

OLM V1 GitHub Project Overview

Background:

The operator framework is reimagining the operator-lifecycle-manager project as a set of tightly focused projects that fulfill specific needs. This reimagination of the project is referred to as OLM V1 and details about the vision of this experience can be found and debated [here](#).

A core aspect of OLM V1 is the introduction of kubernetes controllers that act as primitives fulfilling targeted use cases, for example,

- [RukPak](#) is responsible for retrieving and installing off cluster manifests.
- [Deppy](#) is responsible for identifying packages that fulfill a set of constraints.

The work required to bring OLM V1 to the world will exist as issues across multiple projects. In an effort to provide users with a singular view for OLM V1 Development progress, these issues will be tracked in the [OLM V1 GitHub Project](#). This GitHub Project will contain a number of “views”, such as:

- **Backlog:** A view that acts as a collection of work to be done. Issues that appear in this view will be labeled with `olm-v1/backlog`.
- **Milestone X:** A view that represents a specific milestone in the OLM V1 release path and typically implements some end-to-end workflow. Issues that appear in this view will be labeled with `olm-v1/m#`, where # is the milestone number.
- **GA:** A view detailing work that must be completed prior to OLM V1 reaching the General Availability state. Issues shown in this view will be labeled with `olm-v1/ga`. An example of an issue that should be added to this view is a refactor for a short-term solution that needs to be redesigned.

Frequently Asked Questions:

Why are OLM V1 Milestone Issues tracked with labels instead of milestones?

Many of the issues related to OLM V1 milestones will exist in other GitHub repositories. We did not want to:

- Tightly couple the development of these core components to OLM V1.
- Force each of these repositories to create an OLM V1 Milestone X milestone

Agenda:

Each of these meetings will be recorded and made available on the [official Operator Framework Youtube Channel](#).

December 2nd, 2022

Topics:

- Canceled, will continue weekly in the OLM WG/Issue Triage Meetings

December 1st, 2022

Topics:

- Continue generating stories

November 30th, 2022

Topics:

- Review the [OLM V1 Milestone 1 Issue](#) in the operator-framework/enhancement repo
- Proposed tracking conventions for work
 - Level 1 - Milestone: Issue created in the operator-framework/enhancement repo
 - Level 2 - Meta Issue: Represented by an issue labeled with `meta`
 - Level 3 - Issue: Any issue in a milestone that is not labeled with `meta` or in the operator-framework/enhancement repo
- Review proposed work items for <https://github.com/operator-framework/operator-controller/issues/62>
- Generate issues for <https://github.com/operator-framework/operator-controller/issues/63>

November 29th, 2022

Topics:

- Should we generate issues async and discuss them in the weekly working group meeting
 - [Joe] We should eventually move to a 1x or 2x weekly meeting where we discuss issues
 - [Joe] Issues should be allowed to be generated outside of this meeting, should be spoken about at this meeting
 - i. This matches the process we'll eventually
 - [Joe] There are 3 levels of planning
 - i. Milestones (Synchronous Meetings)
 - ii. Stories (Synchronous Meetings)
 - iii. Work Items (Asynchronous Meetings)
 - Work Items will probably be related 1-1 with a PR
 - iv.

November 22nd, 2022

Topics:

- [Alex]: Provide a quick summary of this initiative and perform a quick walkthrough of the OLM V1 GitHub Project and the proposed process followed by feedback
- We should decide:
 - How far into the future should we plan?
 - What's the ideal venue for async feedback
 - We will need to meet more than 1x a week to start fleshing this out, can we meet again tomorrow? When?
 - How frequently do we meet?

Notes:

- Milestones should be focused on User Experience / Stories
- Milestones should be small and focused
 - Milestone 1 may be an exception because the project needs a starting point
- Will hold off in defining Milestone 2
 - The further into the future we try to plan, the less we know what we're going to do.
 - We will avoid creating a bloated backlog.
 - Once Milestone 1 is completed and milestones are smaller, we could probably plan 1 to 2 milestones out.
 - We need a way to allow the community to propose a focus/user story for an upcoming milestone.
 - Could create milestone labels
 - Could create a "Milestone X" discussion pool
 - ACTION ITEM: Create an issue to track how Milestone suggestions are made.
 - How do we collect feedback on a milestone?
 - Austin suggested involving OKD
 - Jesus suggested promoting the milestone, creating demos, youtube videos, and providing a forum for feedback.
 - Austin: Submit a blog to OpenSource.com
 - ACTION ITEM: Create an issue to track our milestone release processes and how we'll promote and gather feedback
- Async communication should be driven through kubernetes slack channels
 - #Olm-dev
 - #Rukpak-dev
 - #Deppy-dev
- In person meetings
 - Talking about milestone stuff 1x a week
 - OLM Working Group
 - Day to day story breakdown within that story should happen
 - Story breakdown will be led by individual projects.
- Framework doc team wants to get involved with OLM V1
 - How can they get involved?
 - needs-docs label?
 - Should docs be involved at the milestone level?
 - Action Item: A documentation issue should be created for each milestone.