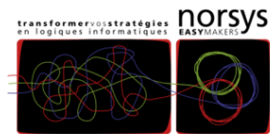




Sponsors  
**Platinum**



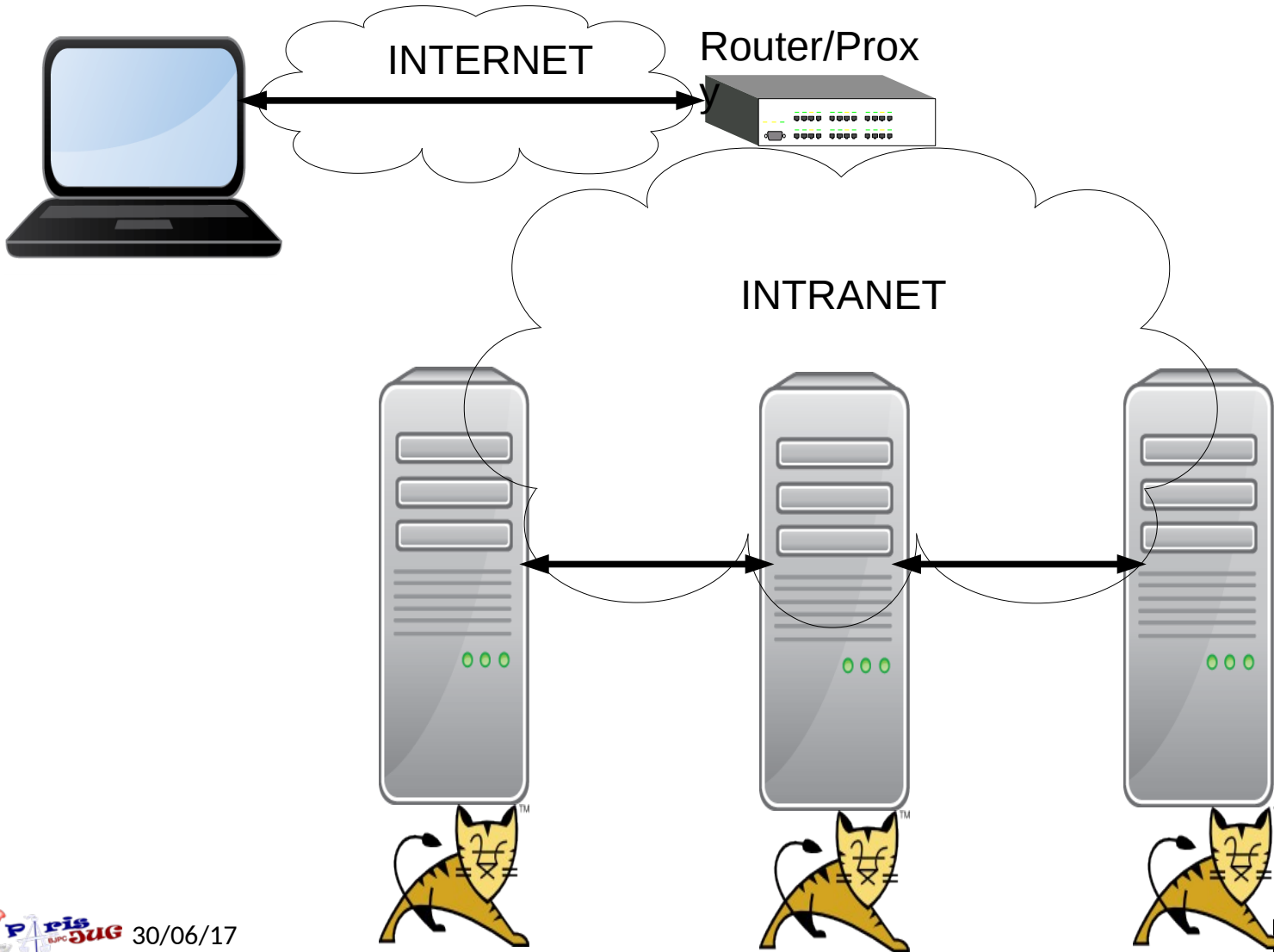
# Tomcat: from a cluster to the Cloud

par Jean-Frederic Clere  
@jfclere



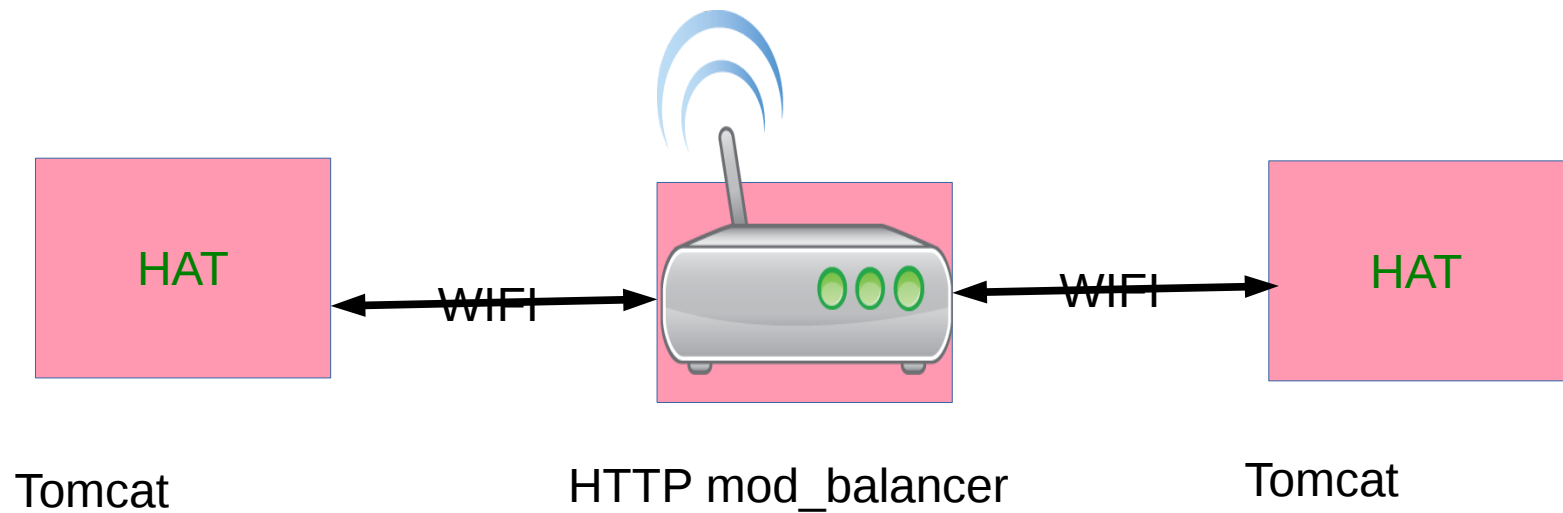
- Who I am
- A cluster:
  - = Session replication and application.
- The cloud:
  - = Nope it doesn't work from scratch.
  - = External session replication
    - Via Infinispan and KubePing
  - = Modify the tomcat cluster (still not finished)
    - AKA dynamic list of nodes
    - Only TCP. (8888 port exported via deployment.yml)
  - = Demos
- What next? Questions / Suggestions

- Jean-Frederic Clere
- Red Hat
- Years writing JAVA code and server software
- Tomcat committer since 2001
- Doing OpenSource since 1999
- Cyclist/Runner etc
- Lived 15 years in Spain (Barcelona)
- Now in Neuchâtel (CH)



- In cluster:
- `<distributable/>` in `web.xml`
- `<Cluster  
className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>`
- Port `upd 45564`
- Ports `tcp range 4000:4100`

- Number Guest clusterized.



FireFox / Chrome

# OPENSIFT

- **A Red Hat project / product**
- **See OpenShift**

<https://www.openshift.com/>

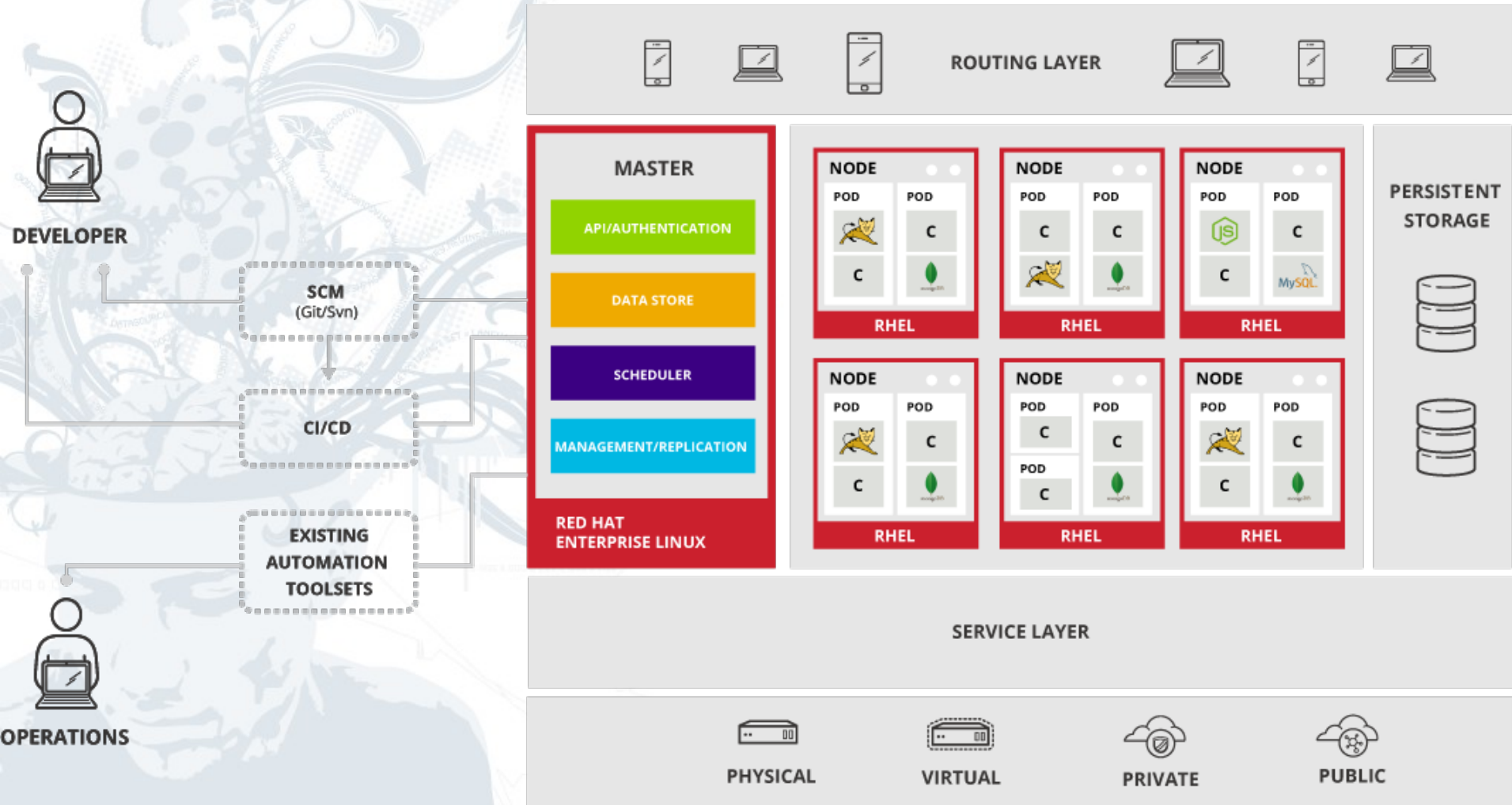
- **Docker 1.4**
- **Kubernetes 1.12**



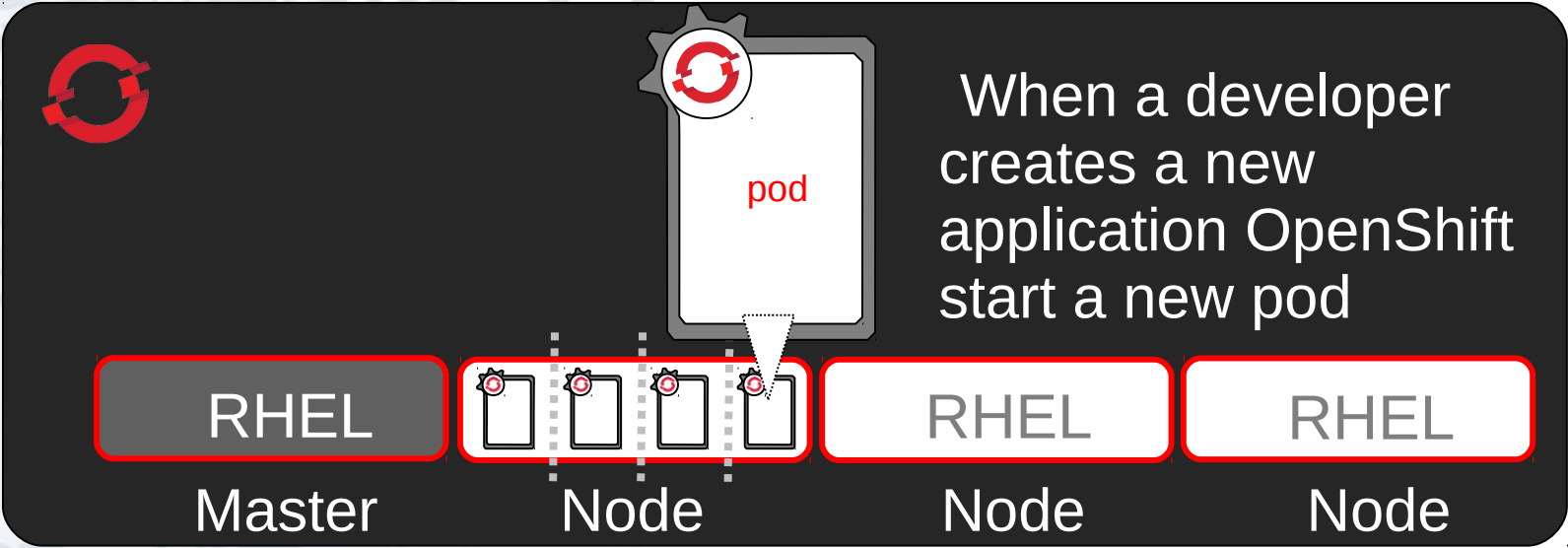
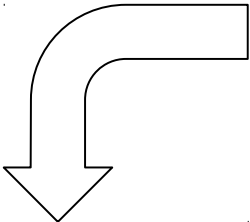
OPENSIFT



# Tomcat in OpenShift



# Developing Tomcat App in OpenShift



AWS / CloudForms / OpenStack (IaaS) / RHEV (Virt) / Bare Metal

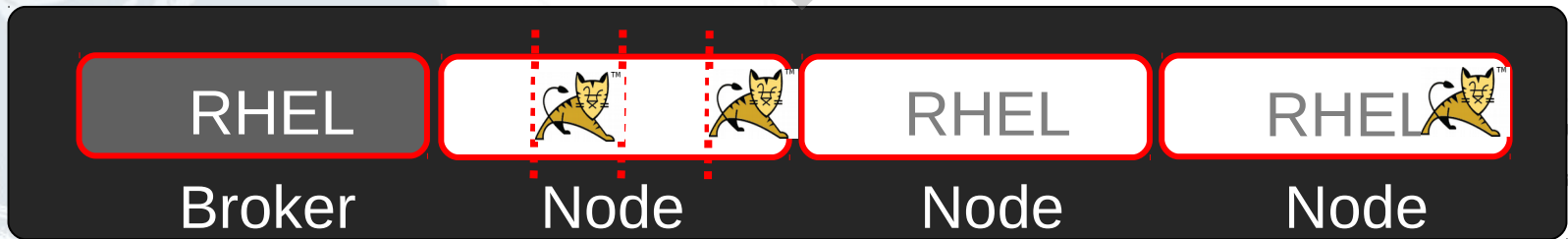
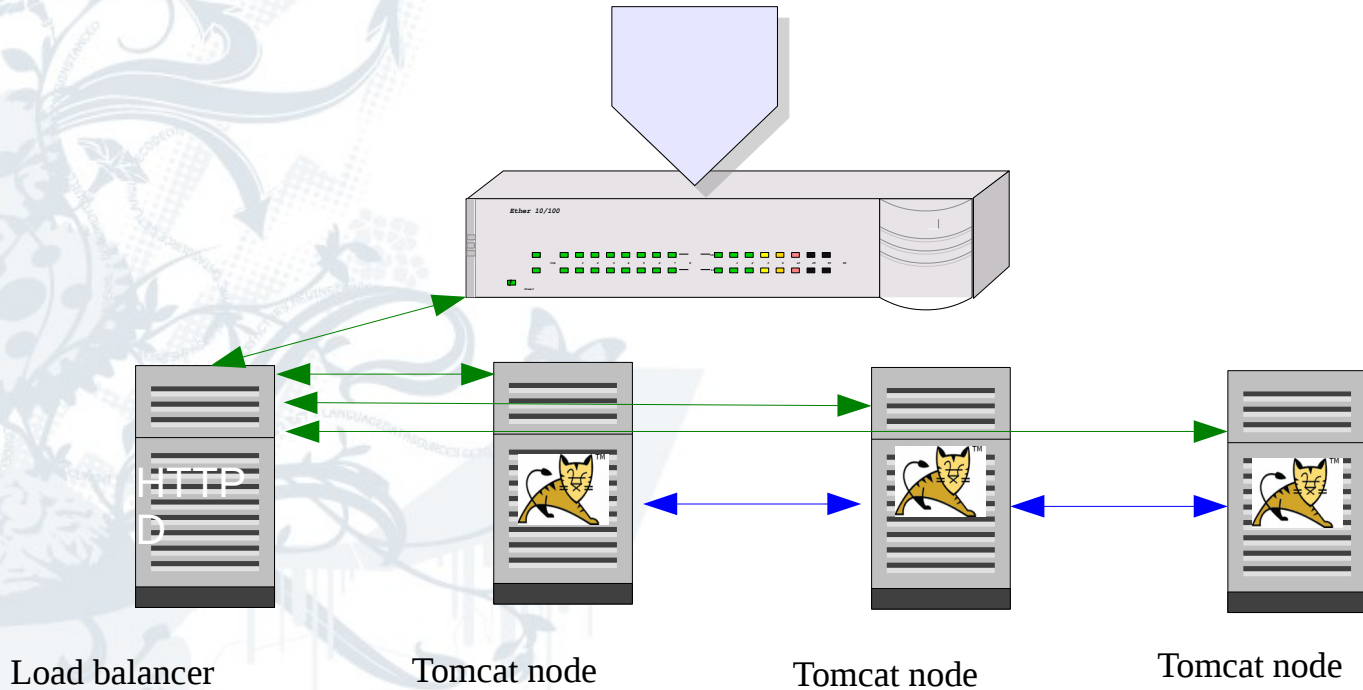
# Getting started

- **minishift:**
  - **Allow to demo on single box.**
  - **Easy to setup**
  - **Small demo**
- **Bare metal / VM:**
  - **Use ansible to install**
  - **2 nodes + master minimal**
- **Tomcat webapp with sessions**
  - **Counter demo.**

# Session replication in a cluster

- **HTTP/1.1**
  - No transaction
  - No persistent connection
- **Web App:**
  - Using cookies to carry session ID
  - Store information in the session:
    - Shopping cart etc.
- **Multi nodes and dynamic**
  - Route request to right node
  - Replicate information

# From a cluster to the Cloud



# Problems for a cluster to cloud...

- **Many ways to solve:**
  - **Embed tomcat with SpringBoot**
  - **Create a docker image**
  - **Extend an existing docker image**
  - **Fabric8**
- **Tomcat session replication:**
  - **No multicast in the cloud.**
  - **Need a “ping” to find the other nodes (KubePing)**
  - **Add view to the system account of the project.**

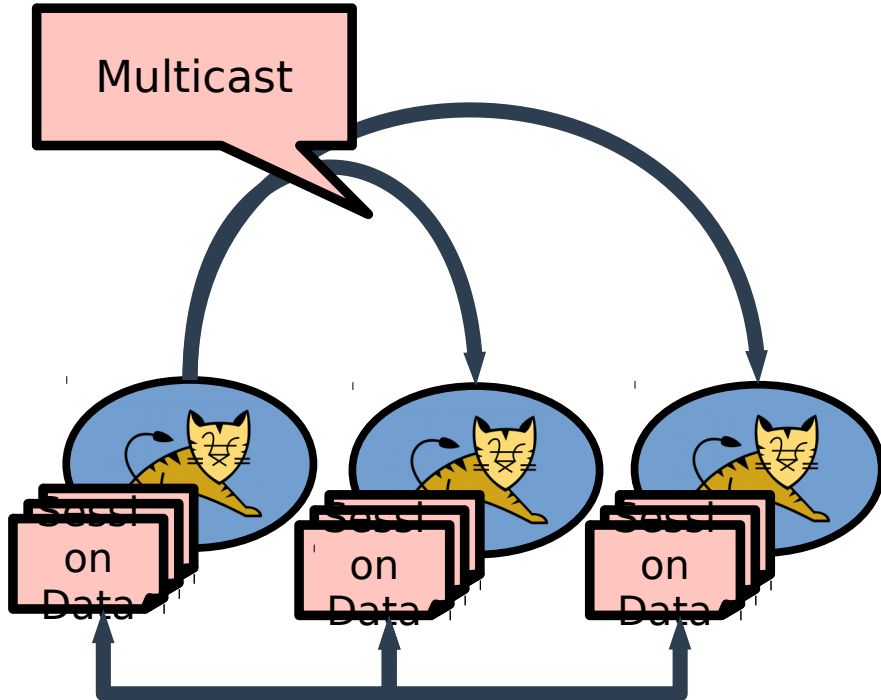
# Goals

- **Have something:**
  - Minimal modification of the tomcat configuration (server.xml)
  - Reuse existing code
- **Some code still missing:**
  - Some in Tomcat
  - Documentation / tests.
- **Some more stuff:**
  - We use ansible for the install.
  - Some maven builds and shells.

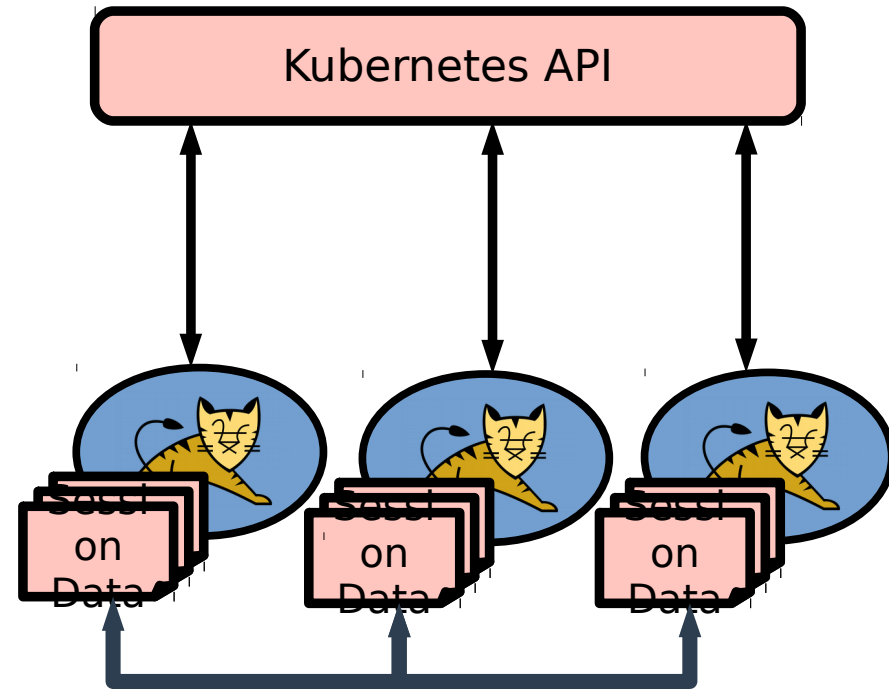
# Solution

Tomcat built-in solution  
Peer discovery through multicast  
heartbeat messages

*Does not work in a cloud environment*



Our solution  
Peer discovery through  
Kubernetes Downward API  
*Works in OpenShift*





# Kubernetes API

Tools for managing a  
Kubernetes cluster

Accessible from the pods  
within the cluster

**GET**  
*/api/v1/namespaces/tomcat-in-  
the-cloud/pods*

→ Return a JSON representation of all  
the pods in the cluster

```
kind: PodList
apiVersion: v1
▼ metadata:
  selfLink: /api/v1/namespaces/tomcat-in-the-cloud/pods
  resourceVersion: 7602
▼ items:
  ▼ 0:
    ▼ metadata:
      name: tomcat-in-the-cloud-1-5xbwm
      generateName: tomcat-in-the-cloud-1-
      namespace: tomcat-in-the-cloud
      ▶ selfLink: /api/v1/namespaces/tomca...at-in-the-cloud-1-5xbwm
      uid: ecac3cff-5361-11e7-9a95-3a314e9cf749
      resourceVersion: 7568
      creationTimestamp: 2017-06-17T13:36:10Z
      ▶ labels: Object
      ▶ annotations: Object
      ▶ spec: Object
    ▼ status:
      phase: Running
      ▶ conditions: [3]
      hostIP: 192.168.42.74
      podIP: 172.17.0.3
      startTime: 2017-06-17T13:36:10Z
      ▶ containerStatuses: [1]
    ▶ 1: Object
    ▶ 2: Object
```

# Architecture

DynamicMembershipService

RefreshThread

- Call memberProvider.getMembers()
- Filter out already known Member
- Inform listeners of new/dead members

MemberProvider

- init(Properties)
- getMembers(): List<Member>

KubernetesMemberProvider

- init():
  - Get URL, cert, ... from environment variables
  - Set startTime
- getMembers():
  - Call api to get pods
  - Filter active pods
  - Compute aliveTime

# Where are we?

- Main sites:

- <https://docs.openshift.com>
- <https://github.com/Project31>
- <https://github.com/iSma/tomcat-in-the-cloud/>

- Thanks:

- Université de Neuchâtel
- Kurt Stam  
<kstam@redhat.com>

Questions ?  
Suggestions?

Merci !

carbon<sup>IT</sup>

arolla  
Mastering Software Development

 Couchbase

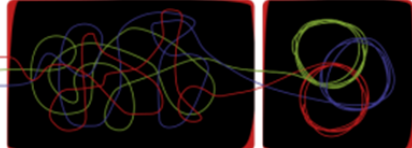
 zenika

OXiane

  
MIRA KL

transformer vos stratégies  
en logiques informatiques

norsys  
EASYMAKERS



SOFTEAM Cadextan

esiea  
ÉCOLE D'INGÉNIEURS  
DU MONDE NUMÉRIQUE

\*WIKI™

VIDEO

IPPON  
Digital . Technologies . Hosting

