ONAP Operations Manager Project

IntroductionThe ONAP Operations Manager (OOM) is responsible for life-cycle management of the ONAP platform itself; components such as MSO, SDNC, etc. It is not responsible for the management of services, VNFs or infrastructure instantiated by ONAP or used by ONAP to host such services or VNFs. OOM uses the open-source Kubernetes container management system as a means to manage the Docker containers that compose ONAP where the containers are hosted either directly on bare-metal servers or on VMs hosted by a 3rd party management system. OOM ensures that ONAP is easily deployable and maintainable throughout its life cycle while using hardware resources efficiently. In summary OOM provides the following capabilities:

- **Deployment** - with built-in component dependency management (including multiple clusters, federated deployments across sites, and anti-affinity rules)
- **Configuration -** unified configuration across all ONAP components
- **Monitoring** - real-time health monitoring feeding to a Consul UI and Kubernetes
- **Heal**- failed ONAP containers are recreated automatically
- **Clustering and Scaling** - cluster ONAP services to enable seamless scaling
- **Upgrade** - change-out containers or configuration with little or no service impact
- **Deletion** - cleanup individual containers or entire deployments

OOM supports a wide variety of cloud infrastructures to suit your individual requirements.

Table of Contents

blocked URL

OOM Team

Committers and contributors can be found here: OOM Team

Contributing to OOM

Feedback: You can provide feedback on the onap-discuss list, sign up here: ONAP Discuss List

Bug reports: Jira

Weekly Meeting Notes: OOM Weekly meeting notes

Contributions: Instructions on how to set up Gerrit is here: Configuring Gerrit. Please join the onap-discuss list (OOM posts are tagged with oom) or attend OOM's weekly meetings for information on how you can help.

Commit requirements: Commits on OOM are always welcome and will be reviewed with care by OOM committers. As OOM is an installer, we ask to have a specific (but very close to Commit Messages) way to create them. They must comply with following pattern:

```
[NAME_OF_COMPONENT|DOC|COMMON|GENERIC] Meaningful title (from OOM side)

at least one sentence explaining the change done in this patch, cause and consequences and possibly more of
course

Issue-ID: AS_WE_ARE_FORCED_BUT_MEANINGLESS
Change-ID: xxx
Sign-off: xxx
```

Commit message will be the last part that will stay with our code so it must clearly explain the changes, the "why" and the consequences. If it change OOM behavior in any way, documentation **must** be also updated.

Merge requests which are not following this pattern will not be merged.

Please read the following pages and follow the guidelines for writing commit message contained therein.

- http://bit.ly/goodcommitmessages
- http://who-t.blogspot.com/2009/12/on-commit-messages.html
- http://dep.debian.net/deps/dep3/

Development Policies, Standards, and Coding

- Developer Best Practices (note that as OOM is not a standard Java project, the coding specific points don't apply)
- Development Procedures and Policies
- Code Review

OOM Quick Start and Cloud Setup Guides

The official OOM Quick Start Guide is found at the onap.readthedocs.io site. If you need to setup a Kubernetes cloud to host your ONAP deployment you can find descriptions of how to do so for many cloud technologies in the OOM Cloud Setup Guide.

OOM User and Developer Guides

OOM uses the Helm / Kubernetes container management system to orchestrate the life cycle of the ONAP infrastructure components. If you're a user of OOM you'll want to read the OOM User Guide. If you're an ONAP developer and would like to add a component to ONAP you'll want to read the OOM Developer Guide.

Demo Video

If you'd like to see the installation of ONAP by OOM take a look at this short video demonstration by Mike Elliott: OOM Demo Video

Amsterdam Release Planning

- M1 Release Planning
- M1 Release Planning Checklist link
- M2 Functionality Freeze Checklist link
- M3 API Freeze Checklist link
- M4 Code Freeze Checklist link
- RCx Checklist link
- Sign-Off Checklist link
- OOM for Planning Milestone Checklist Template

Beijing Release Planning

- OOM Beijing - M1 Release Planning Template

Casablanca Release Planning

- OOM Casablanca Release Artifacts

Links to Further Information

- The OOM project proposal page is here: Approved Project Proposal.
- If you're interested in project status, look here: OOM Deployment Status.