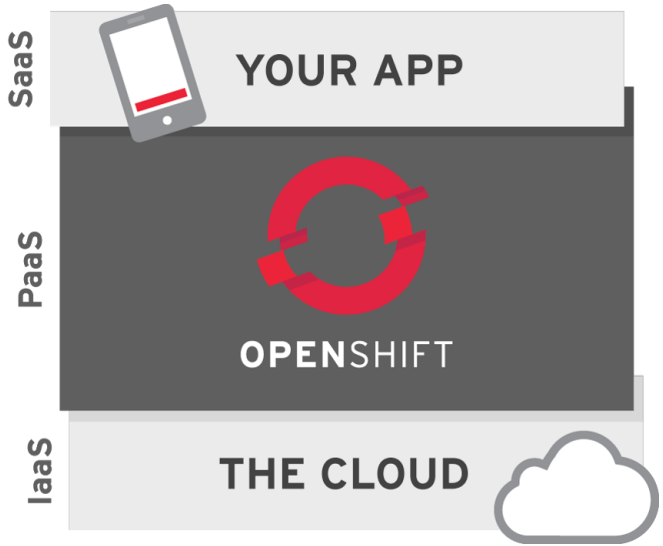
Openshift V3 as PASS

What is it!

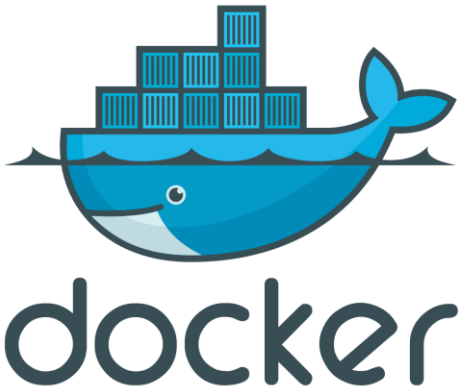
OpenShift is Red Hat's Platform-as-a-Service (PaaS) that allows developers to quickly develop, host, and scale applications in a cloud environment.



- Based on Docker Containers
- Use Kubernete Cluster Management
- Origin is the community version

What is Docker

I understand Openshift is based on docker but what is docker?



Docker containers wrap a piece of software in a complete filesystem that contains everything needed to run: **code, runtime, system tools, system libraries** – anything that can be installed on a server. This guarantees that the software will always run the same, regardless of its environment. It also ensures isolation amongst multiple containers running on the same machine.

What is Kubernetes

I know Openshift uses something called kubernetes but what is the role of kubernetes ?

Kubernetes is an open-source system which groups containers that make up an application into logical units for easy management and discovery. Kubernetes provides automating deployment, horizontal scaling, automated rollouts and rollbacks, storage orchestration, self healing, services discovery and load balancing and management of containerized applications.

Hawkular Cockpit and Heapster

Let's not forget these components which makes openshift more beautiful

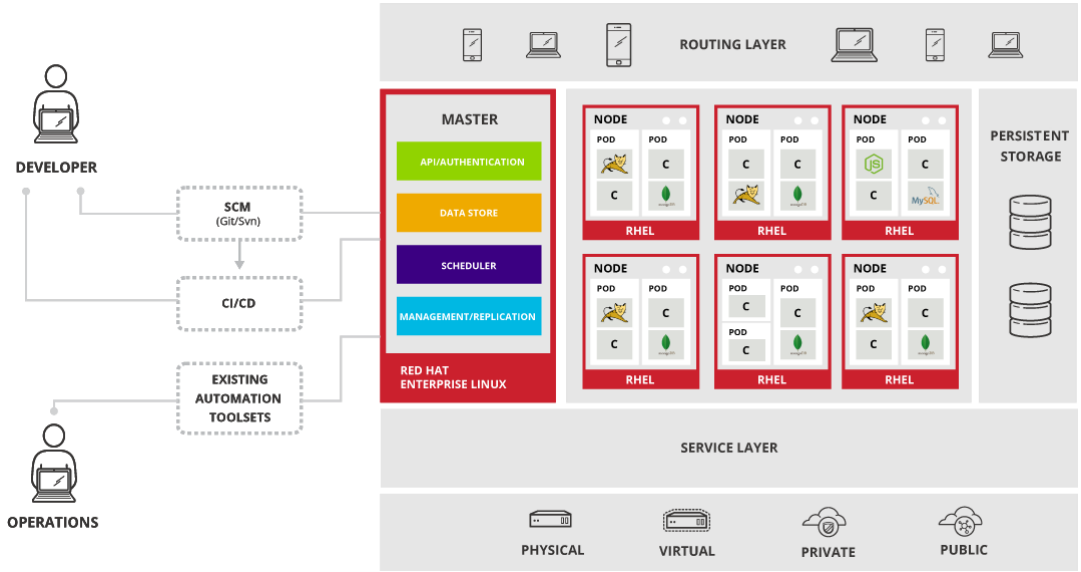
It's a hawk with a monocular. The goal is to be able to monitor things and catch anomalies in fast pace environments.

A monitoring system which allows you to monitor current values and adjust limits on system resources, control life cycle on container instances, and manipulate container images.

Heapster enables Container Cluster Monitoring and Performance Analysis. Normally used with hawkular.

How does it look – Architecture wise

A microservices-based architecture of smaller, decoupled units that work together

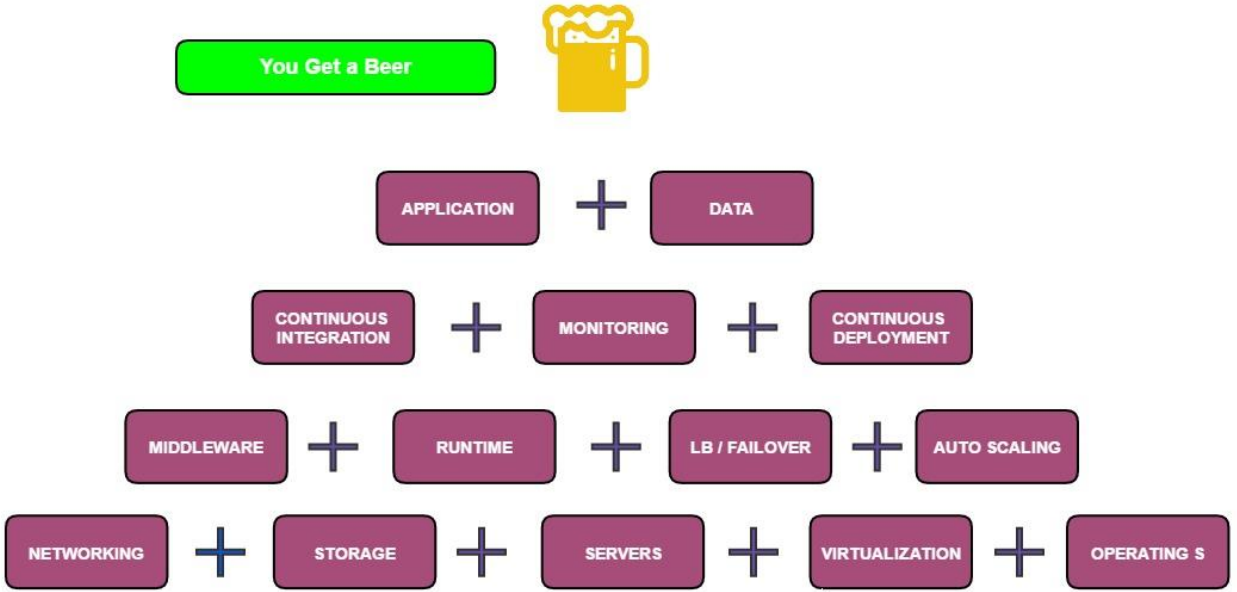


- Master for management
- Node for hosting containers

How does Openshift help

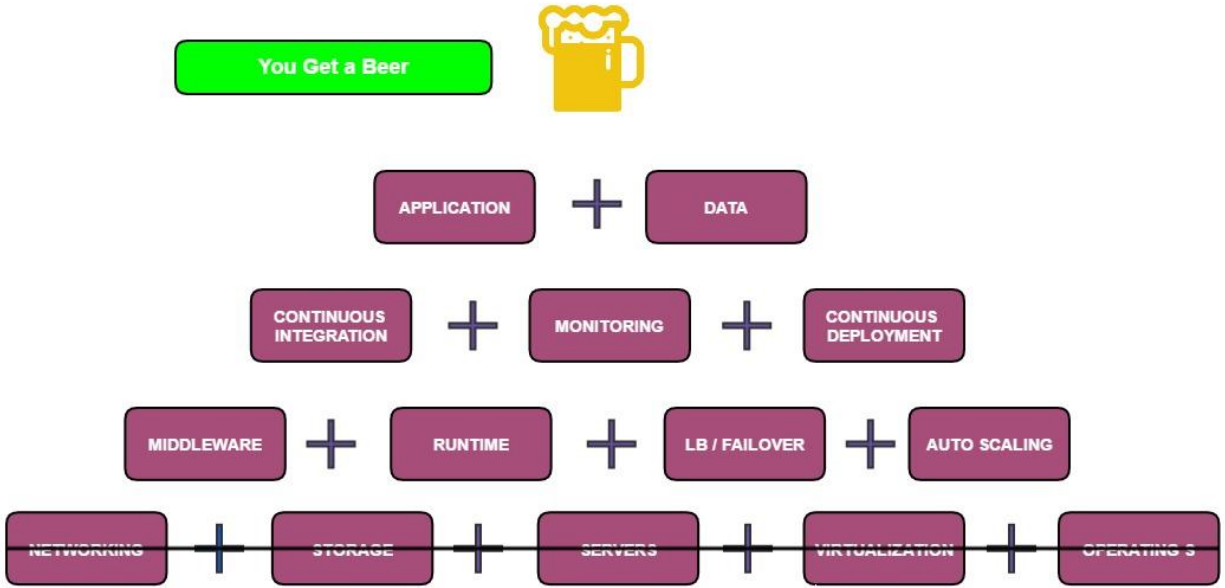
Things before IAAS !

Looks like I am not going to get the beer anytime soon ☹



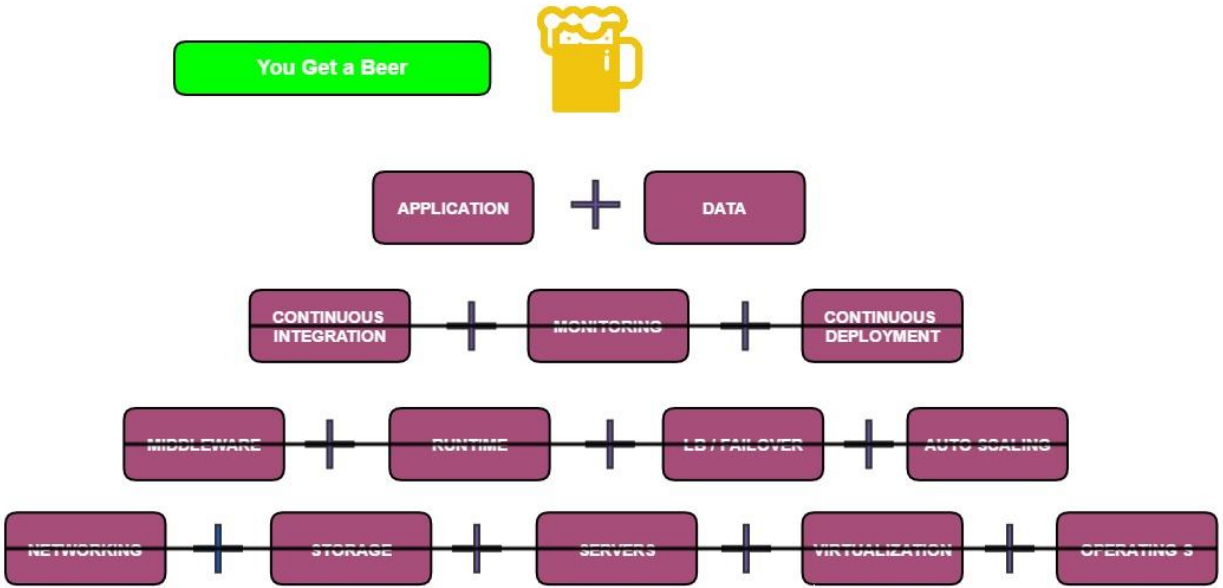
IAAS made life easier!

Still not very easy to reach to the beer mug 😐



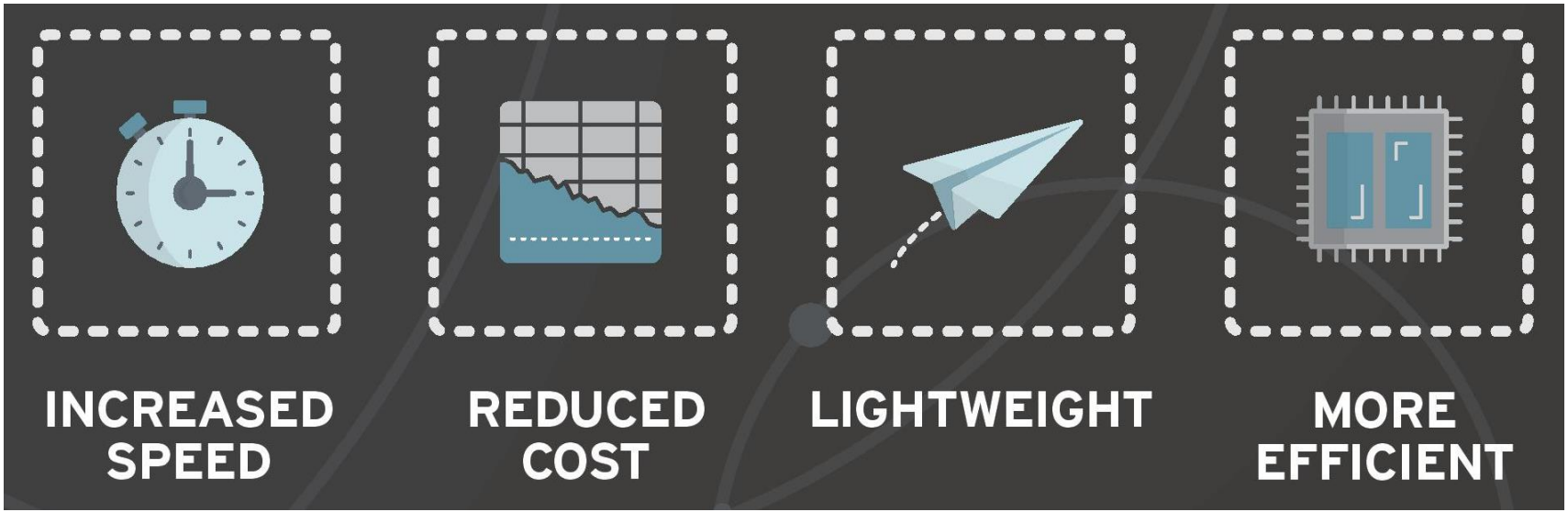
Openshift is the way !

Just the way I like it ! Beer is just a step away ☺



So this is what you get!

A 1000 feat overview of features



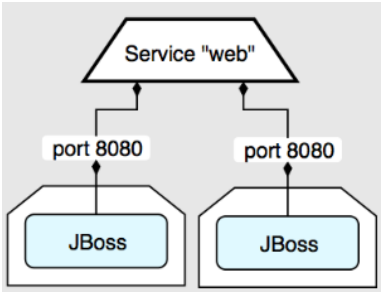
How does Openshift Work

Good to know keywords – Try not to remember

- **Container** Self-contained execution environments with their own, isolated resources
- **Images** A binary that includes all of the requirements for running a single container
- **Pods** One or more containers deployed together on one host defined, deployed, and managed as one unit
- **Services** A load balancer used to proxy the connection to internal pod/pods
- **Projects** A name space used for isolation of resources/access amongst different group of users
- **Build** The process of transforming input parameters into a running image/container
- **Image Stream** A single virtual view of related Docker-formatted container images identified by tags
- **Routes** Something that exposes a service at a host name (Ex www.example.com)
- **Templates** Used to describes a set of objects that can be parameterized and processed
- **Label** Labels are key/value pairs that are attached to objects, such as pods
- **Replication Controller** Ensures that a specified number of pod “replicas” are running at any one time
- **Deployment Configuration** It provides declarative updates for Pods and Replica Sets

Pod & Service

Pod is one or more containers deployed together on one host, and the smallest compute unit that can be defined, deployed, and managed. Pods are the rough equivalent of a machine.



Service is one or more containers deployed together on one host, and the smallest compute unit that can be defined, deployed, and managed. Pods are the rough equivalent of a machine.

Build & Image Stream

Before we start spinning out pods on Openshift we shouul know that we can either build the pod or use it directly from a Image stream

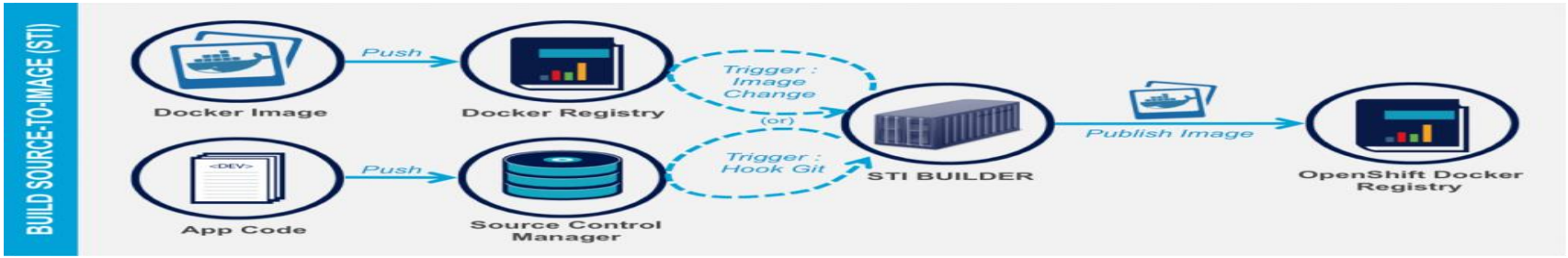
- Build A build is the process of transforming input parameters into a resulting object. Most often, the process is used to transform input parameters or source code into a runnable image. A BuildConfig object is the definition of the entire build process.

- Image Streams An image stream comprises any number of Docker-formatted container images identified by tags. It presents a single virtual view of related images and may contain images from OpenShift's integrated registry or from external docker registries.

Building an Image

Openshift uses three basic strategies to build image

- Source-to-image Strategy Source-to-Image (S2I) is a tool for building reproducible, Docker-formatted container images. It produces ready-to-run images by injecting application source into a container image and assembling a new image.

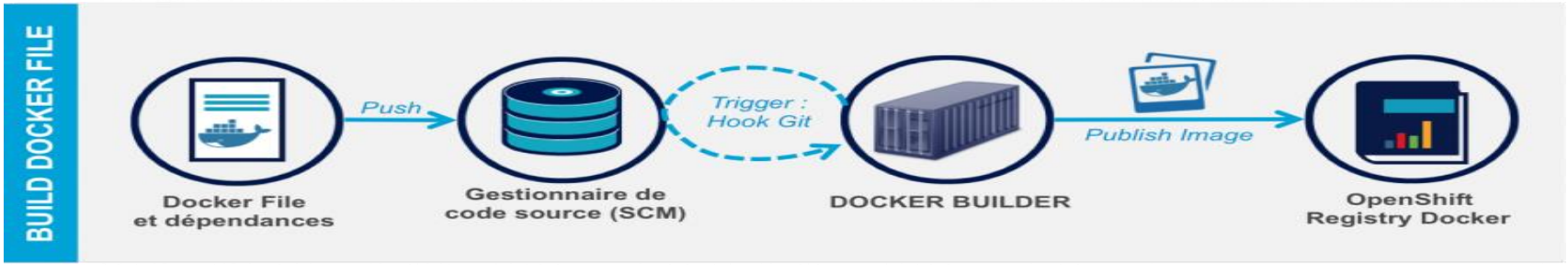


NOTE:- Don’t worry too much, all this is handled by Openshift

Building an Image -- Continue

Openshift uses three basic strategies to build image

- Docker Build The Docker build strategy invokes the docker build command, and it therefore expects a repository with a Dockerfile and all required artifacts in it to produce a runnable image.



NOTE:- Don’t worry too much, all this is handled by Openshift

Building an Image -- Continue

Openshift uses three basic strategies to build image

- Custom Build Define a specific builder image responsible for the entire build process.

NOTE:- Don’t worry too much, all this is handled by Openshift

Replication controller & Deployment

Replication controller & Deployment

- Replication Controller A replication controller ensures that a specified number of replicas of a pod are running at all times. If pods exit or are deleted, the replication controller acts to instantiate more up to the defined number.

- Deployment Deployment just creates a new replication controller and lets it start up pods. It also provide the ability to transition from an existing deployment of an image to a new one and also define hooks to be run before or after creating the replication controller.

Routes & Templates

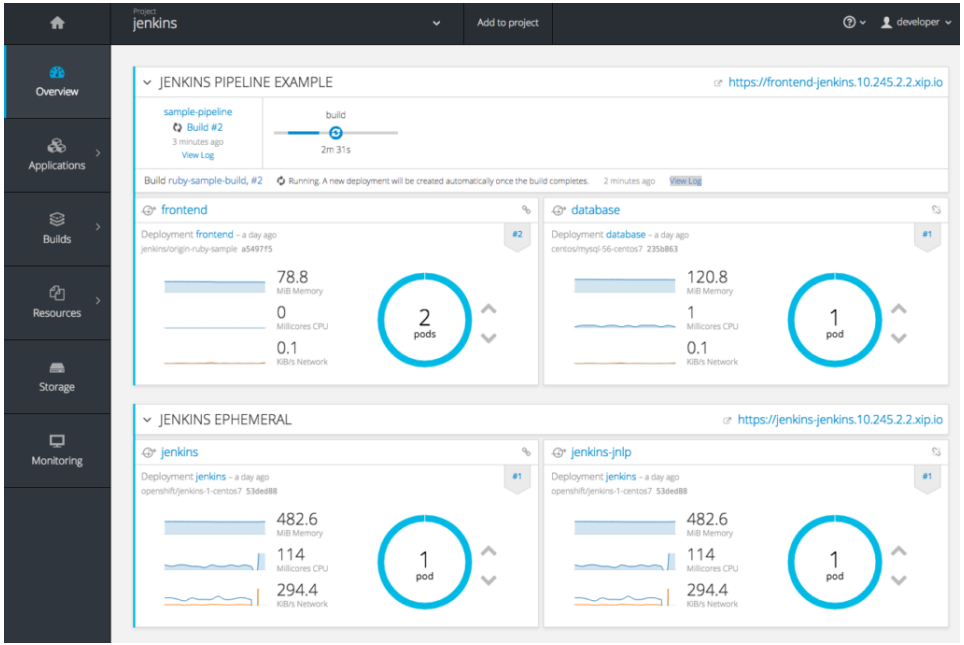
Routes & Template

- Routes An OpenShift exposes a service at a host name, like www.example.com, so that external clients can reach it by name. Simple.

- Template A template describes a set of objects that can be parameterized and processed to produce a list of objects for creation by OpenShift Enterprise. The objects to create can include anything that users have permission to create within a project, for example services, build configurations, and deployment configurations..

And a beautiful UI – Cherry on the top!

Oepnshift provides a beautiful UI to manage things from a web console.



Thank You