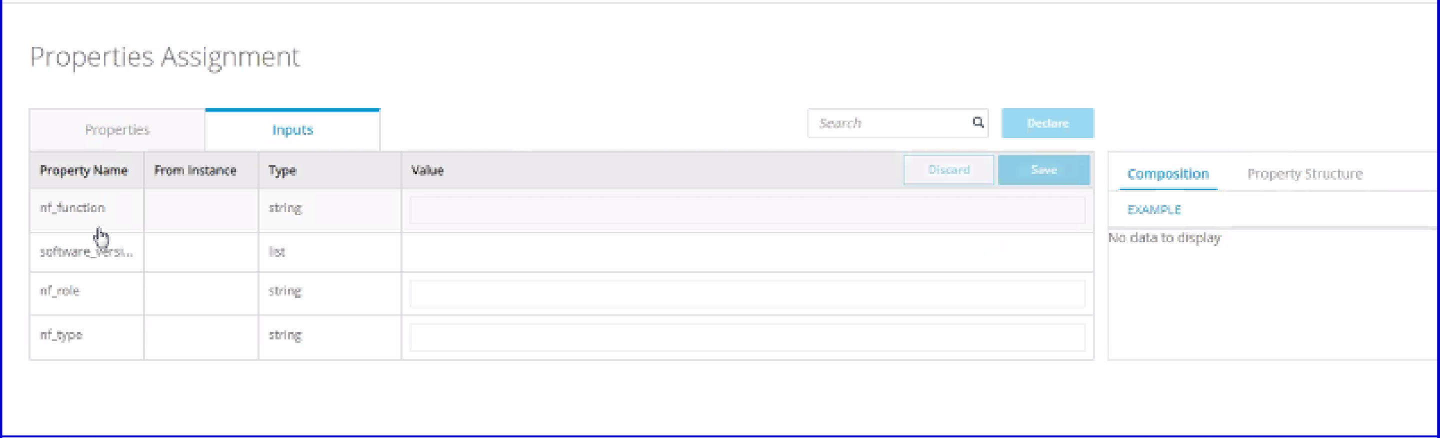
Name	**Identifier**	**Equipment type**	**Equipment model**
New-PNF	New-PNF network unique identifier	Fancy equipment type	Best in class
Equipment vendor	**IPv4 address**	□ **Maintenace mode**	
ONAP SDN-R Community	10.10.10.10		

ONAP SDN-R | ONF Wireless for OpenDaylight Carbon-SR1 - Build: 2018-06-18 13:43 UTC

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

software_versions (SDC Demo)

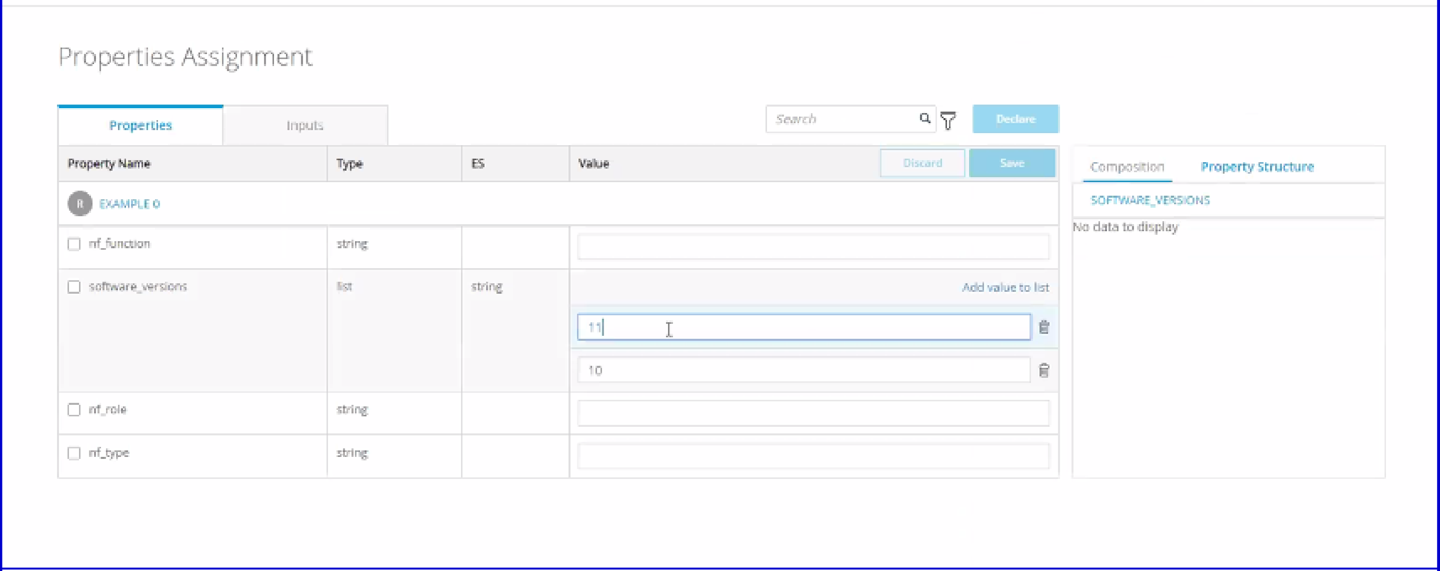
Demo given by Michael Lando (Aug 21, 2018)
Demonstrating software_versions in SDC Design Studio



Properties Assignment

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

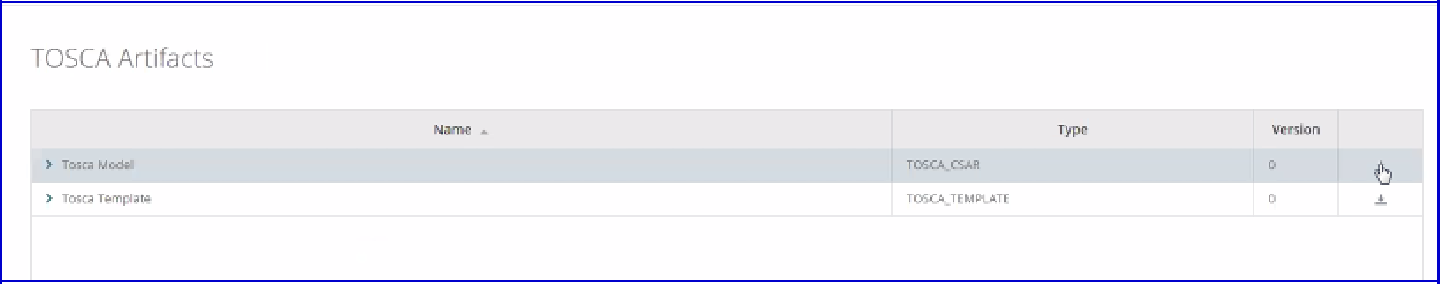
No data to display



Properties Assignment

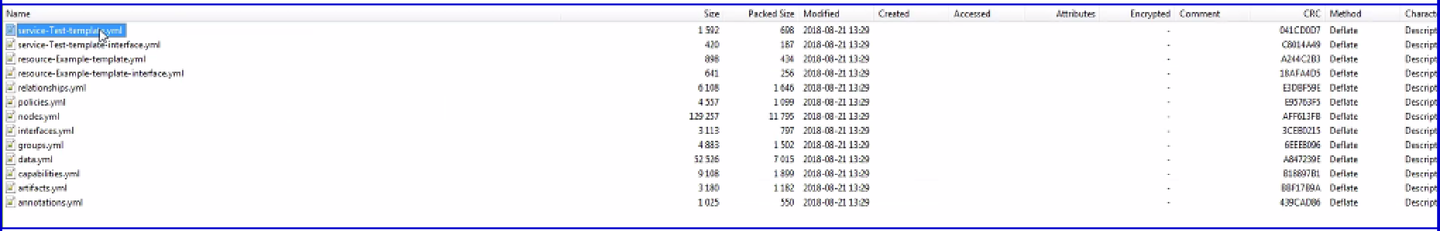
Property Name	Type	ES	Value
EXAMPLE 0			
□ nf_function	string		
□ software_versions	list	string	Add value to list 11 10
□ nf_role	string		
□ nf_type	string		

No data to display



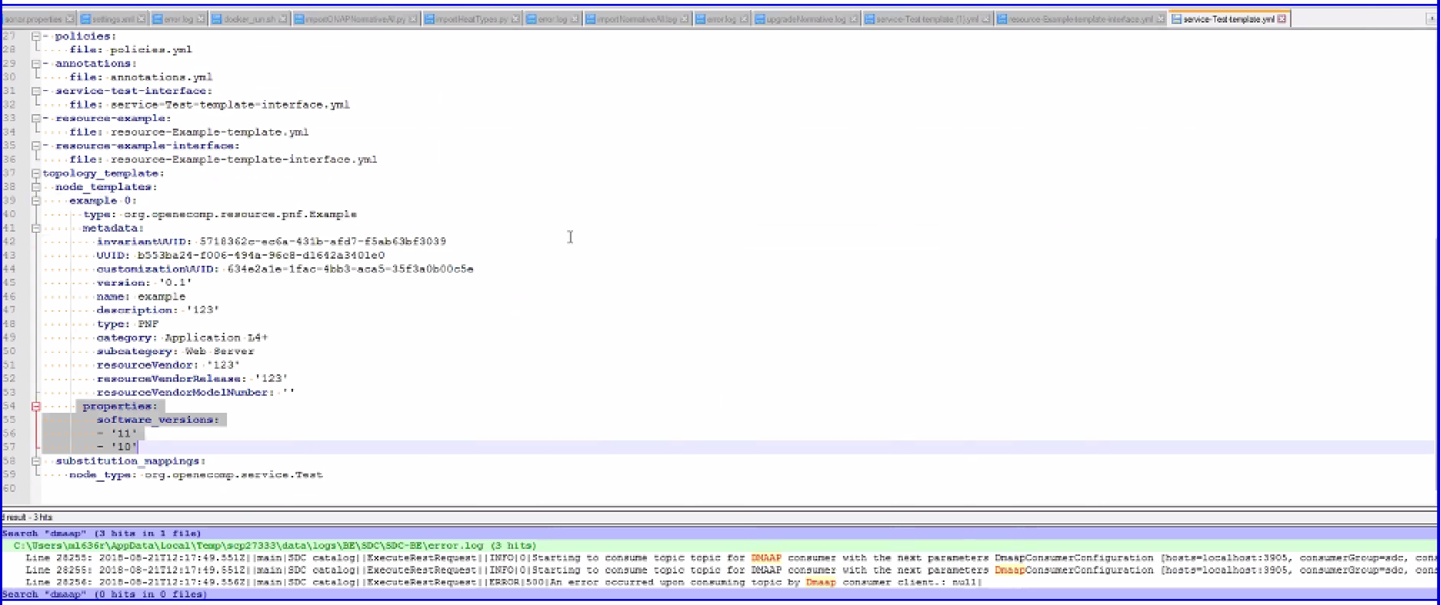
TOSCA Artifacts

Name	Type	Version	
Tosca Model	TOSCA_CSAR	0	
Tosca Template	TOSCA_TEMPLATE	0	



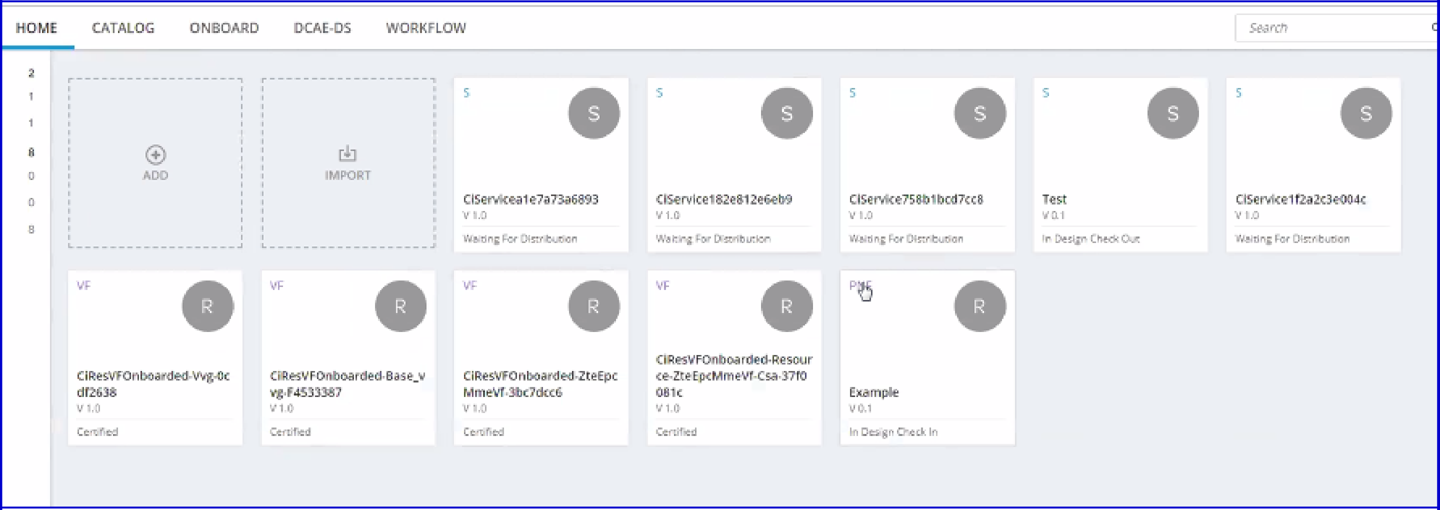
Name	Size	Packed Size	Modified	Created	Accessed	Attributes	Encrypted	Comment	CRC	Method	Charact
service-Test-template.yml	1 592	608	2018-08-21 13:29				-		041CD0D7	Deflate	Descript
service-Test-template-interface.yml	420	187	2018-08-21 13:29				-		C8014A49	Deflate	Descript
resource-Example-template.yml	898	434	2018-08-21 13:29				-		A244C2B3	Deflate	Descript
resource-Example-template-interface.yml	641	256	2018-08-21 13:29				-		18AFA4D5	Deflate	Descript
relationships.yml	6 108	1 646	2018-08-21 13:29				-		E3D8F59E	Deflate	Descript
policies.yml	4 557	1 099	2018-08-21 13:29				-		E95763F5	Deflate	Descript
nodes.yml	129 257	11 795	2018-08-21 13:29				-		AFF613FB	Deflate	Descript
interfaces.yml	3 113	797	2018-08-21 13:29				-		3CEB0215	Deflate	Descript
groups.yml	4 883	1 502	2018-08-21 13:29				-		6EEEB006	Deflate	Descript
data.yml	52 526	7 015	2018-08-21 13:29				-		A847239E	Deflate	Descript
capabilities.yml	9 108	1 899	2018-08-21 13:29				-		B18897B1	Deflate	Descript
artifacts.yml	3 180	1 182	2018-08-21 13:29				-		BBF1709A	Deflate	Descript
annotations.yml	1 025	550	2018-08-21 13:29				-		439CAD86	Deflate	Descript

software_versions (SDC Demo)



```
- policies:
    file: policies.yml
- annotations:
    file: annotations.yml
- service-test-interface:
    file: service-Test-template-interface.yml
- resource-example:
    file: resource-Example-template.yml
- resource-example-interface:
    file: resource-Example-template-interface.yml
topology_template:
  node_templates:
    example 0:
      type: org.openecomp.resource.pnf.Example
      metadata:
        invariantUUID: 5718362c-ec6a-431b-afd7-f5ab63bf3039
        UUID: b553ba24-f006-494a-96c8-d1642a3401e0
        customizationUUID: 634e2a1e-1fac-4bb3-aca5-35f3a0b00c5e
        version: '0.1'
        name: example
        description: '123'
        type: PNF
        category: Application L4+
        subcategory: Web Server
        resourceVendor: '123'
        resourceVendorRelease: '123'
        resourceVendorModelNumber: ''
      properties:
        software_versions:
        - '11'
        - '10'
  substitution_mappings:
    node_type: org.openecomp.service.Test
```

```
d result - 3 hits
Search "dmaap" (3 hits in 1 file)
  C:\Users\ml636r\AppData\Local\Temp\scp27333\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 topic for DMAAP consumer with the next parameters DmaapConsumerConfiguration [hosts=localhost:3905, consumerGroup=sdc, cons
    Line 28255: 2018-08-21T12:17:49.551Z||main|SDC catalog||ExecuteRestRequest||INFO|0|Starting to consume topic topic for DMAAP consumer with the next parameters DmaapConsumerConfiguration [hosts=localhost:3905, consumerGroup=sdc, cons
    Line 28256: 2018-08-21T12:17:49.556Z||main|SDC catalog||ExecuteRestRequest||ERROR|500|An error occurred upon consuming topic by Dmaap consumer client.: null|
Search "dmaap" (0 hits in 0 files)
```



CIServicea1e7a73a6893
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

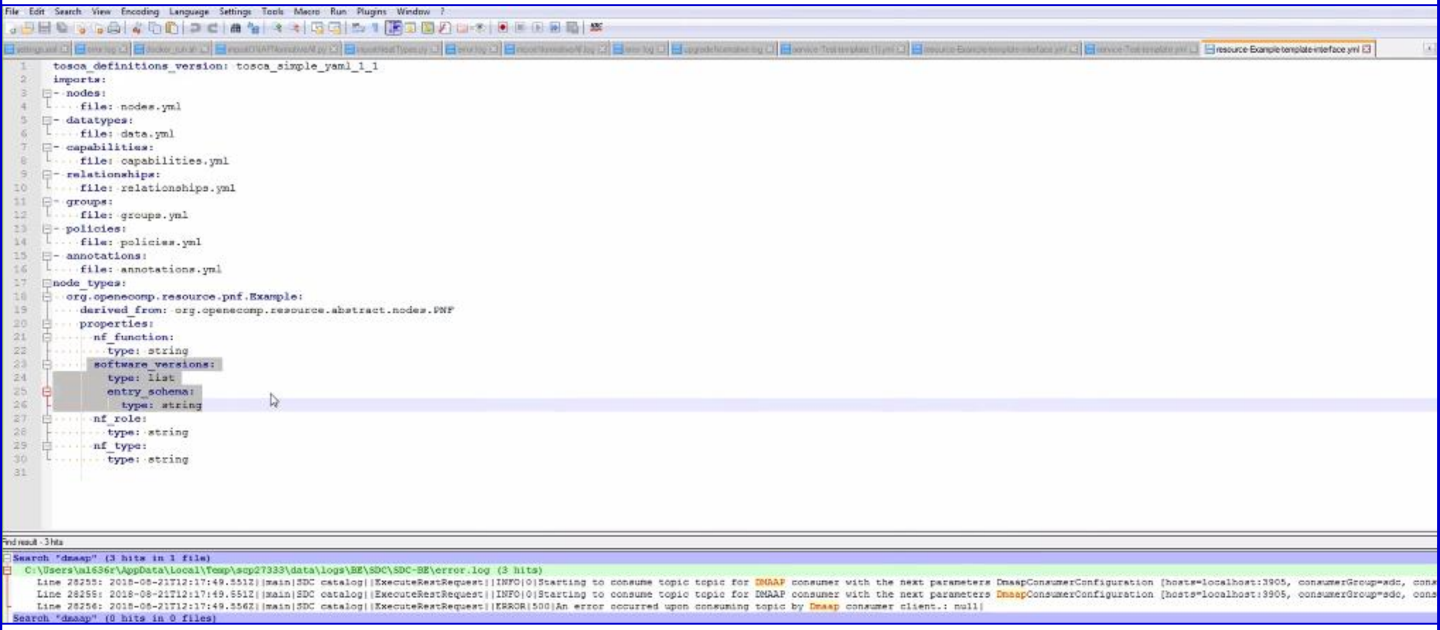
CIResVFOnboarded-Vvg-0cdf2638
V 1.0
Certified

CIResVFOnboarded-Base_vvg-F4533387
V 1.0
Certified

CIResVFOnboarded-ZteEpcMmeVf-3bc7dcc6
V 1.0
Certified

CIResVFOnboarded-Resource-ZteEpcMmeVf-Csa-37f0081c
V 1.0
Certified

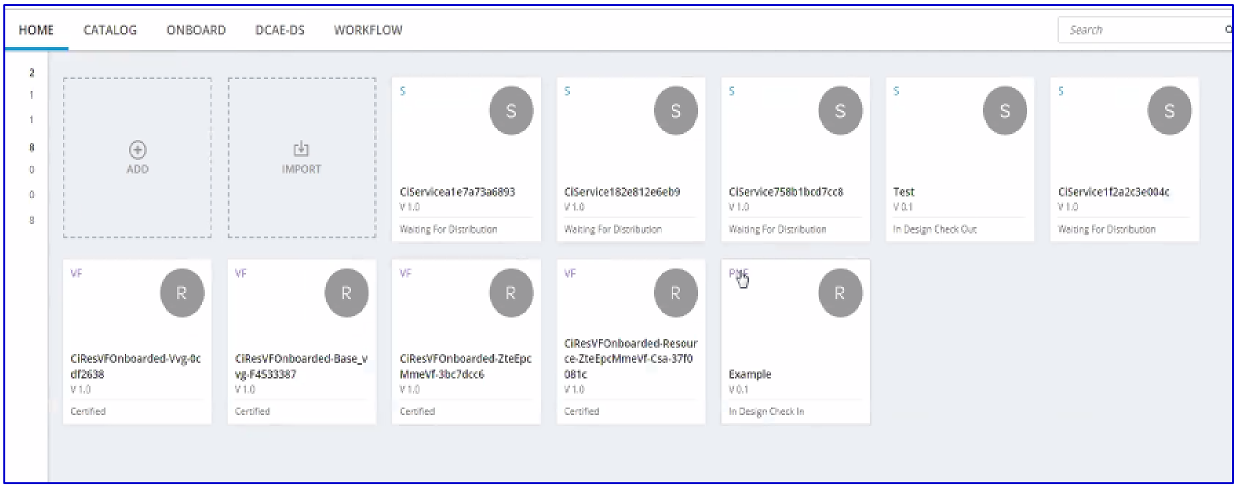
Example
V 0.1
In Design Check In



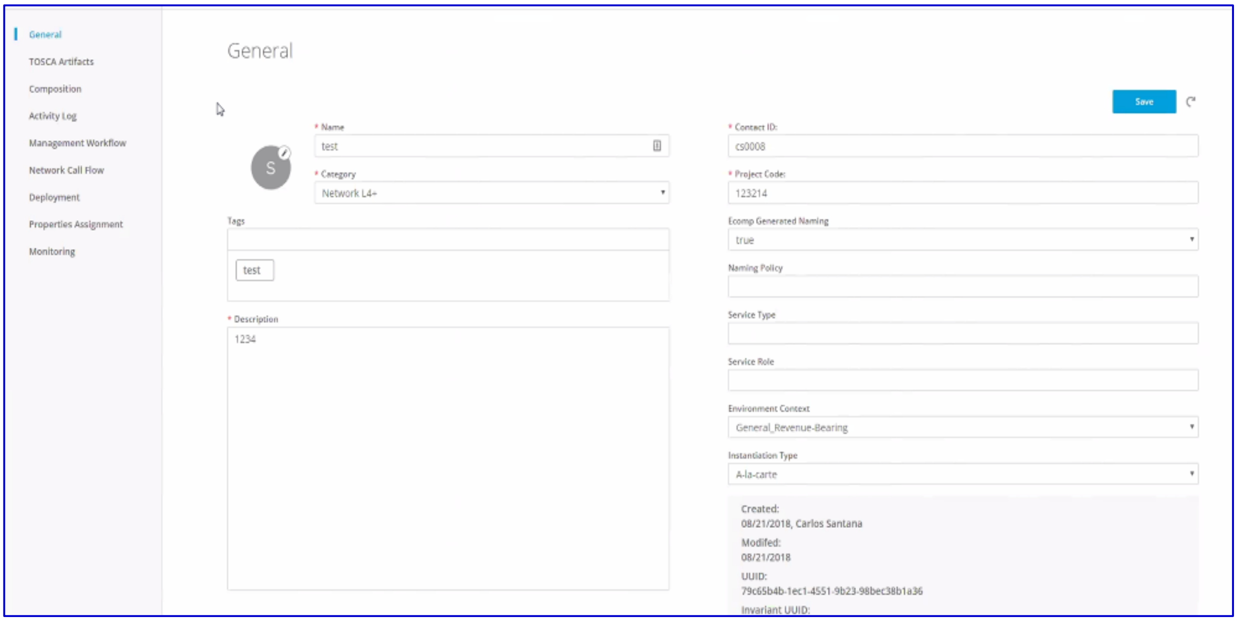
```
tosca_definitions_version: tosca_simple_yaml_1_1
imports:
- nodes:
    file: nodes.yml
- datatypes:
    file: data.yml
- capabilities:
    file: capabilities.yml
- relationships:
    file: relationships.yml
- groups:
    file: groups.yml
- policies:
    file: policies.yml
- annotations:
    file: annotations.yml
node_types:
  org.openecomp.resource.pnf.Example:
    derived_from: org.openecomp.resource.abstract.nodes.PNF
    properties:
      nf_function:
        type: string
      software_versions:
        type: list
        entry_schema:
          type: string
      nf_role:
        type: string
      nf_type:
        type: string
```

```
Find result - 3 hits
Search "dmaap" (3 hits in 1 file)
  C:\Users\ml636r\AppData\Local\Temp\scp27333\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 topic for DMAAP consumer with the next parameters DmaapConsumerConfiguration [hosts=localhost:3905, consumerGroup=sdc, cons
    Line 28255: 2018-08-21T12:17:49.551Z||main|SDC catalog||ExecuteRestRequest||INFO|0|Starting to consume topic topic for DMAAP consumer with the next parameters DmaapConsumerConfiguration [hosts=localhost:3905, consumerGroup=sdc, cons
    Line 28256: 2018-08-21T12:17:49.556Z||main|SDC catalog||ExecuteRestRequest||ERROR|500|An error occurred upon consuming topic by Dmaap consumer client.: null|
Search "dmaap" (0 hits in 0 files)
```

software_versions (SDC Demo)



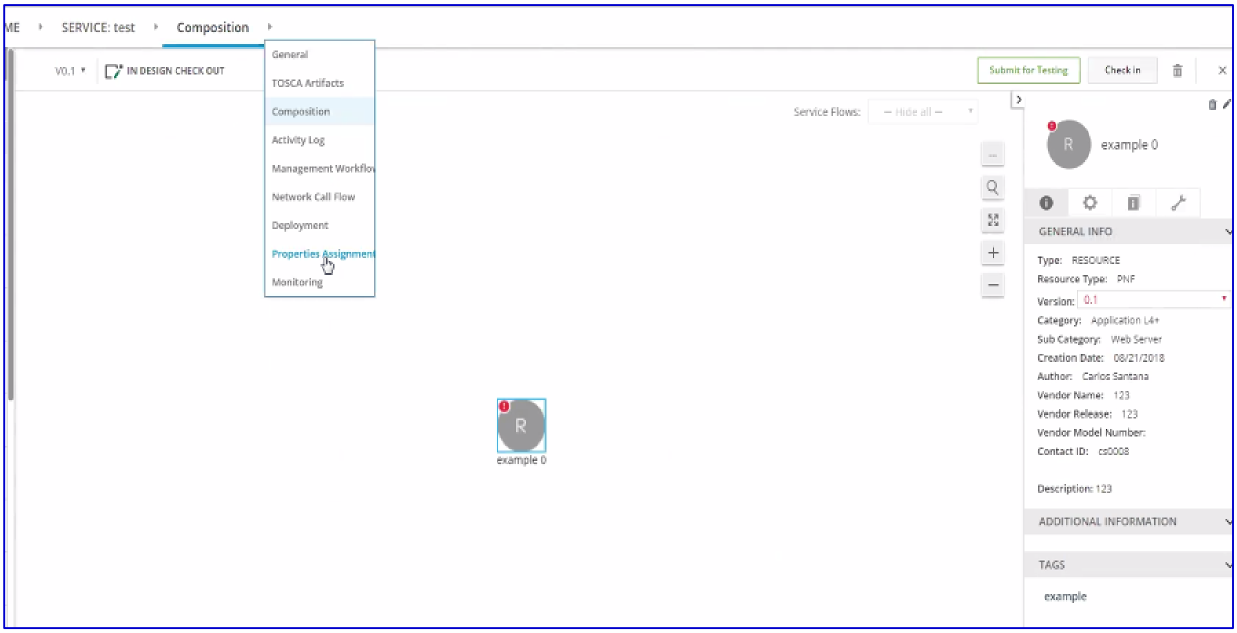
Type	Name	Version	Status
S	CiServicea1e7a73a6893	V 1.0	Waiting For Distribution
S	CiService182e812e6eb9	V 1.0	Waiting For Distribution
S	CiService758b1bcd7cc8	V 1.0	Waiting For Distribution
S	Test	V 0.1	In Design Check Out
S	CiService1f2a2c3e004c	V 1.0	Waiting For Distribution
VF	CiResVFOnboarded-Vvg-0cdf2638	V 1.0	Certified
VF	CiResVFOnboarded-Base_vvg-F4533387	V 1.0	Certified
VF	CiResVFOnboarded-ZteEpcMmeVf-3bc7dcc6	V 1.0	Certified
VF	CiResVFOnboarded-Resource-ZteEpcMmeVf-Csa-37f0081c	V 1.0	Certified
PNF	Example	V 0.1	In Design Check In



General

- * Name: test
- * Category: Network L4+
- Tags: test
- * Description: 1234
- * 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
Modifed:
08/21/2018
UUID:
79c65b4b-1ec1-4551-9b23-98bec38b1a36
Invariant UUID:



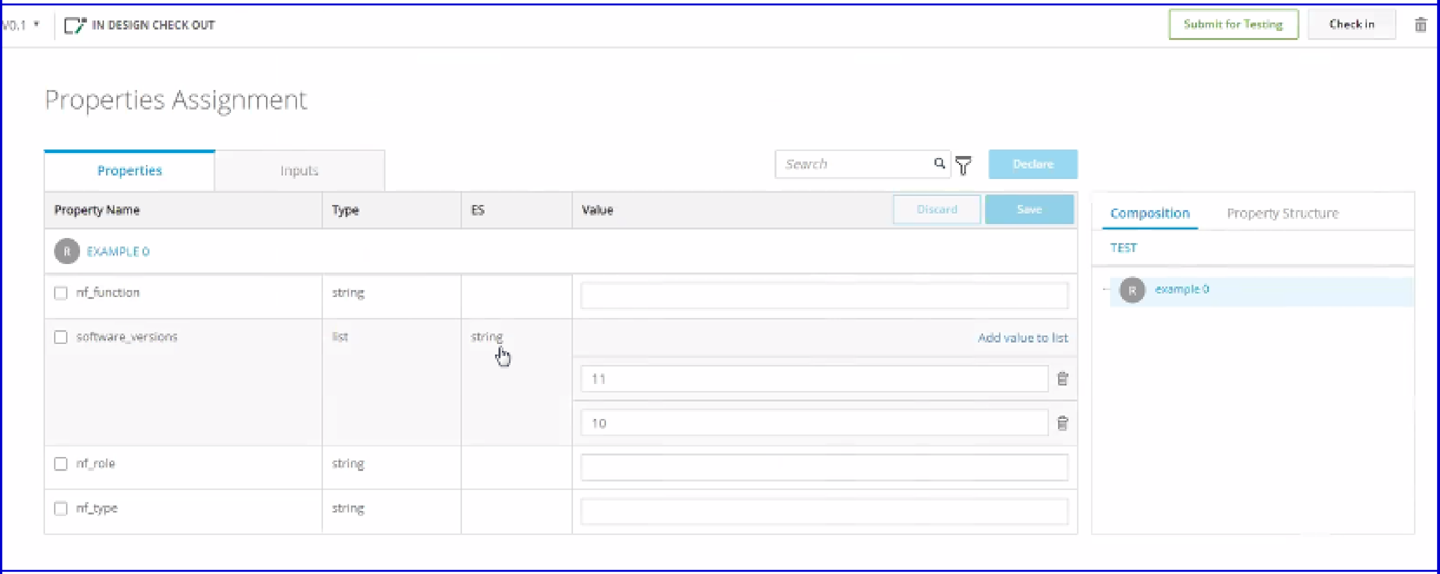
example 0

Type: RESOURCE
Resource Type: PNF
Version: 0.1
Category: Application L4+
Sub Category: Web Server
Creation Date: 08/21/2018
Author: Carlos Santana
Vendor Name: 123
Vendor Release: 123
Vendor Model Number:
Contact ID: cs0008

Description: 123

TAGS
example

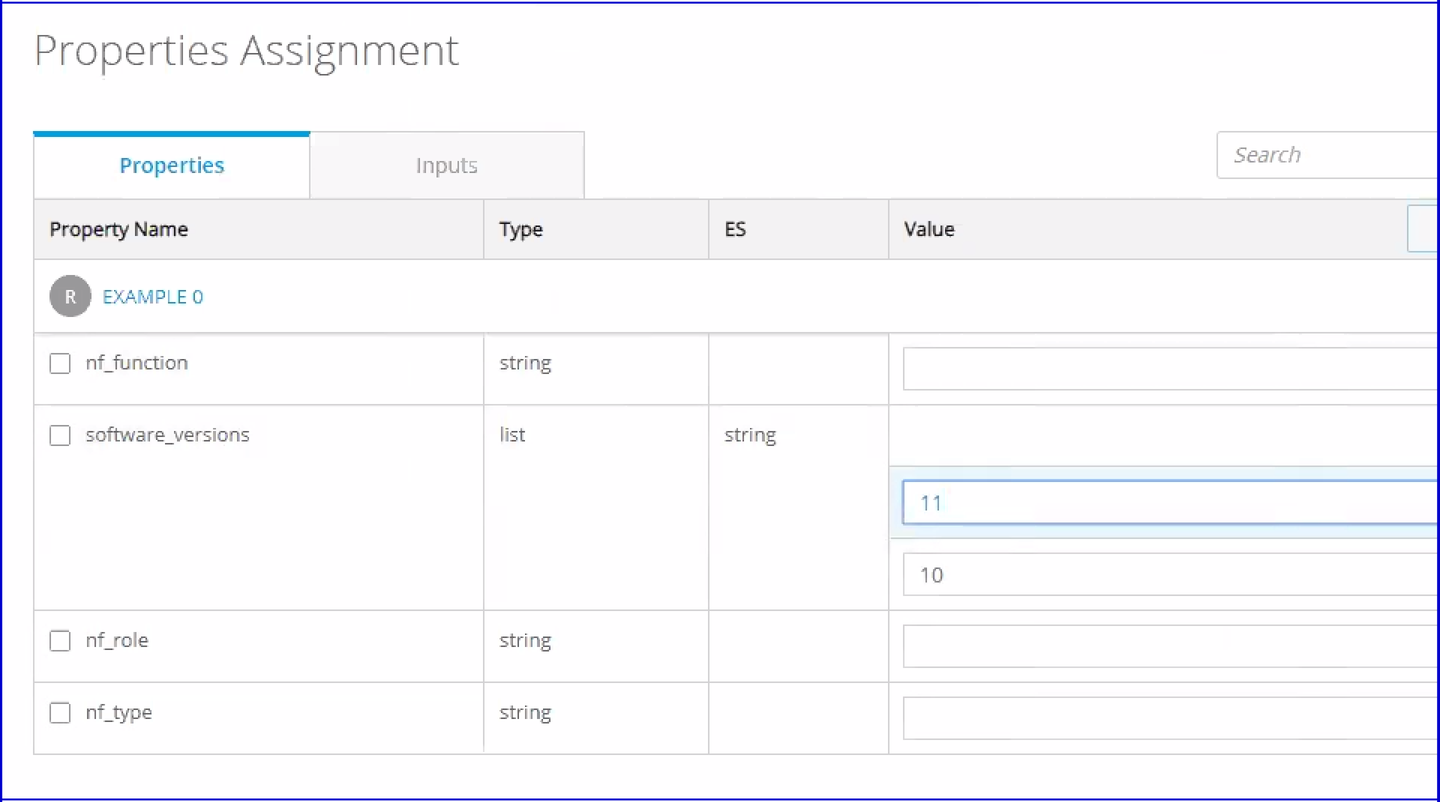
software_versions (SDC Demo)



Properties Assignment

Properties | Inputs

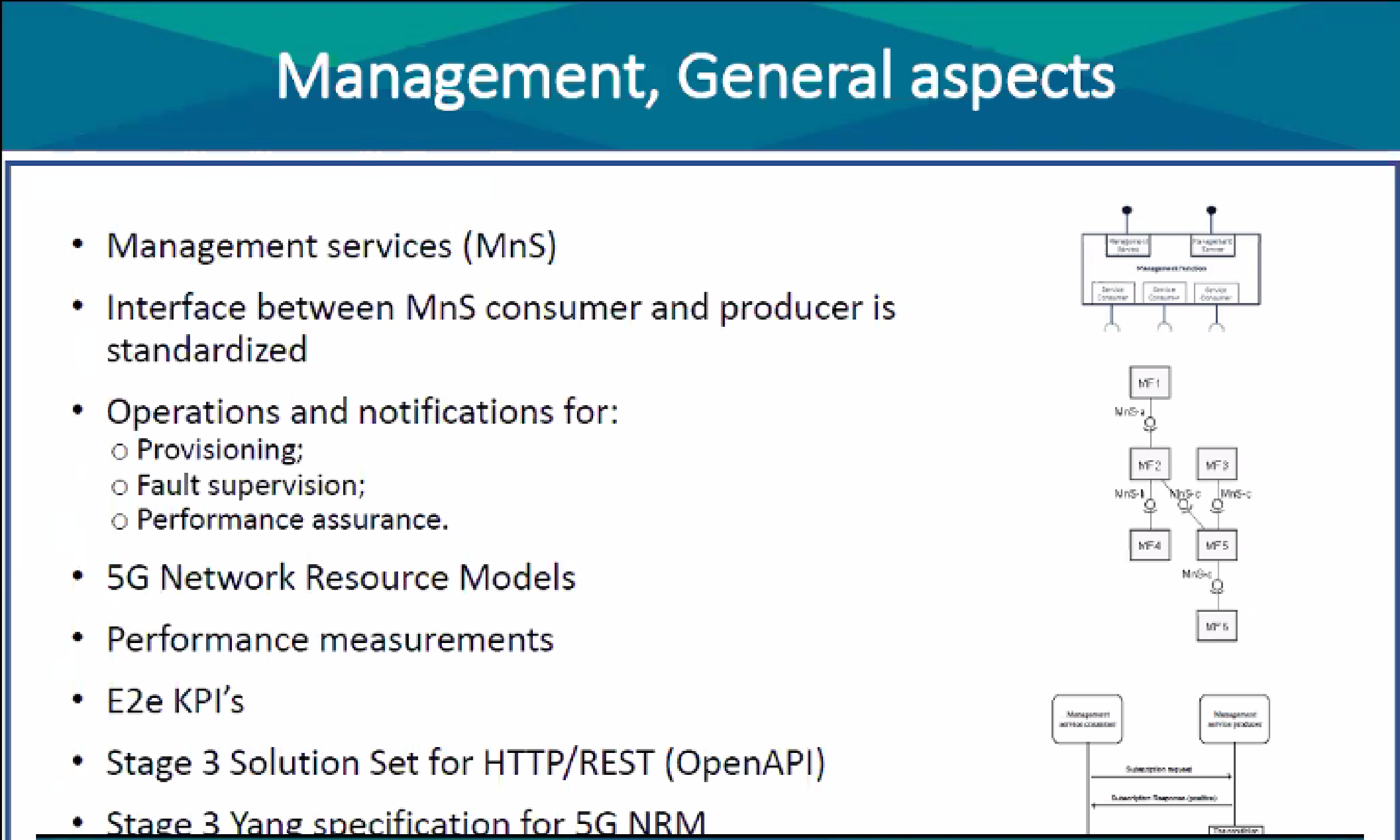
Property Name	Type	ES	Value
R EXAMPLE 0			
□ nf_function	string		
□ software_versions	list	string	Add value to list 11 10
□ nf_role	string		
□ nf_type	string		



Properties Assignment

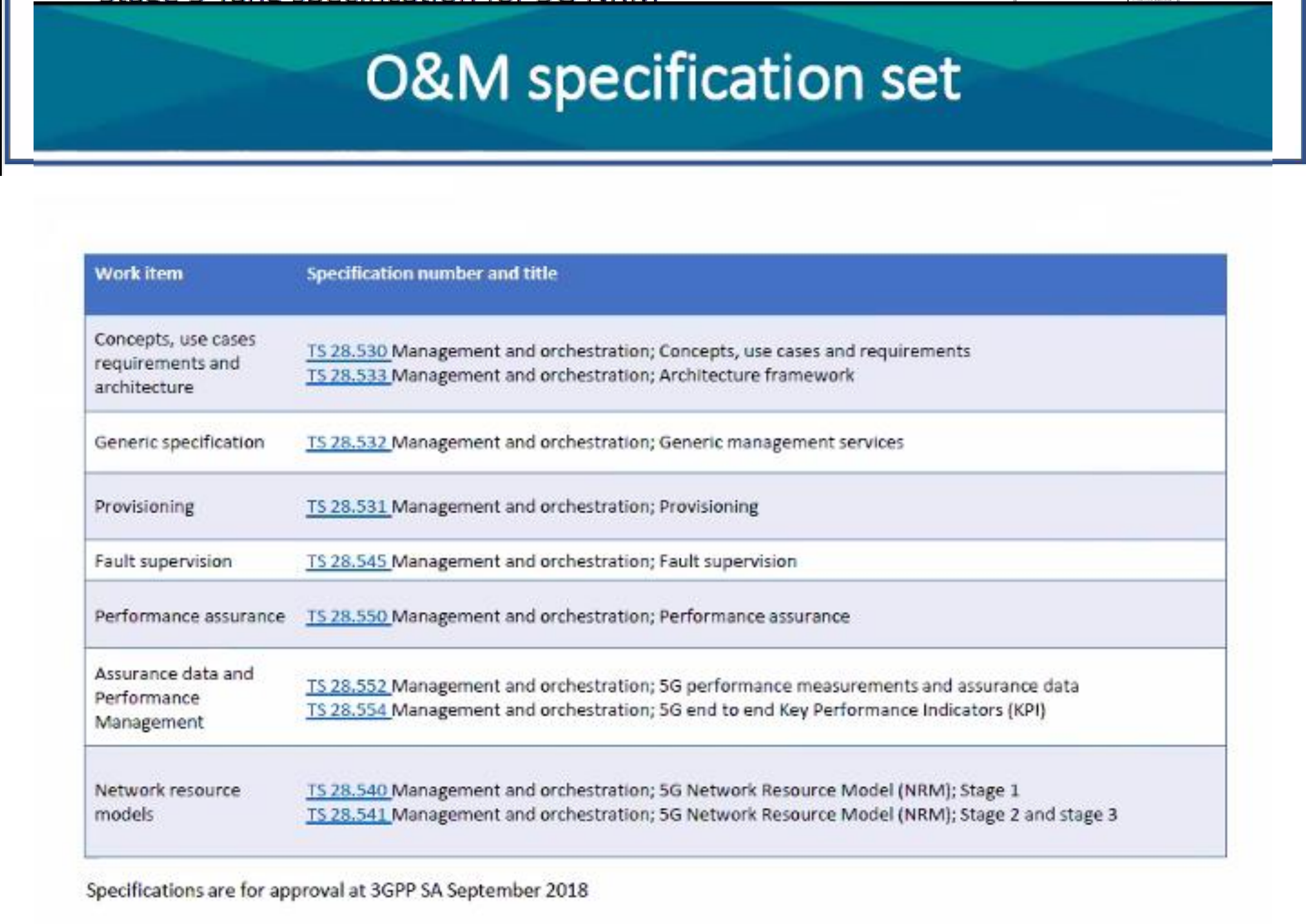
Properties | Inputs

Property Name	Type	ES	Value
R EXAMPLE 0			
□ nf_function	string		
□ software_versions	list	string	11 10
□ nf_role	string		
□ nf_type	string		



Management, General aspects

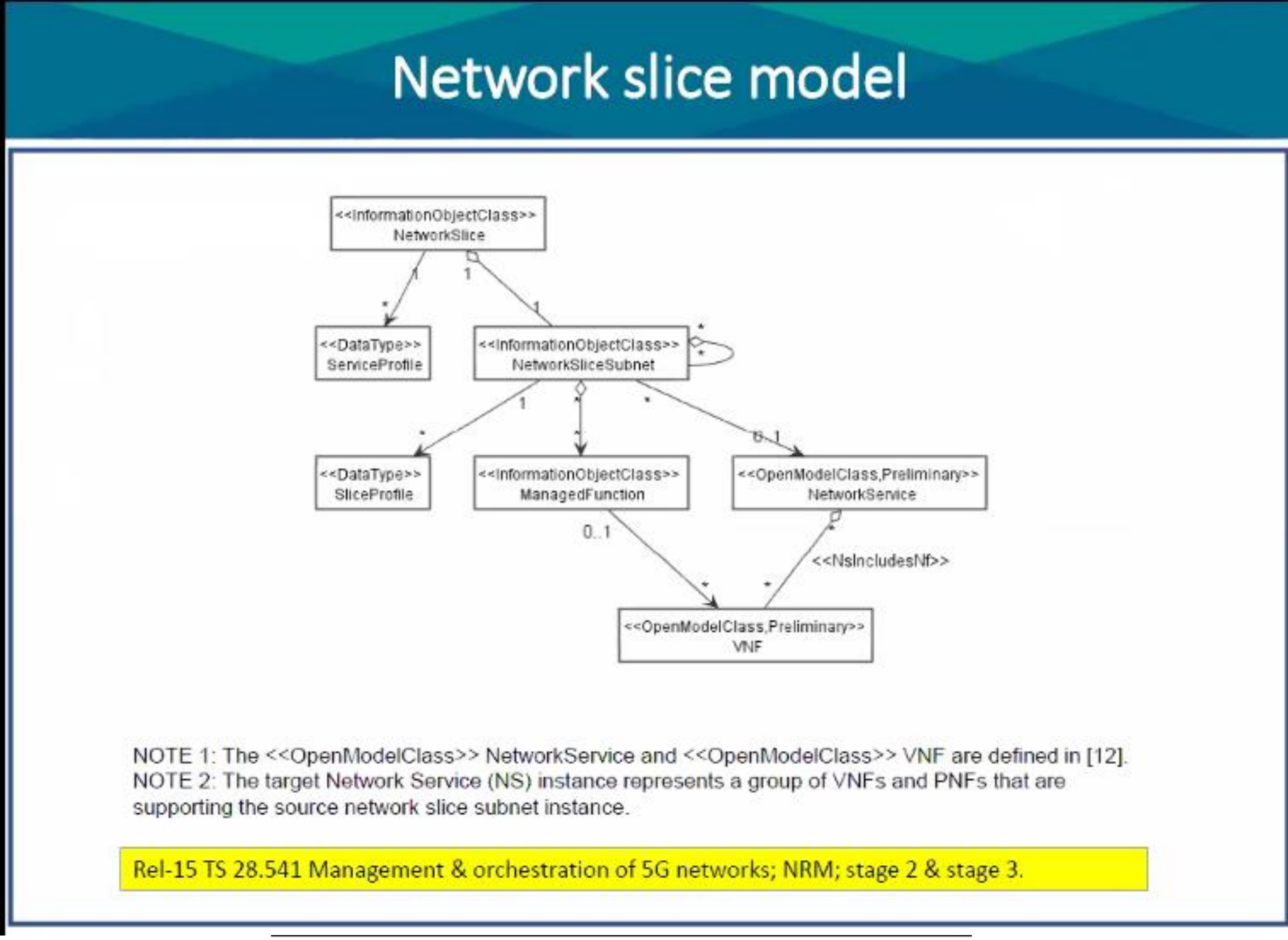
- Management services (MnS)
- Interface between MnS consumer and producer is standardized
- Operations and notifications for:
  - ○ Provisioning;
  - ○ Fault supervision;
  - ○ Performance assurance.
- 5G Network Resource Models
- Performance measurements
- E2e KPI's
- Stage 3 Solution Set for HTTP/REST (OpenAPI)
- Stage 3 Yang specification for 5G NRM



O&M specification set

Work item	Specification number and title
Concepts, use cases requirements and architecture	TS 28.530 Management and orchestration; Concepts, use cases and requirements TS 28.533 Management and orchestration; Architecture framework
Generic specification	TS 28.532 Management and orchestration; Generic management services
Provisioning	TS 28.531 Management and orchestration; Provisioning
Fault supervision	TS 28.545 Management and orchestration; Fault supervision
Performance assurance	TS 28.550 Management and orchestration; Performance assurance
Assurance data and Performance Management	TS 28.552 Management and orchestration; 5G performance measurements and assurance data TS 28.554 Management and orchestration; 5G end to end Key Performance Indicators (KPI)
Network resource models	TS 28.540 Management and orchestration; 5G Network Resource Model (NRM); Stage 1 TS 28.541 Management and orchestration; 5G Network Resource Model (NRM); Stage 2 and stage 3

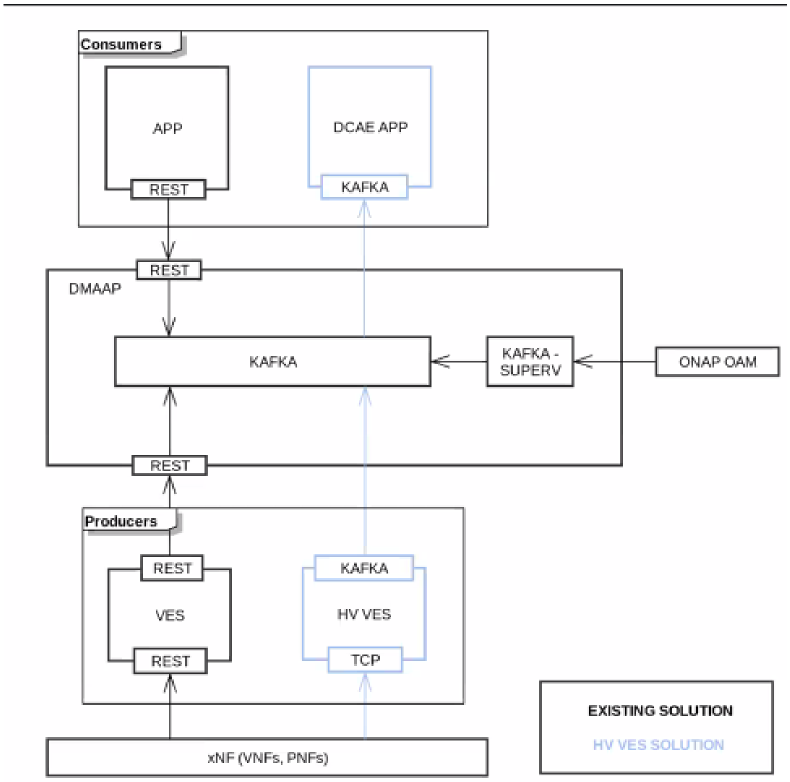
Specifications are for approval at 3GPP SA September 2018



Network slice model

NOTE 1: The <<OpenModelClass>> NetworkService and <<OpenModelClass>> VNF are defined in [12].
NOTE 2: The target Network Service (NS) instance represents a group of VNFs and PNFs that are supporting the source network slice subnet instance.

Rel-15 TS 28.541 Management & orchestration of 5G networks; NRM; stage 2 & stage 3.



Interface Types

R-XXXXX: The VNFD provided by VNF vendor may use the below described TOSCA interface types. An on-boarding entity (ONAP SDC) MUST support them.

tosca.interfaces.nfv.vnf.lifecycle.Nfv supports LCM operations

5.1.9. HPA Requirements

5.1.10. VES Requirements

CSAR provided by VNF vendor may include the ONAP proprietary extensions supported by SOL004 standard. The ONAP extensions should be registered in ETSI as specified in SOL004.

R-XXXXX: The CSAR extensions to support ONAP VES shall be described in both CSAR archive and manifest file as following:

- the CSAR includes:

!------Lfnetworkingonap

!----- ves.yaml

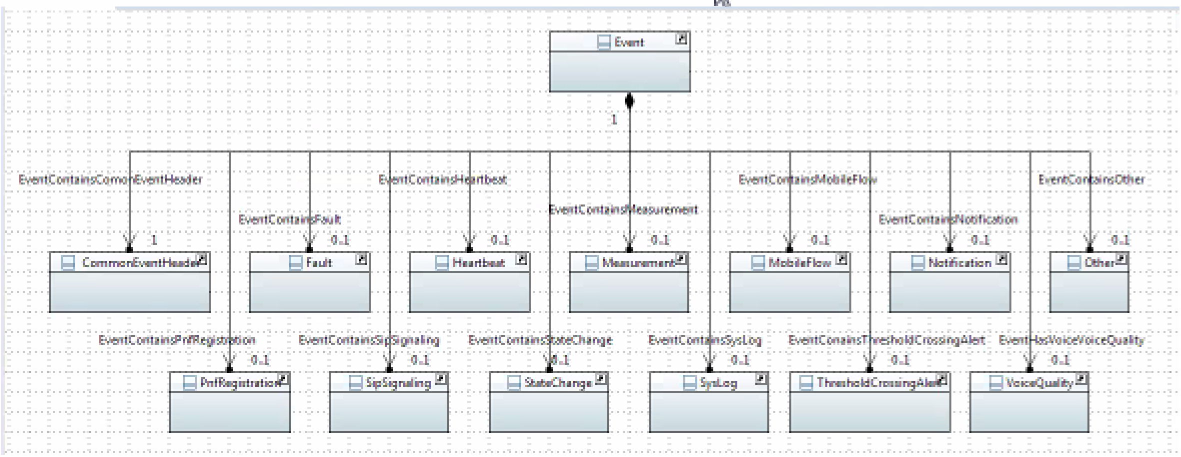
- the manifest file includes:

non_mano_artifacts_sets:

lfn_onap:

Source: Lfnetworkingonap/ves.yaml

Note: The VES extension is not supported in Casablanca release


Event
1
EventContainsComonEventHeader
EventContainsFault
EventContainsHeartbeat
EventContainsMeasurement
EventContainsMobileFlow
EventContainsNotification
EventContainsOther
1
0..1
0..1
0..1
0..1
0..1
0..1
CommonEventHeader
Fault
Heartbeat
Measurement
MobileFlow
Notification
Other
EventContainsPnfRegistration
EventContainsSipSignaling
EventContainsStateChange
EventContainsSysLog
EventConainsThresholdCrossingAlert
EventHasVoiceVoiceQuality
0..1
0..1
0..1
0..1
0..1
0..1
PnfRegistration
SipSignaling
StateChange
SysLog
ThresholdCrossingAlert
VoiceQuality