

Cloud Computing CMPU3007

Lecturer: Dr. Martin O'Connor

Module Structure

Weekly Delivery:

- 2 x 1hours lectures
 - Tuesday @ 10am-11am in room CQ-309
 - Wednesday @ 11am-12pm in room CQ-309
- 2 hours labs
 - Group TU857/3A on Thursday @ 4pm-6pm in lab CQ-236
 - Group TU857/3B on Thursday @ 4pm-6pm in lab CQ-235
- 4 hours independent learning per week.
- **IMPORTANT:** **Always check the Online Timetable for changes/updates for the first four weeks.**

Module Structure

Module Assessment:

- 50% continuous assessment
 - Provisionally: Week 6 In-lab assessment worth 25%
Week beginning 20st October.
 - Provisionally: Week 12 In-lab assessment worth 25%
Week beginning 1st December
- 50% end of term written examination

Module Structure

Brightspace

- Cloud Computing CMPU3007
 - There are two modules that match the above (confusing, I know!)
 - Please ensure you are enrolled in the one with the blue banner with text **TU857**
- ~~Self-enrolment~~
 - ~~Discover Tool Instructions:~~
~~<https://www.tudublin.ie/connect/vle/brightspace-for-students/bitesized-brightspace-student-guides/>~~
 - Install the Brightspace 'Pulse' app on your phone for class notifications

Module Objectives

What?



Why?



How?



What is Cloud Computing



A customer-oriented definition..

- Anytime
- Anywhere
- Any Device
- Any Service



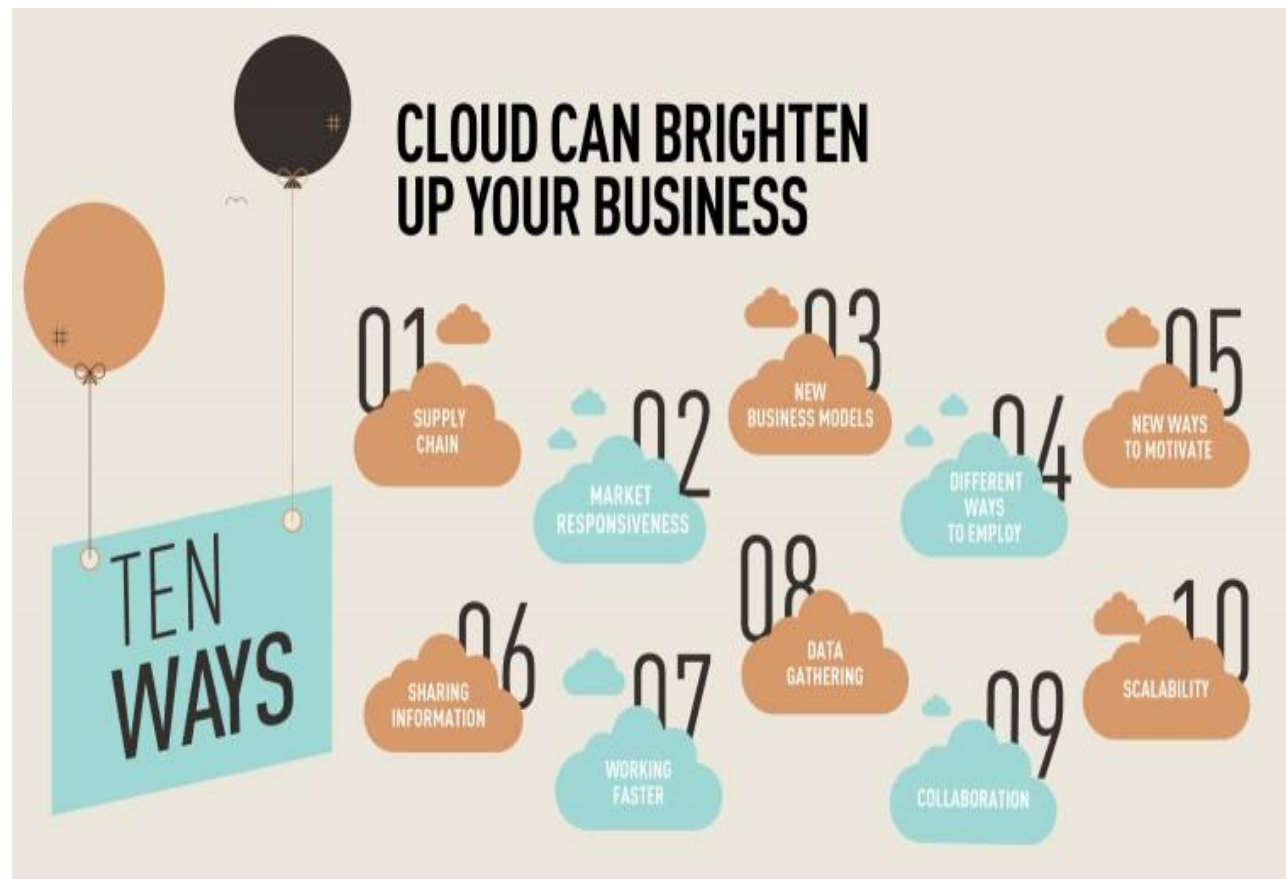
How many of these do you still store on your local computer?

What is Cloud Computing



A business-oriented definition..

- Universal access
- Scalable Services
- Collaborative
- New revenue models



What is Cloud Computing



A developer-oriented definition..

Cloud engineer skills at a glance

- Distributed Computing (hmmm)
- Scalable
- Access to 3rd party Functionality
- Mobility

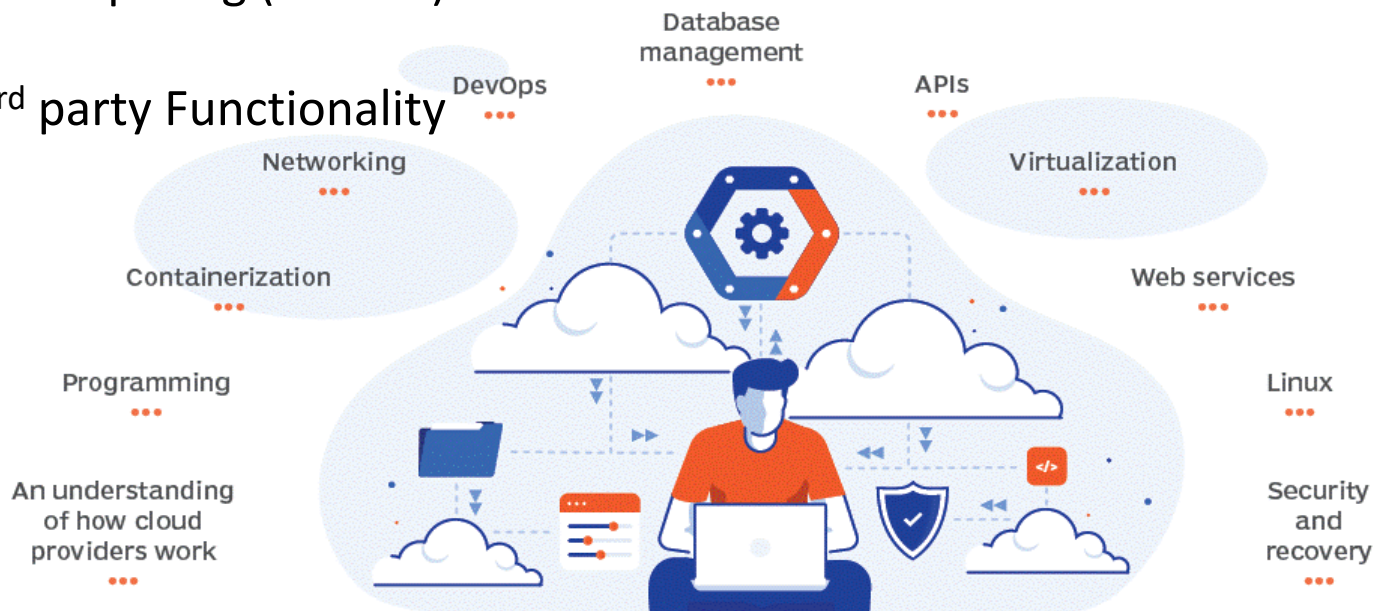


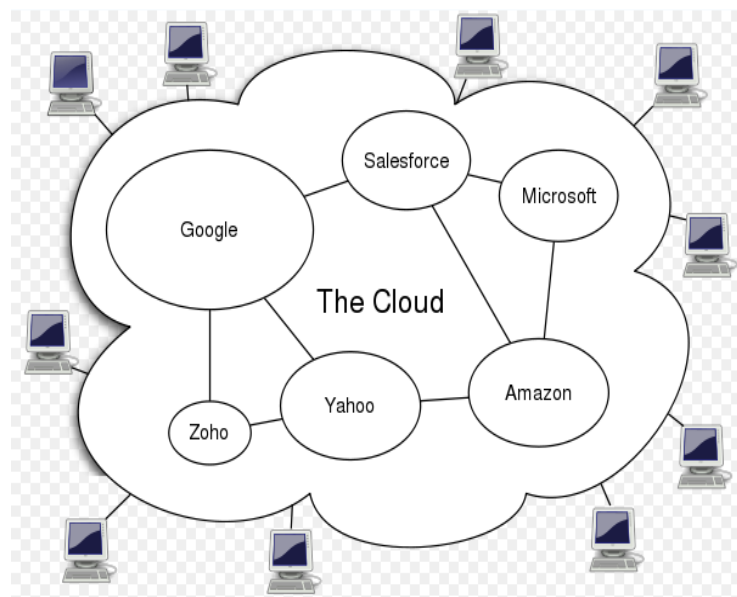
ILLUSTRATION: DARKOVUI/ADOBE STOCK

©2021 TECHTARGET. ALL RIGHTS RESERVED. TechTarget

Why is it important?



A new way to deliver services



Internal v External

SaaS
 Software as a Service

PaaS
 Platform as a Service

IaaS
 Infrastructure as a Service

Everything as a service ...
 ... pay for what you use.

Software as a Service

- SaaS (ready to use)
 - Most basic form of cloud computing
 - No development necessary for user
 - No resources required from user (just a browser)
 - Software managed from a central location
 - Software delivered in a “one to many” model
 - Users not required to handle software upgrades and patches
 - Application Programming Interfaces (APIs) allow for integration between different pieces of software
- Offer powerful tools right at web browser
- Requires no installation
- Requires no specific knowledge of user

SaaS examples (i)

- Google Docs
 - Productivity suite
 - Free to use (free Google Account required)
 - Share documents with others
 - Try it: docs.google.com



Google docs



Google™ Apps



SaaS examples (ii)

- Dropbox



- Online file storage
 - Automatic synchronization across computers
 - Sharing of files with other users
 - Web access + mobile access
 - Backup of data + restore (30 days)

SaaS examples (iii)

- Salesforce (CRM)
 - Sales cloud – automation of workflow
 - Service cloud – customer service
 - Chatter – collaboration tools
 - Jigsaw – customer contact database
 - ...
 - Force.com – Salesforce platform for third party apps (PaaS)
 - Heroku – third party ruby apps (PaaS)



From: <http://www.salesforce.com>

Platform as a Service

- Platform as a Service
 - “...provides developers with proprietary API's to make an application that will run in a specific environment.”
 - Can be any application you can think of
 - Is locked to the platform used for creation
- PaaS example: Metaverse apps



PaaS examples (i)

- PaaS example: metaverse apps
 - PHP, Java, Python, C#
 - Social Plugins
 - Graph API
 - Platform Dialogs
 - Authentication
 - Chats
 - Ads
 - ...



<https://developers.facebook.com/>

PaaS examples (i)

- PaaS example: Facebook apps
 - JavaScript SDK
 - iOS SDK
 - Android SDK
 - PHP SDK



PaaS examples (ii)



App Engine

- PaaS example: Google App Engine Application Platform
 - “Google App Engine lets you run your web applications on Google's infrastructure”
 - Java, Python, Ruby, GO, PHP, C#, Javascript SDKs
 - Develop and test locally, automatically upload, deploy and run in the cloud (serverless)
 - Pay per use model

<https://cloud.google.com/appengine>

PaaS examples (ii)



- PaaS example: Google App Engine
 - dynamic web serving, with full support for common web technologies
 - persistent storage
 - automatic scaling and load balancing
 - APIs for authenticating users and sending email using Google Accounts (SSO – Single Sign-On)
 - local development environment that simulates Google App Engine

- PaaS examples – many more



Platform as a Service (PaaS) Providers



<http://cloudcomputingwire.com>

PaaS Benefits?

- Many services are free to use
 - Pay per use - good for users, not best model for provider
 - Vendor lock in - difficult due to lack of standards
 - So why are company offering PaaS environments?
-
- Key benefit: **D A T A !**
“We’re (i.e. IT Companies) not Google’s customers; we’re Google’s product that they sell to their customers.”
-- Bruce Schneier

Linking? Infrastructure as a Service

- Real business innovation behind SaaS and PaaS is gathering data.
 - Most important asset cloud computing can provide is processing data.
 - Solution to processing vast amounts of data quickly can be found in **IaaS**
 - Infrastructure for
 - developing
 - running
 - storing
- ...applications and data in cloud environments

IaaS

- provides virtually limitless storage
- provides virtually limitless computing power
- potentially negates the need of having physical hardware at hand for doing so
- provides numerous Linux, Unix and Windows environments to work in
- provides variety of tools, services, SDKs and the like running on those OSs
- Usually requires specific knowledge to use APIs for creating and managing the virtual OSs in the cloud infrastructure
- ...therefore usually not user-friendly and simple to use

IaaS provider examples:





































IBM Cloud




CLOUDERA

“Cloud Computing” Model Abstraction

 = Managed for You I	Standalone Servers	IaaS	PaaS	SaaS
Applications				
Runtimes				
Database				
Operating System				
Virtualization				
Server				
Storage				
Networking				

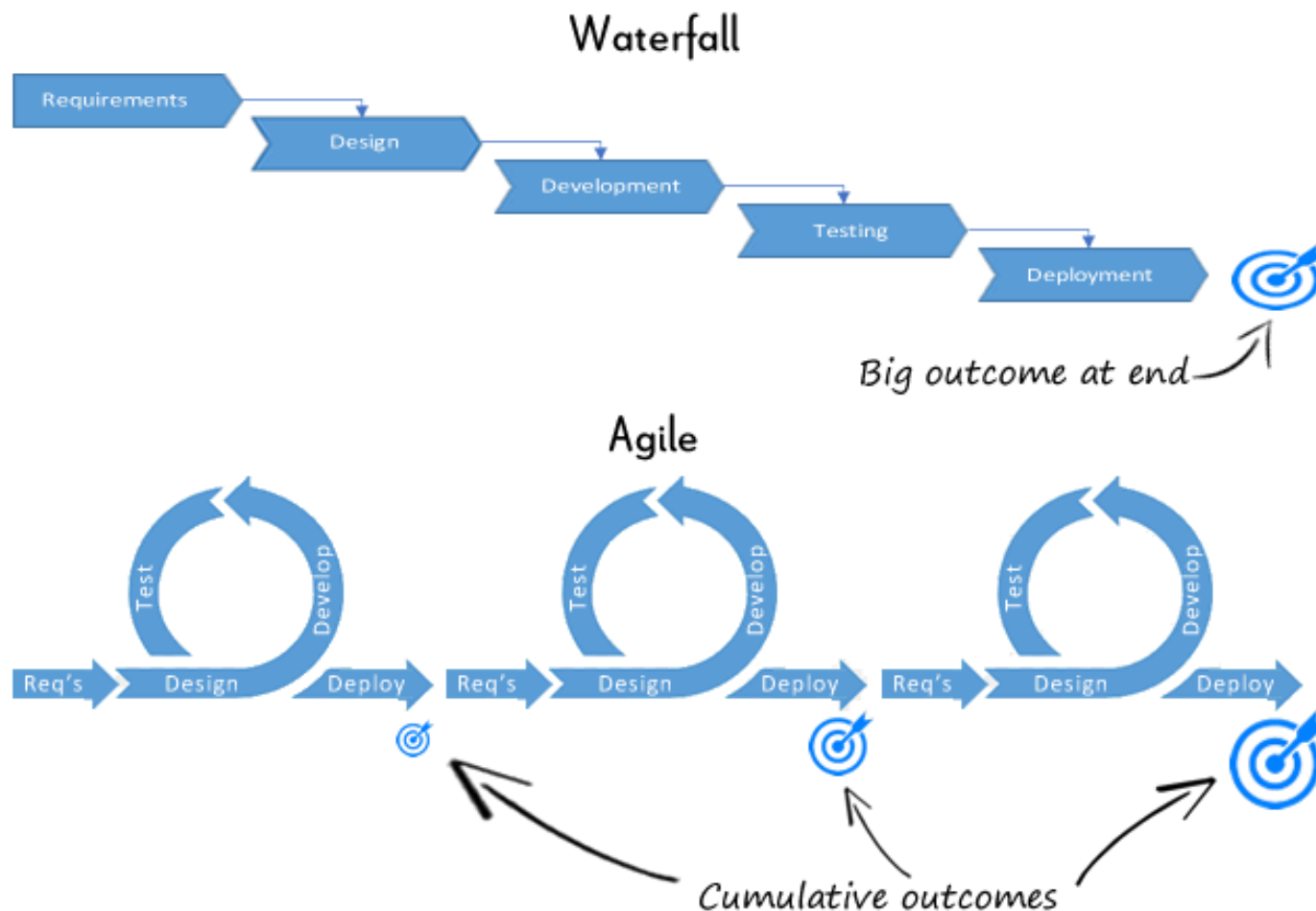
Efficiency 

 **Control**

How do we do cloud development?



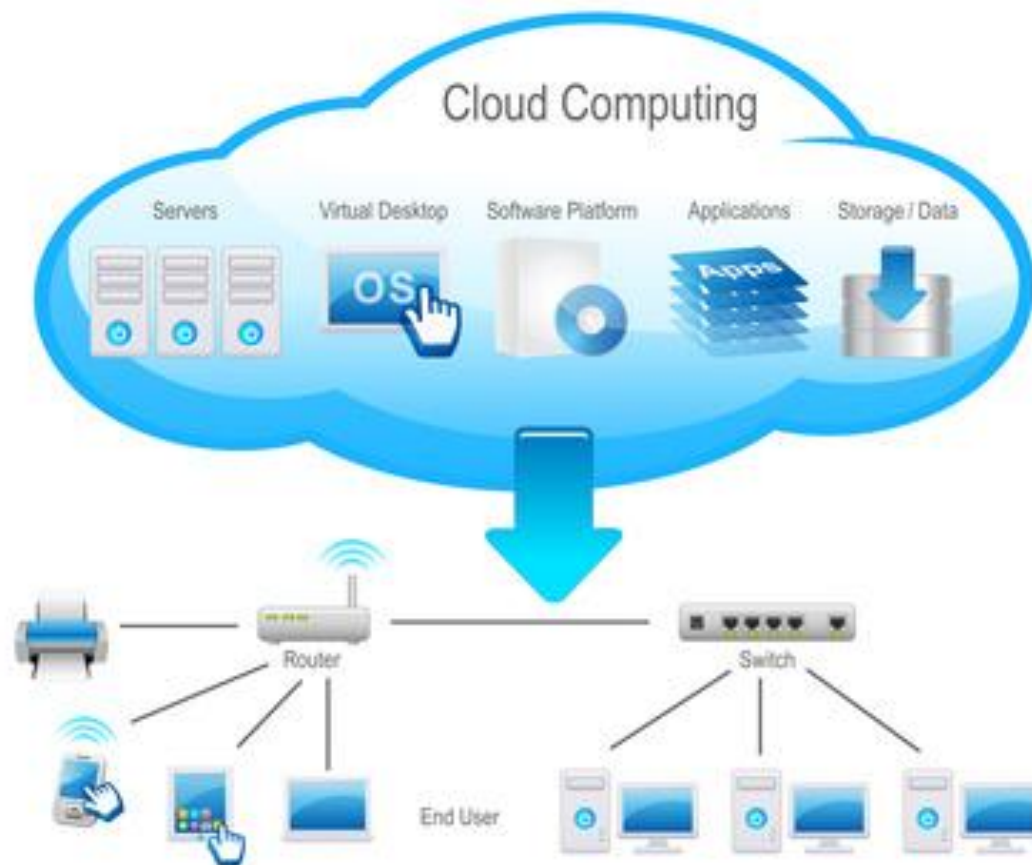
The development process remains the same...



How do we do cloud development?



But ***where*** we do these tasks may change..



How do we do cloud development?



However, **deployment** happens in the cloud.

The major players are:

#1 – Amazon



#2 – Microsoft Azure



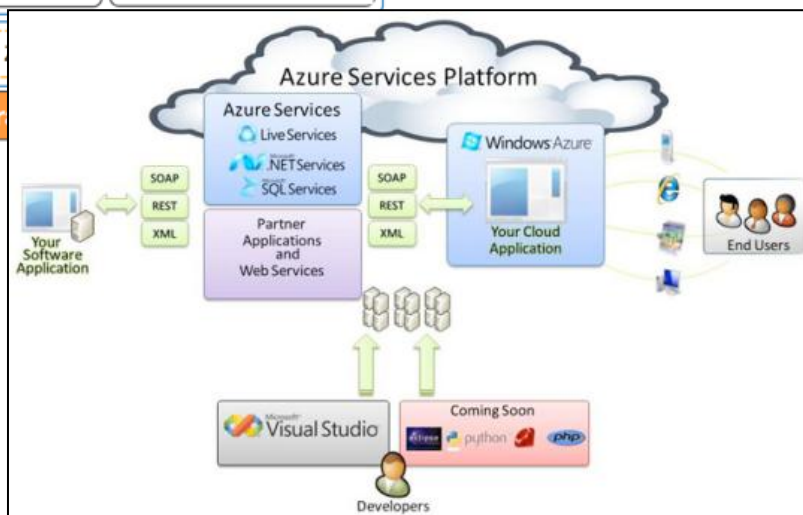
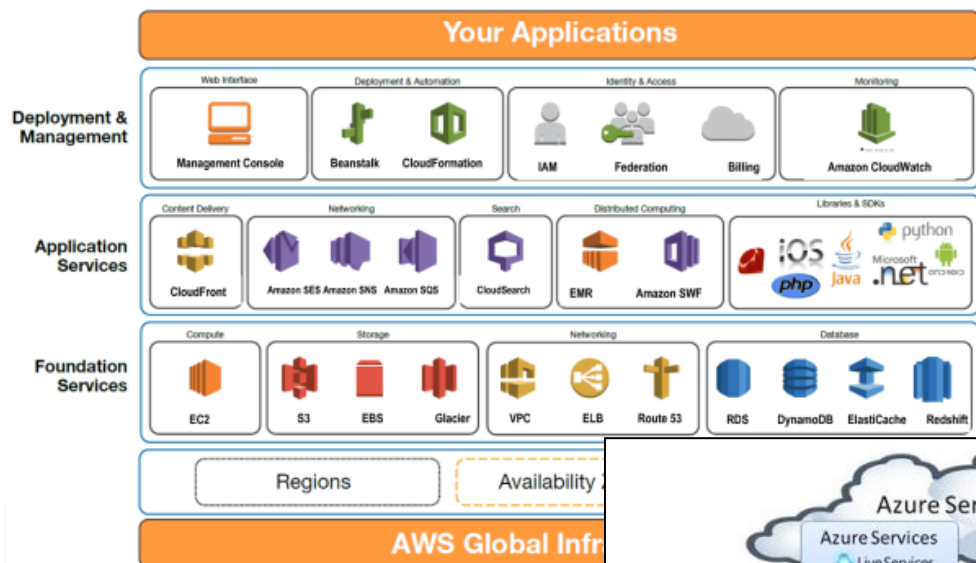
#3 – Google



How do we do cloud development?



Each player has their own set of tools on offer..



Is there any Downside?



In this Module...



You will learn to:

1. Articulate the underlying concepts of distributed computing
2. Describe the physical and virtual architecture of the cloud
3. Setup and configure both a private and public cloud system, deploy and manage the cloud services.
4. Evaluate an sample of public cloud services available from the major public cloud vendors
5. Explain the options for horizontal and vertical scaling of cloud based systems
6. Compare and contrast the economic benefits delivered by various cloud models