Project Maturity Review for ONAP Portal

- Review Submission Date: 7/7/2020
- Project Description:
  - Project Landing Page: Portal Platform Project
  - Project Meeting Minutes: https://wiki.onap.org/pages/viewpage.action?pageId=45305320
- Leadership Resources:
  - Project & Release History
  - Architecture Alignment:
  - Artifact Information:
  - Other Information:

Information for PTL

This template is to be filled out for any project wishing to move to the "Mature" phase of the ONAP project lifecycle. Once completed, remove this info box and the send the link for this page to onap-tsc@lists.onap.org to initiate the maturity review.

Per the ONAP Technical Community Document, the metrics for maturity review are:

- Successful participation in releases: The project demonstrates stable output (code base, documents) within its history of releases in accordance with the release policy.
- Architecture has been reviewed by the Architecture Committee
- Project is active and contributes to ONAP: The project demonstrates a stable or increasing number of contributions across recent releases. Contributions are commits which got merged to a repository of an ONAP project or a related upstream project. Commits can for example be patches to update the requirements document of a project, code addition to an ONAP or upstream project repository, new test cases and so forth.
- Mature artifacts produced: The project demonstrates that the artifacts produced by the project are deployable (where applicable) and have been successfully deployed, configured and used by end users (typically, service providers).

Review Submission Date: 7/7/2020

Project Description:

The ONAP Portal is a platform that provides the ability to integrate different ONAP applications into a centralized Portal Core. It allow decentralized applications to run within their own infrastructure while providing common management services and connectivity. The Portal core provides capabilities including application onboarding & management, centralized access management, and hosted application widgets. Using the provided SDK, application developers can leverage the built-in capabilities (Services / API / UI controls) along with bundled tools and technologies.

Project Landing Page: Portal Platform Project

Project Meeting Minutes: https://wiki.onap.org/pages/viewpage.action?pageId=45305320

Leadership Resources:

<table>
<thead>
<tr>
<th>Role</th>
<th>Name</th>
<th>Linux Foundation Login (LFID)</th>
<th>Email Address (if different from @ macro)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PTL</td>
<td>Sunder Tattavarada</td>
<td>tattasunder</td>
<td></td>
</tr>
<tr>
<td>Committers</td>
<td>Manoop Talasila</td>
<td></td>
<td>talasila</td>
</tr>
<tr>
<td></td>
<td>Lorraine Welch</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Project & Release History

- How long the project has been active: "At launch": 9th November 2017
- Release Participation: Amsterdam through Frankfurt
- Engagement levels for past releases (up to 3):
  - Commits per Release:
• Contributors per Release:
• Companies per release:

Statistics taken from Bitergia: https://onap.biterg.io/app/kibana#/dashboard/Overview

<table>
<thead>
<tr>
<th>Release name and timeframe</th>
<th>commits</th>
<th>contributors</th>
<th>companies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amsterdam (1/1/2017 to 11/16/2017)</td>
<td>183</td>
<td>14</td>
<td>5</td>
</tr>
<tr>
<td>Beijing (11/17/2017 to 6/7/2018)</td>
<td>146</td>
<td>22</td>
<td>8</td>
</tr>
<tr>
<td>Cassablanca (6/8/2018 to 11/30/2018)</td>
<td>149</td>
<td>31</td>
<td>9</td>
</tr>
<tr>
<td>Dublin (12/1/2018 to 7/9/2019)</td>
<td>219</td>
<td>27</td>
<td>8</td>
</tr>
<tr>
<td>El Alto (7/10/2019 to 10/24/2019)</td>
<td>285</td>
<td>14</td>
<td>4</td>
</tr>
<tr>
<td>Frankfurt (10/25/2019 to 6/18/2020)</td>
<td>433</td>
<td>28</td>
<td>6</td>
</tr>
</tbody>
</table>

• Use Cases:
  • User/Access Management for GUI Applications (running with and without AAF)
  • Drive Consistent User experience
  • Single Point of Entry to GUI applications

Architecture Alignment:

• ARC Portal Component Description - Frankfurt

Artifact Information:

• Code, Test Cases, documentation:
  • https://gerrit.onap.org/r/admin/repos/portal
  • https://gerrit.onap.org/r/admin/repos/portal/sdk
  • https://gerrit.onap.org/gitweb?p=integration/csit.git;a=tree;f=tests/portal/testsuites;hb=HEAD
  • https://gerrit.onap.org/gitweb?p=integration/csit.git;a=tree;f=tests/portal-sdk/testsuites;hb=HEAD
  • https://docs.onap.org/projects/onap-portal/en/latest/index.html#master-index

• Successful Deployments and Integration Testing
  • Portal is a critical GUI component and is required to be always up and running for Applications like SDC, VID, Policy Manager GUI, A&AI UI to be accessible
  • It encapsulates AAF's RBAC functionality
  • VID and Policy Manager GUI applications are built on the top of Portal SDK platform
  • Portal Pair Wise Testing for Frankfurt Release (Integration Testing)
  • Portal Pair Wise Testing for El Alto Release (Integration Testing)
  • Portal is being used across the community for Access Management to ONAP GUI applications

Other Information:

• Portal Presentations & Talks

• S3P - Guilin Release Platform Maturity

Portal is a GUI platform that provides the ability to integrate different ONAP platform GUls into a centralized portal.

It provides:

• The capability to allow other ONAP components to run within their own infrastructure while providing common management services and capabilities in a centralized way
• Provides common capabilities such as application on-boarding and management, centralized access management and hosting application widgets, Context-Aware UIControls, Visualization & Reporting Engine
• Provides SDK capabilities to access portal capabilities

Portal is used by:

• VID, Policy, AAI, SDC, MSB and CLI
• VID, Policy, AAI UI, SDC applications require Portal to be up to be accessible to the end user as these applications contact Portal to retrieve roles of the user who is logging in.
• VID, Policy, Portal are built completely on Portal SDK platform which promotes Rapid Application Development and encapsulates Rich UI templates, Security, Logging, communication with AAF/Portal, Transaction Management, Access Management, Reporting etc

In addition to regular user, there are admin roles available for Portal to perform some administrative activities of Portal itself and the administration of on-boarded apps.

From the ONAP Portal, administrators:

• access the same functionality accessible to users
• manage users and application admins
• onboard applications and widgets (developed using the Portal as a platform)
• edit the functional menu