Deploying Distributed Databases and In-Memory Computing Platforms with Kubernetes

Denis Magda
GridGain Product Management
Apache Ignite PMC Chair

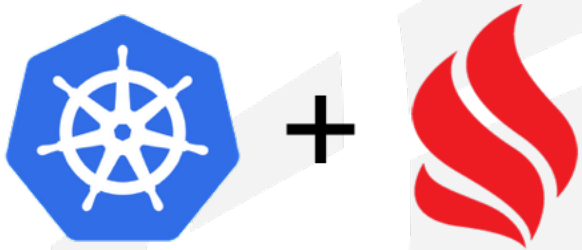
Agenda

- Deployment Specificities with Kubernetes
- Memory-Only Deployments
- Stateful Deployments
- Management and Monitoring
- Demo
- Q&A

Deployment Specificities with Kubernetes

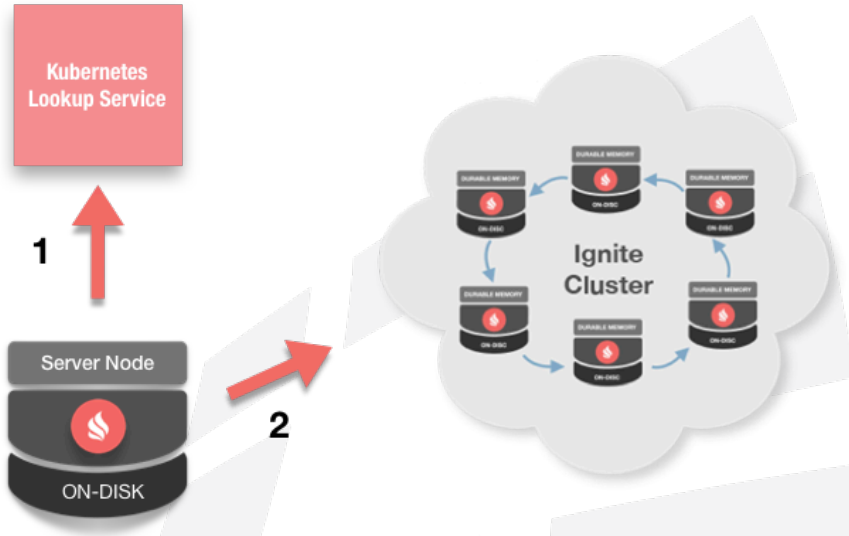
Distributed Database Specificities

- Database is a set of pods
  - IPs are assigned dynamically
  - Auto-discovery is needed
- Applications Deployment
  - Within Kubernetes?
  - Not managed by Kubernetes
- Stateless or Stateful?



Database Pods Auto-Discovery

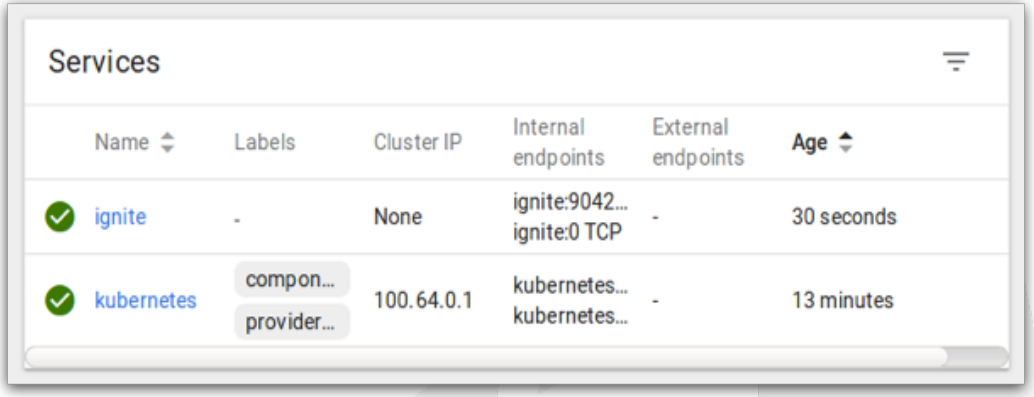
- Kubernetes Lookup Service
  - Tracks a list of all Ignite pods
  - Gateway for remote apps
- Kubernetes IP Finder
  - Consumes IPs from the service
  - Let's node to join the cluster



GridGain

Service Startup

```yaml
apiVersion: v1
kind: Service
metadata:
  name: ignite
  namespace: ignite
spec:
  type: LoadBalancer
  ports:
    - name: rest
      port: 8080
      targetPort: 8080
    - name: sql
      port: 10800
      targetPort: 10800
    - name: thinclients
      port: 10900
      targetPort: 10900
  selector:
    app: ignite
```




```
kubectl create -f ignite-service.yaml
```

IP Finder Configuration

```
<bean class="org.apache.ignite.configuration.IgniteConfiguration">
  <property name="discoverySpi">
    <bean class="org.apache.ignite.spi.discovery.tcp.TcpDiscoverySpi">

      <property name="ipFinder">
        <bean class="
          org.apache.ignite.spi.discovery.tcp.ipfinder.kubernetes.TcpDiscoveryKubernetesIpFinder">
          <!-- Assumed that RBAC is configured for `ignite` namespace. -->
          <property name="namespace" value="ignite"/>
        </bean>
      </property>

    </bean>
  </property>
</bean>
```

Memory-Only Deployments

Apache Ignite Overview

SQL | Key/Value | Transactions | Compute | Services | Streaming | ML

Memory-Centric Storage
Scale to 1000s of Nodes & Store TBs of Data

Ignite Native Persistence
(Flash, SSD, Intel 3D XPoint)

Third-Party Persistence
Keep Your Own DB
(RDBMS, HDFS, NoSQL)

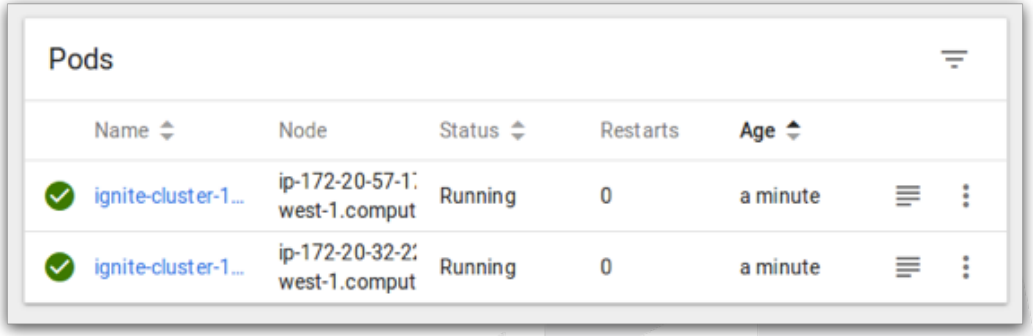
GridGain

Ignite Memory Usage Modes

Mode	Description	Major Advantage
In-Memory	Pure In-Memory Storage	Maximum perfomance possible (data is never written to disk)
In-Memory + 3rd Party DB	Caching layer (aka. in-memory data grid) for existing databases – RDBMS, NoSQL, etc	Horizontal scalability Faster reads and writes
In-Memory + Full Copy on Disk	The whole data set is stored both in memory and on disk	Survives cluster failures
100% on Disk + In-Memory Cache	100% of data is in Ignite native persistence and a subset is in memory	Unlimited data scale beyond RAM capacity

Ignite as Kubernetes Deployment Entity

```yaml
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: ignite-cluster
spec:
  replicas: 2
  template:
    metadata:
      labels:
        app: ignite
    spec:
      containers:
      - name: ignite-node
        image: apacheignite/ignite:2.5.0
        env:
        - name: OPTION_LIBS
          value: ignite-kubernetes
        - name: CONFIG_URI
          value: URL_TO_CONFIG
        ports:
        ...
```




```
kubectl create -f ignite-deployment.yaml
```


GridGain

Stateful Deployments

Stateful Deployments

- Durability With StatefulSet
  - Data persistence to disk
  - Ordered restarts
- Separate Volumes for
  - Data and indexes
  - WAL (aka Transaction Log)
  - Snapshots and backups

```
apiVersion: apps/v1beta2
kind: StatefulSet
metadata:
  name: ignite
  namespace: ignite
spec:
  selector:
    matchLabels:
      app: ignite
  serviceName: ignite
  ...
        volumeMounts:
        - mountPath: "/data/ignite"
          name: ignite-storage
  volumeClaimTemplates:
  - metadata:
      name: ignite-storage
    spec:
      accessModes: [ "ReadWriteOnce" ]
      resources:
        requests:
          storage: 1Gi
```

Cluster Activation

- Manual Activation on First Start

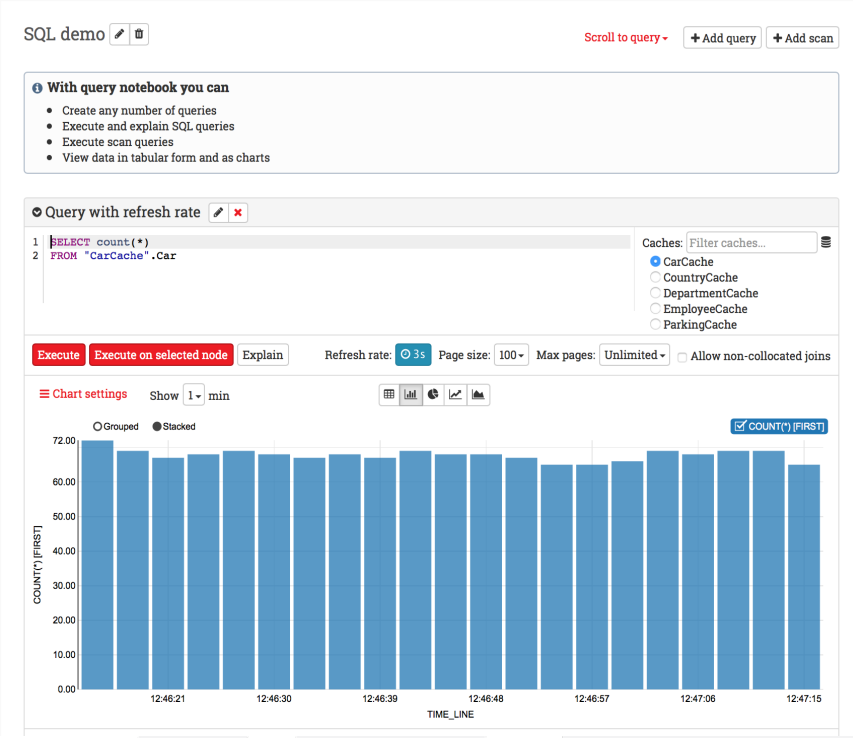
```
kubectl exec -it ignite-0 --namespace=ignite -- /bin/bash
cd /opt/ignite/apache-ignite-fabric/bin/
./control.sh --activate
```

- Automatic Activation on Restarts
  - Baseline topology usage

Management and Monitoring

K8 Dashboard and Ignite Web Console

- Kubernetes Dashboard
  - For Kubernetes environment
- Ignite Web Console
  - For Ignite cluster
  - Deploy Web Agent in K8

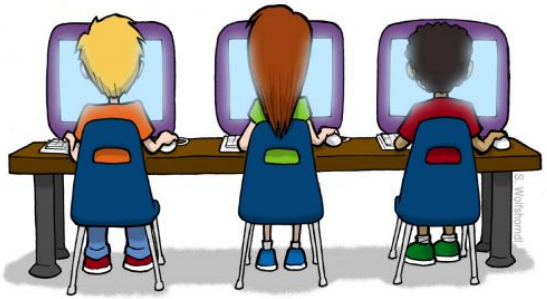


GridGain

Demo

Apache Ignite – We're Hiring ;)

- Very Active Community
- Great Way to Learn Distributed Computing
- How To Contribute:
  - https://ignite.apache.org/



Among Top 5 Apache Projects

Over 1M downloads per year

Top 5 Developer Mailing Lists

1. **Ignite**
2. Kafka
3. Tomcat
4. Beam
5. James

Top 5 User Mailing Lists

1. Lucene/Solr
2. **Ignite**
3. Flink
4. Kafka
5. Cassandra

Top 5 by Commits

1. Hadoop
2. Ambari
3. Camel
4. **Ignite**
5. Beam

Q&A

Thank you for joining us. Follow the conversation.

https://ignite.apache.org

https://gridgain.com/

@denismagda
#apacheignite
#gridgain