

# Beijing-M1-CLI Release planning

- 1 [Overview](#)
- 2 [Scope](#)
  - 2.1 [Use Cases](#)
    - 2.1.1 [Epics](#)
    - 2.1.2 [Stories](#)
  - 2.2 [Longer term road map](#)
- 3 [Release Deliverable](#)
- 4 [Sub-Components](#)
- 5 [ONAP Dependencies](#)
  - 5.1 [Platform Maturity](#)
- 6 [Architecture](#)
  - 6.1 [High level architecture diagram](#)
    - 6.1.1 [CLI Components](#)
    - 6.1.2 [Sample YAML CLI template](#)
  - 6.2 [API Incoming Dependencies](#)
  - 6.3 [CLI Outgoing Dependencies](#)
  - 6.4 [Third Party Products Dependencies](#)
- 7 [Testing and Integration Plans](#)
- 8 [Gaps](#)
- 9 [Known Defects and Issues](#)
- 10 [Risks](#)
- 11 [Resources](#)
- 12 [Release Milestone](#)
- 13 [Team Internal Milestone](#)
- 14 [Documentation, Training](#)
- 15 [Other Information](#)
  - 15.1 [Vendor Neutral](#)
  - 15.2 [Free and Open Source Software](#)

## Overview

Project Name	Enter the name of the project
Target Release Name	Beijing
Project Lifecycle State	incubation
Participating Company	Huawei, AT&T

## Scope

- Stabilise ONAP commands and CLI platform to meet S3P
- Provides Commands for VNF (netconf) & VNF-SDK
  - Standardise CLI for VNF (Stretch goal)
- Closed-loop supporting commands (Stretch goal)
- ONAP command console (OCC)
- Standardise Open Command Specification (OCS).

## Use Cases

As a user/operator, i should be able to perform the ONAP operations from OCC/Linux console, required for Beijing release.

## Minimum Viable Product

Provide OCC to to operate ONAP environment from Portal & OOM.

Provides required commands for SO, SDC, MSB and AAI for performing Beijing use case (change mgmt, scaling) from command line.

## 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
CLI-91	Closed-loop support		Jan 08, 2018	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	CLOSED	Done
CLI-90	PNF/VNF support		Jan 08, 2018	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	CLOSED	Done
CLI-89	Command and Platform stabilisation		Jan 08, 2018	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	CLOSED	Done

3 issues

## Stories

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
CLI-88	Migrate SDC commands to use published REST API		Jan 08, 2018	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved
CLI-74	Setup Mock environment for a command		Nov 16, 2017	Aug 08, 2018		subhash kumar singh	Kanagaraj Manickam	==	CLOSED	Done
CLI-73	Make output format as plug-able component		Nov 09, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved
CLI-72	Add support for expression language in YAML schema		Nov 09, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved
CLI-68	Add CLI support for VNF-SDK		Nov 07, 2017	Sep 20, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	CLOSED	Done
CLI-67	Add support for VNF configuration		Nov 07, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved
CLI-66	Separate OCLIP and plugins into different git repo		Nov 07, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved
CLI-65	Add support for AAF		Nov 07, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved
CLI-57	Fix onap commands issues		Oct 24, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	CLOSED	Done
CLI-55	Implement autogeneration of moco json		Oct 16, 2017	Aug 08, 2018		subhash kumar singh	subhash kumar singh	==	CLOSED	Done
CLI-52	Add validation support for parameters in yaml		Oct 13, 2017	Aug 08, 2018		Kanagaraj Manickam	Kanagaraj Manickam	==	OPEN	Unresolved

11 issues

## Longer term road map

As CLI is critical in devops environment, in future, CLI will be extended to use in VNF boot scripts and Integration projects to make the ONAP integration point smoother and easier.

## Release Deliverable

Deliverable Name	Deliverable Description
CLI Nexus zip archive	ZIP archive used to install the CLI manually
CLI docker container	Docker container to run the ONAP command and will integrate into demo environment
Documentation	Developer guide User guide Release notes Architecture open-cli-schema-1.0 specification

## Sub-Components

NIL

## ONAP Dependencies

1. AAF project
  - a. for Authentication and authorization
2. MSB project
  - a. for service discovery
3. AAI
  - a. for customer, subscription, cloud management
  - b. external system management
4. SO
  - a. Service life cycle management
5. SDC
  - a. Service and resource model management

## Platform Maturity

Referring 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.

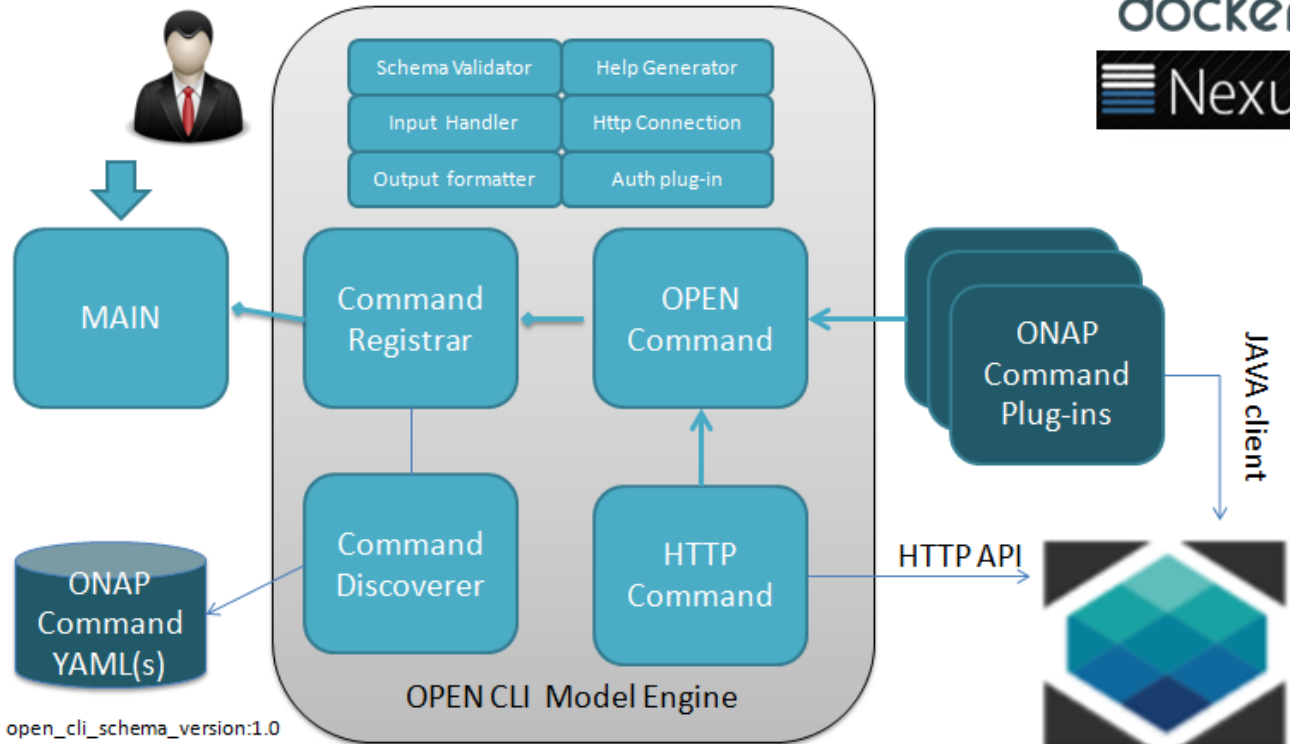
Area	Actual Level	Targeted Level for current Release	How, Evidences	Comments
Performance	0	1		<ul style="list-style-type: none"> <li>• 0 -- none</li> <li>• 1 – baseline performance criteria identified and measured</li> <li>• 2 &amp; 3 – performance improvement plans created &amp; implemented</li> </ul> <p><b>NOTE:</b></p> <p>Performance test is not performed in amsterdam though it performant, which is best observed during the usage.</p> <p>so current level is marked 0. In this release, we will add benchmarking test cases to show the level 1</p>
Stability	0	1		<ul style="list-style-type: none"> <li>• 0 – none</li> <li>• 1 – 72 hours component level soak w/random transactions</li> <li>• 2 – 72 hours platform level soak w/random transactions</li> <li>• 3 – 6 months track record of reduced defect rate</li> </ul> <p><b>NOTE:</b></p> <p>Test cases are not added to test the stability and benchmark it. so current level is marked 0. Otherwise, its very stable at level 1 or more.</p> <p>will address the test cases in this release.</p>

Resiliency	1	2		<ul style="list-style-type: none"> <li>• 0 – none</li> <li>• 1 – manual failure and recovery (&lt; 30 minutes)</li> <li>• 2 – automated detection and recovery (single site)</li> <li>• 3 – automated detection and recovery (geo redundancy)</li> </ul>
Security	0	1		<ul style="list-style-type: none"> <li>• 0 – none</li> <li>• 1 – CII Passing badge + 50% Test Coverage</li> <li>• 2 – CII Silver badge; internal communication encrypted; role-based access control and authorization for all calls</li> <li>• 3 – CII Gold</li> </ul>
Scalability	1	1		<ul style="list-style-type: none"> <li>• 0 – no ability to scale</li> <li>• 1 – single site horizontal scaling</li> <li>• 2 – geographic scaling</li> <li>• 3 – scaling across multiple ONAP instances</li> </ul>
Manageability	1	1		<ul style="list-style-type: none"> <li>• 1 – single logging system across components; instantiation in &lt; 1 hour</li> <li>• 2 – ability to upgrade a single component; tracing across components; externalized configuration management</li> </ul>
Usability	1	2		<ul style="list-style-type: none"> <li>• 1 – user guide; deployment documentation; API documentation</li> <li>• 2 – UI consistency; usability testing; tutorial documentation</li> </ul>

## Architecture

High level architecture diagram

# OPEN CLI Framework for ONAP



## CLI Components

- Open Command** – Models the most of the requirements of CLI and it's the CLI schema engine, understands the OPEN-CLI 1.0 schema and make it as Command
- Command Plug-in** : Provides extensibility support to implement any commands which are depends on java api
- HTTP Command** : An special command plug-in provided by framework to implement the Command on top of REST API without writing any java code. (only YAML template is sufficient)
- Command Discoverer** – Discovers the available CLI templates (YAML file created by using OPEN-CLI 1.0 schema) placed under OPEN\_CLI\_HOME directory or its sub-directories and register them into Command Registrar.
- Command Registrar** – Maintains the map of command name vs actual command executable
- Main** – Provides the interactive/direct command mode to run the commands from Linux OS console

## Sample YAML CLI template

Following sample YAML shows the YAML file used to create the microservice in Open-O using CLI

### openo microservice-create

```
open_cli_schema_version: 1.0
name: microservice-create
description: Register microservice into Open-O
service:
  name: msb
  version: v1

parameters:
- name: service-name
  description: Open-O service name
  type: string
  short_option: x
  long_option: service-name
  is_optional: false
```

```

- name: service-version
  description: Open-O service version
  type: string
  short_option: y
  long_option: service-version
  is_optional: false
- name: service-url
  description: Open-O service base url
  type: url
  short_option: r
  long_option: service-url
  is_optional: false
- name: node-ip
  description: Open-O service running node IP
  type: string
- name: node-port
  description: Open-O service running node port
  type: string
- name: create-or-update
  description: Open-O service create or update
  type: bool
  default_value: true
results:
  direction: portrait
  attributes:
    - name: name
      description: Open-O service name
      scope: short
      type: string
    - name: version
      description: Open-O service version
      scope: short
      type: string
    - name: url
      description: Open-O service base url
      scope: short
      type: url
    - name: status
      description: Open-O service status
      scope: short
      type: long
    - name: nodes
      description: Open-O service running nodes
      scope: long
      type: string
    - name: location
      description: Open-O service location
      scope: long
      type: url
http:
  request:
    uri: /services
    method: POST
    body: '{"serviceName":"${service-name}","version":"${service-version}","url":"${service-url}","protocol":"
REST","visualRange":"1","lb_policy":"hash","nodes":[{"ip":"${node-ip}","port":"${node-port}","ttl":0}]'
    headers:
    queries:
      createOrUpdate: ${create-or-update}
  success_codes:
    - 201
    - 200
  result_map:
    name: ${b{$.serviceName}}
    version: ${b{$.version}}
    url: ${b{$.url}}
    status: ${b{$.status}}
    nodes: ${b{$.nodes[*].ip}}:${b{$.nodes[*].port}}
    location: ${h{Location}}

```

# API Incoming Dependencies

To provide the required commands for each of the ONAP services, this project will depend on the REST API provided by every other services such as SO, AAI, etc

# CLI Outgoing Dependencies

NOTE: This project delivers CLI and not API

# Third Party Products Dependencies

Name	Description	Version
Docker	Docker container for CLI	Aligned with existing Docker version of ONAP
Ubuntu	For running CLI manually	14.04 / 16.04 32-bit/64-bit
Open JDK	For Java RTE	1.8

# 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
NIL	NIL

# 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
<a href="#">CLI-250</a>	<a href="#">Sonar cloud migration</a>		Feb 21, 2020	Feb 21, 2020		jitendra sharma	Kanagaraj Manickam		<b>OPEN</b>	Unresolved
<a href="#">CLI-229</a>	<a href="#">Minor CLI bug fixes</a>		Sep 30, 2019	Oct 14, 2019		Kanagaraj Manickam	Itohan Ukponmwan		<b>CLOSED</b>	Not a Bug
<a href="#">CLI-228</a>	<a href="#">Error Creating Service instance using CLI, CLI puts invalid Request ID</a>		Sep 30, 2019	Oct 01, 2019		Kanagaraj Manickam	Itohan Ukponmwan		<b>CLOSED</b>	Done
<a href="#">CLI-214</a>	<a href="#">Fix json variable name error</a>		Sep 06, 2019	Sep 06, 2019		Huang Haibin	Huang Haibin		<b>CLOSED</b>	Done
<a href="#">CLI-212</a>	<a href="#">Got invalid JSON error creating NS</a>		Aug 27, 2019	Oct 09, 2019		Huang Haibin	Huang Haibin		<b>CLOSED</b>	Done

CLI-200	Command service-instance-list is not available for product version onap-dublin		Aug 09, 2019	Oct 21, 2019	Kanagaraj Manickam	Vamshi Krishna Namilikonda		CLOSED	Done
CLI-199	CLI support for Multicloud		Aug 09, 2019	Aug 09, 2019	Kanagaraj Manickam	Janani B		OPEN	Unresolved
CLI-197	CLI support for AAI		Aug 08, 2019	Aug 08, 2019	Kanagaraj Manickam	Shashikanth VH		OPEN	Unresolved
CLI-168	Old version of CLI image is used in dublin release		May 22, 2019	Jul 10, 2019	Kanagaraj Manickam	Krzysztof Kuzmicki		CLOSED	Done
CLI-165	IllegalArgumentException occurred in CLI while using password with special characters		May 07, 2019	Aug 01, 2019	Ruoyu Ying	Ruoyu Ying		CLOSED	Done
CLI-164	Migrate Policy API to new version and change environment to ONAP-CLI from TEST		Apr 23, 2019	Jul 30, 2019	Ruoyu Ying	Kanagaraj Manickam		CLOSED	Duplicate
CLI-163	CLI bugs		Apr 19, 2019	Jun 03, 2019	Itohan Ukponmwan	Itohan Ukponmwan		CLOSED	Done
CLI-162	No CLI command to print owning Entity ID		Apr 19, 2019	May 06, 2019	Itohan Ukponmwan	Itohan Ukponmwan		CLOSED	Done
CLI-161	service-model-list commands prints out wrong uuid		Apr 18, 2019	May 06, 2019	Itohan Ukponmwan	Itohan Ukponmwan		CLOSED	Done
CLI-160	Created CLI commands do not work with the CLI but the same command works when curl is used directly		Apr 11, 2019	May 02, 2019	Itohan Ukponmwan	Itohan Ukponmwan		CLOSED	Cannot Reproduce
CLI-157	Json passed in the parameter couldn't have space inside		Mar 19, 2019	Mar 19, 2019	Kanagaraj Manickam	Ruoyu Ying		CLOSED	Not a Bug
CLI-155	Enhancements for VSP Certification Feature		Mar 18, 2019	Apr 26, 2019	James Guistwite	James Guistwite		CLOSED	Done
CLI-147	Error in documentation(Cook Book)		Feb 20, 2019	Oct 14, 2019	Kanagaraj Manickam	Ruoyu Ying		CLOSED	Done
CLI-127	Fix the issues in the APP-C CLI		Dec 11, 2018	Jan 23, 2019	Kanagaraj Manickam	Kanagaraj Manickam		OPEN	Unresolved
CLI-125	Hyperlink missing		Nov 09, 2018	Feb 13, 2019	Manish Kumar	Manish Kumar		CLOSED	Done

Showing 20 out of 70 issues

## 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
NONE		-

## 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 recommended to provide these agreements and dates in this section.

It is not expected to have a detailed project plan.

Date	Project	Deliverable
NIL	NIL	NIL

## 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](#).