

AWS

S U M M I T

Migrating and Running Continuous Integration Systems at Scale in AWS

Peter Caron, HERE Technologies

18 May 2017



Introduction

HERE is one the world's leading map data companies and is now delivering the next generation of mobility and location-based services.

“We are no longer a map company, but a data company”

Edzard Overbeek
CEO
HERE



Portfolio

HERE software serves map, traffic and location data to a variety of target platforms

- HERE Open Location Platform (OLP)
- Embedded Automobile Navigation
- Enterprise Extensions
- SDKs and Mobile Apps
- HERE Location Services



Our Challenge

HERE needed a build system that could scale to meet our needs to develop and test complex, heterogeneous products and services.



Agenda

Challenges of CI/CD
Moving to a Cloud
Transition to AWS



Before

Jenkins Servers
in Data Centres

different



Merge Hell



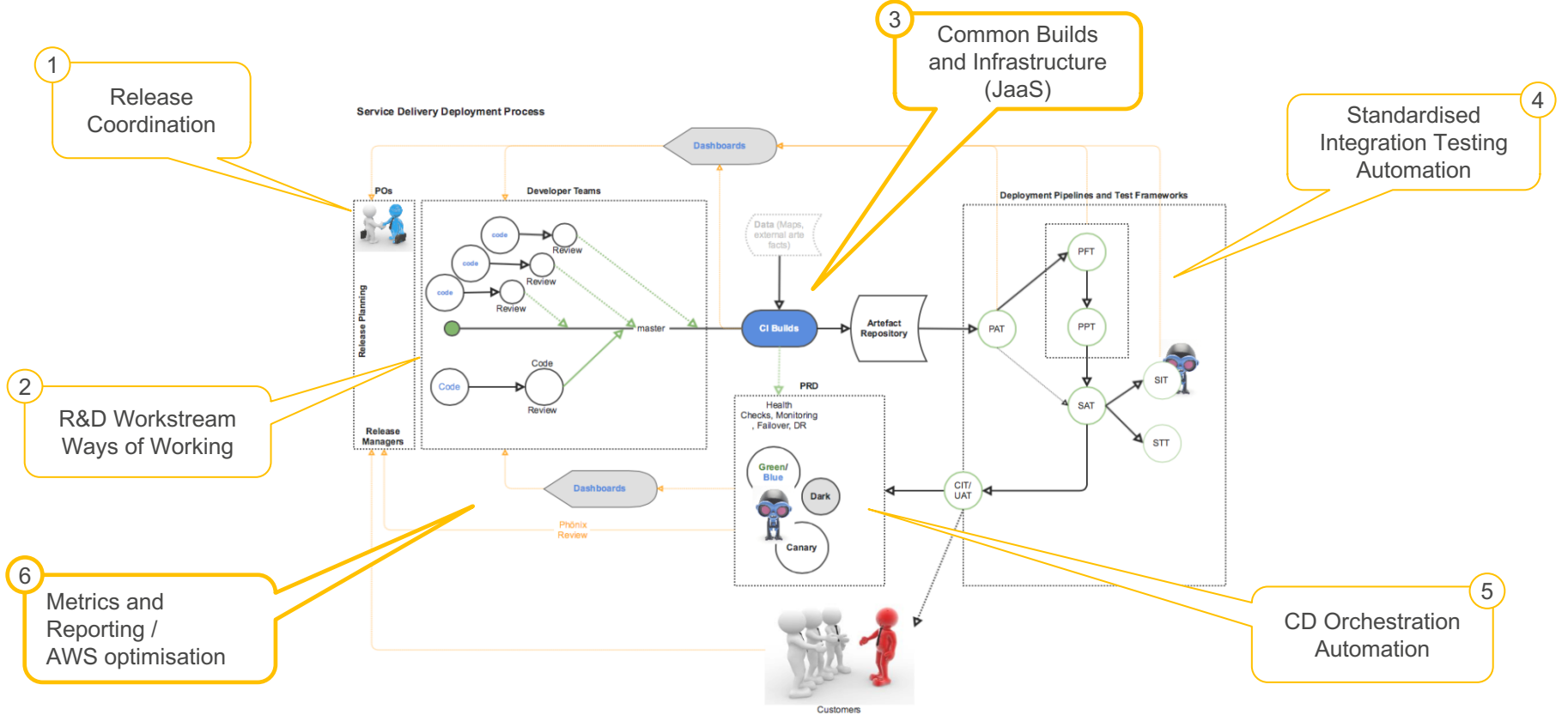
Build Systems

- Dozens of machines under the desk or for each team
- Incompatible build servers
- VM-based build systems in different DCs
- Incompatible workflows
- No metrics

Results

~5+	Builds per month
40K+	Tests / cycle
1000+	VMs on VMWare
~1	Deployment/month
00s	Acceptance tests/month

After



AWS Services in Production



Common CI Systems – CCI (Jenkins)

140K+

Builds per day

25M+

Tests per day

CI for Micro-services – (CI as a Service)

130

Products and services

CD Platform Pipelines (GoCD as a Service)

40

Services

668

Unique pipelines

600+

Instances on AWS

40+

Deployments/month

00s

Acceptance tests/day



Moving CI / CD Toolsets and Infrastructure to AWS ...

- Git
- Gerrit
- Jenkins
- Artifactory
- GoCD
- Splunk



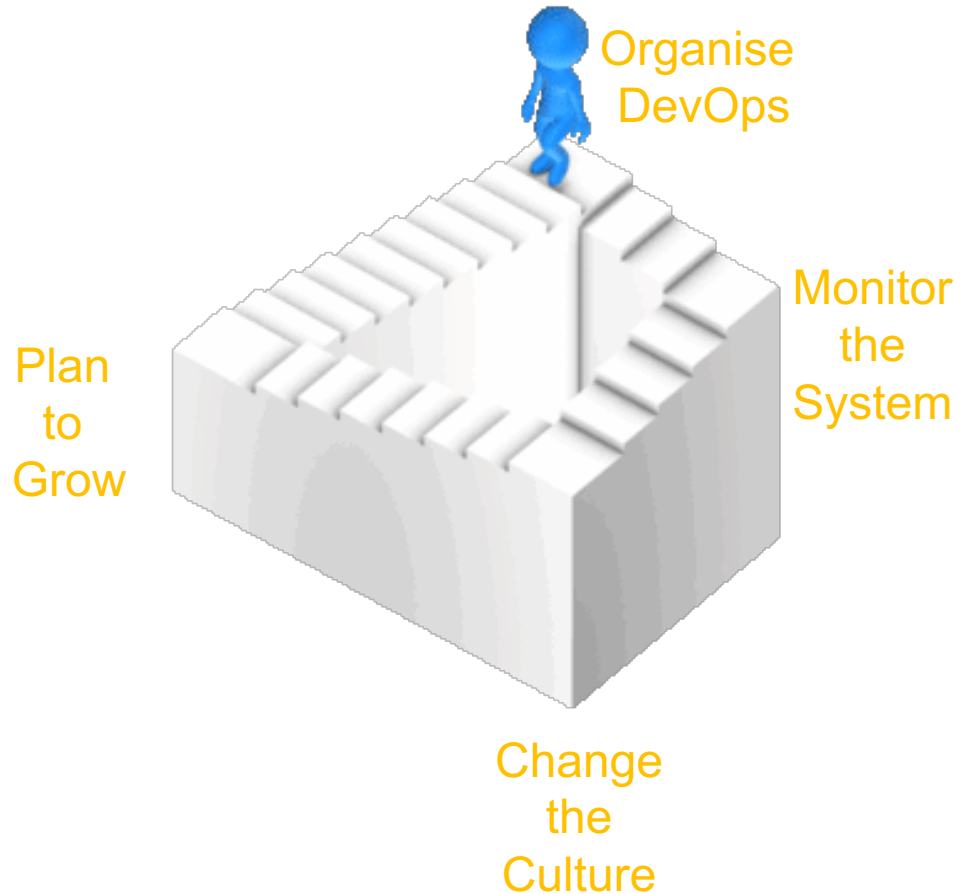
It was a *simple* lift and shift from our local infrastructure

... and everything worked well from Day 1



Not exactly!

Steps to Success



Plan to Grow

- Get your Workflow right
- Know your Capacity and Limits
- Focus on Testing
- Set Expectations Internally
- Know your Fallback options
- Monitor changes (costs)



Change the Culture

- Start small, iterate
- Do CI right *before* introducing AWS
- Infrastructure as Code
- Understand your changes
- Measure Results and Adapt



What did we learn?

- Capacity is always underestimated.
- Costs will be high at first
- Trust your developers and listen to their complaints
- Moving to the Cloud is not an answer
- Plan a fallback
- People will use the system
- One size does not fit all
- Track your costs closely



... and what could we have done better?

@HERE Technologies Today

Next Steps

Infrastructure as Code
Build on Secure DevOps mode
Deployment Orchestration
Completely self-service pipelines
Software Defined Environments

We're building the best location-based products in the world! To do that, we need to have the best development systems in the world.



AWS

S U M M I T

peter.caron@here.com

