Chapter 1
Introduction

Mastering Cloud Computing
Paul Talaga

UNIVERSITY OF Cincinnati

As of now, computer networks are still in their infancy, but as they grow up and become sophisticated, we will probably see the spread of 'computer utilities' which, like present electric telephone utilities, will service individual homes and offices across the country.

-Leonard Kleinrock, 1969, ARPANET

UNIVERSITY OF Cincinnati

The vision...


I need to grow my infrastructure, but I do not know for how long...
I have a lot of infrastructure that I want to rent ...
I have a surplus of infrastructure that I want to make use of
I cannot invest in infrastructure, I just started my business....
I have infrastructure and middleware and I can host applications
Global Cloud Marketplace
I want to focus on application logic and not maintenance and scalability issues
I have infrastructure and provide application services
I want to access and edit my documents and photos from everywhere..

Defining Cloud Computing

UNIVERSITY OF Cincinnati

Huh?

- It's a Buzzword!
- Term for MANY ideas and concepts

Cloud computing refers to both the applications delivered as services over the Internet and the hardware and software in the datacenters that provide those services. - Armburst

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. - NIST

UNIVERSITY OF Cincinnati

Reese's definition (as a utility)

- The service is accessible via a Web browser or a Web services application programming interface (API)
- Zero capital expenditure is necessary to get started.
- You pay only for what you use as you use it.

UNIVERSITY OF Cincinnati

A cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers.

Buyya et al, (our text author)

UNIVERSITY OF Cincinnati

What types of clouds exist?

Cloud Deployment Models

Public/Internet Clouds

* 3rd party, multi-tenant Cloud infrastructure & services:

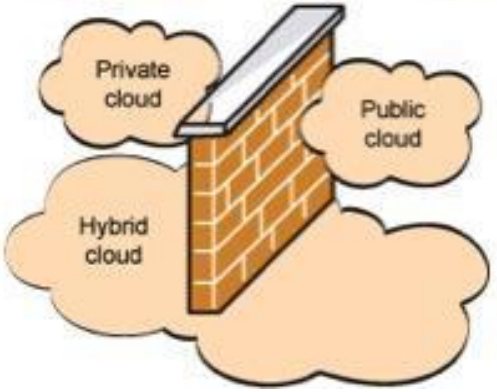
* available on subscription basis to all.

Private/Enterprise Clouds

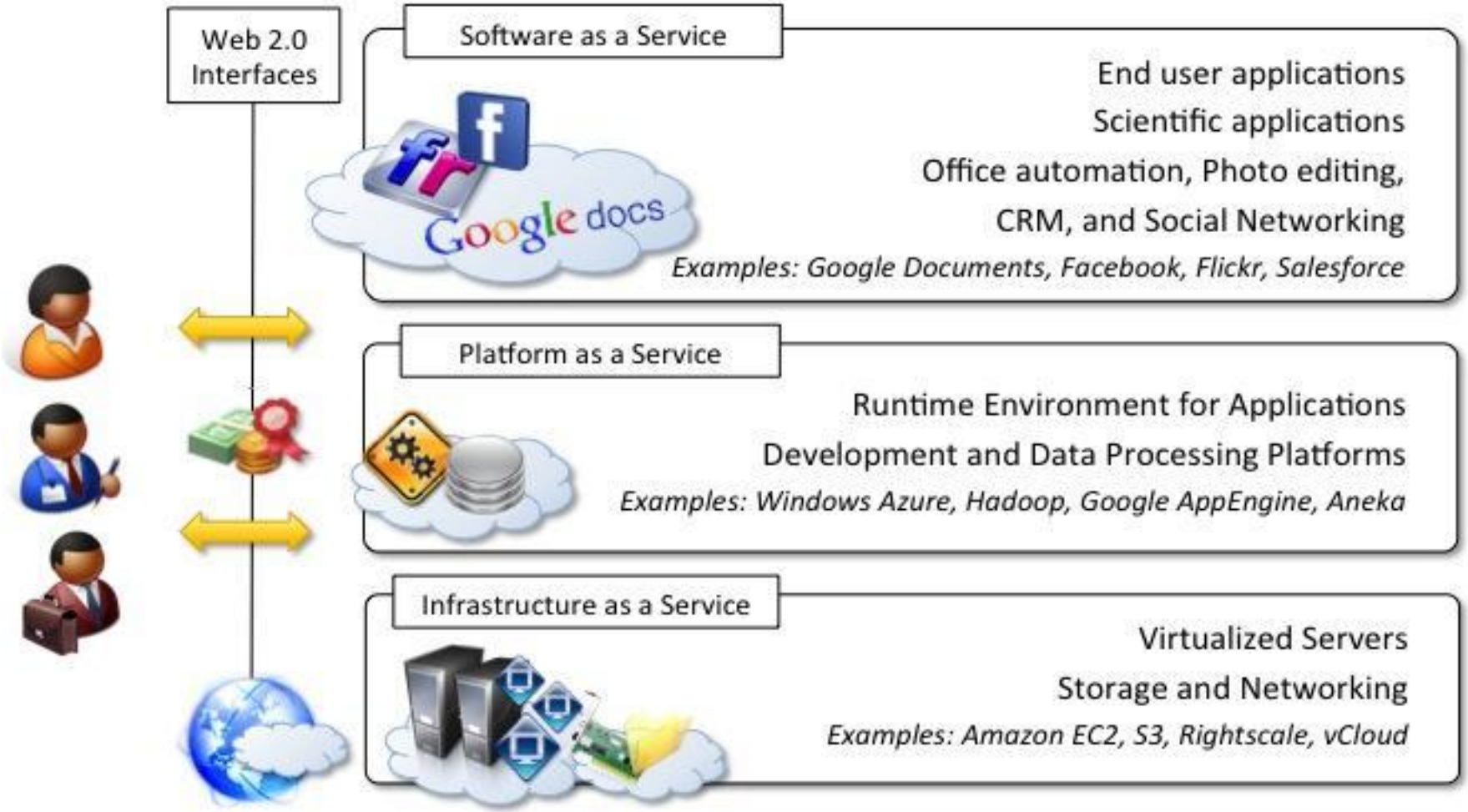
* A public Cloud model within a company's own Data Center / infrastructure for internal and/or partners use.

Hybrid/Inter Clouds

* Mixed usage of private and public Clouds: Leasing public cloud services when private cloud capacity is insufficient



Reference Models



What are the benefits to a (regular) business?

- No up-front commitments
- On-demand access
- Nice pricing (capital costs -> utility costs, no depreciation)
- Simplified app acceleration and scalability
- Efficient resource allocation
- Energy efficiency??
- Seamless creation and use of third-party services

UNIVERSITY OF Cincinnati

Benefits for a software company going to SaaS?

- NO deployment issues (CDs, downloads, etc)
- No need to support multiple OSs
- Faster to market
- A/B testing of features
- Efficiency and reliability now key
- More efficient developers, just try it!

UNIVERSITY OF Cincinnati

Storms may be on the horizon..

- Security
  - Confidentiality, Secrecy, Protection
- Legal
  - Google/Facebook privacy
  - Differing viewing laws
- Latency & Data Location (later)

UNIVERSITY OF Cincinnati

How did we get here?

5 core technologies:

1. Distributed Systems
2. Virtualization
3. Web 2.0
4. Service-oriented computing
5. Utility-oriented computing

UNIVERSITY OF Cincinnati

1. Distributed Systems

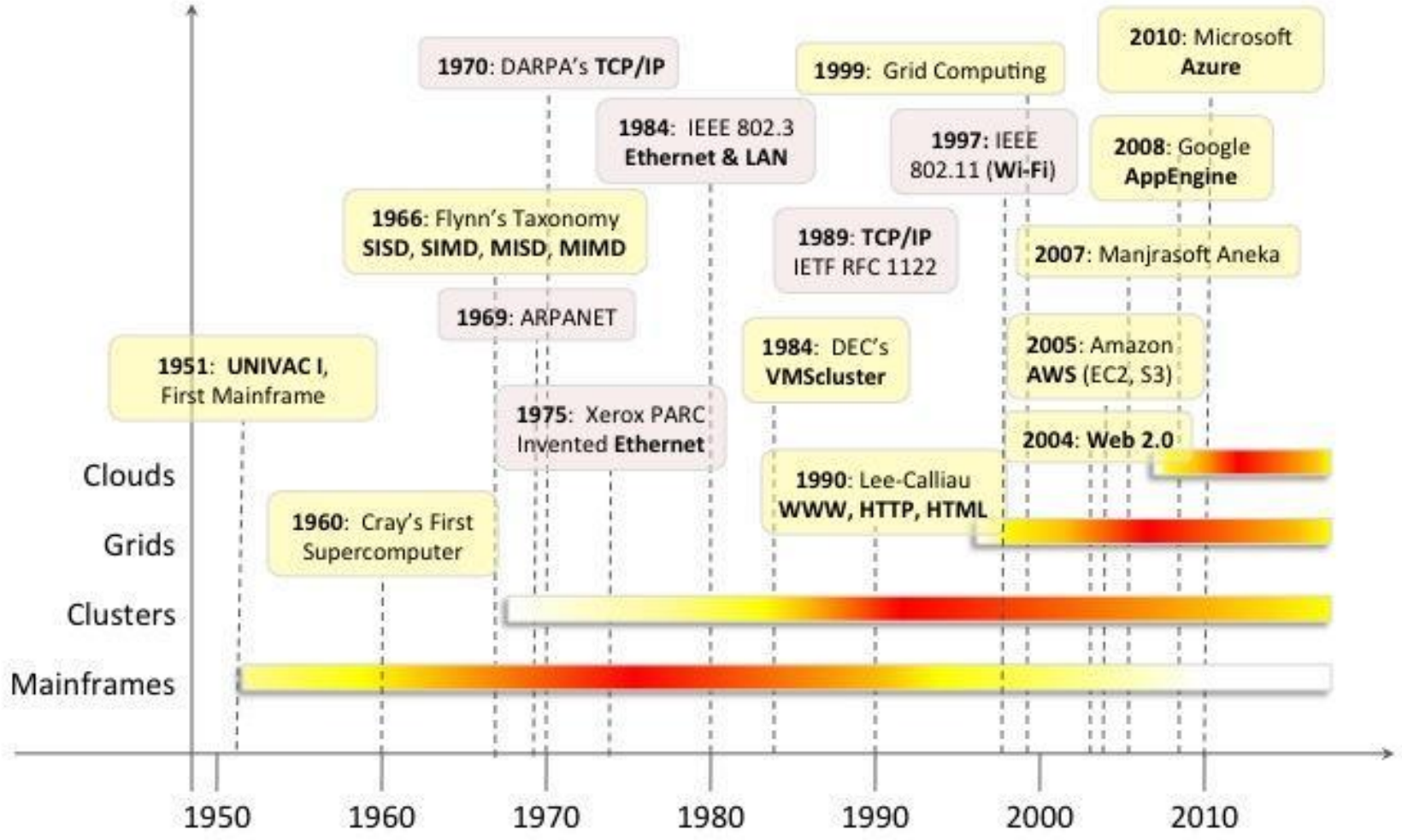
A distributed system is a collection of independent computers that appears to its users as a single coherent system. - Tanenbaum

To share resources for better utilization.

We have a course in Distributed Systems!

UNIVERSITY OF Cincinnati

What is old is new again...



Mainframes (1951 onward)

- Multiple processing units
- Powerful, reliable, IO optimized
- Timesharing systems
- Replacement of components while running - always on
- Still used for transaction processing: banking, airline ticketing, registration

IBM z990
2003
256 GB RAM
1k+ VM Linux
11k SSL con/sec

IBM EC12
2012
101 CPUs
5.5Ghz hex core
Integrated SSDs
6k Linux VMs

UNIVERSITY OF Cincinnati

Clusters (1968 onwards)

- Networked commodity (cheap) machines
- Physically close (room/building/LAN)
- Good for processing work, not IO
- Easy to expand
- Some use spare cpu cycles
- Condor, Beowulf Clusters, Message Passing Interface (MPI)
- Ohio Supercomputer

UNIVERSITY OF Cincinnati

Grid Computing (1990s onwards)

- Like cluster, but heterogeneous nodes
- Large physical distances
- Utility computing idea
- Needed high bandwidth connections (Internet)
- Good for processing work, not IO
- SETI@Home, BOINC

UNIVERSITY OF Cincinnati

But what about latency!?!

Talaga's Dissertation/Digression

- A semi full of hard drives has HUGE bandwidth!
- We get sold with bandwidth, but low latency is what we're after.
- But the speed of light is fixed!
  - Significant drop in sales for > 500ms response
  - Google goal is < 200ms
  - NY to LA is 74ms RTT
- Grids may be good for some workloads, but HORRIBLE for others (IO and communication)
- Mainframe->Cluster->Grid, latency between nodes increases.
- Data location matters, and how your app uses data.

UNIVERSITY OF Cincinnati

Cloud Computing

- Next evolution after grid computing
- Infinite capacity
- Resilient to failures
- Always on
- Built using commodity machines
- Pay-per-use (Utility vision)

UNIVERSITY OF Cincinnati

True root (my view)

The implementation and wider use of **distributed systems** theory to build distributed databases and file systems allowed cloud computing to take off.

- Huge datasets (multiple petabytes)
- Vector clocks (Lamport timestamps)
- Paxos (Wiki)

UNIVERSITY OF Cincinnati

2. Virtualization

- Abstract core computing elements away
  - Processor
  - Storage
  - Networking
- Hardware virtualization (VMware, VirtualBox, XEN, EC2, etc.…)
  - Most performance issues solved
- Process virtualization (Google AppEngine, Azure, Java)

UNIVERSITY OF Cincinnati

3. Web 2.0

- Web 1.0? - Static pages
- Web 2.0 is:
  - “Web as platform” - *John Battelle and Tim O'Reilly*
  - Interactivity & flexibility - Allow users to change a site’s content!
  - Asynchronous JavaScript and XML (AJAX)
  - Web Services
- Not a ‘next version’ of the web, but a way of using HTTP/HTML
- Examples: Google Docs, Facebook, YouTube, Wikipedia

UNIVERSITY OF Cincinnati

4. Service-oriented Computing

A component that can perform any function.

- Loosely coupled
- Reusable
- Programming language independent
- Location transparent

By layering services we can build a service-oriented architecture (SOA)

UNIVERSITY OF Cincinnati

Service-Oriented Computing

Important attributes:

- Quality of Service (QoS)
  - Service attributes, response times, security, uptime, etc…
- New software delivery model
  - Can sell components, not entire programs
  - Access through the internet
  - HTTP
  - Web Service Description Language (WSDL)
  - Simple Object Access Protocol (SOAP)

5. Utility-oriented Computing

Storage, compute, applications, infrastructure, all on a pay-per-use basis.

Old idea.. job queueing systems, OS time-slicing, all developed in mainframes to charge for use.
Buying services/products online common now.

UNIVERSITY OF Cincinnati

Current Platforms

- Amazon Web Services (AWS) - Current leader in IaaS: Elastic Compute Cloud (EC2), Simple Storage Service (S3), many many others…..
- Google AppEngine - PaaS - Python, Java, Go
- MS Azure - IaaS & PaaS
- Hadoop - Open Source framework for MapReduce
- Force.com & Salesforce.com