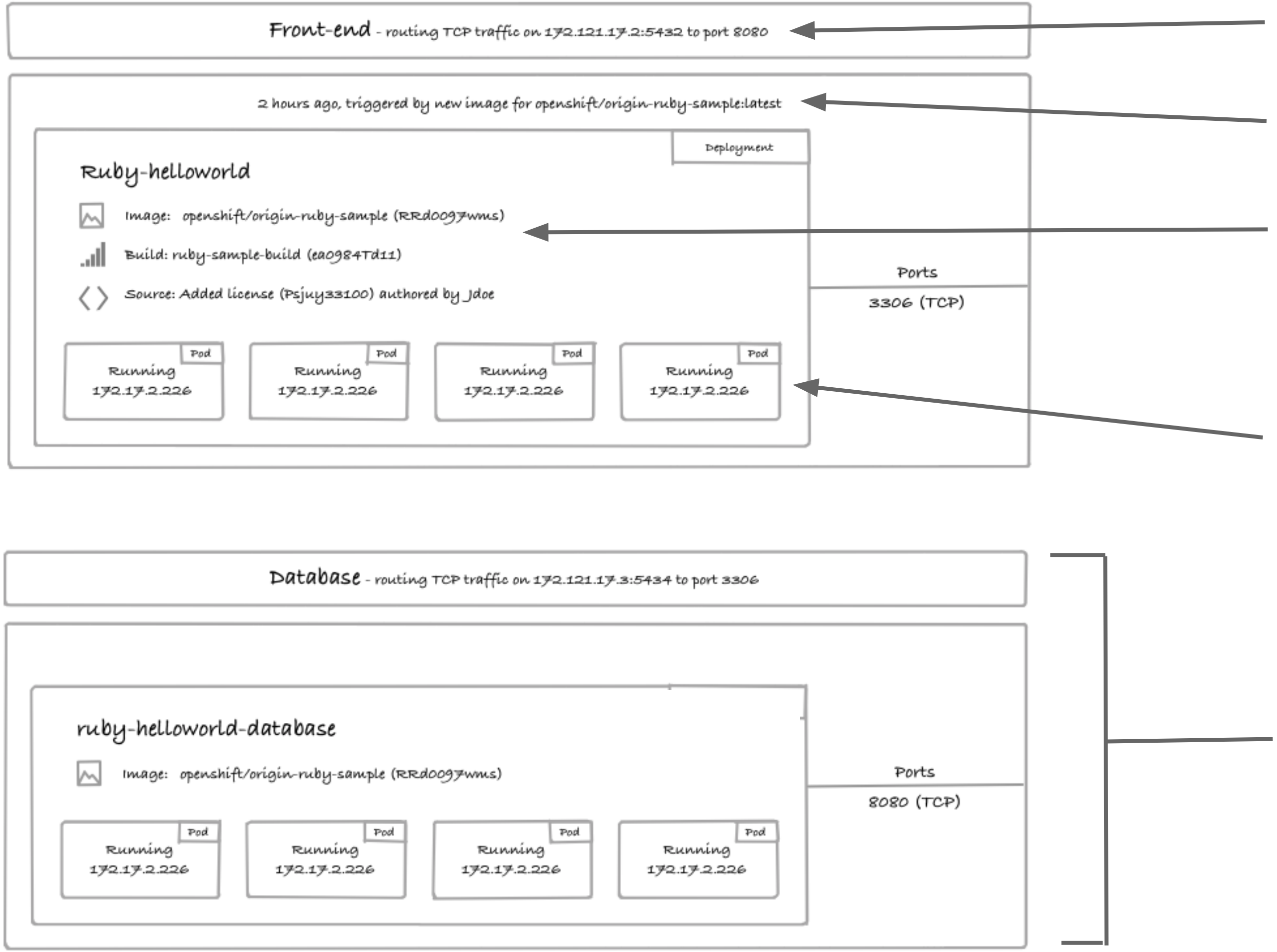
Structural overview of k8s and OpenShift resources


Front-end - routing TCP traffic on 172.121.17.2:5432 to port 8080
2 hours ago, triggered by new image for openshift/origin-ruby-sample:latest
Deployment
Ruby-helloworld
Image: openshift/origin-ruby-sample (RRd0097wms)
Build: ruby-sample-build (ea0984Td11)
Source: Added license (Psjuy33100) authored by Jdoe
Ports
3306 (TCP)
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Database - routing TCP traffic on 172.121.17.3:5434 to port 3306
ruby-helloworld-database
Image: openshift/origin-ruby-sample (RRd0097wms)
Ports
8080 (TCP)
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226

What's being shown:

k8s Service, shows the IP and port that is being exposed within the cluster. Visually it should be associated with the things that it is routing to.

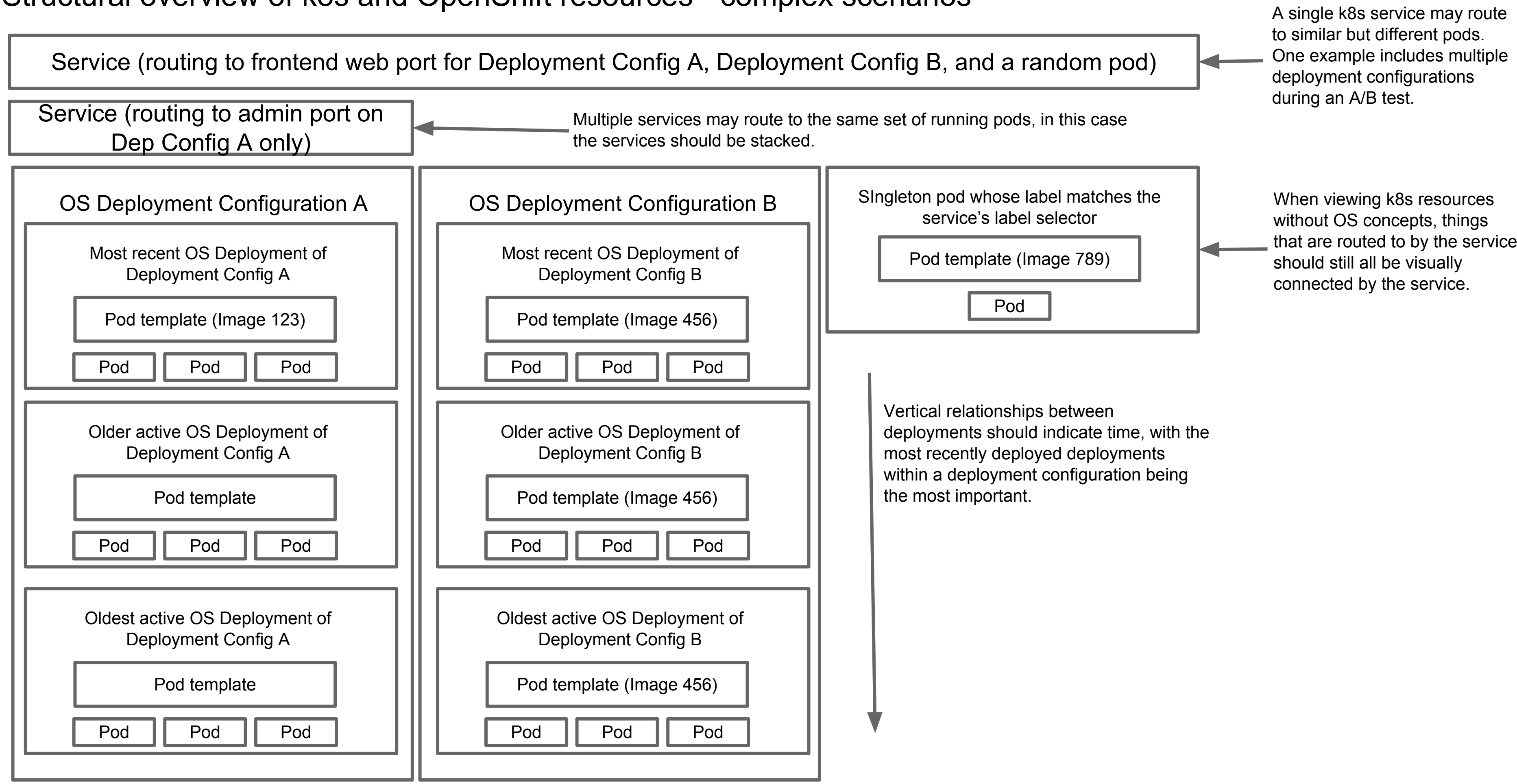
OpenShift Deployment, shows when and why the deployment was triggered, visually associated with what is running in the deployment

k8s/OS Pod Template (what am I actually running), shows what image is running. if the image came from OS build process then it includes build info. Shows source info that went into the build when it is known.

k8s Pods, the actual instances of the pod template including a minimal amount of status (state, IP, etc).

Example of the same visualization applied to only k8s resources.

Structural overview of k8s and OpenShift resources - complex scenarios



Service (routing to frontend web port for Deployment Config A, Deployment Config B, and a random pod)

A single k8s service may route to similar but different pods. One example includes multiple deployment configurations during an A/B test.

Service (routing to admin port on Dep Config A only)

Multiple services may route to the same set of running pods, in this case the services should be stacked.

OS Deployment Configuration A

Most recent OS Deployment of Deployment Config A

Pod template (Image 123)

Pod Pod Pod

Older active OS Deployment of Deployment Config A

Pod template

Pod Pod Pod

Oldest active OS Deployment of Deployment Config A

Pod template

Pod Pod Pod

OS Deployment Configuration B

Most recent OS Deployment of Deployment Config B

Pod template (Image 456)

Pod Pod Pod

Older active OS Deployment of Deployment Config B

Pod template (Image 456)

Pod Pod Pod

Oldest active OS Deployment of Deployment Config B

Pod template (Image 456)

Pod Pod Pod

SIngleton pod whose label matches the service’s label selector

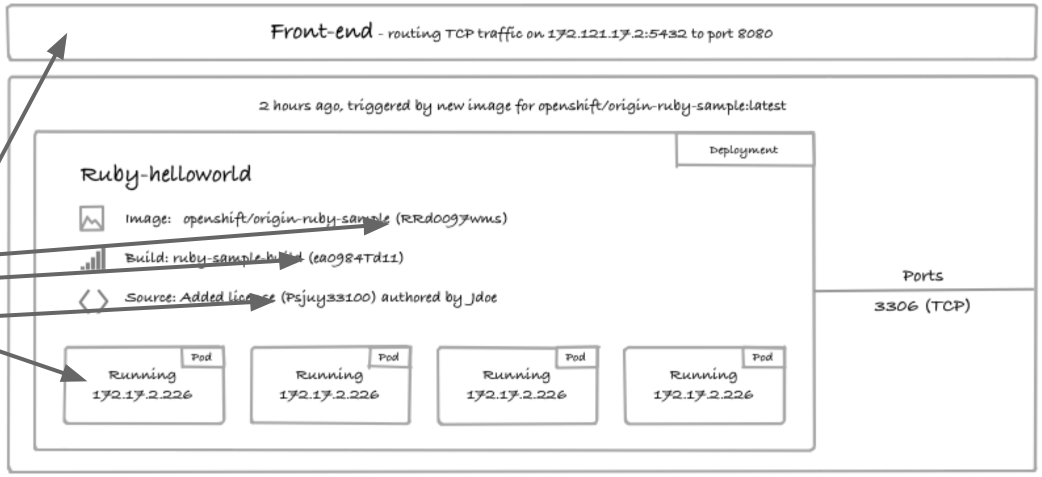
Pod template (Image 789)

Pod

When viewing k8s resources without OS concepts, things that are routed to by the service should still all be visually connected by the service.

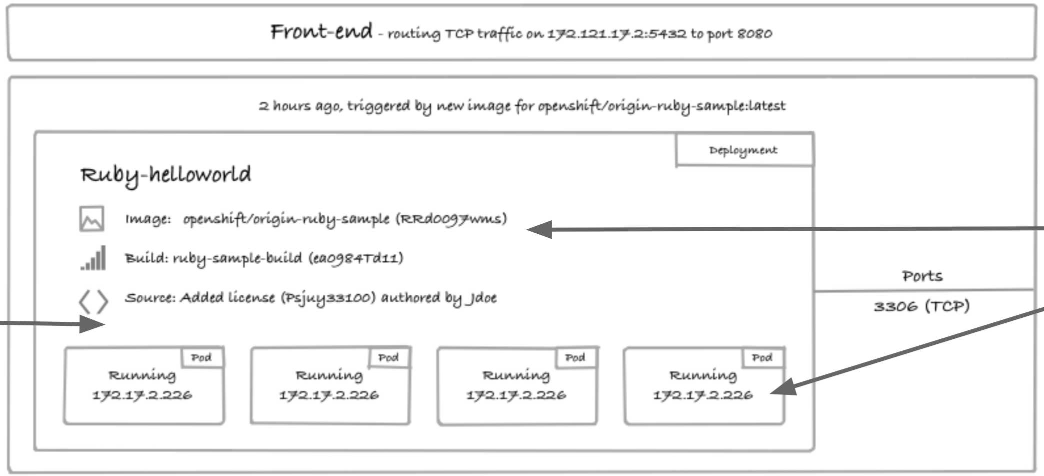
Vertical relationships between deployments should indicate time, with the most recently deployed deployments within a deployment configuration being the most important.

Interaction with the overview


Front-end - routing TCP traffic on 172.121.17.2:5432 to port 8080
2 hours ago, triggered by new image for openshift/origin-ruby-sample:latest
Deployment
Ruby-helloworld
Image: openshift/origin-ruby-sample (RRd0097wms)
Build: ruby-sample-build (ea0984Td11)
Source: Added license (Psjuy33100) authored by Jdoe
Ports
3306 (TCP)
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226
Pod
Running
172.17.2.226

Clicking on a specific resource navigates to the detailed information and status for that resource (service, image, build, source, pod, etc). In OS navigation structure this would be underneath Browse

Possible extension points within the overview



Front-end - routing TCP traffic on 172.121.17.2:5432 to port 8080

2 hours ago, triggered by new image for openshift/origin-ruby-sample:latest

Ruby-helloworld

Image: openshift/origin-ruby-sample (RRd0097wms)

Build: ruby-sample-build (ea0984Td11)

Source: Added license (Psjuy33100) authored by Jdoe

Additional details for what is running in the pods. Ex: if we know it is a JBoss image and that it has Camel routes configured then inject a link to view the Camel routes

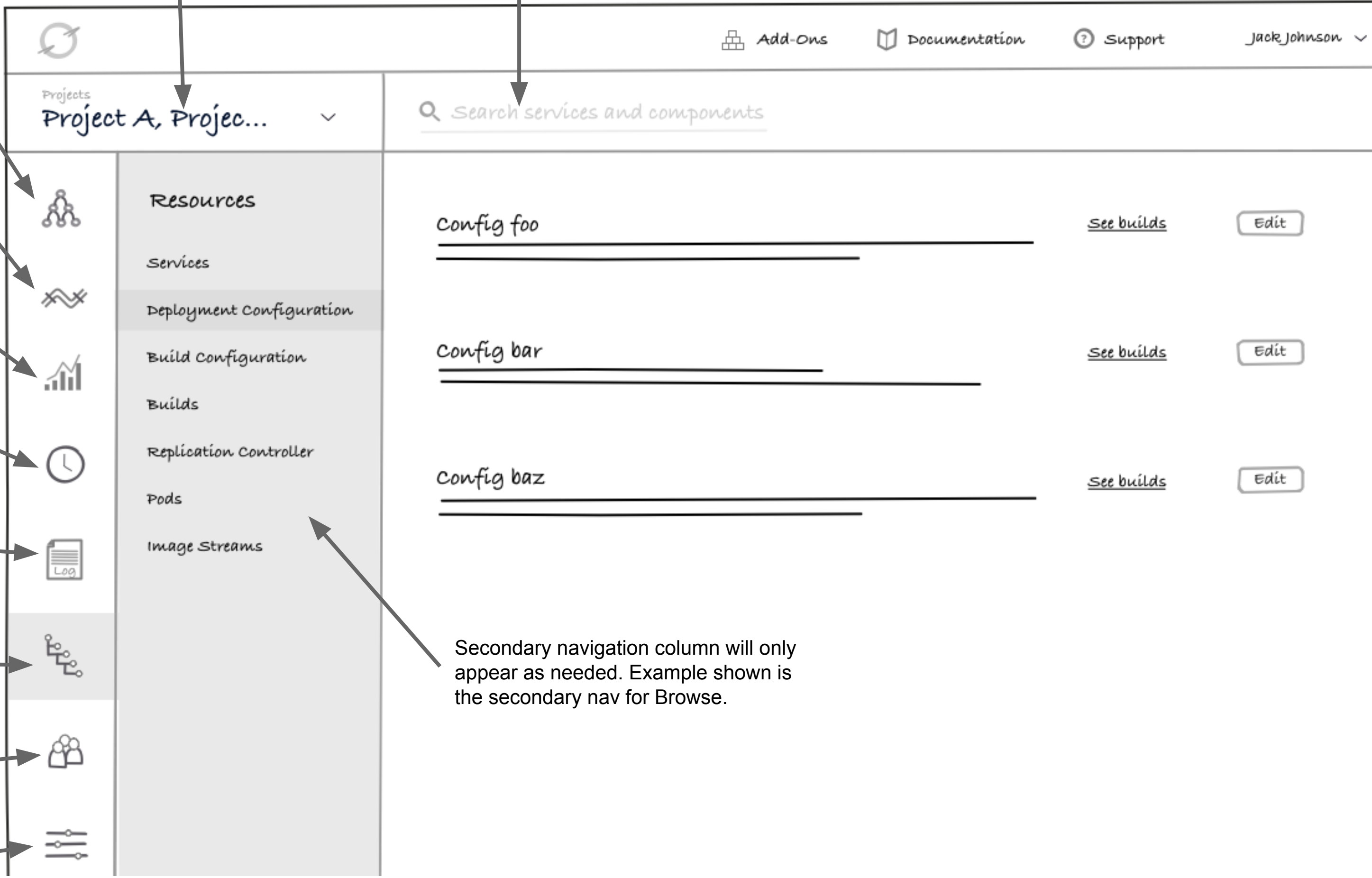
Entire sections could be swapped out with other visual representations. Pod template, pod details, etc

Area beneath the deployment may provide suggestions. Ex: I am running the Rails image but I have no database images running in my project, suggest setting up a DB service.

Navigating the console

Switch between projects, the current selected left tab is maintained when switching

Filter the current view by labels



Structural overview (default tab when going to a project)

Pipelines. This includes build pipelines (source -> build -> image), known environment pipelines (dev -> test -> production)

Metrics and monitoring. This includes both container level system resource monitoring and extended monitoring details such as JVM monitoring (think threads, etc) and event bus monitoring.

Events (timeline), important things happening within your project.

Logs. Viewing the logs for your running containers. Viewing logs for your already completed or running builds.

Browse. Dig down into lists of resources for any resource type. Ex: I want to look at a list of all the pods in my project

Membership. View/edit the users/teams that have access to this project and its resources.

Settings. General settings for the project.

Secondary navigation column will only appear as needed. Example shown is the secondary nav for Browse.

Additional wireframes live in the OpenShift Origin repo