From Git Fork to Server Farm
Mesos in research infrastructure &
An approach to traceability in the mesos ecosystem

Naoise Dunne (Insight Centre for Data Analytics)
Overview

1. Why Mesos as research infrastructure
2. Overview of support infrastructure for Mesos
3. How we implement traceability on mesos infrastructure in continuous Integration/deployment
4. Demo
Why Farms and Forks

In Food and Farming Traceability and holistic approach makes

• better crafted food
• happier farm communities
• Safer food
• Tastier food
• Happier customers
Why Farms and Forks

In **Infrastructure**
Traceability and holistic approach makes
- better crafted code
- happier dev communities
- Safer Applications
- Tastier Applications
- Happier customers
Who am I?

Naoise Dunne

Research Fellow
Work on Distributed Applications
Focusing on linked data analytics at large scales
Driving “research-ops” at insight

Insight Centre

Centre for Data Analytics
60M investment in research
Europe's largest research centre for Data Analytics
Empowering a data-driven society to enable better decisions by individuals, communities, business and governments.
Why Mesos?

The Challenges that helped us choose Mesos for our research infrastructure
What are the challenges for infrastructure at Insight?
Insight Centre Infrastructure Challenges

Characteristics of Infrastructure for Analytics Research

• Mix of Data Science skillsets and roles
  • Phd students, Post Doc researchers, University Administration
  • Partner Institutes, Commercialisation Partners
What are the profiles of these Data Scientists?
Data Science skillsets and roles at Insight

Engineer
Focused on the technical problem of managing data
Normally strong software developers

Creative
Need to explain the meaning of the data.
Good generalists, can code, with a flare for the visual or data narrative.

Researcher
People with deep academic background in science, maths, machine learning
Reluctant coders, amazing analysts
Insight Centre Infrastructure Challenges

Characteristics of Infrastructure for Analytics Research

• Mix of Data Science skillsets and roles
  • Phd students, Post Doc researchers, University Administration
  • Partner Institutes, Commercialisation Partners

• Unscheduled bursts of Activity
  • Little or no planning, loose communication
  • Very real, very immediate need for access to resources
Characteristics of Infrastructure for Analytics Research

• Mix of Data Science skillsets and roles
  • Phd students, Post Doc researchers, University Administration
  • Partner Institutes, Commercialisation Partners

• Unscheduled bursts of Activity
  • Little or no planning, loose communication
  • Very real, very immediate need for access to resources
    • Need to create Proposal
    • Paper is accepted, need to reproduce expensive query on huge dataset
    • We need last years Big Data demo up on Tuesday

Planning is impossible
Very real, very immediate need for access to resources
Insight Centre Infrastructure Challenges

Characteristics of Infrastructure for Analytics Research

• Mix of Data Science skillsets and roles
  • Phd students, Post Doc researchers, University Administration
  • Partner Institutes, Commercialisation Partners

• Unscheduled bursts of Activity
  • Little or no planning, loose communication
  • Very real, very immediate need for access to resources

• Very small ops team
  • Ratio of about 80:1 developers to operations staff

• Data Science and “Big Data” focus
Small ops team

Bottlenecks

• Could not keep up with workload - thrashing release cycle
• Research becomes waste and huge backlog, apps get dropped
• Cutting corners hurts security etc.

Stovepipes

• When researchers leave can’t manage their applications
  • tacit knowledge exists only within research teams - easily lost
• No shared approach to managing applications
  • digital archiving
Insight Centre Infrastructure Challenges

What was the result of these challenges?

• Mix of Data Science skillsets and roles
  • huge waste as teams “baked in” overlapping skillsets and resources
• Unscheduled bursts of Activity
  • At best brittle deliveries, quickly failing services, vaporware
• Very small ops team
  • workload bottlenecks, thrashing delivery cycle, infrastructure suffered, software rot
• Data Science and “Big Data” focus
  • Hadoop and similar infrastructure goes to seed… a lot
How did our researchers feel about this?
Enter Mesos
Mesos Ecosystem

How we use mesos at Insight
Insight Centre Infrastructure Challenges

What we needed to meet these challenges...

- Mix of Data Science skillsets and roles
  - **Service Mix** right DB & services for our 3 kind of scientists
- Unscheduled bursts of Activity
  - **Agility** change our application mix with no turnaround
- Very small ops team
  - **Efficiency** best use of computing resources
- Data Science and “Big Data” focus
  - **Scalability** grow to the current demand of our apps
2 level scheduler: flexible, agile

- Can Schedule many kinds of applications
- Frameworks (such as spark) are delegated the per application scheduling
- Mesos responsible for resource distribution between applications and enforcing overall fairness
- Very modular, due to 2 level scheduling. frameworks manage apps as they like
Applications work with frameworks to get resources they need

Frameworks Negotiate with mesos to run their jobs

Docker manages isolation on Linux servers

Resources cpu mem disk Managed by Mesos

Managed by Mesos

Mesos - scheduler short jobs

Mesos - scheduler long run jobs

Graph Jobs

Datastores HDT, Neo4J

Granatum

Revealed

Marathon Framework

graphX

Spark Fwk

Chronos Fwk

Resources

cpu mem disk

Managed by Mesos

Mesos

Mesos - resource management

OS Monitor

Docker

mesos client

Linux Server

Docker

mesos client

Linux Server

Docker

mesos client

Linux Server

Docker

mesos client

Linux Server
We use graph X for large graph batch jobs.

We use both HDT (RDF Store) and Neo4J (Graph).

We deploy specialised linked data applications to cluster.
Linked Data Infrastructure

Mesos
- resource management
- scheduler short jobs
- scheduler long run jobs

Datastores
- HDT, Neo4J

Spark Fwk

Chronos Fwk

GraphX

Granatum

Revealed

Docker

mesos client

mesos client

mesos client

Linux Server

OS Monitor

Mesos Monitor
Looking at the wider Mesos environment
What more than Mesos

We need HDFS for large storage on Spark Jobs

Marathon can now use HDFS to store large Dependencies

We need Mesos DNS for service discovery

Every service should be deployed through Jenkins

Everything, absolutely everything should be configured through Git

HDFS

Zookeeper

Mesos & frameworks needs zookeeper

Need Mesos DNS for service discovery

Docker

Glue

Docker Registry

Glue Registry

you will need docker reg for marathon

To run mesos you will need dcos or glue

Mesos

Mesos - resource management

Spark Fwk

Chronos Fwk

Marathon Fwk

Mesos - scheduler short jobs

Mesos - scheduler long run jobs
Deployment Flow for Web services

- Git
- Jenkins
- Docker Repository
- Marathon
- Mesos
Mesos Delivers...

- **Efficiency** - scheduler gives best use of resources
  - we can build our own! try alternatives to FAIR
- **Agility** - change our app mix with no turnaround
  - marathon for web service, spark for batch
- **Scalability** - grow to the current demand of our app
  - most framework take advantage of mesos flexibility
- **Modularity** - Mesos allows quick repurposing of cluster
  - Want to run hadoop rather than spark no problem
Good news! We are saved

Mesos ecosystem now providing:

● Efficiency
● Agility
● Scalability
● Modularity

However...

In real world we also need...

● Accountability (traceability)
Traceability
from git fork to server farm
The need for accountability

After going live with mesos...

• Agility was good, but providence of application instances was difficult

• Multiple developers/researchers deploying the same application at the same time

• As applications could not be traced to code - difficult to debug and a lot of finger pointing between dev and ops

• Reporting was difficult across so many systems
We needed to understand the application Flow and trace the life of code
Tracing Deliveries along deployment flow

1. Git
2. Jenkins
3. Docker Repository
4. Marathon
5. Mesos
Tracing Deliveries: Actual Flow

- Git
- Jenkins
- Docker Repository
- Marathon
- Mesos

- Marathon Artifact store
- Deployment tool
- Mesos DNS
Tracing Deliveries: Monitoring

Git → Jenkins → Docker Repository → Marathon → Mesos

Monitoring (elasticsearch)
What we trace

Deployment Domain

Name

Commit ➔ Instance
Deployment Flow

Git
- Name
- Commit

Jenkins
- Job
- Build

Docker Repository
- Image
- Tag

Marathon
- ID
- Instance
- ID

Mesos
- Name
- ID
Demo

walk through of tracing on live system
Advantages of Traceability

- Consistent reporting
- Eases log aggregation
- Holistic view of Deployment
- Simplifies communication between Ops & Developers
- Multiple versions problem can be managed
After going live with mesos we

In Infrastructure traceability and holistic approach makes
- better crafted code
- happier dev communities
- Safer Applications
- Tastier Applications
- Happier customers
Thank you.
Questions?