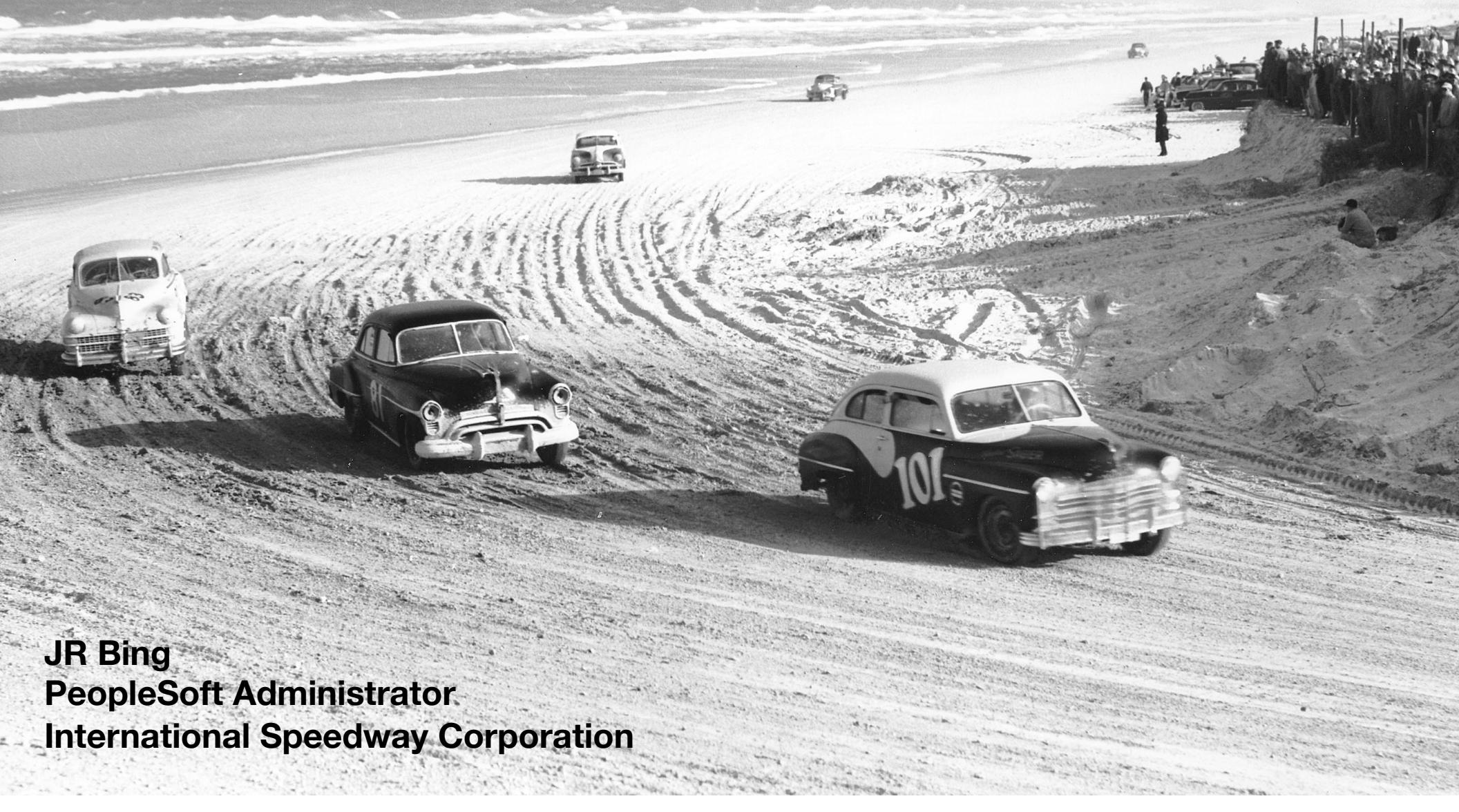


Running PeopleSoft Elasticsearch on Kubernetes



JR Bing
PeopleSoft Administrator
International Speedway Corporation

- 13 tracks across United States
- Headquarters in Daytona Beach
- PeopleSoft FN & HR
- PeopleTools 8.54 & 8.56
- Elasticsearch 2.3.2 on Kubernetes 1.10.0
- Images courtesy of motorsportsarchives.com



Agenda

- Motivation for running PeopleSoft Elasticsearch on Kubernetes
- Short background explanation of Containers, Docker, and Kubernetes
- Overview of deployment lifecycle
- Demonstration
- Pitfalls to avoid
- Considerations to make when planning a Kubernetes deployment
- Implications of containerization on the rest of the PeopleSoft stack

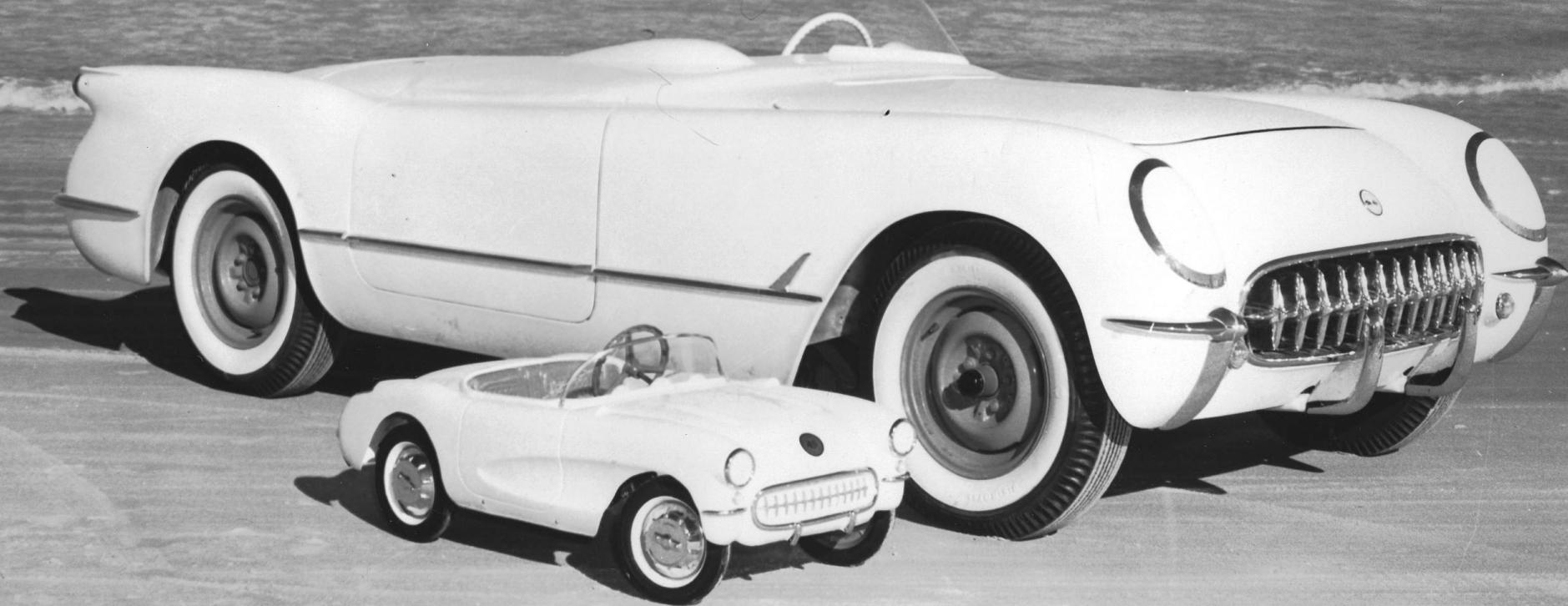
Motivation

- Explosive growth of containers & Docker
- Emergence of Kubernetes as de facto cluster scheduler
- Desire to gain experience with Containers / Docker / Kubernetes
- Migration to 8.56 using Elasticsearch
- Prior experience with Elasticsearch
- Depth of preexisting resources documenting how to run vanilla Elasticsearch on Kubernetes

Background



Containers



*more specifically; Docker containers on Linux

Containers

- Type of OS virtualization
- Enabling technologies
 - cgroups - provide limitation and prioritization of resources
 - namespaces - provide an isolated environment for applications
- Advantages over hardware virtualization
 - significantly lower overhead
 - better resource utilization
 - quicker startup

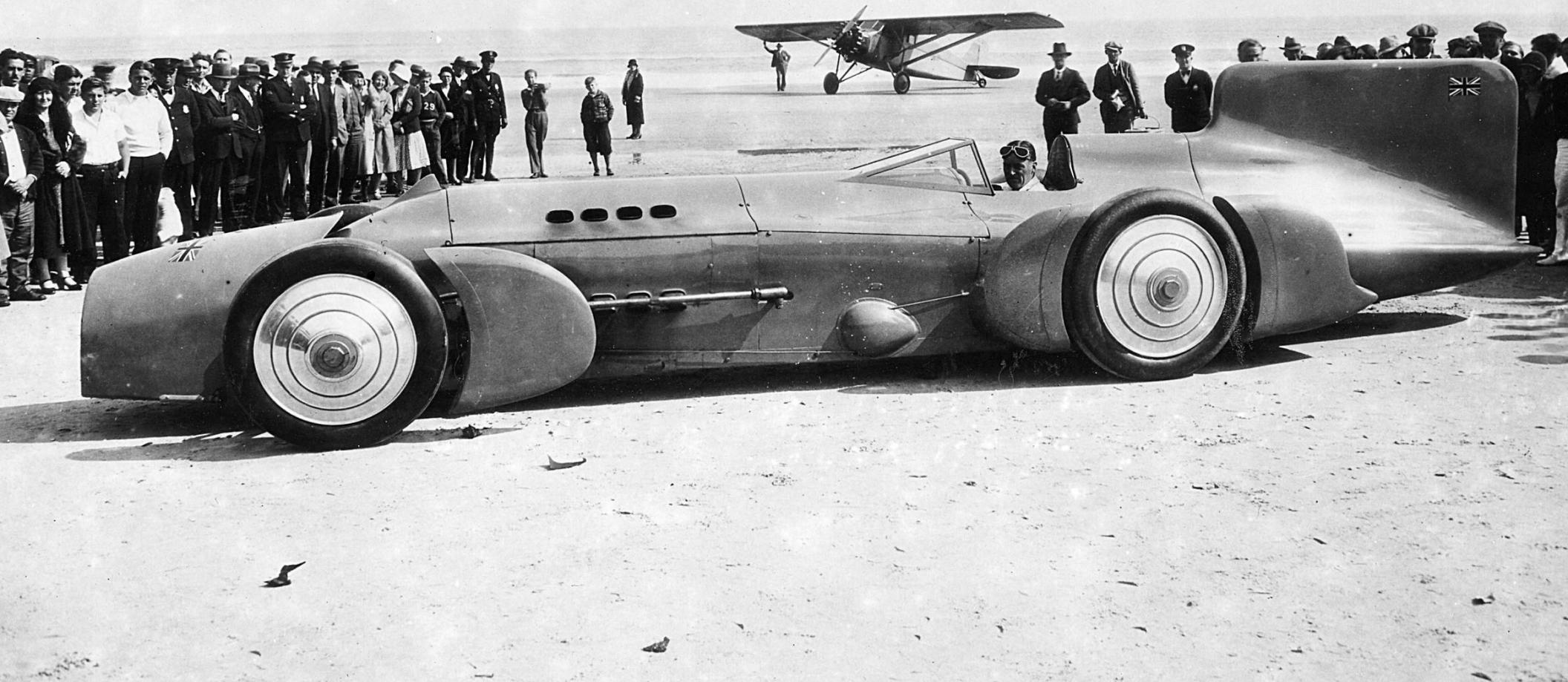
Docker



Docker

- Released March 2013
- Provides a user accessible command line utility for building, running, and distributing container images
- Dockerfile for building docker images
- Registry for publishing and retrieving images

Kubernetes



Kubernetes

- Version 1.0 released July, 2015
- Influenced by Google's "Borg" scheduler
- Has become the most widely accepted container scheduler
- Supported by all major cloud vendors



EVERYTHING KUBERNETES:

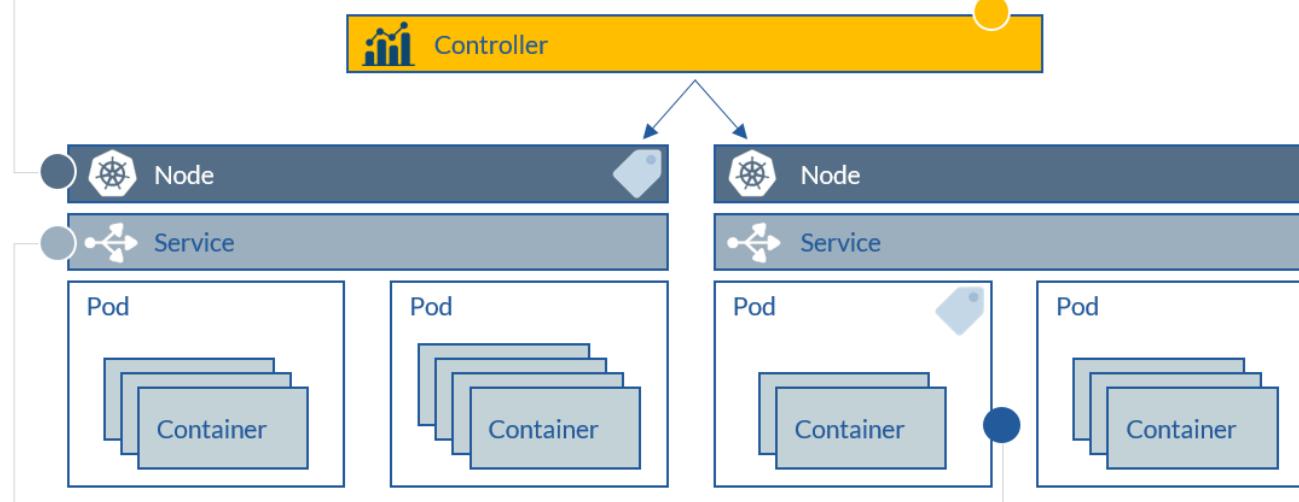
The building blocks

NODE

A Kubernetes cluster consists of one or more nodes. The nodes are bare-metal servers, on-premises VMs, or VMs on a cloud provider.

CONTROLLER

Manages a set of pods and ensures that the cluster is in the specified state.



SERVICE

A service uses a selector to define a logical group of pods and defines a policy to access them. There are several types of services in Kubernetes, including ClusterIP, NodePort, LoadBalancer.

LABEL

A label is a key/value pair that is attached to Kubernetes resource, for example, a pod.

SELECTOR

A label selector can be used to organize Kubernetes resources that have labels.

POD

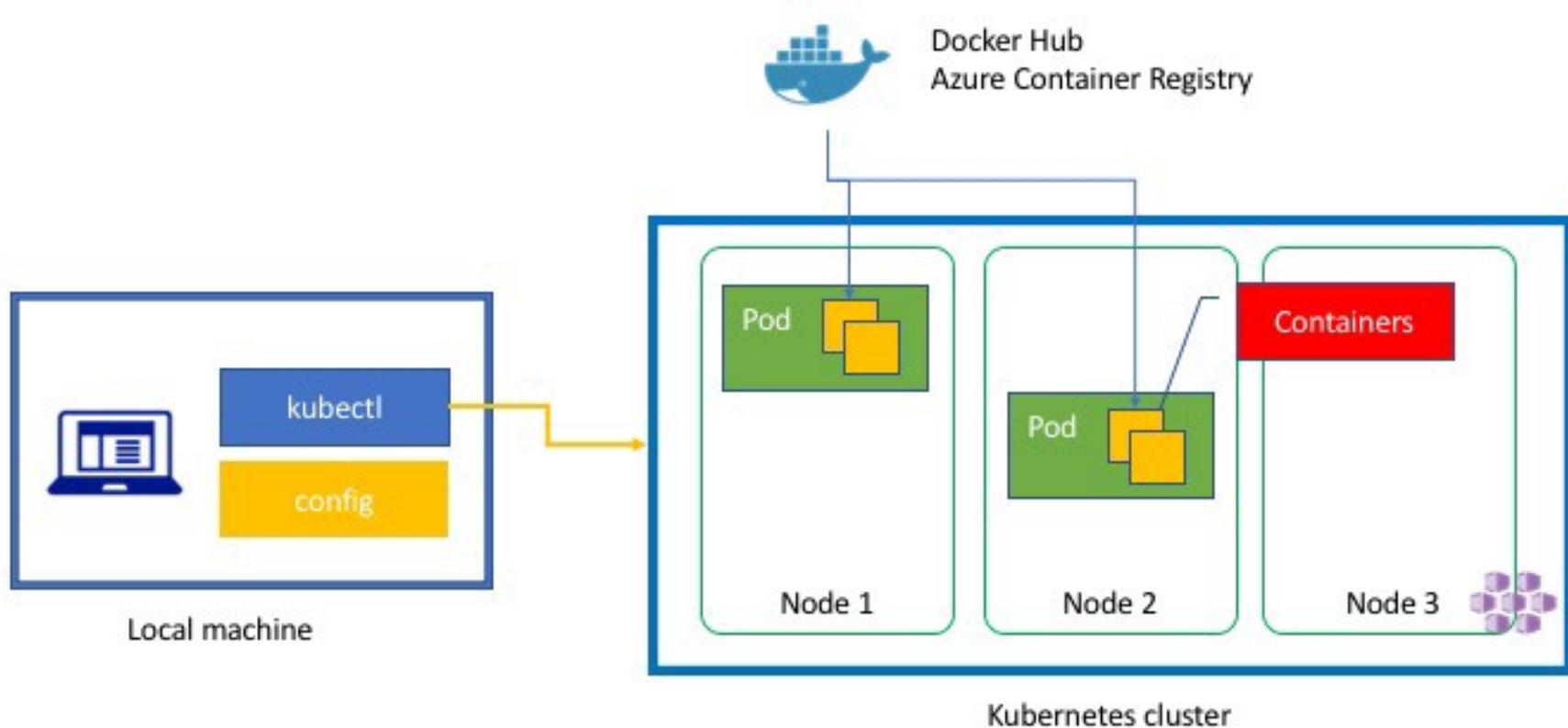
A pod is a logical group of one or more containers that share the same IP address and port space. The main purpose of a pod is to support co-located processes, such as an application server and its local cache.



- StatefulSets - high level workload object that manages the deployment and scaling of stateful applications
 - stable, unique network identifiers
 - stable, persistent storage
- Secrets - a way to store sensitive information in the Kubernetes API, which can then be accessed by pods
 - can be used by a pod as mounted files or environment variables

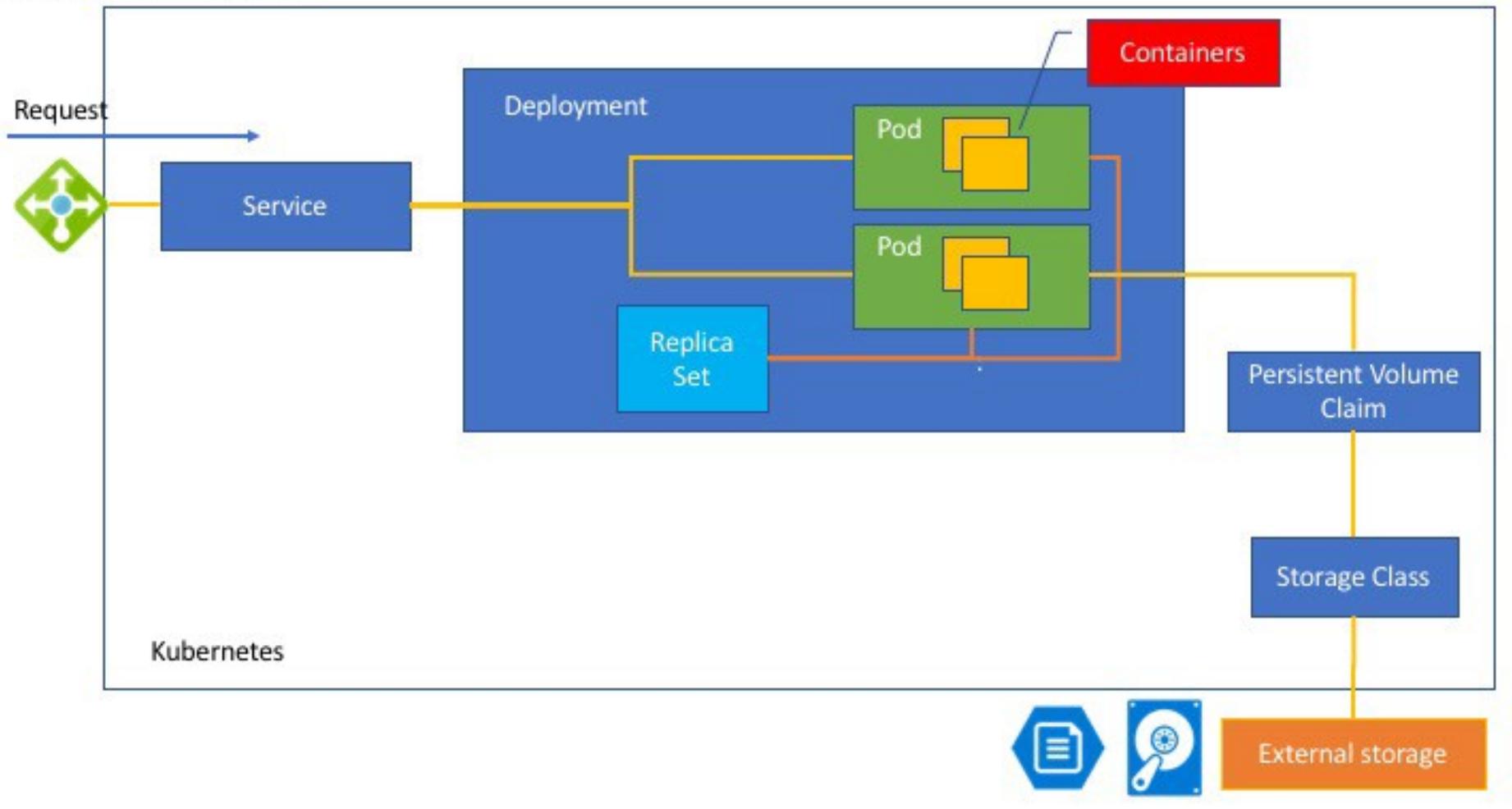


Deployment



<https://medium.com/@tsuyoshiushio/kubernetes-in-three-diagrams-6aba8432541c>

Kubernetes Objects



<https://medium.com/@tsuyoshiushio/kubernetes-in-three-diagrams-6aba8432541c>



```
1 # Dockerfile
2 FROM oraclelinux:7-slim
3
4 LABEL maintainer="JR Bing <jr@jrbing.com>" \
5       base.image="oraclelinux:7-slim" \
6       version="1.0"
7
8 # Set environment variables
9 ENV ES_VERSION="2.3.2" \
10    ES_JAVA_OPTS="-Xms512m -Xmx512m" \
11    JAVA_VERSION_MAJOR=8 \
12    JAVA_VERSION_MINOR=0 \
13    JAVA_VERSION_BUILD=144 \
14    JAVA_HOME=/opt/jre \
15    LANG="en_US.utf8"
16
17 # Set derived environment variables
18 ENV ES_ARCHIVE_FILE="pt-elasticsearch${ES_VERSION}.tgz" \
19    JAVA_ARCHIVE_FILE="pt-jre1.${JAVA_VERSION_MAJOR}.${JAVA_VERSION_MINOR}_${JAVA_VERSION_BUILD}.tgz" \
20    JCE_ARCHIVE_FILE="jce_policy-${JAVA_VERSION_MAJOR}.zip" \
21    PATH="${PATH}:/elasticsearch/bin:/opt/jre/bin"
22
23 # Define the volume to be used for Elasticsearch data
24 VOLUME ["/data"]
25
26 # Expose the http and node transport ports
27 EXPOSE 9200 9300
```



```
1 # Dockerfile (continued)
2 # Copy and unpack java archive
3 ADD ${JAVA_ARCHIVE_FILE} /opt/jre
4
5 # Download and install the unlimited jce policy from Oracle, update the java
6 # networkaddress.cache.ttl setting so that successfull lookups correctly expire
7 # after 10 seconds, and cleanup unnecessary java binaries & libraries.
8 RUN set -ex \
9     && curl -jksSLH "Cookie: oraclelicense=accept-securebackup-cookie" -o /tmp/jce_policy-${JAVA_VERSION_MAJOR}.zip \
10    http://download.oracle.com/otn-pub/java/jce/${JAVA_VERSION_MAJOR}/jce_policy-${JAVA_VERSION_MAJOR}.zip \
11    && cd /tmp \
12    && unzip /tmp/jce_policy-${JAVA_VERSION_MAJOR}.zip \
13    && cp -v /tmp/UnlimitedJCEPolicyJDK8/*.jar /opt/jre/lib/security/ \
14    && sed -i 's/#networkaddress.cache.ttl=-1/networkaddress.cache.ttl=10/' ${JAVA_HOME}/lib/security/java.security \
15    && rm -rf /opt/jre/plugin \
16        # ....
17        /tmp/*
```

```
 1 # Dockerfile (continued)
 2 # Create the Elasticsearch user and group
 3 RUN set -ex \
 4     && mkdir /elasticsearch \
 5     && groupadd -g 1000 elasticsearch \
 6     && adduser -u 1000 -g 1000 -d /elasticsearch elasticsearch
 7
 8 # Copy and unpack the Elasticsearch archive
 9 ADD --chown=1000:1000 ${ES_ARCHIVE_FILE} /elasticsearch
10
11 # Set ownership on Elasticsearch files & data, set executable bit
12 # on Elasticsearch binaries, and install additional Elasticsearch
13 # plugins.
14 RUN set -ex \
15     && chown -R elasticsearch:elasticsearch /elasticsearch \
16     && chown -R elasticsearch:elasticsearch /data \
17     && chmod +x /elasticsearch/bin/elasticsearch \
18     && chmod +x /elasticsearch/bin/elasticsearchuser \
19     && chmod +x /elasticsearch/bin/plugin \
20     && /elasticsearch/bin/plugin install cloud-azure --verbose \
21     && /elasticsearch/bin/plugin install io.fabric8/elasticsearch-cloud-kubernetes/${ES_VERSION} --verbose
22
23 # Set the current working directory
24 WORKDIR /elasticsearch
25
26 # Copy the Elasticsearch configuration files
27 COPY --chown=1000:1000 config /elasticsearch/config
28
29 # Copy the script to start the Elasticsearch instance
30 COPY run.sh /
31
32 # By default, run the Elasticsearch startup script
33 CMD ["/bin/bash", "-c", "/run.sh"]
```

```
1 # secrets.yaml
2 ---
3 apiVersion: v1
4 kind: Secret
5 metadata:
6   name: pses-user
7 type: Opaque
8 data:
9   # echo "people" | base64
10  username: cGVvcGx1Cg==
11  password: cGVvcGx1Cg==
12 ---
13 apiVersion: v1
14 kind: Secret
15 metadata:
16   name: pses-admin
17 type: Opaque
18 data:
19   # echo "esadmin" | base64
20  username: ZXNhZG1pbgo=
21  password: ZXNhZG1pbgo=
```



```
1 # services.yaml
2 ---
3 apiVersion: v1
4 kind: Service
5 metadata:
6   name: pselasticsearch-discovery
7   labels:
8     app: pselasticsearch
9     component: pselasticsearch
10    role: master
11 spec:
12   type: ClusterIP
13   selector:
14     component: pselasticsearch
15     role: master
16   ports:
17     - name: transport
18       port: 9300
19       protocol: TCP
```

```
1 # stateful-set.yaml
2 ---
3 apiVersion: apps/v1
4 kind: StatefulSet
5 metadata:
6   name: pselasticsearch
7   labels:
8     app: pselasticsearch
9 spec:
10   selector:
11     matchLabels:
12       component: pselasticsearch
13       role: master
14   serviceName: pselasticsearch
15   replicas: 1
16   template:
17     metadata:
18       labels:
19         component: pselasticsearch
20         role: master
```



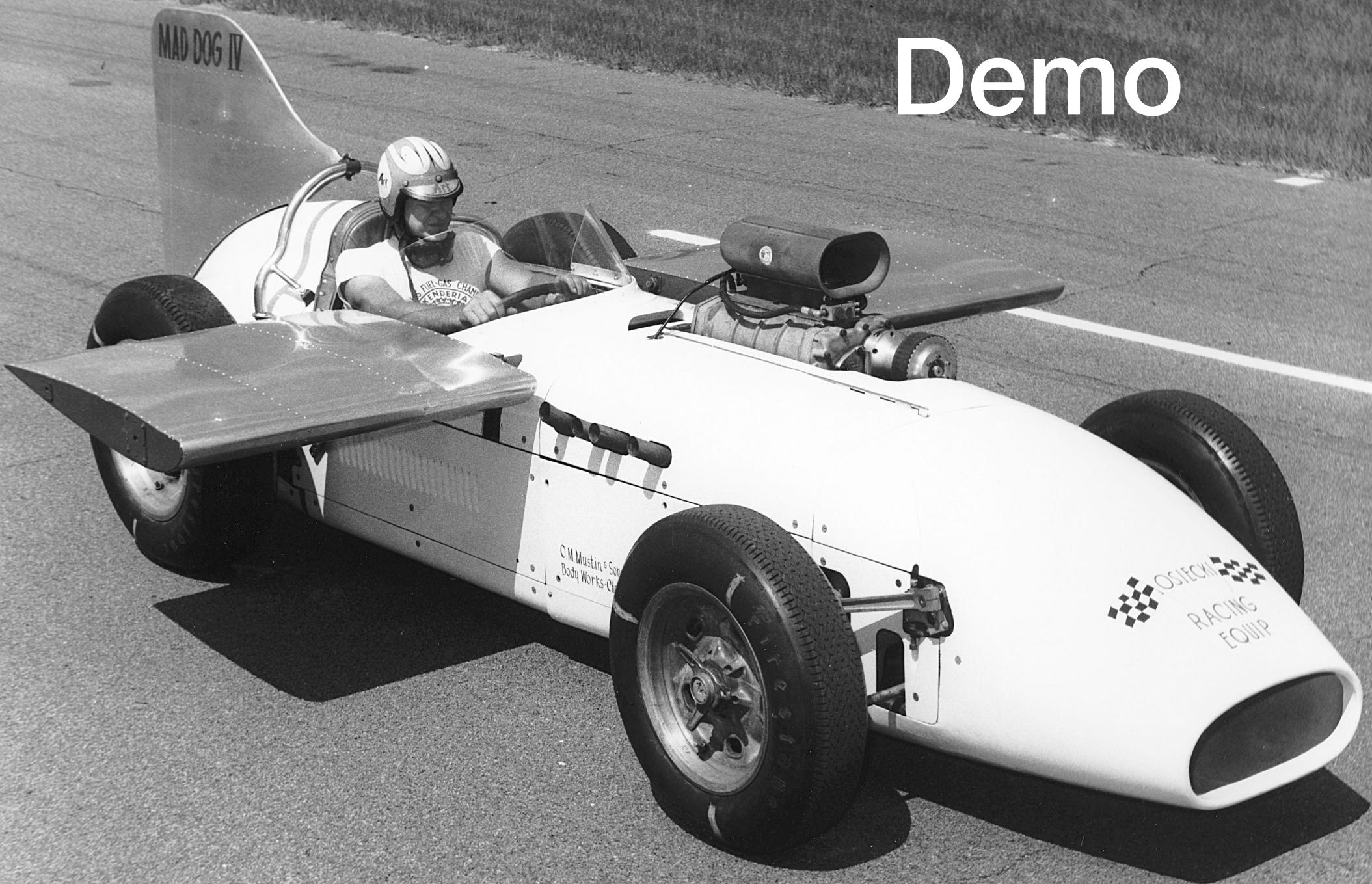
```
1 # stateful-set.yaml (continued)
2   - name: ES_JAVA_OPTS
3     value: "-Xms1024m -Xmx1024m"
4   - name: ESADMIN_USER
5     valueFrom:
6       secretKeyRef:
7         name: pses-admin
8         key: username
9   - name: ESADMIN_PASSWORD
10    valueFrom:
11      secretKeyRef:
12        name: pses-admin
13        key: password
```



```
1 # stateful-set.yaml (continued)
2     resources:
3         requests:
4             memory: 1024Mi
5         ports:
6             - containerPort: 9300
7                 name: transport
8                 protocol: TCP
9         livenessProbe:
10            tcpSocket:
11                port: 9300
12                initialDelaySeconds: 20
13                periodSeconds: 10
```

```
1 # loadbalancer.yaml
2 ---
3 apiVersion: v1
4 kind: Service
5 metadata:
6   name: pselasticsearch-lb
7   labels:
8     app: pselasticsearch
9     component: pselasticsearch
10    role: master
11 spec:
12   type: LoadBalancer
13   selector:
14     component: pselasticsearch
15     role: master
16   ports:
17     - name: http
18       port: 9200
19       targetPort: 9200
20       protocol: TCP
```

Demo



Pitfalls



- Single process model
- Change in toolsets used for administration and troubleshooting
- Registry, Network, Ingress, & Storage not included with vanilla kubernetes
- Docker updates may break compatibility
- Options for self-hosted container native storage not yet mature

A black and white photograph of a vintage racing car, likely from the early 20th century. The car features a large, multi-cylinder engine with prominent cooling fins. A driver wearing a helmet and goggles is seated behind the steering wheel. Several spectators, including men in hats and a woman, are visible in the background.

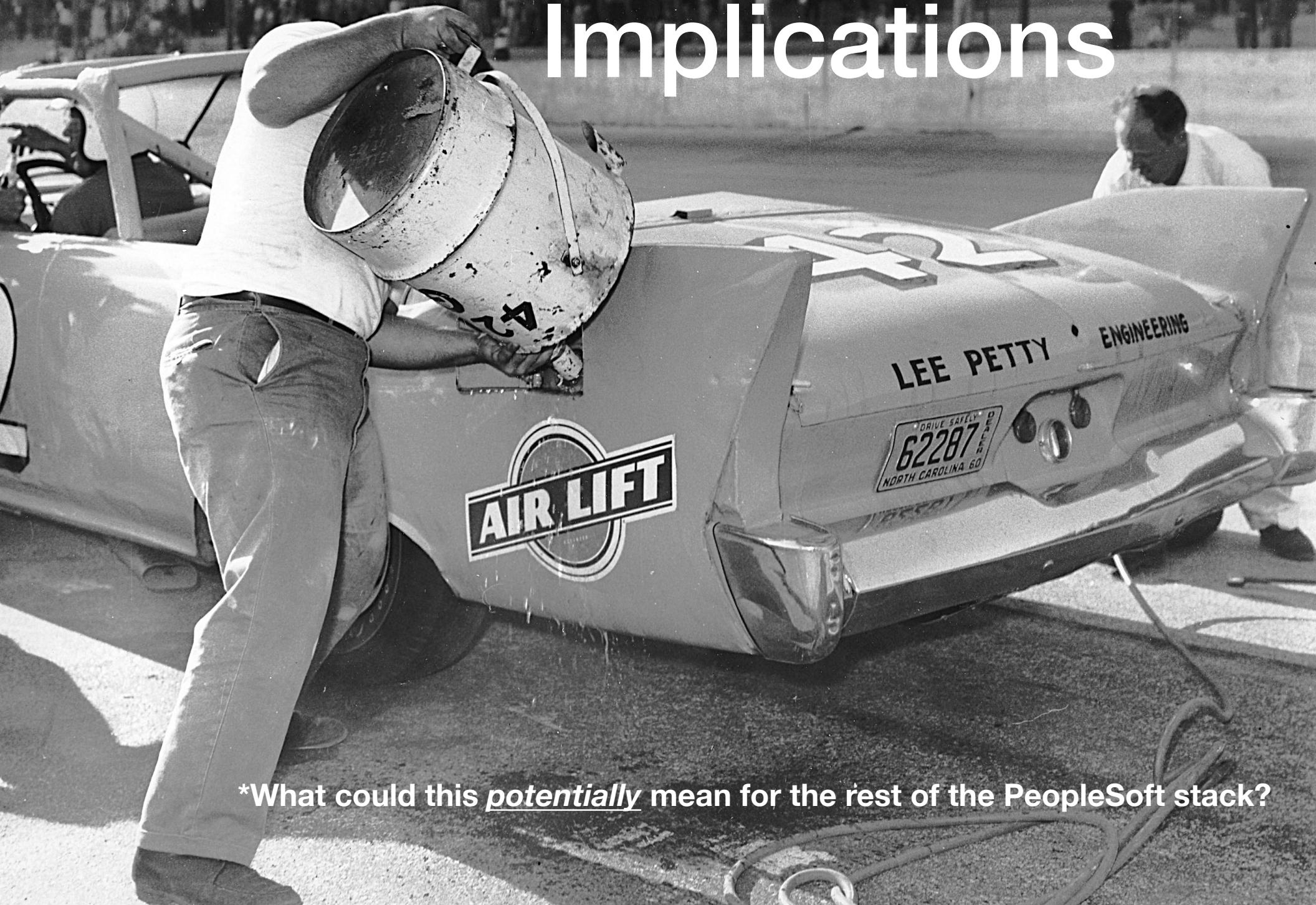
Considerations

- Onsite vs. Cloud - where to run Kubernetes
- Network - how to connect pods
- Storage - how to store persistent data
- Ingress - how to get traffic in/out of cluster
- Registry - where to store container images

Additional Components

- Helm - package management
- Prometheus - metrics and alerting
- Grafana - monitoring and visualization
- Ark - backup and recovery
- Clair - vulnerability scanner for container images

Implications



*What could this *potentially* mean for the rest of the PeopleSoft stack?

- Automated build & deployment pipelines
- Rolling updates for patches and configuration changes
- Operators and Custom Resources
 - <https://github.com/oracle/weblogic-kubernetes-operator>

Questions?



Thanks!



<https://github.com/jrbing/pses-kube-presentation>