Dependability in a World of Cloud

Thomas Michael Bohnert,
Andy Edmonds, Christof Marti

INTEL European Research and Innovation Conference
Barcelona, Oct 2012
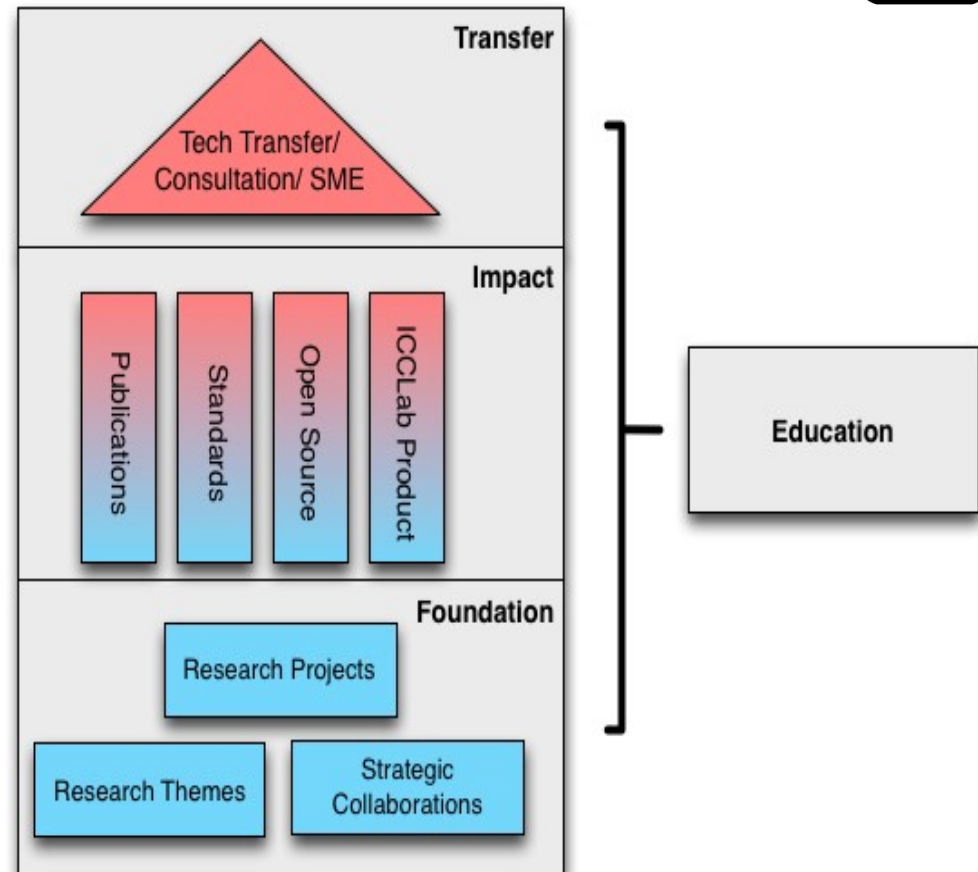
ICCLab
www.cloudcomp.ch, @ICC_Lab, #icclab

The ICCLab

Research Themes
- **IaaS**
- **PaaS**
- **MobileCloud**
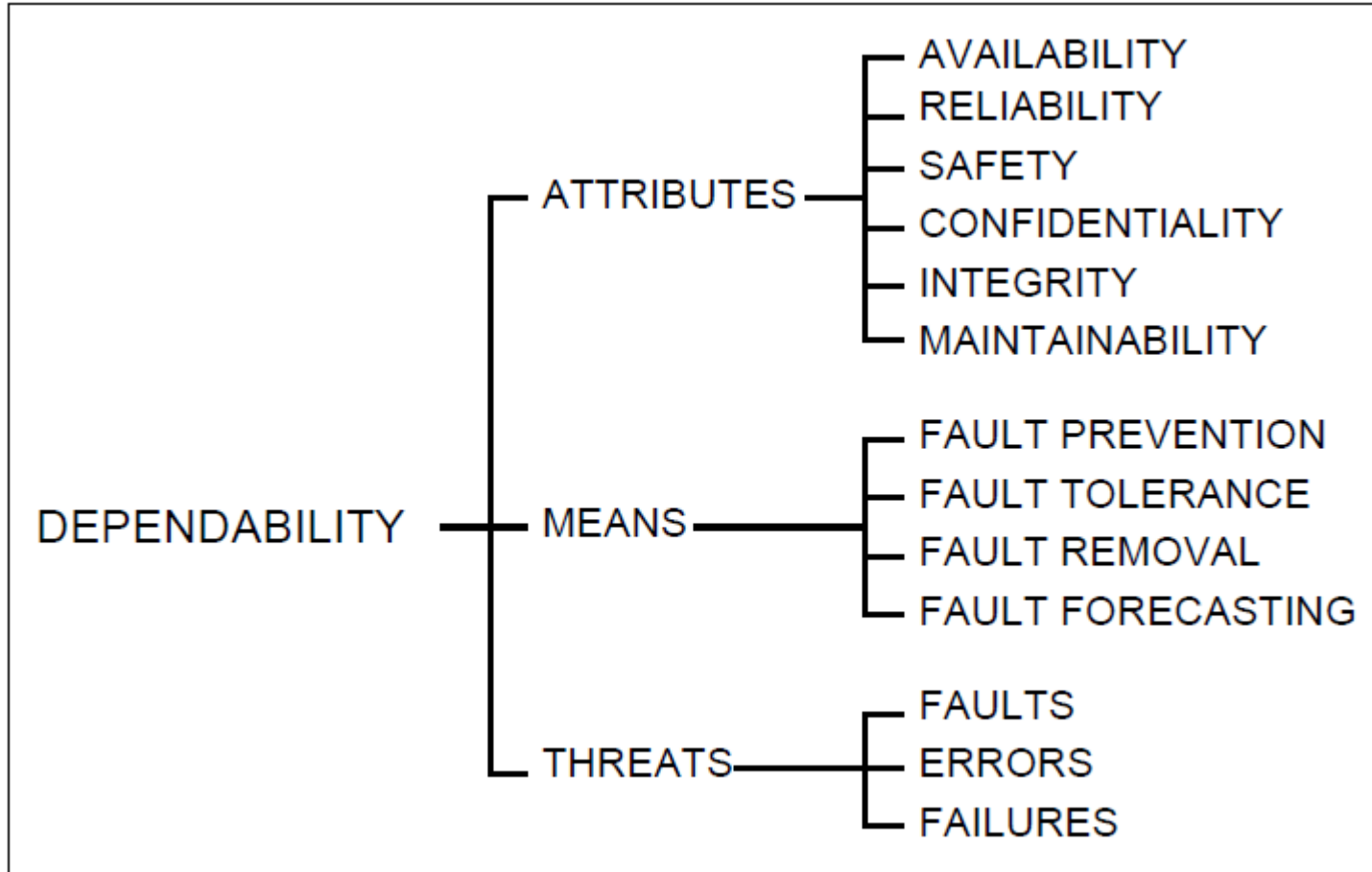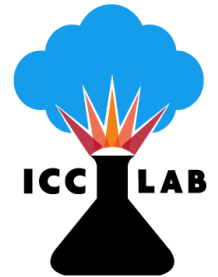- **Dependability**
- **SDN**

Genesis

- Concept of dependable computing first appears in 1830's
  - Babbage's Calculating Engine

- IEEE-CS TC on Fault-Tolerant FAILURES in Computing, 1970

- IFIP WG 10.4 Dependable Computing and Fault Tolerance, 1980
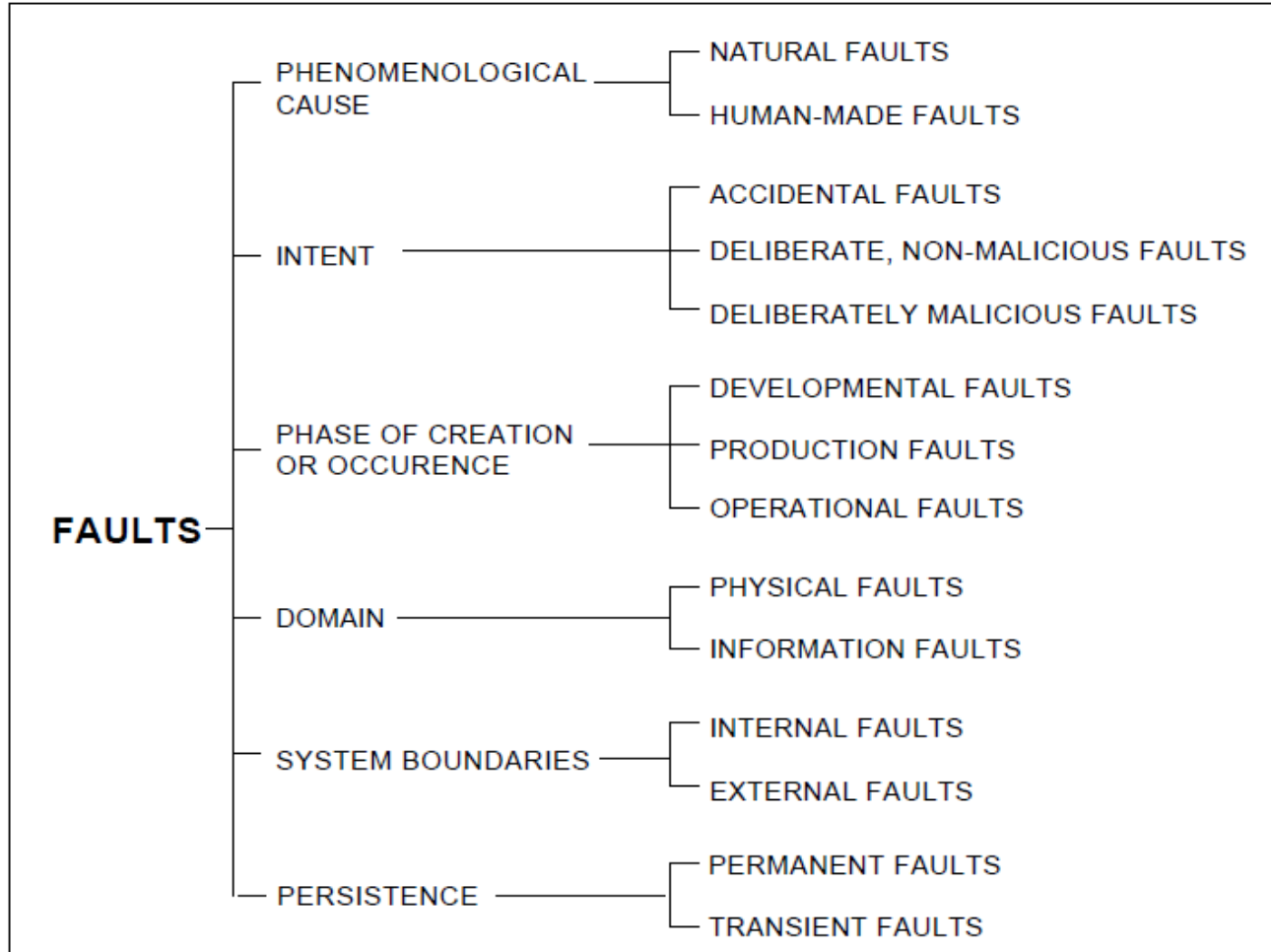
- Product perspective vs (Internet of) Service

Avizienis, Laprie, Randell, "Fundamental Concepts of Dependability"

Taxonomy

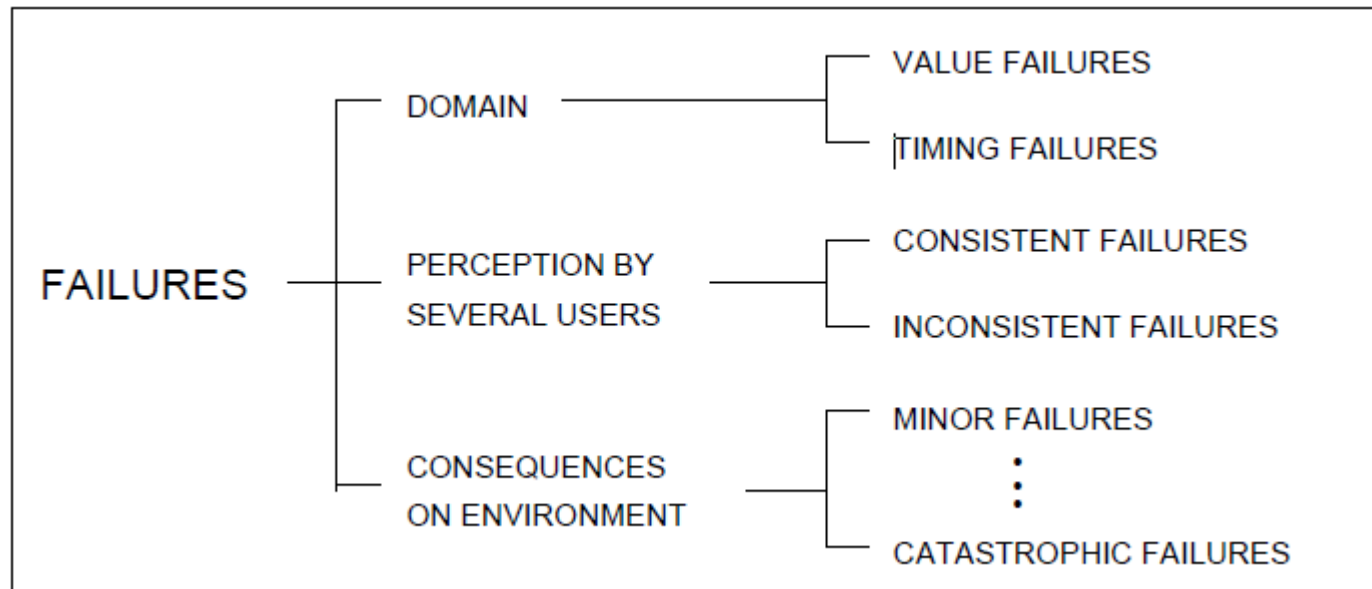Avizienis, Laprie, Randell, "Fundamental Concepts of Dependability"

Threats

- Faults, cause
  - Errors, cause
    - Failures, cause
      - Negative quality perception, causes
        - Low to no value

Avizienis, Laprie, Randell, "Fundamental Concepts of Dependability"

Dependability

Avizienis, Laprie, Randell, "Fundamental Concepts of Dependability"

Dependability

Avizienis, Laprie, Randell, "Fundamental Concepts of Dependability"

Cloud Computing

The Panacea!

Figure 1. Hype Cycle for Cloud Computing, 2011
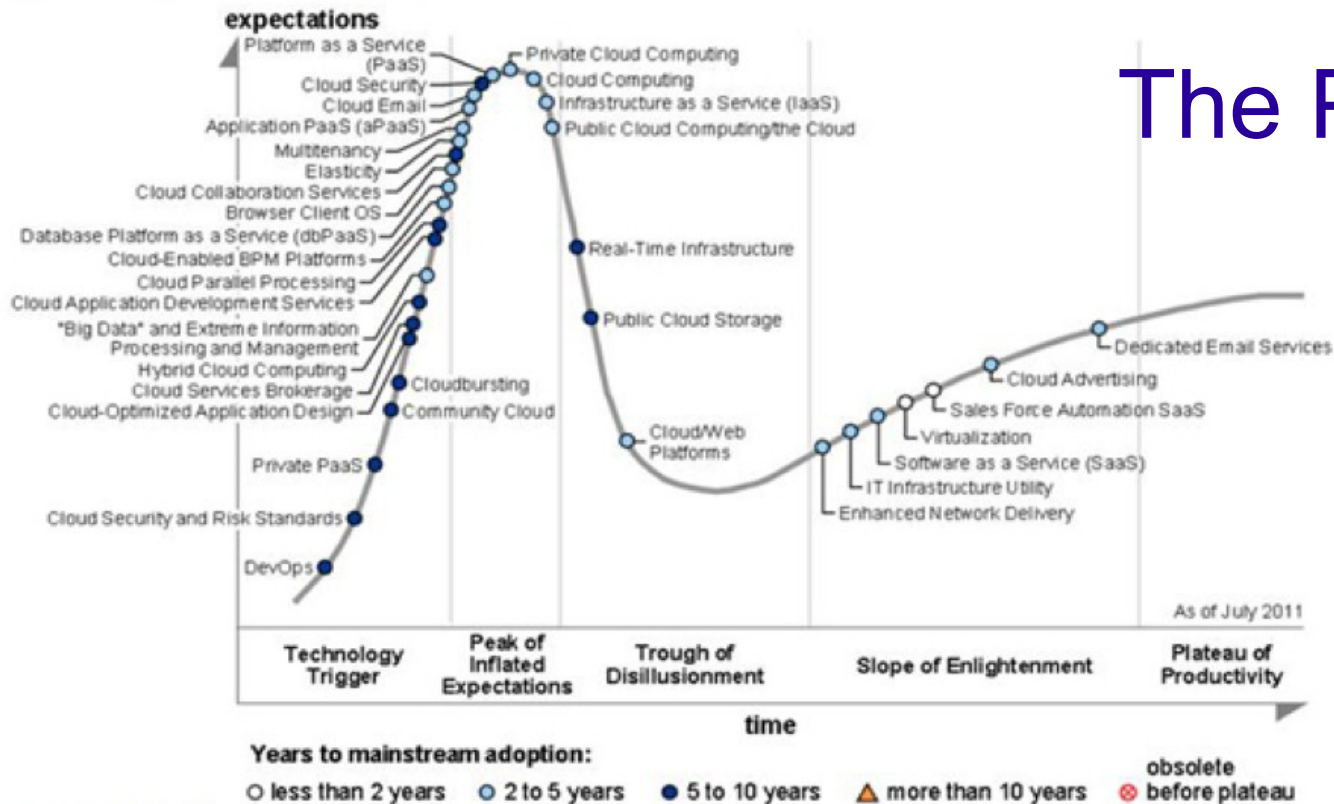
Cloud Computing
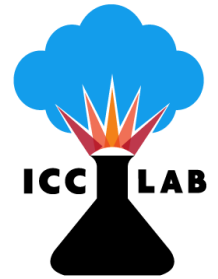
No worries ...
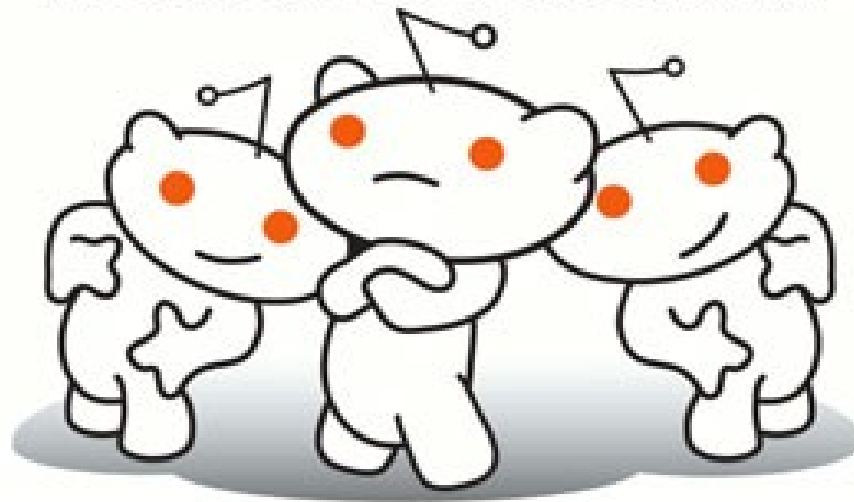
… it's in the cloud

No worries …

Amazon is currently experiencing a degradation. They are **working on it.**

reddit is down.

Apr 2011, Apr 2012, Jun 2012 …

No worries …

Reddit And Netflix Down? Amazon Network Issue Causes Downtime For Multiple Sites (UPDATE)

Posted: 10/22/2012 2:55 pm EDT Updated: 10/22/2012 5:56 pm EDT

An apparent issue with Amazon Elastic Cloud Compute network caused partial outages of several popular sites on Monday afternoon.

Social news site Reddit was inaccessible for some users. The site confirmed the downtime in a tweet sent around 1:30 p.m. ET on Monday: "The site is down right now. It appears to be a network-related issue. We are investigating."

(At the time of writing, the site appeared to be at least partly up, but pages were slow to load when they loaded at all.)

Diagnostics site DownRightNow also reported around the same time that Netflix was experiencing a "likely service disruption."

Image-sharing site Imgur was also down for some, as was vacation-booking site Aribnb, Mashable reports.

Reddit-watcher blog the Daily Dot was affected and tweeted that the problem was on the Amazon network's end. "Amazon has broken the Internet and taken the Daily Dot with it. We'll be back just as soon as possible," read a tweet from the site's official feed.

According to TechCrunch, Amazon's AWS servers (in addition to EC2 servers) in its north Virginia data center were the source of the problem: "[Amazon's] Elastic Block Store part seems to be down. All services hosted on Heroku's virtual private servers are still down at the time of writing." Twitter users reported that GitHub, Pinterest, Gamespot and other services were also down as a result.

Have you had a problem with any of these sites today? Let us know in the comments or tweet us @HuffPostTech.

UPDATE: 5:55 p.m. Amazon is working to restore normal service to the servers affected, per the company's Web Services Service Health Dashboard.
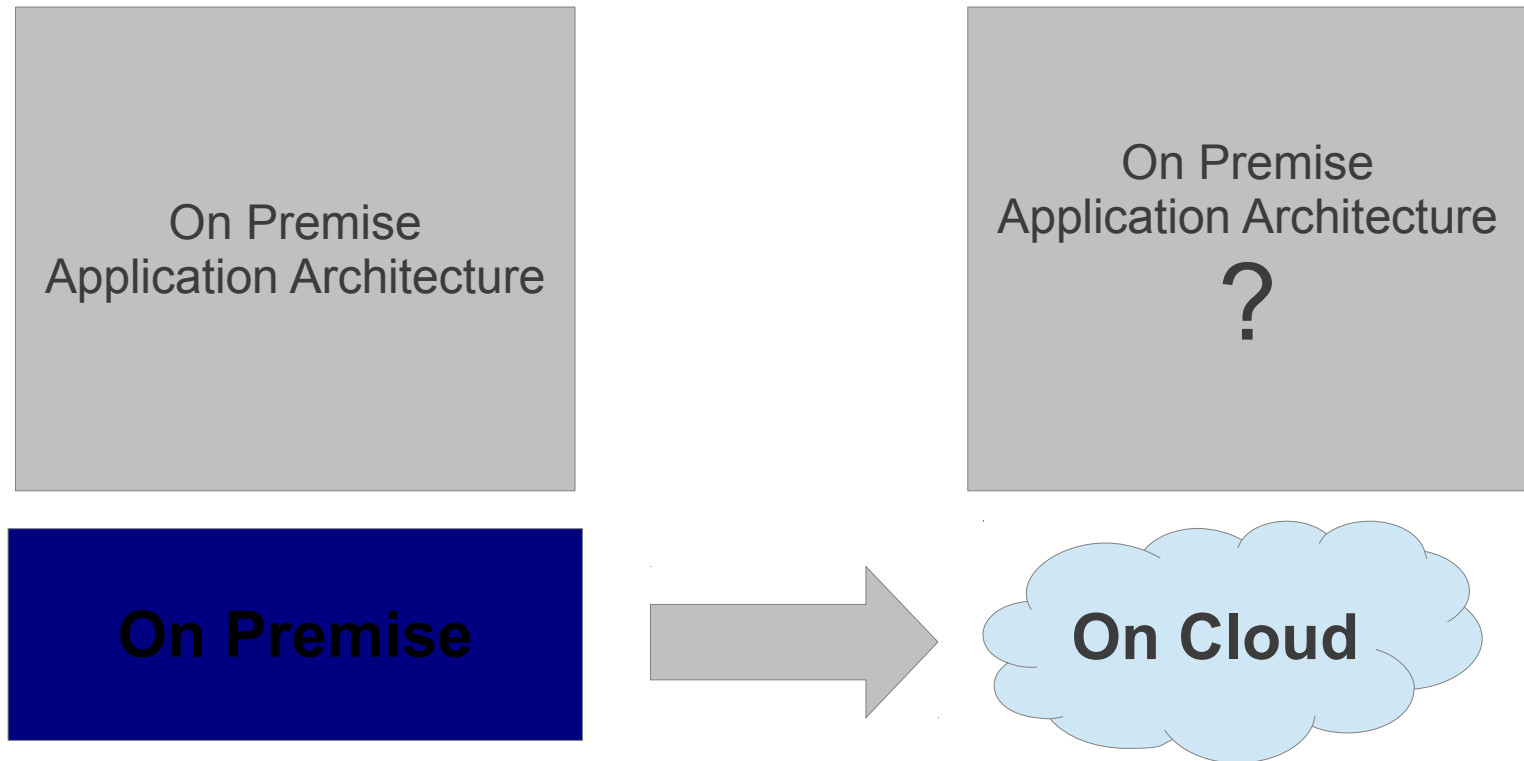
Yesterday!
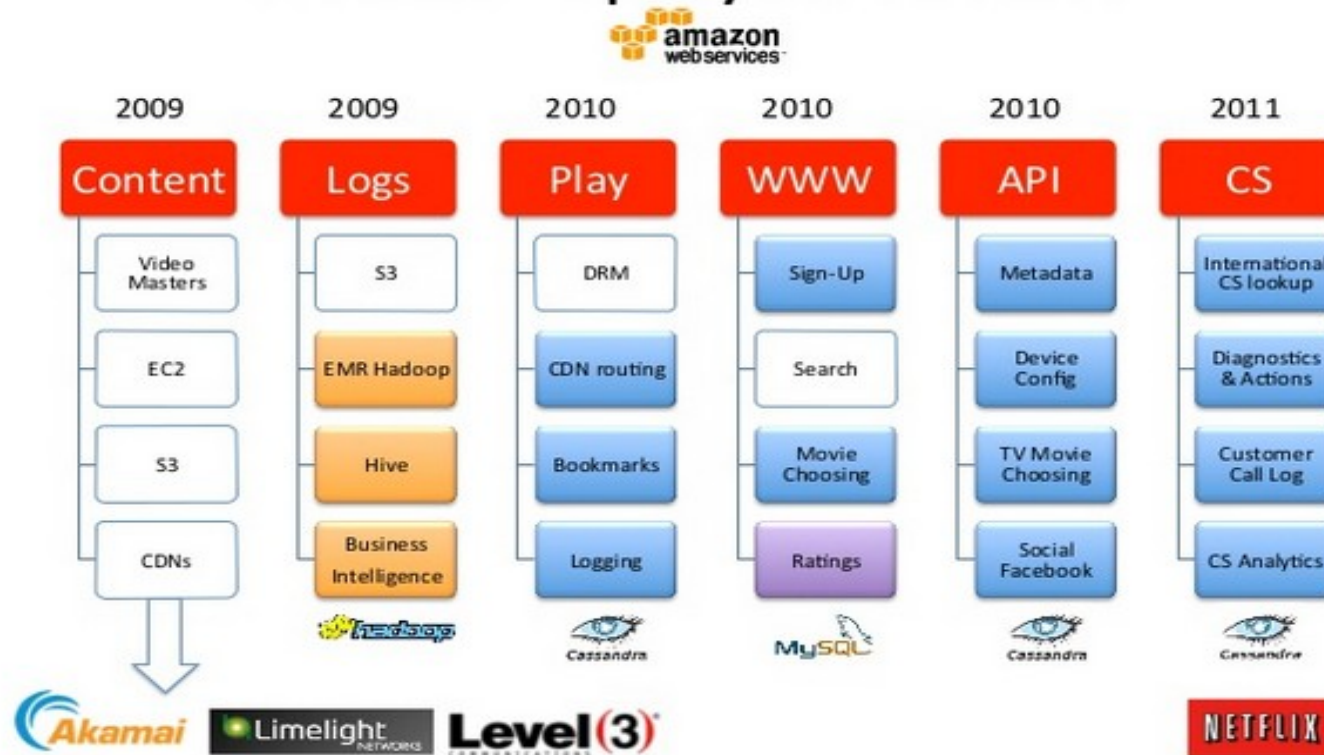
Cloud Computing Basics

Ease of Use, Reliance

Software as a Service — User

Platform as a Service — Developer

Infrastructure as a Service — IT Department

Tech Skill, Control

Design for dependability

On Premise
Application Architecture

On Premise

On Premise
Application Architecture
?

On Cloud

Design for dependability

Netflix Deployed on AWS

Netflix in the Cloud at SV Forum

by Adrian Cockcroft on Mar 27, 2012

Design for dependability

Design for dependability

Design for dependability

- Stateless Services

  - In the failure case requests can be routed to another service instance

- Data Stored Across Zones

  - Multiple redundant hot copies of the data spread across zones

Design for dependability

- Graceful Degradation

  - Fail Fast: Set aggressive timeouts

  - Fallbacks: Each feature is designed to degrade or fall back to a lower quality representation

  - Feature Removal: Remove on-critical features early

Design for dependability

- "N+1" Redundancy
  - Allocate extra capacity
- Create Real Failures
  - Simulate service failure
  - Drill your engineering team to accept and deal with failures as part of life
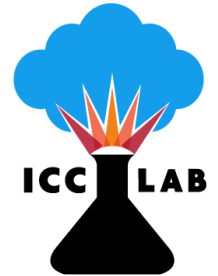
Design for dependability

- Multiple Region Support
  - Automate inter-zone migration
- Avoid EBS dependency
  - Keep stateful instances on S3

Dependability through federation

- Amazon essentially is an internal federated cloud
  - Zones within Regions
- Nonetheless, users are dependent on Amazon
  - Single point of failure, lock-in
- Need to be able to federate amongst Cloud Providers
  - Beyond Amazon → AWS + CloudSigma (+ ...)
- How?

Dependability through federation

Protocol and API for Management Of Cloud Service Resources.

Implementations

CompatibleOne
cloudstack — open source cloud computing
OpenNebula.org
amazon web services™
openstack™

Recommendations

NIST — National Institute of Standards and Technology — U.S. Department of Commerce
siena — Standards and Interoperability for eInfrastructure impleMeNtation initiAtive
egi

Cabinet Office
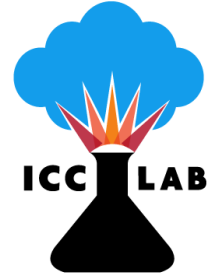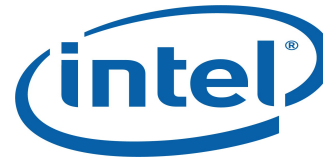Bundesministerium für Wirtschaft und Technologie — WIRTSCHAFT. WACHSTUM. WOHLSTAND.

Dependability through federation

- OCCI – An Intel innovation
- Demo?
  - Go see the demo in the show case area!

Dependability

- Definition needs to be updated
  - Need for trustworthiness through service inspection, introspection
- Propose the additional attribute of
  - **Transparency**
- **But why?**

Dependability

http://future-internet.appspot.com/

Version: 1 ▼

Total Logs Storage: 30 KBytes spanning 90 days (0% of the Retention limit) Total Logs Storage for Version: 30 KBytes (100% of Logs St

Show: ● All requests ○ Logs with minimum severity: Error ▼   Timezone: (GMT
⊞ Options

Tip: Click a log line to show or hide its details.

‹ Prev 20 **1-20** Next 20 › (Top: 1:27:32 ago)

⊞ 2012-10-23 00:44:15.006 / 304 61ms 0kb Mozilla/5.0 (compatible; Exabot/3.0; +http://www.exabot.com/go/robot)

⊞ 2012-10-23 00:44:14.651 /robots.txt 404 66ms 0kb Mozilla/5.0 (compatible; Exabot/3.0; +http://www.exabot.com/go/robot)
  W 2012-10-23 00:44:14.651 No handlers matched this URL.

⊞ 2012-10-23 00:32:08.882 / 200 2ms 0kb Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)

⊞ 2012-10-23 00:32:08.790 /robots.txt 404 12ms 0kb Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
  W 2012-10-23 00:32:08.790 No handlers matched this URL.

⊞ 2012-10-23 00:02:10.414 /guestbook 302 49ms 0kb Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)

⊞ 2012-10-23 00:02:10.052 /guestbook 302 2401ms 0kb Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)
  I 2012-10-23 00:02:10.052 This request caused a new process to be started for your application, and thus caused your application code

⊞ 2012-10-23 00:02:07.512 /robots.txt 404 76ms 0kb Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)
  W 2012-10-23 00:02:07.511 No handlers matched this URL.

⊞ 2012-10-22 00:46:39.384 /favicon.ico 404 651ms 0kb Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/536.11 (KHTML, like Gecko) Ubuntu
  W 2012-10-22 00:46:39.373 No handlers matched this URL.

www.cloudcomp.ch

Enough?

Transparency

But Why?

- Know what you're paying for

- Early detection of degrading performance

- Empower user to act

Joyent and CloudSigma are good examples

- expose great deal of information to end users

- Enable users to take control of "their" services

- Users are pro-actively involved in the SLA

Closing

- Dependability
  - Needs service perspective
- Cloud dependability
  - To be supported by cloud provider
    - Transparency
  - But ultimately also your responsibility
- Design for dependability
  - Cloud application design

Thanks, questions?

Backup

SLAs

Slide/topic could be dropped

Not necessary to talk about