CEP-2 Kubernetes Operator

Working doc

https://docs.google.com/document/d/18Ow4R3tB9GIvdcFO7WmUvjb0a-sT6h0zSCEnfHsPz58/edit?usp=sharing

Everything below this line is currently a WIP taking place in the link above

----

Status

Current state: *Under Discussion*

Discussion thread: *here*

JIRA: *here*

Released: N/A

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Motivation

To make it easy to run Cassandra on Kubernetes. If an operator is like a robot in your datacenter running your Cassandra cluster, what does that operator need: a) To make informed decisions. b) Exposed from node to cluster to take actions
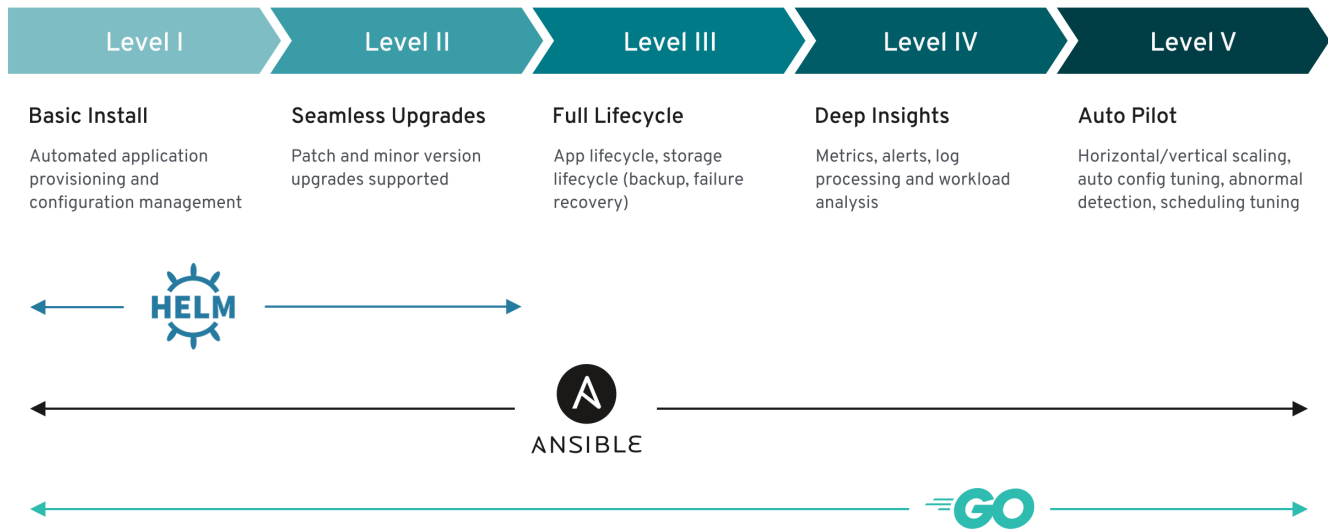
Audience

- Devops
- Developers
- Ops

Goals

Operator Capability Level

(Taken from https://operatorframework.io/operator-capabilities/)

Level I	Level II	Level III	Level IV	Level V
Basic Install	**Seamless Upgrades**	**Full Lifecycle**	**Deep Insights**	**Auto Pilot**
Automated application provisioning and configuration management	Patch and minor version upgrades supported	App lifecycle, storage lifecycle (backup, failure recovery)	Metrics, alerts, log processing and workload analysis	Horizontal/vertical scaling, auto config tuning, abnormal detection, scheduling tuning

HELM ←——————————————→

ANSIBLE ←——————————————————————————————————→

GO ←——————————————————————————————————→

- Lower the impedance between Kubernetes and Cassandra operations
- Achieve Level 3 Operator compliance, plus Horizontal Scaling from Level 5.
- Provide pathway to Level4
- Listed on https://operatorhub.io/

What does Level 1, 2 and 3 look like in the Cassandra world?

E.g.

- *L1: How do we support all the configuration options etc?*
- *L2: Patch management and pausing broken upgrades / running mixed versions etc.*

Level 1 operator for Apache Cassandra

Level 2 operator for Apache Cassandra

Level 3 operator for Apache Cassandra

Non-Goals

- *Remove the need for any Cassandra administration. (Not Level 5)*
- *Provide a serverless facade for Cassandra*
- *Official Docker images??*

Proposed Changes

A new repository as a sub-project for Apache Cassandra specifically for a Kubernetes Operator

New or Changed Public Interfaces

- pluggable components (SPIs) like authorisation, triggers, ..? - Better support for K8s service discovery mechanisms
- configuration - Probably something related to k8s service discovery e.g. a k8s seed provider maybe?
- The operator will define a Kubernetes Custom Resource (CRD), this will be the primary API for interacting with the operator.

Compatibility, Deprecation, and Migration Plan

- Target version: 3.x and above
- What impact (if any) will there be on existing users? **None**
- The operator will need to support minor and major version upgrades of Apache Cassandra

Test Plan

- Used as a part of CI/CD for Apache Cassandra project
  - dtest
  - harry
  - fallout
- TBD - Acceptance framework for k9s Operators. (Need guidance on that criteria)

Rejected Alternatives

- *HELM charts*