

Platform as a Service (PaaS)

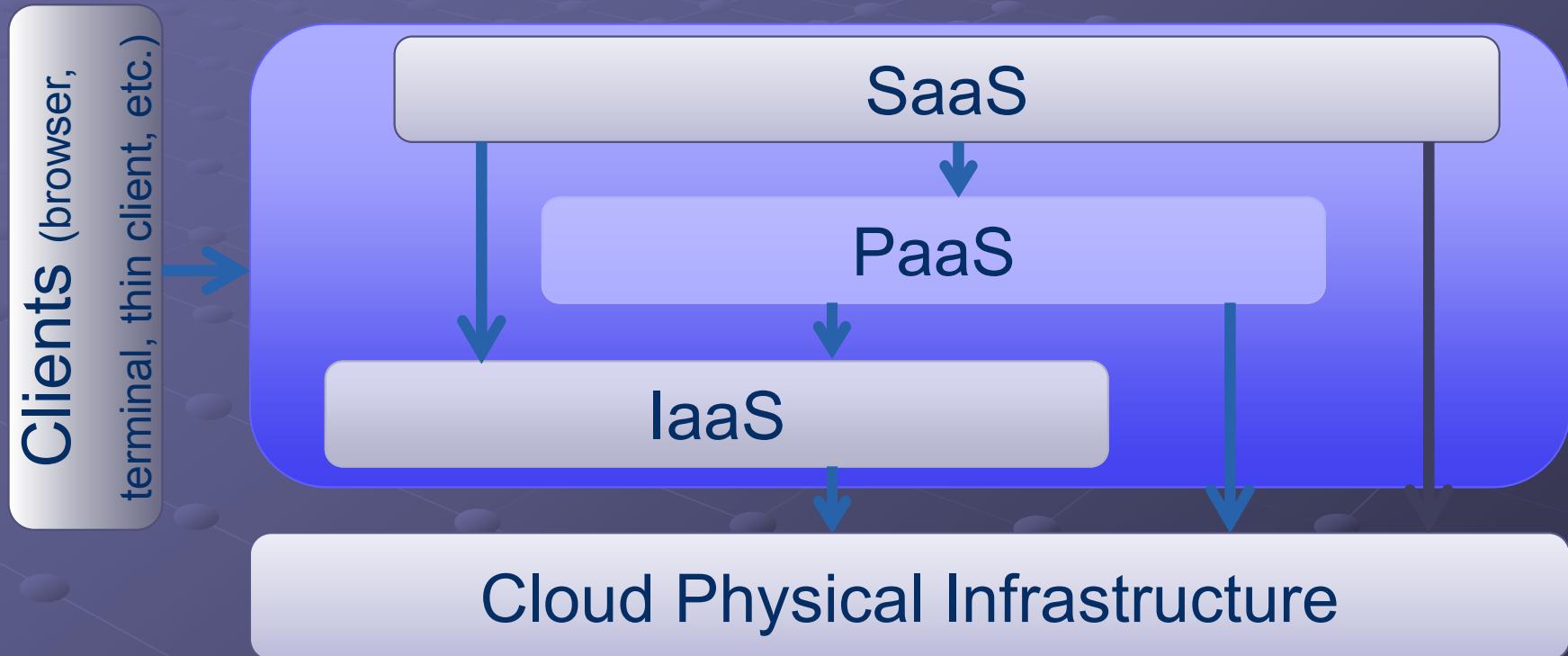
Prof. Tahar Kechadi

School of Computer Science & Informatics

Outline

- Define PaaS
- Pros and Cons
- Case studies

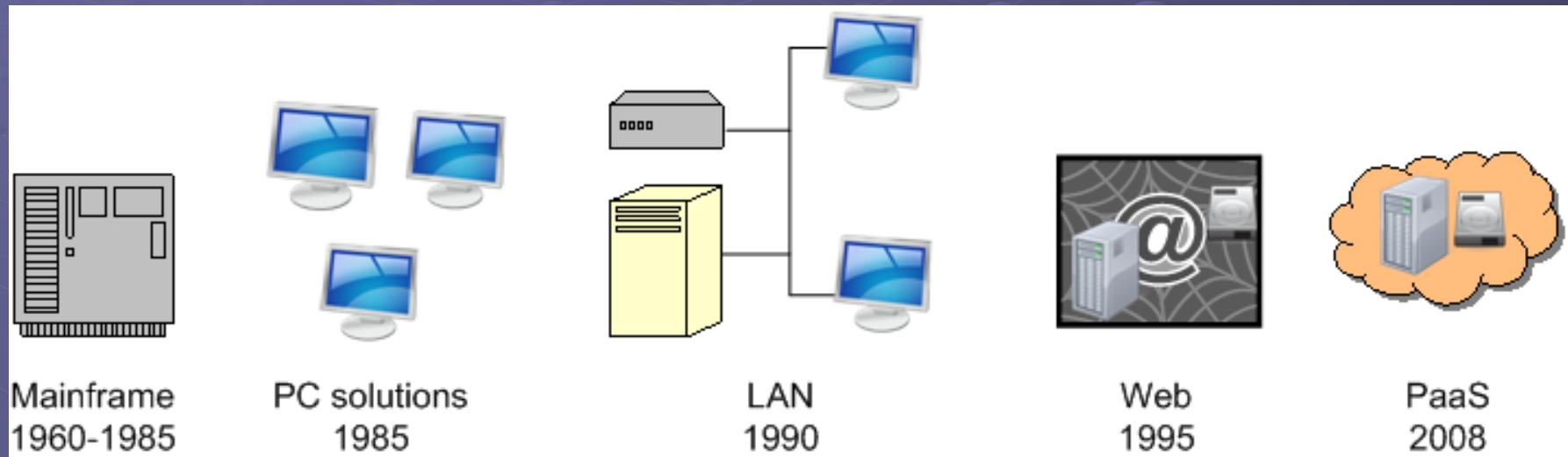
Keep the hierarchy in mind!



Introduction

- Built on the top of IaaS, Platform as a Service (PaaS) is another application delivery model. PaaS supplies all the resources required to build & deploy applications and services
- PaaS services include application design, development, testing, deployment, and hosting
- Cloud Computing Platform

Evolution of technology



From ISP to PaaS

Internet Services Providers

- Maintained webservers and high-speed, high-bandwidth connections
- Reduced cost
- Less: server administration, hardware to purchase and maintain
- Greater system uptime
- Potential scalability

Used Windows-, Linux-based web servers, laid the groundwork for the eventual creation-> cloud-based PaaS solutions

Internet Service Providers

- First ISP: Australia, 1989
- Services
 - Access (internet access, email box)
 - UPC, Eircom, ...
 - Hosting (web, email, storage, VM, servers)
 - Blacknight, Justhost, ...
 - Evolved into and influenced by PaaS

Hosting ISP today

- Shared hosting (pre-PaaS)
 - Basic (web, email, storage), cheap
- Hosted applications (SaaS)
 - MS Exchange, ...
- Hosted application frameworks (PaaS)
 - MS SharePoint, GoMobi, ...
- Virtual machines (IaaS)
 - Price depends on the amount of resources
- Dedicated servers (IaaS)

ISP: shared hosting

- ➊ Shared instance of OS/Web/DB
 - Load and security issues
- ➋ Control panel (cPanel, Plesk, ...) integrated components, often open-source
 - Web server user settings (Apache, IIS, ...)
 - Storage + web file browsers + FTP
 - Email + webmail (Roundcube, ...)
 - RDBMS(MySQL, ...) + web/phpMyAdmin, ...)
 - Schedulers (Crontab, ...)
 - Backup

ISP: shared hosting

Screen ID: 2.21.20.01 [Refresh](#)

Service Information

Subscription: Linux Shared Minimus

Diskspace: 275.3 MB used of 10 GB

Traffic: 102.4 MB used of 200 GB

MySQL5 Databases: 4 unit used of 30 unit

[All Resource Usage](#) [Subscription Resources](#)

Mail

This section provides you with managing facilities for your Qmail e-mails and maillists.

[Add New E-mail Address](#) [E-mail addresses](#)

Websites

Set up websites. Manage website content and view statistics on how people use your websites.

[Manage Website](#) [File Manager](#)
[Web Applications](#) [Add New Subdomain](#)

[More →](#)

More Services

This is where you can manage special features of your current subscription.

[AWStats Web Statistics](#) [Databases](#)
[Backups](#)

[More →](#)

Account

Outstanding Invoices: 0.00 EUR

- [Financial Documents](#)
- [Admins](#)
- [Subscriptions](#)

[More →](#)

Users

Create and manage users. Assign services to created users.

- [Users](#)
- [Create User](#)

Store

Here you can buy additional services, domains and more.

- [Buy More Services](#)
- [Buy Additional Resources](#)
- [Register New Domain](#)
- [Change Service Plan](#)

Domains

Manage your domains and domain names.

- [Registered Domains](#)
- [Hosted Domains](#)

ISP: Service-Level Agreement

- Abandonment Rate
 - Percentage of calls abandoned while waiting to be answered.
- Average Speed to Answer
 - Average time it takes for a call to be answered by the service desk.
- Time Service Factor
 - Percentage of calls answered within a definite timeframe.
- First-Call Resolution
 - Percentage of incoming calls that can be resolved without the use of a callback or without having the caller call back the helpdesk to finish resolving the case.
- Turn-Around Time
 - Time taken to complete a certain task.
- Mean Time To Recover
 - Time taken to recover after an outage of service.

PaaS

The PaaS model can support add-ons to SaaS applications, stand-alone environments for general development, and application delivery-only environments, supporting hosting.

Examples include:

- ❑ Google App Engine
- ❑ Microsoft's Azure
- ❑ VMware's Cloud Foundry
- ❑ And many others...

PaaS: abstraction from IaaS

- The Infrastructure layer provides users with direct access to the underlying infrastructure
- Isolate users from the resource interaction to the lower levels of resource interaction
- Allow developers to create new software that is not susceptible to the number of provisioned machines or their network configuration

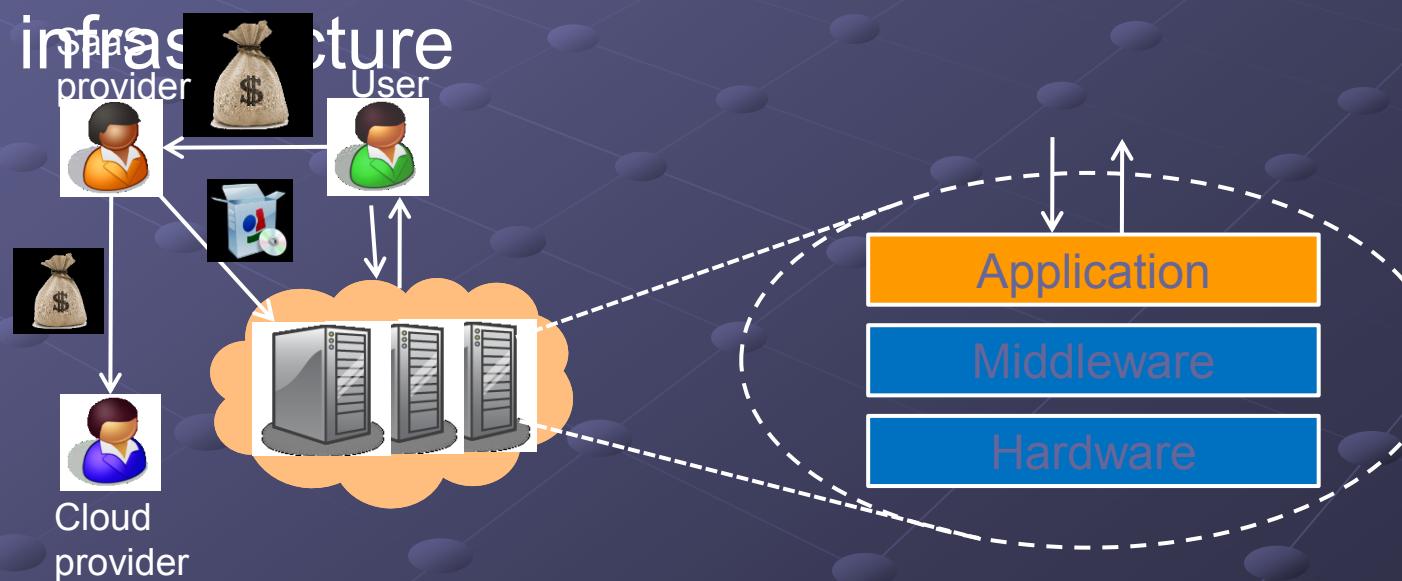
PaaS: API to support SaaS

- PaaS allows developers to build new software that takes advantage of the available resources.
- PaaS solution is usually designed with a set of APIs that directly influence the programs that can be built on the Cloud.
- PaaS solutions are deeply tied to Cloud vendors.

PaaS and SaaS

Cloud provides middleware/infrastructure

- For example, Microsoft Common Language Runtime (CLR)
- Customer pays SaaS provider for the service; SaaS provider pays the cloud for the infrastructure



Pros of PaaS

- Lower total cost of ownership
- Lower administrative overhead
- More current system software
- Increased business and IT alignment
- Scalable solutions

PaaS Benefits for Developers

- Focus only on innovation that provide real business value instead of infrastructure setup
- Zero infrastructure
- Lower Risk
- Lower cost
- Easy and quick development
- Reusable code
- ...

Cons of PaaS Solutions

- Concern about data security
- Challenges to integrating cloud solutions with legacy software
- Risk of breach by PaaS provider

Example: Google App Engine (GAE)

- Let developers create and host web-based applications that reside and run on services managed by Google
- GAE features:
 - Support for dynamic web pages
 - Data storage and query support
 - Load balancing for application scalability
 - API
 - SDK
 - Administrative console for managing applications and databases

Summary

- Describe PaaS
- Pros and Cons
- Example