

ContainerOps - DevOps Orchestration

Quanyi Ma <maquanyi@huawei.com>



Who Am I?



Quanyi Ma

DevOps & Open Source Expert
Senior Architect & Full Stack Developer

Email: maquanyi@huawei.com

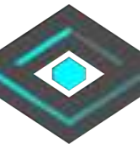
Twitter: [@genedna](https://twitter.com/genedna)

Github: <https://github.com/genedna>

Agenda

— — —

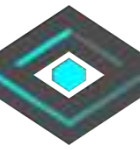
1. The story of DevOps
2. Why developed the DevOps concept in the IT process
3. What's ContainerOps
4. The ContainerOps Component
5. The ContainerOps core – DevOps Workflow Engine
6. Architecture of ContainerOps



DevOps Story

- 2007** - While consulting on a data center migration for the Belgium government, system administrator Patrick Debois becomes frustrated by conflicts between developers and system admins. He ponders solutions.
- Agile Conference 2008 in Toronto** - Andrew Clay Shafer's "*birds of a feather*" ad hoc session called Agile Infrastructure. The only person who showed up was Patrick Debois. Shafer and Debois started a Google group called "*Agile System Administration*"
- O'Reilly Velocity 2009 Conference** - Presentation at Velocity of *10+ Deploys per Day: Dev and Ops Cooperation at Flickr* by John Allspaw and Paul Hammond – Debois watched by streaming video, tweeted.
- October 2009** - Organized through Twitter. Conversation continued on Twitter and the #DevOps hashtag was born, dropping "Days" for brevity.
- 2010 Mountain View, CA** - DevOpsDays
- Mar. 2011** - Gartner's first notes about DevOps
- April 2012** - In an InfoQ video interview, Debois admitted that naming the movement was not as intentional as it might seem: "I picked 'DevOpsDays' as Dev and Ops working together because '*Agile System Administration*' was too long," he said. "**There never was a grand plan for DevOps as a word.**"

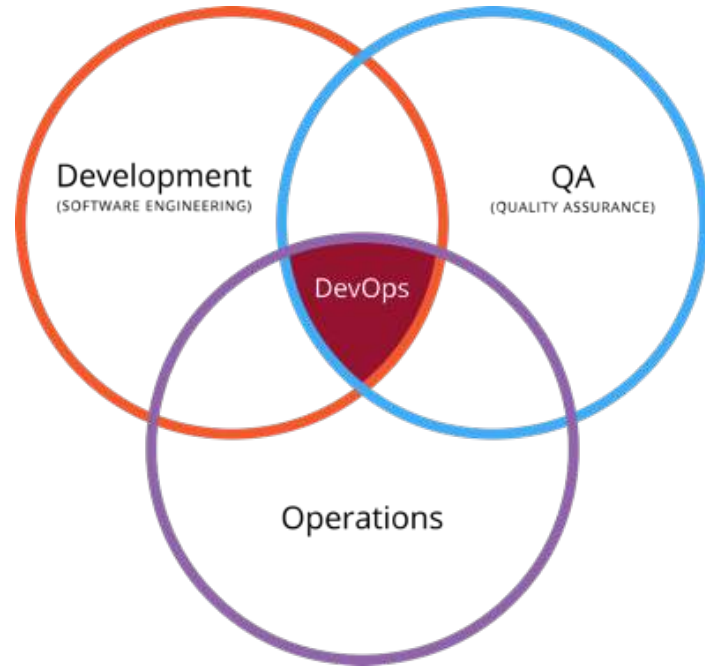




What's the DevOps?

DevOps is "a portmanteau of 'development' and 'operations'" and is "a software development method that stresses communications, collaboration, integration, automation and measurement of cooperation between software developers and other IT professionals".

-From Wikipedia



DevOps is an operational philosophy that promotes better communication between development and operations as more elements of operations become programmable.

What's the DevOps ultimate AIM?

— — —

Goal ->

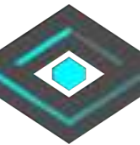
The ultimate is break down barriers between developer, QAs and operators.

How ->

Define the operation environment at development stage.

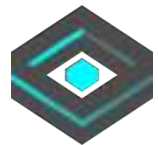
Define the process from development to the production.

Automate everything.



Why improve so hard?

1. Don't break the original DevOps workflow.
2. Add DevOps orchestration tool adaptive the workflow.
3. Improve the process with customize DevOps task.
4. Add DevOps service like Travis CI.
5. Everyone is happy!!!



Why improve so hard?

— — —

1. Don't break the original DevOps workflow.
2. Add DevOps orchestration tool adaptive the workflow.
3. Improve the process with customize DevOps task.
4. Add DevOps service like Travis CI.
5. Everyone is happy!!!



ContainerOps - DevOps Orchestration

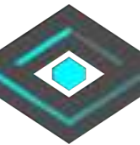
Defining -> Component

Drawing -> Workflow

Running -> Container Orchestration

Open Source @ ->

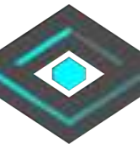
<https://github.com/Huawei/containerops>



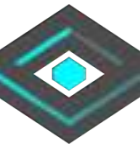
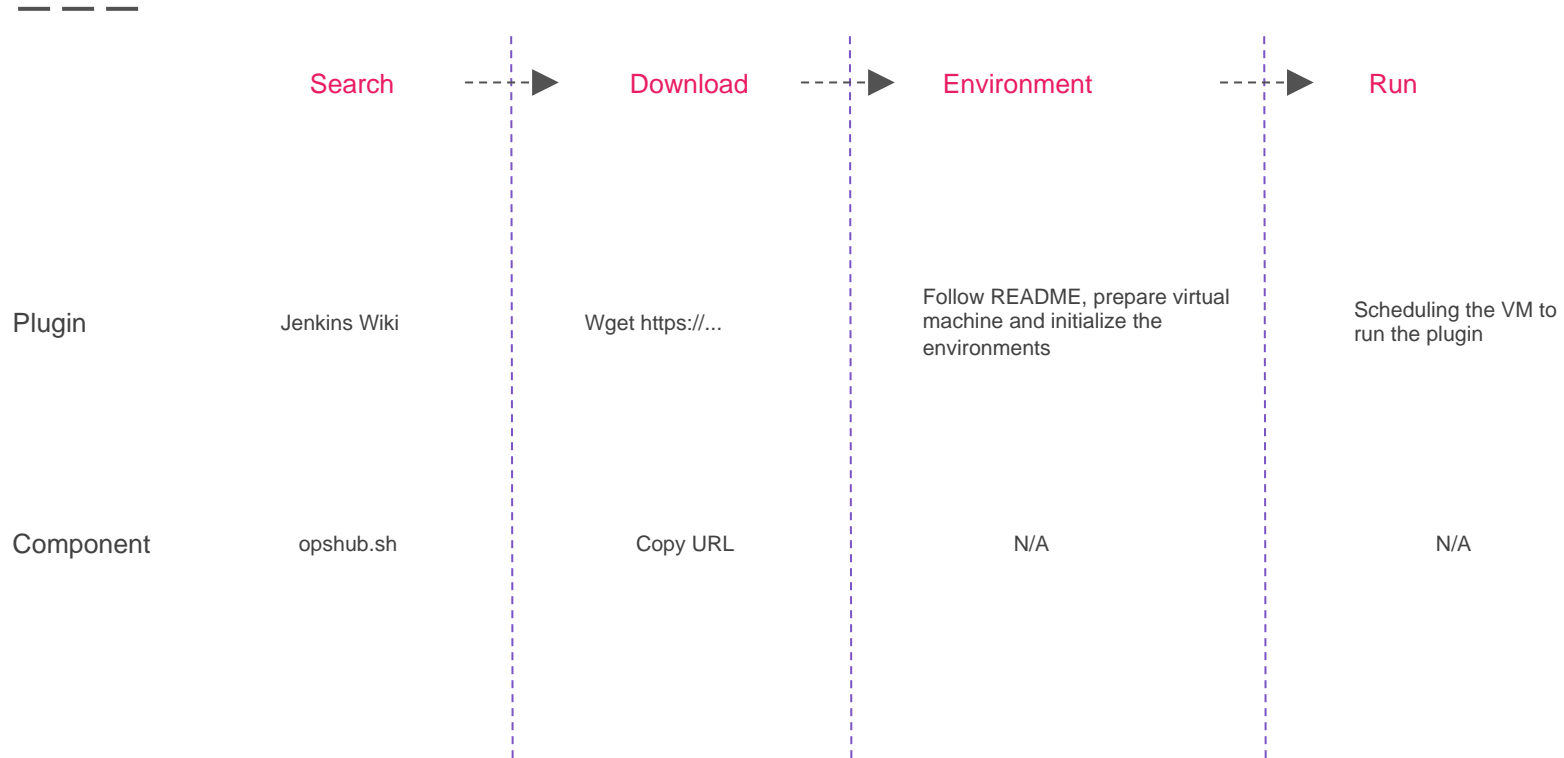
DevOps Component

Defining Component - Container Image For DevOps

1. Encapsulating your DevOps task in a container image.
2. Defining the input & output JSON data.
3. Management the component lifecycle in the containerops platform.
4. Share your component at <https://opshub.sh>



Why Use DevOps Component?



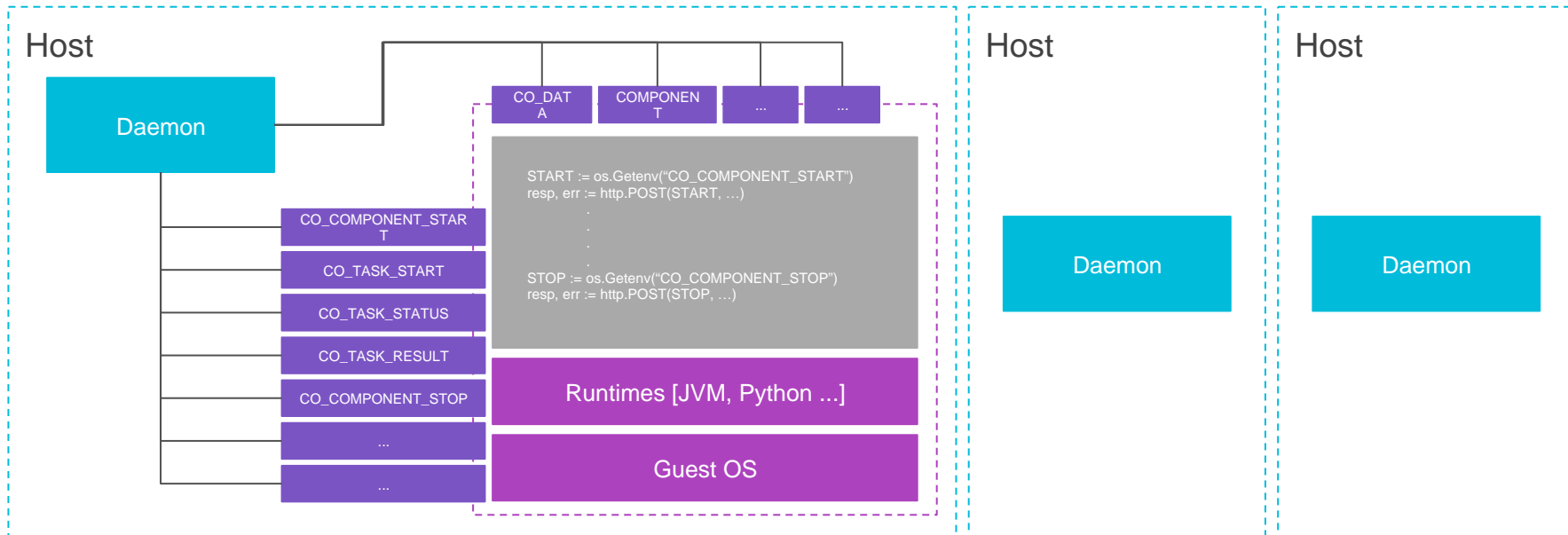
Component Input & Output JSON Data

```
{
  "pull_request": {
    "url": "https://api.github.com/repos/baxterthehacker/public-repo/pulls/1",
    "id": 34778301,
    "html_url": "https://github.com/baxterthehacker/public-repo/pull/1",
    "diff_url": "https://github.com/baxterthehacker/public-repo/pull/1.diff",
    "patch_url": "https://github.com/baxterthehacker/public-repo/pull/1.patch",
    "issue_url": "https://api.github.com/repos/baxterthehacker/public-repo/issues/1",
    "number": 1,
    "state": "open",
    "locked": false,
    "title": "Update the README with new information",
    ...
  }
  ...
  "repo": {
    "id": 35129377,
    "name": "public-repo",
    "full_name": "baxterthehacker/public-repo",
    "owner": {
      "login": "baxterthehacker",
      ...
      "type": "User",
      "site_admin": false
    },
    "private": false,
    "html_url": "https://github.com/baxterthehacker/public-repo",
    ...
  }
  ...
}
```

```
{
  "certificate": {
    "state": true,
    "username": "genedna",
    "name": "Meaglith Ma",
    "email": "maquanyi@huawei.com"
  },
  "cla": {
    "state": true,
    "username": "genedna",
    "name": "Meaglith Ma",
    "email": "maquanyi@huawei.com",
    "date": "2016-12-12 22:22:22:000"
  }
}
```

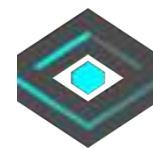
ContainerOps Workflow Engine

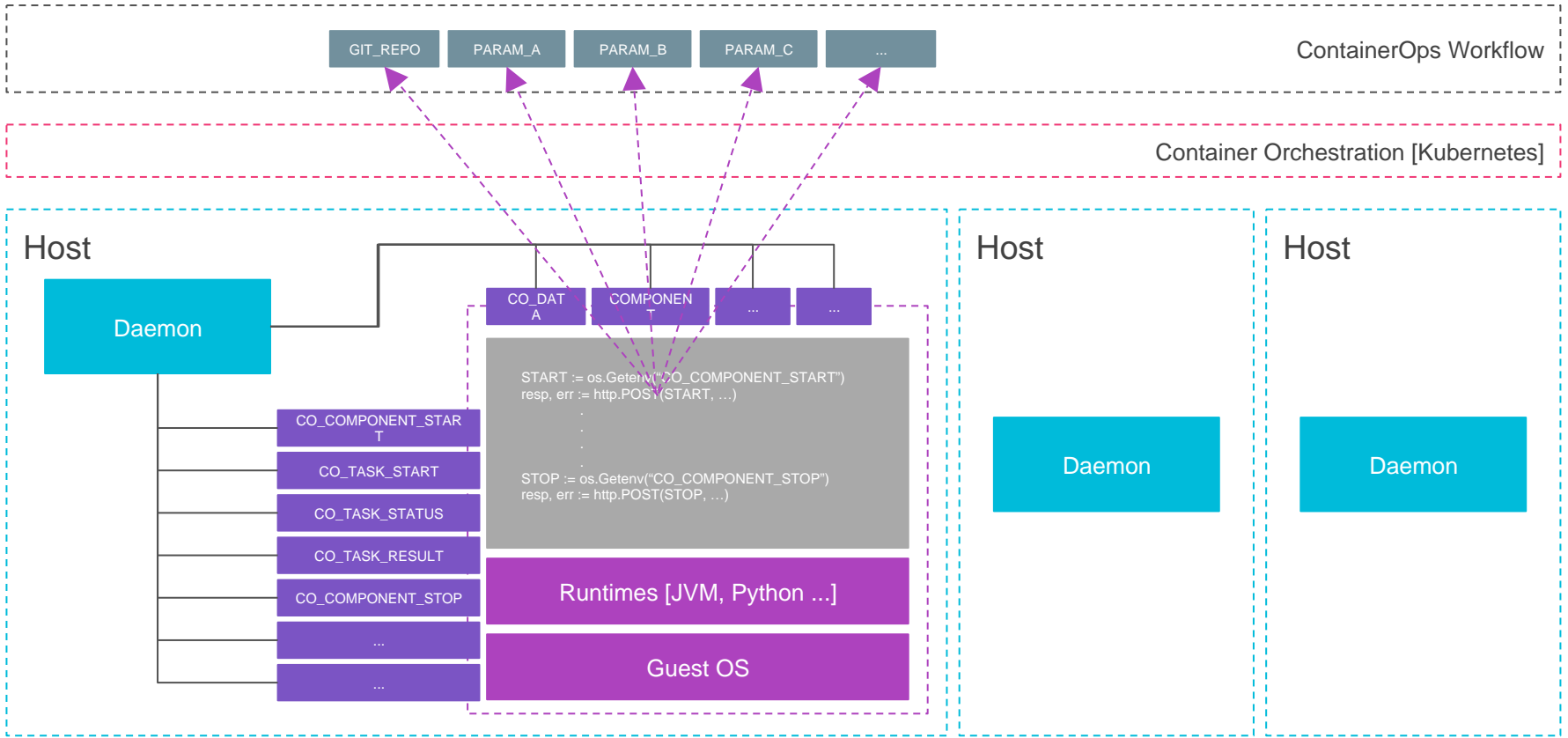
Container Orchestration [Kubernetes]



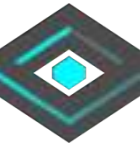
DevOps Component

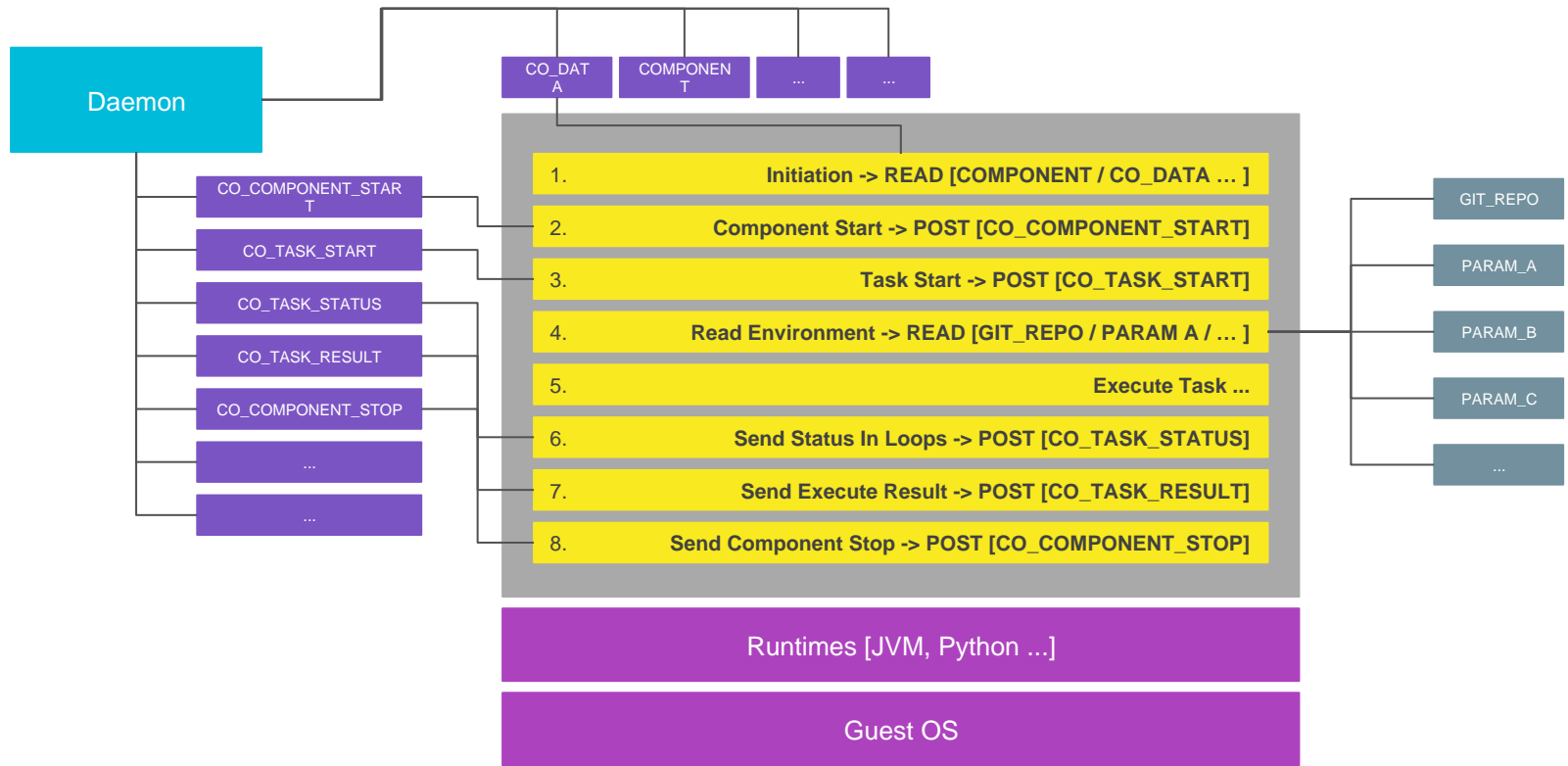
`CO_COMPONENT_START = https://containerops.sh/v1/genedna/pilotage/workflow/start/...`



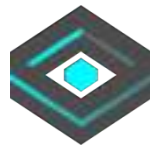


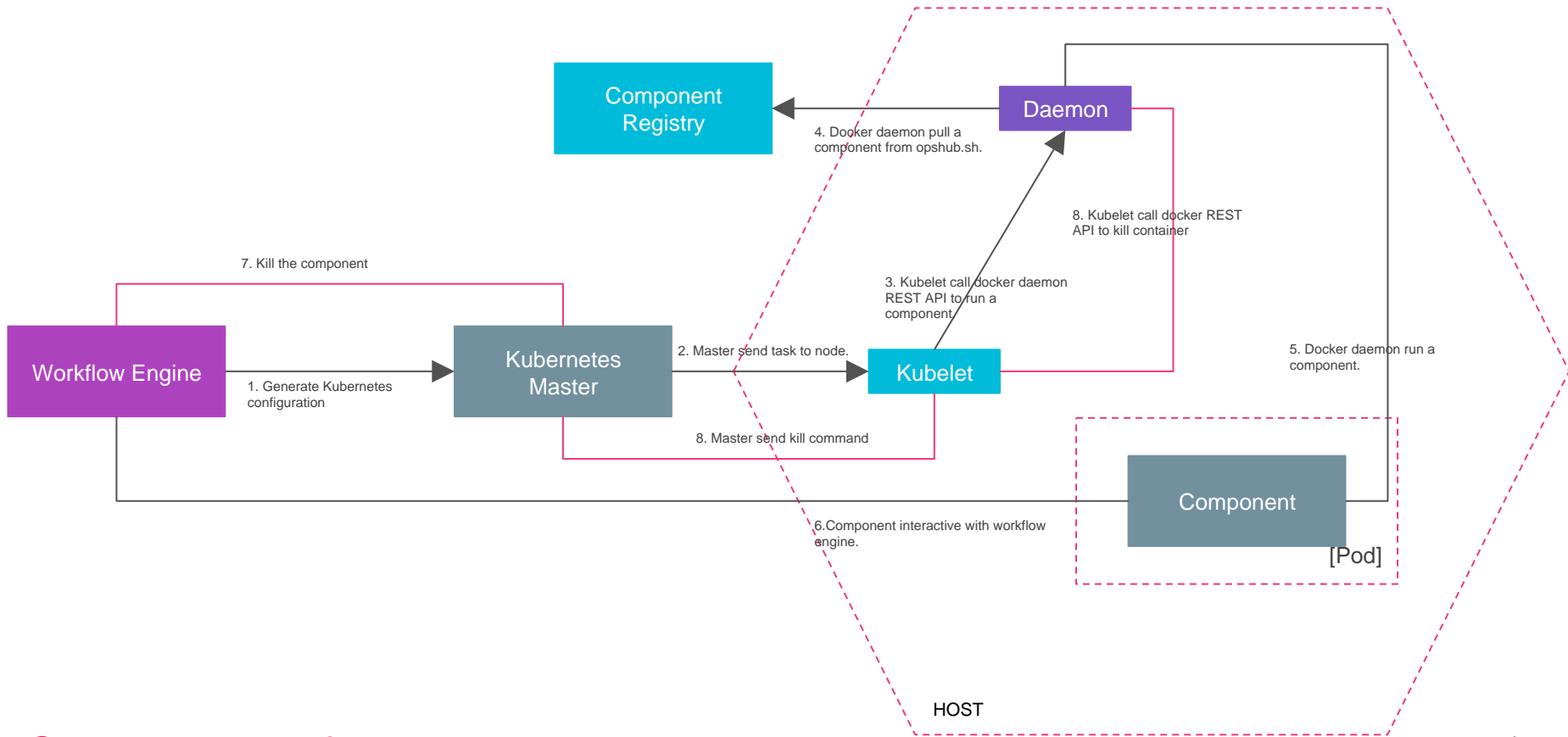
DevOps Component - Environment Parameters



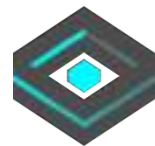


DevOps Component





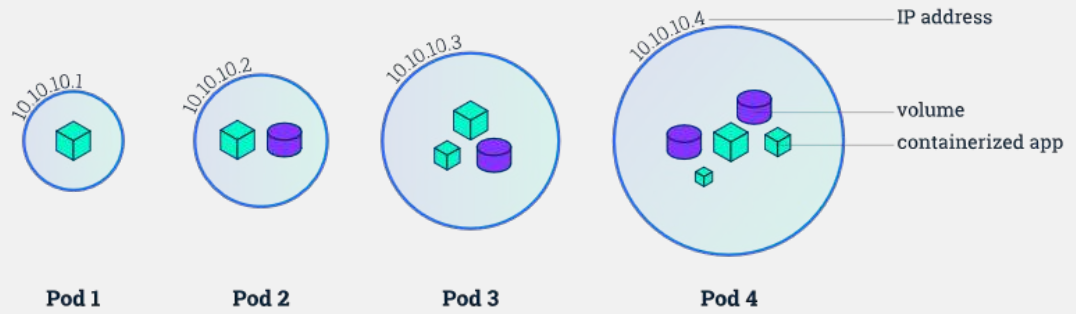
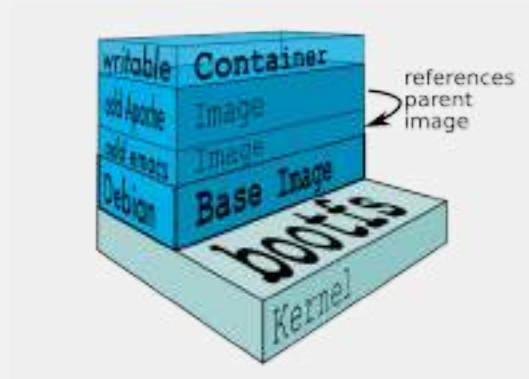
Component Lifecycle



Why A Component Not A Pod?

— — —

1. Adaptive other container orchestration system like Docker Swarm.
2. Only one task in the component to easily maintain and share.



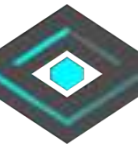
Demo

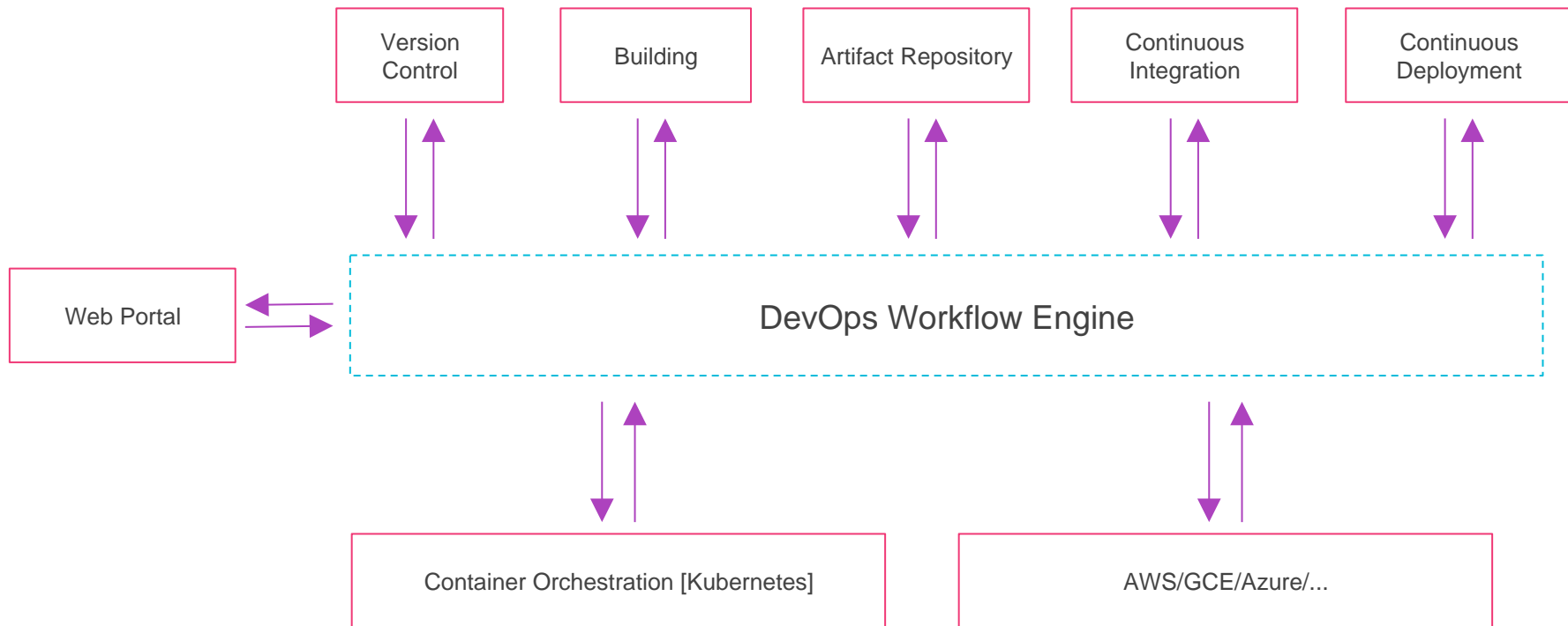
<https://www.youtube.com/watch?v=gWC6I7vSy6A>

Workflow

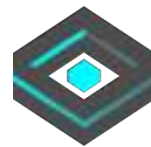


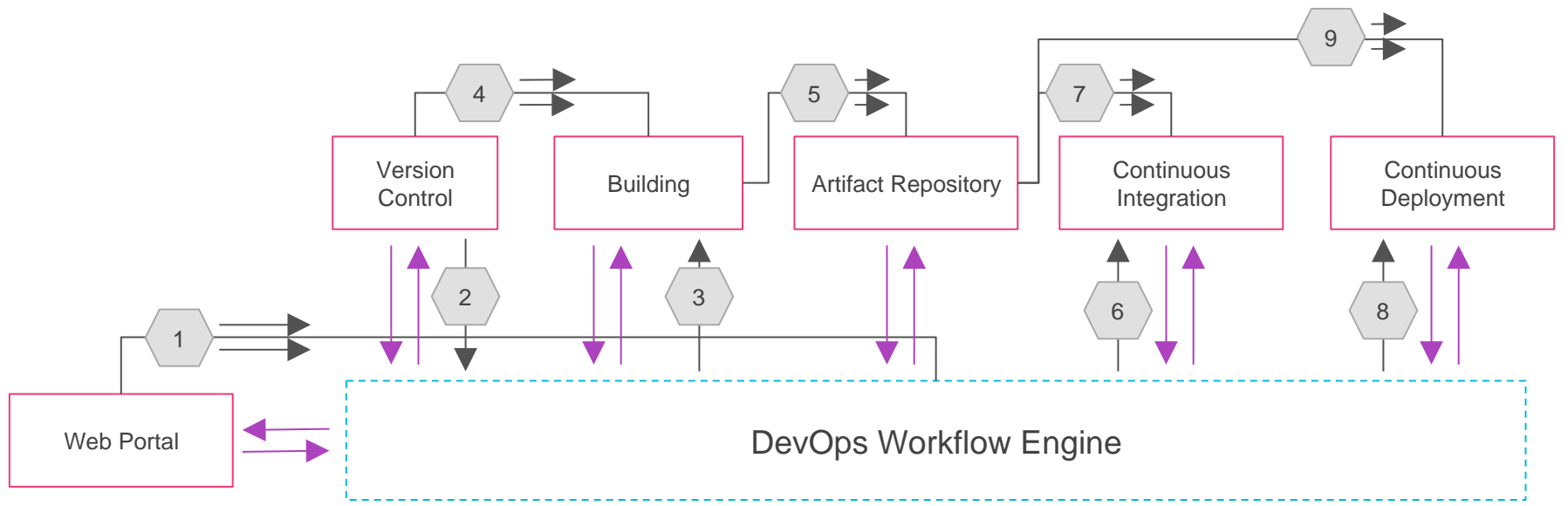
Drawing DevOps Workflow And Event Data
Workflow



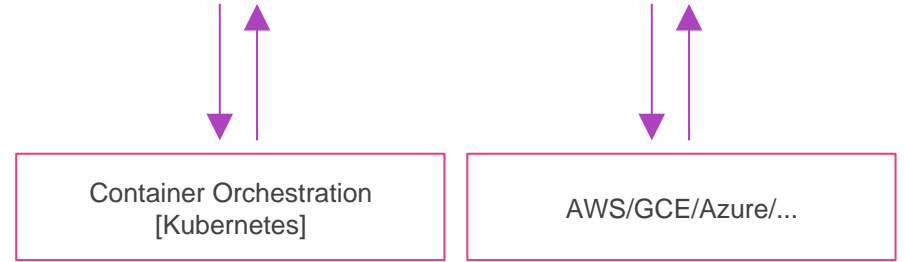


ContainerOps Classic DevOps Workflow

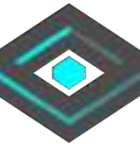




1. Drawing DevOps workflow from web portal.
2. Developer submit commit or PR triggering the workflow execute.
3. Engine notify the building service like Jenkins or Concourse to execute artifact or container image pipeline.
4. The building service pull source code from version control service like Github or Gitlab.
5. The building service push artifact or image to the repository.
6. Engine notify the CI service execute integrated pipeline.
7. The CI service pull the artifact or image from the repository.
8. Engine notify the CD service execute deployment pipeline.
9. The CD service pull the artifact or image to deploy in different stage.



ContainerOps Workflow Engine

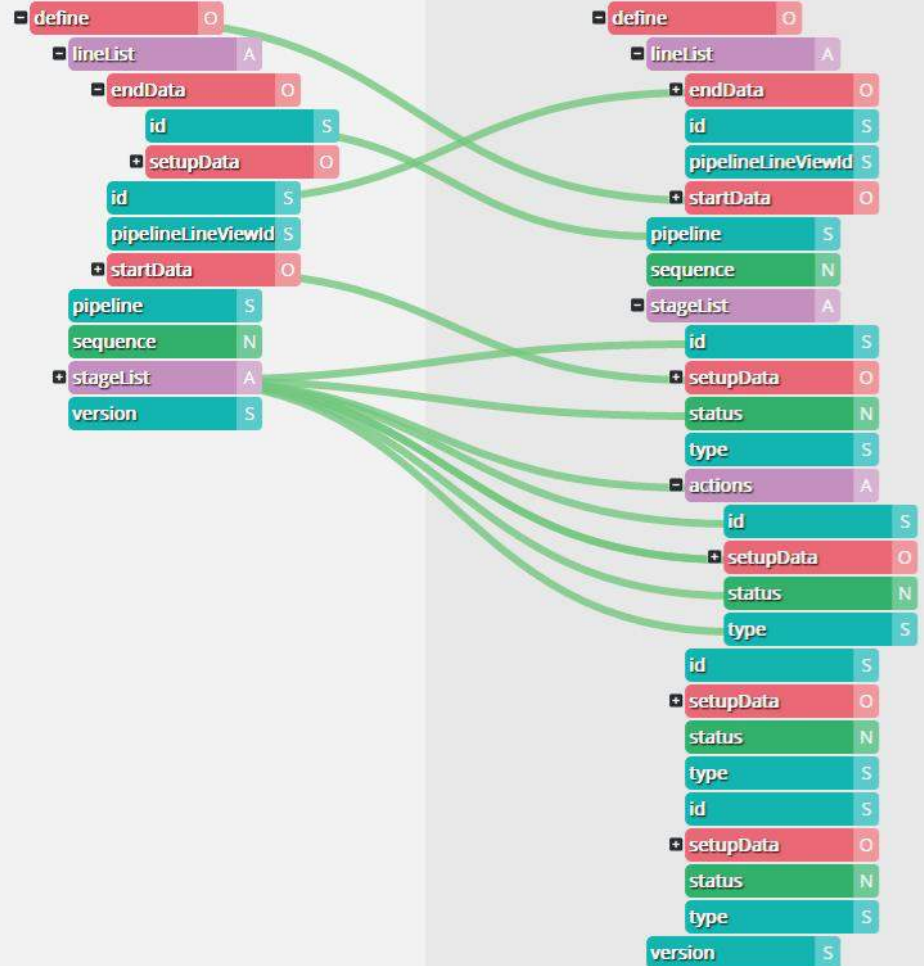


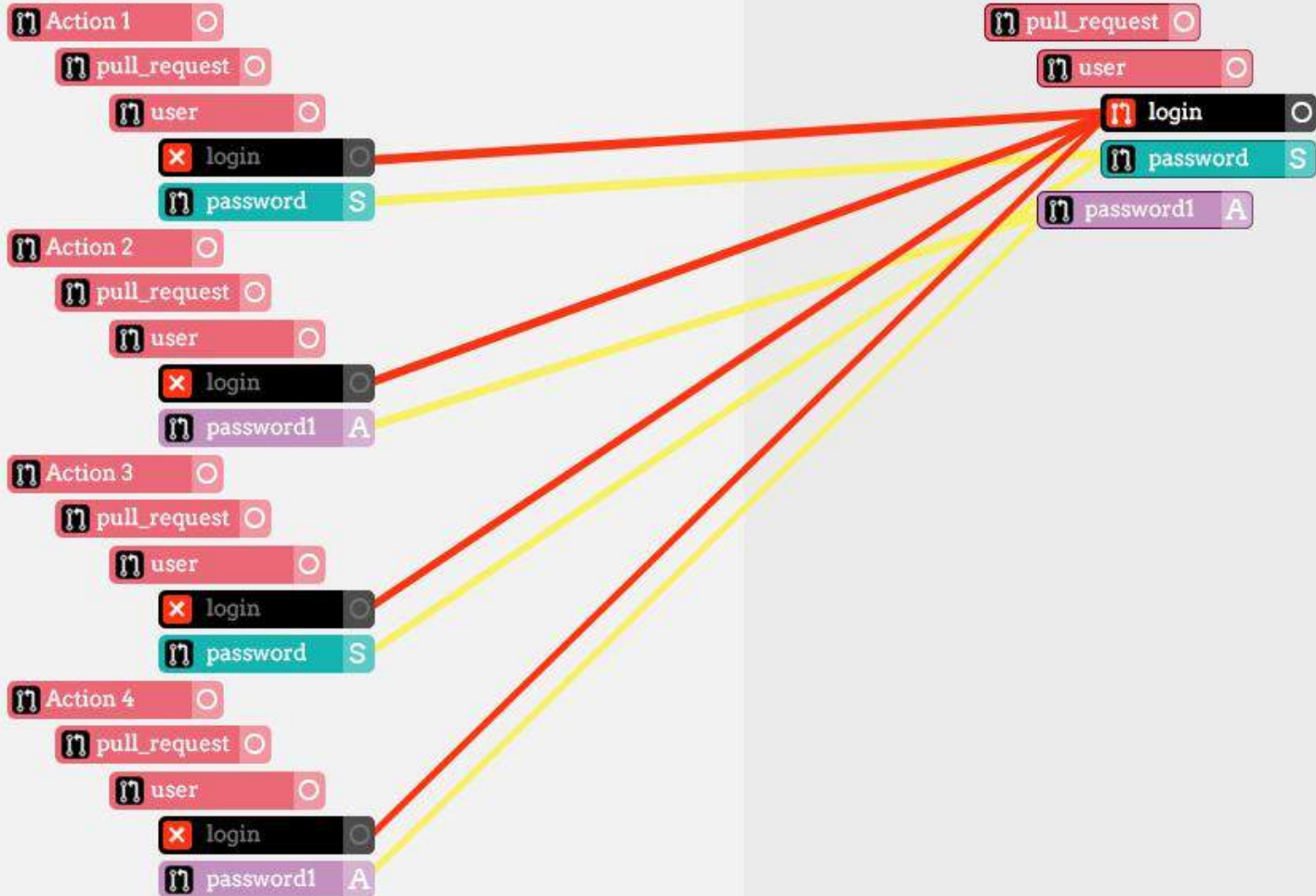
Demo

<https://www.youtube.com/watch?v=gWC6I7vSy6A>

Event Data Link

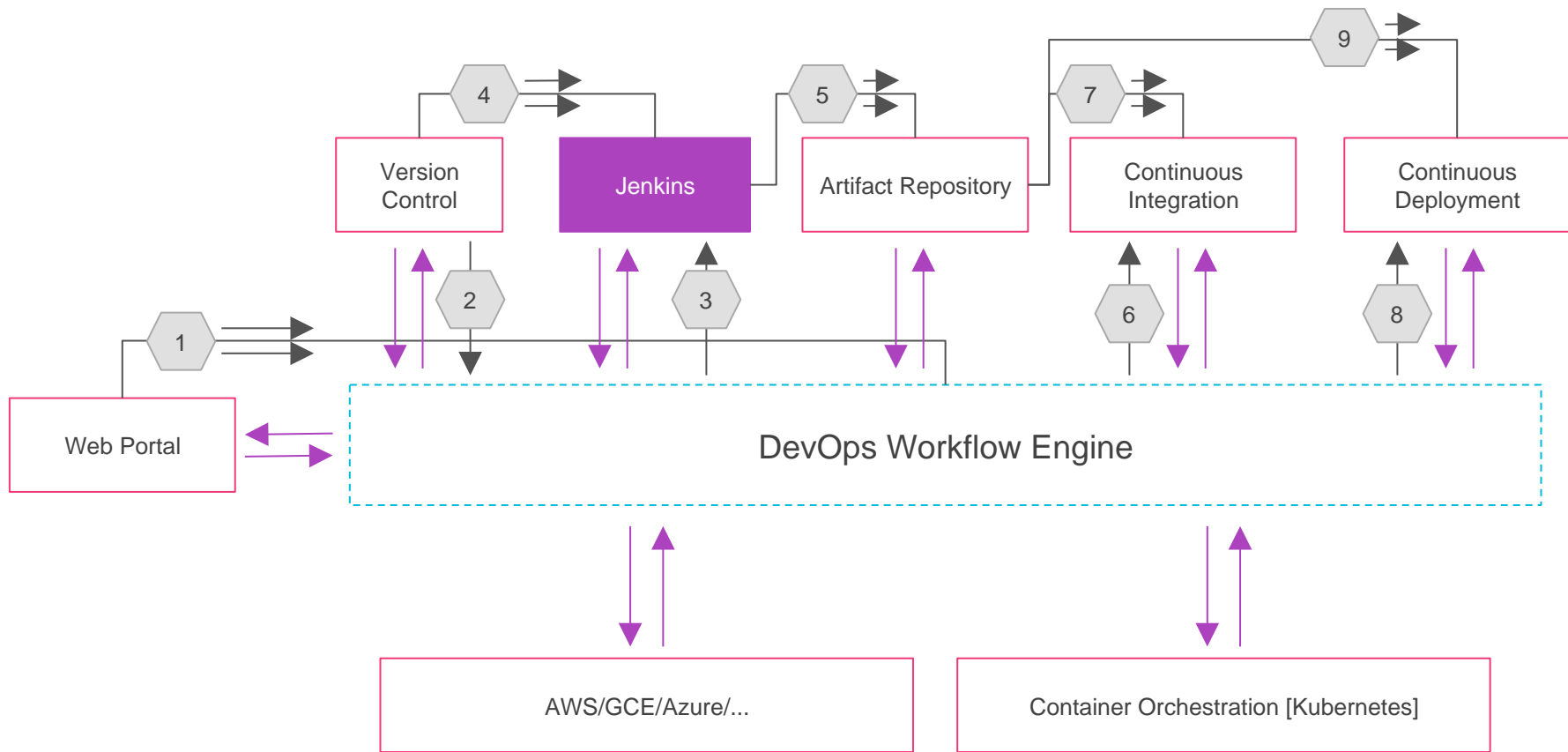
- 1.Components and components
- 2.Services and services
- 3.Components and Services



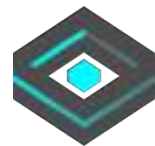


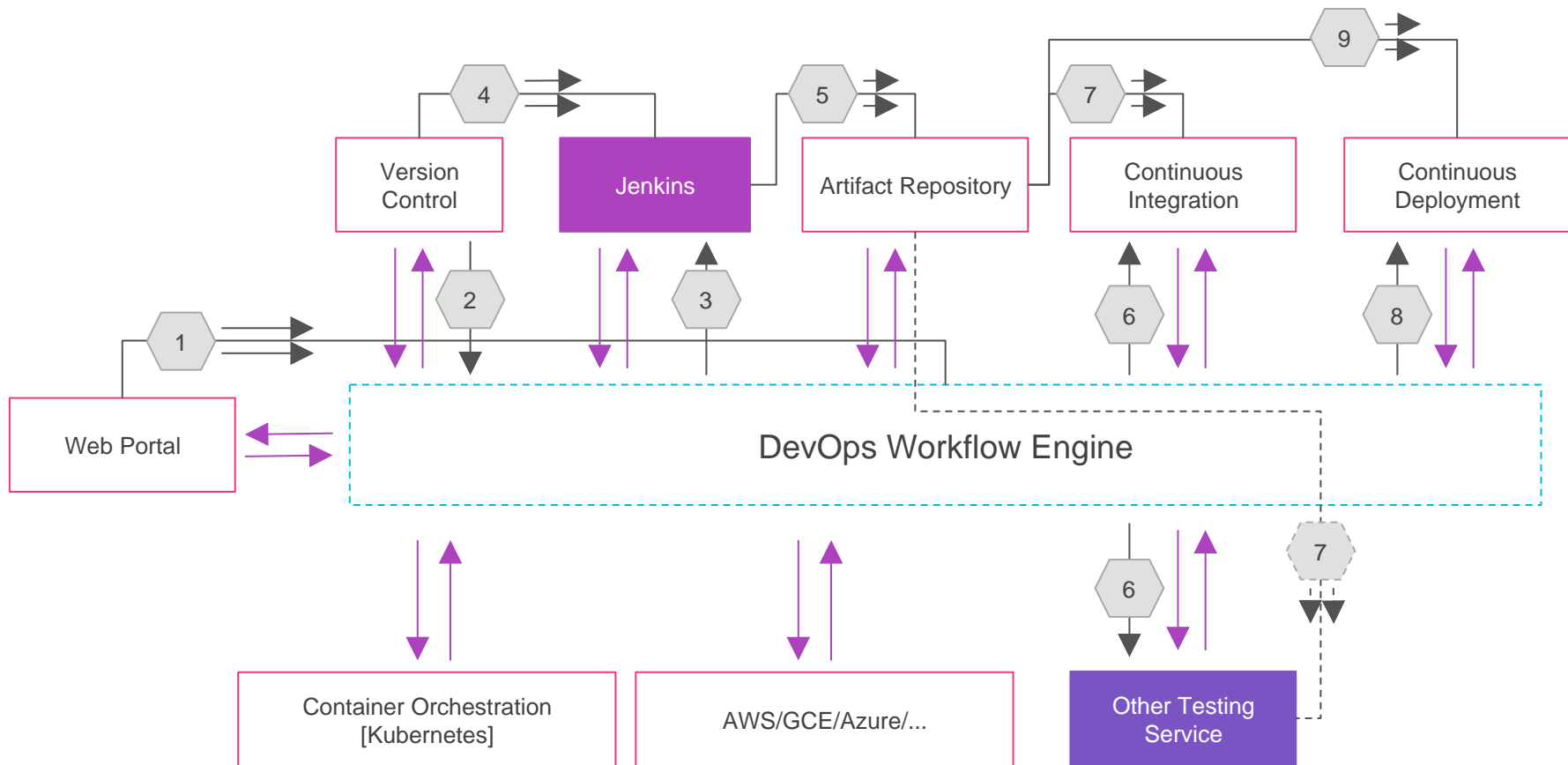
Event
Conflict

Running

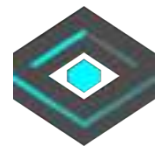


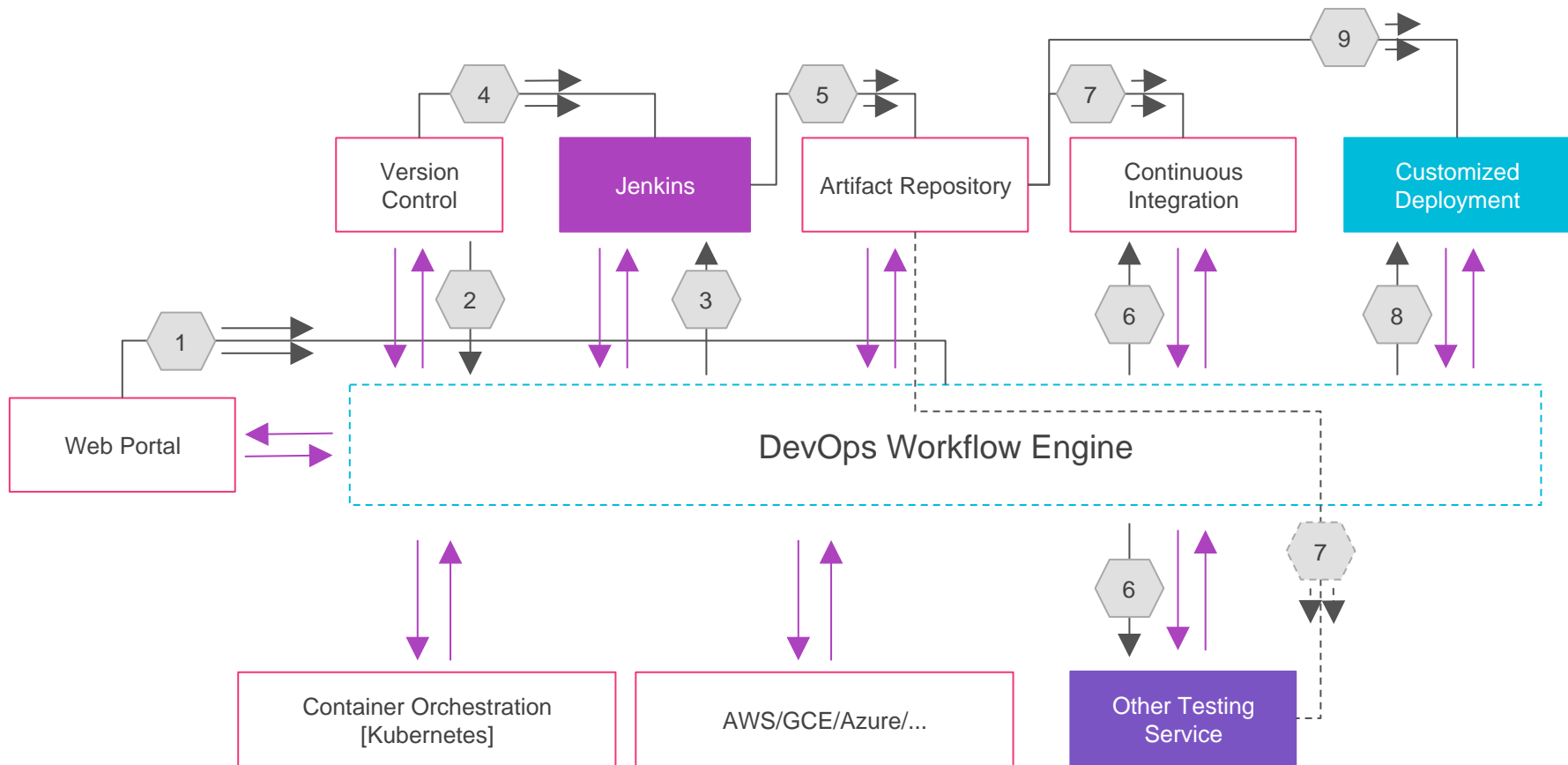
Customize DevOps Workflow With Engine - 1



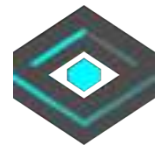


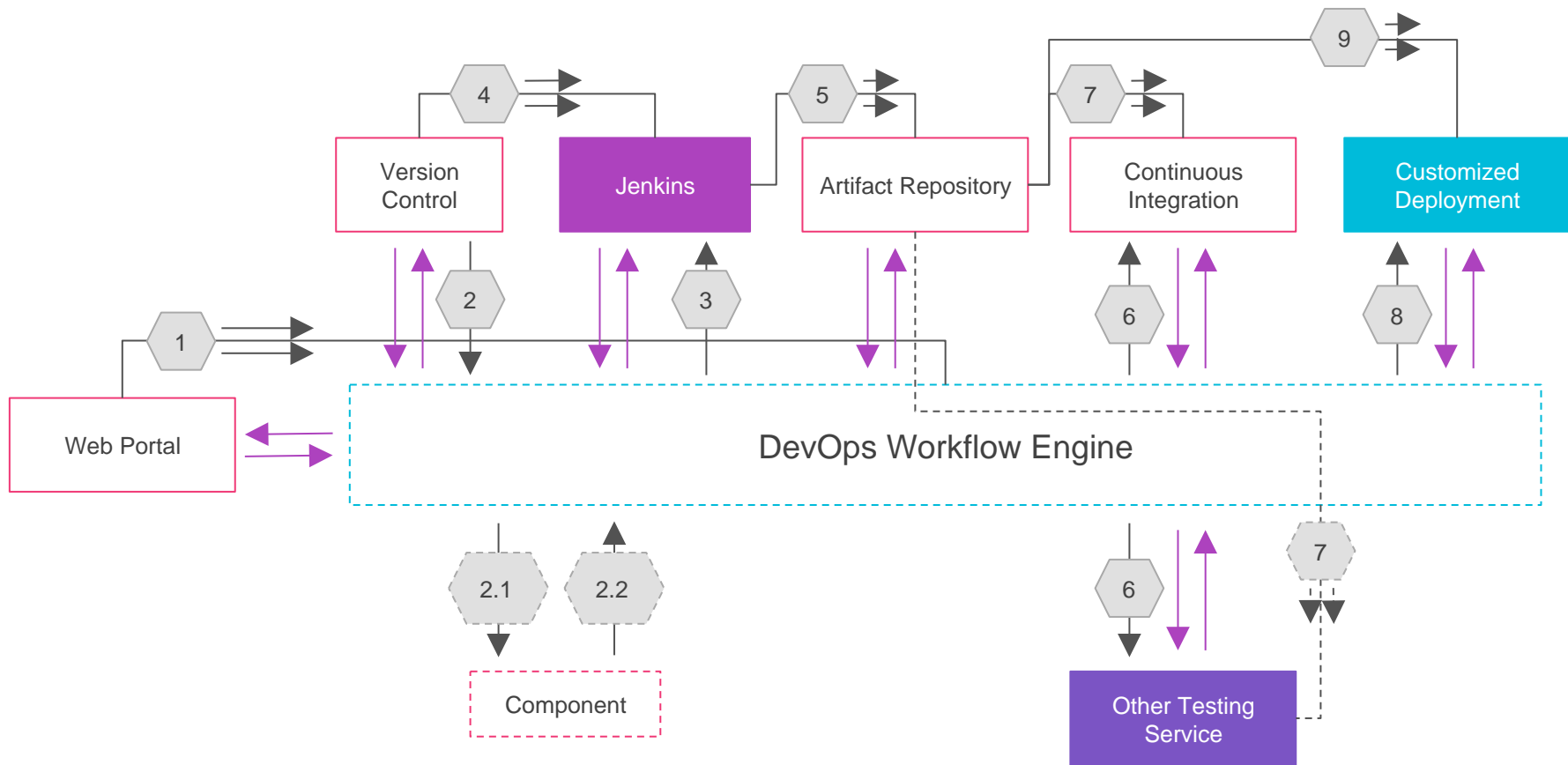
Customize DevOps Workflow With Engine - 2



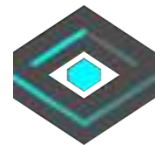


Customize DevOps Workflow With Engine - 3





Customize DevOps Workflow With Engine - 4

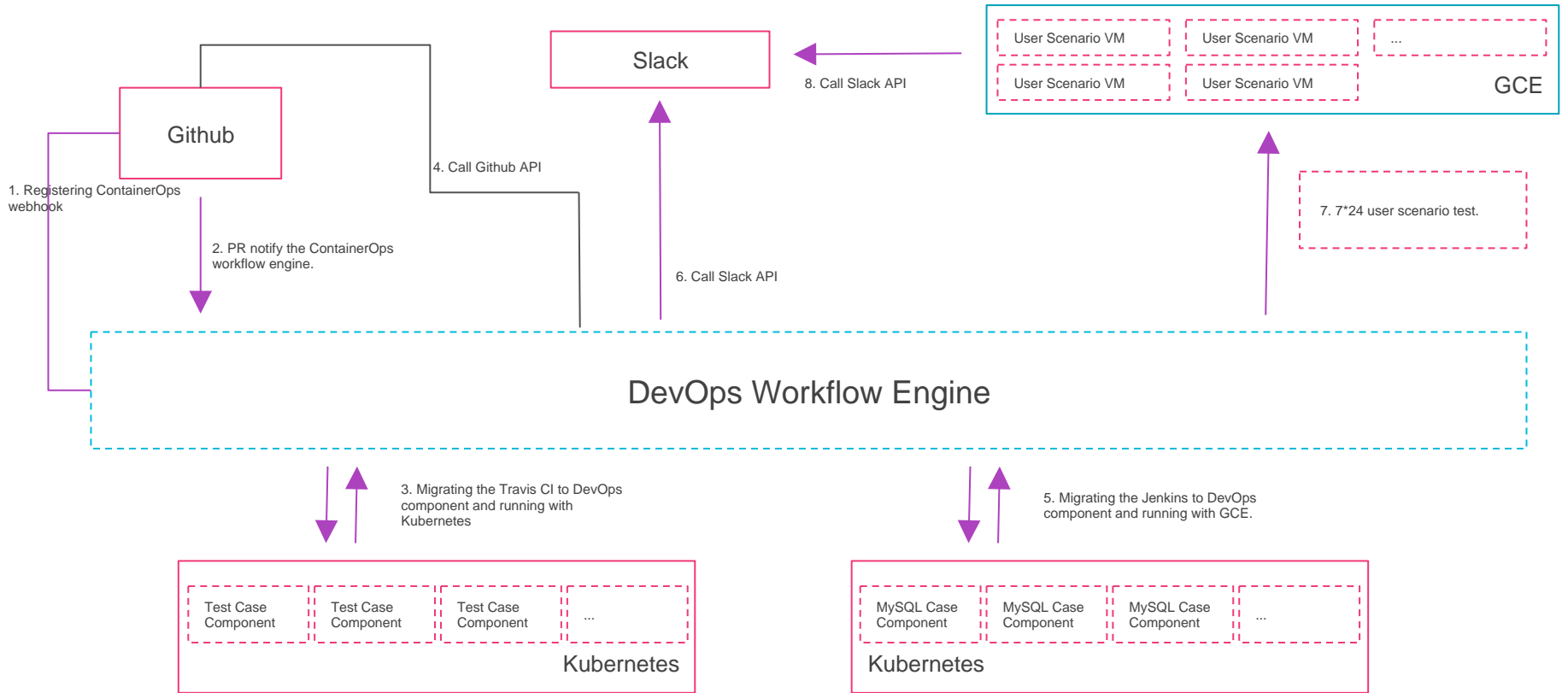


User Case

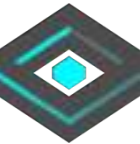
TiDB/TiKV/PB Case

3 DevOps Stage with different systems.

- a. Test case Within Travis CI
- b. Merge Stage: 1000000+ MySQL test case with Jenkins.
- c. Release Stage: 7*24 hours user scenario with manual.



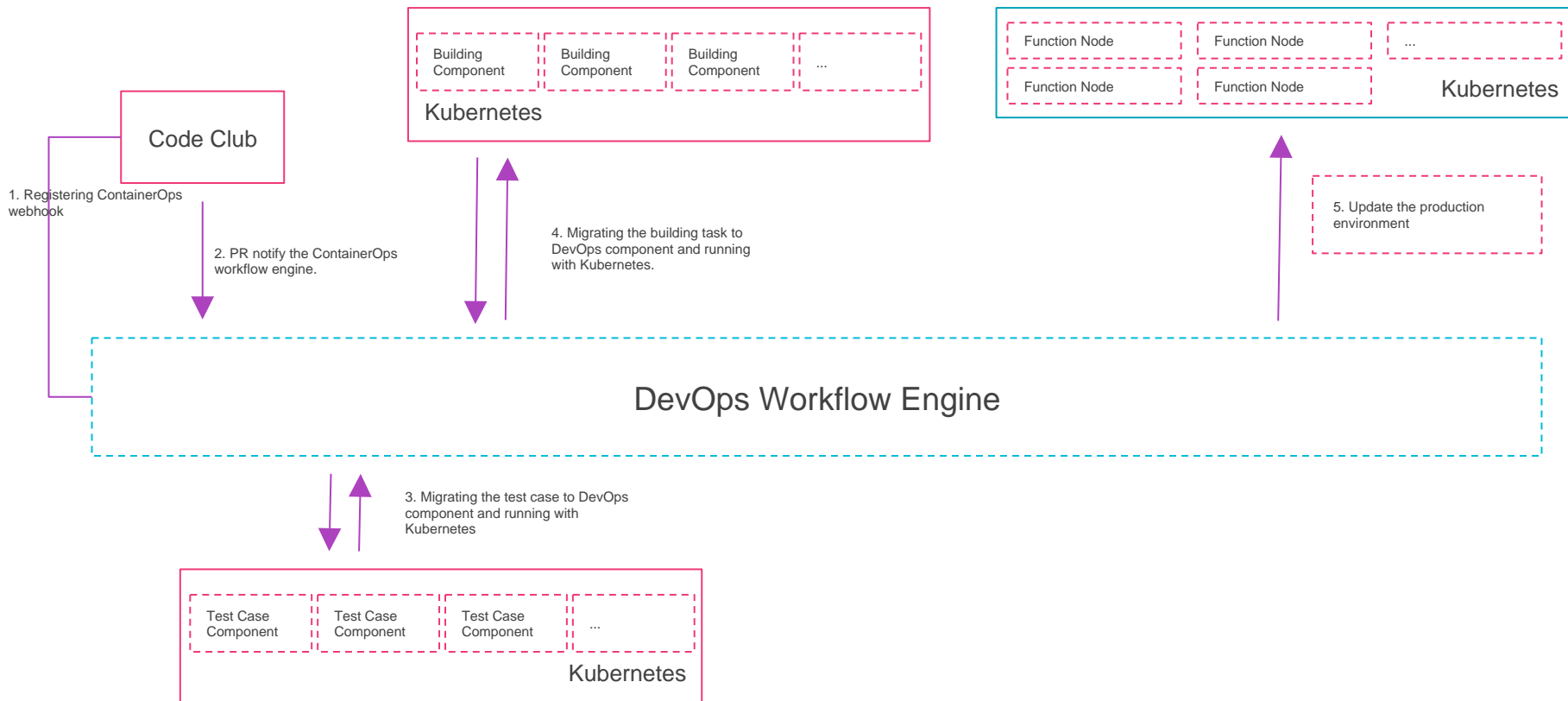
ContainerOps Refactoring DevOps Workflow



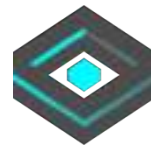
Huawei Internal Project Case

2 DevOps Stage with different systems.

- a. Jenkins: Run test case and build.
- b. Manual: Ops with Kubernetes.



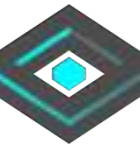
ContainerOps Refactoring DevOps Workflow

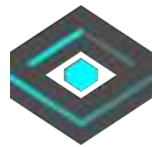


ContainerOps - DevOps Orchestration

With Container, For
Container

containerops.sh & opshub.sh





End & Thanks

<https://github.com/Huawei/containerops>