

CLAMP R6 - M1 Release Planning

- 1 [Overview](#)
- 2 [Scope](#)
 - 2.1 [What is this release trying to address?](#)
 - 2.2 [Use Cases](#)
 - 2.3 [Minimum Viable Product](#)
 - 2.4 [Functionalities](#)
 - 2.4.1 [Epics](#)
 - 2.4.2 [Stories](#)
 - 2.5 [Longer term Roadmap](#)
- 3 [Release Deliverables](#)
- 4 [Sub-Components](#)
- 5 [Architecture](#)
 - 5.1 [High level architecture diagram](#)
 - 5.1.1 [Architecture](#)
 - 5.2 [Platform Maturity](#)
 - 5.3 [API Incoming Dependencies](#)
 - 5.4 [API Outgoing Dependencies](#)
 - 5.5 [Third Party Products Dependencies](#)
- 6 [Testing and Integration Plans](#)
- 7 [Gaps](#)
- 8 [Known Defects and Issues](#)
- 9 [Risks](#)
- 10 [Resources](#)
- 11 [Release Milestone](#)
- 12 [Team Internal Milestone](#)
- 13 [Documentation, Training](#)
- 14 [Other Information](#)
 - 14.1 [Vendor Neutral](#)
 - 14.2 [Free and Open Source Software](#)
- 15 [Charter Compliance](#)
- 16 [Release Key Facts](#)

Overview

Project Name	Enter the name of the project
Target Release Name	Frankfurt Release
Project Lifecycle State	Incubation.(Refer to ONAP Charter, section 3.3 Project Lifecycle for further information)
Participating Company	AT&T, Ericcson, Huawei, (TBC)

Scope

What is this release trying to address?

CLAMP want to enhance the flexibility and ease of introducing new Control loop by:

1. implemnting a new Control Loop creation flow: Self Serve Control Loop .
2. Add Tosca policy-model support for Operationnal Policies definitions.
3. Add integration to CDS for Actor/Action selection.
4. Move from SearchGuard to OpenDistro
5. Document current upgrade component strategy (TSC must havve)
6. SECCOM Perform Software Composition Analysis - Vulnerability tables (TSC must have)
7. SECCOM Password removal from OOM HELM charts (TSC must have)
8. SECCOM HTTPS communication vs. HTTP (TSC must have) - CLAMP has already moved to to HTTPS so no work is required here

Scope	Priority	Committer Lead	Resources Committed	Epic	Dependencies
-------	----------	----------------	---------------------	------	--------------

CLAMP Self Serve Control Loop	high	Gervais-Martial Ngueko	AT&T, Ericsson	CLAMP-514 - Getting issue details... STATUS	DCAE
CLAMP use policy model for Ops Policy	high	Sébastien Determe	AT&T, Ericsson	CLAMP-492 - Getting issue details... STATUS	Policy
CLAMP - CDS integration	high	Vidyashree Rama	AT&T, Huawei	CLAMP-490 - Getting issue details... STATUS	CDS
SearchGuard to OpenDistro move	high	Gervais-Martial Ngueko	AT&T	CLAMP-483 - Getting issue details... STATUS	
Document current upgrade component strategy	high	Gervais-Martial Ngueko	AT&T	CLAMP-546 - Getting issue details... STATUS	
SECCOM Password removal from OOM HELM charts	high	Gervais-Martial Ngueko	AT&T	CLAMP-547 - Getting issue details... STATUS	
SECCOM Perform Software Composition Analysis - Vulnerability tables	high	Gervais-Martial Ngueko	AT&T		

Use Cases

The existing use cases are still going to be supported and additional use cases will be supported for the Frankfurt Release (as defined by the Control loop sub committee and TSC)

Minimum Viable Product

The minimum viable product that we aim to reach within R6 is to have the CLAMP application El Alto(R5) features at least running with, the new Operational policy-model and the new additional self-serve Control Loop flow working.

Functionalities

List the functionalities that this release is committing to deliver by providing a link to JIRA Epics and Stories. In the JIRA Priority field, specify the priority (either High, Medium, Low). The priority will be used in case de-scoping is required. Don't assign High priority to all functionalities.










Epics

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
CLAMP-837	Release Candidate 2 Integration and Test		Apr 21, 2020	May 27, 2020	May 14, 2020	Unassigned	None	==	CLOSED	Done
CLAMP-834	Release Candidate 1 Integration and Test		Apr 21, 2020	May 05, 2020	Apr 30, 2020	Unassigned	None	==	CLOSED	Done
CLAMP-657	Release Candidate 0 Integration and Test		Mar 01, 2020	Apr 03, 2020	Mar 26, 2020	Unassigned	None	==	CLOSED	Done
CLAMP-631	Code Freeze		Jan 31, 2020	Mar 04, 2020	Mar 05, 2020	Unassigned	None	==	CLOSED	Done
CLAMP-623	SECCOM or related work		Jan 21, 2020	Apr 27, 2020		Unassigned	None	==	CLOSED	Done
CLAMP-556	Functionality and API Freeze		Nov 22, 2019	Jan 20, 2020	Jan 21, 2020	Unassigned	None	==	CLOSED	Done

CLAMP-520	Frankfurt release planning milestone		Oct 04, 2019	Dec 19, 2019	Oct 11, 2019	Unassigned	None	=	CLOSED	Done
CLAMP-514	support self serve control Loop for Frankfurt release		Oct 02, 2019	Jul 02, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-492	CLAMP to support any policy type without additional coding		Sep 10, 2019	Jun 26, 2020		Unassigned	None	^	CLOSED	Done
CLAMP-490	get list of action for a given closed loop from CDS		Sep 10, 2019	Mar 31, 2020		Unassigned	None	^	CLOSED	Done
10 issues										

Stories

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
CLAMP-885	CLAMP update documentation		Jun 29, 2020	Jul 01, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-655	CLAMP remove csit not working		Feb 27, 2020	Mar 04, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-643	CLAMP (&dashboard) jenkins job update		Feb 05, 2020	Feb 27, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-625	CLAMP containers configured per secure recommendation		Jan 21, 2020	Apr 06, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-624	CLAMP code coverage Frankfurt		Jan 21, 2020	Mar 04, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-620	[automatically generated] Outdated dependency commons-codec		Jan 21, 2020	Mar 31, 2020		Unassigned	None	=	CLOSED	Not a Bug
CLAMP-616	[automatically generated] Outdated dependency joda-time		Jan 21, 2020	Mar 31, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-611	[automatically generated] Outdated dependency org.springframework		Jan 21, 2020	Mar 31, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-610	[automatically generated] Outdated dependency org.springframework.boot		Jan 21, 2020	Mar 31, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-606	[automatically generated] Outdated dependency org.apache.tomcat		Jan 21, 2020	Mar 31, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-605	[automatically generated] Outdated dependency org.apache.tomcat.embed		Jan 21, 2020	Mar 31, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-586	Modify NPM package build for ReactJS component library		Dec 19, 2019	Feb 27, 2020	Mar 05, 2020	Unassigned	None	=	CLOSED	Done
CLAMP-581	CLAMP user force refresh json schema for Ops Policy		Dec 10, 2019	Jan 16, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-580	CLAMP Metadata support in Policy-model Tosca		Dec 10, 2019	Mar 02, 2020		Unassigned	None	=	CLOSED	Done
CLAMP-577	Update the policy api that clamp uses to get the policy deployment states		Dec 05, 2019	Jan 16, 2020		Unassigned	None	=	CLOSED	Done

CLAMP-554	Ability to create control loops based on a template from the CLAMP UI		Nov 20, 2019	Mar 04, 2020	Unassigned	None	=	CLOSED	Done
CLAMP-547	CLAMP password removal from OOM helm chart		Nov 06, 2019	Jun 26, 2020	Unassigned	None	^	CLOSED	Done
CLAMP-546	CLAMP Frankfurt document upgrdae procedure		Nov 06, 2019	Jul 02, 2020	Unassigned	None	^	CLOSED	Done
CLAMP-545	Create Service object to store service related info		Nov 04, 2019	Nov 28, 2019	Unassigned	None	=	CLOSED	Done
CLAMP-543	Clamp ability to Upload the Blueprints and Policy Tosca models manually		Oct 28, 2019	Nov 28, 2019	Unassigned	None	=	CLOSED	Done
CLAMP-531	CLAMP integrate new oParent into Build process		Oct 07, 2019	Nov 14, 2019	Unassigned	None	=	CLOSED	Done
CLAMP-518	CLAMP query Policy to get policy-model of mS		Oct 02, 2019	Feb 17, 2020	Unassigned	None	^	CLOSED	Done
CLAMP-517	CLAMP Support new Control Onboarding flow		Oct 02, 2019	Mar 02, 2020	Unassigned	None	=	CLOSED	Done
CLAMP-516	loop through list of CL flow microservice to instantiate		Oct 02, 2019	Mar 02, 2020	Unassigned	None	^	CLOSED	Done
CLAMP-509	Improve test coverage		Sep 19, 2019	Nov 04, 2019	Unassigned	None	v	CLOSED	Done

Showing 25 out of 34 issues

Longer term Roadmap

Indicate at a high level the longer term roadmap. This is to put things into the big perspective.

The long term goal is to reach a common platform for managing control loops within ONAP :

CLAMP is a platform for designing and managing control loops. It is used to setup a control loop, configure it with specific parameters for a particular network service, then deploying and undeploying it. Once deployed, the user can also update the loop with new parameters during runtime, as well as suspending and restarting it.

It interacts with other systems to deploy and execute the closed loop. For example, it receives CSAR service package from SDC, associating it with the VF resource. It requests from DCAE the instantiation of microservices to manage the closed loop flow. Further, it creates and updates multiple policies in the Policy Engine that define the control loop flow.

The ONAP CLAMP platform abstracts the details of these systems under the concept of a control loop model. The setup of a control loop and its management is represented by a workflow in which all relevant system interactions take place. This is essential for a self-service model of creating and managing control loops, where no low-level user interaction with other components is required.

At a higher level, CLAMP is about supporting and managing the broad operational life cycle of VNFs/VMs and ultimately ONAP components itself. It will offer the ability to configure, test, deploy and update control loop automation - both closed and open. Automating these functions would represent a significant saving on operational costs compared to traditional methods.

Another Key long term goal is to provide a better user experience by having more flexibility to add mico-service without code development.

A Dashboard has been introduced to allow the user to get a quick overview of the status and events of running control loops.

Release Deliverables

Indicate the outcome (Executable, Source Code, Library, API description, Tool, Documentation, Release Note...) of this release.

Deliverable Name	Deliverable Description	Deliverable location
CLAMP Docker container	Docker images available on nexus3	Nexus3 docker registry

Source Code	Code of the Designer and run time of CLAMP	CLAMP git repository
Deployment scripts	Scripts that can be used to help with the container instantiation and configuration	CLAMP git repository
Property Files	Properties files that can be used to tune the configuration of CLAMP depending on the environment	CLAMP git repository
Documentation	Release specific documentation (Release Note, user guide, deployment guide) provided through readthedocs	CLAMP readthedoc ONAP section : https://docs.onap.org/en/latest/index.html

Sub-Components

....

Architecture

High level architecture diagram

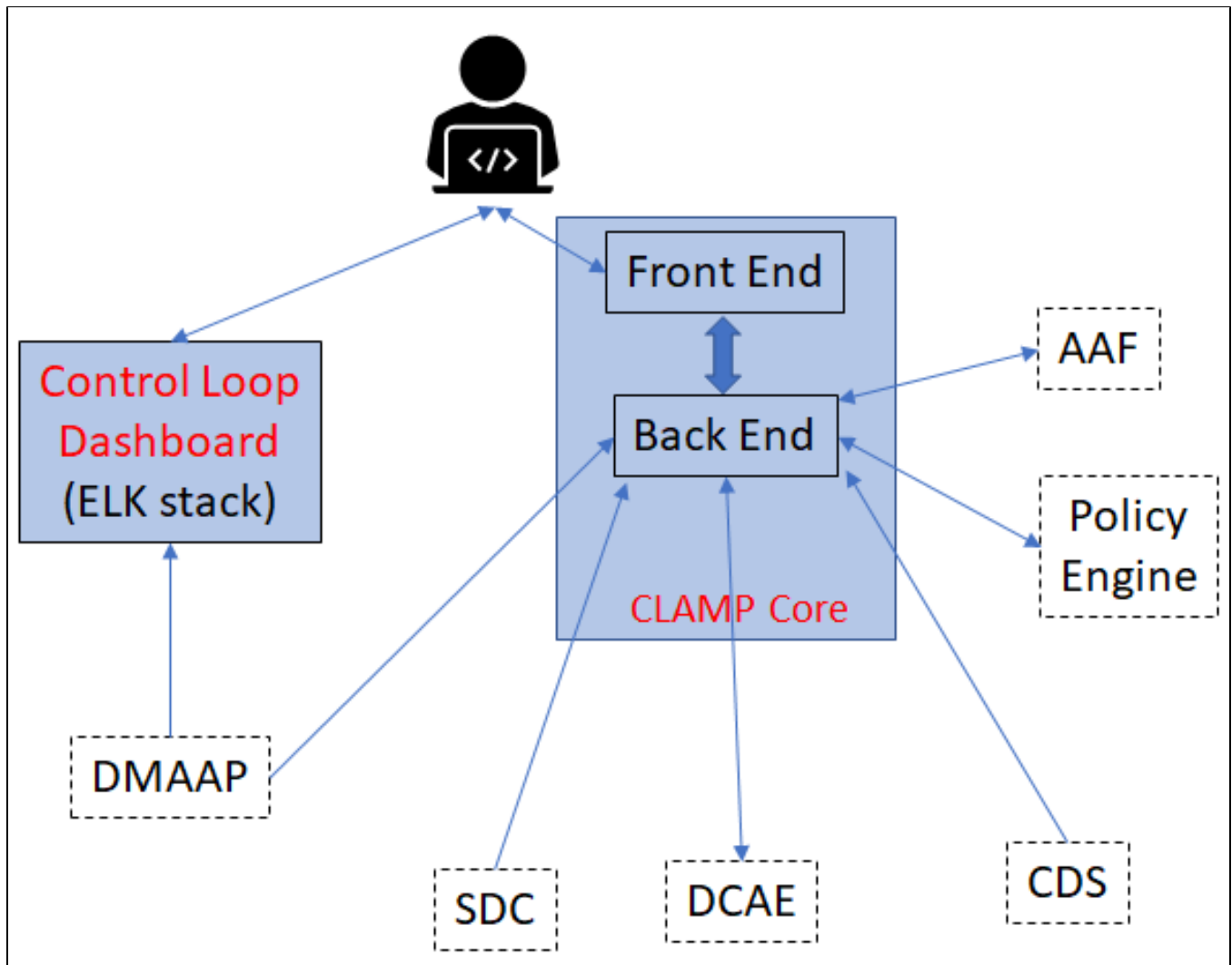
At that stage within the Release, the team is expected to provide more Architecture details describing how the functional modules are interacting.

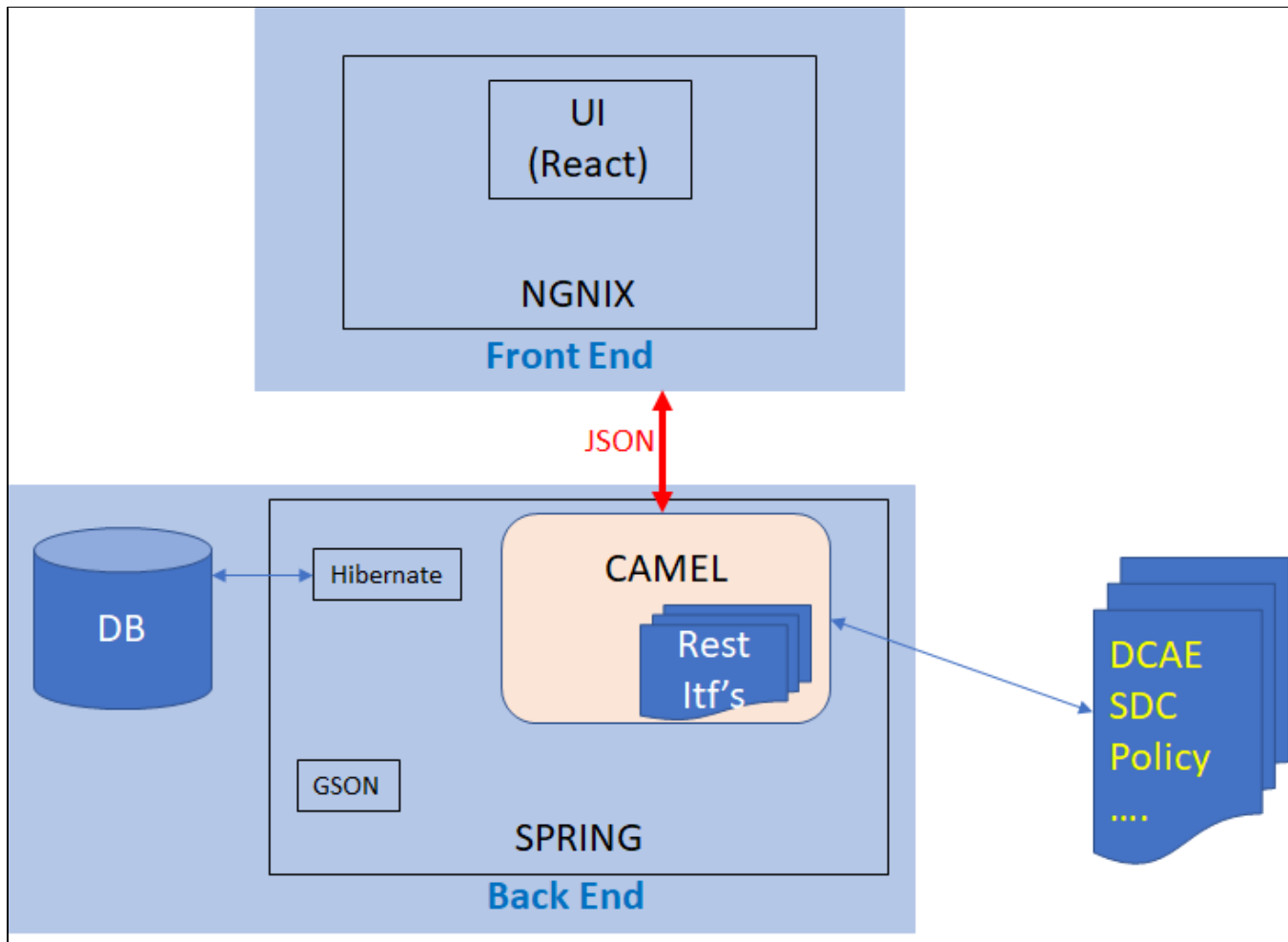
Block and sequence diagrams showing relation within the project as well as relation with external components are expected.

Anyone reading this section should have a good understanding of all the interacting modules.

Architecture

- Below we show how the CLAMP application fits into ONAP. The red figure below shows the CLAMP application components. There is a design portion and an operations component.





•**CLAMP** is separated in 3 areas, which are currently (in seed code) both supported by a single application:

1. **Design/Setup Time**(Cockpit/UI to Configure the received templates)
 - a. SDC will distribute a CSAR, for a service, the part of the CSAR that CLAMP will use are:
 - i. the Control Loop flow Templates(e.g: blueprint) are defined in DCAE-D(sub-component of SDC) and distributed to CLAMP by SDC. The templates format is **TOSCA**. **The blueprint is also** pushed, by **SDC**, to **DCAE** platform orchestration engine.
 - ii. The policy-models defining the DCAE μ S used inside the blueprint. note that policy-engine will also receive this SDC distribution and so should be also aware of those policy-models.
 - b. policies (configuration and operational policies) are pushed/provisioned towards the Policy Component of ONAP. (those policies will be triggered by **DCAE** during Closed Loop operations).
 - i. The **DCAE** team needs to provide models to **Policy** team in order for the Configuration policy to be built.
 - c. DCAE (in Frankfurt release), plan to introduce the DCAE-DESIGNER to(this is part of the new Self Serve Control Loop initiative):
 - i. design control loop flow (sequences of μ S)
 - ii. expose an API for CLAMP to retrieve the list of existing flow, so that CLAMP can use it to create new Control Loop's
 - d. CDS (in Frankfurt release):
 - i. expose by an API (or through CSAR distribution via SDC), the list of actors and corresponding actions, that CLAMP can then configure as part of an Operational policy configuration
2. **Run time**(**DCAE-Policy**, grabbing events and triggering policies based actions)
 - a. the triggering to deploy(and then effectively start the closed loop) a blueprint will be manual (via CLAMP cockpit) an automatic deployment based on an event will come in future release.
 - b. The CLAMP cockpit will support the following action at runtime:
 - i. start (start the provisioned Closed Loop on **DCAE**)
 - ii. stop (stop a provisioned Closed loop on **DCAE**)
3. **Dashboard** (ELK based)

API Name	API Description	API Definition Date	API Delivery date	API Definition link (i.e. swagger)
Same as previous release	SDC Client(jar library provided by SDC team) used to get service template (describing control loop flow) and blueprint id(to know which blueprint has been distributed to DCAE for this Control Loop template)		Already available	
Same as previous release	API exposed by Policy to create/update guard policies (used for scale out use case operational policies)		Already available	
Same as previous release	API exposed by Policy to create/update policies		Already available	
	API exposed by policy to get a policy-model given the policy-model-id	ongoing	TBD	
	API exposed by Policy to create polidy pdp group	ongoing	TBD	
Same as previous release	API exposed by DCAE to start/stop a Closed Loop		Already available	
Same as previous release	API exposed by DCAE to trigger the deployment/undeployment of a Control Loop template		Already available	
Same as previous release	API exposed by DCAE to get status of a CLAMP deployed μS		Already available	
	API exposed by DCAE to get status of all μS	ongoing	TBD	
	API exposed by DCAE to get the list of Control Loop Flow available to use by CLAMP	ongoing	TBD	

	API exposed by CDS to retrieve CDS actors/actions	ongoing	TBD	
--	---	---------	-----	--

API Outgoing Dependencies

API this release of CLAMP is delivering to other ONAP Component(s) releases.

API Name	API Description	API Definition Date	API Delivery date	API Definition link (i.e.swagger)
N/A				

Third Party Products Dependencies

Third Party Products mean products that are mandatory to provide services for your components. Development of new functionality in third party product may or not be expected.

List the Third Party Products (OpenStack, ODL, RabbitMQ, Elasticsearch, Crystal Reports, ...).

Name	Description	Version
Camel	framework to define routing and mediation rules	2.24.0
Docker	Container engine	1.17
MariaDB	database container	10.3.12
Spring boot	Spring boot Framework dependencies	2.1.5

In case there are specific dependencies (Centos 7 vs Ubuntu 16. Etc.) list them as well.

Testing and Integration Plans

Provide a description of the testing activities (unit test, functional test, automation,...) that will be performed by the team within the scope of this release.

Describe the plan to integrate and test the release deliverables within the overall ONAP system.

Confirm that resources have been allocated to perform such activities.

CLAMP will invest in CSIT tests to allow further integration testing, CLAMP already provided some tests as part of previous releases.

Gaps

This section is used to document a limitation on a functionality or platform support. We are currently aware of this limitation and it will be delivered in a future Release.

List identified release gaps (if any), and its impact.

Gaps identified	Impact
Testing/Integration	limited testing of final product

Known Defects and Issues

Please refer to [Frankfurt Defect Status](#)

Risks

List the risks identified for this release along with the plan to prevent the risk to occur (mitigation) and the plan of action in the case the risk would materialized (contingency).

Please update any risk on the centralized wiki page - [Frankfurt Risks](#)

Resources

[Link toward the Resources Committed to the Release](#) centralized page.

Release Milestone

The milestones are defined at the [Release Level](#) and all the supporting project agreed to comply with these dates.

Team Internal Milestone

This section may be used to document internal milestones that the team agreed on.

Also, in the case the team has made agreement with other team to deliver some artifacts on a certain date that are not in the release milestone, provide these agreements and dates in this section.

It is not expected to have a detailed project plan.

Date	Project	Deliverable
To fill out	sdc	sdc UI/UX SDK

Documentation, Training

- Highlight the team contributions to the specific document related to the project (Config guide, installation guide...).
- Highlight the team contributions to the overall Release Documentation and training asset
- High level list of documentation, training and tutorials necessary to understand the release capabilities, configuration and operation.
- Documentation includes items such as:
 - Installation instructions
 - Configuration instructions
 - Developer guide
 - End User guide
 - Admin guide
 - ...



Note

The Documentation project will provide the Documentation Tool Chain to edit, configure, store and publish all Documentation asset.

Other Information

Vendor Neutral

If this project is coming from an existing proprietary codebase, ensure that all proprietary trademarks, logos, product names, etc. have been removed. All ONAP deliverables must comply with this rule and be agnostic of any proprietary symbols.

Free and Open Source Software

FOSS activities are critical to the delivery of the whole ONAP initiative. The information may not be fully available at Release Planning, however to avoid late refactoring, it is critical to accomplish this task as early as possible. List all third party Free and Open Source Software used within the release and provide License type (BSD, MIT, Apache, GNU GPL,...). In the case non Apache License are found inform immediately the TSC and the Release Manager and document your reasoning on why you believe we can use a non Apache version 2 license.

Each project must edit its project table available at [Project FOSS](#)

Charter Compliance

The project team comply with the [ONAP Charter](#).

Release Key Facts

Fill out and provide [a link toward the centralized Release Artifacts](#).

