



KubeCon



CloudNativeCon

North America 2019

# Intro: Kubernetes SIG Apps

*Adnan Abdulhusein*  
*Matt Farina*



# SIG Chairs



KubeCon



CloudNativeCon

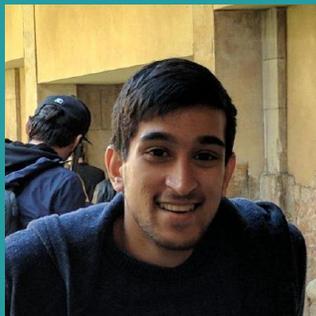
North America 2019



**Matt Farina**  
Samsung SDS  
@mattfarina



**Ken Owens**  
Brex  
@kow3ns



**Adnan Abdulhusein**  
VMware  
@prydonius



**Janet Kuo**  
Google  
@janetkuo



KubeCon



CloudNativeCon

North America 2019



# In The Beginning



KubeCon



CloudNativeCon

North America 2019



Why GitHub? ▾ Enterprise Explore ▾ Marketplace Pricing ▾

Search



Sign in

Sign up

kubernetes / kubernetes

Watch 3.1k

Star 59.7k

Fork 21k

Code

Issues 2,273

Pull requests 1,092

Projects 9

Security

Insights

Branch: release-1.0 ▾

kubernetes / pkg / controller /

Create new file

Find file

History

This branch is 544 commits ahead, 70882 commits behind master.

Pull request

Compare



mikedanese run gofmt on everything we touched

Latest commit 0496a2f on Aug 17, 2015

..

|                                |  |             |
|--------------------------------|--|-------------|
| framework                      | rewrite go imports                         | 4 years ago |
| controller_utils.go            | run gofmt on everything we touched         | 4 years ago |
| doc.go                         | Make copyright ownership statement generic | 5 years ago |
| replication_controller.go      | run gofmt on everything we touched         | 4 years ago |
| replication_controller_test.go | rewrite go imports                         | 4 years ago |



# Way Too Early Helm



KubeCon



CloudNativeCon

North America 2019



Why GitHub? ▾ Enterprise Explore ▾ Marketplace Pricing ▾

Search



Sign in

Sign up

This repository has been archived by the owner. It is now read-only.

helm / helm-classic Archived

Watch 31

Star 583

Fork 54

Code

Issues 42

Pull requests 0

Projects 0

Security

Insights

(OBSOLETE) Helm Classic v1 <https://github.com/helm/helm>

158 commits

8 branches

12 releases

26 contributors

View license

Tag: 0.0.1 ▾

Find file

Clone or download ▾



technosophos Merge pull request #120 from technosophos/feat/git-logger ...

✓ Latest commit da7874b on Nov 2, 2015



ci

feat(ci): adjust include/upload patterns

4 years ago



docs

Merge pull request #59 from gabrtv/publish

4 years ago



helm

Merge pull request #120 from technosophos/feat/git-logger

4 years ago



skel

doc(skel) make the skeleton Chart generic

4 years ago



test

feat(test): add ci test infrastructure scaffolding

4 years ago



.gitignore

feat(ci): add publishing of binaries

4 years ago



.travis.yml

test(travis): speed up glide pkg handling

4 years ago

# SIG Apps Is Born



KubeCon



CloudNativeCon

North America 2019



mattfarina Adding SIG Apps

dce5eb4 on May 11, 2016

1 contributor

22 lines (17 sloc) | 928 Bytes

Raw

Blame

History



## SIG Apps

A Special Interest Group for deploying and operating applications in Kubernetes.

## Goals:

- Discuss running applications in k8s
- Discuss how to define and run apps in k8s (APIs, CLIs, SDKs, package management tools, etc.)
- Suggest k8s features where we see friction
- Be the voice of the people running applications into the k8s development (developers and devops)
- Help people get involved in the kubernetes community
- Show early features/demos of tools that make running apps easier

# SIG Apps Today



KubeCon



CloudNativeCon

North America 2019

## SIG Apps Charter

---

This charter adheres to the conventions described in the [Kubernetes Charter README](#) and uses the Roles and Organization Management outlined in [sig-governance](#).

### Scope

---

SIG Apps covers developing, deploying, and operating applications on Kubernetes with a focus on the application developer and application operator experience.

### In scope

#### Code, Binaries and Services

- APIs used for running applications (e.g., Workloads API)
- Tools and documentation to aid in ecosystem tool interoperability around apps (e.g., Application CRD/Controller)
- Grandfathered in tools used to aid in development of and management of workloads (e.g., Kompose)

#### Cross-cutting and Externally Facing Processes

- A discussion platform for solving app development and management problems
- Represent the needs and persona of application developers and operators

# Personas / Roles



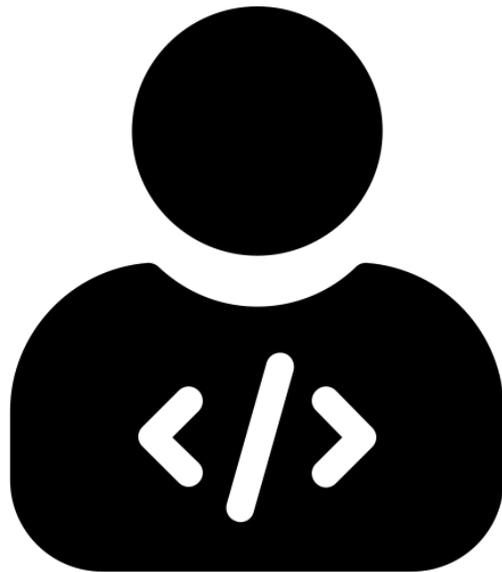
KubeCon



CloudNativeCon

North America 2019

## Application Developer



Created by Shastry  
from Noun Project

## Application Operator



Created by Adrien Coquet  
from Noun Project

# Some Core Controllers...

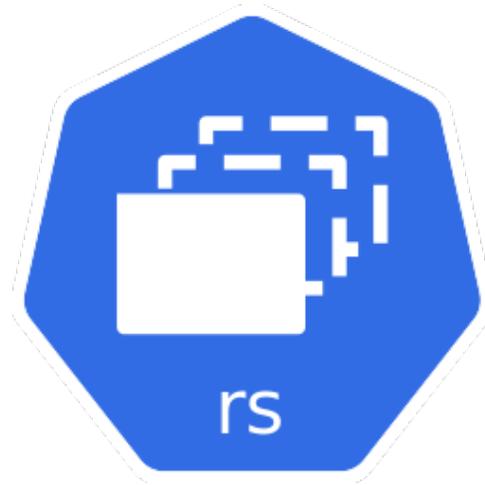


KubeCon



CloudNativeCon

North America 2019



And more...

# Application CRD/Controller



KubeCon



CloudNativeCon

North America 2019

Why GitHub? Enterprise Explore Marketplace Pricing Search Sign in Sign up

kubernetes-sigs / application Watch 40 Star 264 Fork 75

Code Issues 11 Pull requests 8 Projects 0 Security Insights

### Application metadata descriptor CRD

k8s-sig-apps

129 commits 2 branches 0 packages 1 release 21 contributors Apache-2.0

Branch: master New pull request Find file Clone or download

k8s-ci-robot Merge pull request #126 from janetkuo/patch-1 Latest commit f15a09f 3 days ago

|                 |  |               |
|-----------------|--|---------------|
| cmd/manager     | add comment for reconcile time   | 9 months ago  |
| config          | Merge pull request #103 from barney-s/issue_102                          | 2 months ago  |
| docs            | Merge pull request #56 from tossmilestone/fix-example-error              | last year     |
| e2e             | WIP: Upgrade to version 1.0.5 of Kubebuilder                             | last year     |
| hack            | WIP: Upgrade to version 1.0.5 of Kubebuilder                             | last year     |
| pkg             | Merge pull request #106 from barney-s/issue_105                          | 2 months ago  |
| vendor          | Absorb vendor/kubesdk into application/pkg. Since kubesdk is vendored... | 10 months ago |
| .gitignore      | WIP: Upgrade to version 1.0.5 of Kubebuilder                             | last year     |
| CONTRIBUTING.md | Upgrade resource to v1beta1  | last year     |

# Old Process For New To Core

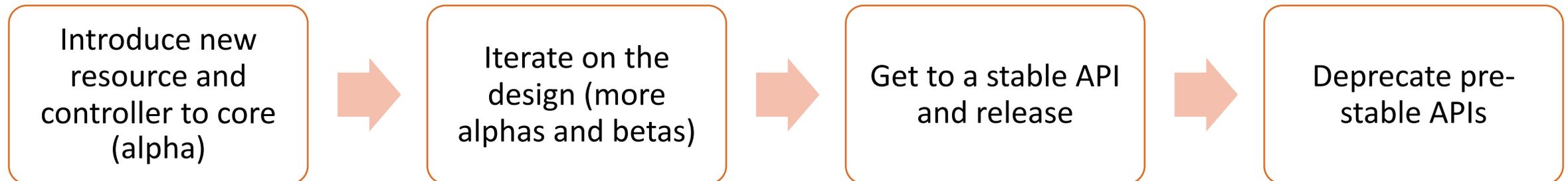


KubeCon



CloudNativeCon

North America 2019



# Current Process

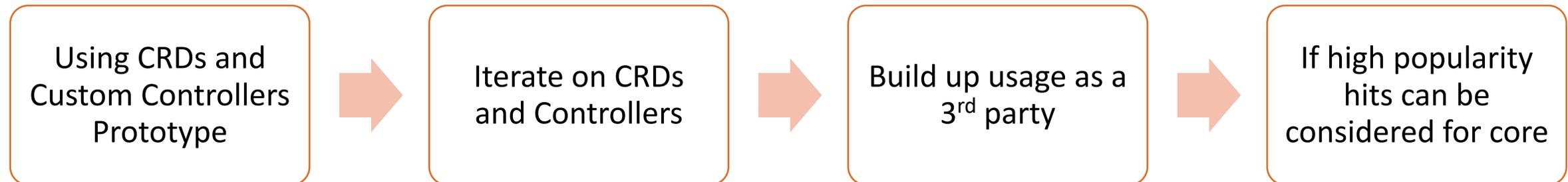


KubeCon



CloudNativeCon

North America 2019



\* This process can happen as a SIG sponsored project like the Application CRD / Controller

# Execution Hook



KubeCon



CloudNativeCon

North America 2019

## Summary

---

This proposal is to introduce an API (ExecutionHook) for dynamically executing user's commands in a pod/container or a group of pods/containers and a controller (ExecutionHookController) to manage the hook lifecycle. ExecutionHook provides a general mechanism for users to trigger hook commands in their containers for their different use cases. Different options have been evaluated to decide how this ExecutionHook should be managed and executed. The preferred option is described in the Proposal section. The other options are discussed in the Alternatives section.

## Motivation

---

The volume snapshot feature allows creating/deleting volume snapshots, and the ability to create new volumes from a snapshot natively using the Kubernetes API. However, application consistency is not guaranteed. An user has to figure out how to quiece an application before taking a snapshot and unquiece it after taking the snapshot.

So we want to introduce an `ExecutionHook` to facilitate the quiesce and unquiesce actions when taking a snapshot. There is an existing lifecycle hook in the `Container` struct. The lifecycle hook is called immediately after a container is created or immediately before a container is terminated. The proposed execution hook is not tied to the start or termination time of the container. It can be triggered on demand by callers (users or controllers) and the status will be updated dynamically.

\* Notice: This is very new

# Documentation



KubeCon



CloudNativeCon

North America 2019

kubernetes Documentation Blog Partners Community Case Studies English ^ v1.16 ^

## Concepts

HOME GETTING STARTED **CONCEPTS** TASKS TUTORIALS REFERENCE CONTRIBUTE

### Concepts

- ▶ Overview
- ▶ Cluster Architecture
- ▶ Containers
- ▼ Workloads
  - ▶ Pods
  - ▼ Controllers
    - ReplicaSet
    - ReplicationController
    - Deployments**
    - StatefulSets
    - DaemonSet
    - Garbage Collection
    - TTL Controller for Finished Resources
    - Jobs - Run to Completion
    - CronJob
- ▶ Services, Load Balancing, and Networking
- ▶ Storage
- ▶ Configuration
- ▶ Security
- ▶ Policies
- ▶ Scheduling
- ▶ Cluster Administration
- ▶ Extending Kubernetes

## Deployments

A *Deployment* provides declarative updates for *Pods* and *ReplicaSets*.

You describe a *desired state* in a *Deployment*, and the *Deployment Controller* changes the actual state to the desired state at a controlled rate. You can define *Deployments* to create new *ReplicaSets*, or to remove existing *Deployments* and adopt all their resources with new *Deployments*.

**Note:** Do not manage *ReplicaSets* owned by a *Deployment*. Consider opening an issue in the main Kubernetes repository if your use case is not covered below.

- [Use Case](#)
- [Creating a Deployment](#)
- [Updating a Deployment](#)
- [Rolling Back a Deployment](#)
- [Scaling a Deployment](#)
- [Pausing and Resuming a Deployment](#)
- [Deployment status](#)
- [Clean up Policy](#)
- [Canary Deployment](#)
- [Writing a Deployment Spec](#)
- [Alternative to Deployments](#)

## Use Case

The following are typical use cases for *Deployments*:

- [Create a Deployment to rollout a ReplicaSet](#). The *ReplicaSet* creates *Pods* in the background. Check the status of the rollout to see if it succeeds or not.
- [Declare the new state of the Pods](#) by updating the *PodTemplateSpec* of the *Deployment*. A new *ReplicaSet* is created and the *Deployment* manages moving the *Pods* from the old *ReplicaSet* to the new one at a controlled rate. Each new *ReplicaSet* updates the revision of the *Deployment*.

kubernetes Documentation Blog Partners Community Case Studies English ^ v1.16 ^

## Tutorials

HOME GETTING STARTED CONCEPTS TASKS **TUTORIALS** REFERENCE CONTRIBUTE

### Tutorials

- Hello Minikube
- ▶ Learn Kubernetes Basics
- ▶ Online Training Courses
- ▶ Configuration
- ▼ Stateless Applications
  - Exposing an External IP Address to Access an Application in a Cluster
  - Example: Deploying PHP Guestbook application with Redis**
  - Example: Add logging and metrics to the PHP / Redis Guestbook example
- ▶ Stateful Applications
- ▶ Clusters
- ▶ Services

## Example: Deploying PHP Guestbook application with Redis

This tutorial shows you how to build and deploy a simple, multi-tier web application using Kubernetes and Docker. This example consists of the following components:

- A single-instance *Redis* master to store guestbook entries
- Multiple *replicated Redis* instances to serve reads
- Multiple web frontend instances

- [Objectives](#)
- [Before you begin](#)
- [Start up the Redis Master](#)
- [Start up the Redis Slaves](#)
- [Set up and Expose the Guestbook Frontend](#)
- [Scale the Web Frontend](#)
- [Cleaning up](#)
- [What's next](#)

## Objectives

- Start up a *Redis* master.
- Start up *Redis* slaves.
- Start up the guestbook frontend.
- Expose and view the Frontend Service.
- Clean up.

## Before you begin

You need to have a Kubernetes cluster, and the *kubectl* command-line tool must be configured

# KEPs



KubeCon



CloudNativeCon

North America 2019

- “A Kubernetes Enhancement Proposal (KEP) is a way to propose, communicate and coordinate on new efforts for the Kubernetes project.”
- More details at <https://github.com/kubernetes/enhancements/tree/master/keps>

| 11 Open | 23 Closed  | Author | Labels   | Projects  | Milestones | Reviews | Assignee | Sort |
|---------|--|--------|--|---|------------|---------|----------|------|
|         | Proposal to standardize status for core resources            |        | needs-ok-to-test sig/apps size/XL  | cncf-cla: yes kind/kep                            |            |         |          | 2    |
|         | Add KEP for new cronjob concurrency policy: CatchUp          |        | lifecycle/stale ok-to-test sig/apps size/L   | cncf-cla: yes kind/kep                            |            |         |          | 23   |
|         | [WIP] Enhance daemonset on rollingupdate.                    |        | do-not-merge/work-in-progress kind/kep sig/apps size/L                                   | cncf-cla: yes                                     |            |         |          | 26   |
|         | Supports PreSidecars and PostSidecars                        |        |  | cncf-cla: yes kind/kep sig/apps size/M            |            |         |          | 5    |
|         | implement job allocation completions index enhancement       |        | needs-ok-to-test sig/apps size/M   | cncf-cla: yes kind/kep                            |            |         |          | 17   |
|         | Added a KEP for the Stateful Application Data Management API |        | kind/kep lifecycle/stale sig/apps sig/storage size/XL                                    | cncf-cla: yes                                     |            |         |          | 31   |
|         | KEP for Graduating CronJob to GA                             |        | sig/apps size/L  | cncf-cla: yes kind/kep lifecycle/stale ok-to-test |            |         |          | 41   |
|         | ConfigMap / Secret Orchestration                             |        |  | cncf-cla: yes kind/kep ok-to-test sig/apps size/L |            |         |          | 7    |
|         | Add KEP for application pod graceful update                  |        | ok-to-test sig/apps size/L   | cncf-cla: yes kind/kep lifecycle/stale            |            |         |          | 20   |
|         | KEP for adding ProgressDeadlineSeconds to job                |        | ok-to-test sig/apps size/L   | cncf-cla: yes kind/feature kind/kep               |            |         |          | 26   |
|         | Add StatefulSet Volume Expansion Kep                         |        | do-not-merge/hold kind/api-change kind/feature kind/kep lgtn sig/apps sig/storage size/L | api-review approved cncf-cla: yes                 |            |         |          | 124  |

# How You Can Get Involved

# Attend A Meeting



KubeCon



CloudNativeCon

North America 2019

- Meetings every other Monday at 9am Pacific Time
- Recordings on YouTube
- More information at <https://github.com/kubernetes/community/tree/master/sig-apps>

The screenshot shows a YouTube page for the 'SIG Apps Meetings' playlist. The playlist contains 11 videos, each with a thumbnail, title, duration, and uploader information. The videos are:

| Video Number | Title                        | Duration | Uploader           | Added by           |
|--------------|------------------------------|----------|--------------------|--------------------|
| 1            | Kubernetes SIG Apps 20190923 | 56:10    | Janet Kuo          | Janet Kuo          |
| 2            | Kubernetes SIG Apps 20190812 | 35:37    | Janet Kuo          | Janet Kuo          |
| 3            | Kubernetes SIG Apps 20190729 | 25:44    | Matt Farina        | Matthew Farina     |
| 4            | Kubernetes SIG Apps 20190722 | 15:14    | Matt Farina        | Matthew Farina     |
| 5            | Kubernetes SIG Apps 20190715 | 57:23    | Matt Farina        | Matthew Farina     |
| 6            | Kubernetes SIG Apps 20190701 | 30:24    | Adnan Abdulhussein | Adnan Abdulhussein |
| 7            | Kubernetes SIG Apps 20190624 | 22:34    | Janet Kuo          | Janet Kuo          |
| 8            | Kubernetes SIG Apps 20190617 | 25:37    | Janet Kuo          | Janet Kuo          |
| 9            | Kubernetes SIG Apps 20190603 | 50:54    | Matt Farina        | Matthew Farina     |
| 10           | Kubernetes SIG Apps 20190513 | 58:30    | Matt Farina        | Matthew Farina     |
| 11           | Kubernetes SIG Apps 20190506 | 14:06    | Matt Farina        | Matthew Farina     |

# Contribute Documentation



KubeCon



CloudNativeCon

North America 2019

kubernetes Documentation Blog Partners Community Case Studies English ^ v1.16 ^

## Concepts

HOME GETTING STARTED **CONCEPTS** TASKS TUTORIALS REFERENCE CONTRIBUTE

### Concepts

- Overview
- Cluster Architecture
- Containers
- Workloads
  - Pods
  - Controllers
    - ReplicaSet
    - ReplicationController
    - Deployments**
    - StatefulSets
    - DaemonSet
    - Garbage Collection
    - TTL Controller for Finished Resources
    - Jobs - Run to Completion
    - CronJob
- Services, Load Balancing, and Networking
- Storage
- Configuration
- Security
- Policies
- Scheduling
- Cluster Administration
- Extending Kubernetes

## Deployments

A *Deployment* provides declarative updates for *Pods* and *ReplicaSets*.

You describe a *desired state* in a *Deployment*, and the *Deployment Controller* changes the actual state to the desired state at a controlled rate. You can define *Deployments* to create new *ReplicaSets*, or to remove existing *Deployments* and adopt all their resources with new *Deployments*.

**Note:** Do not manage *ReplicaSets* owned by a *Deployment*. Consider opening an issue in the main Kubernetes repository if your use case is not covered below.

- Use Case
- Creating a Deployment
- Updating a Deployment
- Rolling Back a Deployment
- Scaling a Deployment
- Pausing and Resuming a Deployment
- Deployment status
- Clean up Policy
- Canary Deployment
- Writing a Deployment Spec
- Alternative to Deployments

## Use Case

The following are typical use cases for *Deployments*:

- Create a *Deployment* to rollout a *ReplicaSet*. The *ReplicaSet* creates *Pods* in the background. Check the status of the rollout to see if it succeeds or not.
- Declare the new state of the *Pods* by updating the *PodTemplateSpec* of the *Deployment*. A new *ReplicaSet* is created and the *Deployment* manages moving the *Pods* from the old *ReplicaSet* to the new one at a controlled rate. Each new *ReplicaSet* updates the revision of the *Deployment*.

No controller intro



Documentation Blog Partners Community Case Studies English ^ v1.16 ^

## Creating the Guestbook Frontend Deployment

```
application/guestbook/frontend-deployment.yaml
apiVersion: apps/v1 # for versions before 1.9.0 use apps/v1beta2
kind: Deployment
metadata:
  name: frontend
  labels:
    app: guestbook
spec:
  selector:
    matchLabels:
      app: guestbook
      tier: frontend
  replicas: 3
  template:
    metadata:
      labels:
        app: guestbook
        tier: frontend
    spec:
      containers:
        - name: php-redis
          image: gcr.io/google-samples/gb-frontend:v4
          resources:
            requests:
              cpu: 100m
              memory: 100Mi
          env:
            - name: GET_HOSTS_FROM
              value: dns
              # Using 'GET_HOSTS_FROM=dns' requires your cluster to
              # provide a dns service. As of Kubernetes 1.3, DNS is a built-in
              # service launched automatically. However, if the cluster you are using
              # does not have a built-in DNS service, you can instead
              # access an environment variable to find the master
              # service's host. To do so, comment out the 'value: dns' line above,
              # uncomment the line below:
              # value: env
      ports:
        - containerPort: 80
```

Not Recommended Labels

3 1/2 year old image version

1. Apply the frontend Deployment from the `frontend-deployment.yaml` file:

```
kubectl apply -f https://k8s.io/examples/application/guestbook/frontend-deployment.yaml
```

2. Query the list of *Pods* to verify that the three frontend replicas are running:

# Code!



KubeCon



CloudNativeCon

North America 2019

- There's a lot of opportunity to code
- Bugs need to be fixed
- CronJob needs a stable release (e.g., needs to switch to shared informers)
- Application CRD/Controller and Execution Hook need work

The screenshot shows the GitHub interface for the `kubernetes/kubernetes` repository. The file `pkg/controller/cronjob/cronjob_controller.go` is open, showing a commit by `smarterclayton` titled "Ignore namespace termination errors when creating pods or jobs" from 22 days ago. The code diff shows the following content:

```
392 lines (339 sloc) | 15.2 KB
1  /*
2  Copyright 2016 The Kubernetes Authors.
3
4  Licensed under the Apache License, Version 2.0 (the "License");
5  you may not use this file except in compliance with the License.
6  You may obtain a copy of the License at
7
8      http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the License is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 See the License for the specific language governing permissions and
14 limitations under the License.
15 */
16
17 package cronjob
18
19 /*
20 I did not use watch or expectations. Those add a lot of corner cases, and we aren't
21 expecting a large volume of jobs or scheduledJobs. (We are favoring correctness
22 over scalability. If we find a single controller thread is too slow because
23 there are a lot of Jobs or CronJobs, we can parallelize by Namespace.
24 If we find the load on the API server is too high, we can use a watch and
25 UndeltaStore.)
26
27 Just periodically list jobs and SJs, and then reconcile them.
28
29 */
30
31 import (
32     "context"
33     "fmt"
34     "sort"
35     "time"
36
37     "k8s.io/klog"
38
39     batchv1 "k8s.io/api/batch/v1"
```

# Build More Tools

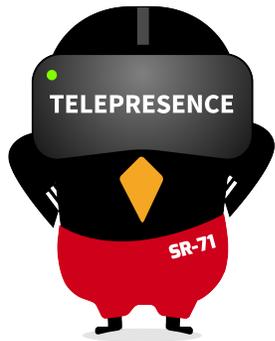


KubeCon



CloudNativeCon

North America 2019



More are needed...

**Questions / Discussion?**