Putting a Red Nose on the Cloud

COMIC RELIEF

armakuni

About Comic Relief

- Comic Relief is a major charity based in the UK which strives to create a just world free from poverty
- Since we first set up shop in 1985, we've been doing three main things:
  - We raise millions of pounds through two big fundraising campaigns – Red Nose Day and Sport Relief.
  - We spend that money in the best possible way to tackle the root causes of poverty and social injustice.
  - We use the power of our brand to raise awareness of the issues that we care most about.

RED NOSE DAY

15th March

Do Something Funny for Money

- Every two years, we encourage thousands of people to do something funny for money.
- A year of planning
- 6 week media campaign
- 7 hours of TV on the 15th March

armakuni

What we had

- 8 year old Java application
- Deployed and scaled with the help of 12 partners
- Took months to achieve this, run through user testing, penetration testing and authentication
- Changes were kept to an absolute minimum between years for stability and to reduce risk

armakuni

Key Aims of New Platform

- Unlimited by technology
- Minimise PCI exposure
- Remove reliance on any single third party supplier
- Cost-effective
  - All the money raised by the public is spent by Comic Relief to help poor and disadvantaged people in the UK and the world's poorest countries.

armakuni

What we have now

Reminder: QCon Session Code : 9221

Over to you Tim...

armakuni

Thanks Zenon...

This talk is a case study that intends to:

- Give you an insight into the solution we have delivered over the last 9 months
- Discuss the patterns we have applied and how we (and as a consequence, Comic Relief) have benefitted

armakuni®

Platform Requirements

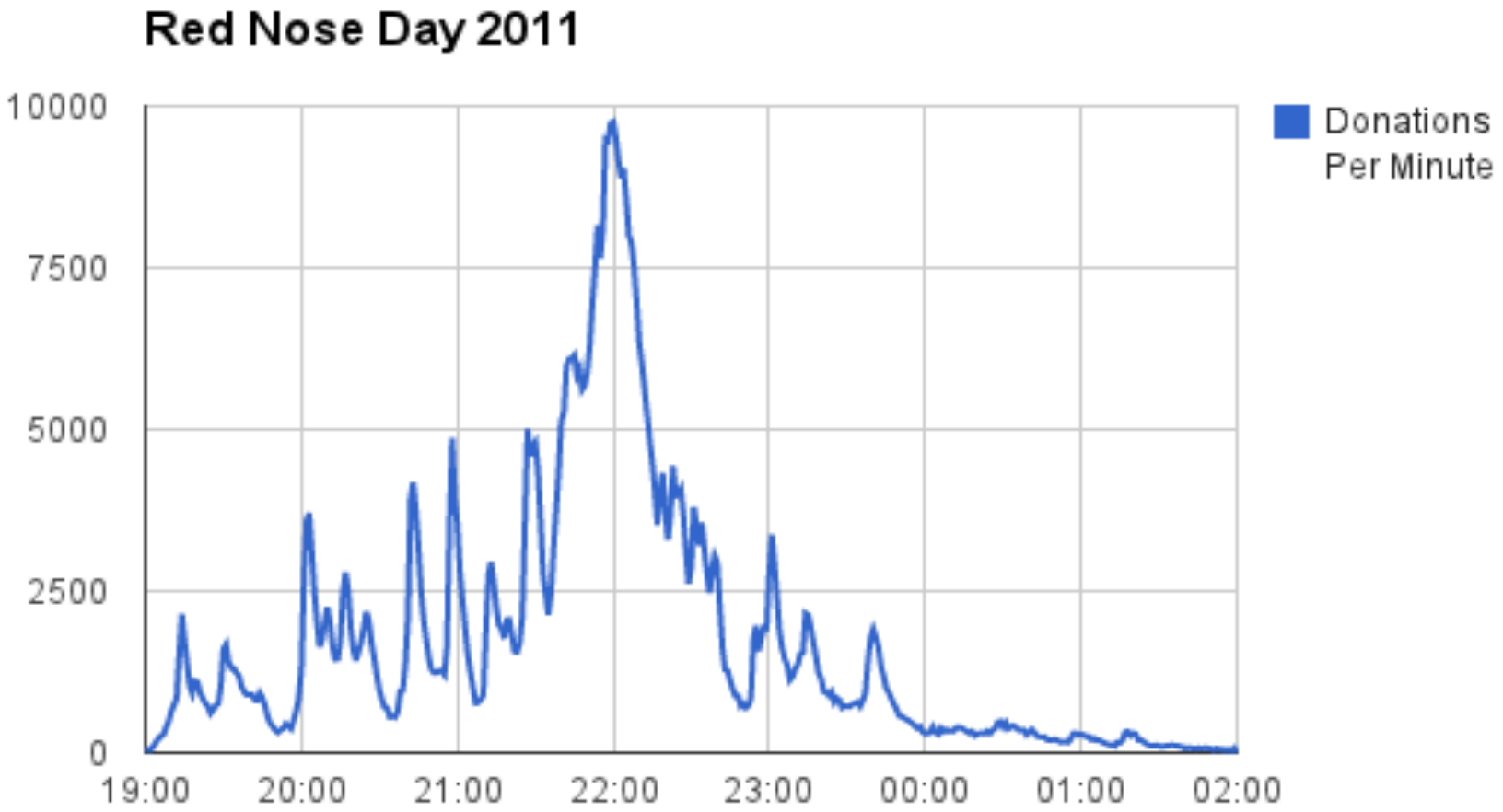
- **The platform is required to:**
  - serve a donation page for the public
  - manage a lightweight call centre interface
  - process in the region of 600,000 transactions in 7 hours
  - handle in excess of 10,000 call centre operators
  - handle a peak of 300 donations completing per second
  - be out of scope for PCI

armakuni

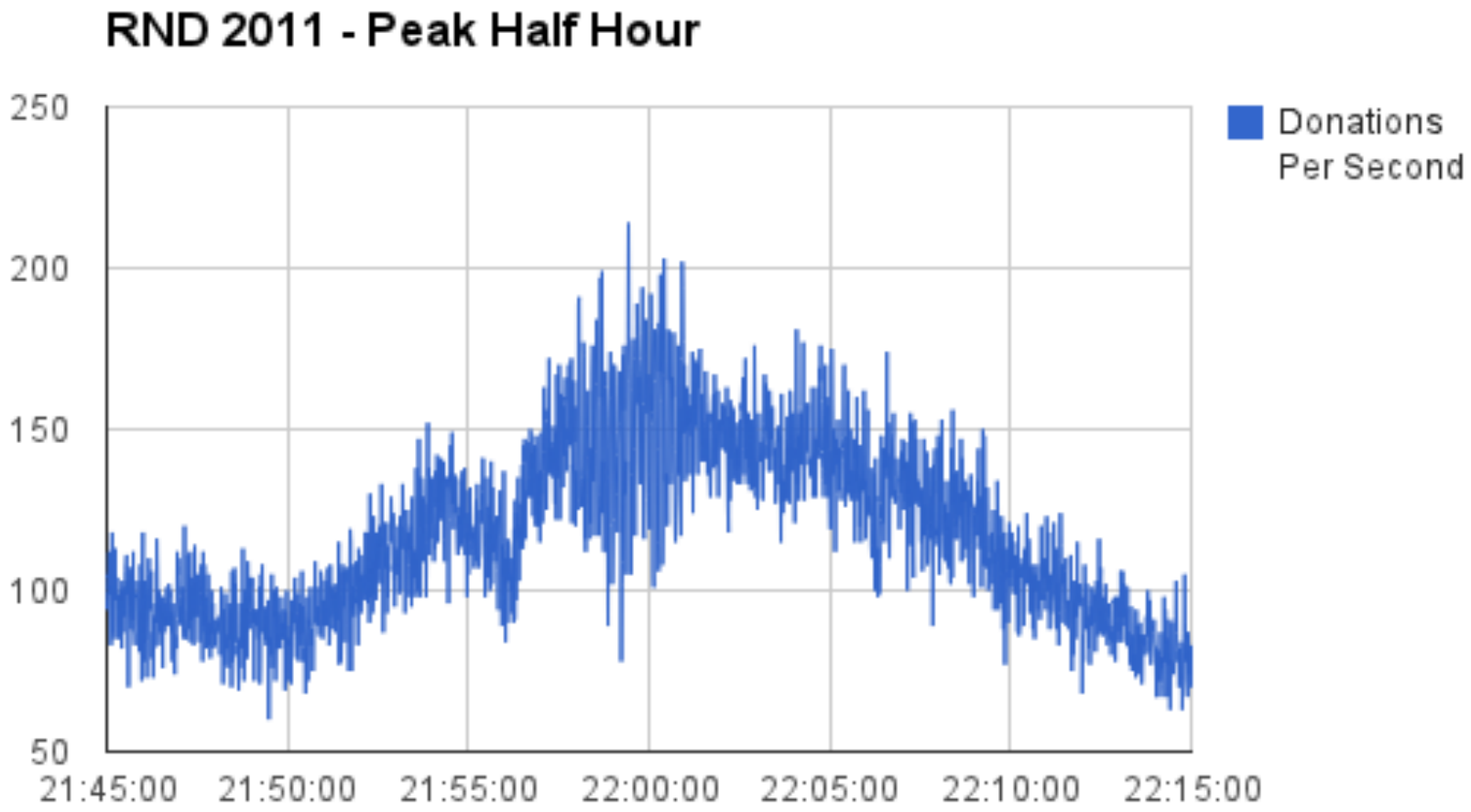
What does that look like?

armaxuni®

Donations Per Minute



Donations Per Second



Challenges

- We don't get a second chance
- Its only used once a year for 7 hours

armaxuni®

Previous Issues

- Testing, Integration and deployment problems
- Lack of consistency
- Single Points of Failure
  - Infrastructure provider
  - Platform & Networking
  - Bandwidth
- Multiple provider relationships

- 1 year feedback cycle

armakuni

Solution Patterns

- Distributed architecture
- Multiple Infrastructure as a Service (IaaS)
- Multiple Platform as a Service (PaaS)
- Stateless pattern
- Eventually consistent data
- Minimum Time to Recovery

armakuni

Solution Patterns

Stateless/Eventual Consistency

- No High Availability datastore
- Message Queue architecture
- Enables a distributed architecture

armakuni

Solution Patterns

PaaS & IaaS

- PaaS
  - Homogenised platform
  - Enables multi Iaas
- Multi IaaS
- Costs benefits for Comic Relief
- Prevents vendor lock in for Comic Relief
- Enabled rapid rollout of supporting applications

armakuni

Solution Patterns

Minimum time to recovery

- History
- Build for failure
- Reduce time to recovery

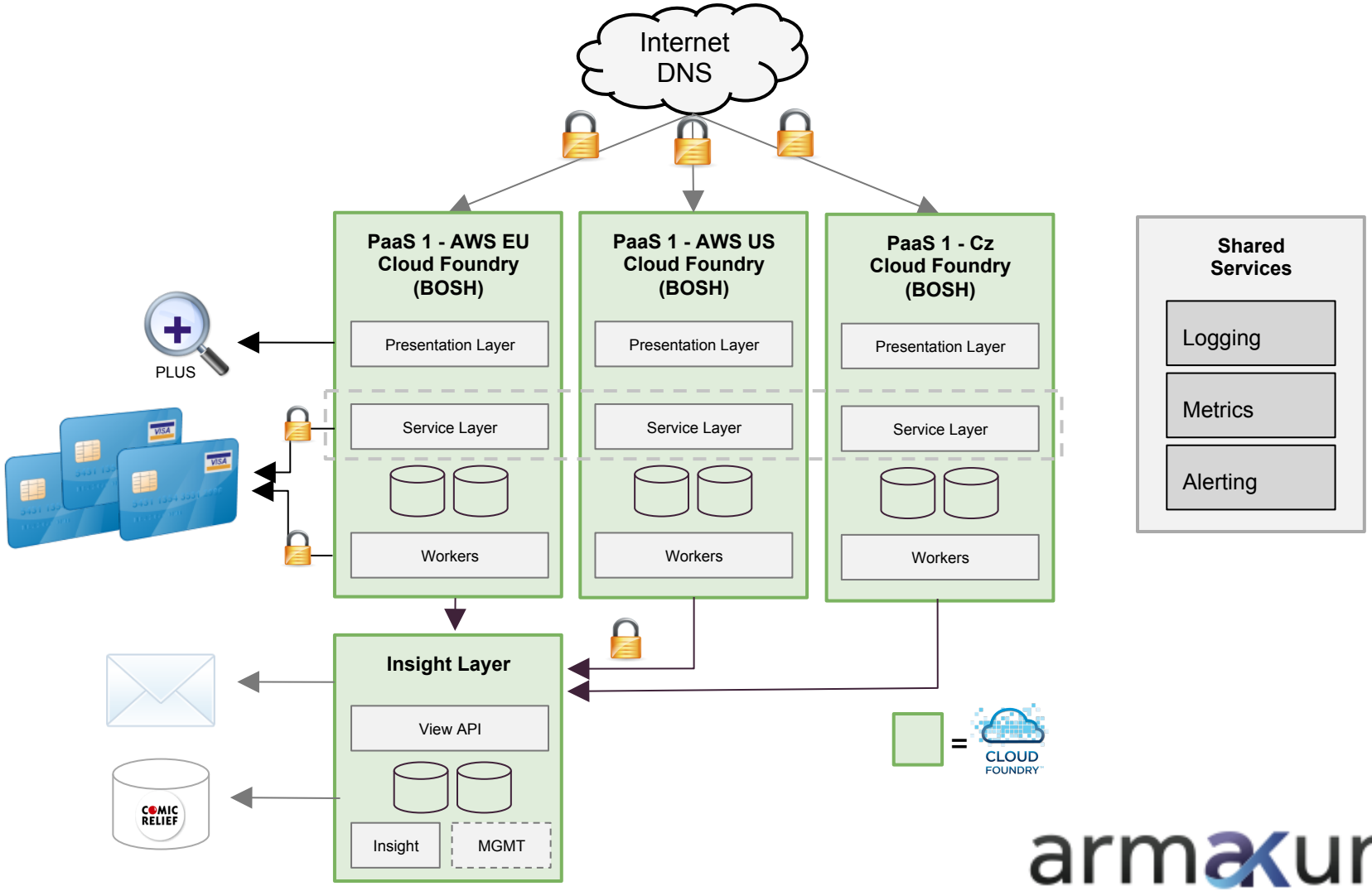
armaxuni®

Commoditise Dependencies

- Dependency on 3rd parties
- Usage commoditised
- IAAS
  - We can easily deploy across multiple service providers
  - Info provided by OpenCloudBrokers
- Payment Service Providers
  - We load balance across multiple providers, allowing us to ensure that our service is continuous, and able to cope with projected loads.

armakuni

What does the platform look like?



armakuni

Pipelines

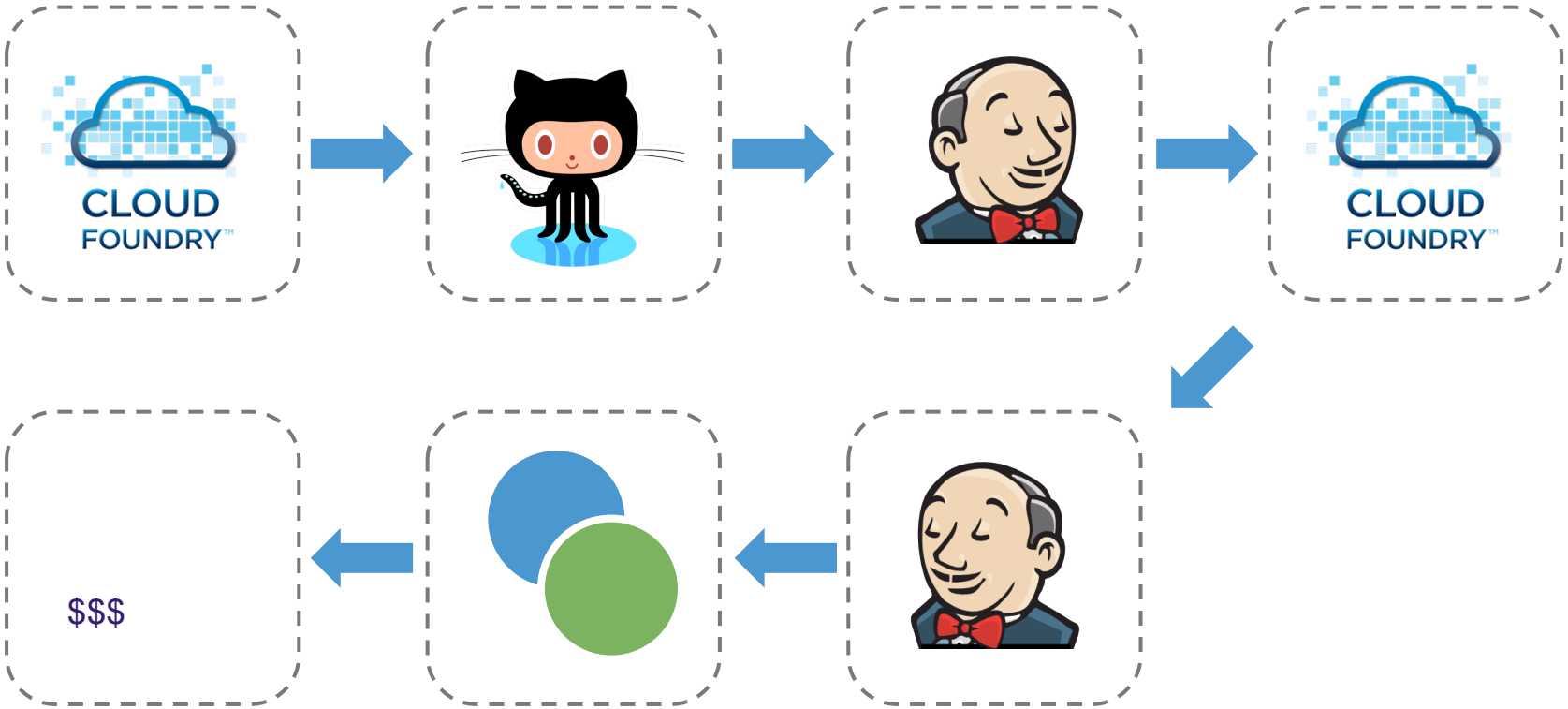
Continuous Deployment to Production

- 2 pipelines integrated
  - Infrastructure
  - Applications
- Converging on multiple test platforms
- Development team managing services

armakuni

Pipeline - Infrastructure

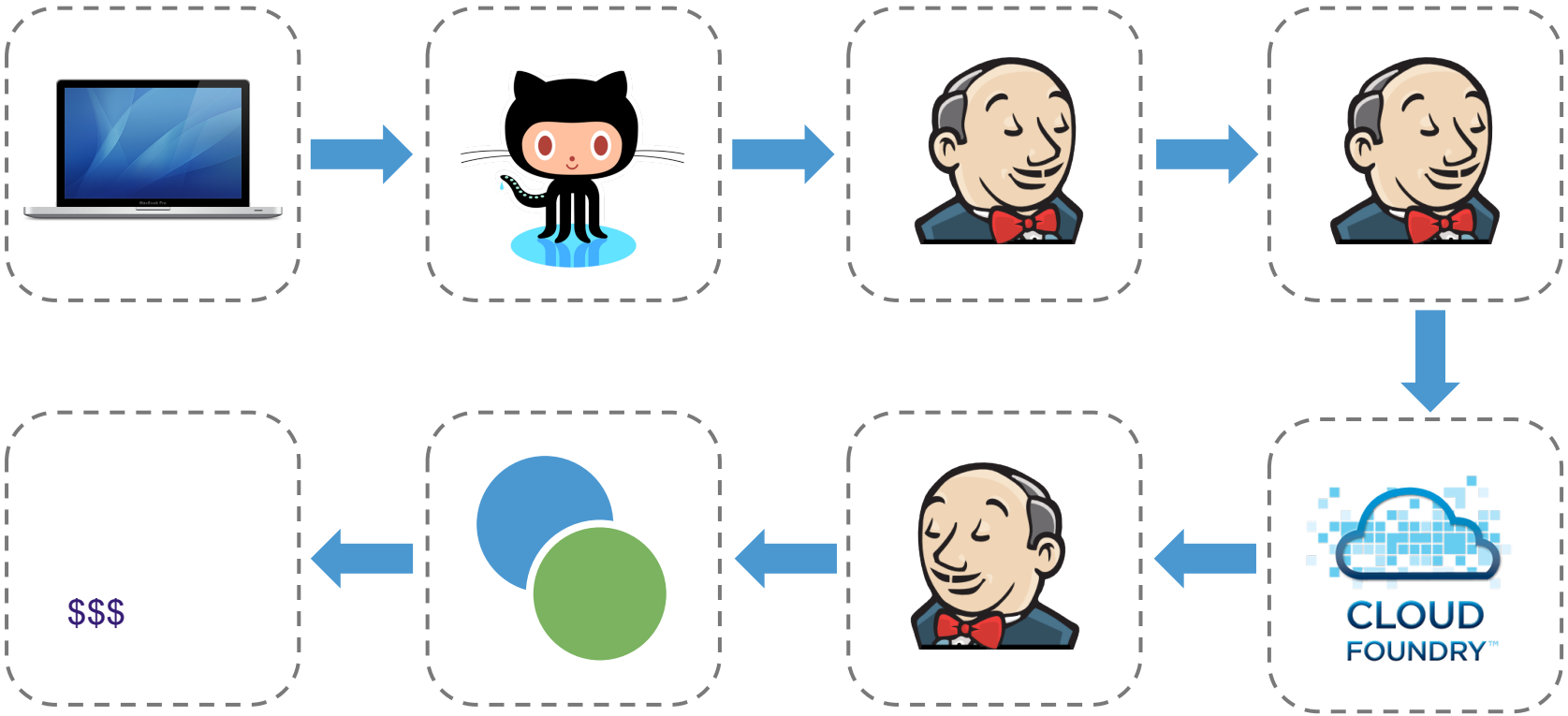
Local changes to deployed platform



armakuni

Pipeline - Applications

Local changes to deployed platform



armaxuni

Continuous Integration Testing

The value in our pipeline comes from the testing that gives us **confidence** in the **consistency** of our solution

- RSpec - unit tests
- Cucumber - feature/integration tests
- ZAProxy - security tests
- Grinder - benchmarking load tests

armakuni

Other Testing

Load Testing

- In addition to small scale load testing as part of our CI deployments
- Grinder, using chef to deploy
- 20 minutes lead time, up to 120 nodes used, 60,000 concurrent users (zero wait times)
- Global capability

armakuni

Failure Tolerant

- DNS round robin across multiple shards
- Scripted DNS enabling a measure of load balancing
- "Failure wagons" standing in in case of shard failure and handing off to alternate shards

armakuni

Failure Tolerant

- Minimum time to recovery vs high availability (HA)
  - Eventual consistency
  - Stateless requests
  - Message queue architecture
- **Expecting failure**

armakuni

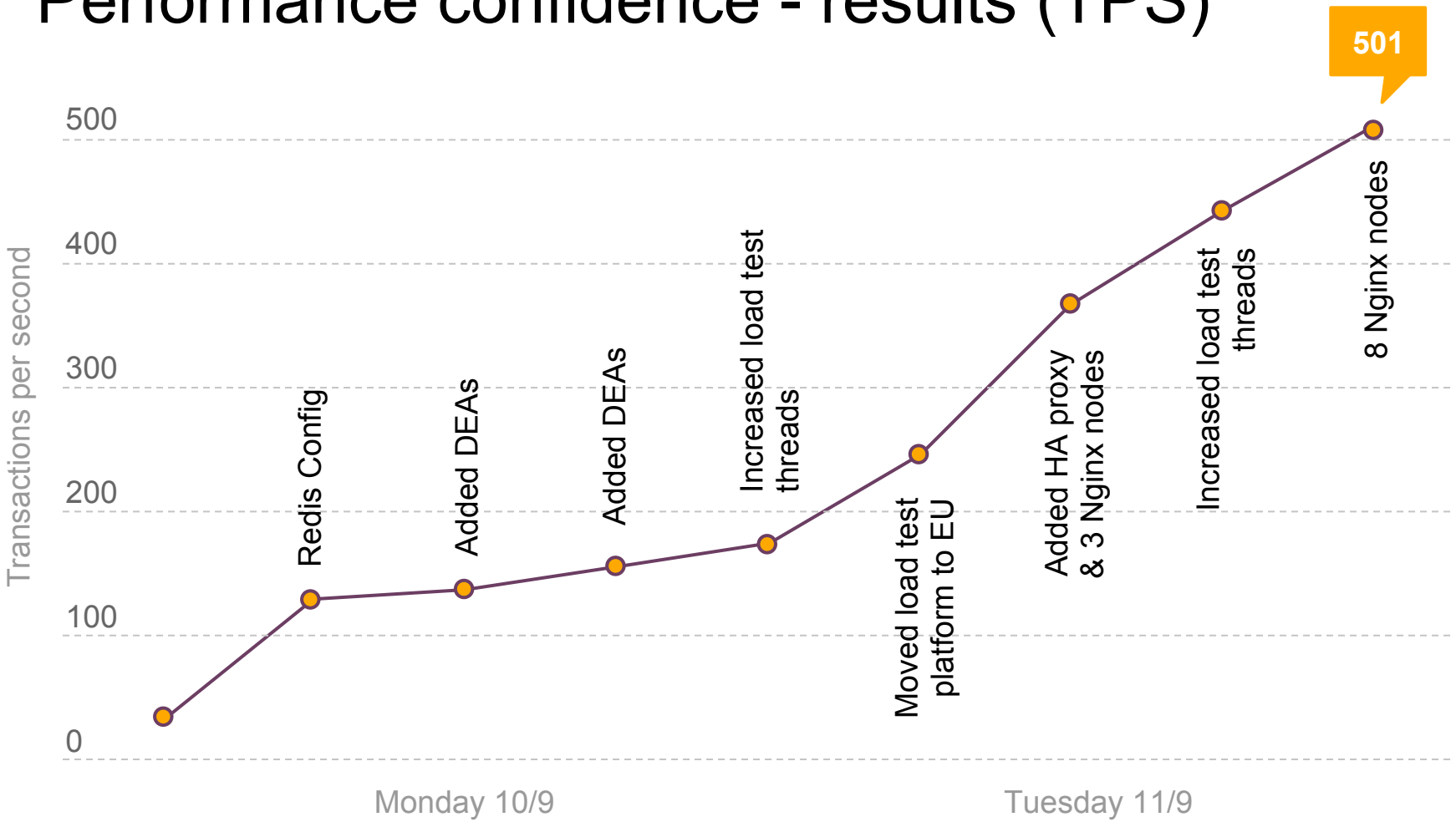
Solution Challenges

- Reliance on inflexible third-party providers
- Multiple payment providers, we are able to ensure that we have the redundancy we need.
- Managing and automating complexity

armakuni®

Flexibility - Load testing

Performance confidence - results (TPS)



armakuni

Flexibility - Supporting Platforms

Whilst building the main platform, we have also built a range of supporting platforms, including:

- Payment provider mocks (>= 500 Donations/sec)
- An email service mock
- A data api mock
- Globally-distributed load test platform (zero to hero in 20 minutes)

armakuni

Flexibility - Payment Service Providers

- We have performed implementations with 11 different payment providers/interfaces, (several of which are not being used.)

- These 3rd party integrations are key to the delivery of our service, and so this enabled us to really understand how they worked, what performance issues we might encounter.

armakuni

The part that's missing!

- no actual data/results
- please watch this space
- only 9 days to go

- The last 9 months have been tough but fun
- The pipelines, once created, have been the driving force of this project
- 3rd party service commoditisation has allowed Comic Relief to stay in control of the risk

- Thank you

armakuni

In Conclusion

- QCon is two weeks too soon
- By using the cloud we have put ourselves in a strong position
- New Platform will only be proven on 15th March
- We have a back-up platform built by BT

Don't forget to use the engage feature on the QCon app to rate the talk and ask questions

armakuni

tim.savage@armakuni.com
@timjsavage

z.hannick@comicrelief.com
@zenonhannick

armakuni