HITACHI
Inspire the Next

System Operator: A Tool for System Management in Kubernetes Clusters

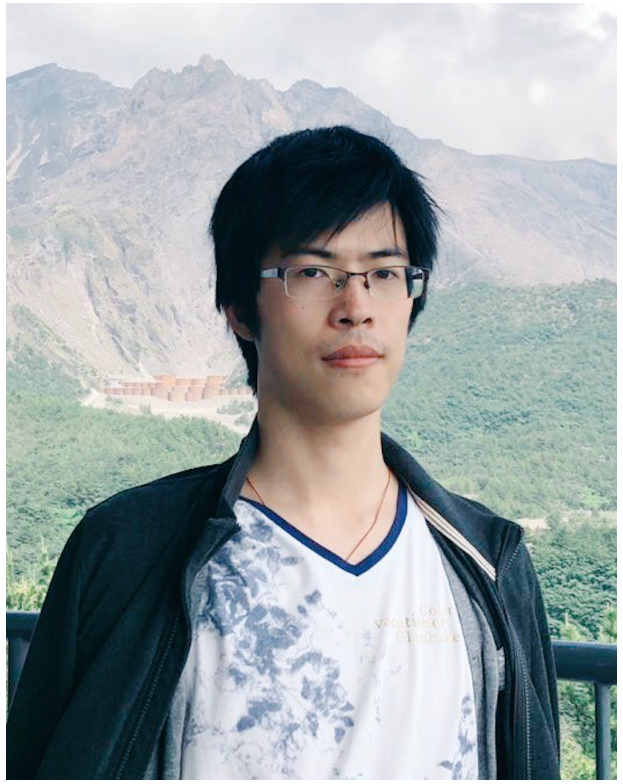
IARIA Cloud Computing 2020

Jiye Yu, Yuki Naganuma, Takaya Ide

Hitachi, Ltd. R&D Group

Email: jiye.yu.kb@hitachi.com

About Me



Jiye Yu

Researcher @ Hitachi, Ltd.

Personal Github Account: JiyeYu

HITACHI
Inspire the Next

Content

1. Operator and System Operator
2. Architecture of System Operator
3. GUI and System Operator Server
4. System Operator Controller
    a. Custom Resource and Secondary Resources
    b. Sysem Regulation

What is Kubernetes

HITACHI
Inspire the Next

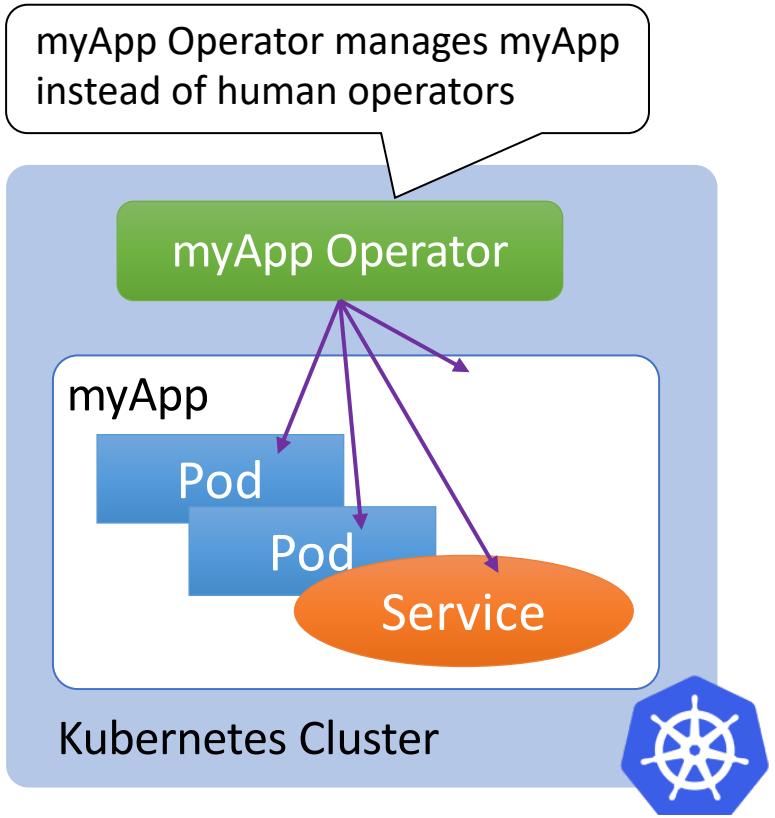
- OSS container orchestrator
- Running one or more applications
- Features
  - ✓ Declarative API
  - ✓ High resilience
  - ✓ Pluggable using Custom Resources

What is a Kubernetes Operator?

HITACHI
Inspire the Next

Issue: It is infeasible for Kubernetes to manage thousands of applications with various working styles

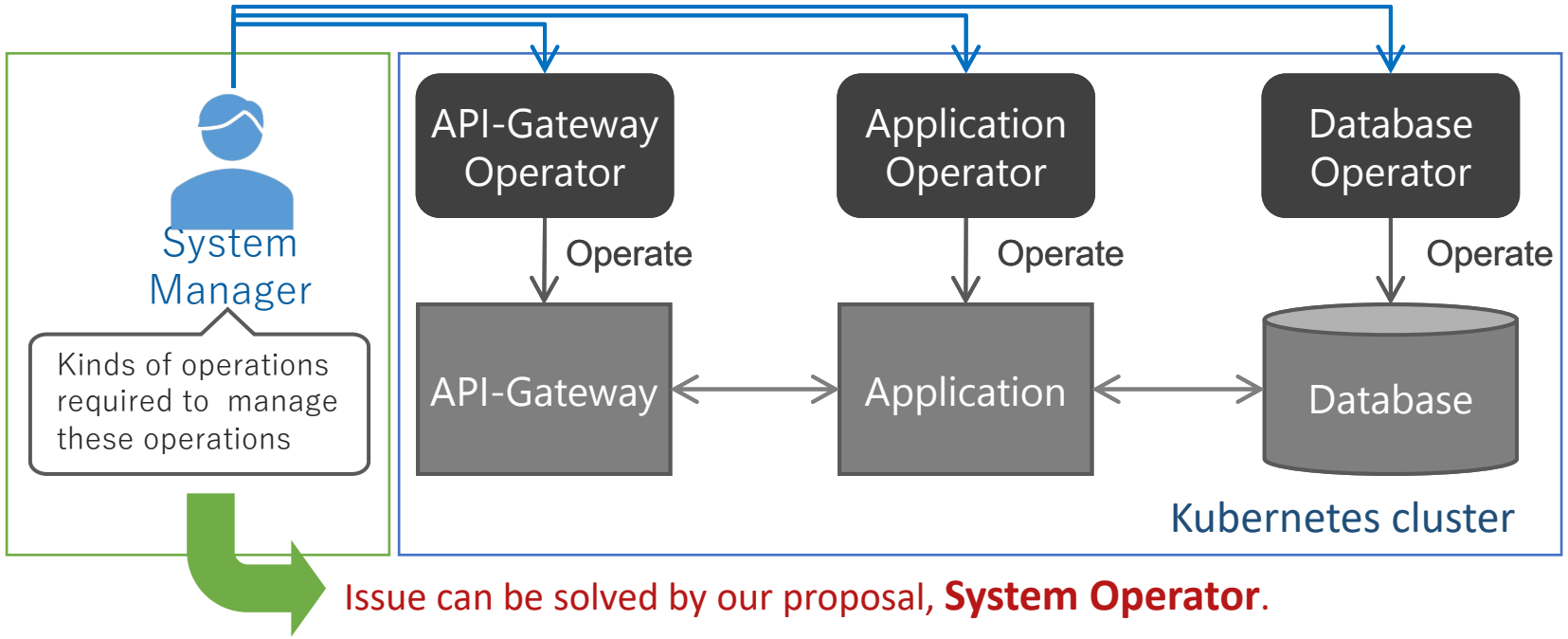
- **Kubernetes Operator** is a solution
  - ✓ Concept of Operator is raised by CoreOS
  - ✓ Operator is designed for packaging, deploying and managing a Kubernetes application automatically
- How Kubernetes Operator works
  - ✓ Human operators who have deep knowledge on specific applications 'teach' Operators to repeat their work patterns on those applications.
- Benefits of Kubernetes Operator
  - ✓ Operators extend Kubernetes functionality
  - ✓ Make migration easier among various Kubernetes clusters
  - ✓ Ecosystem is growing – OperatorHub shares existing Operators to users



Issue when managing a system made of multiple applications

HITACHI
Inspire the Next

- Normally, each App Operator* is developed for a single application.
  However, systems that enterprise uses are normally made of multiple applications.
  -> System Manager needs to manage multiple App Operators and the whole system.
  - ✓ App Operators' deployment/Upgrade, Backup setting, labeling management
  - ✓ Kubernetes Services creating for system's construction



* To distinguish Operator from System Operator, we will use App Operator as Operator in the slides.

Propose of System Operator

HITACHI
Inspire the Next

- System Operator is a tool which can be used to coordinate multiple App Operators, build and operate the entire system made of multiple applications.

- Features Provided*:

  Level1

  ✓ System construction by deploying applications via App Operator
  (Including network connection between applications)
  ✓ App Operator and related application's update

  Level2

  ✓ Deploy and configure a monitoring environment for applications and system
  ✓ Disaster recovery and regular backup

  Level3

  ✓ Auto-scaling to applications

* Take the Capability Level of the Operator which is advocated by Red Hat as a reference.

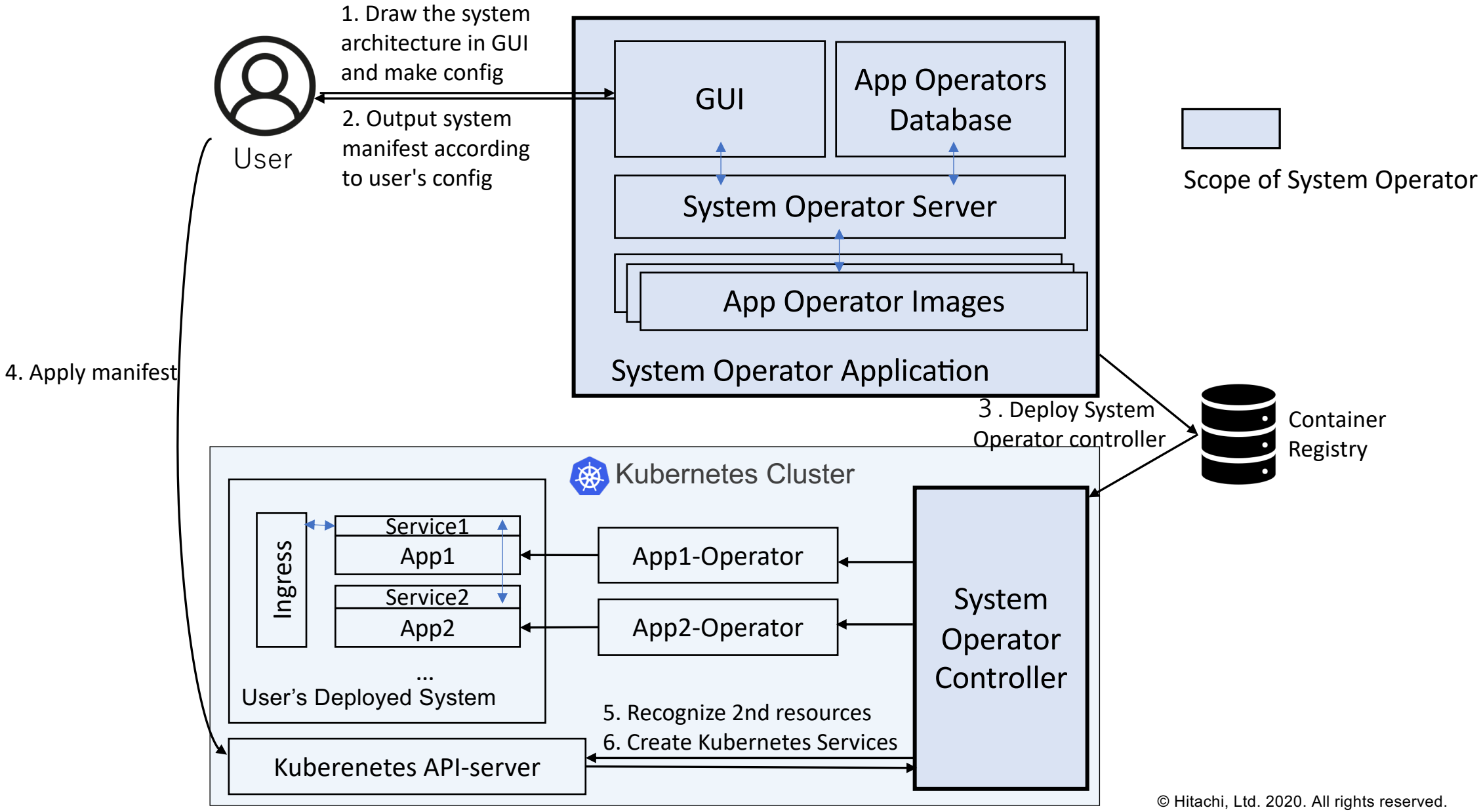
Comparison between related technologies

HITACHI
Inspire the Next

	App Operator	Helm*	System Operator
Single App Deploy	✓	✓	✓
Upgrade App	✓	✓	✓
Auto-scaling	✓	✗	✓
Monitoring	✓	✗	✓
Backup	✓	✗	✓
Entire System Deploy	✗	✓	✓
Entire System Ops (monitoring, backup etc.)	✗	✓	✓
Connect Apps	✗	✓	✓

* Helm is the package manager for Kubernetes. For further information, check here: https://helm.sh/docs/

System Operator's Architecture



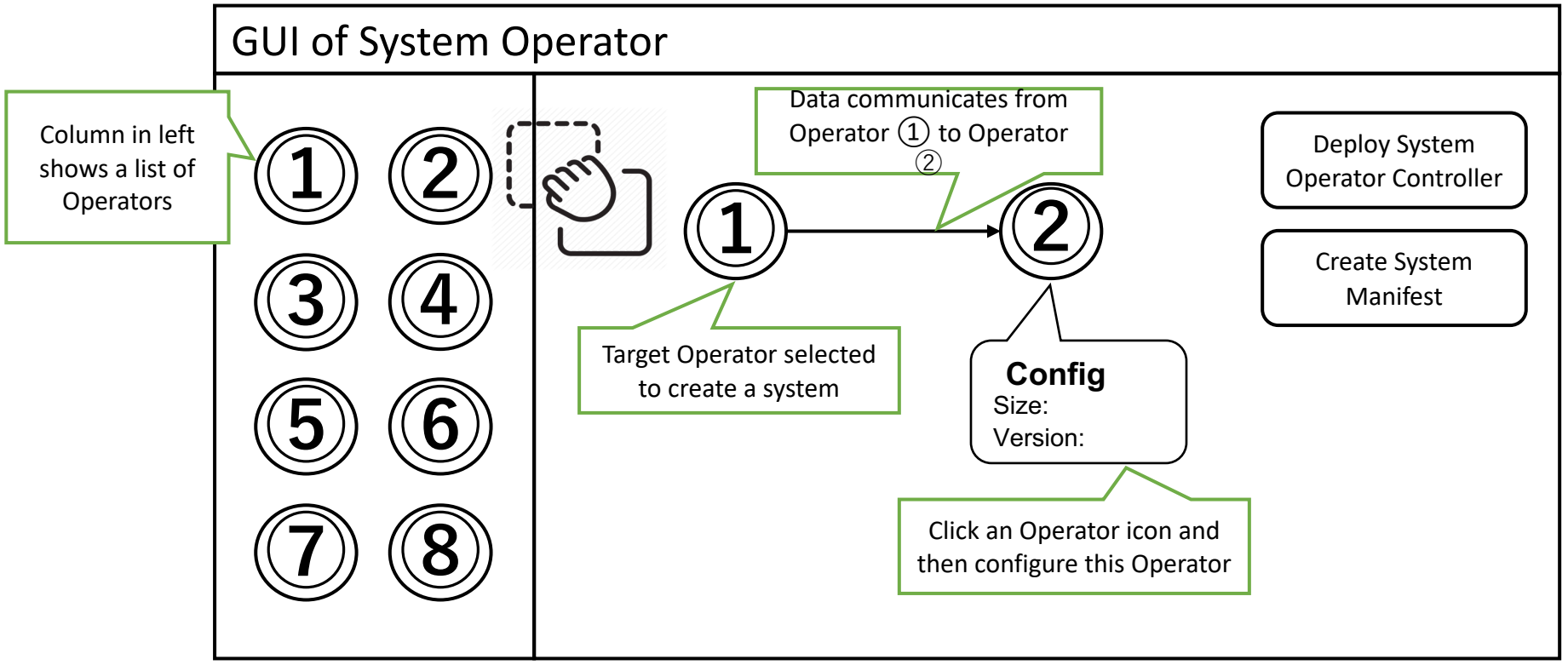
Overview of System Operator components and Features provided

HITACHI
Inspire the Next

Component	Functions	Description
GUI	User Interface	Generate a framework of system by selecting Operators
App Operators Database	Database	Save information of Operators
App Operator Images	Volume Storage	Local storage of images of Operators
System Operator Server	Core Functions	Core operational processing; Create Controller image based on the input from GUI
System Operator Controller	Distributed End Controller	Manage Operators for each target system in Kubernetes cluster

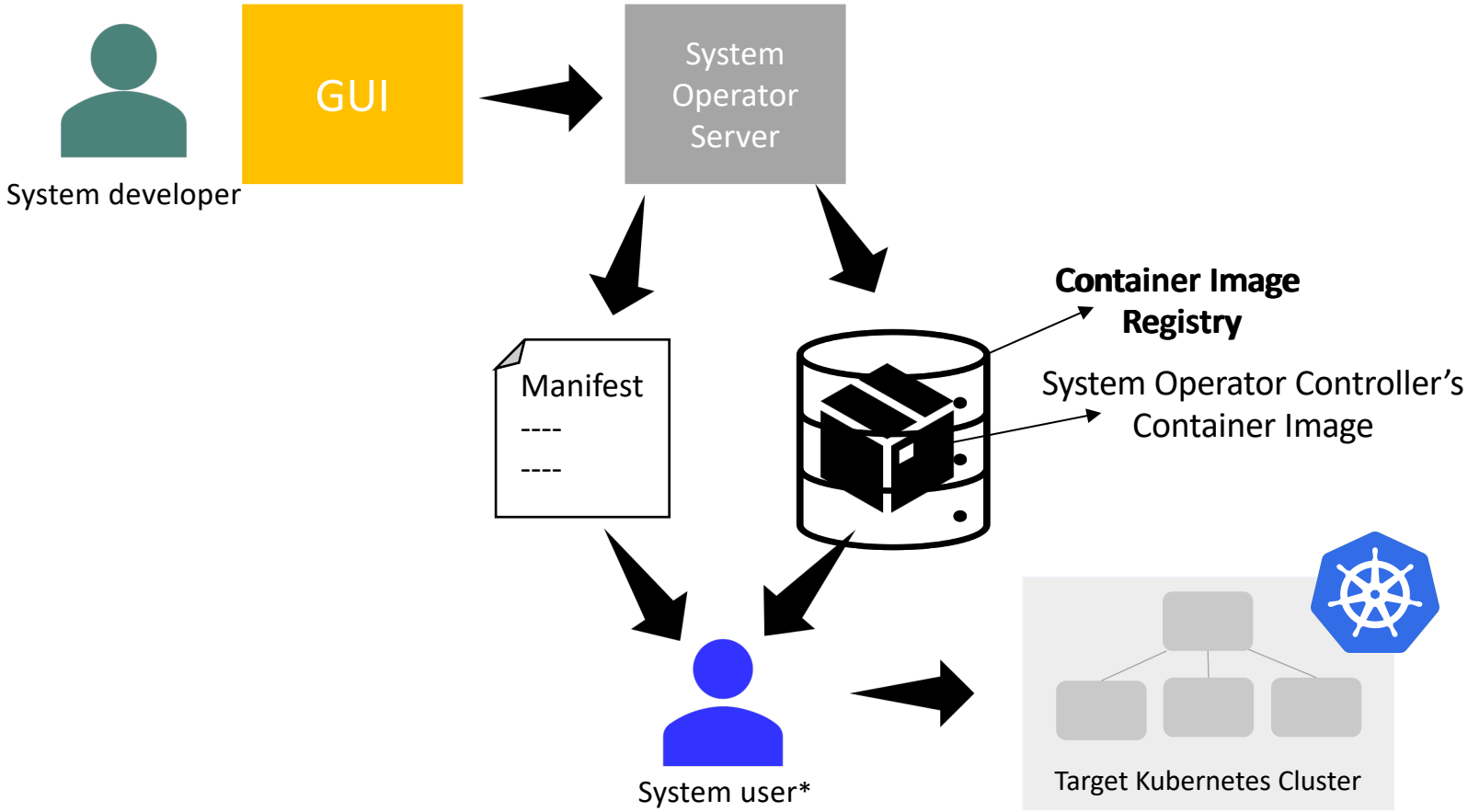
System Operator GUI's design

A GUI is necessary for System Operator to make it easy to use.



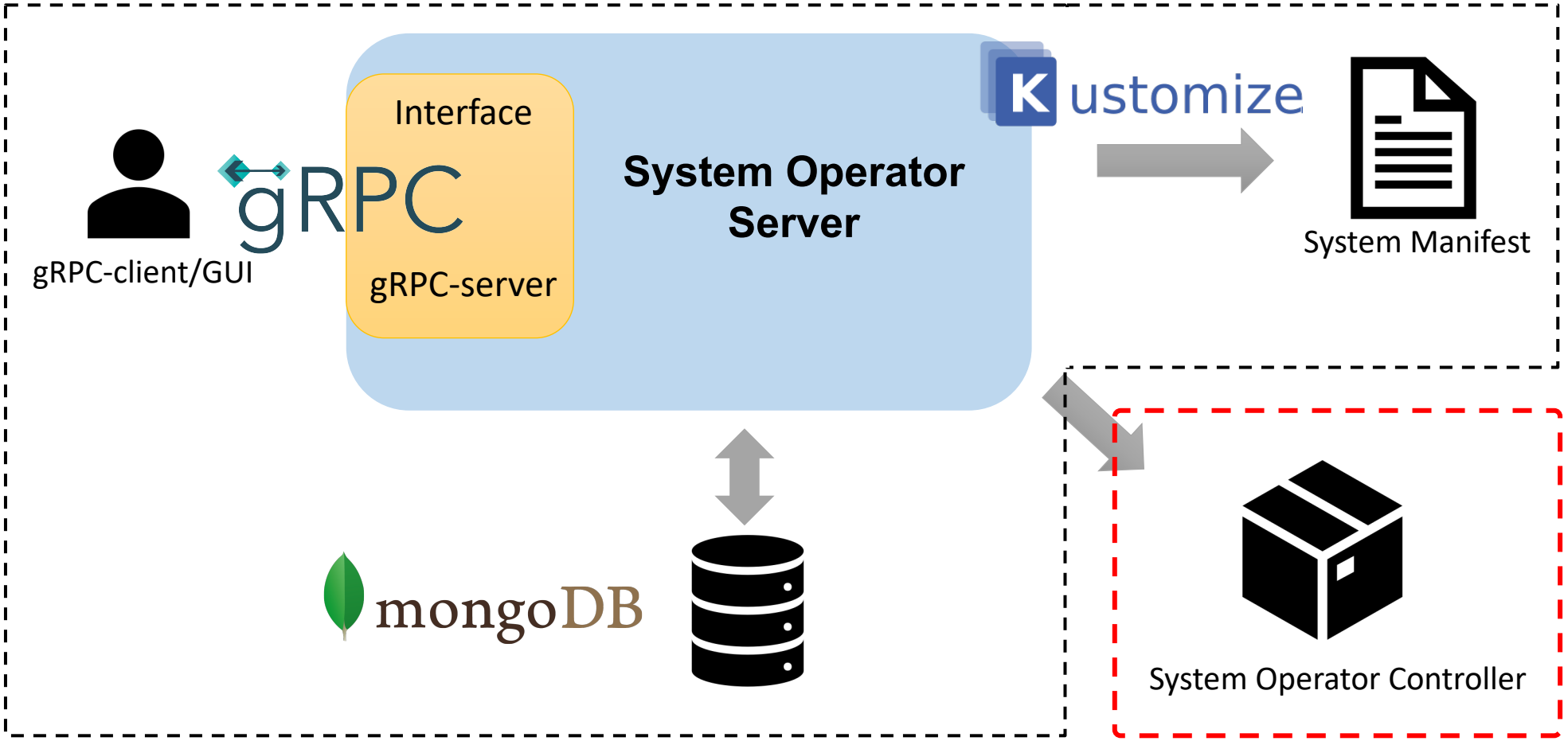
Workflow of System Operator Server

- System Operator Server receives various inputs from GUI and creates the image of System Operator Controller and the manifest for the whole system.



* System developer and system user can be the same or different people here.

Basic Architecture of System Operator Server



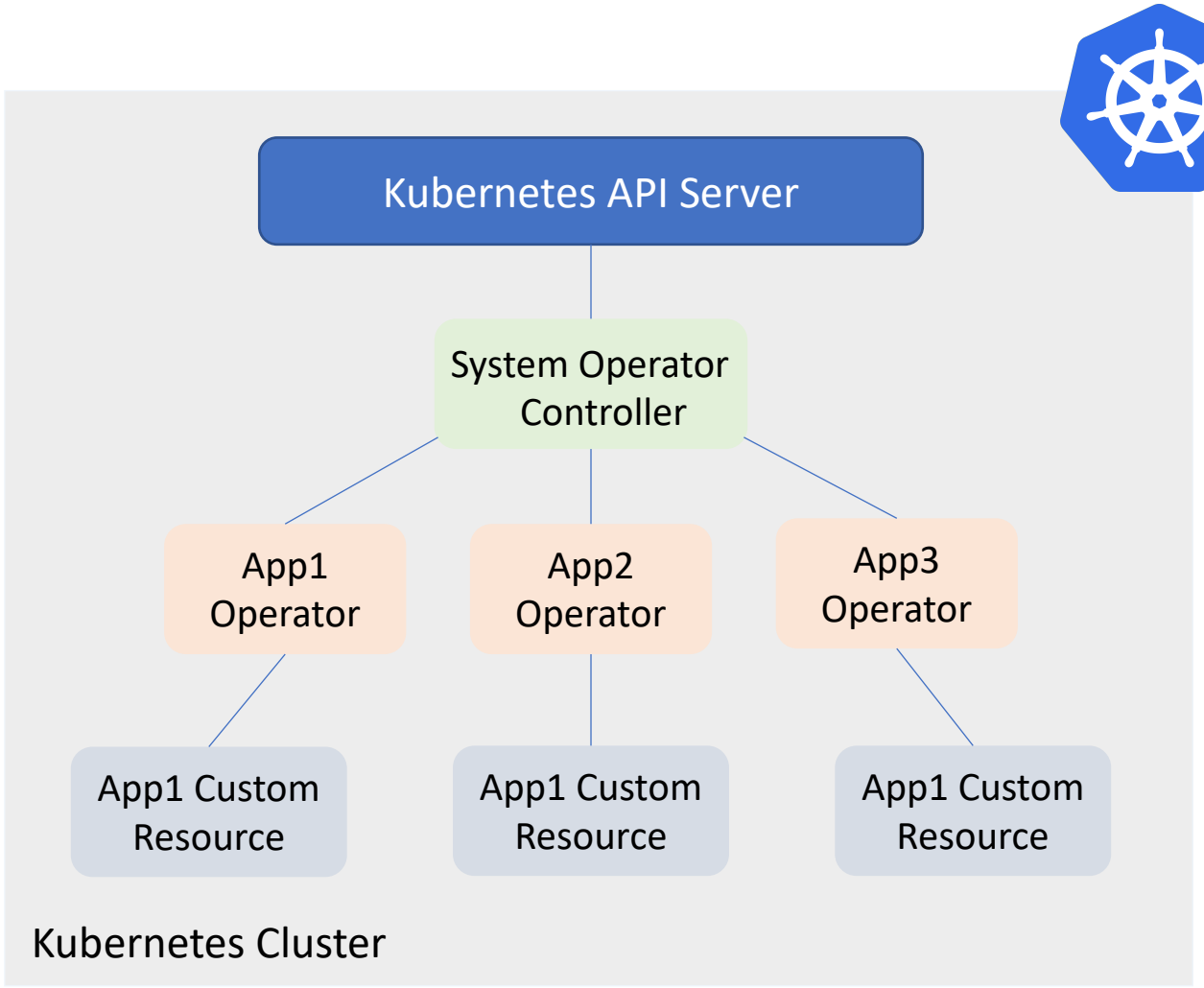
System Operator Controller

HITACHI
Inspire the Next

System Operator Controller is the core part to manage target system in Kubernetes clusters.

Expected features:

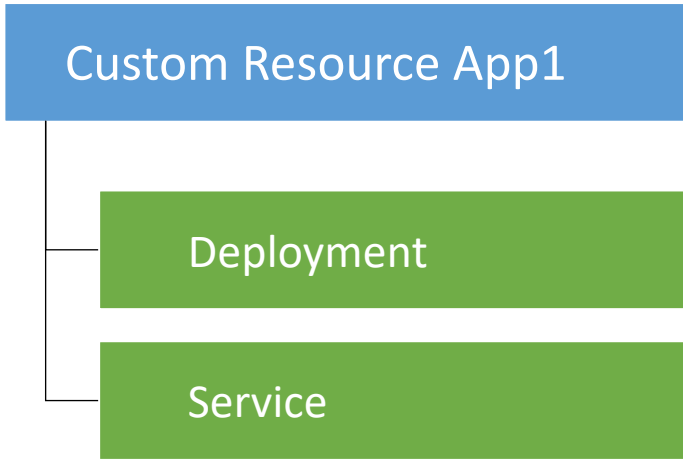
1. Operators Deployment
2. Tracking Status of Custom Resources
3. System Monitoring and Regulation

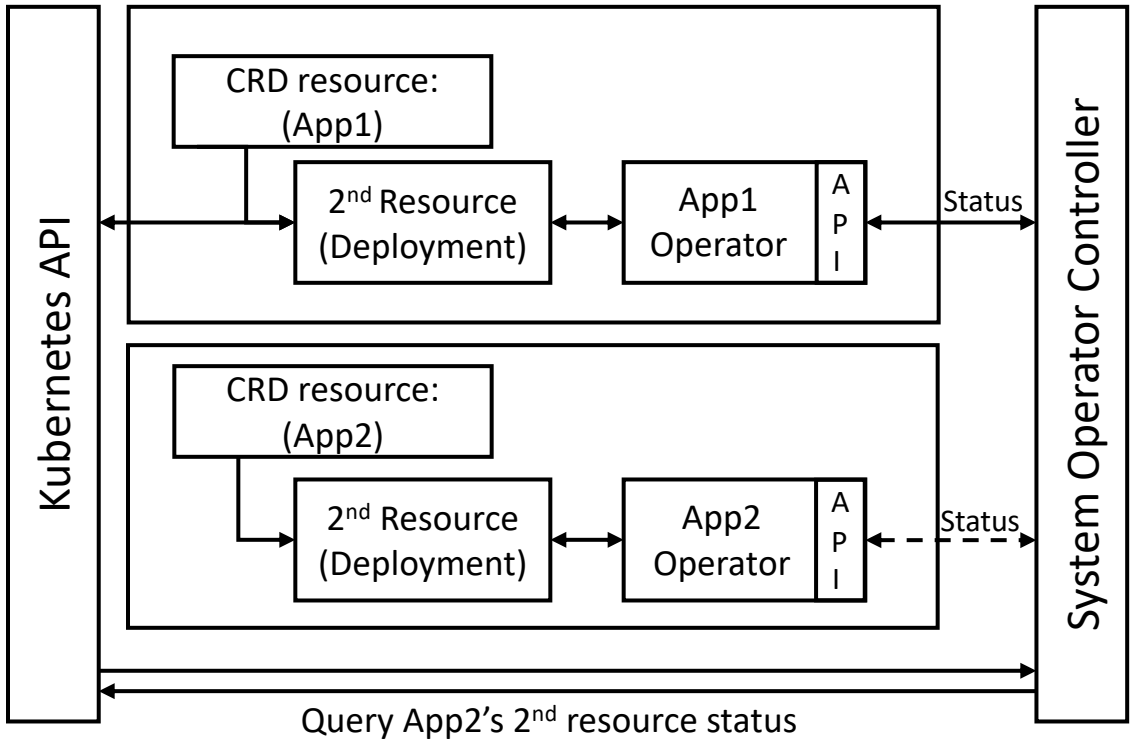


Custom Resource and it Secondary Resources

HITACHI
Inspire the Next

- Secondary resources are defined as Kubernetes resources created by App Operator and managed by App Operator's Custom Resource





In the other word, Custom Resource is the owner of secondary resources. To get this dependence information, we can check the value of field `metadata.ownerReferences` of resources to acquire their owner resource.

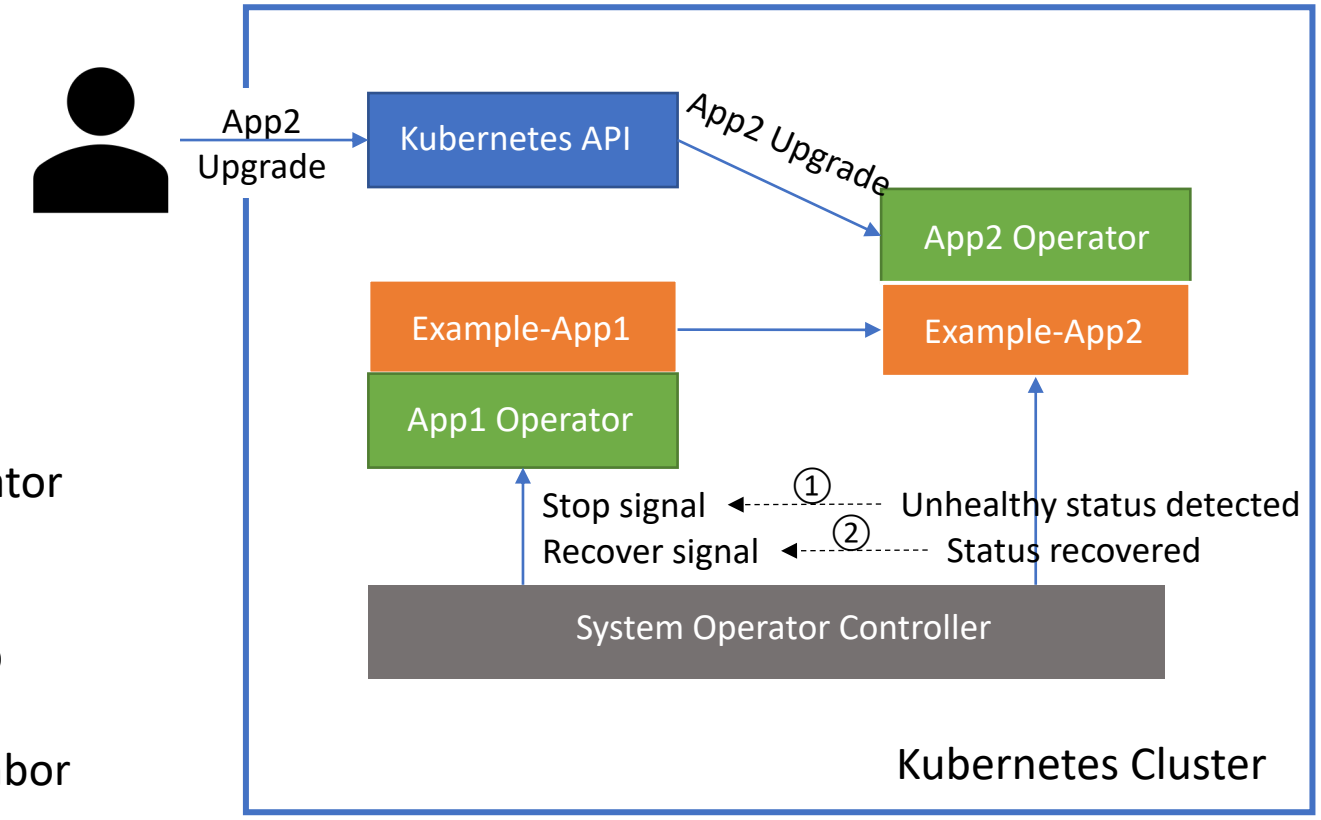
Status of Custom Resource

Sometimes, there is no status defined in a Custom Resource. Then we can check its secondary resources' status instead.

System Regulation

System Regulation happens when:

1. Error happens on some application
2. Application Upgrade happens



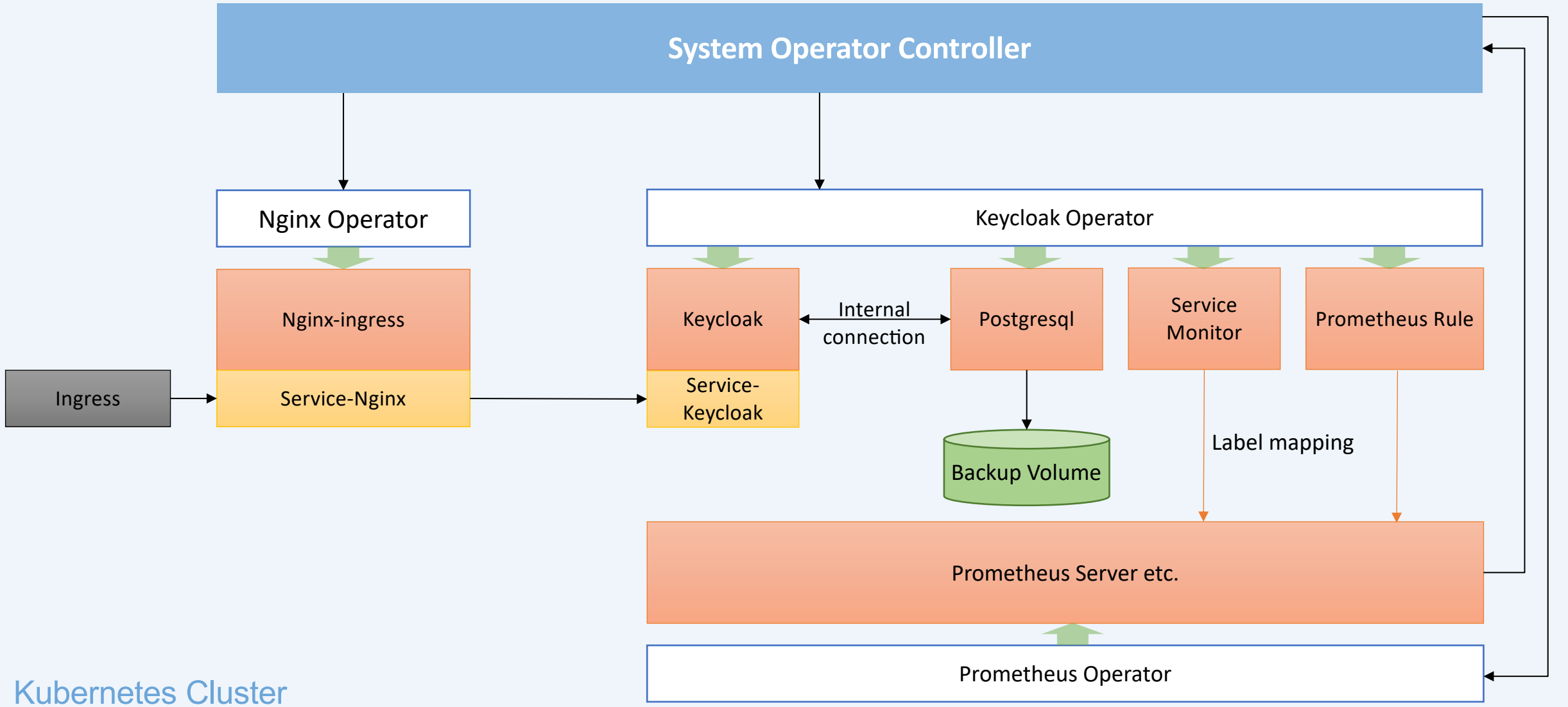
These changes will cause the status of application unhealthy, which will be detected by System Operator Controller.

- Unhealthy status detected -> Stop signal sent to neighbor App Operator*
- Status recovered -> Recover signal sent to neighbor App Operator

Actions the App Operator does when signals accepted should be defined by users

* Neighbor App Operator means App Operator of application that connects with the target application

An example of System Operator



Summary

HITACHI
Inspire the Next

System Operator allows Kubernetes users to

- Design system by connecting App Operators
- Share system by sharing system manifest and System Operator Controller image
- Easily build a Kubernetes system by utilizing App Operators

System Operator is still under research and development

- Prototype of System Operator is being developed
- More features will be introduced into System Operator

Trademarks

HITACHI
Inspire the Next

- "Kubernetes" is a registered trademark of The Linux Foundation.
- "gRPC" is a registered trademark of The Linux Foundation.
- "MongoDB" is a registered trademark of MongoDB, Inc.
- "Kustomize" is a registered trademark of The Linux Foundation.
- "Helm" is a registered trademark of The Linux Foundation.
- All other company names, product names, service names and other proper nouns are trademarks or registered trademarks of their respective companies.
- The TM mark is not shown in the text and figures in this slide.

Thank you for your attention!