Setting up and using the Red Hat OpenShift Container Platform.

Mark Vinkx: System engineer infrastructure & operations
Introduction
UZ Leuven

1,955

9,000+

Centralized IT

large hospital in Belgium
Clinical Work Station (KWS)

- One stop shop for all care providers
- Patient Management
  - Planning
  - Order entry
  - Electronic drug prescription
  - Bedside scanning
  - Drug-drug interactions
  - Lab results
- Patient Centric
- Workflow oriented

and much more

EPR
nexuzhealth

20 (18 prod)

10,000+

25,000+
nexuzhealth service points
Challenges

- **Uptime**
- **Quality of the data**
- **Flexibility**
  - Growth
  - New projects, ..
  - Data explosion
  - IoT, ...
- **Security and Privacy**
Where do we come from
Database

- High availability
- Strong team
- Business logic
- Highly optimized

Sybase
Middle tier

Critical
Java webstart
Action and settings

Management

Frontend for other systems

Becoming more important
nexuzhealth
External services
Mobile applications

11 server
78 apps
First Steps
The Team

Developer

Unix admin

JBoss admin

Network storage specialist
Application catalog

Technologies

- Business Process Services
  Model, automate, and orchestrate business processes across applications, services, and data.

- Continuous Integration & Deployment
  Automate the build, test, and deployment of your application with each new code revision.

- Messaging
  Facilitate communication between applications and distributed processes with a messaging server.

- Single Sign-On
  A centralized authentication server for users to log in, log out, register, and manage user accounts for applications and RESTful web services.

- Data Stores
  Store and manage collections of data.

- Uncategorized
Learning about openshift

- Services
- Deployments
- Routes
- Projects
Learning about containers

- Builds
- Docker
- Layered file system
- Images
Setting up the production cluster
Production cluster

- Deployment options
- Subscription model
- Installation
- Impact on Development
CloudNativeCon pushed us to the limit
Cloud Native Computing Foundation

• Mission
  – drive the adoption of a new computing paradigm that is optimized for modern distributed systems environments

• Role
  – Fostering the growth and evolution of the ecosystem
Cloud Native Computing Foundation

- Kubernetes: Orchestration
- Prometheus: Monitoring
- OpenTracing: Distributed Tracing API
- Fluentd: Logging
- linker: Service Mesh
- gRPC: Remote Procedure Call
- CoreDNS: Service Discovery
- containerd: Container Runtime
- rkt: Container Runtime
- CNI: Networking API
- Envoy: Service Mesh
- Jaeger: Distributed Tracing
CloudNativeCon

- Kubernetes
- Ecosystem of tools
- A lot of traction
Then the dark times came
Problems

• Container to container communication
• Healthchecks

Result

• failing applications
• Bootloops
Tackling the problems

• Support case
• Escalation engineer

Getting it fixed
• Kubernetes Core
• Openshift network plugins
Putting it all together
Building confidence

• Stabilize the cluster
• Integrate in our environment
• Expand the team
Enjoying the benefits

• Applications in production
• More environments
• Greater level of control
Greater level of control

<table>
<thead>
<tr>
<th>Deployment</th>
<th>Status</th>
<th>Created</th>
<th>Trigger</th>
</tr>
</thead>
<tbody>
<tr>
<td>#67 (latest)</td>
<td>☮️ Active, 2 replicas</td>
<td>5 days ago</td>
<td>Image change</td>
</tr>
<tr>
<td>#66</td>
<td>✓ Complete</td>
<td>6 days ago</td>
<td>Config change</td>
</tr>
<tr>
<td>#65</td>
<td>✓ Complete</td>
<td>6 days ago</td>
<td>Image change</td>
</tr>
<tr>
<td>#64</td>
<td>✓ Complete</td>
<td>7 days ago</td>
<td>Manual</td>
</tr>
</tbody>
</table>
Looking beyond
Things to do

• Setting up multiple clusters
• Interface for development
• Metrics and monitoring
Is this the end?