OpenShift

Network and Storage
- OpenShift SDN, or OVN
- Cinder, Manila, EBS, Azure Disk, Azure File, GCE PD, vsphere volume

Dev, Test, Deploy
Registry
Jenkins

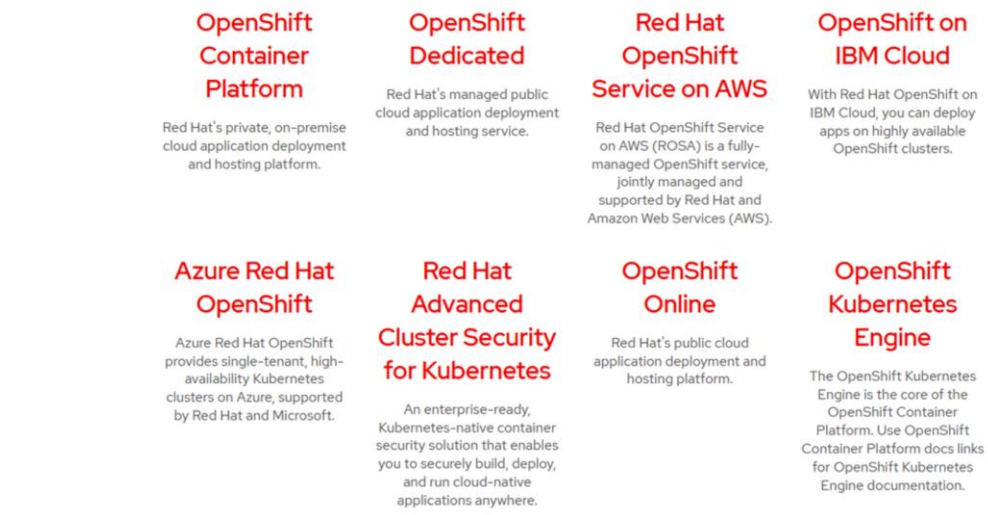
Infrastructure
CoreOS
Node mgt with Ansible
Auto-scaling

App Management
Web Console
EFK
Istio

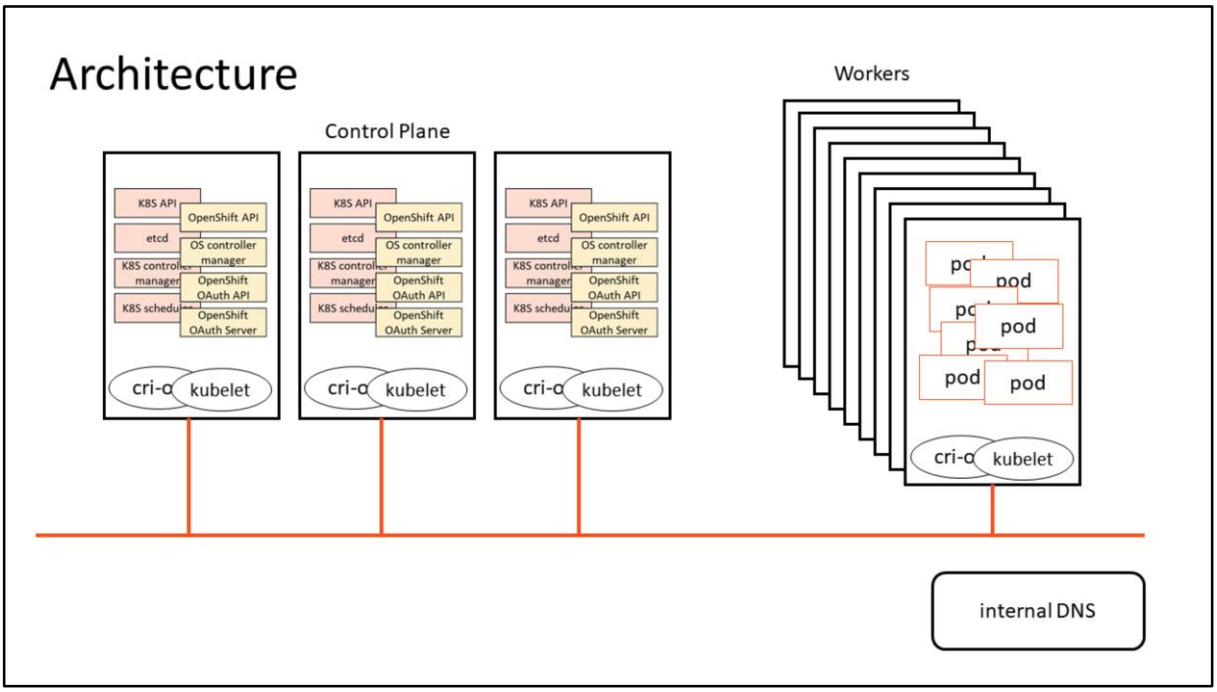
Security
RBAC
IAM
Authentication

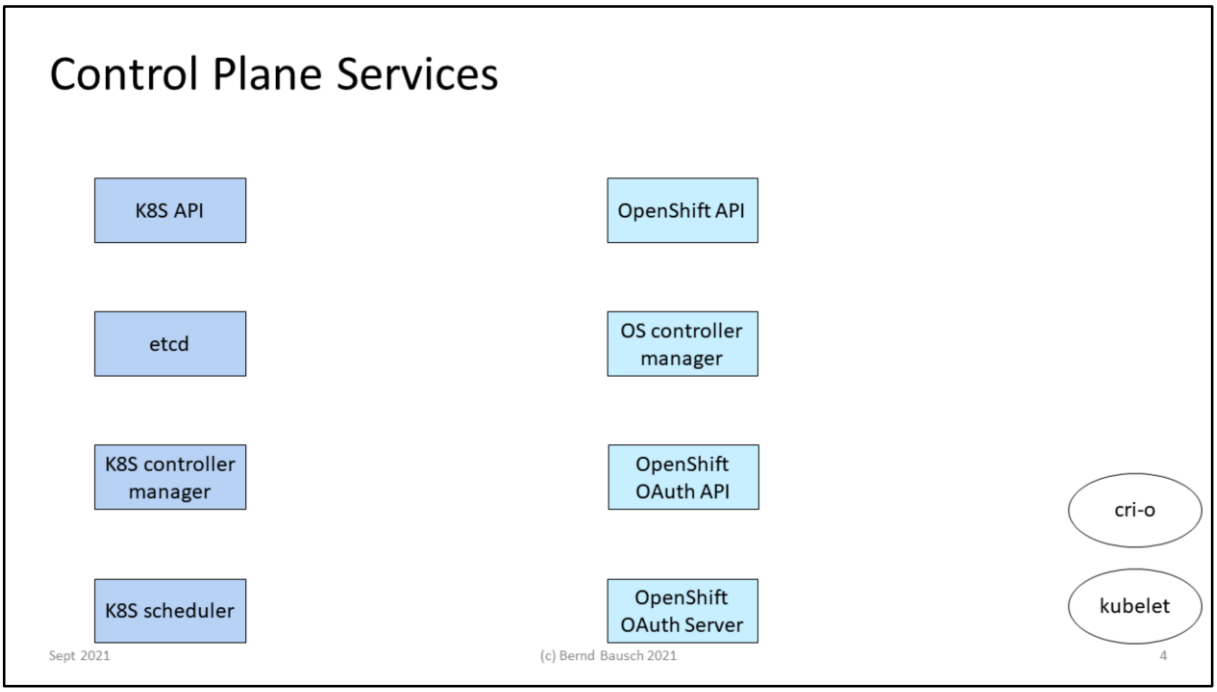
Sept 2021 (c) Bernd Bausch 2021 1

OpenShift Products

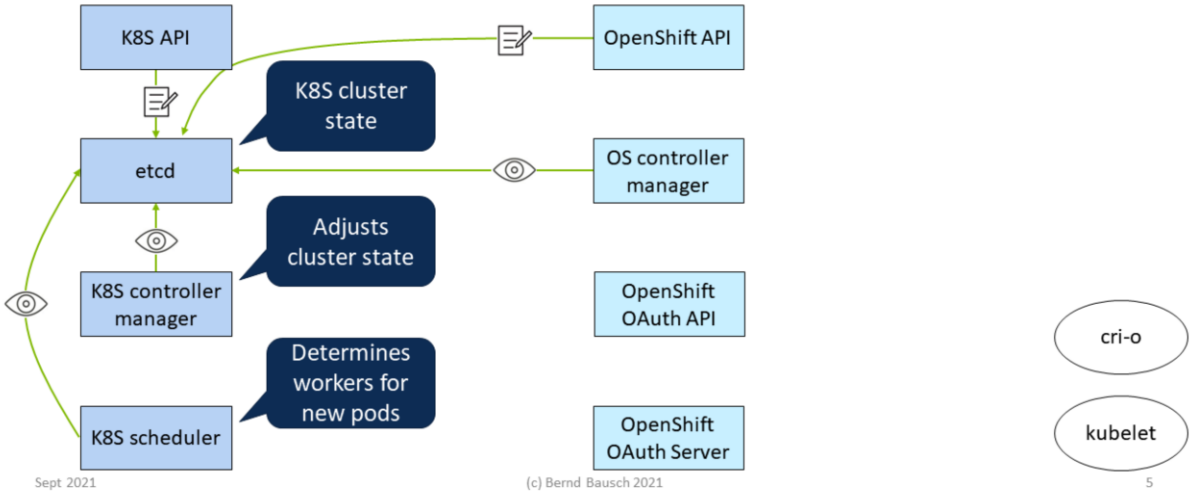


Screenshot from https://docs.openshift.com

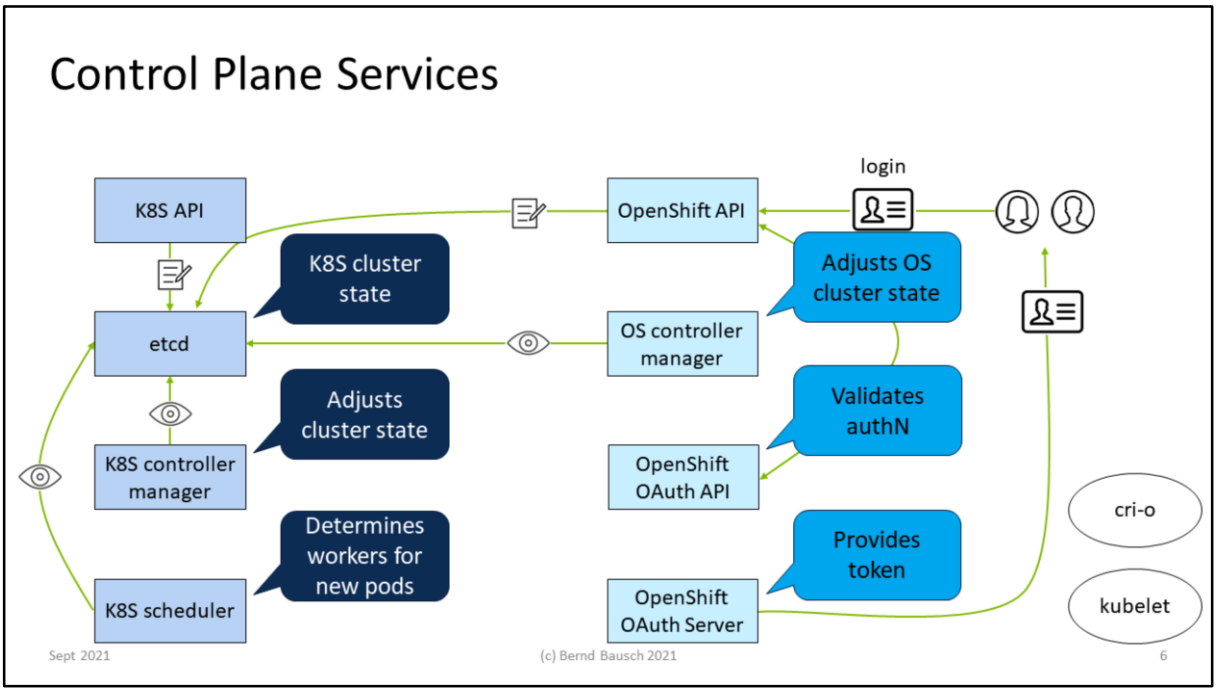

Architecture
Control Plane
K8S API
OpenShift API
etcd
OS controller manager
OpenShift OAuth API
OpenShift OAuth Server
cri-o
kubelet
Workers
pod
internal DNS

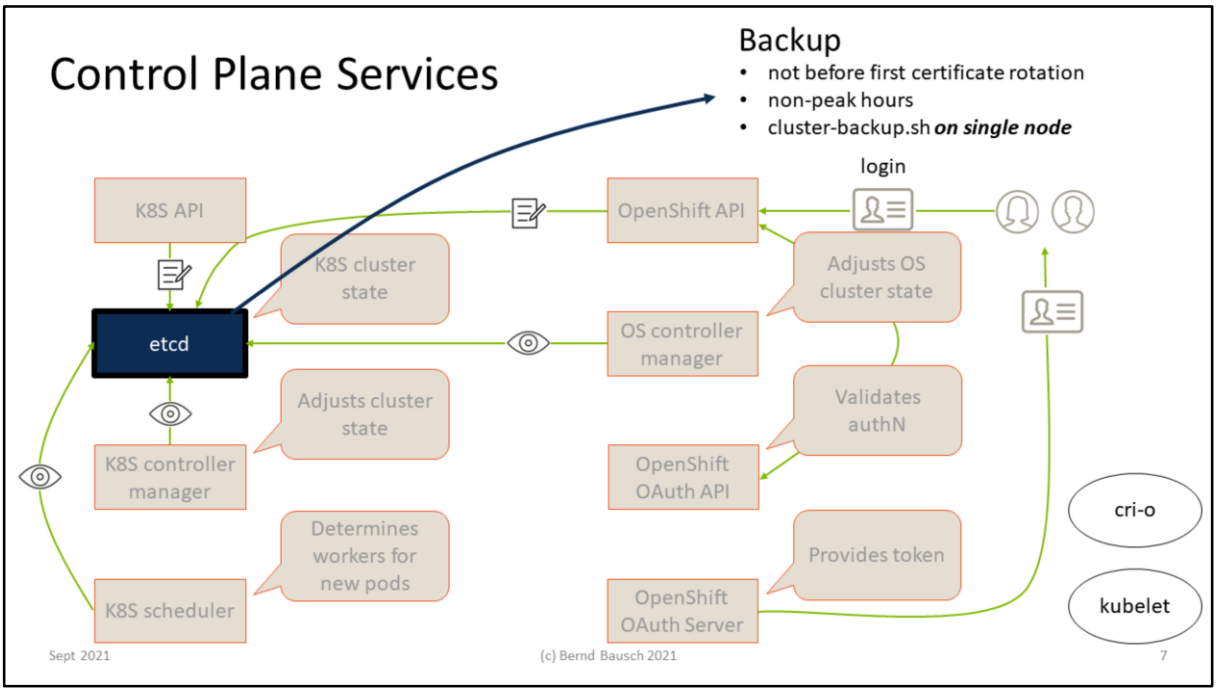

Control Plane Services
K8S API
etcd
K8S controller manager
K8S scheduler
OpenShift API
OS controller manager
OpenShift OAuth API
OpenShift OAuth Server
cri-o
kubelet
Sept 2021
(c) Bernd Bausch 2021
4

Control Plane Services



Sept 2021 (c) Bernd Bausch 2021 5


Control Plane Services
K8S API
etcd
K8S controller manager
K8S scheduler
K8S cluster state
Adjusts cluster state
Determines workers for new pods
OpenShift API
OS controller manager
OpenShift OAuth API
OpenShift OAuth Server
login
Adjusts OS cluster state
Validates authN
Provides token
cri-o
kubelet
Sept 2021
(c) Bernd Bausch 2021
6

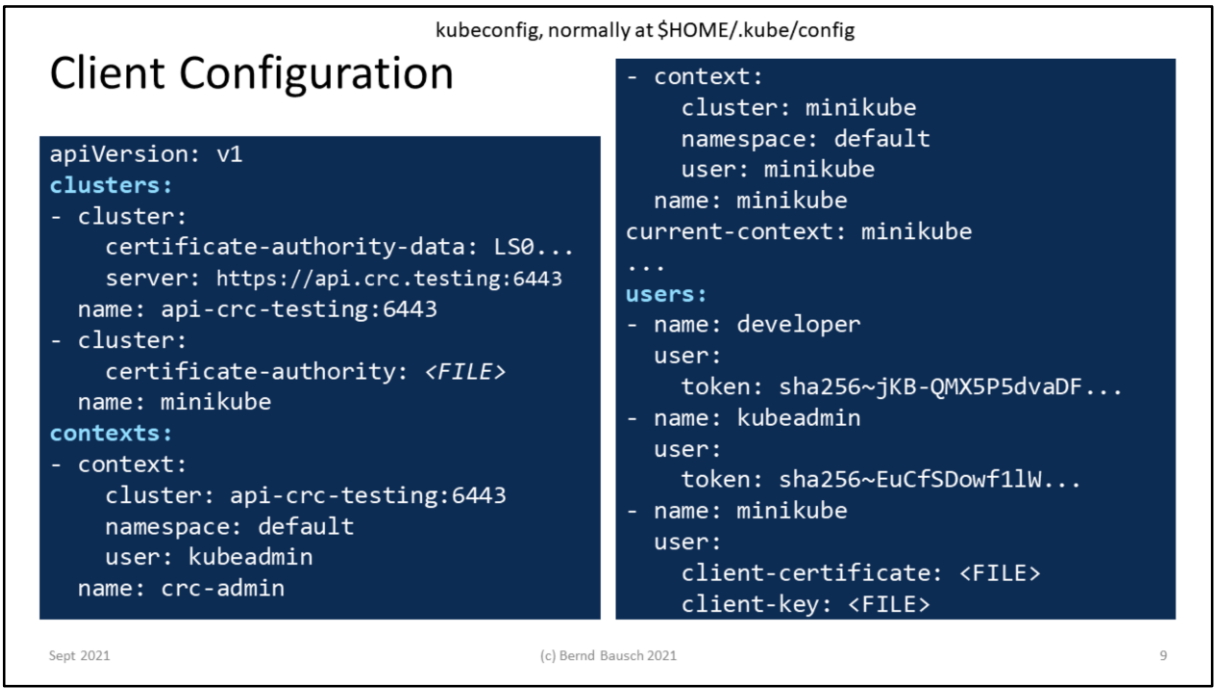

Control Plane Services
Backup
not before first certificate rotation
non-peak hours
cluster-backup.sh on single node
login
K8S API
OpenShift API
K8S cluster state
Adjusts OS cluster state
etcd
OS controller manager
Adjusts cluster state
Validates authN
K8S controller manager
OpenShift OAuth API
cri-o
Determines workers for new pods
Provides token
K8S scheduler
OpenShift OAuth Server
kubelet
Sept 2021
(c) Bernd Bausch 2021
7

The command line

oc	verb	object type	object
	get	pod	
	describe	replicaSet, rs	
	create	deployment, deploy	
	edit	daemonSet, ds	
	delete	configMap	
	apply	secret	
	explain	buildconfig	
	login	project	
	...	...	

Sept 2021 (c) Bernd Bausch 2021 8

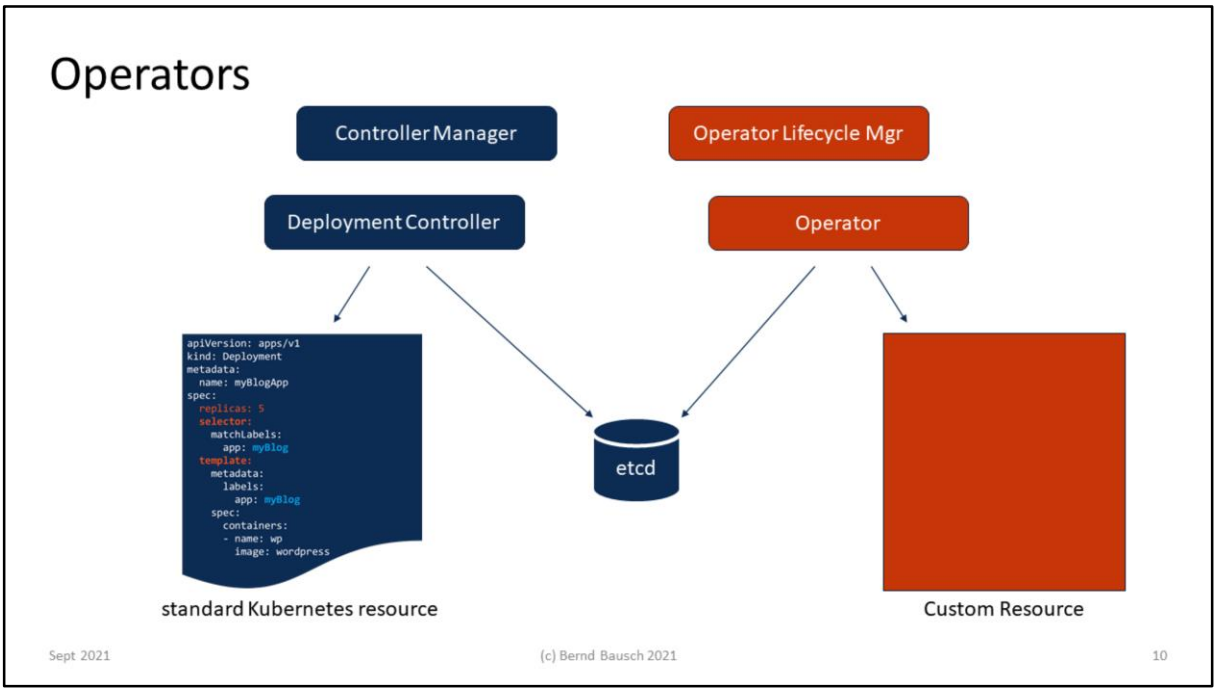
The OpenShift command line client is named oc. It is a superset of kubectl; this means that you can use all kubectl commands with oc.



Like kubectl, oc also uses a kubeconfig file ($HOME/.kube/config by default) to learn about clusters and identities.
A **cluster** is described by its API endpoint (a URL) and an X.509 certificate authority that is used for authentication. In the example, the CRC cluster (OpenShift) includes the entire CA in the config file, whereas the Minikube cluster's CA is stored in a separate crt file.

The **users** section stores users' credentials. The example uses so-called *bearer tokens* to authenticate with the OpenShift cluster and an X.509 certificate for Minikube.

A **context** is essentially a combination of cluster and user account, plus the default Kubernetes *namespace* that will be used for the login session.



An operator
- is an app-specific controller
- extends the Kubernetes API
- to create/configure/manage complex apps on behalf of a K8s user

An controller runs in a loop, comparing current with desired state. When current state deviates from the desired state, it takes action.

While a normal controller watches over standard Kubernetes resources like pods or deployments, an operator watches over Custom Resources (CR), declared by a Custom Resource Definition (CRD). A CRD lists all possible configurations of a CR.
More precisely, a CR is an **API extension**, or a new API endpoint, and a CRD is a schema for a CR.

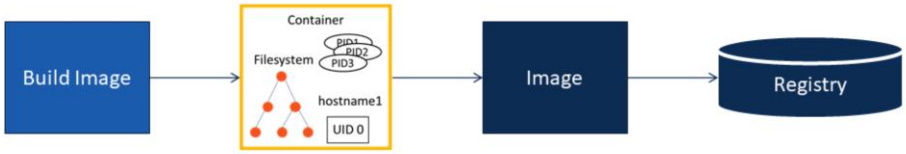
An operator may perform tasks like monitoring its application, backing up/recovering data, or updating the application.
"It encodes in software the skills of an expert administrator" (from the O'Reilly book).

Example etcd cluster. When an etcd pod crashes, Kubernetes can launch a new one, but it doesn't know how to synchronize it with the remaining etcd nodes. One could write an etcd-specific operator that has that knowledge.

Building and Deploying

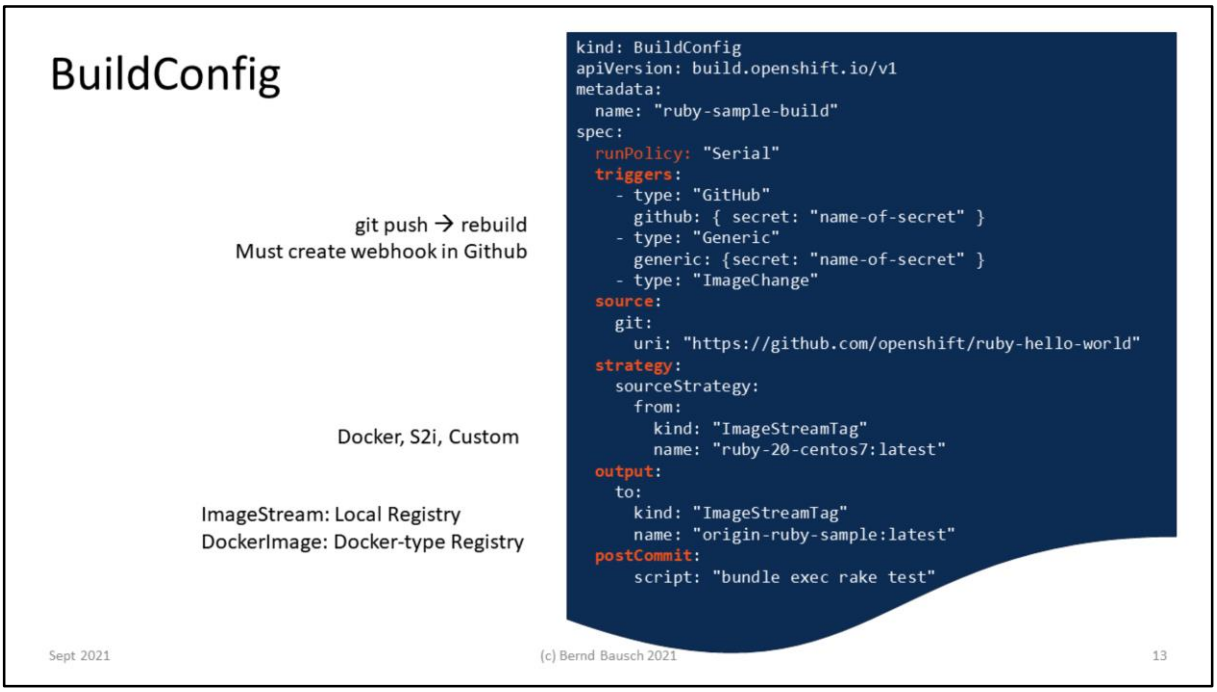
BuildConfig, DeploymentConfig, ImageStream

Build



- Docker
  Dockerfile, buildah
- Source-to-Image
  S2i image, source code, buildah
- Custom
  Custom build image, privileged users only

Sept 2021 (c) Bernd Bausch 2021 12



A BuildConfig is an OpenShift object usually described by a manifest. It describes how a build is carried out. In particular, it defines the build strategy - Docker, S2i, Custom - and strategy-specific parameters such as the build image for S2i. It also determines what happens with the build result and it can define conditions that trigger an automatic rebuild, such as the availability of a new build image version.
Build hooks are tasks that are run at certain points during the build process. The example runs a rake script after committing the result image.
The run policy determines whether several builds can take place in parallel or must be performed serially (the default).