



The road to OpenStack at (one of?) UK's largest academic private cloud: University of Edinburgh

Jan Winter
jan.winter@ed.ac.uk
+44 131 650 5009

Kenneth Tan
kenneth.tan@sardinasystems.com
+44 798 941 7838

1.

Who?

University of Edinburgh

one of the leading universities in the world, amongst top 6 in the UK
ranked #6 or #7 in Europe, depending on whom you ask

founded a little while ago ... in 1582 (older than some countries!)

population: ca. 45k (undergraduate + postgraduate students, admin
staff, academic staff)

University of Edinburgh

Research Services, IT Infrastructure Division: part of University's central Information Services

Responsible for operating university-wide IT infrastructure / services for research

Research Services: 10 people (yes, only! and I am 10%!)

2.

Strategy and objective

Strategy

move higher cost services delivery channels to lower cost delivery channels

automation + self-service: lower cost and increase speed of response and lesser errors

enable higher value HR, deliver higher value services

Cloud objective

provide flexible infrastructure of compute, storage, network,
services for research

... enable higher value HR, deliver higher value services

3.

**Where is OpenStack
in this picture?**

Role of OpenStack

provide users with flexible, self-serviced provisioned of computing infrastructure

compute, storage, networking, applications portfolio

OpenStack should not result in net increase of workload for Research Services operations team

Role of OpenStack

serve disparate user groups, varying and competing needs,
long-lived and short-lived workloads

hpc/technical computing workload, eg: CERN, particle physics

data science workload: hadoop, spark

cannot make assumption about users' level of technical
expertise

more importantly ...

need to meet organization mission

don't throw money at the challenge

money does not grow on trees

4.

**The partner (not in
crime!)**

Sardina Systems

European cloud platform software vendor with presence in Slovakia, Romania, Russia, UK

technical team's expertise in large scale, business and operations critical facilities in finance, defense, government, meteorology and automotive industries

OpenStack Foundation Corporate Sponsor

5.

The solution

Route to the solution

problem definition and solution design

system architecture

hardware specification

project management: what's happening when, by who

Route to the solution

plan for production from day 0

build for production from day 0

it's not a toy: high availability is not optional

monkey-capable no-magic scaling up process

Route to the solution

not a guinea pig!

... to consider full lifecycle

deploy

operate

upgrade

6.

The solution design

The solution design

management: 16 nodes

storage: 9 ceph nodes

compute: ca. 200+ nodes (design size; varies from 50+, can dynamically add using other servers in the farm)

network: 5 VLANs, 10 GbE

integrates with broader University IT services

7.

Deploy phase

From design to deployment

the ingredients: what's the configuration needed

ingredients ready: make sure hardware are available (servers, storage, networking)

system configuration: single source of truth at all times

Automate, Automate, Automate

system configuration = input for automation

deployment of entire system: fully automated

reduced project risk = increased operational confidence

Automate, Automate, Automate

base OS deployment with Foreman

post OS deployment configuration with SaltStack

full OpenStack + Ceph deployment with FishOS Deployer
(SaltStack variant)

principle: standard enterprise tools, nothing fancy!

Deploy: 1, 2, 3 ... showtime

1. input: configuration and inventory
2. run deployer
3. 2 cups of coffee
... done

8.

Concept to reality

Deployer

pre-tested and proven off-site

ci/cd process

no knowledge of OpenStack necessary

Implementation: proof of solution

detailed planning: what is to be done when, what will be available when, who will do what by when?

time from zero-to-operation: 8 weeks (and 6 weeks waiting for storage hardware to arrive)

9.

Upgrade phase

Run at all times

n to n+1 upgrade every 6 months (releases eol in 12 months)

2n upgrades in n years: more risky option

it's not a toy!

system has to continue operation at all times

Zero-downtime?

get solution architecture right (from deploy phase) ...

else, find out problem 6 months later

monkey-capable fully automated upgrade process

10.

The operate phase

Challenges

uptime + availability

resource management

infrastructure-as-code operation

idempotent: flexible to change at any time

Reliability, Uptime, Availability

high availability: not an luxury

service consumers only interested in availability +
reliability

Downtime

any time any service is not available to service consumer, whatever the cause may be

Downtime during operate phase

intrinsic — vs — extrinsic

downtime risks

Downtime during operate phase

intrinsic — vs — **extrinsic**

downtime risks

Extrinsic downtime risk

dealt with via highly available solution architecture

What is OpenStack?

a series of intercommunicating services

HTTP, MQ, DB

The types of services

1. with data + configuration
2. configuration only

Safeguards: data

replication, replication, replication

2-node model or quorum/odd-node model

Safeguards: data

traditional high availability operations model

http services

http: it's a web server! let's treat it as such!

high availability for web servers

What else? ... mq

mq: put stuff in queue, take stuff from queue

replicate reader/writer

in short, for extrinsic downtime risks ...

keep extra copy available at all times

Smarts ...

it's in the solution architecture

get it right, else it can come back to bite you!

Other requirements for production

usage accounting and billing

log + metrics management

monitoring

capacity planning

11.

**Problems
encountered**

Problems encountered

downtime due to running out of disk space,
affecting ceilometer, mq

networking hardware problem (being investigated),
resulting in packet loss

cinder problem encountered during Ocata upgrade

12.

**Nearly 2 years
later ...**

Worked well, growing

system expansions (storage and compute)

easily add compute nodes as necessary

growing by another ca. 2000 cores in
coming weeks

system can be supported by just 1 person

13.

Key lessons learnt?!

Key lessons learnt

1. didn't spin own Linux distribution, so don't spin own OpenStack either

Key lessons learnt

2. not planning, planning, planning ... but rather ...
experience + planning, experience + planning, and more
experience + planning

Key lessons learnt

3. make sure your vendor is able to support your operation when needed



Jan Winter
jan.winter@ed.ac.uk
+44 131 650 5009

Kenneth Tan
kenneth.tan@sardinasystems.com
+44 798 941 7838

The road to OpenStack at (one of?) UK's largest academic private cloud: University of Edinburgh

edinburgh.ac.uk | sardinasystems.com