Release Planning Template - Doc Beijing Release (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.

Overview

Project Name	Enter the name of the project
Target Release Name	Beijing
Project Lifecycle State	Incubation Refer to ONAP Charter, section 3.3 Project Lifecycle for further information
Participating Company	AT&T, Amdocs, Huawei, Nokia, China Mobile, Orange, ZTE, Accenture

Scope

What is this release trying to address?

1. Update project-level documentation for Beijing Release requirements
2. Improving the Usability of the Platform:
   - Establish consistent tool chain and guidelines for API documentation
   - Create Welcome to ONAP and "sandbox" functionality for novices to ONAP
   - Automate verification process in the CI/CD Documentation tool chain
   - Migrate seed documentation currently in the wiki or gerrit that is being maintained by approved projects
   - Refine / expand current documentation for VNFs, OpenStack interfaces, UIs, and LF Tool Chain
   - Establish feedback mechanisms in Wiki and/or Readthedocs to improve documentation

Use Cases

The documentation created by this project must support ONAP high level Beijing use cases.

Lower level use cases specific to documentation project scope include:

1. Publish a finished set of formal change-controlled documentation for any user audience working with the ONAP Beijing release.
2. Establish a consistent method for documenting APIs
3. Create welcoming documentation and a place to play for novices to ONAP
4. Create a way for users to provide feedback

Minimum Viable Product

Final documentation for ONAP Beijing Use Cases

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

Summary	T	Reporter	P	Status

No issues found

Stories

Longer term roadmap

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

Release Deliverables

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

Required (Level 1)	**Deliverable Description**
Beijing Project-level Updates	Published release documentation for all Beijing projects
Stretch Goals (Level 2)	**Deliverable Description**
Content Initiatives	• Published Guidelines for API documentation • Published "Welcome to ONAP" and "sandbox" functionality • Migrated seed documentation currently in the wiki or gerrit • Refined / expanded documentation for VNFs, OpenStack interfaces, UIs, and LF Tool Chain
Tool Chain Initiatives	• Tool Chain established for API documentation • Refined and hardened CI/CD tool chain and processes • Feedback mechanisms implemented in Wiki and/or Readthedocs

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.

The diagram below illustrates what is accomplished in the setup steps above from the perspective of a file structure created for a local test, a jenkins verify job, and/or published release documentation including:

- ONAP gerrit project repositories,
- doc project repository master document index.rst, templates, configuration, and other documents
- submodules directory where other project repositories and directories/files are referenced
- file structure: directories (ellipses), files(boxes)
- references: directory/files (solid edges), git submodule (dotted edges), sphinx toctree (dashed edges)

blocked URL

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	N/A	N/A		0 -- none 1 – baseline performance criteria identified and measured 2 & 3 – performance improvement plans created & implemented
Stability	N/A	N/A		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	N/A	N/A		0 – none 1 – manual failure and recovery (< 30 minutes) 2 – automated detection and recovery (single site) 3 – automated detection and recovery (geo redundancy)
Security	N/A	N/A		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	N/A	N/A		0 – no ability to scale 1 – single site horizontal scaling 2 – geographic scaling 3 – scaling across multiple ONAP instances
Manageability	N/A	N/A		1 – single logging system across components; instantiation in < 1 hour 2 – ability to upgrade a single component; tracing across components; externalized configuration management
Usability	1	1	Usability Initiatives for Beijing (Release 2)	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)
N/A				

- ## 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)
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
N/A		

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
N/A	

- ## Known Defects and Issues

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

- ## 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
Scope of release requirements	Complete and Interlock on Release Index	To fill out
Availability of contributors	Recruit resources as needed	Adjust scope & deliverables accordingly

- ## 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.