Dublin-M1-CLI Release planning

- Overview
- Scope
  - 2.1 Use Cases
    - 2.1.1 Epics
    - 2.1.2 Stories
  - 2.2 Longer term road map
- 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

<table>
<thead>
<tr>
<th>Project Name</th>
<th>Enter the name of the project</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Release Name</td>
<td>Dublin</td>
</tr>
<tr>
<td>Project Lifecycle State</td>
<td>incubation</td>
</tr>
<tr>
<td>Participating Company</td>
<td>Huawei, Intel</td>
</tr>
</tbody>
</table>

Scope

- Support VTP for LFN CVC
- Support ONAP for enabling service specific CLI

Use Cases

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

Minimum Viable Product

- Enable required features in OCLIP for LFN CVC VTP

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

<table>
<thead>
<tr>
<th>Key</th>
<th>Summary</th>
<th>T</th>
<th>Created</th>
<th>Updated</th>
<th>Due</th>
<th>Assignee</th>
<th>Reporter</th>
<th>P</th>
<th>Status</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLI-129</td>
<td>Support VTP for CVC</td>
<td>⚡️</td>
<td>Jan 23, 2019</td>
<td>Apr 26, 2019</td>
<td>Kanagaraj Manickam</td>
<td>Kanagaraj Manickam</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-116</td>
<td>Meet CLI badging for dublin</td>
<td>⚡️</td>
<td>Jun 27, 2018</td>
<td>Apr 12, 2019</td>
<td>Kanagaraj Manickam</td>
<td>Kanagaraj Manickam</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
</tbody>
</table>

2 issues

## Stories

<table>
<thead>
<tr>
<th>Key</th>
<th>Summary</th>
<th>T</th>
<th>Created</th>
<th>Updated</th>
<th>Due</th>
<th>Assignee</th>
<th>Reporter</th>
<th>P</th>
<th>Status</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLI-168</td>
<td>Old version of CLI image image is used in dublin release</td>
<td>✅️</td>
<td>May 22, 2019</td>
<td>Jul 10, 2019</td>
<td>Kanagaraj Manickam</td>
<td>Krzysztof Kuzmicki</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-167</td>
<td>Please add Jim Baker to your project CLI badging editors</td>
<td>✓</td>
<td>May 22, 2019</td>
<td>May 24, 2019</td>
<td>Kanagaraj Manickam</td>
<td>Pawel Pawlak</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-163</td>
<td>CLI bugs</td>
<td>🔴</td>
<td>Apr 19, 2019</td>
<td>Jun 03, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-162</td>
<td>No CLI command to print owning Entity ID</td>
<td>🔴</td>
<td>Apr 19, 2019</td>
<td>May 06, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-161</td>
<td>service-model-list commands prints out wrong uuid</td>
<td>🔴</td>
<td>Apr 18, 2019</td>
<td>May 06, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-160</td>
<td>Created CLI commands do not work with the CLI but the same command works when curl is used directly</td>
<td>🔴</td>
<td>Apr 11, 2019</td>
<td>May 02, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Cannot Reproduce</td>
<td></td>
</tr>
<tr>
<td>CLI-159</td>
<td>Complete Vulnerability Review Table</td>
<td>✓</td>
<td>Mar 25, 2019</td>
<td>Apr 18, 2019</td>
<td>Apr 10, 2019</td>
<td>Kanagaraj Manickam</td>
<td>Amy Zwarico</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
</tr>
<tr>
<td>CLI-158</td>
<td>Create CLI commands to use for vft model in Dublin</td>
<td>🔴</td>
<td>Mar 20, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-156</td>
<td>Create CLI command to delete VLM</td>
<td>🔴</td>
<td>Mar 18, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-155</td>
<td>Enhancements for VSP Certification Feature</td>
<td>🔴</td>
<td>Mar 18, 2019</td>
<td>Apr 26, 2019</td>
<td>James Guistwite</td>
<td>James Guistwite</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-154</td>
<td>Create CLI command to list vendor license models</td>
<td>🔴</td>
<td>Mar 16, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-153</td>
<td>Create CLI commands to add cloud regions to a customer subscription</td>
<td>🔴</td>
<td>Mar 05, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-152</td>
<td>Create CLI command to delete cloud using multicloud</td>
<td>🔴</td>
<td>Feb 26, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-151</td>
<td>Create CLI Command to list the flavors of a Cloud</td>
<td>🔴</td>
<td>Feb 26, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-150</td>
<td>Update Cloud-list command to printout other important parameters</td>
<td>🔴</td>
<td>Feb 26, 2019</td>
<td>Jul 10, 2019</td>
<td>Kanagaraj Manickam</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Won't Do</td>
<td></td>
</tr>
<tr>
<td>CLI-148</td>
<td>Update cloud-create command to include more parameters</td>
<td>🔴</td>
<td>Feb 22, 2019</td>
<td>Jul 12, 2019</td>
<td>Itohan Ukponmwan</td>
<td>Itohan Ukponmwan</td>
<td></td>
<td>CLOSED</td>
<td>Done</td>
<td></td>
</tr>
</tbody>
</table>
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

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

Sub-Components

NIL

ONAP Dependencies

1. VNFSDK
   a. API to conduct tests, verification and market places

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.

<table>
<thead>
<tr>
<th>Area</th>
<th>Actual Level</th>
<th>Targeted Level for current Release</th>
<th>How, Evidences</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 0 -- none</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 1 -- baseline performance criteria identified and measured</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 2 &amp; 3 -- performance improvement plans created &amp; implemented</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td>------</td>
<td>-----------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Stability</td>
<td>0 – none</td>
<td>1 – 72 hours component level soak w/random transactions</td>
<td>2 – 72 hours platform level soak w/random transactions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 – 6 months track record of reduced defect rate</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resiliency</td>
<td>0 – none</td>
<td>1 – manual failure and recovery (&lt; 30 minutes)</td>
<td>2 – automated detection and recovery (single site)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 – automated detection and recovery (geo redundancy)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>0 – none</td>
<td>1 – CII Passing badge + 50% Test Coverage</td>
<td>2 – CII Silver badge; internal communication encrypted; role-based access control and authorization for all calls</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 – CII Gold</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scalability</td>
<td>0 – no ability to scale</td>
<td>1 – single site horizontal scaling</td>
<td>2 – geographic scaling</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 – scaling across multiple ONAP instances</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manageability</td>
<td>1 – single logging system across components; instantiation in &lt; 1 hour</td>
<td>2 – ability to upgrade a single component; tracing across components; externalized configuration management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usability</td>
<td>1 – user guide; deployment documentation; API documentation</td>
<td>2 – UI consistency; usability testing; tutorial documentation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Architecture**

High level architecture diagram
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 Open 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.
- **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

```yaml
open 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[0].ip}:${.nodes[0].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 service such as SO, AAI, etc.

CLI Outgoing Dependencies

NOTE: This project delivers CLI and not API.

Third Party Products Dependencies

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docker</td>
<td>Docker container for CLI</td>
<td>Aligned with existing Docker version of ONAP</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>For running CLI manually</td>
<td>14.04 / 16.04 32-bit/64-bit</td>
</tr>
<tr>
<td>Open JDK</td>
<td>For Java RTE</td>
<td>1.8</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Gaps identified</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIL</td>
<td>NIL</td>
</tr>
</tbody>
</table>

Known Defects and Issues

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

<table>
<thead>
<tr>
<th>Key</th>
<th>Summary</th>
<th>T</th>
<th>Created</th>
<th>Updated</th>
<th>Due</th>
<th>Assignee</th>
<th>Reporter</th>
<th>P</th>
<th>Status</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLI-466</td>
<td>Node port issue</td>
<td>🔴</td>
<td>Jun 28, 2022</td>
<td>Sep 19, 2022</td>
<td>Dan Xu</td>
<td>Kanagaraj Manickam</td>
<td></td>
<td>OPEN</td>
<td>Unresolved</td>
<td></td>
</tr>
<tr>
<td>CLI-440</td>
<td>Sonar issues present in ONAP vnfsdk validation</td>
<td>🔴</td>
<td>Mar 20, 2022</td>
<td>Mar 20, 2022</td>
<td>prakash eswaramoorthy</td>
<td>prakash eswaramoorthy</td>
<td></td>
<td>OPEN</td>
<td>Not a Bug</td>
<td></td>
</tr>
<tr>
<td>CLI-438</td>
<td>issue in CLI docker image build</td>
<td>🔴</td>
<td>Mar 08, 2022</td>
<td>Mar 08, 2022</td>
<td>prakash eswaramoorthy</td>
<td>prakash eswaramoorthy</td>
<td></td>
<td>DELIVERED</td>
<td>Done</td>
<td></td>
</tr>
<tr>
<td>CLI-422</td>
<td>fix documentation</td>
<td>🔴</td>
<td>Oct 28, 2021</td>
<td>Sep 19, 2022</td>
<td>Dan Xu</td>
<td>Thomas Kulik</td>
<td></td>
<td>OPEN</td>
<td>Unresolved</td>
<td></td>
</tr>
<tr>
<td>CLI-376</td>
<td>update discovery, run robot scripts</td>
<td>🔴</td>
<td>Mar 29, 2021</td>
<td>Sep 19, 2022</td>
<td>Dan Xu</td>
<td>sharath reddy</td>
<td></td>
<td>OPEN</td>
<td>Unresolved</td>
<td></td>
</tr>
</tbody>
</table>
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).

<table>
<thead>
<tr>
<th>Risk identified</th>
<th>Mitigation Plan</th>
<th>Contingency Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>NONE</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Date</th>
<th>Project</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIL</td>
<td>NIL</td>
<td>NIL</td>
</tr>
</tbody>
</table>

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.
- 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.