SIG PM Deep Dive
KubeCon NA 2018

Ihor Dvoretskyi ([@idvoretskyi](https://github.com/idvoretskyi))
SIG PM Chair

Stephen Augustus ([@justaugustus](https://github.com/justaugustus))
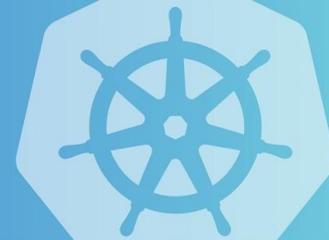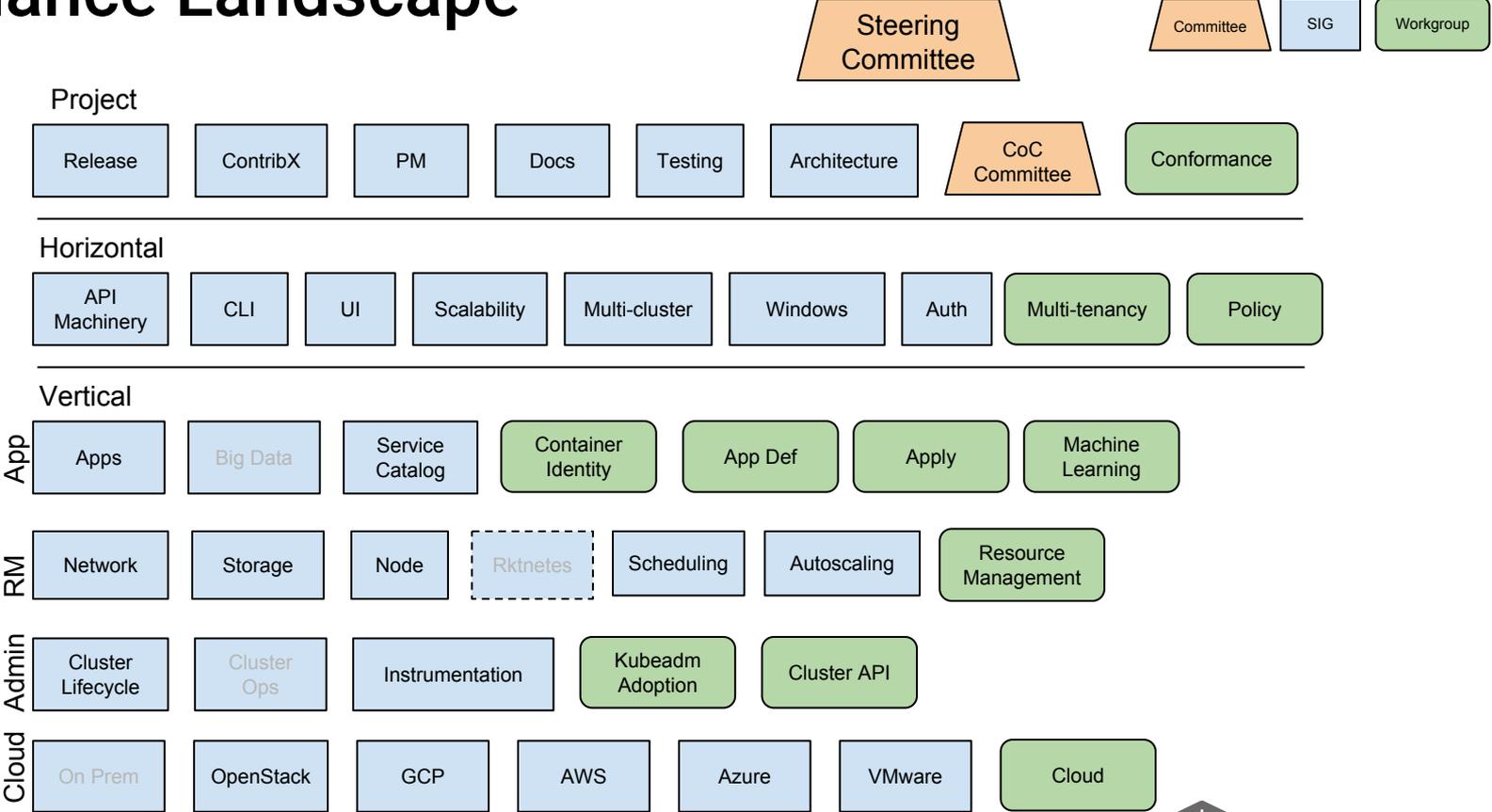*Kubernetes Product Management Chair*

kubernetes

History

- SIG PM was established in Q1'2017, as a successor to previously existing Product Management Working Group (formed in mid-2016)
- With SIG PM establishing, we have continued PM WG efforts, but made the group activities more formal (we have regular recorded meetings, mailing list, slack channel, etc.)

kubernetes

Governance Landscape

Steering Committee

Committee SIG Workgroup

Project

Release	ContribX	PM	Docs	Testing	Architecture	CoC Committee	Conformance

Horizontal

API Machinery	CLI	UI	Scalability	Multi-cluster	Windows	Auth	Multi-tenancy	Policy

Vertical

App

Apps	Big Data	Service Catalog	Container Identity	App Def	Apply	Machine Learning

RM

Network	Storage	Node	Rktnetes	Scheduling	Autoscaling	Resource Management

Admin

Cluster Lifecycle	Cluster Ops	Instrumentation	Kubeadm Adoption	Cluster API

Cloud

On Prem	OpenStack	GCP	AWS	Azure	VMware	Cloud

kubernetes

SIG-PM Charter

- Responsible for the forward-looking translation of portfolio-level objectives into project, product, marketing, and other specialized implementation strategies
- Staffing of specific release roles:
  - Release Team Enhancements Lead
  - Release Team Communications Coordinator
  - Product Roadmap committee

kubernetes

What we did last cycle

- SIG PM Survey
  (https://docs.google.com/presentation/d/1WgK-zLHIz5T0cj65hAL-rkDcO0ALPI4uY09W7_hj6qU/edit?usp=sharing)

What we did last cycle (continued)

- Revitalized the KEP process
  - KEP-1a: Meta KEP Implementation was marked as Implementable
  - Project tracking for KEP Implementation
    i. Project board: https://github.com/orgs/kubernetes/projects/5
    ii. Umbrella issue: https://github.com/kubernetes/enhancements/issues/617
  - Enhancements
    i. Term to encompass the set of KEPs (Kubernetes Enhancement Proposals) and Features
      1. KEPs: multi-release efforts comprised of multiple Features
      2. Features: units of work scoped to a single release cycle
  - Categorized open KEPs / design proposals as kind/kep
  - Renamed k/features to k/enhancements
  - Migrated KEPs from k/community to k/enhancements

Why KEPs?

Proposals came in many different shapes, sizes, and forms

if it works for other projects like Rust and Python...

- Proposals were inconsistent

- Lifecycle management is necessary

- Alleviate duplication of effort especially in the release process

Bringing Consistency Forward

photo: Sandip Dey

- Easier to review

- Lifecycle can be tracked in metadata, keeping everything in source control

- Single artifact can serve multiple purposes

kubernetes

When do you need a KEP?

"If an enhancement would be described in either written or verbal communication to anyone besides the KEP author or developer then consider creating a KEP."

Some things may *seem* easy to describe, until you have to actually do it.

kubernetes

Anatomy of a KEP

METADATA

MOTIVATION

SUMMARY

ARTIFACTS

```
authors:
- jdumars
title: Dry Run
kep_number: null
reviewers:
- jdumars
approvers:
- jdumars
state: implementable
superseded_by: []
last_updated: 2018-12-09T01:37:52.328024Z
created: 2018-12-09T01:36:50.934484Z
uuid: f77b7655-5930-4d64-b0ab-8ab476dd15a3
sections:
- summary.md
- motivation.md
- README.md
- guides/developer.md
- guides/operator.md
- guides/teacher.md
- graduation_criteria.md
owning_sig: sig-api-machinery
sig_wide: true
```

Must be one of **provisional**, **implementable**, **implemented**, **deferred**, **rejected**, **withdrawn**, or **replaced**.

Sections added as the KEP state changes and moves toward implementation
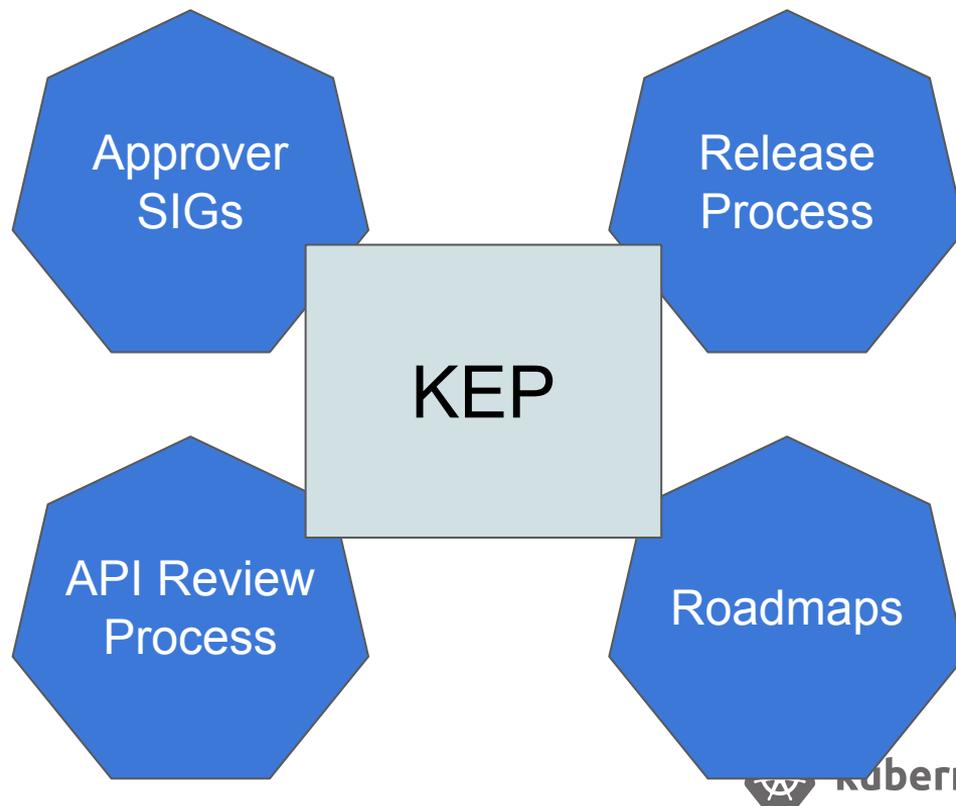
KEPs Serve Many Audiences

Hey! It's a lighthouse!

Approver SIGs

Release Process

KEP

API Review Process

Roadmaps

Motivation

What is the **need**?

- What is the problem being solved?

- Describe the significance of the problem well enough that everyone can understand why we should spend time solving it and maintaining a solution

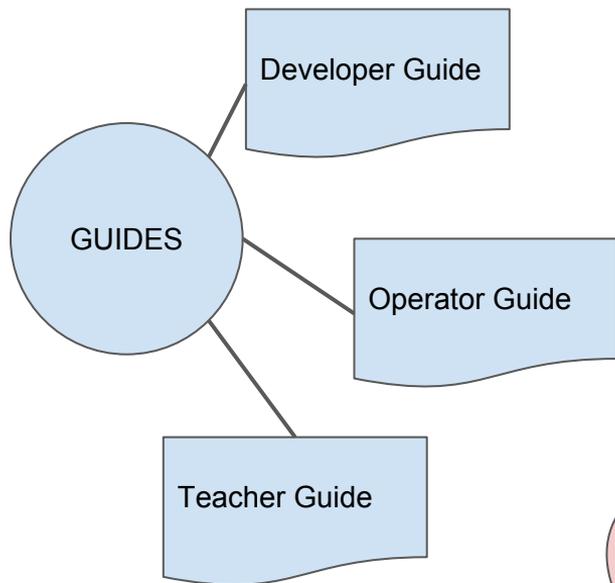*The motivation helps reviewers and participants decide if they want to prioritize this work*

kubernetes

Artifacts

GUIDES
- Developer Guide
- Operator Guide
- Teacher Guide

GRADUATION CRITERIA

Experience Reports

Random boats for no reason.

➜ **`draft`**:
   - ◆ The KEP has been authored but not reviewed

➜ **`provisional`**:
   - ◆ Proposed and actively being defined. The owning SIG has accepted that this is work that needs to be done.

➜ **`implementable`**:
   - ◆ The approvers have approved this KEP for implementation.

Other KEP End States

➔ *deferred*:
  ◆ The KEP is proposed but not actively being worked on.

➔ *rejected*:
  ◆ The approvers and authors have decided that this KEP is not moving forward. The KEP is kept around as a historical document.

➔ *withdrawn*:
  ◆ The KEP has been withdrawn by the authors.

➔ *replaced*:
  ◆ The KEP has been replaced by a new KEP. The `superseded-by` metadata value should point to the new KEP.

INIT

KEP AUTHOR ACTION

- Generates templates
- Creates initial metadata
- Gives a chance to see if the cli is set up properly

PROPOSE

KEP AUTHOR ACTION

KEP STATE: DRAFT

- Submitter has populated the templates
- Motivation is clear
- Preps for SIG reviewers

ACCEPT

SIG APPROVER ACTION

KEP STATE: PROVISIONAL

- SIG has reviewed and commits to shepherding it towards implementation
- Not tied to a release yet
- Not ready to have PRs associated yet

PLAN

KEP AUTHOR ACTION

KEP STATE: PROVISIONAL

- KEP author(s) iterate on submitted documentation, getting it ready to be worked by the SIG
- May be a lengthy period as artifacts are developed as early as possible
- May feed into API review process next

APPROVE

SIG APPROVER ACTION
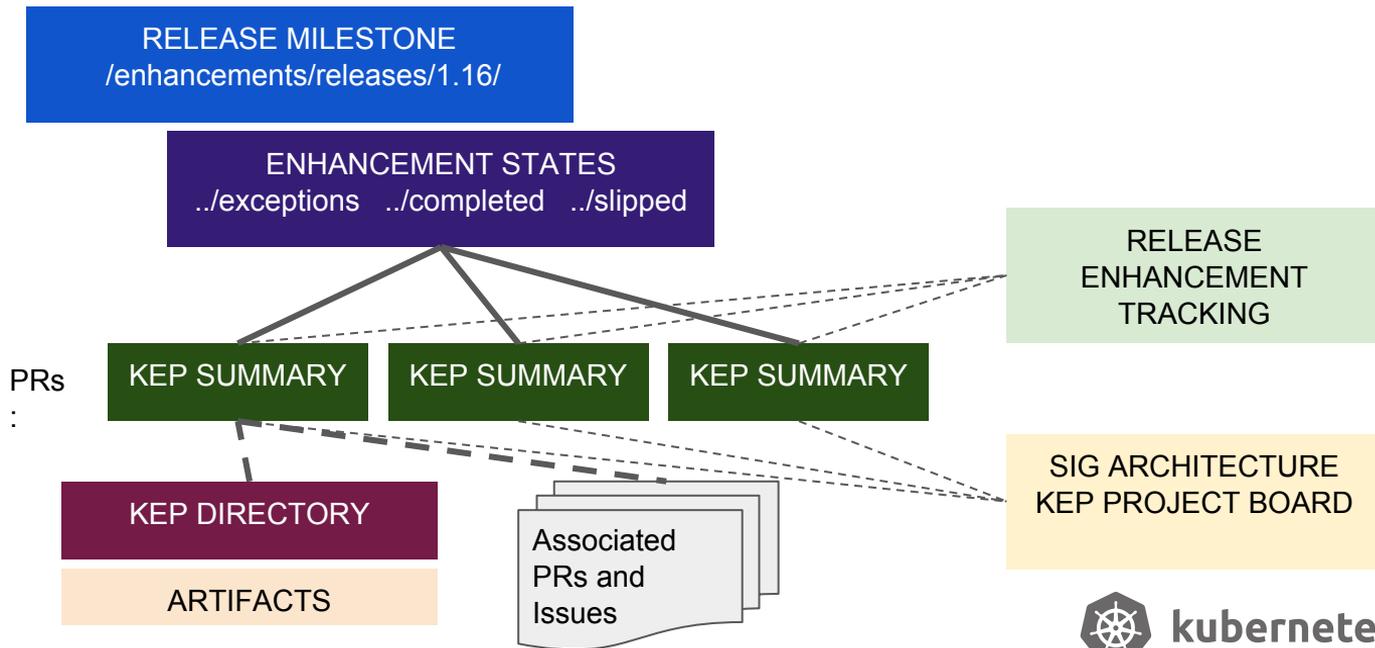
KEP STATE: IMPLEMENTABLE

- KEP reviewer/approver is satisfied that the contents of the KEP are in alignment with the SIG, such that coding and other activities may proceed. It is now a trusted document of the SIG.
- API reviews should be complete

Future States

Release Enhancements Management

RELEASE MILESTONE
/enhancements/releases/1.16/

ENHANCEMENT STATES
../exceptions   ../completed   ../slipped

PRs
:

KEP SUMMARY   **KEP SUMMARY**   **KEP SUMMARY**

KEP DIRECTORY

ARTIFACTS

Associated
PRs and
Issues

RELEASE ENHANCEMENT TRACKING

SIG ARCHITECTURE KEP PROJECT BOARD

Been too long since we've seen a ship. I know this is about the future, and that is an old ship, sorry.

The Elusive Roadmap...

2019

1.14	1.15	1.16	1.17

KEP SUMMARY

KEP SUMMARY

KEP SUMMARY

KEP SUMMARY

The better planned the KEPs, the clearer our roadmap as a project!

kubernetes

Plans for upcoming cycles

- Revamp the SIG PM Charter
- Clean up process documentation
    i. Who owns what (DACI / RACI)
    ii. Glossary of terms
    iii. KEP Workflow (entry / exit criteria for states)
- Migrations!
    i. Design proposals –> k/enhancements (maybe?)
    ii. Architecture design & diagrams –> k/enhancements (maybe?)
- KEP CLI tool (generate / approve / transition stages)
- KEPs on Contributor Site
- KEPs to GA in 1.15/1.16 (tentative)
- Replace the Features / Enhancements tracking sheets (with literally anything else)
- Cross-SIG project tracking
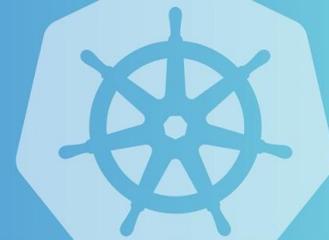   i. Release Planning events
    ii. Roadmaps

Related KEPs

Read / understand / suggest changes!

- KEP Template: https://git.k8s.io/enhancements/keps/0000-kep-template.md
- KEP Process: https://git.k8s.io/enhancements/keps/0001-kubernetes-enhancement-proposal-process.md
- KEP Implementation: https://git.k8s.io/enhancements/keps/0001a-meta-kep-implementation.md
- Contributor Site discussion: https://github.com/kubernetes/community/issues/1819

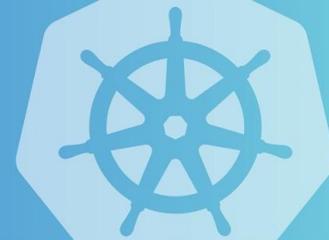# How to get involved!

- Chairs
  - Aparna Sinha, Google
  - Ihor Dvoretskyi, CNCF
  - Caleb Miles, Google
  - Program Management & Release
    - Jaice Singer DuMars, Google
  - Product Management
    - Stephen Augustus, Red Hat
    - Dustin Kirkland, Google
- Home page: https://git.k8s.io/community/sig-pm/
- Slack channel: https://kubernetes.slack.com/messages/sig-pm
- List: https://groups.google.com/forum/#!forum/kubernetes-pm

kubernetes

Questions?

Find us on the internet!

Ihor Dvoretskyi

Developer Advocate, CNCF

Twitter: @idvoretskyi
Slack: @ihor.dvoretskyi
Other social media: http://idvoretskyi.com/

Stephen Augustus

Specialist Solution Architect - OpenShift Tiger Team, Red Hat

Twitter: @stephenaugustus
Slack: @justaugustus
Other social media: https://just.agst.us/

kubernetes