LF NETWORKING

LFN Developer & Testing Forum

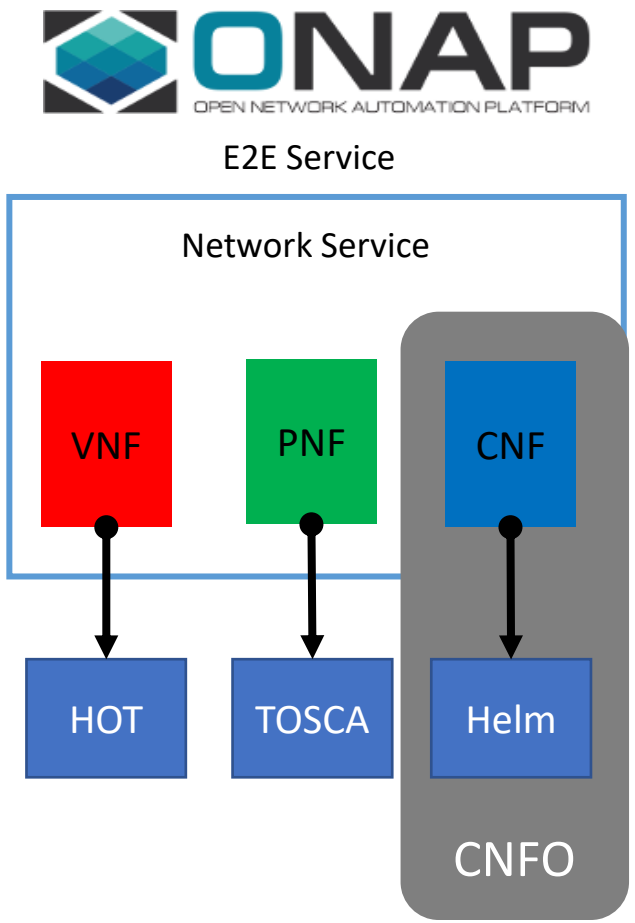
CNF Tutorial

CNF Orchestration in ONAP Step by Step

Marcin Krasowski (Samsung)
Seshu Kumar (Huawei)
Lukasz Rajewski (Orange)

11.01.2022

CNF Orchestration (1)



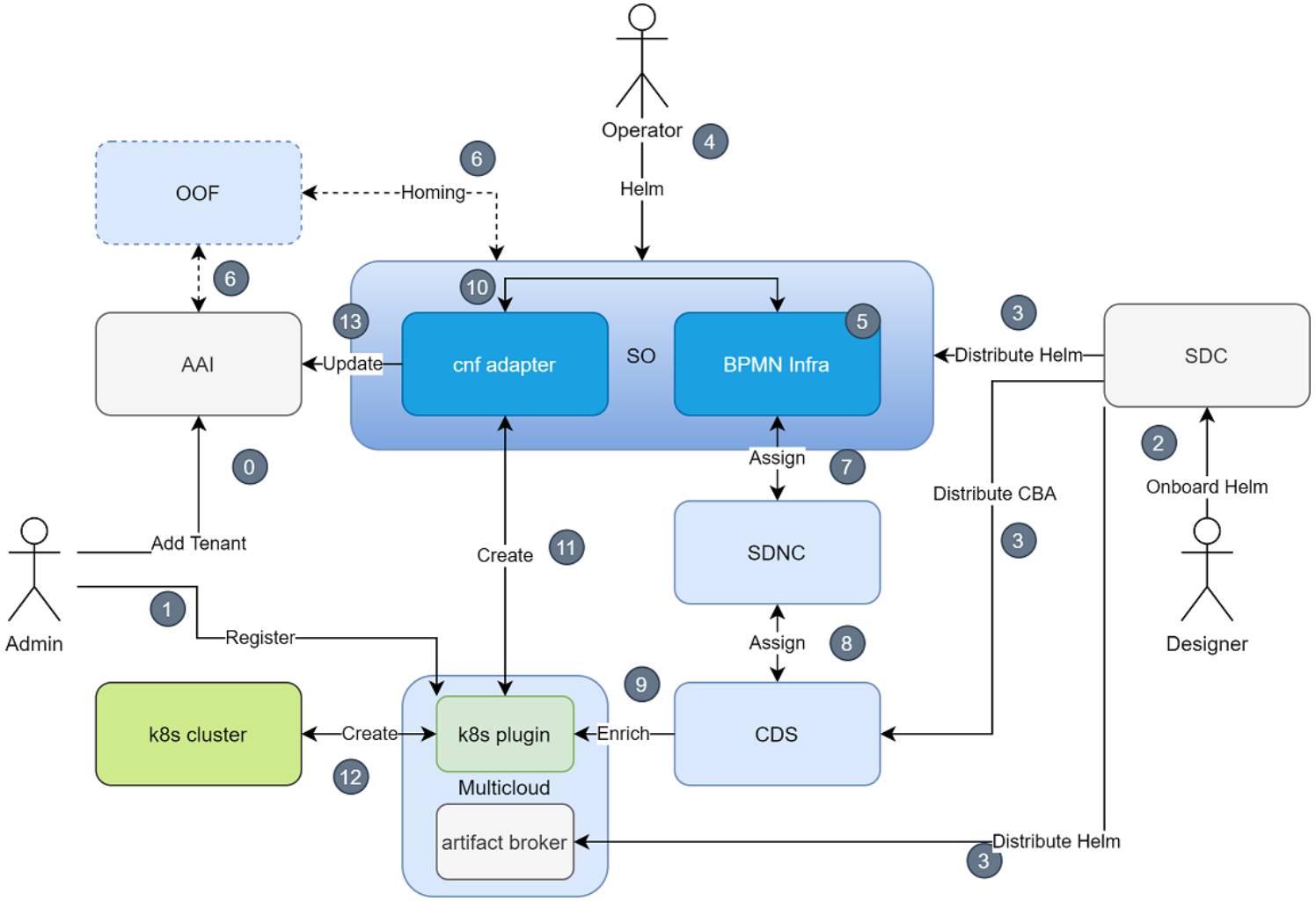
REQ-341
Guilin

REQ-458
Honolulu

REQ-627
Istanbul

REQ-890
Jakarta

CNF Orchestration (2)



- **Helm Onboarding** **[SDC]**
- **Day 0/1 Customization** **[CDS/MC]**
- **Instantiation** **[SO/MC]**
- **Day 2 Configuration** **[CDS/MC]**
- **State Synchronization** **[SO/AAI/MC]**
- **Healthcheck** **[SO/CDS/MC]**
- **Multi-cluster deployment** **[SO/MC]**
- **VNF/PNF Integration** **[ALL]**
- **E2E Service Automation** **[SDK]**

CNFO Istanbul Changes & Demo

CNF orchestration enchancements

- AAI synchronization
  - Added simplified CNF Model in AAI to allow user to know about available resources and where to get their exact status from
  - AAI is updated when service is instantiated and terminated
  - No support (yet!) for automatic AAI updates when CNF resources change (e.g. scale in/out, etc)
- Support for health check
  - Health check exposed through Northbound SO API
  - Health check can be executed on demand
- Enhancements in multicloud (k8s) plugin API

Jakarta
REQ-890

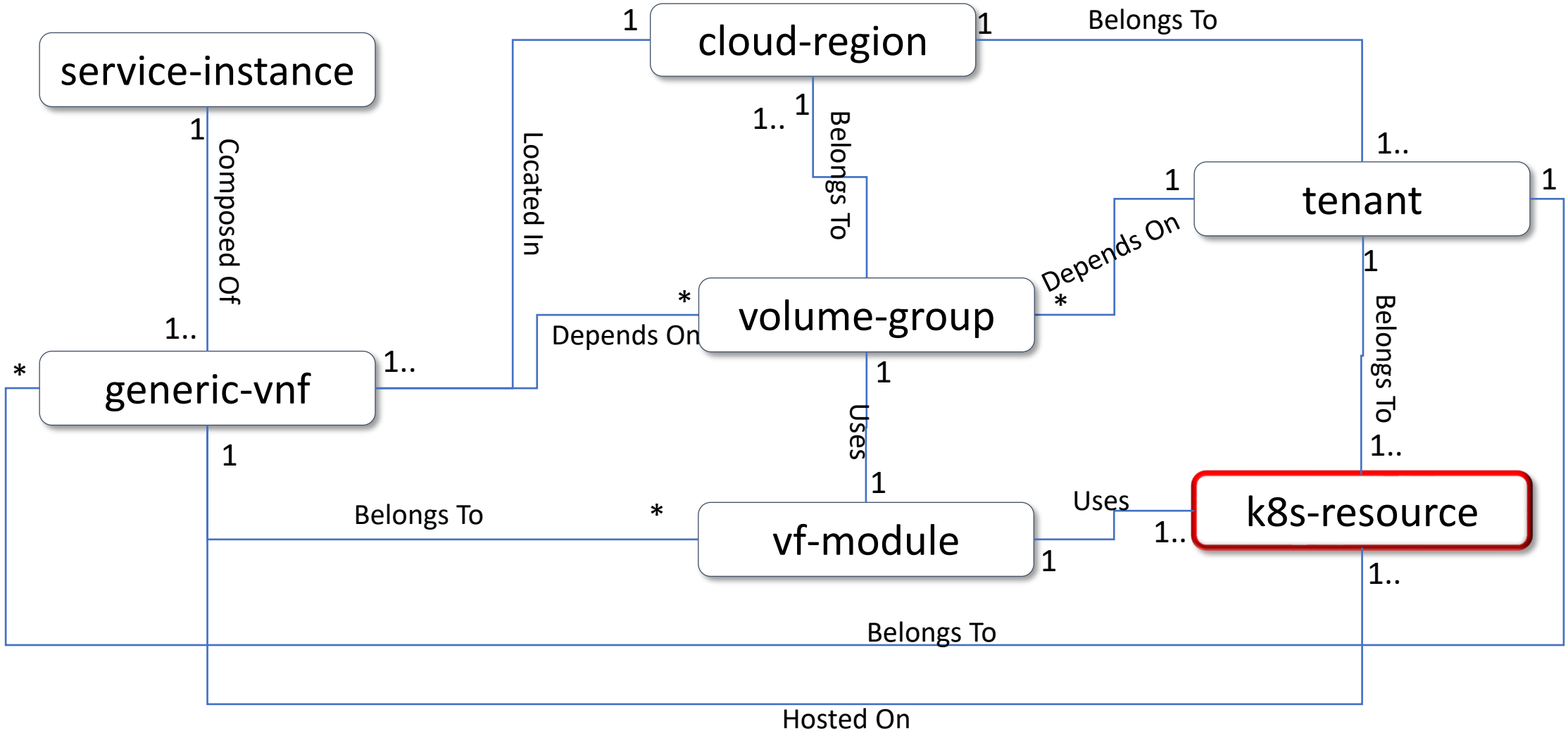
AAI model: k8s resource object

Attribute	Type	Mandatory
id	UUID	Yes (PK)
name	String	Yes
group	String	Yes
version	String	Yes
kind	String	Yes
labels	List of strings	No
namespace	String	Yes
selflink	URI	Yes

K8s resource is basic AAI entity to model resources created in K8s cluster.

It plays similar role as vserver resource for standard VNFs.

AAI model: relations



https://wiki.onap.org/display/DW/Simplified+K8S+Resource+Model+-+IM

K8s Resource ID Calculation

- K8s resource id is calculated from values available to client; it is SHA 256 of a concatenation of the following values:
  1. K8s RB Instance ID (Heat Stack ID param if AAI vf-module or label in resource in K8s)
  2. K8s resource name
  3. K8s namespace
  4. K8s resource kind
  5. K8s resource group
  6. K8s resource version
  7. CloudOwnerName (From AAI, for tenant in which CNF is deployed)
  8. CloudRegionName (From AAI, for tenant in which CNF is deployed)
  9. TenantId (From AAI, for tenant in which CNF is deployed)
- The above may be used to calculate K8s resource ID i.e. base on the data gathered from Prometheus to access AAI k8S Resource relationships (vnf-id, service-instance-id)

K8s Resource ID Calculation cont.

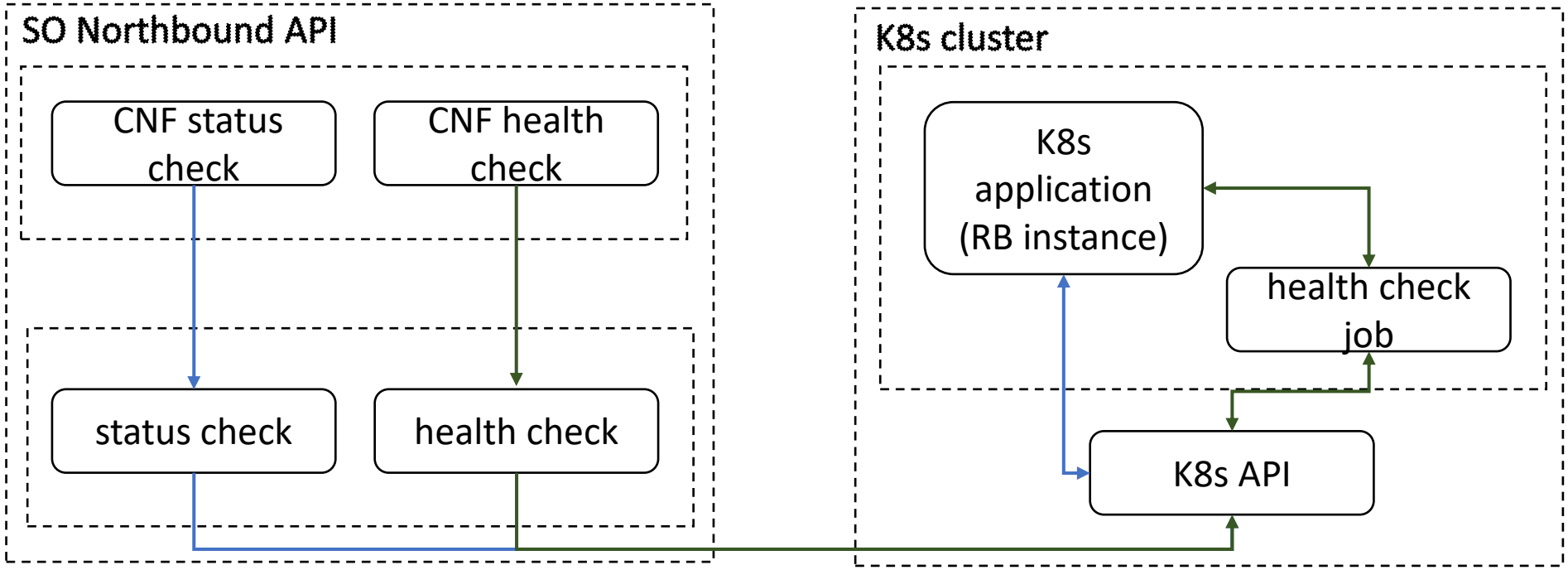
- K8s Resource ID is calculated in:
  - `org.onap.so.adapters.cnf.service.aai.AaiIdGeneratorService.java`

```
String generateId(K8sRbInstanceResourceStatus resourceStatus, AaiUpdateRequest aaiUpdateRequest) {
        K8sRbInstanceGvk gvk = resourceStatus.getGvk();
        K8sStatusMetadata metadata = resourceStatus.getStatus().getK8sStatusMetadata();
        String originalString = aaiUpdateRequest.getInstanceId() + resourceStatus.getName() +
                                (metadata.getNamespace() != null ? metadata.getNamespace() : "") +
                                gvk.getKind() + gvk.getGroup() +
                                gvk.getVersion() + aaiUpdateRequest.getCloudOwner() +
                                aaiUpdateRequest.getCloudRegion() + aaiUpdateRequest.getTenantId();
        return Hashing.sha256().hashString(originalString, StandardCharsets.UTF_8).toString();
}
```

Changes in ID generation algorithm are very unlikely, however please be aware that the above code snippet is for illustrative purposes only; always check source code or most recent documentation !

Support for on-demand healthcheck

- SO integration with API healthcheck
  - Healtcheck can be executed on demand through SO REST API



Enhancements in multicloud (k8s) plugin

- Major Changes
  - Helm 3 support
  - Pre-/Post- Instantiation and Deletion Hooks
  - Status/Query API Enhancements
    - Status can cover resources created outside ONAP (k8s operator case)
    - Refined Query API
  - Change of Helm distribution to enable Helm update
- Created/Updated APIs target:
  - Enhancements for Status/Query API functionalities
    - Implementation of Query API on root level
    - Improvements in status recognition for instantiated k8s resources

CNFO Capabilities - Tutorial

Exemplary Use Cases

LFN Developer & Testing Forum

Apache

- ❑ Day 0/1/2 C(N)F
- ❑ Full automation in ONAP
- ❑ Standard K8S Cluster
- ❑ ONAP Istanbul+
- ❑ **The simplest**

Smoke Use Case

vFW CNF

- ❑ Day 0/1/2 CNF
- ❑ Full automation in ONAP + Postman
- ❑ Required KUD K8S Cluster
- ❑ ONAP Guilin+
- ❑ Used to validate CNFO
- ❑ **The best documentation**

DTF June 2021 Video

Free5GC

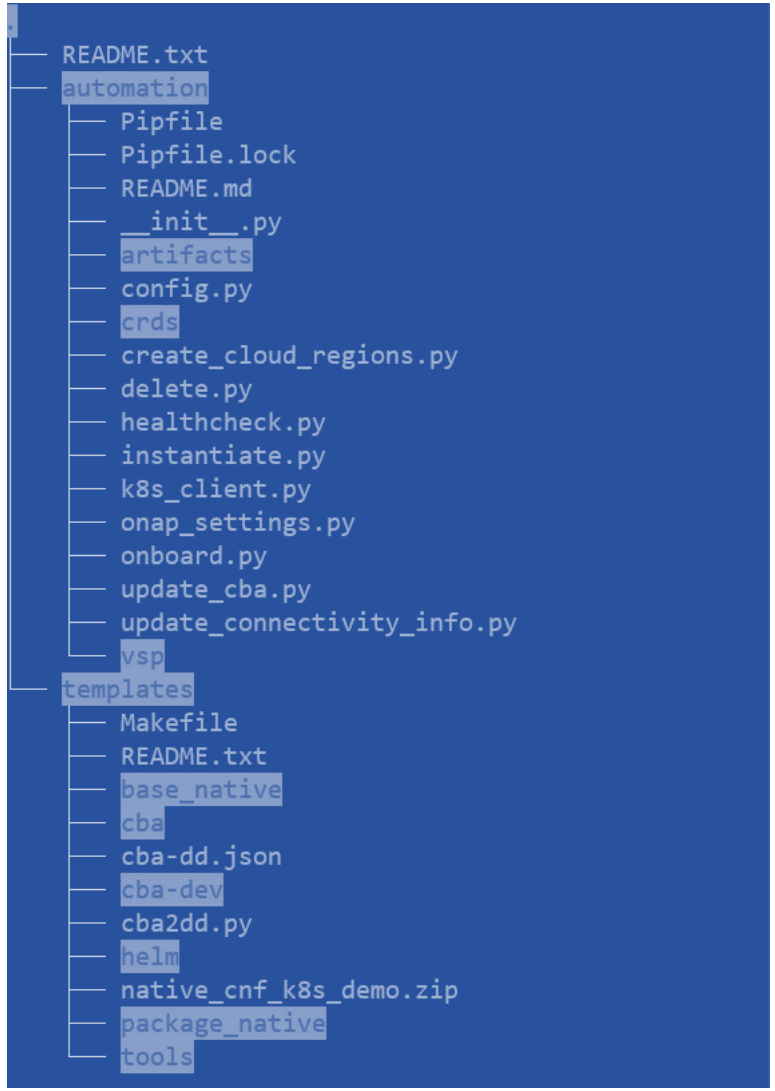
- ❑ Day 0/1/2 CNF + PNF
- ❑ Full automation in ONAP
- ❑ Required Dedicated K8S Cluster
- ❑ ONAP Istanbul+
- ❑ **The most complete example**

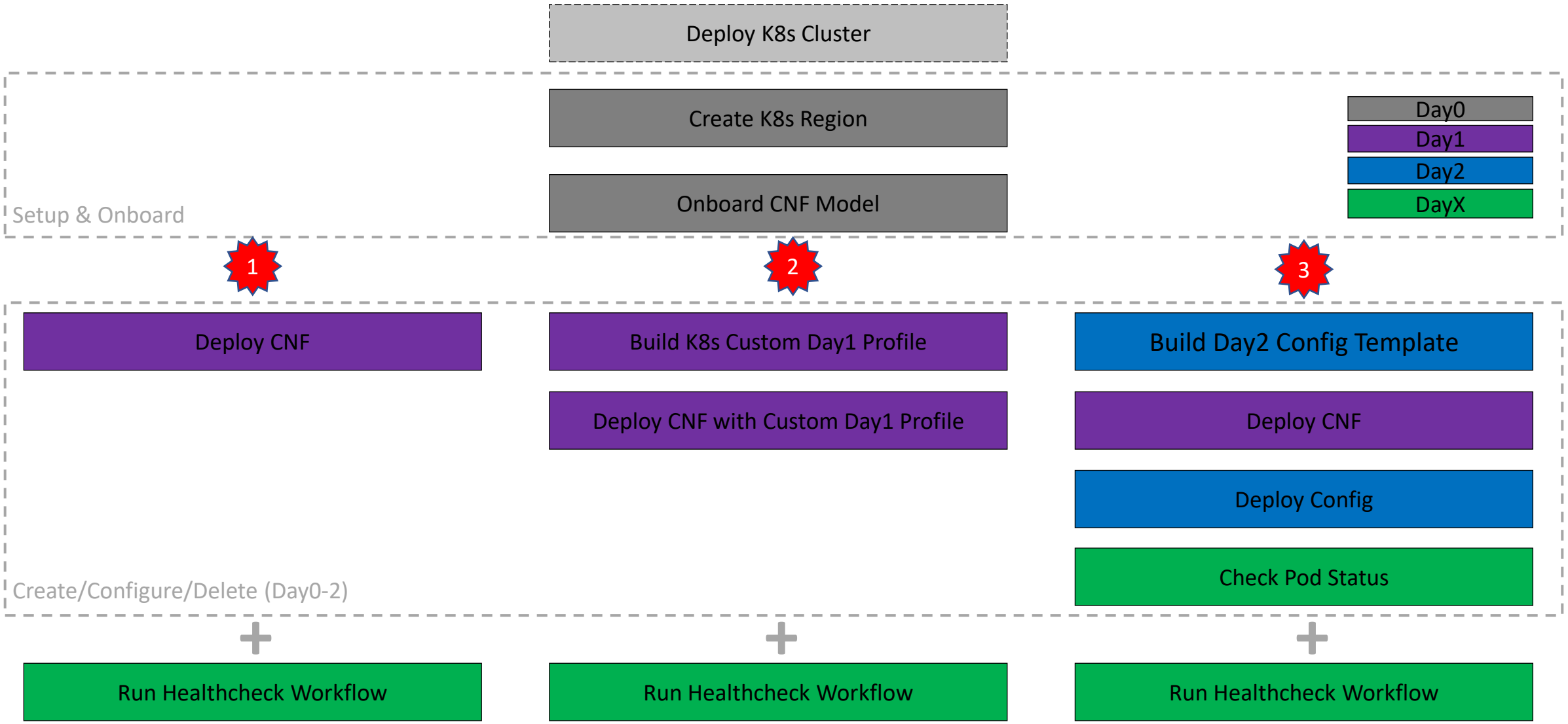
DTF Session on Jan. 13th

Automation for CNF Use Cases

LFN Developer & Testing Forum

- Python ONAP-SDK Based
- Templates Folder
  - Build and Test CBA
  - Build VSP with make
- Automation Folder
  - Step-by-step README
  - Create K8S Region
  - Onboard Service
  - Instantiate Service
  - Delete Service
  - Check Health of CNF




Deploy K8s Cluster
Create K8s Region
Onboard CNF Model
Day0
Day1
Day2
DayX
Setup & Onboard
1
2
3
Deploy CNF
Build K8s Custom Day1 Profile
Deploy CNF with Custom Day1 Profile
Build Day2 Config Template
Deploy CNF
Deploy Config
Check Pod Status
Create/Configure/Delete (Day0-2)
Run Healthcheck Workflow
Run Healthcheck Workflow
Run Healthcheck Workflow

K8S & Helm Requirements

- Kubernetes
  - Cluster must support v1.19 API
  - Image repository managed outside ONAP
  - Authentication through the static kubeconfig file
    - No exec command support in kubeconfig file
- Helm
  - Helm 3.5 used
  - No chart repository suport – chart dependencies must be embedded as subcharts
  - One Helm Chart instance = only one k8s namespace
  - Helm create/delete hooks supported
  - Helm Upgrade (including upgrade hooks) not yet supported

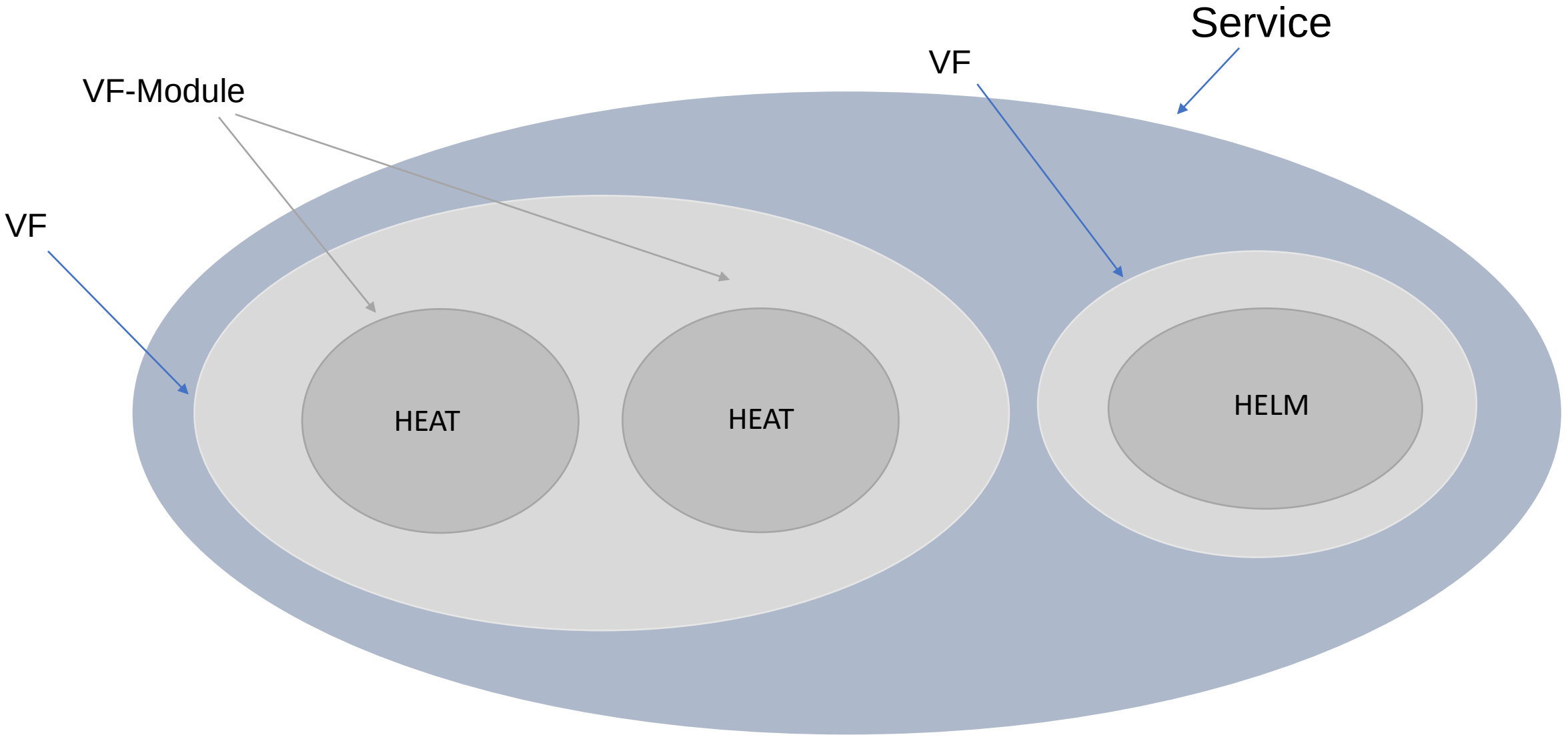
kubernetes

HELM

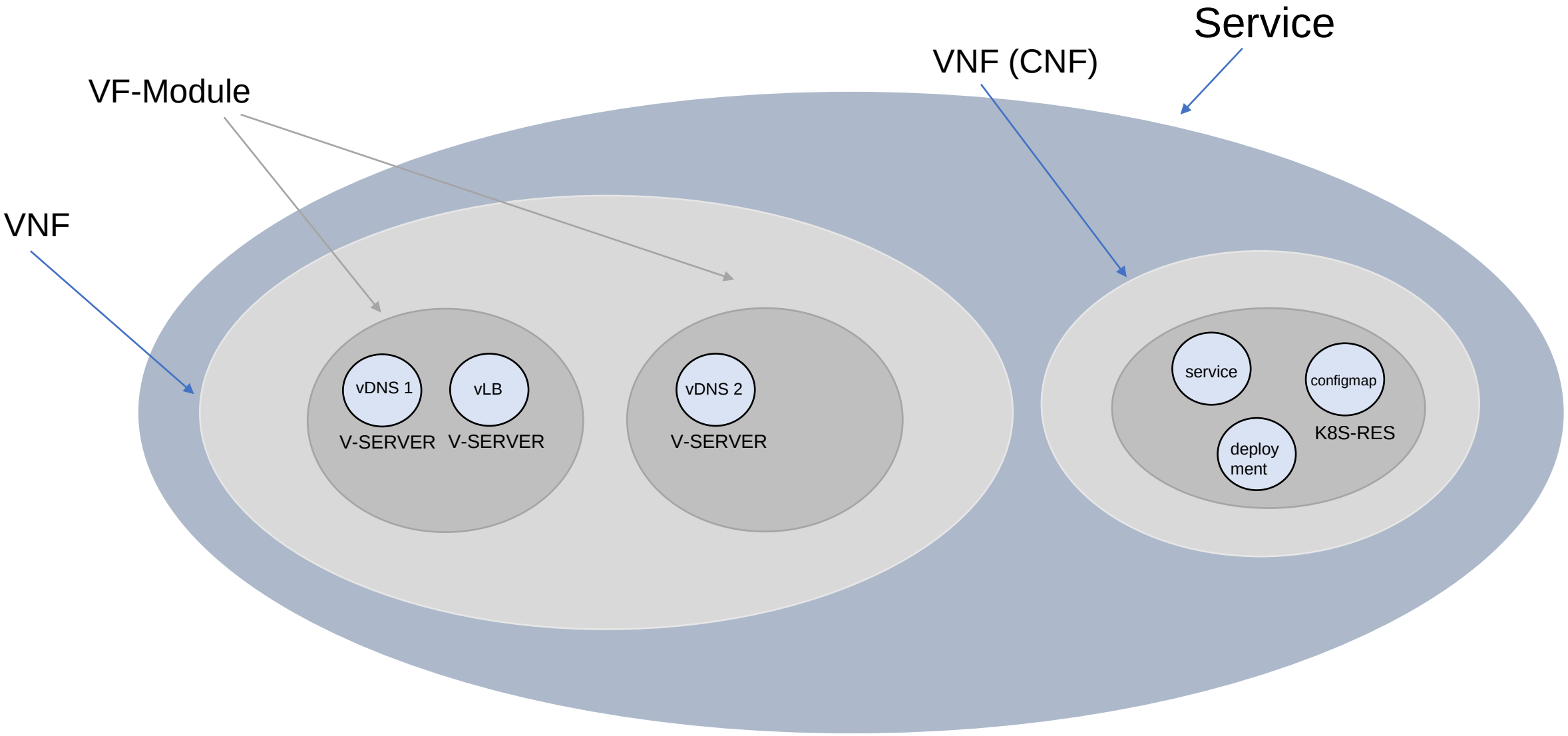
Jakarta

REQ-890

ONAP modeling concept (SDC)

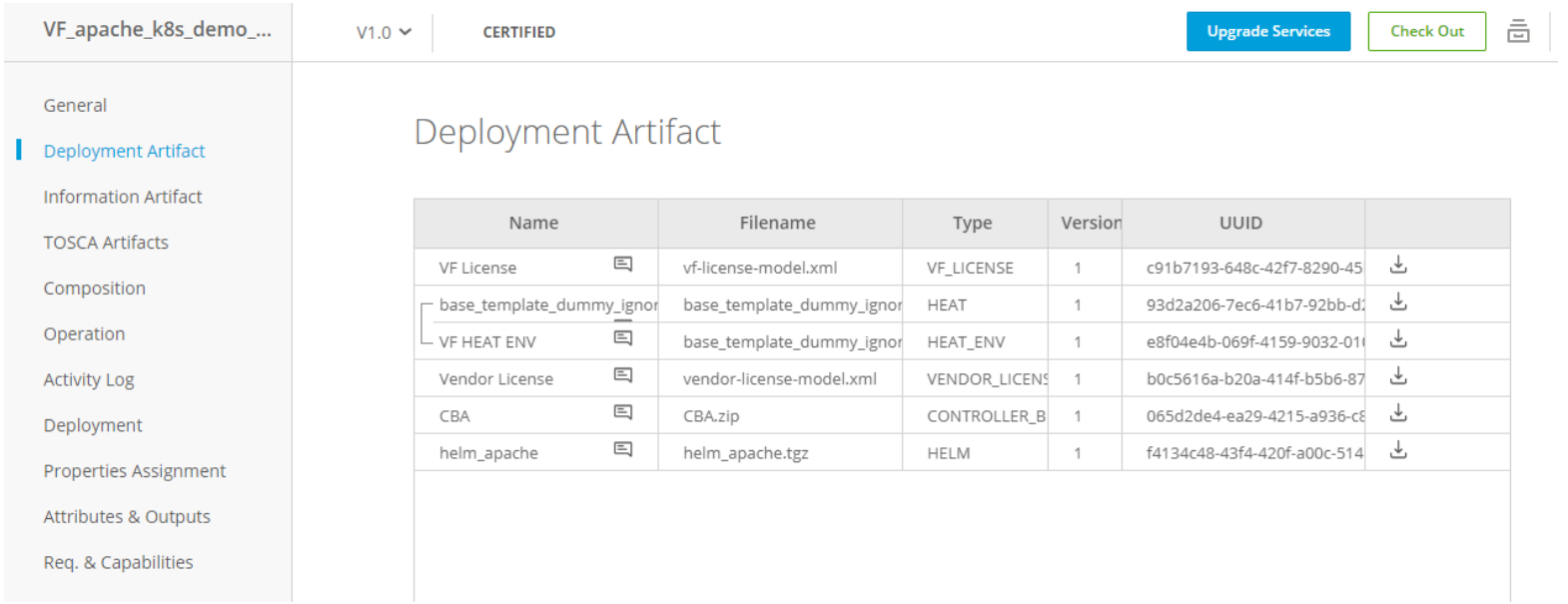


ONAP modeling concept (AAI)



CNF Onboarding

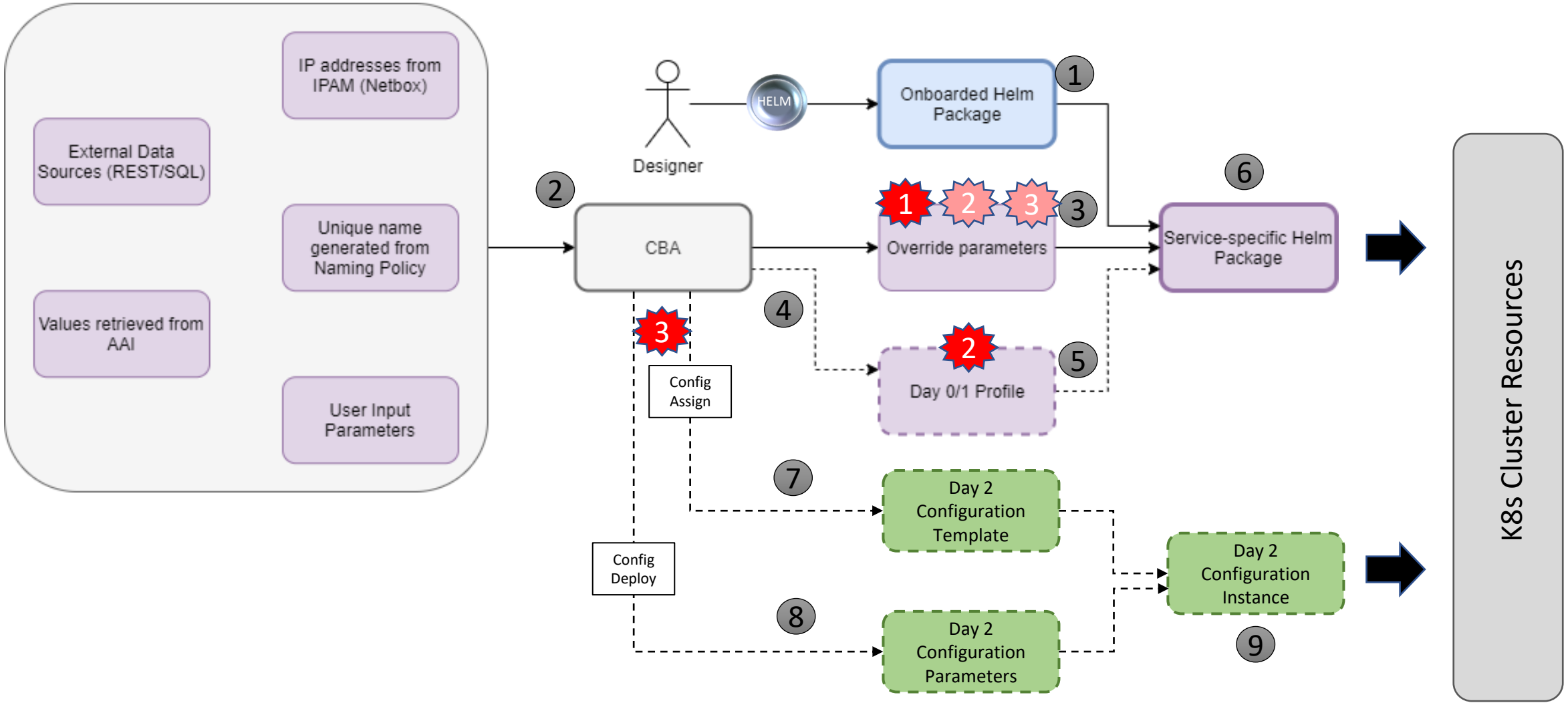

```
{
    "name": "simpleCNF",
    "description": "",
    "data": [
        {
            "file": "CBA.zip",
            "type": "CONTROLLER_BLUEPRINT_ARCHIVE"
        },
        {
            "file": "helm_apache.tgz",
            "type": "HELM",
            "isBase": "true"
        }
    ]
}
```



- Standard Simple VSP Package (ZIP)
- In the future may be replaced with ASD

- Service -> VF -> VF-Module (Helm Package)
- One VSP may have many Helm packages
  - They will be instantiated together
  - Eventual dependencies to be defined only during instantiation

Helm Package Day 0/1 + Day2



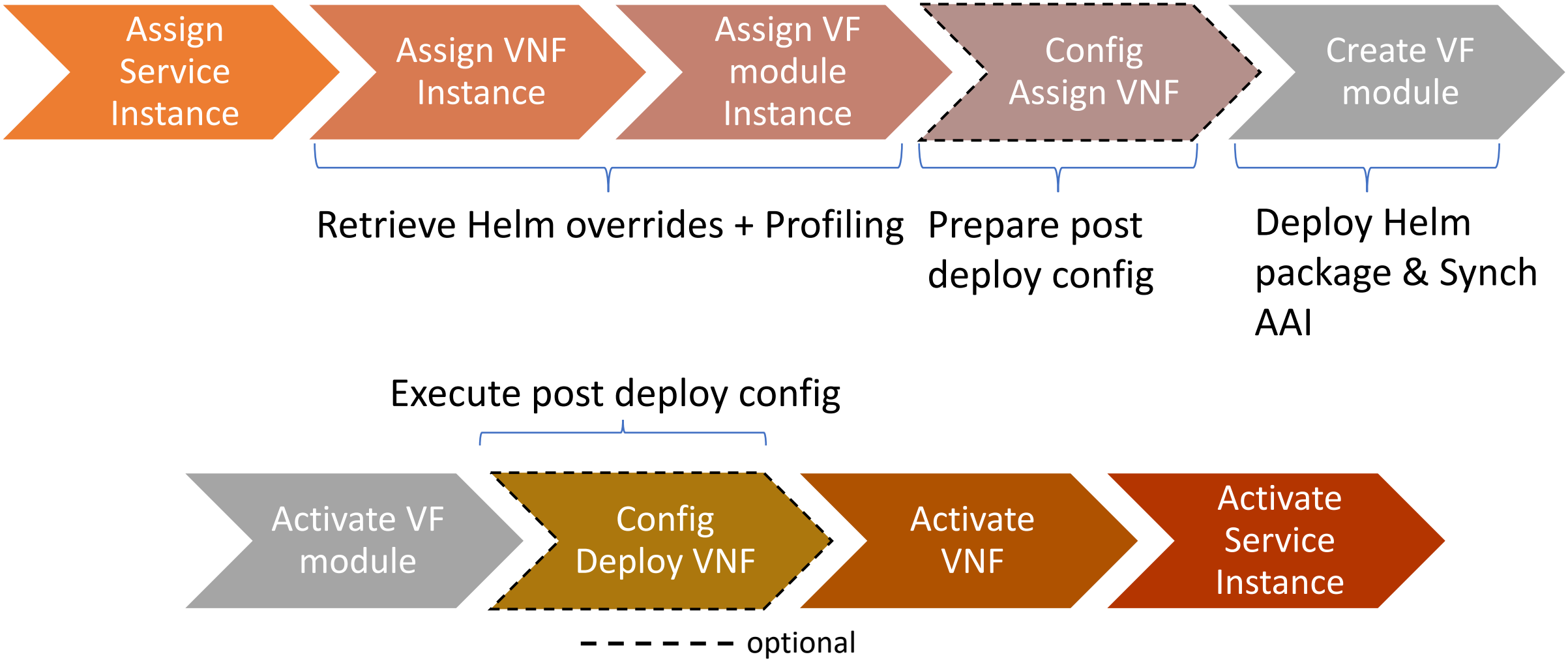
CNF Day 0 – Helm Enrichment


```
"resource-assignment": {
    "steps": {
        "resource-assignment": {
            "description": "Resource Assign Workflow",
            "target": "resource-assignment",
            "activities": [
                {
                    "call_operation": "ResourceResolutionComponent.process"
                }
            ],
            "on_success": [
                "profile-upload"
            ]
        },
        "profile-upload": {
            "description": "Generate and upload K8s Profile",
            "target": "k8s-profile-upload",
            "activities": [
                {
                    "call_operation": "K8sProfileUploadComponent.process"
                }
            ]
        }
    },
```

1
2
3
2

- CNF instance based
- Modifies Helm package from VSP
- Part of Resource Assignment in CDS
- Native mechanisms in CDS
  - Customizable by CBA
- Modification of Helm values
  - Main
  - Nested
- Modification of Helm templates in the package from VSP
- Provisioning of new Helm templates in the package from VSP

Instantiation (macro mode) (1)



Instantiation (macro mode) (2)

- VNF and VF-Module Ordering
  - Order of CNF instantiation if more than one CNF in Service
  - Order of VF-Module instantiation
  - The order is not applied on delete
- Config-deploy for reconfiguration and coordination
  - Check CNF status
  - Reconfigure CNF
  - Synchronize configuration with other components/systems
  - Config-deploy may be used to coordinate CNF/Helm package deployments

CNF Day 2 – Config Preparation

LFN Developer & Testing Forum

```
"config-assign": {
    "steps": {
        "config-setup": {
            "description": "Gather necessary input for config template upload",
            "target": "config-setup-process",
            "activities": [
                {
                    "call_operation": "ResourceResolutionComponent.process"
                }
            ],
            "on_success": [
                "config-template"
            ]
        },
        "config-template": {
            "description": "Generate and upload K8s config template",
            "target": "k8s-config-template",
            "activities": [
                {
                    "call_operation": "K8sConfigTemplateComponent.process"
                }
            ]
        }
    },
```

3

- CNF instance based
- Config Template (CfT)
  - Helm package
  - Build or modified by CDS
  - Part of CBA
- CfT preparation is part of Config-Assign in CDS
- Native mechanisms in CDS
  - Customizable by CBA
- Config Setup merges data
  - CBA
  - AAI i.e. vf-modules info
  - MDSAL – i.e. resolved Day 0
  - K8s – i.e. k8s resource status info
  - Kotlin, Python, REST
  - Complex JSON

CNF Day 2 – Config Creation


```
"config-deploy": {
    "steps": {
        "config-setup": {
            "description": "Gather necessary input for config init and status verification",
            "target": "config-setup-process",
            "activities": [
                {
                    "call_operation": "ResourceResolutionComponent.process"
                }
            ],
            "on_success": [
                "config-apply"
            ],
            "on_failure": [
                "handle_error"
            ]
        },
        "config-apply": {
            "description": "Activate K8s config template",
            "target": "k8s-config-apply",
            "activities": [
                {
                    "call_operation": "K8sConfigTemplateComponent.process"
                }
            ],
            "on_success": [
                "status-verification-script"
            ]
        },
```

3

- CNF instance based
- Config Instance (CfI)
  - Instantiates CfT
  - Provides overrides for CfT
- CfI creation is part of Config-Deploy in CDS
  - Creates new k8s resources
  - Modifies k8s resources of existing CNF instance
- Native mechanisms in CDS
  - Customizable by CBA
- In vFW CNF Use Case followed by simple Status Check
  - Checks Pod Status until „Running”
  - Fails after 30 retries

CNF Upgrade Support (1)

- No Helm a'like package upgrade support — REQ-890 (Jakarta)
- Build and Replace Scenario Supported
  1. New VSP with modified Helm package
  2. Modify VF base on new VSP
  3. Modify Service and include new VF version

  — REQ-627 (Istanbul)
  4. Run Service Instance Upgrade workflow
  5. Delete Old Version CNF Instance
  6. Create New CNF Instance

  — REQ-883 (Istanbul+)

CNF Upgrade Support (2)

- CNF Update with Config API
  - Include Helm package change into CBA
    - Whole package or only changed part
  - Create custom workflow in CBA for Update
  - Add Step for Creation of Config Template (Helm Package)
  - Add Step for Creation of Configuration Instance
  - Distribute new CBA
  - Execute CDS Workflow (Standalone or over SO)
- Input values are taken from existing Instance and may be amended
- Will not work properly if change considers deletion of the resource
- No update hooks executed

Future Steps – Jakarta++

- CNF Upgrade
  - From SDC Onboarding with new Helm package definition
  - Upgrade of existing service instance with CNF inside
- AAI Synchronization after instantiation
  - When something changes in result of Day2
- Policy & CLAMP support for CNFs
- Integration with external k8s orchestrators
- Utilization of ASD and new information hold there

REQ-890

What's NEXT?

LFN Developer & Testing Forum