Multi VIM/Cloud Release Planning for R2

DRAFT PROPOSAL FOR COMMENTS

The content of this template is expected to be fill out for M1 Release Planning Milestone.

> **Info**
>
> Use the "Copy" and "Move" options (available under the ..., top right of this page) to duplicate this template into your project wiki.
> Use the Wiki to document the release plan. Don't provide PowerPoint.
> Use as much diagrams and flow charts as you need, directly in the wiki, to convey your message.

- 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
  - 5.2 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

Overview

Project Name	Enter the name of the project
Multi VIM/Cloud	Beijing
Project Lifecycle State	Incubation
Participating Company	Amdocs, AT&T, ChinaMobile, Huawei, Intel, Lumina Networks, Microsoft, Mirantis, VMware, WindRiver, ZTE

Scope

What is this release trying to address?

Describe the problem being solved by this release

Provide a Cloud Mediation Layer supporting multiple Clouds.

1. Project framework alignment with ONAP R2 S3P goals

2. Transforming towards Model driven

3. Function enhancement as Multi VIM/Cloud mediation layer

Use Cases

Describe the use case this release is targeted for (better if reference to customer use case).

All use cases approved for R2

Minimum Viable Product

Describe the MVP for this release.

1 Event/Alert/Metrics federation

2 Elastic API exposure

3 Parallelism improvement of Multi Cloud Services

4 Logging Enabling

5 Image service

6 Clustering service interfaces for query and placement of a group of resources

7 HPA resource discovery and representation for HPA awareness orchestration

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.

Functionality Name	In or Out	Priority	Stretch
Clustering placement/query	IN	H	Query capacity and status for a group of resources
Message bus pub/sub	IN and OUT	H	Publish and subscribe events/notification
Resource Hierarchy graph	IN	H	Traverse resource Hierarchy in given Cloud
Image upload/download	IN	M	Support upload and download images

Integration with other project

Functionality Name	In or Out	Priority	Stretch
Logging Integration	In	M	Integrate with Logging
A&AI	In	M	Integrate with A&AI on Image Manager and Resource Modeling
OOF	In	H	Integrate with OOF
VFC	In	H	Integrate with VFC on VM status part
APPC	In	H	Integrate with APPC on VM status part
SO	In	H	Integrate with SO on parser adapter

Epics

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
MULTICLOUD-148	This epic will be used to contain all works about function features		Jan 16, 2018	Aug 12, 2023		Unassigned	None	=	CLOSED	Done
MULTICLOUD-147	This epic will be used to contain all works related to S3p improvement		Jan 16, 2018	Mar 04, 2019		Unassigned	None	=	CLOSED	Done

2 issues

Stories

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
MULTICLOUD-211	Add licenses for Multi VIM /Cloud Repo		Apr 11, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-207	Image API for multicloud plugin		Mar 29, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-206	Enhancement HPA for ocata		Mar 28, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-204	Add event/alert check api for ocata, newton and titanium cloud		Mar 27, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-203	Add UT token v2 for ocata, newton and titanium cloud plugin		Mar 26, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-200	Add HPA for Titanium Cloud plugin		Mar 23, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-197	Event/Alert/Metrics federation - ceilometer enabling		Mar 21, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-179	Update HPA enablement to support AAI Schema v13		Mar 07, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-178	Logging enhancement for MultiCloud OpenStack		Mar 07, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-177	Logging enhancement for MultiCloud OpenStack		Mar 07, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-176	Sprint Backlog for M3		Mar 07, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-175	Improve Code Coverage		Mar 07, 2018	Aug 12, 2023		Unassigned	None		CLOSED	Done
MULTICLOUD-174	Mark End-Of-Life to MultiCloud plugin service forKilo		Mar 07, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-164	Sprint Backlog for M2		Feb 22, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-162	Add Traits API starting in Ocata		Feb 06, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Won't Do
MULTICLOUD-159	Add UI and tutorial documentations		Jan 22, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-157	HPA resource discovery and representation for HPA awareness orchestration		Jan 16, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-156	Clustering service interfaces for query and placement of a group of resources		Jan 16, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done
MULTICLOUD-155	Image service		Jan 16, 2018	Aug 08, 2018		Unassigned	None		CLOSED	Done

MULTICLOUD-152	Elastic API exposure		Jan 16, 2018	Aug 12, 2023	Unassigned	None		CLOSED	Done

Showing 20 out of 26 issues

Longer term roadmap

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

This release we targeted to implement MVP features and finish regression of all existing use cases. The longer goal is to intergate with different dependent modules to consume the advanced features.

1 Standardized Infrastructure Resource Information Model (Towards model-driven)

2 FCAPS modeling (Towards model-driven)

3 TOSCA-based orchestration to deploy workload on multiple clouds (Towards model-driven)

4 MultiCloud support for Azure (Function enhancement)

Release Deliverables

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

Deliverable Name	Deliverable Description
Source Code	Source code for all Multi VIM/Cloud components
Docker Containers	Docker container associated componets
Documentation	Developer and user documentations

Sub-Components

List all sub-components part of this release.
Activities related to sub-components must be in sync with the overall release.

Sub-components are repositories and are consolidated in a single centralized place. Edit the Release Components name for your project in the centralized page.

Architecture

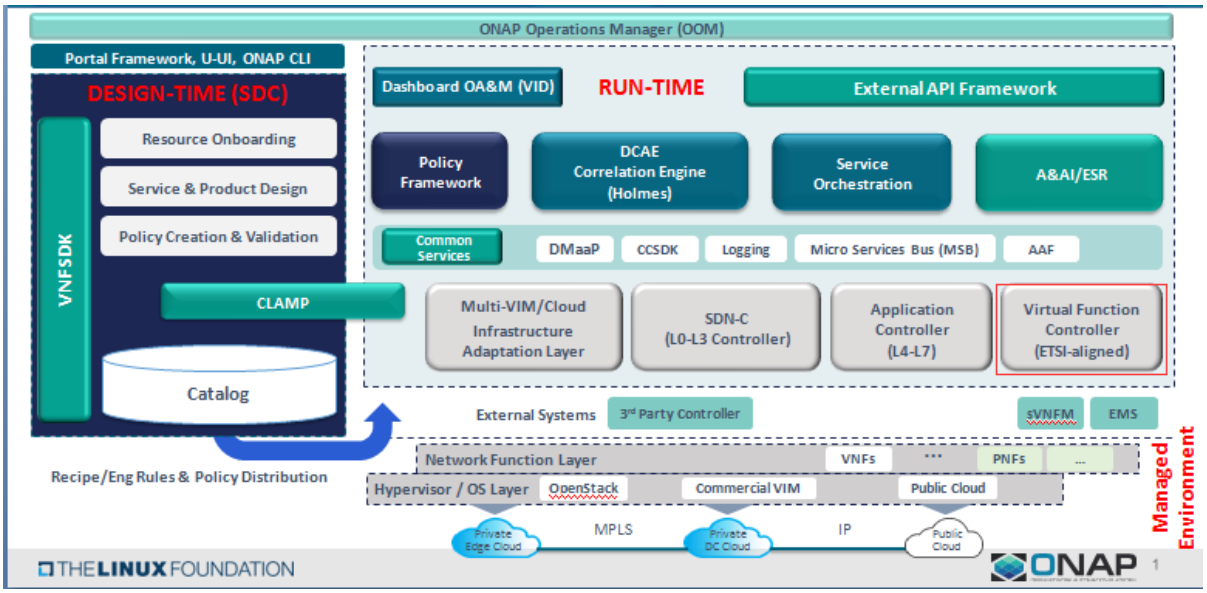
High level architecture diagram

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

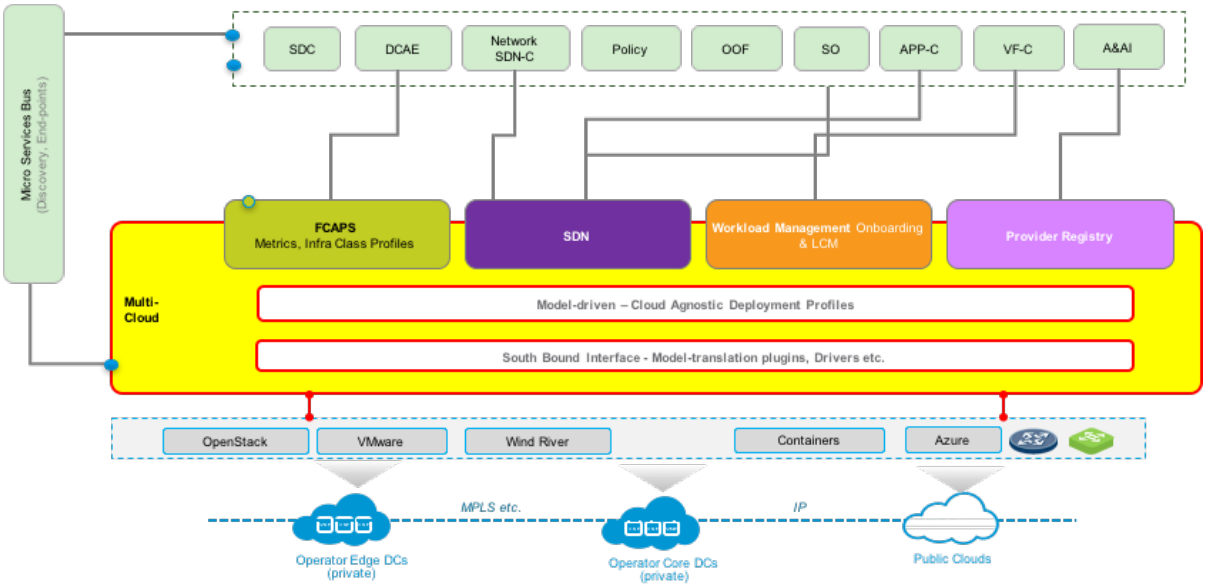
Indicate where your project fit within the ONAP Archiecture diagram.

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

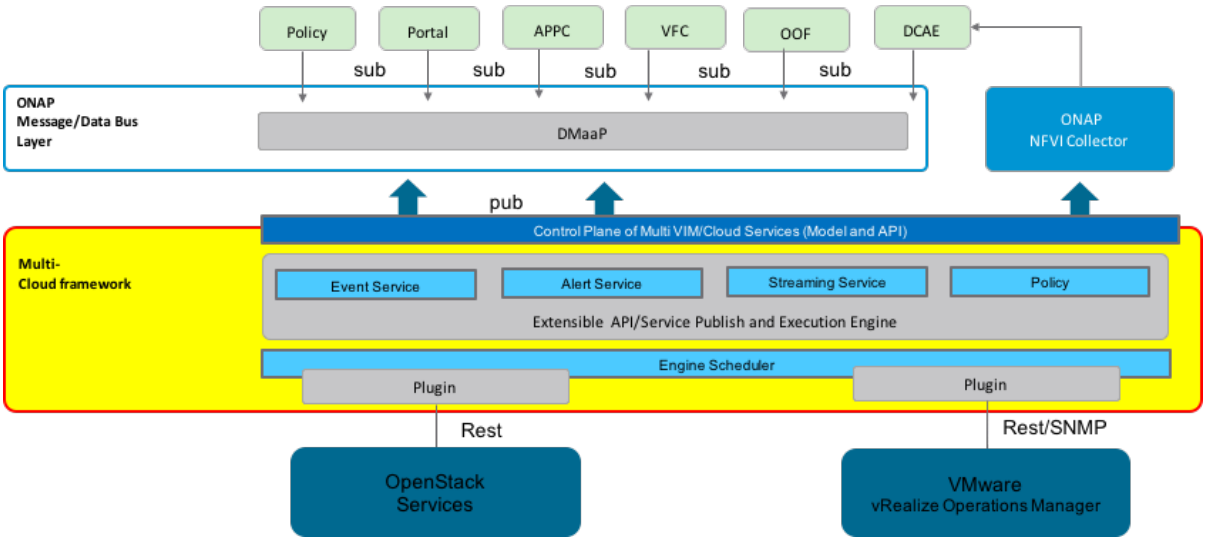
Anyone reading this section should have a good understanding of all the interacting modules.



Please refer to below graph for a high level project scope architecture. the Fcaps part will involve data collection and event/metrics/alerts federation part as below graph.



Please refer to below chart for the even-driven framework.



Platform Maturity

Refering to CII Badging Security Program and Platform Maturity Requirements, fill out the table below by indicating the actual level , the targeted level for the current release and the evidences on how you plan to achieve the targeted level.

Area	Actual Level	Targeted Level for current Release	How, Evidences	Comments
Performance	0	1	show basic performance profiling data in log	• 0 -- none • 1 – baseline performance criteria identified and measured • 2 & 3 – performance improvement plans created & implemented
Stability	0	1	show long run result in log	• 0 – none • 1 – 72 hours component level soak w/random transactions • 2 – 72 hours platform level soak w/random transactions • 3 – 6 months track record of reduced defect rate
Resiliency	1	2	add Multi VIM service and recovery capability	• 0 – none • 1 – manual failure and recovery (< 30 minutes) • 2 – automated detection and recovery (single site) • 3 – automated detection and recovery (geo redundancy)
Security	0	1	add CII badge passing tests	• 0 – none • 1 – CII Passing badge + 50% Test Coverage • 2 – CII Silver badge; internal communication encrypted; role-based access control and authorization for all calls • 3 – CII Gold
Scalability	1	1	depends on gepgraphic scaling	• 0 – no ability to scale • 1 – single site horizontal scaling • 2 – geographic scaling • 3 – scaling across multiple ONAP instances
Manageability	0	1	enable single logging output	• 1 – single logging system across components; instantiation in < 1 hour • 2 – ability to upgrade a single component; tracing across components; externalized configuration management
Usability	2	2	No UI and tutorial documentation need more work	• 1 – user guide; deployment documentation; API documentation • 2 – UI consistency; usability testing; tutorial documentation

- API Incoming Dependencies

List the API this project is expecting from other projects.
Prior to Release Planning review, Team Leads must agreed on the date by which the API will be fully defined. The API Delivery date must not be later than the release API Freeze date.

Prior to the delivery date, it is a good practice to organize an API review with the API consumers.

API Name	API Description	API Definition Date	API Delivery date	API Definition link (i.e.swagger)
Azure API	southbound API deisgn			

- ## API Outgoing Dependencies

API this project is delivering to other projects.

API Name	API Description	API Definition Date	API Delivery date	API Definition link (i.e.swagger)
Clustering placement/query	Query capacity and status for a group of resources			
Message bus pub/sub	Publish and subscribe events/notification			
Resource Hierarchy graphs	Traverse resource Hierarchy in given Cloud			
VIM query	Query available VIMs			
Image upload/download	Support upload and download images			

- ## 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
Parser Adapter in SO	Translate Tosca template into VIM function description	

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.

- # 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
To fill out	To fill out

- # Known Defects and Issues

Provide a link toward the list of all known project bugs.

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
No issues found										

- # 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).

Risk identified	Mitigation Plan	Contingency Plan
APPC event integration	keep the existing way	work together on vm polling part of code
OOF event integration	keep the existing way	work together on Clustering query and resource graphs update
DCAE integration	keep the existing way	identify examples of alert/metrics
AAI integration	keep the existing way	identify examples of HPA

- # Resources

Fill out 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 is optional and may be used to document internal milestones within a project team or multiple project teams. For instance, 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, it is erecommended to provide these agreements and dates in this section.

It is not expected to have a detailed project plan.

Date	Project	Deliverable
To fill out	To fill out	To fill out

- # Documentation, Training

- Highlight the team contributions to the specific document related to he 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.