



# SOL003 Adapter Architecture, Technical Debt and Roadmap




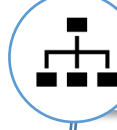



Kista, 11-14 June, 2019

Byung-Woo Jun, Michael Morris, Gareth Roper, Waqas Ikram, Ericsson

Seshu Kumar, Huawei

Yan Yang, CMCC

# Agenda

-  SOL003 Adapter Requirements
-  SO VNFM Adapter Architecture in Dublin
-  SO VNFM Adapter Features and Candidates (Dublin, El Alto, Frankfurt)
-  SO VNFM Adapter Components and Testing & Sequence Diagrams
-  SO VNFM Adapter Architecture Technical Debt in Dublin
-  VNFM Adapter –Refactoring for Frankfurt
-  Roadmap Proposals and Discussions  
(Package Management, ETSI Catalog, SOL003 Adapter Placement)

# SOL003 Adapter Requirements

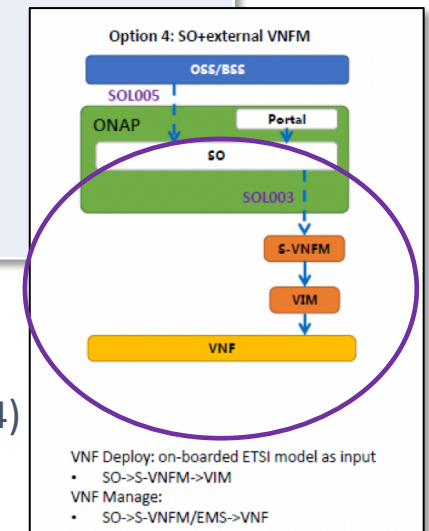


## SO VNFM Adapter supports the following requirements:

- ONAPARC-310: SO Adapter which uses SOL003 to connect to S/G VNFM
- ONAPARC-315: ONAP interfaces with an external VNF Manager using ETSI NFV SOL003
- ONAPARC-390: ONAP tracking of VNF dependency on an external ETSI compliant VNF Manager (VNFM)
- SO-1508: ETSI Alignment – SO SOL003 plugin support to connect to external VNFMs
  - Leverage ETSI standards for VNF LCM
  - Generic VNFM Adapter, supporting SOL003-compliant SVNFM
  - Support SOL003 APIs for VNF LCM
    - Create/Instantiate/Terminate/Delete (including Granting/Subscription/Notification) in Dublin
    - More APIs to support in El Alto & Frankfurt

Note: it is one of the VNF provisioning options (option #4)

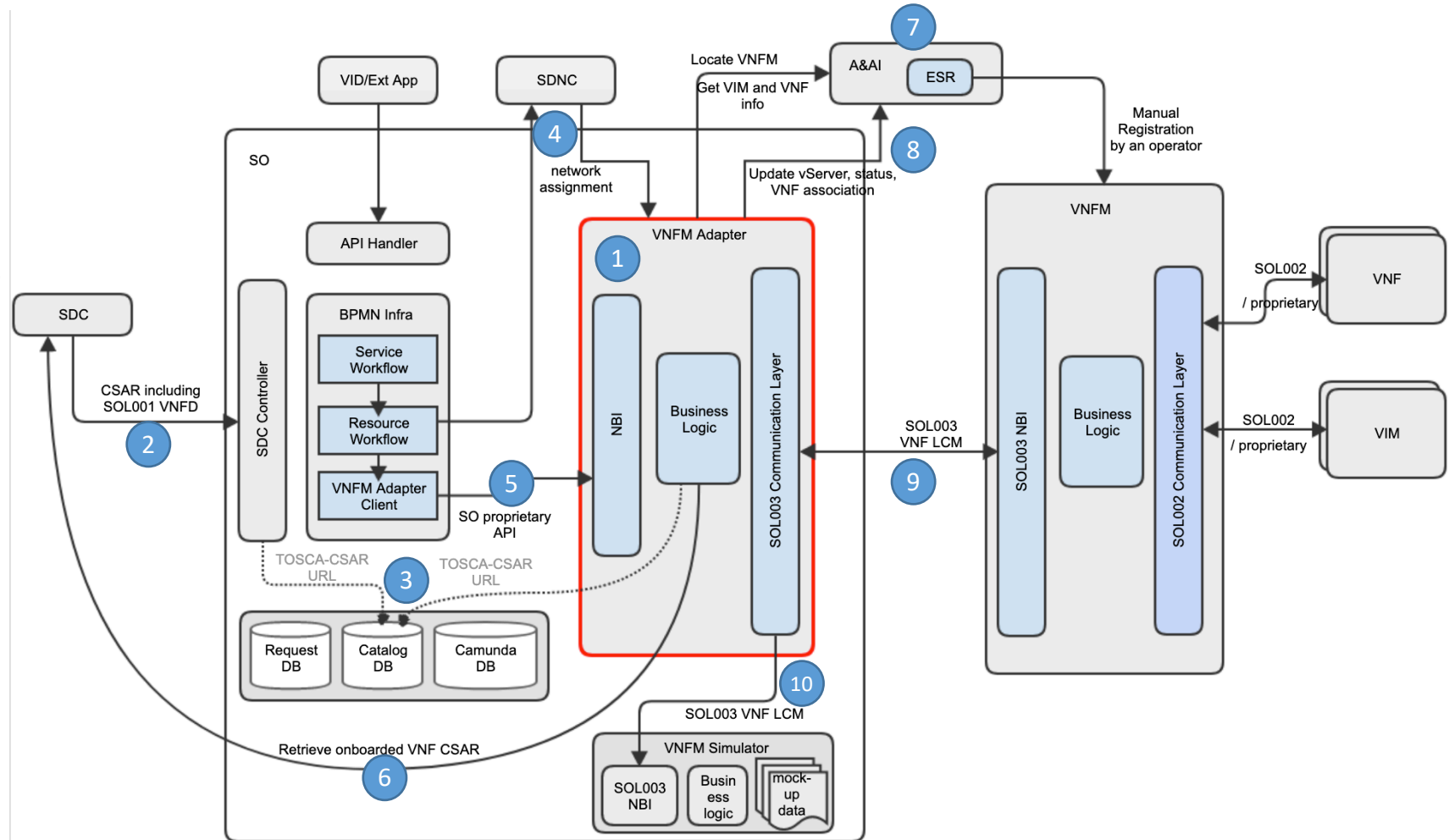
- source: VNF Management in ONAP



# SO VNFM Adapter Architecture in Dublin



1. VNFM Adapter is an SO Microservice component, running in a container (minor coupling with SO, & its relocation is under discussion)
2. SO gets CSAR including SOL001 VNFD and the original vendor VNF package
3. SO stores CSAR to Catalog DB, and VNFM Adapter retrieves CSAR from Catalog DB (it did not happen in Dublin and design will be updated)
4. Network Assignment is handled to/from SDNC
5. Interfaces between SO BPMN Infra (VNF-level) and VNFM Adapter are SO-specific
6. VNFM Adapter gets VNFDs from SDC directly (Dublin restriction; SOL004 Package Management /Distribution is under discussion – see proposals)
7. Operator registers VNFMs into A&AI ESR, and VNFM Adapter locates a proper VNFM based on VNF NF Type – see technical debt
8. Associations between VNFM and VNF instances are made in A&AI.
9. Interfaces between VNFM Adapter and SVNF are SOL003-based (2.5.1)
10. For integration testing, VNFM Simulator is used (currently, it is an SO microservice component)

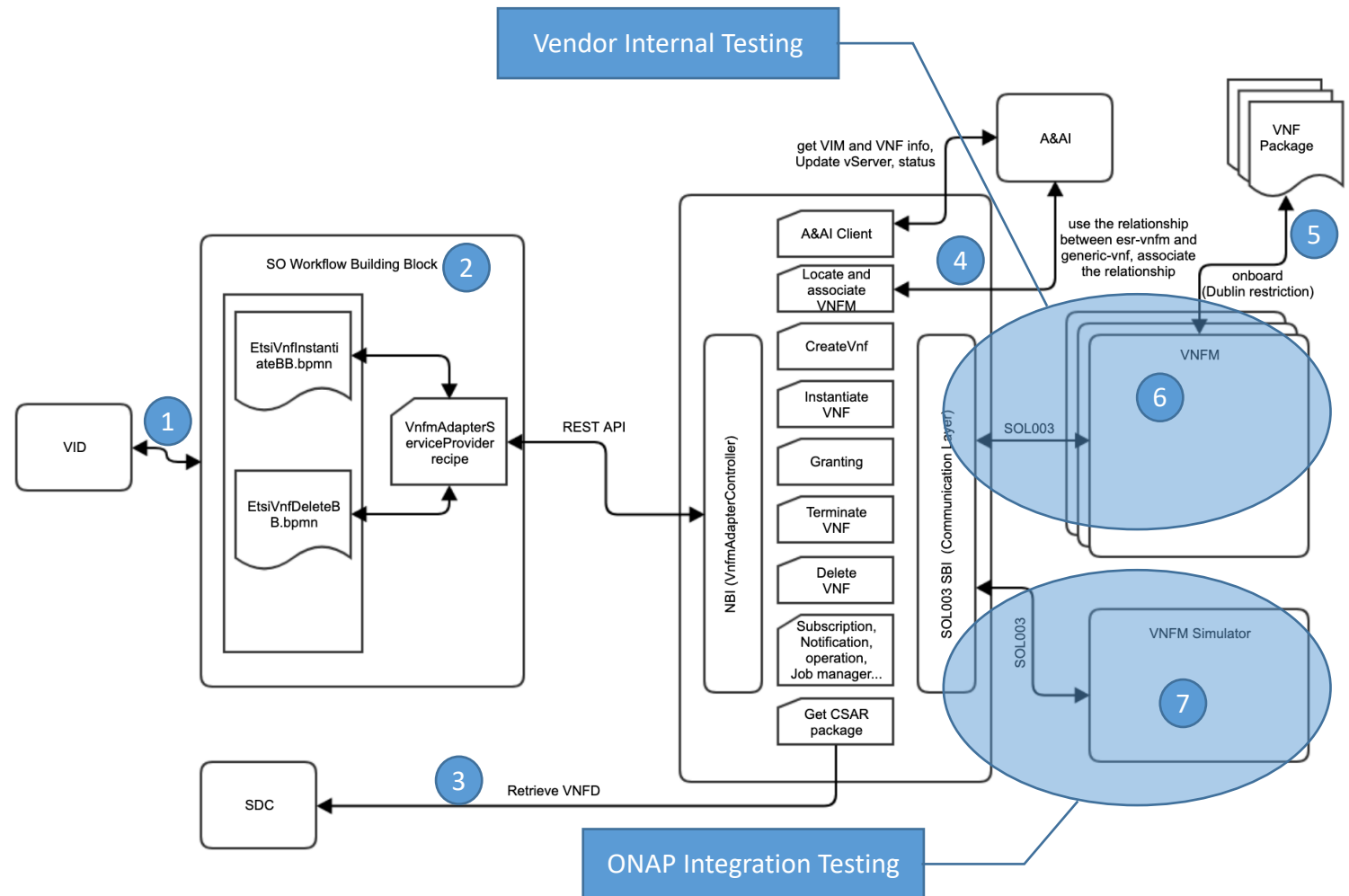


For more architecture and design details: <https://wiki.onap.org/pages/viewpage.action?pageId=48529911>

# SO VNFM Adapter Components and Testing

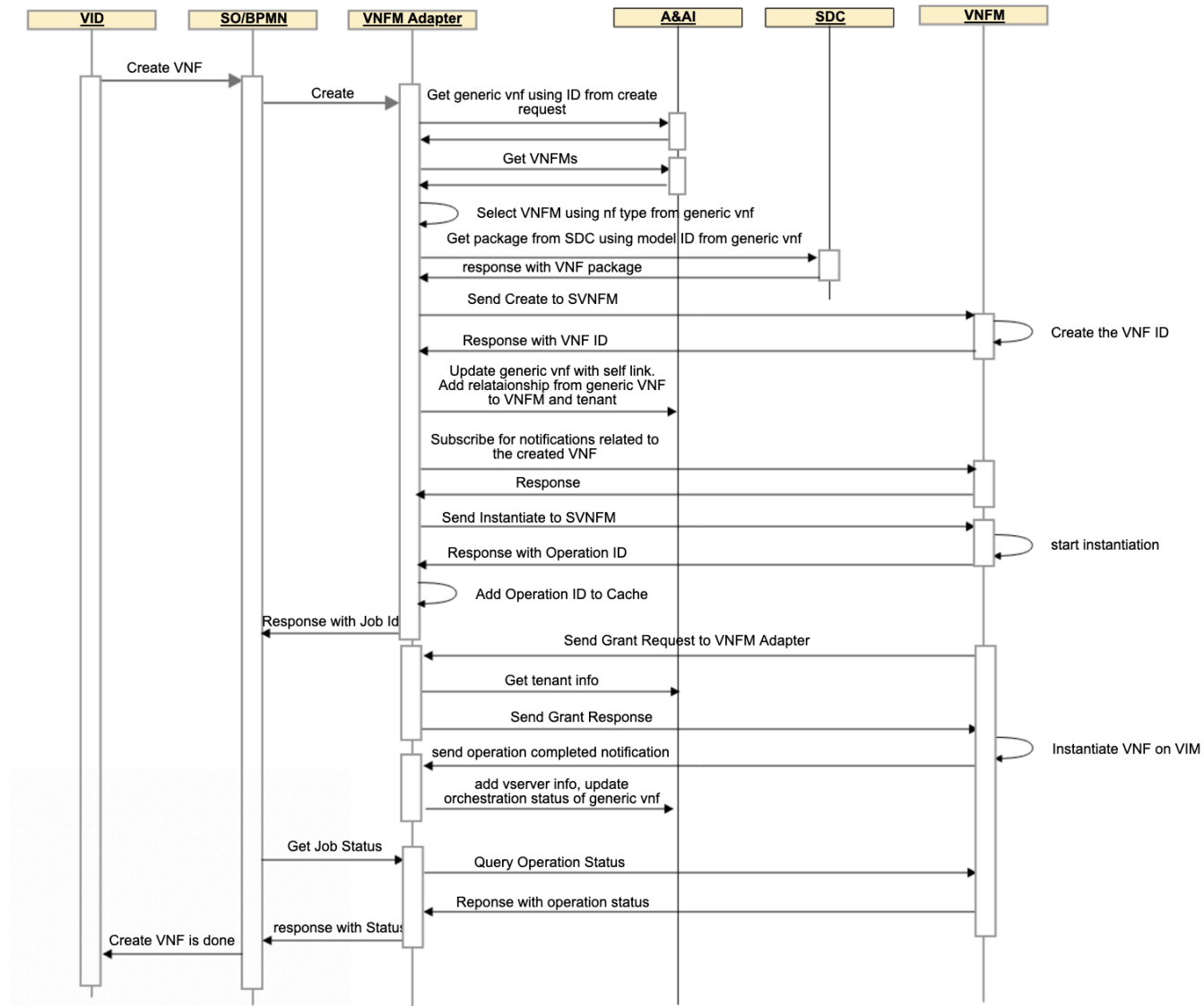


1. VID sends Ala Carte-based VNF requests to SO over GR-API
2. SO Workflow Building Blocks support 1) Create and Delete VNF and 2) Monitor VNF LCM.
  - A. SO CreateVNF Workflow handles both Create and Instantiate VNF operations in the VNFM Adapter
  - B. SO DeleteVNF Workflow handles both Terminate and Delete VNF operation in the VNFM Adapter
3. VNFM Adapter retrieves VNFD directly from SDC
4. VNFM Adapter locates VNFM, based on A&AI ESR VNF registration.
5. VNF Package is onboarded to the vendor VNFM directly because the the SOL004 package distribution is broken (Frankfurt enhancement point) – Dublin restriction
6. VNFM vendors test VNF LCM with their SVNFMs
7. ONAP integration testing tests VNF LCM by using the VNFM Simulator



# SO VNFM Adapter Run-Time Sequence Diagram

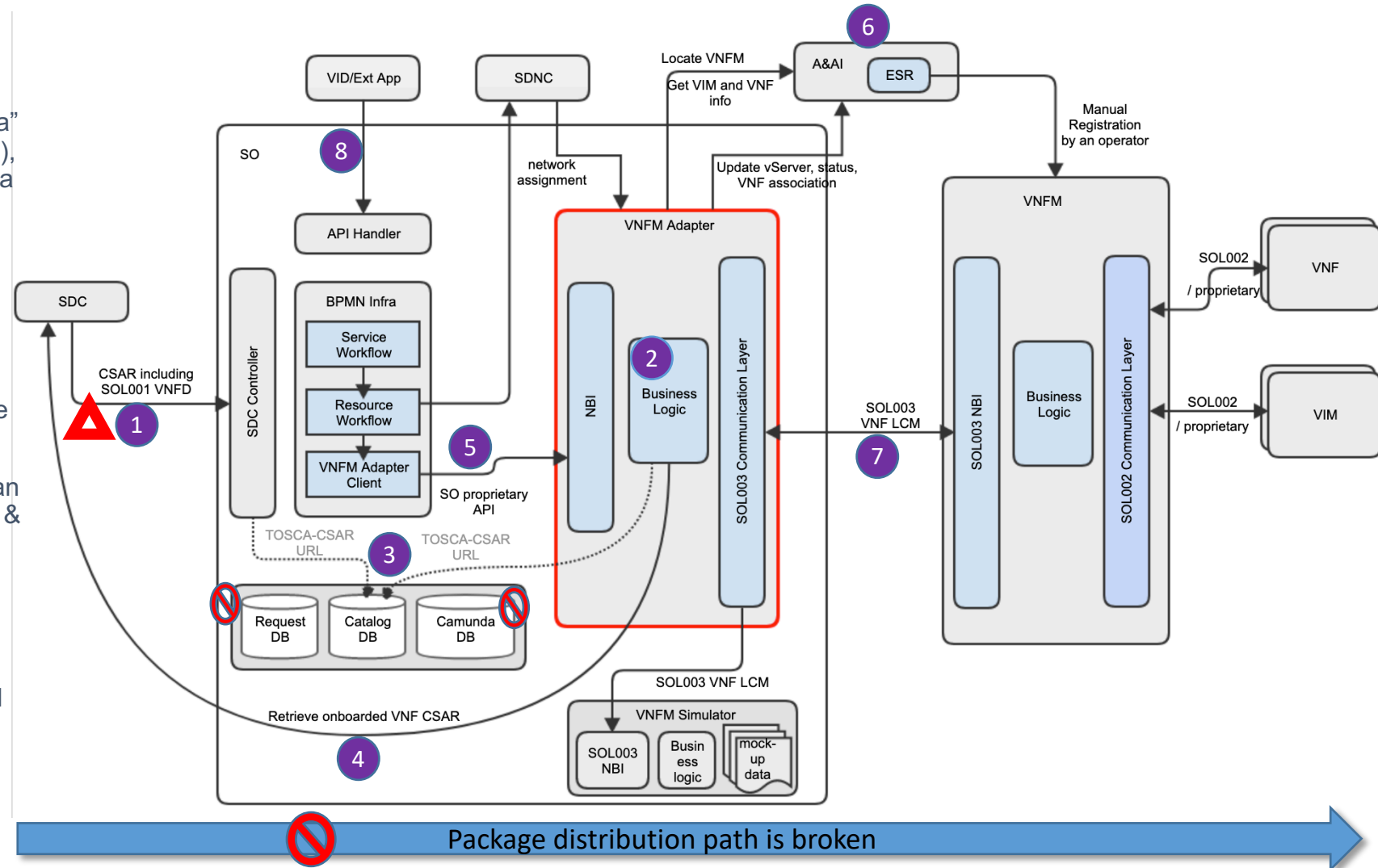
- The diagram depicts VNF Creation and Instantiation run-time sequences.



# SO VNFM Adapter Architecture Technical Debt in Dublin



1. SOL004 and SOL001 VNF distribution is only partially supported due to SDC limitations
  - A. Only PNF onboarding was tested in Dublin
  - B. No proper SOL001 mapping to/from SDC AID DM
2. VNFM Adapter by default looks for the "TOSCA.meta" file in the TOSCA-metadata directory (as per Sol004), however this can be configured to look for the file in a different directory in order to support SDC Package Structure.
3. SO does not support ETSI Catalog - needs proper ETSI package management
4. VNFM Adapter gets VNFDs directly from SDC
5. MSB could be used for locating VNFM Adapter in the future
6. Selecting a VNFM is based on VNF NF Type (this can be enhanced with vnf info:type, Cloud Region, FM & PM data and others)
7. Authentication between the VNFM Adapter and the VNFM is needed
8. Only Ala Carte-based requests are supported by SO over GR-API; End-to-End use cases were not tested



# SO VNFM Adapter Features and Candidates



## List of Features and Candidates

### Features included in Dublin

- Create VNF, Instantiate VNF, Terminate VNF and Delete VNF, including Granting, Subscription and Lifecycle Notifications
- Tracking capability: which VNFM instance has handled which VNF instance.
- BPMN Building Block Workflows and Java-based recipes for Create (Create+Instantiate) and Delete (Terminate+Delete) VNF
- VNFM Simulator, which is used for Integration Testing

### Feature Candidates for El Alto

- Authentication between the VNFM Adapter and the VNFM
- VNFM Simulator enhancement and refactoring
- CSIT of SOL003 ETSI Alignment (SDC → SO → SOL003 VNFM Adapter → VNFM)
- SO-Monitoring HTTPs Support
- Preload using user\_param (without UI changes)
- Add support of Query

### Feature candidates for Frankfurt

- VNFM Adapter exposes its NBI to any VNFM Adapter client
- Package Management of SOL004 including SOL001, based on SOL005 and SOL003
- Mapping between ScalingAspect+Delta and VF-Module for Scaling
- Policy-based Scaling (with VNF Indicator & VES event handling)
- SO ETSI Catalog DB handling for NS and VNF packages
- Addition of ETSI SOL003 operations (Modify-Scale-Operation Status, FM, PM, Heal, VNF Indicator, Grant enhancement, & retry, rollback, failing, cancelling, Resource Quota Available Notification – to be determined for operation selections and priorities)

### Other Related Feature Requests (SDC Enhancement Requests) for El Alto or Frankfurt

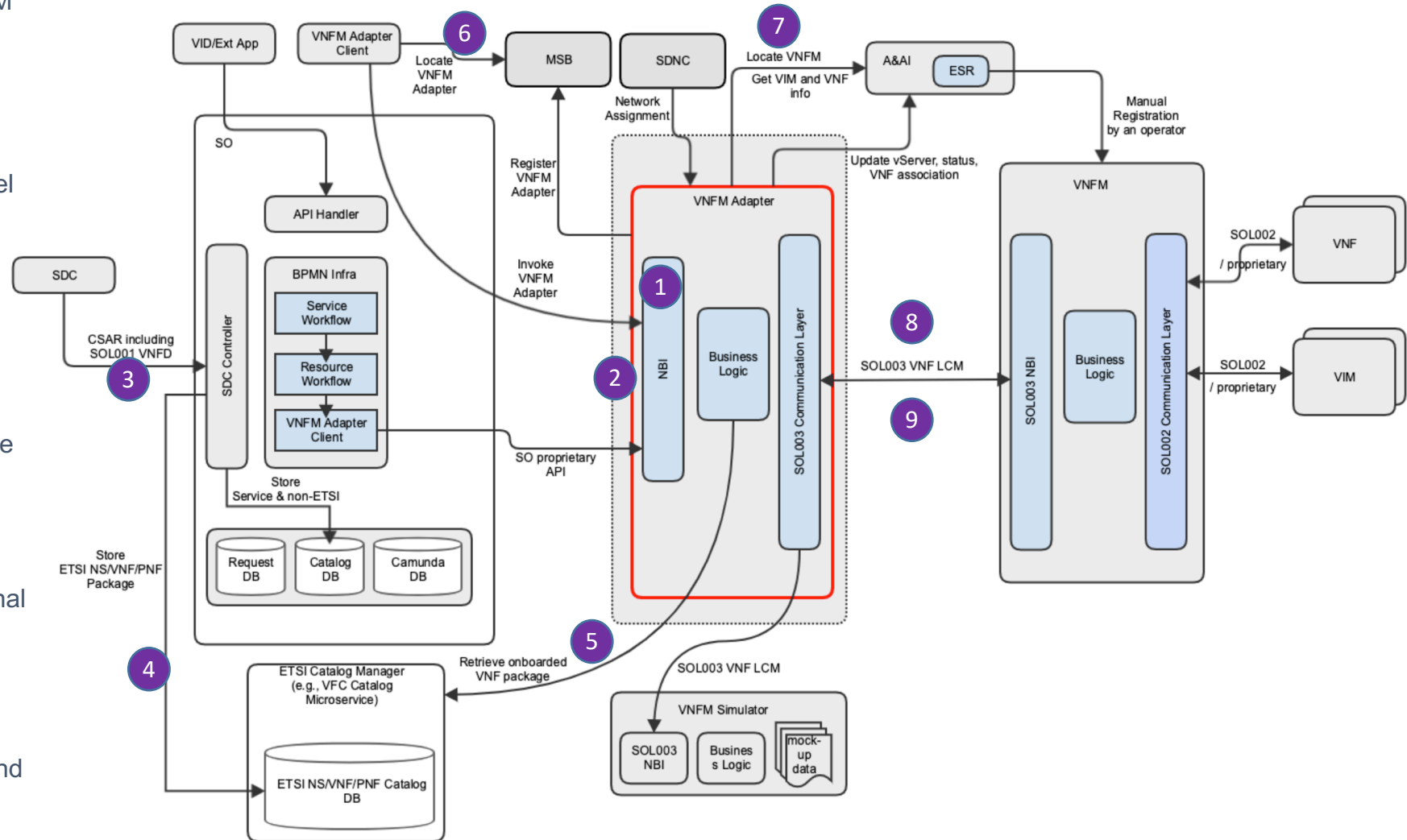
- SOL004 VNF onboarding and distribution support (SDC-2072, SDC-2282)
- Mapping between SOL001 VNFD to SDC AID DM, including ScalingAspect+Delta and VF-Module - Not all VNFD needs to be transformed to the SDC AID DM



# VNFM Adapter Enhancements For Frankfurt



1. VNFM Adapter exposes its NBI to any VNFM Adapter client
2. Interfaces are refactored to be generic (accessed by other ONAP/External components) - TBD
3. SDC CSAR including the SDC internal model and the vendor original SOL004 package is supported
4. SO leverages ETSI Catalog Microservice for ETSI-based NS, VNF and PNF
5. VNFM Adapter retrieves VNF package from Catalog Manager
6. VNFM Adapter Client uses MSB to locate the VNFM Adapter
7. VNFM Adapter locates a VNFM based on a better VNFM locating mechanism
8. VNFM Adapter and SVNFM support additional SOL003 operations, possibly including modification for configuration
9. VNFM Adapter and SVNFM support authentication and authorization
  - AAF will be used for authentication and authorization



# SDC VNF/PNF Onboarding Flow



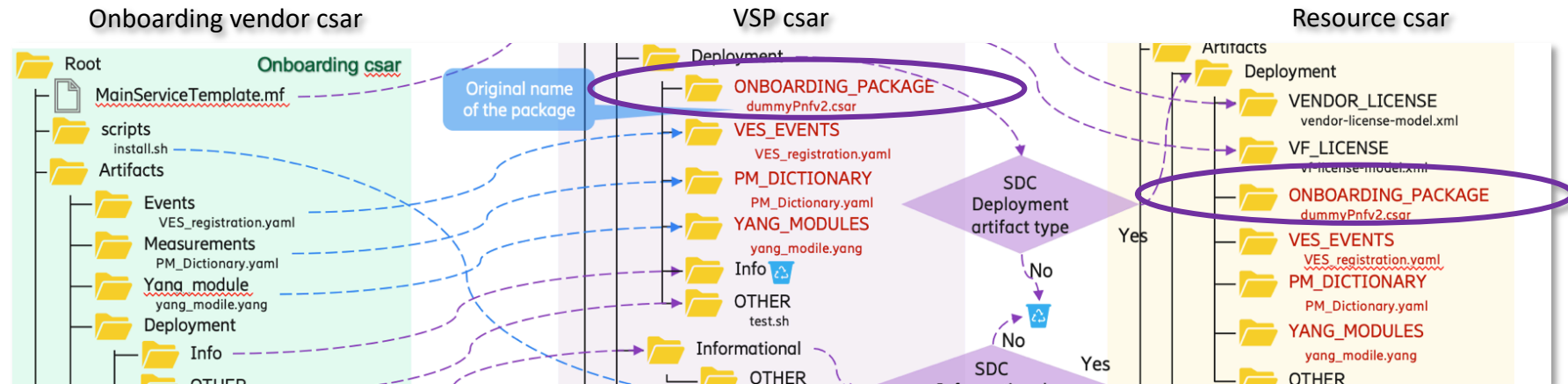
- SDC supports VNF/PNF onboarding with .zip and .csar file extensions
- During the onboarding process, SDC checks an onboarding file extension.
  1. If the file has .csar extension, SDC considers it as non-SOL004 csar and processes the existing VNF TOSCA CSAR onboarding procedure.
  2. If the file has .zip extension, SDC unzips it and checks if it contains .cert and .cms.
    - A. If not, SDC processes the existing Heat template onboarding procedure
    - B. If yes, SDC retrieves root certificate and validates security.
      - A. If the security validation is OK, SDC
        - I. validates packages,
        - II. maps SOL001 VNFD into SDC AID DM
        - III. adds the original CSAR as onboarding package artifacts.
- Question: Vendor VNF package certificate and signature need to be sorted out.
- Note: currently, VF-C supports CSAR-format without certificate or signature

# SDC Package Transformation



- From the Ericsson package transform slide deck for PNF
  - Applicable to VNF packages
- SDC VSP and Resource csar files have the ONBOARDING\_PACKAGE, which contains the original vendor VNF package
- SOL001 VNFD mapping to SDC AID DM needs to be defined
- The VNFM and external NFVO use the original vendor VNF/NS packages.

Source: the Ericsson package transform slide deck



# Package Management based on SOL005 and SOL003

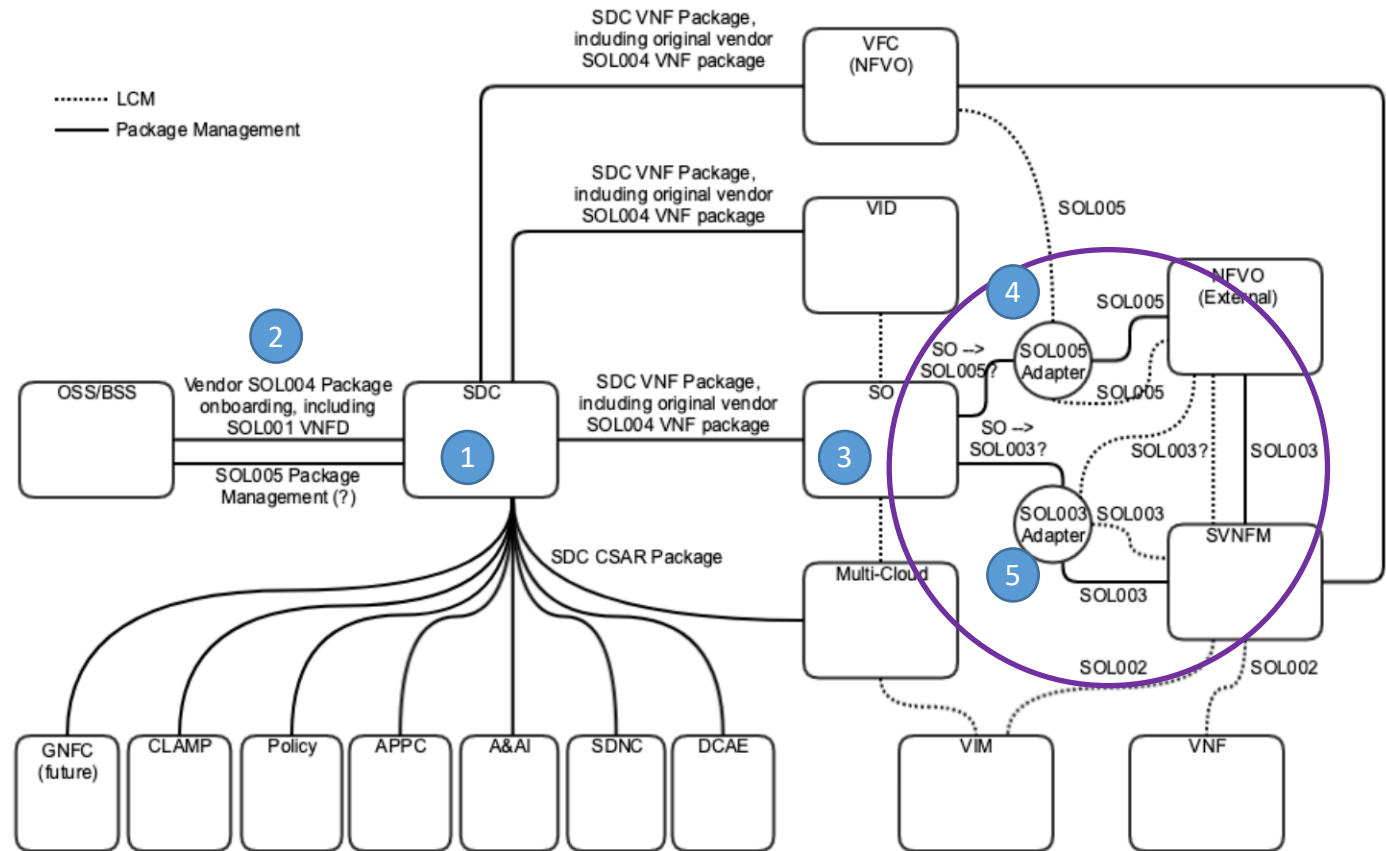


The diagram depicts a possible package management architecture. It is posted in ONAP Wiki, <https://wiki.onap.org/display/DW/SOL005+and+SOL003+Package+Management>

1. SDC supports SOL004 VNF/PNF package onboarding, and stores the original vendor VNF/PNF package inside the SDC package – Ericsson contribution in Dublin
  - A. PNF onboarding was tested
  - B. VNF onboarding is being tested in El Alto / Frankfurt
2. SDC support VNF/PNF package onboarding and/or accepts VNF/PNF package management interfaces from OSS/BSS via SOL005 Package Management APIs (Create...)
3. ONAP Runtime components store SOL004 Packages as needed
  - A. For the SO case, see the subsequent page, SO ETSI Catalog DB Support for NS, VNF and PNF
4. SOL003 VNFM Adapter provides VNFMs Query/Fetch VNF packages/contents/artifacts, Reading VNFD and subscription/notification services – see the following page
5. SOL005 Adapter provides NSD/PNFD management and SOL005 VNF package management – see the following page

**Note:**

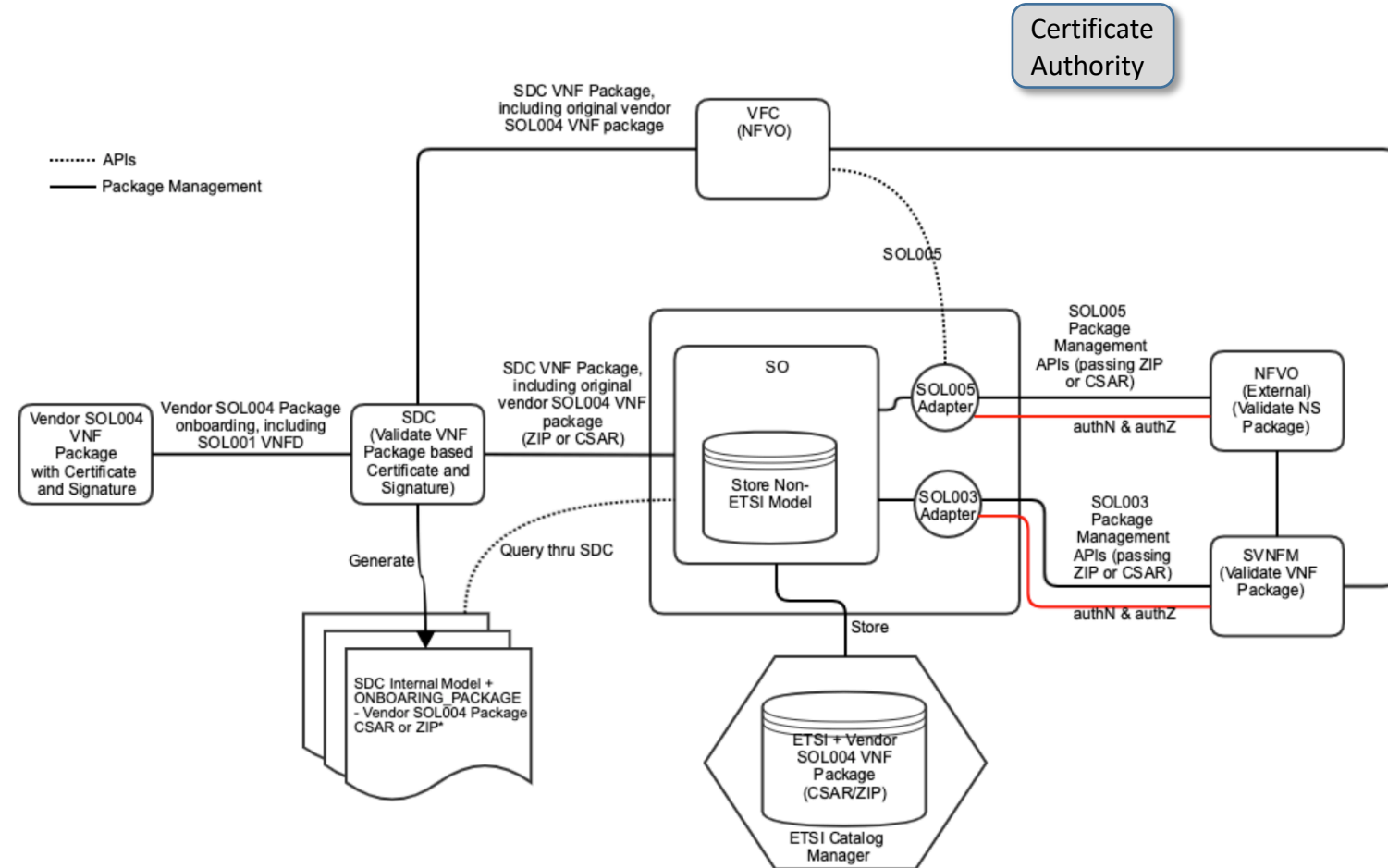
- I. SOL007 (NS package) support is under discussion.
- II. Location of SOL003 and SOL005 Adapters are under discussion, and where/how the adapters get packages is under discussion



# Package Security on SOL005 and SOL003



- Requirement: External NFVO and SVNFM need to validate incoming ETSI package
- Vendor SOL004 VNF Package with certificate and signature is onboarded into SDC
  - ZIP-format VNF package includes CSAR, Signature and Certificate
- SDC validates VNF package based on the certificate and signature
- SDC generates SDC internal model plus the vendor SOL004 package CSAR and ZIP (with certificate and signature) – the supported format is TBD based on the security requirement
- SO queries the package thru SDC
- SO stores Service + Non ETSI model
  - Not all VNFD needs to be transformed to the SDC AID DM - TBD
- SO stores ETSI / vendor VNF package CSAR or ZIP – the supported format is TBD based on the security requirement
- SOL003 Adapter provides VNF package management APIs
- SOL005 Adapter provides NS package management APIs
- If CAR-format is chosen, NFVO/SVNFM trusts integrity (authN and authZ) between the Adapters and NFVO/SVNFM.
- For certificate validation, we need to resolve Certificate Authority placement.

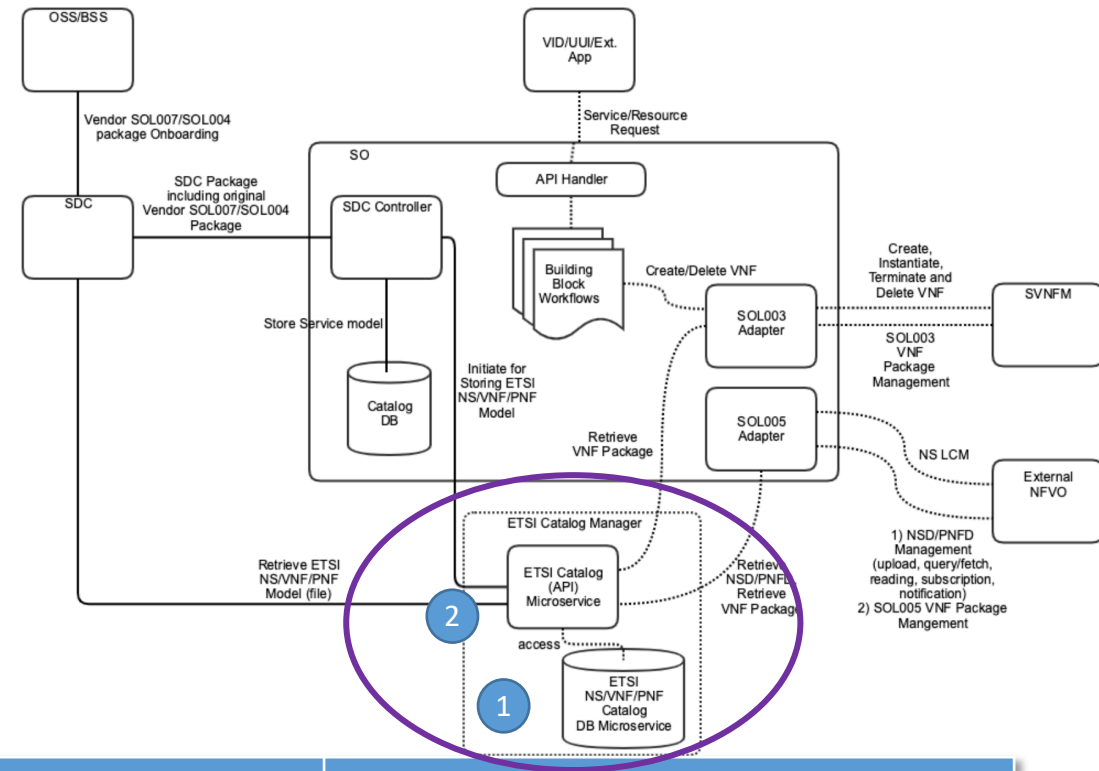


# SO ETSI Catalog DB Support for NS, VNF and PNF



Instead of building ETSI Catalog management in SO, leverage the existing ETSI Catalog Manager function – Working with CMCC and plan to discuss with CMCC sometime next week

- Consideration of leveraging VF-C Catalog Microservices for the NS, VNF and PNF catalog support, <https://wiki.onap.org/pages/viewpage.action?pageId=63996543>
    - VFC Catalog Manager function consists of two microservices: VF-C DB and VF-C Catalog. Deployment of these microservices is independent of VF-C, and the microservices have no dependency with other VF-C components – a good stepping stone towards common ETSI Catalog management
1. VF-C Catalog DB Microservice
    - Database: `nfvocatalog (vfc-nfvo-catalog-createdb.sql // create db scripts)`
    - Database Table (`vfc-nfvo-catalog-createobj.sql // create tables scripts`)
      - `Catalog_NSPackage`
      - `Catalog_VNFPackage`
      - `Catalog_PNFPackage`
      - `Catalog_SoftwareImageModel`
  2. VF-C Catalog (API) Microservice
    - `vfc-catalog docker`
- VFC Catalog REST APIs (following is part of Swagger REST APIs), [https://docs.onap.org/en/latest/downloads/0fea611e8b07fb4042e1a66ce202898c/CATALOG\\_API\\_Specification\\_v1.json](https://docs.onap.org/en/latest/downloads/0fea611e8b07fb4042e1a66ce202898c/CATALOG_API_Specification_v1.json)

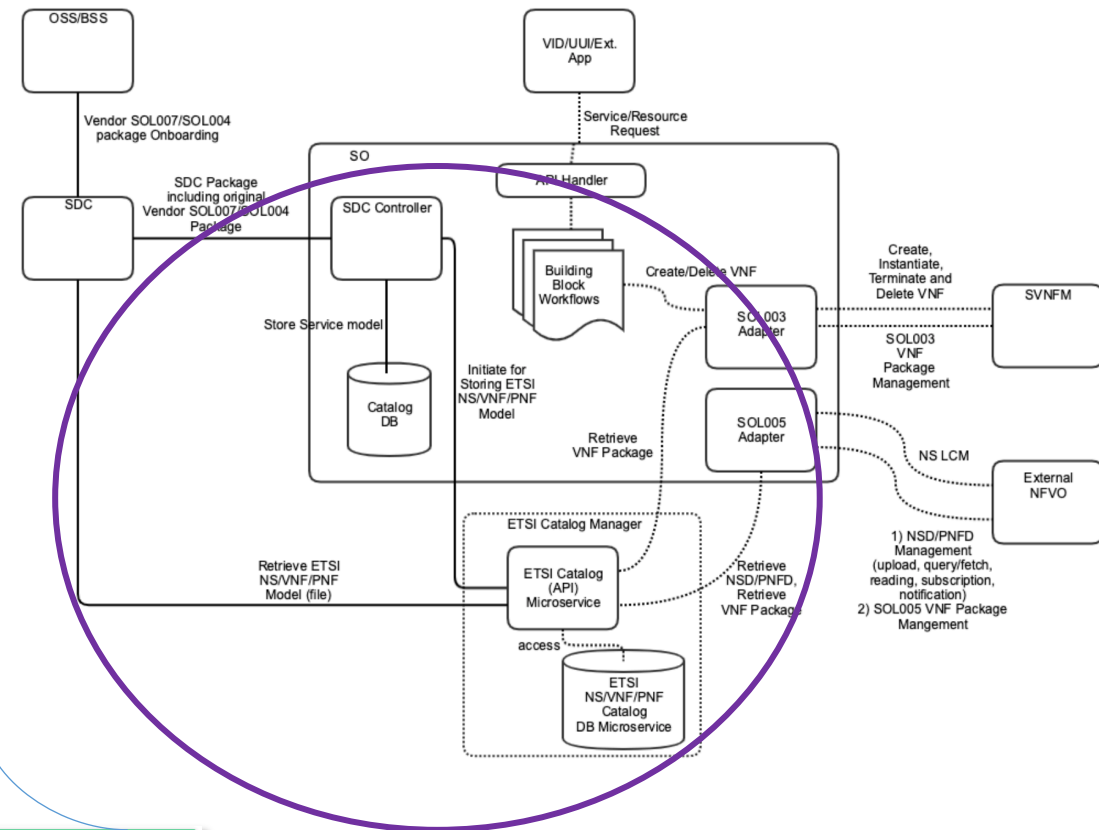


NS Package Management	VNF Package Management	PNF Management
<ul style="list-style-type: none"> <li>• GET /nspackages // query NS package info</li> <li>• POST /nspackages // NS package distribute / create</li> <li>• GET /nspackages/{csarId} // query NS package info</li> <li>• DELETE /nspackages/{csarId} // delete NS package</li> </ul>	<ul style="list-style-type: none"> <li>• GET /vnfpackages //query vnf package info</li> <li>• POST /vnfpackages // vnf package distribute / create</li> <li>• GET /vnfpackages/{csarId} // query vnf package info</li> <li>• DELETE /vnfpackages/{csarId} // delete vnf package</li> <li>• PUT /vnfpackages/{csarId}. // create/upload vnf package</li> </ul>	<ul style="list-style-type: none"> <li>• GET /pnf_descriptors</li> <li>• POST /pnf_descriptors</li> <li>• GET /pnf_descriptors/{pnfdInfold}</li> <li>• DELETE /pnf_descriptors/{pnfdInfold}</li> </ul>
Model Parsing	Job Management (used for async LCM)	More...
<ul style="list-style-type: none"> <li>• POST /parsersnd // NS package model</li> <li>• POST /parservnf // VNF package model</li> </ul>	<ul style="list-style-type: none"> <li>• GET /jobs/{jobId} // Job Status</li> <li>• POST /jobs/{jobId} // Update Job Status</li> </ul>	

# SDC – SO SDC Controller – ETSI Catalog Manager



- SO ASDCController and ToscaResourceInatall code needs to be updated for handling SOL004 packages including SOL001 VNFD and PNFD.
- Service-level catalog and other non-ETSI catalog (SDC AID DM) will be stored in SO Catalog DB.
- VNF/PNF/NS-level catalog handling would be simplified:
  - By using the ETSI Catalog Manager.
  - ETSI VNF/PNF/NS-level catalogs (i.e., original vendor packages) will be stored in ETSI Catalog DB
  - SO itself does not need to manage VNF LCM (delegates the LCM to VNFM through the SOL003 VNFM Adapter)
- In Frankfurt, VNF package management is the first focus.
- ETSI Catalog Manager POST API will be used to store the VNF packages
  - **POST /api/catalog/v1/vnfpackages // passing csarId, vimIds and labVimId**
- Question:
  - How does VFC handle VNF packages from SDC? Does use SDC AID DM? Does use VF-Modules?
    - Currently, there is NO VF-Module support from VF-C and SOL003 VNFM Adapter
  - VF-Module mapping and handling need to be sorted out.



```

POST /api/catalog/v1/vnfpackages
On distribute NF package

Parameters
Cancel

Name Description
data * required Edit Value Model
(body)
{
  "csarId": "string",
  "vimIds": [
    "string"
  ],
  "labVimId": "string"
}
    
```

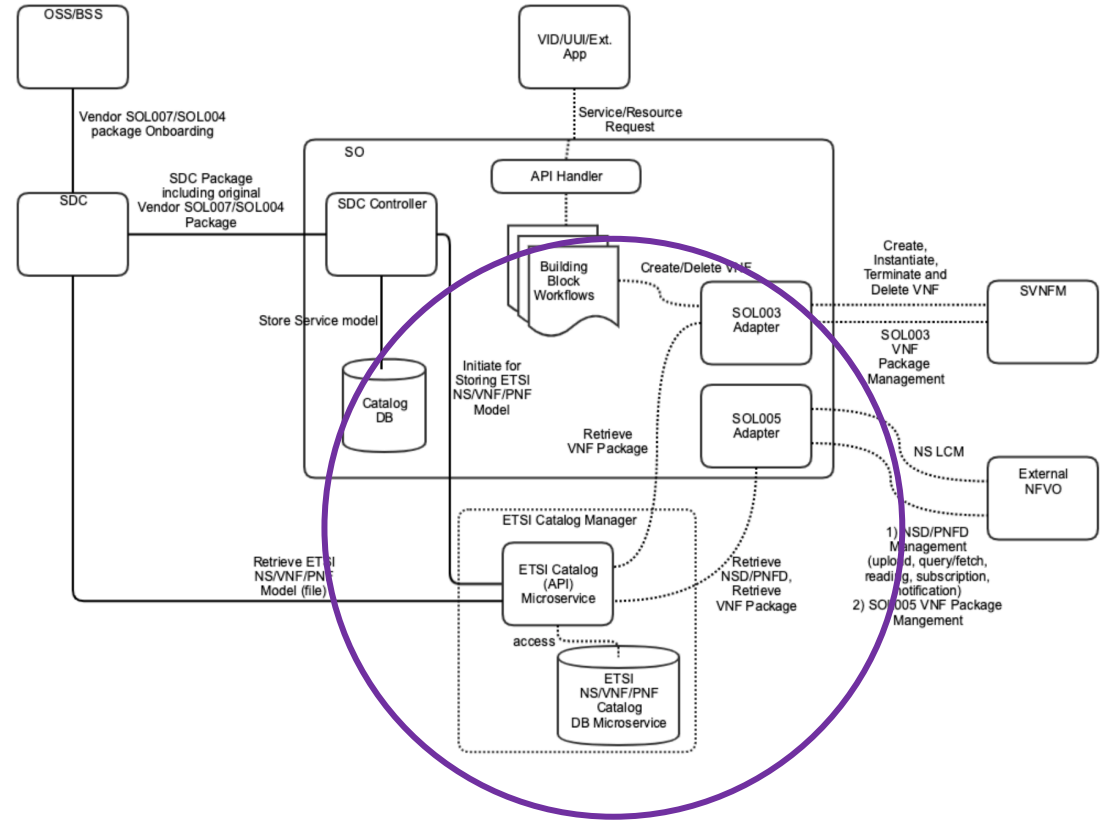
# ETSI Catalog Manager – SOL003/SOL005 Adapter



- ETSI Catalog Manager consists of two microservices:
  - ETSI Catalog API
  - ETSI Catalog DB
- ETSI Catalog Manager provides RESTful services:
  - ETSI package management for the Adapters.
    - GET /api/catalog/v1/vnfpackages/{csarId}
    - Get the original vendor VNF package
      - ETSI Catalog APIs will be enhanced to extract the original vendor VNF package from the ONBOARDING\_PACKAGE directory
  - Retrieving VNFD for SOL003 Adapter
  - Retrieving NSD/VNFD for SOL005 Adapter

```

200
Example Value | Model
{
  "csarId": "string",
  "packageInfo": {
    "vnfdId": "string",
    "vnfpPackageId": "string",
    "vnfdProvider": "string",
    "vnfdVersion": "string",
    "vnfVersion": "string",
    "csarName": "string",
    "vnfdModel": "string",
    "downloadUrl": "string"
  },
  "imageInfo": [
    {
      "index": "string",
      "fileName": "string",
      "imageId": "string",
      "vimId": "string",
      "vimUser": "string",
      "tenant": "string",
      "status": "string"
    }
  ]
}
    
```

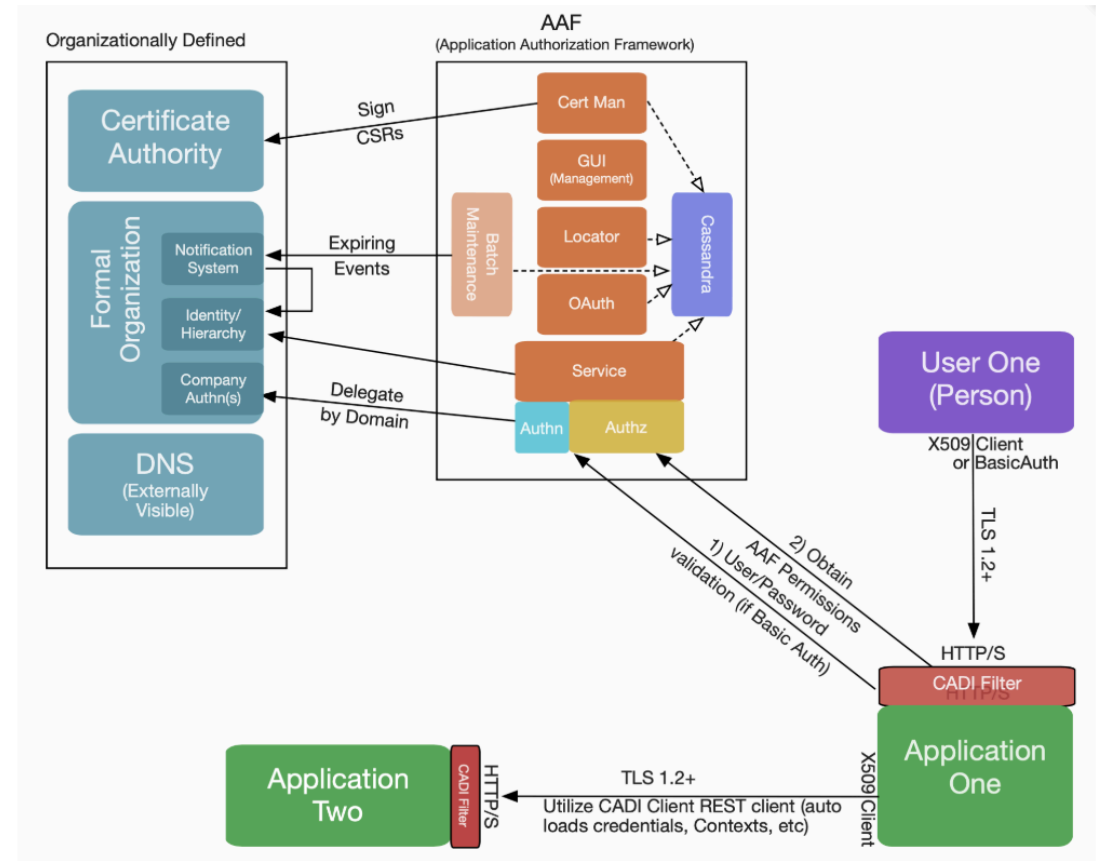




# Authentication and Authorization



- Leverage AAF for authentication and authorization to secure communications among ONAP component and SVNFM and external NFVO. The following is input from Steve Smokowski.
  - OAuth2 is not yet used in ONAP. Start with HTTP Basic Authentication with HTTPS
  - Update an application pom file and add properties; i.e., no application code changes?
    - Remove Spring Security as well, as we cannot have both in place
  - There is no need to use the CADI Rest Client at all
  - The CADI filter can be configured to handle authorization, that is the method AT&T use, or the application can enforce the authorization. It supports basic URI matching semantics
  - Generate certificates by AAF for HTTPS is the current gap.
    - We plan to resume the discussion with Elena, Steve and others sometime next week to learn AT&T implementation for this.
- Authentication and Authorization on SOL003 and SOL005 APIs need to be supported.
- How does ONAP support vendor-specific SVNFM security?
- Verizon (John D'Elia) input, <https://wiki.onap.org/pages/viewpage.action?pageId=50202249&preview=%2F50202249%2F63998229%2FVerizon+RBAC+Requirements+for+ONAP+-+042619.pptx>
  - Roles - One or more roles are assigned to each user (managed in external IdP)
  - A role maps to a set of ONAP-specific permission groups
  - Each permission group maps to a set of permissions
  - ONAP applications guard access based on permissions
  - Permissions are triplets of the form: `basename|resource|actions`



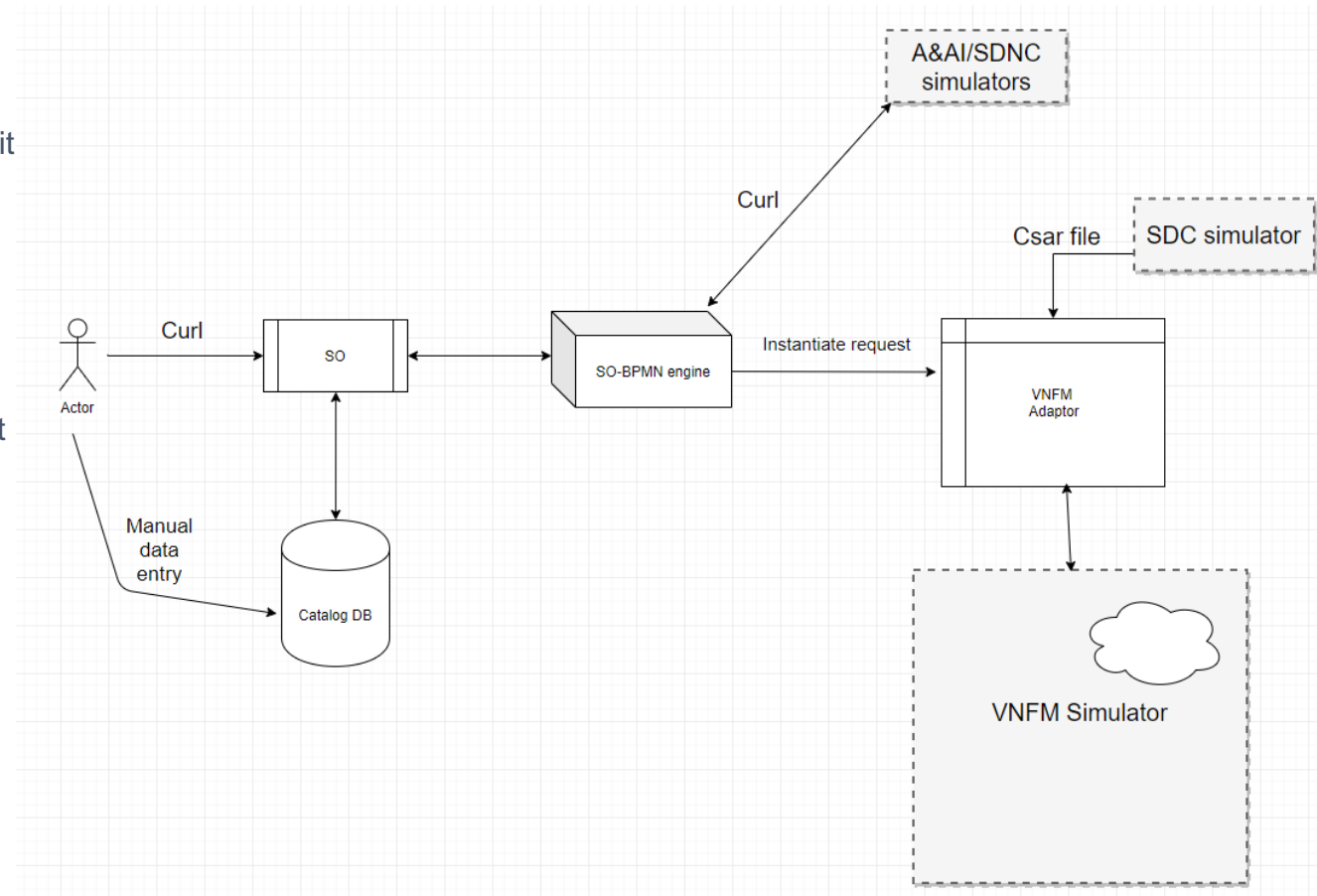
# CSIT for ETSI Alignment



## Task list

1. Simulate Request for SDC in catalog DB
  - CSIT does not have DMAAP and real SDC, so it will update service information manually
2. Compose a curl command mimicking the data sent from VID to SO and send it to the running SO container
3. Create entries in Database as needed according to the request from the user
4. Simulate A&AI, SDNC and enable interactions in A&AI & SDNC to communicate with SO-BPMN: A&AI and SDNC simulators must have a URL matching every call that is required in the BPMN flow being invoked.
5. Update BPMN to point to the simulated A&AI and SDNC : Update curl request to point to Simulators
6. Setup the VNFM Simulator for SOL003 operations
7. Create SDC simulator to respond to a request with a CSAR file.
8. Evaluate on what is considered as an acceptance criteria test cases
9. Spike: Expose REST interface for SO to give it notification instead of polling.

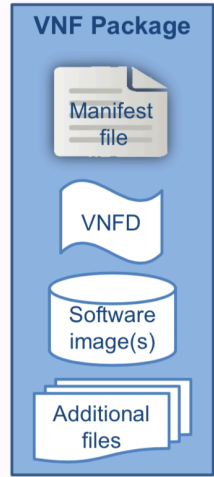
Note: detailed design needs to be settled.



# ETSI VNF Package and VNFD

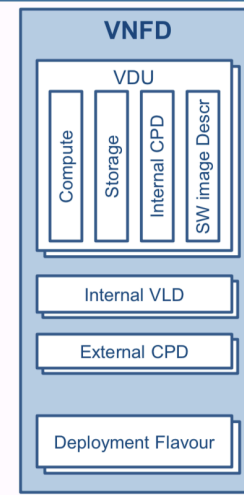


- The **VNF Package** contains:
  - the **VNF descriptor (VNFD)** that defines metadata for package onboarding and VNF management,
  - the **software images** needed to run the VNF, and
  - Manifest file** that provides package integrity and authenticity
  - (optional) **additional files** to manage the VNF (e.g. scripts, vendor-specific files etc.).
- The VNF Package is delivered by the VNF provider as a whole and is immutable (protected from modification).
- The VNF Package or its Manifest file is **digitally signed**
- The VNF Package is **stored in a repository** by the NFVO.
- The VNF Package **can be accessed by VNFM**.



Reference:  
 - ETSI GS NFV-IFA 011  
 - ETSI GS NFV-SOL 004

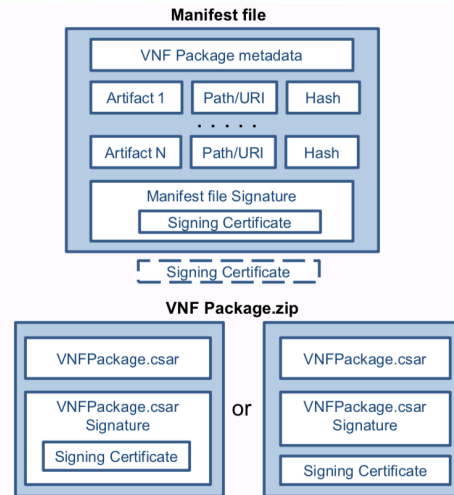
- The **VNFD** defines **VNF properties and requirements**, such as:
  - Resources needed (amount and type of Virtual Compute, Storage, Networking),
  - Connectivity:
    - External Connection Points (described via CP Descriptors, CPD).
    - Internal Virtual Links (described via VL Descriptors, VLD)
    - Internal Connection Points (described via CP Descriptors, CPD)
  - LCM behavior (e.g. scaling, instantiation), operations, and configuration
  - References to SW images, LCM scripts and other files located or referred in VNF package
  - Affinity / anti-affinity and other policy rules
  - Deployment flavours (size-bounded deployment configurations, e.g. related to capacity).
- The **VNFD** is the main input to VNF instances lifecycle management



References:  
 - ETSI GS NFV-IFA 011  
 - ETSI GS NFV-SOL 004

© ETSI 2017. All rights reserved

- Option 1: **Manifest file** - based if there are both local and external artifacts
  - A Digest hash per each artifact
  - Manifest file is signed with VNF provider private key
  - VNF provider's certificate includes a VNF provider public key
  - The certificate may be a separate artifact or included in the signature container, e.g. CMS
- Option 2: **CSAR**-based if all artifacts are located inside a CSAR
  - CSAR file is digitally signed with the VNF provider private key
  - VNF provider delivers one zip file containing a CSAR file, a signature file and a certificate file that includes a VNF provider public key
  - The certificate may be a separate artifact or included in the signature container, e.g. CMS
- Both options rely on existence in the NFVO of a root certificate of a trusted certificate authority, delivered via a trusted channel separately from a VNF package



```

!----- MRF.yaml
!----- MRF.mf
!----- MRF.cert
!----- ChangeLog.txt

!----- Tests
!----- file(s)

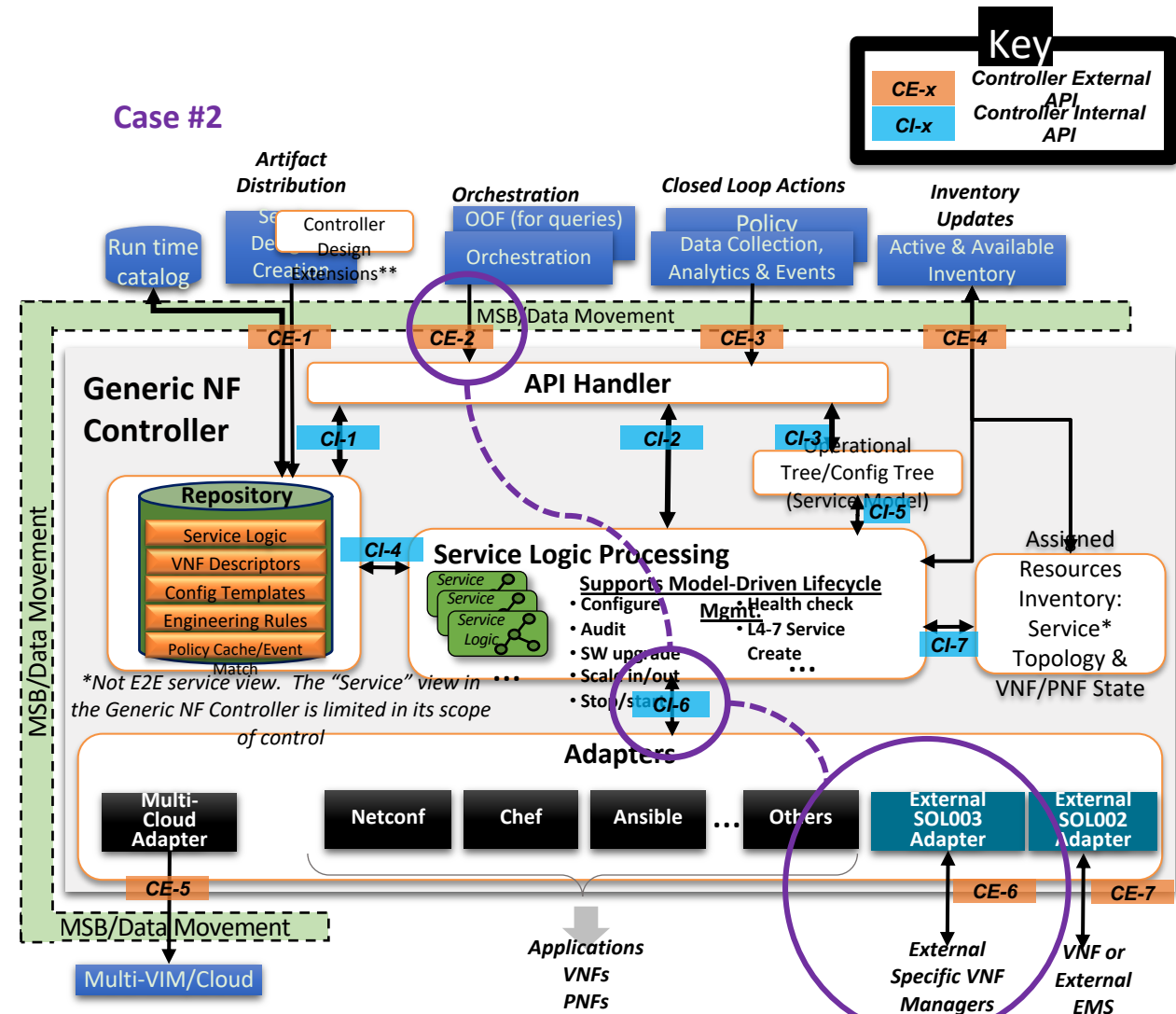
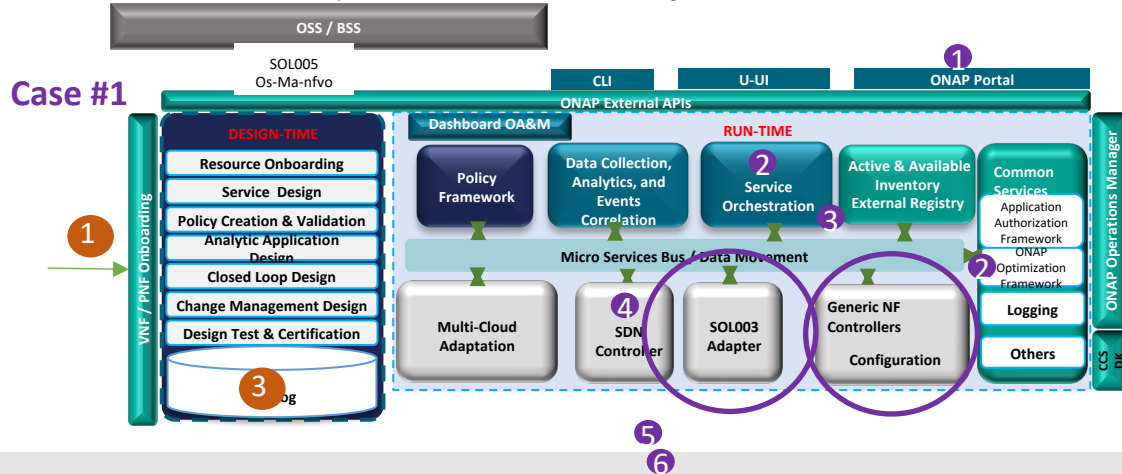
!----- Licenses
!----- file(s)

!----- Artifacts
!----- install.sh
!----- images
!----- templates
!----- start.yang
    
```

# ETSI SOL003 Adapter, Possible Placement Options



- One SOL003 Adapter or Two SOL003 Adapters?
  - Case #1: One SOL003 Adapter:** a proposal for GNFC has an external SOL003 Adapter, which supports create/instantiate/terminate/scale/heal VNF and LCN, and also configurations.
    - For this, the current SO VNF Manager Adapter needs to be decoupled from SO
    - The SOL003 Adapter supports all the SOL003 operations including configuration
  - Case #2: Two SOL003 Adapters:** a proposal that one SOL003 adapter which is attached to SO handles VNF LCM and LCN; another SOL003 adapter in the GNFC handles configurations - allowing separate evolutions of LCM and Configurations
- For the GNFC use case,
  - CE-2 NBI needs to be enhanced to support VNF LCM
  - Connections/Paths between CE-2 and CI-06 need to be defined/designed
  - SOL003 Adapter NBI needs to be adjusted, based on CI-6



# References

- SO Plug-in Support for VNFM (SO VNFM Adapter), <https://wiki.onap.org/pages/viewpage.action?pageId=48529911>
- SO VNFM Adapter APIs, <https://wiki.onap.org/display/DW/SO+VNFM+Adapter+APIs>
- SO VNFM Adapter Test Case, <https://wiki.onap.org/display/DW/SO+VNFM+Adapter+Test+Case>
- SO VNFM Adapter Feature Candidates for Frankfurt, <https://wiki.onap.org/display/DW/SO+VNFM+Adapter+Feature+Candidates+for+Frankfurt>
- SO ETSI Catalog DB handling for NS and VNF/PNF packages, <https://wiki.onap.org/pages/viewpage.action?pageId=63996543>



**ONAP**

OPEN NETWORK AUTOMATION PLATFORM

Thank you!