

ETSI-Alignment Architecture - Honolulu

- 1- Brief Project Overview (brief as it should be known)
 - 1.1 Overall Architecture
 - 1.2 ONAP SO CNF Orchestration Support
- 2- New component capabilities for Honolulu, i.e. the functional enhancements, if applicable
- 3- New or modified interfaces
- 4- If they are modified, are they backwards compatible?
- 6- Interface naming (point to an example)
- 7- Consumed API from other projects
- 8- Published API
- 10- What are the system limits?
- 11- Involved use cases, architectural capabilities or functional requirements.
- 12- Listing of new or impacted models used by the project (for information only).
- 13- Test plan/Testing Strategy
- 14- Any other details that are specific to this functional enhancement or UseCase.

1- Brief Project Overview (brief as it should be known)

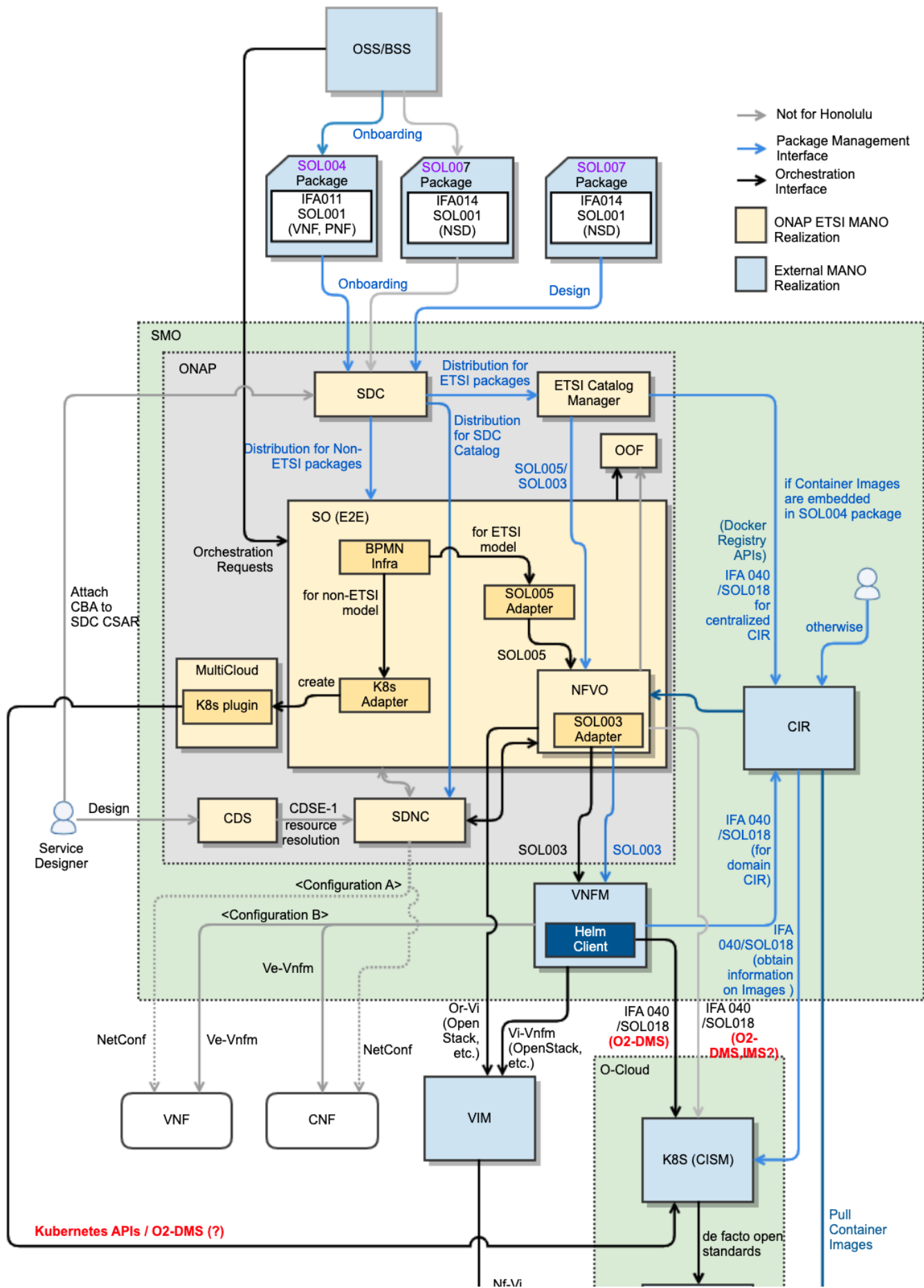
The use case ([REQ-400 - Getting issue details...](#) **STATUS** , [ONAPARC-648 - Getting issue details...](#) **STATUS**) intends to demonstrate the ETSI-Aligned modeling, package distribution, and orchestration by conforming to ETSI NFV. This use case shall support:

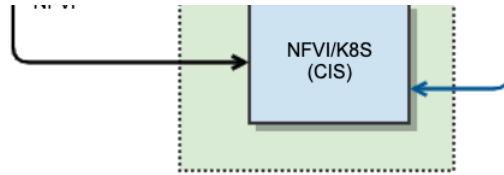
1. VNF modeling with minimum CNF enhancements from ETSI NFV 4.1.1, by conforming to the ETSI SOL004 Package and SOL001 Modeling standards
2. automated NSD and VNF/CNF package E2E distribution, by conforming to SOL005 and SOL003 Package Management APIs
3. NS and VNF Orchestration through a modular architecture and standards-based (ETSI NFV) interfaces, by conforming to SOL005 and SOL003 LCM standards

1.1 Overall Architecture

The following diagram depicts overall ETSI-Alignment Architecture for Honolulu.

- SDC supports SOL004 Package onboarding
- SDC supports SOL007 Design
 - SOL007 Onboarding is not yet supported in Honolulu
- ETSI Catalog Manager receives SDC Package notification
- ETSI Catalog Manager gets the original vendor ETSI NS and VNF packages and stores them into its database
- ETSI Catalog Manager sends the package onboarding notification to its subscribers (SO NFVO, SOL003 Adapter, in the future SOL005 Adapter)
- If the SOL004 VNF package embeds Container Images, ETSI Catalog Manager pushes the images to CIR via Docker Registry APIs
 - Currently, ETSI Catalog Manager defined this Container Image handling as a stretch goal
 - If not (due to SDC 2MB Image file size limitation), ETSI Catalog Manager does not handle Container Images. In that case, the operator will upload Container Image(s) to the CIR component directly
 - In this case, SOL004 Package metadata will reference Container Image file(s)
- CIR will be realized by Nexus with Docker Registry (GitHub is under consideration)
 - CIR component can be delivered part of ONAP or the operator can leverage their CIR component(s)
 - Multiple CIR instances can be used
 - Registration CIR into AAI for ETSI Catalog Manager to find a proper instance of CIR
 - Access APIs to CIR will be Docker Registry APIs, by conforming to ETSI IFA040/SOL018
 - K8S CISM and CIS need to have their own way to find a proper CIR instance to pull Container Images.
- K8S CISM will obtain information on Container Image(s) thru Docker Registry APIs
- K8S CIS will pull Container Image(s) thru Docker Registry APIs
- The Interface between VNFM and K8S CISM will follow K8S APIs
 - Interface between VNFM and CISM: use k8s APIs, potential to leverage an alignment with O2-DMS
 - Interface between MultiCloud K8s Plugin and CISM: use k8s APIs, potential to leverage alignment with O2-DMS
 - VNFM Enhancement (PoC) to invoke CISM on O-Cloud through O2-DMS
 - Note: support O2-DMS is a stretch goal; most likely use K8S APIs
- Architecture wiki page: [ETSI-Aligned CNF Support - Honolulu](#)





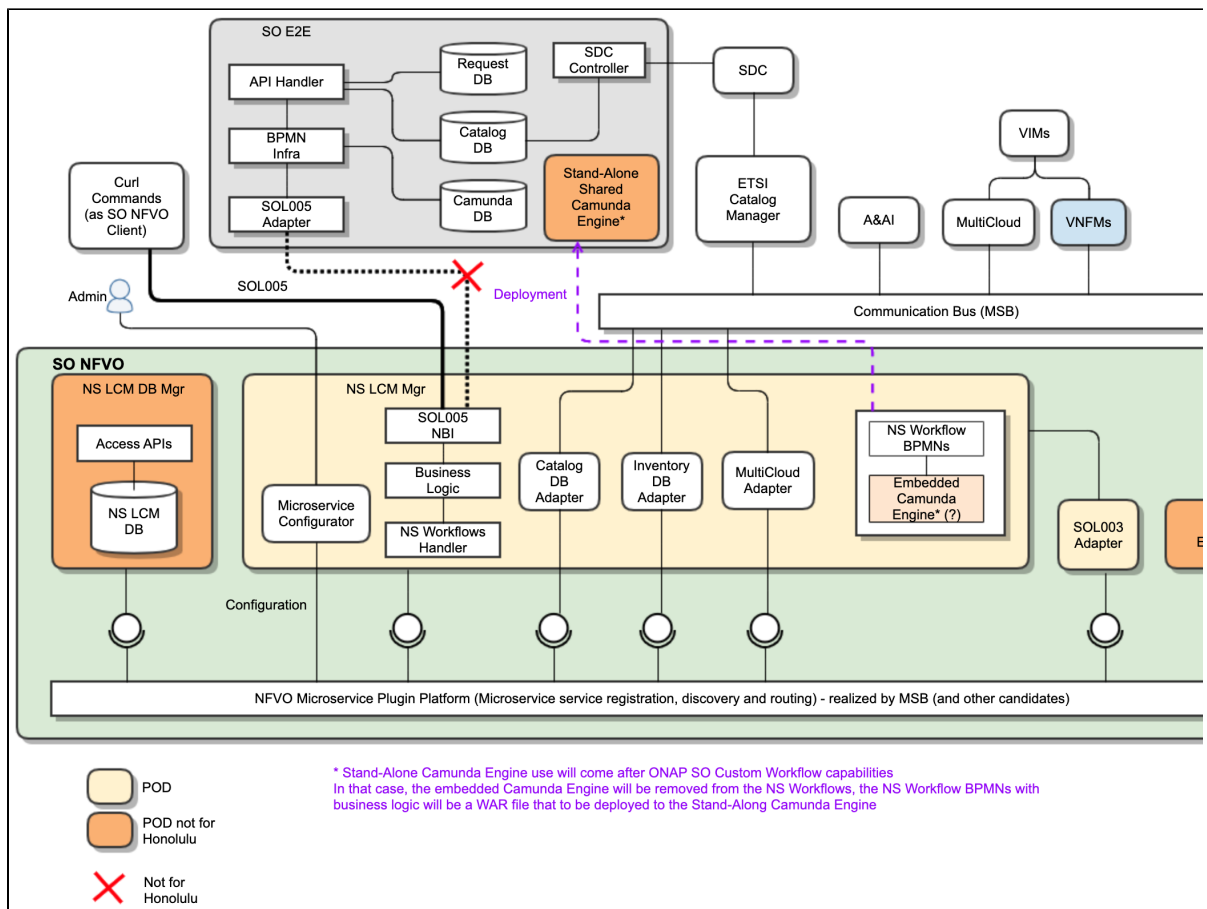
1.2 ONAP SO CNF Orchestration Support

There are two tracks in ONAP for CNF support: ONAP ETSI-Alignment and Direct Path for CNF (Based on models, SO selects a proper path):

- **ETSI-Alignment path**
 - Following ETSI NFV IFA specifications for CNF
 - Leveraging and extending existing ONAP ETSI-Alignment Architecture
- **Direct Path**
 - Some think there is no need for the full VNFD for LCM, just Helm Charts and Images; delegate LCM to CISM; it may work for simple CNFs
 - Note: there are several cases where a more complete VNFD will be needed, such as 5G Core NFs.
- **Collaboration Areas:**
 - Unified Modeling, Packaging & Onboarding (SDC) – optional VNFD?
 - Package Distribution (SDC – ETSI Catalog Manager – CIR)
 - Common Components (e.g., CIR, A&AI, CDS)
- **Note: CIR will be realized by Nexus with Docker Registry (possibly migrate it to Github), and CIR access APIs will be based on Docker Registry APIs.**

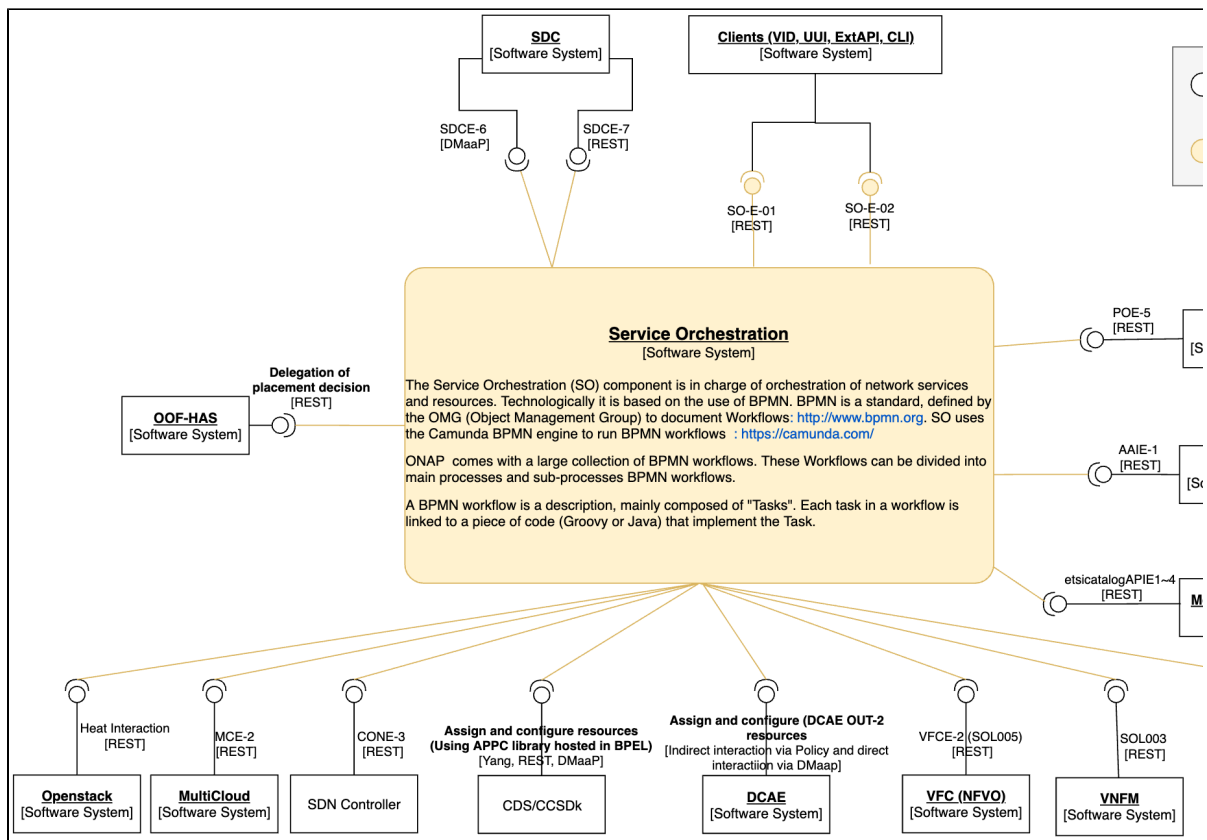
2- New component capabilities for Honolulu, i.e. the functional enhancements, if applicable

- **SDC**
 - Enhancements related to ETSI 3.3.1 SOL004 compliant VNF / CNF package onboarding and distribution
 - [ETSI Package Management \(SDC Enhancements\) - Honolulu](#)
 - [ETSI Package Management \(SDC Enhancements\) - Honolulu#ETSIPackageManagementArchitecture](#)
 - Supports ETSI 3.3.1 VNF with minimum CNF enhancements from 4.1.1
 - [VNF/CNF Data Model Based on ETSI v3.3.1 SOL001 plus CNF Enhancements](#)
 - [CSAR Models: ETSI-Alignment VNFD and Example\(s\)](#)
 - Enhancements related to ETSI 3.3.1 SOL007 compliant NSD package Design and Distribution
 - [ETSI Package Management \(SDC Enhancements\) - Honolulu#NSOnboardingDesign](#): Design SOL007
 - [ETSI Package Management \(SDC Enhancements\) - Honolulu#DesignNS](#): Design NS
 - Enhancements related to ETSI 3.3.1 SOL004 VNF and SOL007 NS package security
 - [ETSI Package Management \(SDC Enhancements\) - Honolulu#SDCSOL004/SOL007VNFPackageSecurity](#)
 - Architecture Diagram for ETSI package handling and distribution



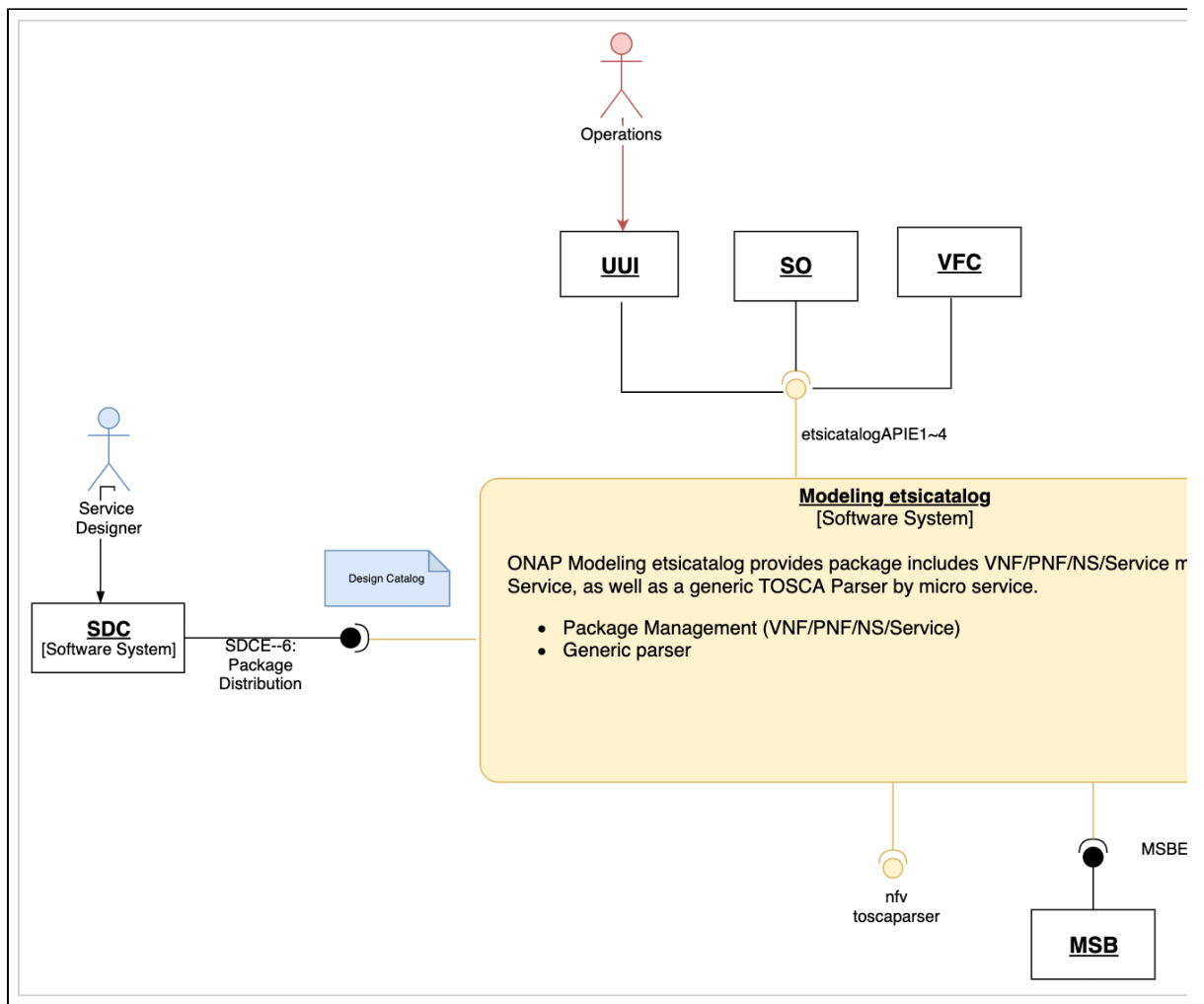
- SO Interface Update

- The following diagram depicts SO interfaces. The interactions with ETSI Catalog Manager (a.k.a. Modeling etsicalog) are added
- See the link for details, [ARC Service Orchestrator Component Description - Honolulu-R8](#)

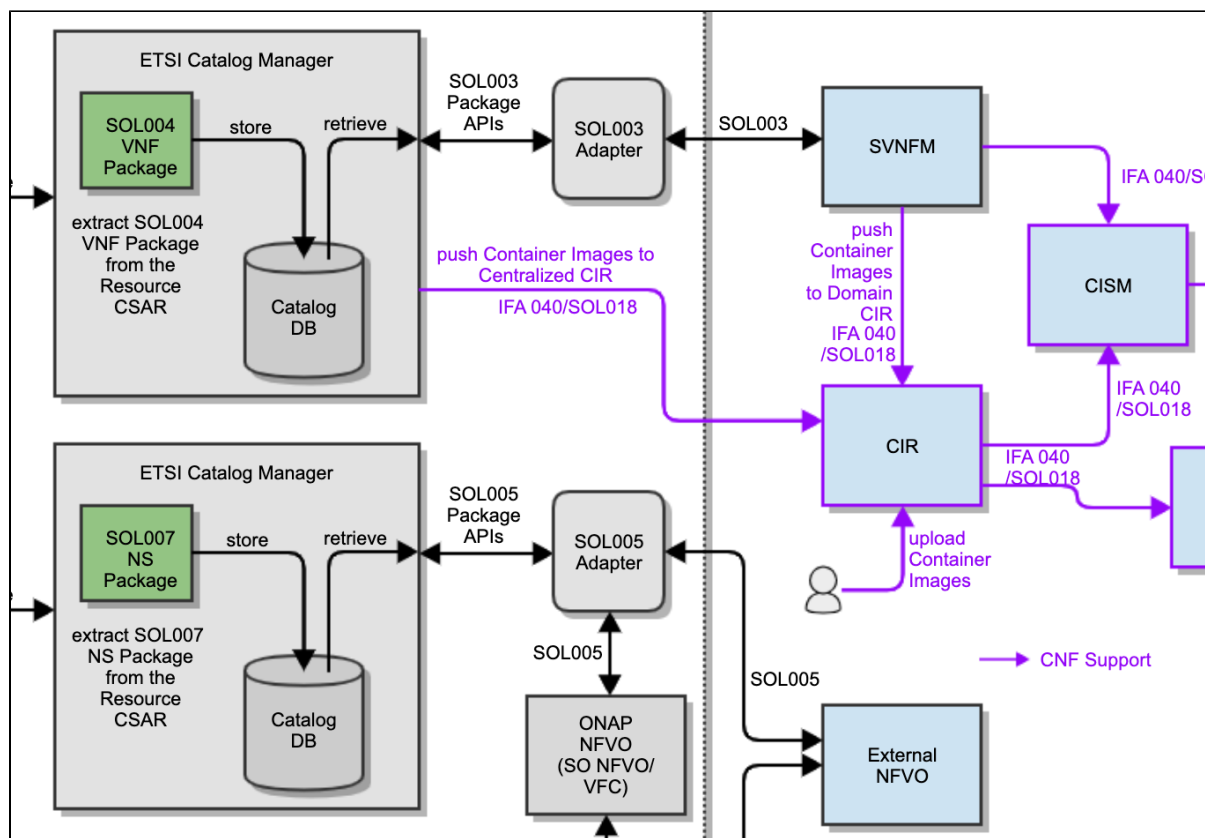


- **ETSI Catalog Manager**

- Enhancements related to direct SDC Package distribution support thru SDCE-6
- Enhancements related to NSD Subscription and Notification
- Architectural Relationships and Interfaces support: [ARC Modeling Component Description – Honolulu-R8](#)
- Architecture Diagram:



- Enhancements related to Container Image handling is a stretch goal
 - Most likely due to SDC 2MB file size limitation, Container Images will be uploaded to the CIR component by the Admin
 - i.e., in that case, no impact on ETSI Catalog Manager in Honolulu for Container Image handling



- **AAI**
 - No impacts since the SO NFVO does not support Scaling or Healing
 - AAI impacts will be analyzed for the Istanbul release

3- New or modified interfaces

- SDC ETSI Catalog Manager
- ETSI Catalog Manager SO NFVO

4- If they are modified, are they backwards compatible?

- SDC Onboarding supports one-level backward compatibility.
 - For Honolulu release, SDC supports ETSI v3.3.1 and v2.7.1/2.6.1
- SDC data model does not have the version support, the SDC data models will be super-set data structures to hold both versions
 - SDC data model version support would be an Istanbul release consideration

6- Interface naming (point to an example)

7- Consumed API from other projects

Project	API Dependency	Notes	API Specs (Swagger)
SO / SO NFVO	ETSI Catalog Manager Package Management APIs for NS	ETSI Catalog Manager APIs	https://wiki.onap.org/download/attachments/79200593/etsicatalog_API_v1.yaml?version=1&modificationDate=1579574658000&api=v2

SOL003 Adapter	ETSI Catalog Manager Package Management APIs for VNF	ETSI Catalog Manager APIs	https://wiki.onap.org/download/attachments/79200593/etsicatalog_API_v1.yaml?version=1&modificationDate=1579574658000&api=v2

8- Published API

Project	API	Notes	API Specs (Swagger)
SOL003 Adapter	ETSI SOL003 Package Management & LCM APIs	Support APIs for VNFM	https://forge.etsi.org/jenkins/view/All%20jobs/job/NFV%20-%20Network%20Functions%20Virtualisation/job/sol002-sol003-master/lastSuccessfulBuild/artifact/build/

9- Reference to the interfaces.

(Reference to the the swagger.json file(s) whenever possible)

10- What are the system limits?

.

11- Involved use cases, architectural capabilities or functional requirements.

12- Listing of new or impacted models used by the project (for information only).

- Identify any High Level Information Model Requirements. See: ONAP R8 Modeling High Level Requirements
 - Models based on information exchanges from Use Cases
 - Models documenting existing implementations
 - Forward looking models that may be implemented in future releases
- Describe how exposed APIs are mapped to information models

(list all the relevant Jira tickets)

13-Test plan/Testing Strategy

- Unit Testing
- Dev-to-Dev Testing and
- Integration

14- Any other details that are specific to this functional enhancement or UseCase.

