



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

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

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 |
| | |

PNF PnP: MODELING ENHANCEMENTS

DESCRIPTION

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



PROJECTS

SDC, CDT

PNF ONBOARDING / PNF PACKAGE

DESCRIPTION

PNF Onboarding and PNF Package

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

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



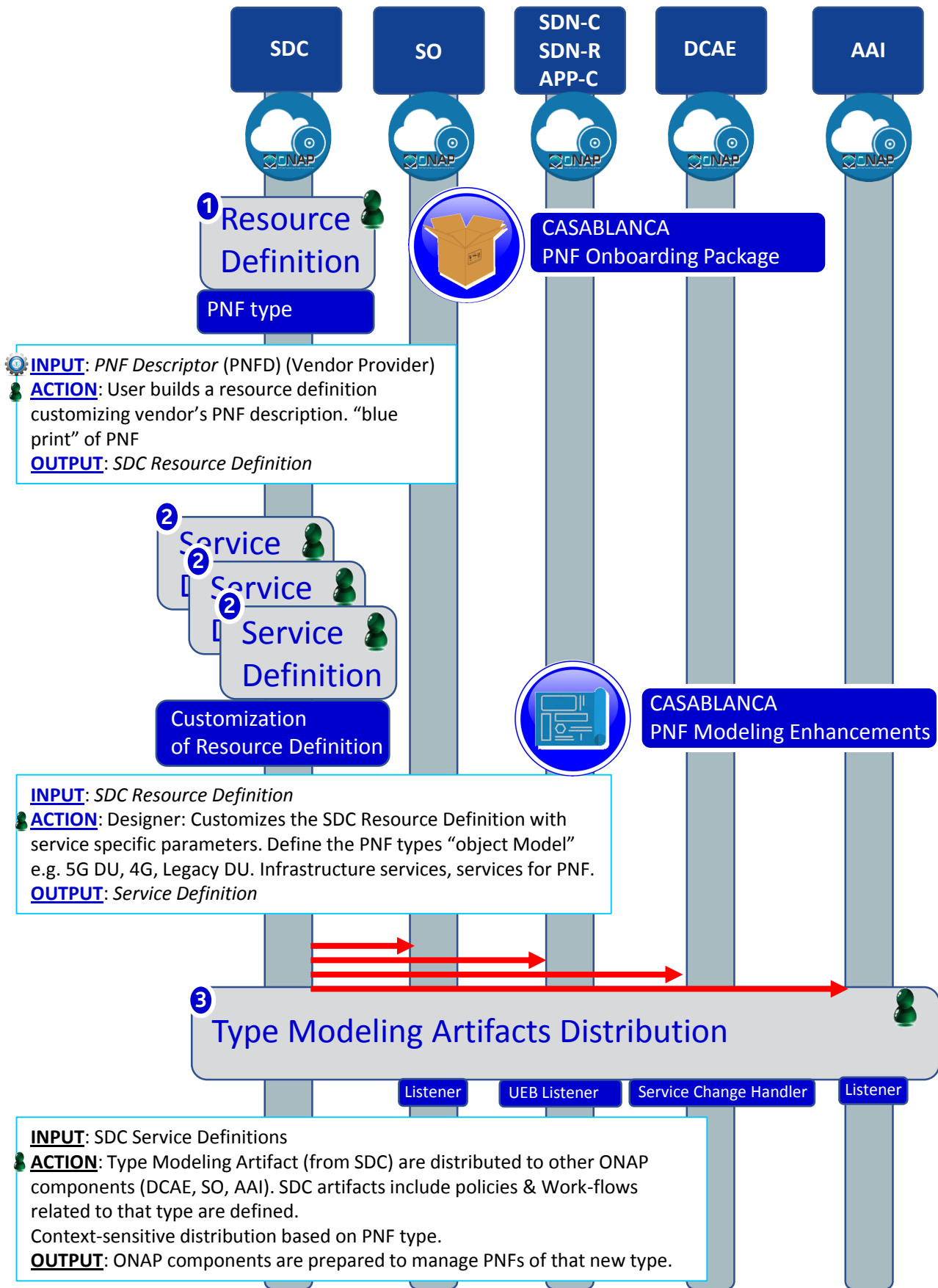
PROJECTS:
SDC, APP-C



NF Modeling in SDC IN CASABLANCA (R3)

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

Design Time (ONAP)



PNF PnP: MODELING ENHANCEMENTS

DESCRIPTION

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



PROJECTS

SDC, CDT

PNF ONBOARDING / PNF PACKAGE

DESCRIPTION

PNF Onboarding and PNF Package

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

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

TOSCA Meta data - main service template, TOSCA template.

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

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

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

(2) Controller – What is the PNF controller



PROJECTS:

SDC, APP-C

| STEP | DESCRIPTION |
|------|---|
| 1 | <p>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</p> <p>PNF RESOURCE Definition</p> <ul style="list-style-type: none"> 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. <p>Note: The user may provide whatever information in the above fields they know. Note: Consumer vs Enterprise deployments. Consumer systems pre-registered, distributed throughout a region. For a consumer deployment you might not know the MAC address/Serial number (PND IF) until the PNF connects to ONAP.</p> |
| 2 | <p>SERVICE Definition (uses a PNF)</p> <ul style="list-style-type: none"> 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 <p>The “basic” model are extended. Inherit (OO) from existing model. Vendor takes standard node types and creates their own extension. CDT (Configuration Design Tool) (GUI) to build artifacts to be used by APP-C (Tosca models) for a configure Template.</p> |
| 3 | <p>DISTRIBUTION – Event Monitoring Templates distributed. (?)</p> |

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 implement 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 SDC & Modeling Project Impacts

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

MODELING PRINCIPLES



Planner Personnel

Optimization
Network Planning



Provider Personnel

Operators
Technicians



Vendor Personnel

Technician
Developers
Product Support

SERVICE MODEL

Services

Application Data

Operational Operator

Functional Aspects

Run-Time

ONAP Service vs ETSI/OPENO/3GPP Service

Orchestrating ONAP component & resources

RESOURCE MODEL

Physical Resources

Application Data

Operational Operator

Physical Aspects

Run-Time

PLATFORM MODEL

NF Interconnectivity

ONAP Platform-level information

Design-time Operator

Templates

Meta-data

SDC Design Studio Catalog



ONAP Personnel

(Operator
Governor
Designer
Administrator
Tester)

ASSETS MANAGED (WIKI)

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

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

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

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

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

SDC COMPONENTS (WIKI)

There are four major components of SDC:

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

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

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

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

SDC COMPONENTS (WIKI)

1. Resource Model

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

2. Inventory Model

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

3. Configuration Model

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

4. Event “model”

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

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

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 | <p><u>PNF PACKAGE DEFINITION</u></p> <p>PNF packages similar to VNF packages. PNF Descriptors, artifacts. In PNF not doing deployment process in SDC. Only PNF configuration. Model a PNF. Onboard PNFs (create templates service configuration). Orchestrate a service on a PNF. Service provisioning. Life cycle: Template/service orchestrated. For PNF every PNF vendor makes this package. Need specific PNF properties. Image details. ONAP updates the image.</p> <p><u>VALIDATION OF A SDK PACKAGE</u></p> <p>VNF-SDK (validation, Package definition, verification tool) – package compliant. Allows creation/validation of packages. PNF-SDK <i>validates</i> the package.</p> |
| WORK FLOW (SDC) | Create work-flow for PNF (Srini) SDC Impacts related to work-flow. ACTION: Sample Work-flow for PNF. |
| 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. |

DCAE-DS IMPACTS

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

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

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

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

MODELING IMPACTS

Notes:

- 1) **EXTERNALS** - Not trying to model the internals of PNFs. What is exposed by the box is what is modeled.
- 2) **INTERRELATIONS** - Focus on relations of PNFs/VNFs. Interworking between PNFs/VNFs.
- 3) **VISIBILITY** - CP/UP visibility
Not M-Plane (as this is 3GPP standardized)
- 4) **MODELING ANALYSIS** - Modeling activity to assess PNF, and check SDC model is sufficient to cover Casa use cases if additional parameters need to be added (e.g. relations between other NFs). Expanding the “Release 0 model” for Casa. PNF type vs PNF instance. Design-time vs Run-time model.

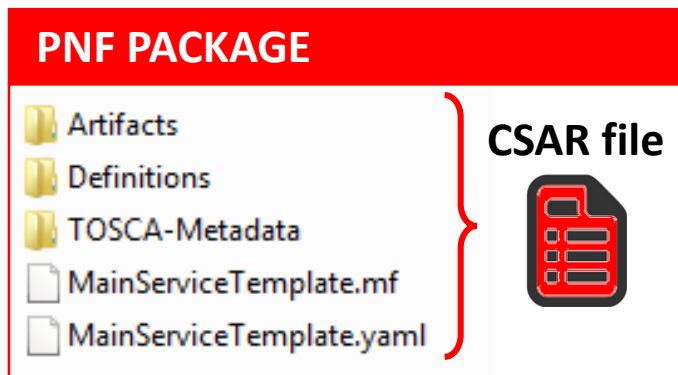
 Suggested VES Event Entry - Fault' Domain Datatypes
For Alarm Dictionary Index (in **Dublin**)

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

PNF PACKAGE

PNF has no onboarding package.

Just model the PNF from the modeling screens.



CSAR – decompile info stored in SDC model.

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

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

ARTIFACTS

DEFINITIONS

PNF – 5G Base Stations

Backhaul Ports – PNF & VNF and want to communicate. In a VNF can describe a port a TOSCA. Model onboarded understand what can connect to what. CP connections. Can see they can connect. Model needs to capture info for modeling parts representing connections. Model allow someone designing service to connections. Or requirements from VNF/PNF from the model. A virtual link. One VNF & PNF connected via virtual network/link.

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

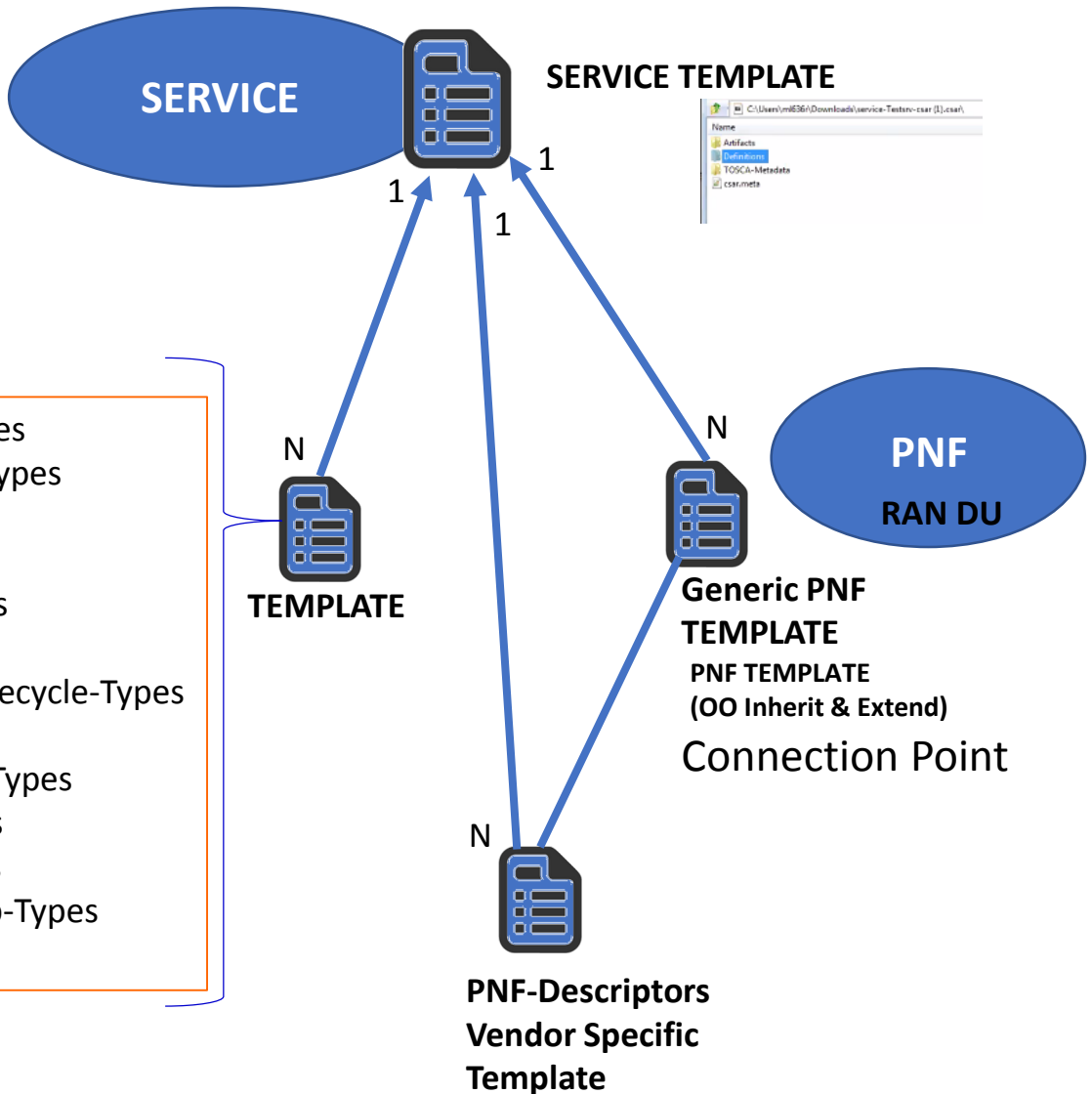
PNF Policies -

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

Software Version

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

SDC MODELING (Design Time) – Casa R3



| Name | Is |
|--------------------------|----|
| artifact-type | is |
| capability-type | is |
| category | is |
| data-type | is |
| group-type | is |
| heat-type | is |
| interface-lifecycle-type | is |
| nfv-type | is |
| normative-type | is |
| onap-type | is |
| policy-type | is |
| relationship-type | is |
| user | is |

```

sdccatalog-be> src> main> resources> import> tosca> heat-types> Generic_PNF
Open New folder
Name Date modified Type Size
Generic_PNF 9/10/2017 12:49 JSON File
Generic_PNF 9/10/2017 12:49 YAML File
Generic_PNF 9/10/2017 12:49 Compressed (zip...)
tosca_definitions_version: tosca_simple_yaml_1_0_0
nodes:
- org.openecomp.resource.abstract.nodes.PNF:
  derived from: tosca.nodes.Root
  properties:
  nf_function:
    type: string
  nf_role:
    type: string
  nf_type:
    type: string
  type: string
  
```

Controller Type
["common fields"] ...
Vendor-field1

PNFD Definition in ETSI-NFV-IFA014v242

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

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

| Contents | Description |
|---------------------------|--|
| 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. (pnfdInvariantId 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. |

VERSION MODELING

Diagram of Software Version Management for a PNF



PNF-Descriptor (version)

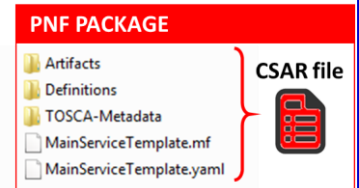
Vendor Provided

PNFD version = 6



PNF-Package (version)

Minimal PNFPackage version = 7.1
(SDC Versions the Package)



PNF Software (version)

[Detected Software, Expected Software]

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

ONAP EXPECTED PNF S/W 3.0 (modeling)



PNF Hardware (version)

Hardware Version

Firmware Version

Product Model version

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

PNF MODELING Information (IN SDC)

| 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 |
| 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 |
| SWversionList (opt) | The EXPECTED software to be supported by the PNF. |
| PackageVersion (Dublin+) | 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) |

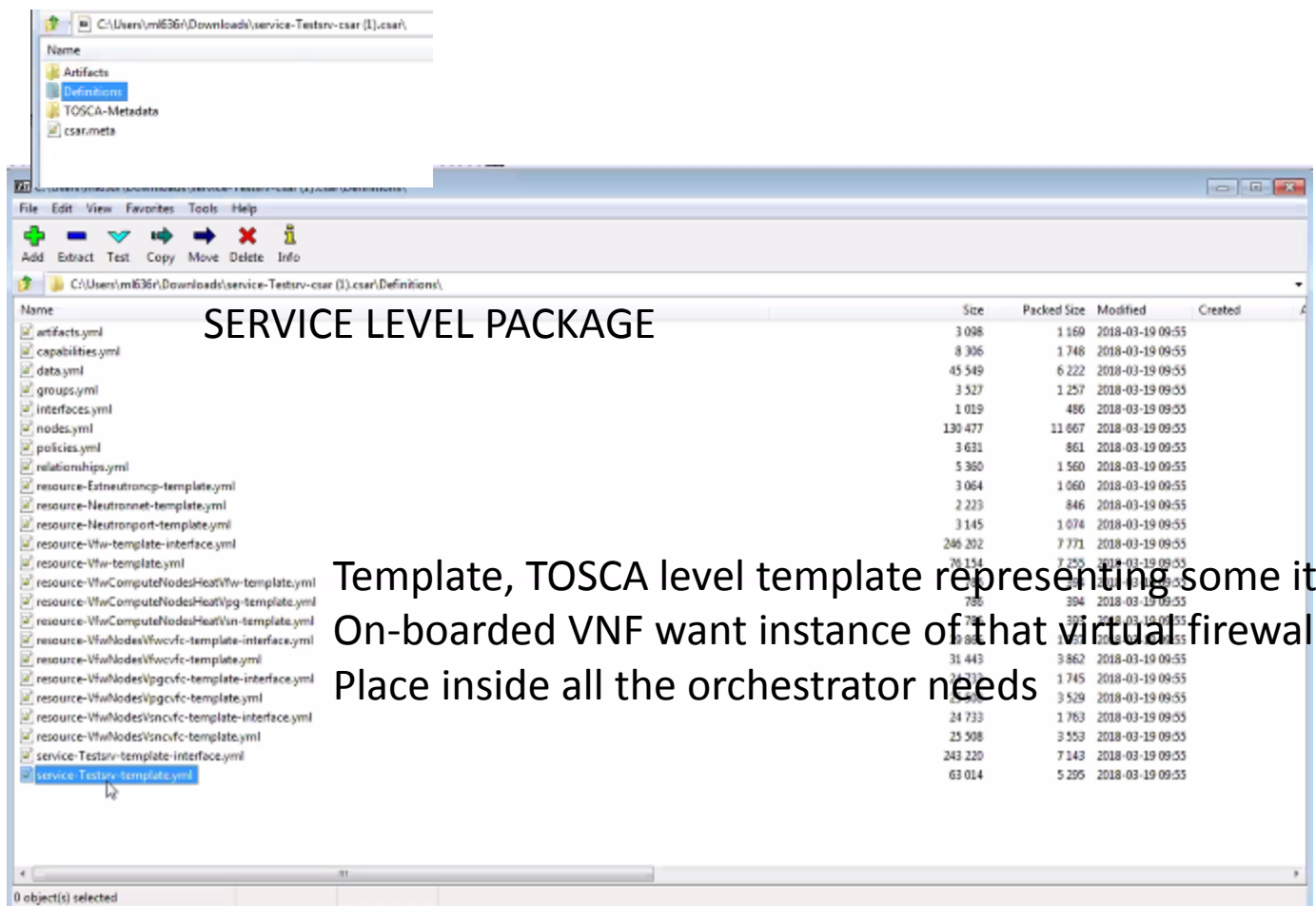
```

node_template {
  sdc: {
    type: org.onap.pnf.resource.cp.extCP
  }
  metadata {
    lowerCaseMID: 9d8223e-33c0-476d-acc0-97af62b044
    uuid: c8b79ac-5da4-468b-8ac3-28d2380a48
    customIdentifier: 7ea3d0c-b0cf-637e-bd60-11b0e009a1
    version: 1.0
    name: ExtCP
    description: The sTQC Connection Point base type all other CP derive from
    type: CP
    category: General
    subcategory: Network Element
    resourceNumber: ST (22ca)
    resourceVendorName: 1.0.0,extCP
    resourceIdentifierNumber:
  }
  properties {
    min_required:
    min_optional:
    is_required: false
    ext_managed:
    mosp_generated_name: true
  }
}
    
```

• Content of PNF software version List

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

***Already supported in Beijing**



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

```

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

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

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

The screenshot shows a TOSCA editor interface. On the left is a component palette with various types like 'ExtVL', 'Generic Neutron...', 'NeutronNet', 'NeutronPort', 'Port', 'subInterface', 'VDU Cap', and 'Infrastructure'. The main workspace displays a network diagram with two ports, 'Port 0' and 'Port 1', connected to a central node. On the right, a metadata panel shows details for a connection point, including its invariant and customisation UUIDs, name, description, type, category, subcategory, and vendor information.

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

Creation Date: 07/19/2018
Author: Carlos Santana
Vendor Name: 12344
Vendor Release: 1243235
Vendor Model Number: 1232454
Contact ID: cs0008
Description: 123123
ADDITIONAL INFORMATION
TAGS
examplepnf



Associating a Controller for a NF

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

NF Controller (Casablanca)

PROBLEM STATEMENT

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

OBJECTIVES

As automated as possible

Using discovery if possible

Flexible operator could design PersonaA for PNF1, PersonaB for PNF2

RESULT

SO knows which API to use for PNF controller

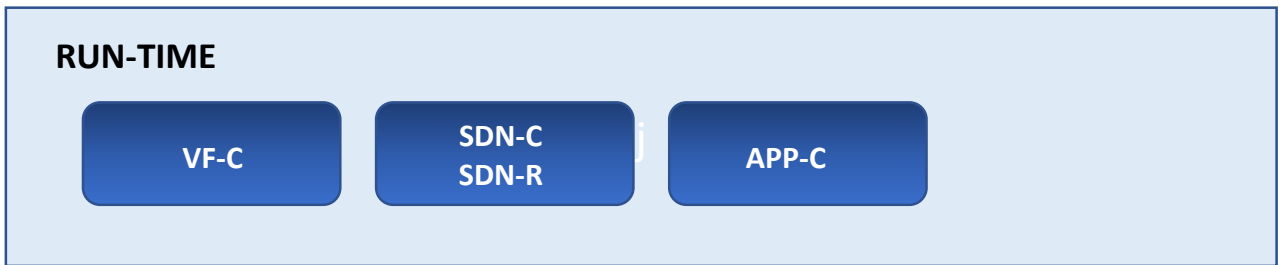
OTN PNF (CCVPN), Routers, 5G DU are PNFs are early adopters using this

SOLUTION (Casablanca)

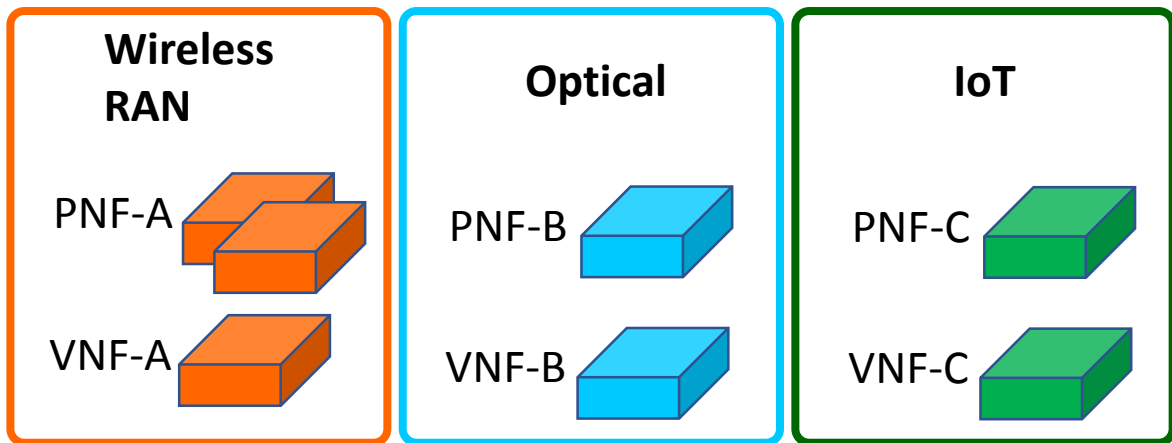
Hard-Code controller to PNF.

NF Controller Concepts

ONAP Deployment



Technology
Domain



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.

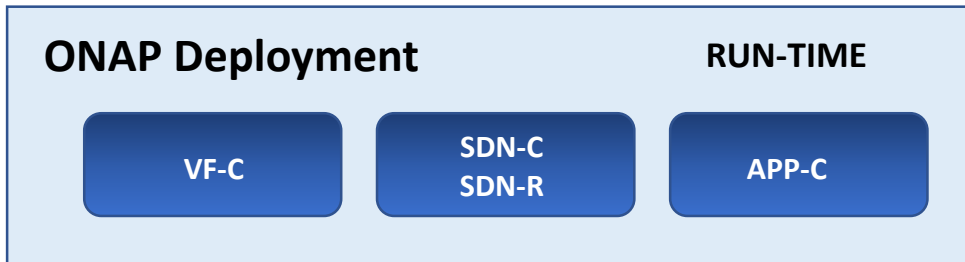
NF Controller Concepts

ONAP PLATFORM TYPE CONTROLLERS

(SDN-C, SDN-R, VF-C, APP-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.



DOMAIN CONTROLLERS

Regional Deployment (instances) of Controllers

DESCRIPTION

Domain 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

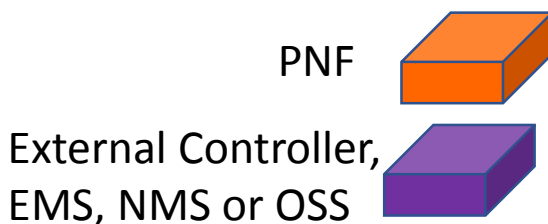


EXTERNAL CONTROLLERS

EMS, NMS, OSS

DESCRIPTION

External Controllers that reside outside of ONAP that perform management functions with the PNF and VNFs.



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.

PROPOSAL #2

Policy Driven

DESCRIPTION

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

PROPOSAL #3

Table Driven Association

DESCRIPTION

Table-Driven Look-up solution based on NF function type. For example a controller may support a particular technology domain (wireless/wireline/optical). Controller support domain and auto-populates the tables. Could be a GUI in SDC (a run-time catalog table).

The Table could be onboarded. Design-time field. The PNF needs to have a “Technology” domain (a user or designer). Specific images (S/W loads) to specific Controllers.

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

1 “ONAP platform type” SDN-C (SDN-R) VF-C APP-C

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

ONAP deployment [controller] – Domain Controller – ONAP Controller

3. Vendor / External Controller

NF Controller

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.
-

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) IMPACTS

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. QUESTION: ALTITUDE (is that in the Complex Object?)

Software Version

DETECTED SOFTWARE PNF VERSION(S) – 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 have *Active/Passive*. A&AI already has PNF name S/W version. Graceful roll-out, Fallback. VNF v1.0 reason (amount of time to maintain S/W)

Content of PNF software version List

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

Christina: This will require schema changes in A&AI to support Passive S/W

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. CLLI code used to set PS1 to identify OPs machine.

Manager IP Address



Manager IP Address – provides an additional IP address for the NF that is vendor-specific and relevant to the OAM management of the NF.
SUGGESTION (from Christina A&AI PTL) model the NMS as a NF itself (and the NMS will have parameters to represent itself; and then NF can be associated with it). “Manager” != APP-C, SDN-C ... “Manager” NMS (EMS).

S/W Image Repository



S/W Image Repository – Where the S/W is located

ACTIVE INVENTORY (A&AI) IMPACTS

PNF A&AI ENTRY (From Beijing)

- PNF has a “*pnf-name*” = Key in AAI.

Pnf-name is first 3 letters of vendorName concatenated with serialNumber for a unique PNF instance ID = PNFid = PNF Correlation ID = pnf-name.

Example: PNFid = NOK123451ZW3. The PNFid is in A&AI pnf-name field.

PNFid = [VENDOR][SERIALNUMBER]

equip-type (PNF Type). *equip-vendor* (optional); *equip-model* (optional);
pnf-id (PNF ID) = UUID

(Step 33) adds *ipaddress-v4-oam*; *ipaddress-v6-oam* This is the “manager IP Address” which for a DU might be a CU IP address. IP address on the PNF address itself.

(FYI/ *ipaddress-v4-loopback-0*).

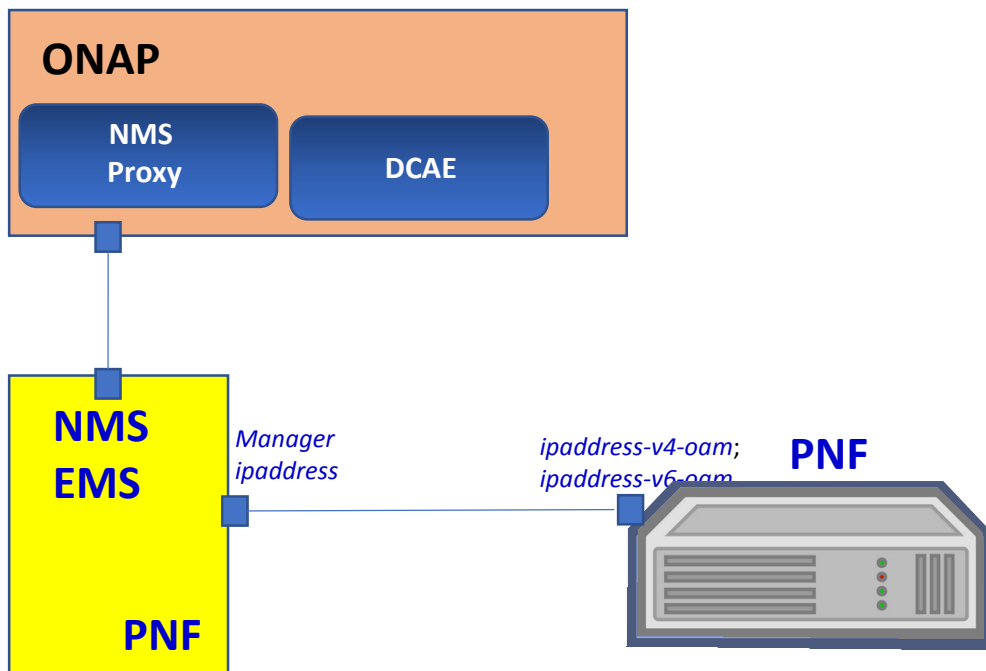
mac-address & *serial-number*,

PNF:: proxy IP address

Active Software Version

Image Repository (Directory URL, S/W image, URI in OpenStack)

Manager (NMS/EMS) to PNF



OAM
FCAPS
Logging
SWM
Identity

1. ONAP -> NMS command
2. Relay (NMS trusted source)
3. Network Analytics – (DCAE AA)

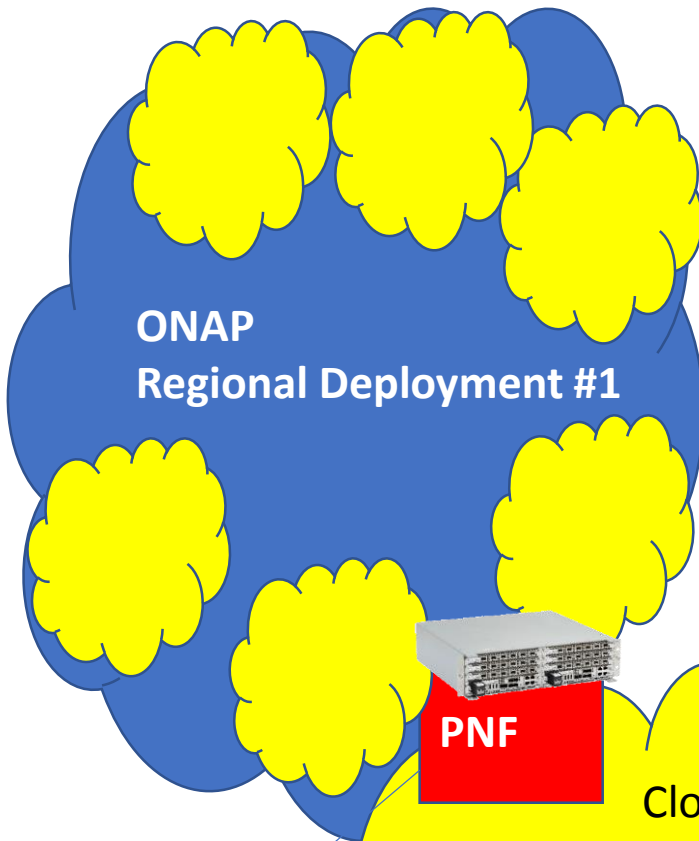
COMPLEX OBJECT

Physical Location

Manhattan
Freedom Tower
123 Main St.



Staten Island



ONAP
Regional Deployment #1

PNF

LATA

PNF
Controller

Cloud Region
Object

Cloud Region #1

Complex
Object

Complex
Object

Complex
Object





PNF Plug and Play ROADMAP (After Casablanca)

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

PNF PACKAGE

ALARM DEFINITIONS

Alarm Dictionary
(Vendor provided)



YAML Definitions
(Vendor provided)



PNF Alarm #22
#22 S/W problem xyz
VES = fault fields; "fault" domain publish DMaaP
Clamp, Analytic, Holmes – [#22 event]
Dictionary looks up #22 ->

MEASUREMENT DEFS

Measurement
Dictionary
(Vendor provided)



Measurement
Schema
(Vendor provided)



YAML Definitions
(Vendor provided)



NF PACKAGE

Artifacts
Definitions
TOSCA-Metadata
MainServiceTemplate.mf
MainServiceTemplate.yaml

CSAR file



CONFIGURATION DEFS

Configuration
Schema
(Vendor provided)



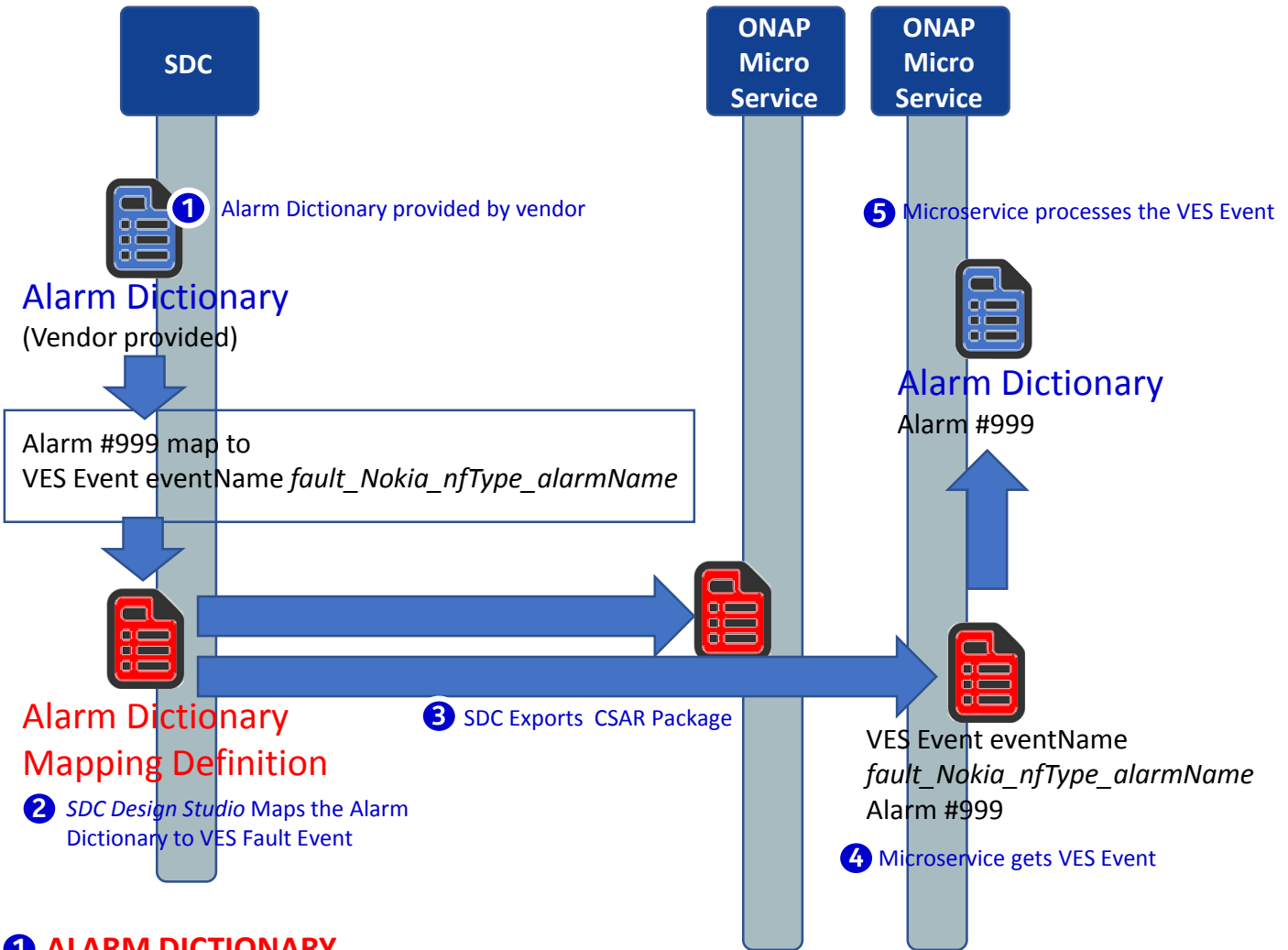
YAML Definitions
(Vendor provided)



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]

Alarm Dictionary Usage



1 ALARM DICTIONARY

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

2 SDC DESIGN STUDIO MAPS VES FAULT Event to Alarm Dictionary

SDC Design studio does mapping of Alarm dictionary entries to VES Fault Events to produce a "mapping" definition in the CSAR package. (RATHER than mS doing the mapping because it is more model driven)

3 SDC DISTRIBUTES DEFINITIONS

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

DEFINITIONS & ARTIFACTS (CSAR PACKAGE)

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

4 MICRO SERVICE GETS FAULT EVENT

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

5 MICRO SERVICE PROCESSES EVENT

Microservice processes the VES Event using the Alarm Dictionary

Alarm Dictionary Usage

ALARM DICTIONARY PURPOSE


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

FAULT DICTIONARY PURPOSE

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

NF ALARM DICTIONARY FIELDS (Template)



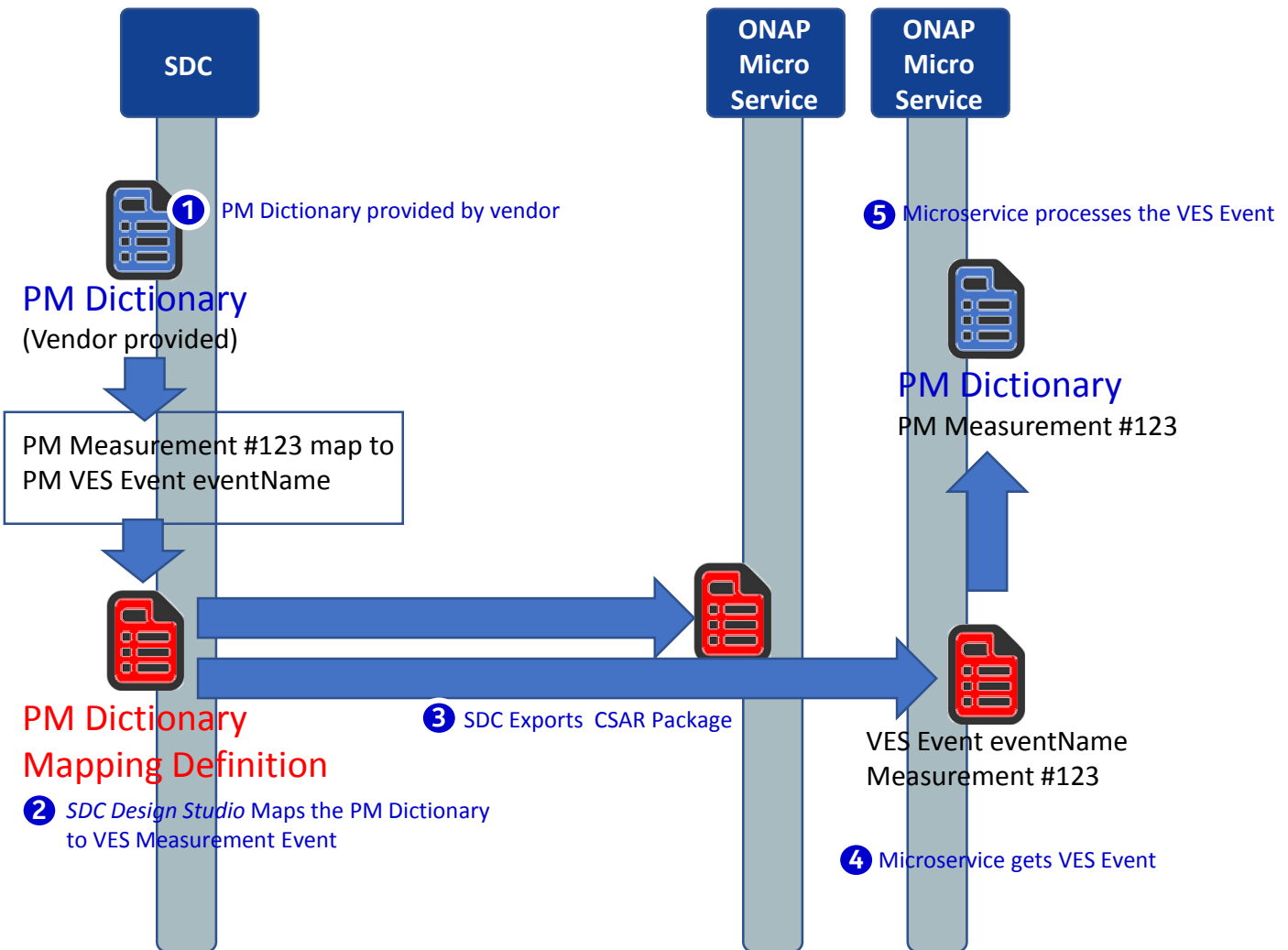
| ALARM FIELD | DESCRIPTION |
|--|---|
| Alarm Dictionary Index  | <p>Gives the Identifier for the alarm. This is also the Identifier that is used in the VES event so it can be used to associate the event with the definition entry.</p> <p>The VES Event <i>EventID</i> would encode the Alarm number which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

NF FAULT DICTIONARY FIELDS (Template)



| DEFINITION FIELD | DESCRIPTION |
|--------------------------------|---|
| 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 <i>this maps to the alarmCondition in the VES Fault Event in faultevent fields.</i> 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 <i>this maps to the eventSourceType in the VES Fault Event in faultevent fields.</i> 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 <i>vendor specific</i> additional fault information. |

PM Dictionary Usage



1 PM DICTIONARY

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

2 SDC DESIGN STUDIO MAPS VES FAULT Event to PM Dictionary

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

3 SDC DISTRIBUTES DEFINITIONS

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

DEFINITIONS & ARTIFACTS (CSAR PACKAGE)

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

4 MICRO SERVICE GETS MEASUREMENT EVENT

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

5 MICRO SERVICE PROCESSES EVENT

Microservice processes the VES Event using the PM Measurements Dictionary



APPENDIX & Meeting Notes



The screenshot displays the ONAP Service Orchestrator interface. At the top, the header includes the Open Daylight logo, the ONAP logo, and the text 'Service Orchestrator'. On the right side of the header, there is a 'Logout (admin)' link and a menu icon. Below the header, a navigation sidebar on the left lists various components: Nodes, Connect, ONAP AAI, ONAP DCAE, ONAP SO, pnf Fault, pnf Config, pnf PM Current, pnf PM History, pnf PM Link, pnf Inventory, pnf Topology, pnf Mediator, and Help. The main content area is titled 'Design Phase' and 'Create PNF via SO in A&AI'. It features a form with several input fields: 'Name' (containing 'New-PNF'), 'Identifier' (containing 'New-PNF network unique identifier'), 'Equipment type' (containing 'Fancy equipment type'), 'Equipment model' (containing 'Best in class'), 'Equipment vendor' (containing 'ONAP SDN-R Community'), and 'IPv4 address' (containing '10.10.10.10'). There is also a 'Maintenance mode' checkbox. A blue button labeled '+ Create PNF in Active and Available Inventory' is positioned below the form. At the bottom of the main area, a footer line reads 'ONAP SDN-R | ONF Wireless for OpenDaylight Carbon-SR1 - Build: 2018-06-18 13:43 UTC'. In the top right corner of the main area, there is a status bar showing 'Nodes: 1 | Alarm status: 4 | 0 | 0 | 2 | Sum: 6 | Help'.

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