



# Introduction to Cloud Computing

In few words...



- « On-demand Software services over a network »
- Cloud computing is a major trend in Software.
- Cloud computing is a technical...
- ... a commercial...
- ... and an organizational revolution.

# History

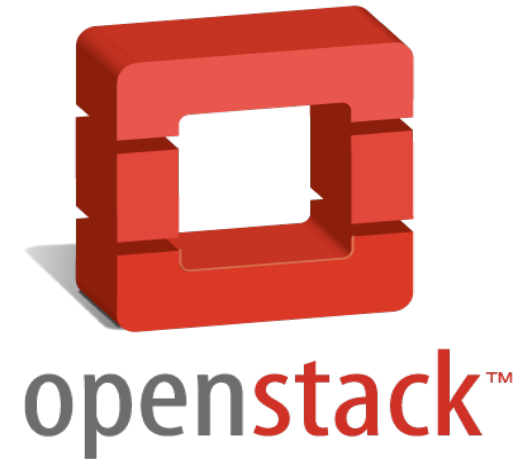


- Amazon is one of the biggest online stores
- It hosts and manages its own infrastructure
- This infrastructure is huge (several thousands of servers)
- Critical period => load picks (e.g. Christmas)
- Many servers stopped most of the time
- Idea: why not rent them? That was in the 2000's...



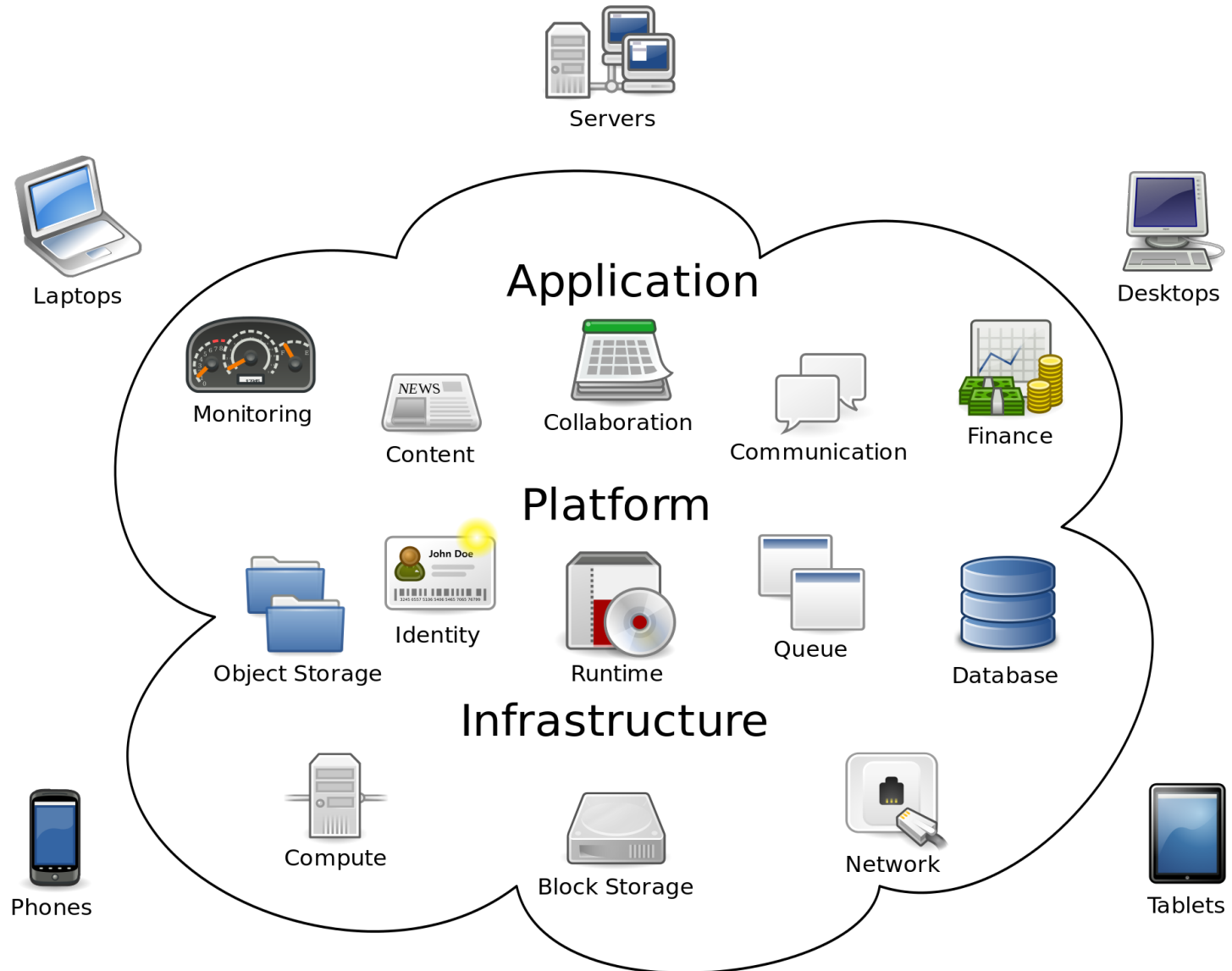
- In 2006, Amazon launched Amazon Web Services
- Virtual Software infrastructure (VMs, network, etc)
- Need an account
- « Pay for what you use »
- Manageable through a web interface
- Manageable through a REST API

# A Growing Eco-System since then...



... and went beyond just infrastructures.

# Cloud Computing

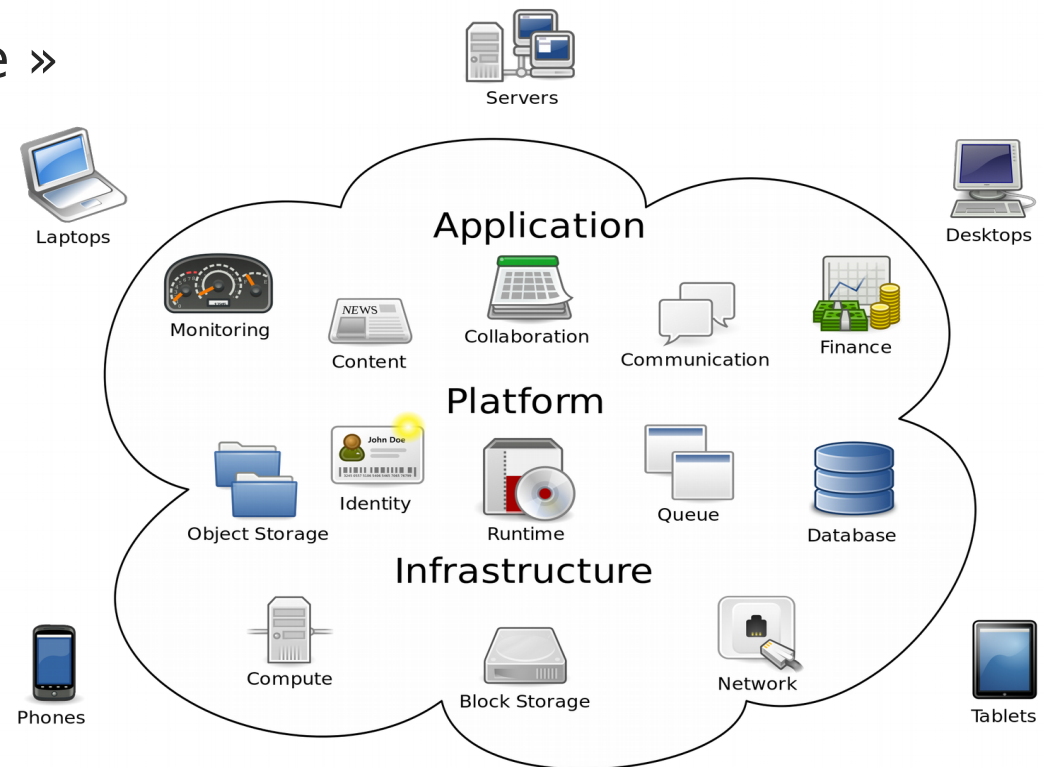


Source: wikipedia

## Cloud computing



- 3 main levels are identified
  - Infrastructure as a Service (IaaS)
  - Platform as a Service (PaaS)
  - Software as a Service (SaaS)
- Solutions made available to clients
  - « Pay for what you use »



Source: wikipedia

- Infrastructure as a Service
  - Virtualization platform (VMs)
  - Manage VMs, networks, storage...
  - Manageable through an API (e.g. REST) or a user interface
- Each platform has its own API, but...
  - Amazon Web Services' API is considered as the standard *de facto*
  - Platforms generally support AWS' API and their own one
- Attempts to create standards (OCCI Infrastructure)
  - Implemented and supported by some platforms
  - Rarely deployed in production
- Many projects provide client libraries
  - Per-IaaS ou multi-IaaS (e.g. Apache JClouds)

- Platform as a Service
  - Generally built and deployed upon a IaaS
  - Platforms to host applications and benefit from IaaS features
  - Generally also cover compilation, test and deployment
- These platforms target developers
  - REST / CLI / Web access
  - Simple integration (push on Git, commit on SVN, etc)
- Highly concurrent domain
  - Most of the PaaS are open source
  - They try to create communities (of both users and clients)
- Support for usual application patterns
  - Web application, JEE, NodeJS, etc.

- Software as a Service
  - Generally built and deployed upon a PaaS / IaaS
- These solutions target end-users
  - Example: accountancy application, chat, messaging, etc.
  - « Pay for what you use »
- Each level brings higher value for its users
  - IaaS provide infrastructure
  - PaaS provide solutions for application developers
  - SaaS designate applications (which provide solutions for users)
- XaaS
  - « Everything as a Service »
  - Extension of cloud concepts for whatever you want

- « Cloud » is also a buzz word
  - Marketing people broadcast the word everywhere
- For general publics, « cloud » means their data are hosted...
- ... somewhere over internet
  - Not saved locally
  - Available (and thus sharable) to any connected device
  - There is not mandatory a real cloud infrastructure behind
- Not really related to cloud computing as understood by companies and organizations
  - Users will not lost data
  - Way for companies to access and explore user data (business goal)

# Technical VS. Commercial

- IaaS

- Virtualization platform with an API
- It is thus possible to build self-adapting applications

- PaaS

- Macro-applications to manage development workflows
- They benefit from IaaS features (on-demand infrastructure, scalability, load picks, etc)
- Support one or many programming languages
- Support one or many programming frameworks

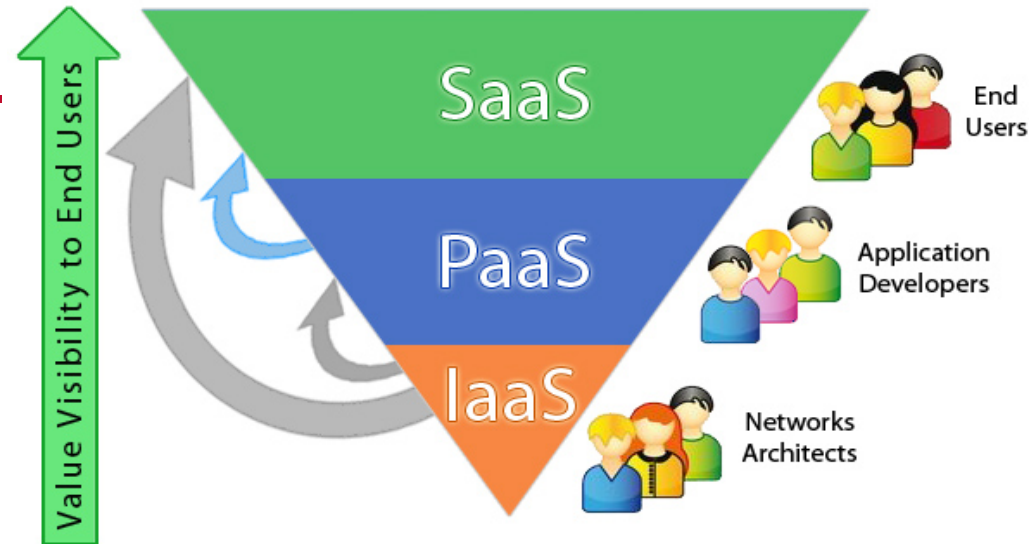
- SaaS

- They benefit from IaaS and PaaS features
- Coherent with PaaS limits
- Developers can focus on features

# Commercial Overview

- The IaaS / PaaS / SaaS trinity...

Source: qarea.com



- ... is most of all a set of business models
  - Technically, you could directly build a SaaS upon a IaaS
  - Technically, you could build a SaaS upon any infrastructure
  - Technically, you could build a PaaS upon usual servers
- IaaS = rent infrastructure
- PaaS = rent developer tools and sell one-click hosting
- SaaS = sell services to end-users



« Pay for what you use »

... and this can apply to any Software solution,  
provided you make it available through an API over a network.

# Public / Private / Hybrid Clouds

- **Public clouds designate cloud solutions available to anyone**
  - IaaS: AWS, Microsoft Azure, etc.
  - PaaS: Heroku, etc.
  - SaaS: hosted by any external service provider.
- **Private clouds designate internal cloud solutions**
  - For a company, install its own cloud solution.
  - IaaS: Openstack, Open Nebula, etc.
  - PaaS: Openshift, Cloud Foundry, etc.
  - SaaS: hosted by the company or the organization itself.
- **Hybrid clouds associate public and private clouds**

## Example: US weather forecast (from a real story)

- Main infrastructure hosted on a private cloud
  - Keep their data safe
  - Hand on the infrastructure
- Weather disaster made a part of the private cloud unavailable
  - Degraded service provided to users
  - The disaster made people eager to get weather forecast
  - Hence a high demand the infrastructure cannot face
- Applications were quickly deployed to AWS (public cloud)
  - Very fast deployment + Bigger infrastructure
  - In less than a couple of hours, the service quality was restored

# Pros and Cons

	Pros	Cons
Private Clouds	<ul style="list-style-type: none"><li>- Hands on the infrastructure</li><li>- Keep data private</li><li>- Custom configurations</li></ul>	<ul style="list-style-type: none"><li>- Complex to manage</li></ul>
Public Clouds	<ul style="list-style-type: none"><li>- Generally big infrastructures</li><li>- The hard part is managed by the hosting</li></ul>	<ul style="list-style-type: none"><li>- Data can leak (PRISM)</li><li>- Care must be taken about the cloud location</li></ul>
Hybrid Clouds	<ul style="list-style-type: none"><li>- Best of both worlds</li><li>- Generally, the public cloud is used in case of problem or pick load</li></ul>	<ul style="list-style-type: none"><li>- Requires a good design</li><li>- Requires unambiguous procedures</li></ul>

# Cloud Brokers

- Cloud brokers are proxies to cloud infrastructures
  - Since « you pay for what you use »...
  - ... it was reasonable to think one could change its cloud provider when prices were too high.
  - « Trading for cloud providers »
- Cloud brokers handle...
  - ... user accounts, permissions and quotas
  - ... runtime infrastructure (VMs, configuration, etc).
- Example
  - [ManageIQ](#) (basis of RedHat's Cloud Forms)

# Impact in Companies and Organizations



# A Major Impact

- Cloud computing is not just a technical change
- It has a major impact on business organizations
  - Classic approach: 1 project => affected machines
  - Cloud approach: share the infrastructure
- Team relations
  - No relations required between dev' projects and infrastructure teams
  - Dev' focus on their applications
  - Infrastructure people monitor and manage the servers
- Business Impact
  - No need of a huge amount of money to start a project
  - Start small and grow, rent what you need => new business players

# Example: Netflix

- Netflix is an on-demand video provider
  - Internet access
  - Watch films and TV series on any device
  - All over the world (or almost)
- Netflix infrastructure
  - Most of Netflix's infrastructure is hosted by AWS
  - A small part is hosted privately
  - Scale the infrastructure up or down depending on user connections
- Developments
  - Netflix developed quite a lot of tools for AWS
  - <https://github.com/netflix>

# Thanks for your attention

**LINAGORA - headquarters**

80, rue Roque de Fillol  
92800 PUTEAUX  
FRANCE

**Phone :** +33 (0)1 46 96 63 63

**Info :** [info@linagora.com](mailto:info@linagora.com)

**Web :** [www.linagora.com](http://www.linagora.com)

 **@linagora**

 **facebook.com/Linagora/**

