

CLOUD COMPUTING | DIGITAL TALENT TRAINING KOMINFO

MODEL LAYANAN KOMPUTASI AWAN

Ardhi Putra Pratama Hartono

Pusat Ilmu Komputer Universitas Indonesia

PUSILKOM UI

15 Oktober 2018 – 5 Desember 2018

Quality Result by Bridging Theories and Best Practices



AGENDA | Model Layanan

1 REVIEW CLOUD ARCHITECTURE

2 CLOUD SERVICE MODEL

2 SAAS

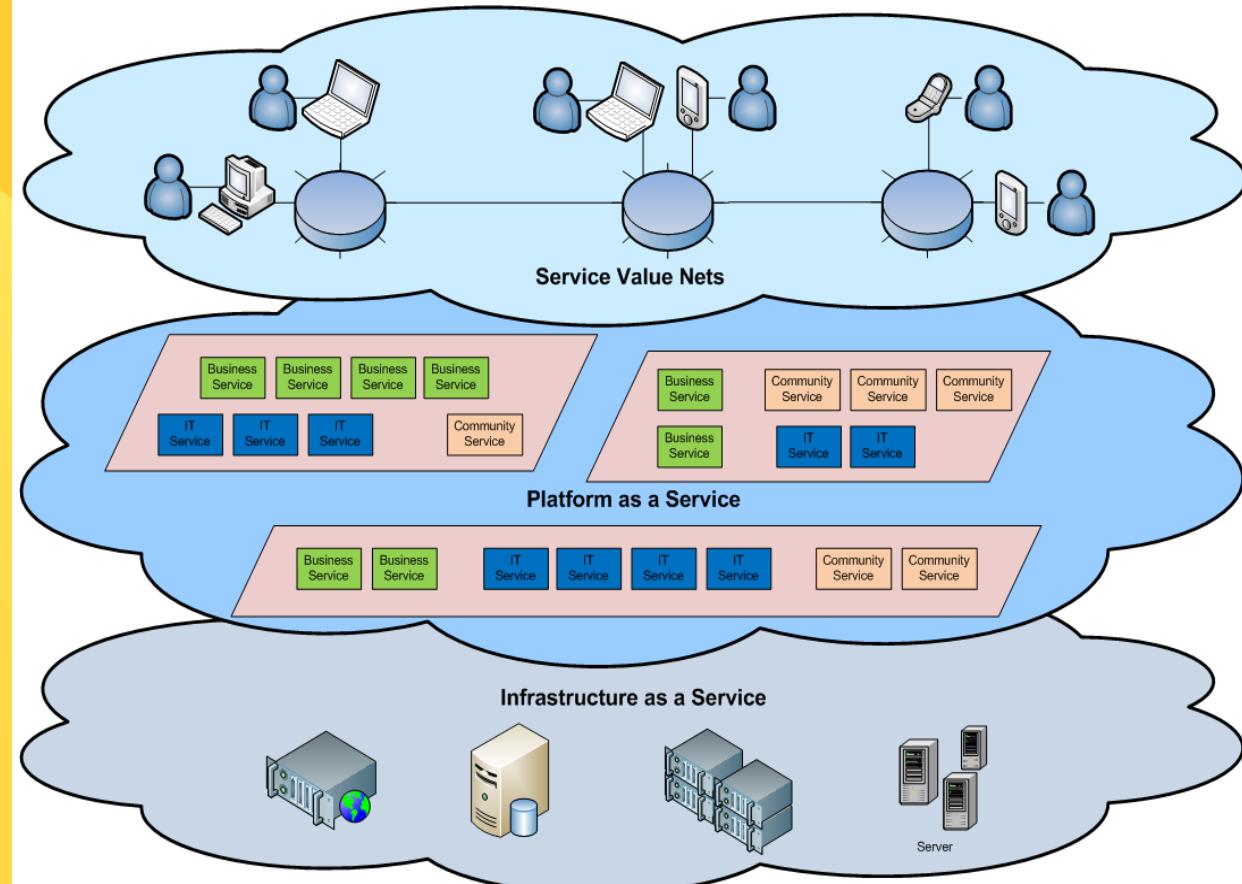
4 IAAS

5 PAAS

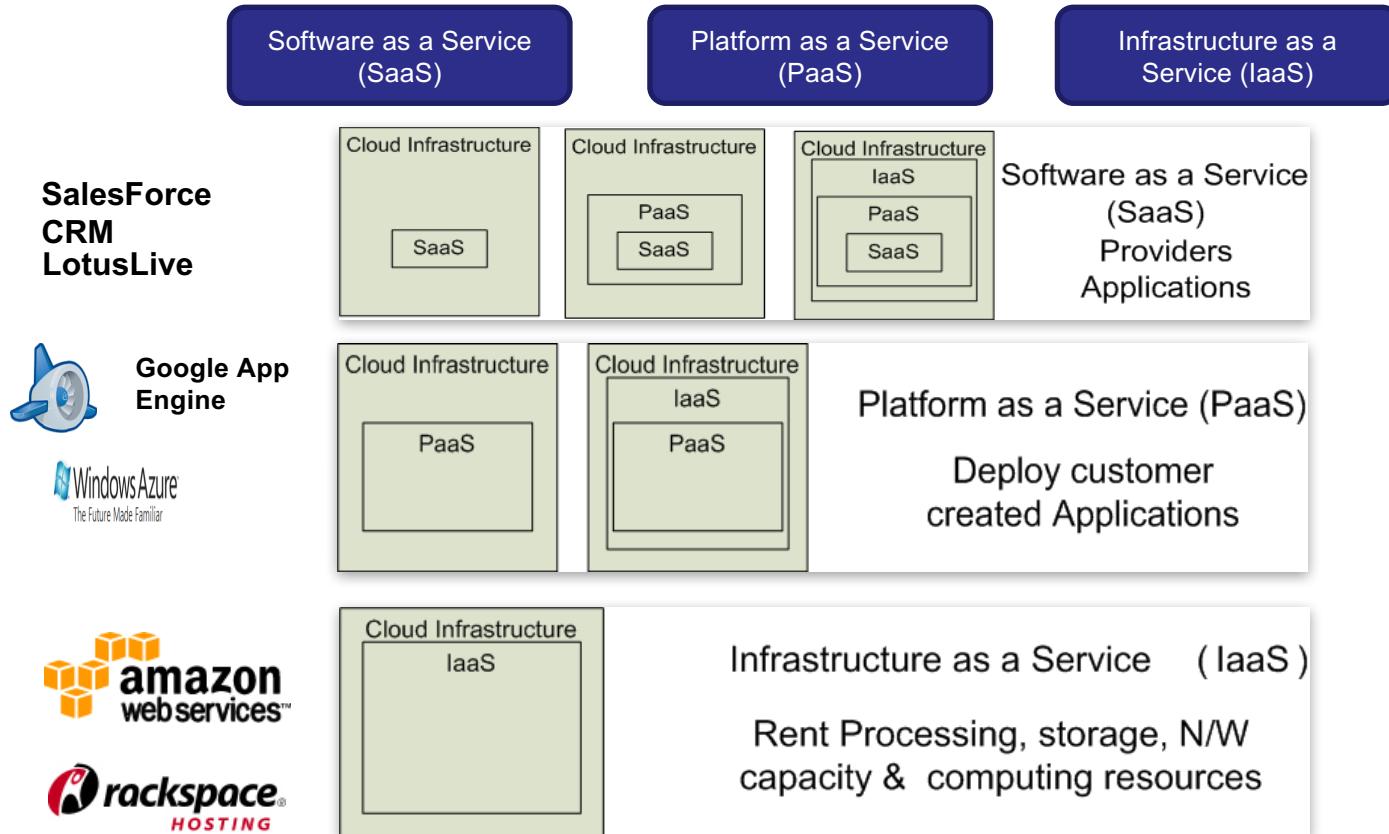
6 DEPLOYMENT MODELS



CLOUD ARCHITECTURE



CLOUD SERVICE MODEL





WHAT IS SOFTWARE AS A SERVICE (SAAS)?

“

SaaS is a software delivery methodology that provides licensed multi-tenant access to software and its functions remotely as a Web-based service.

-
- ! Usually billed based on usage
 - ! Usually multi tenant environment
 - ! Highly scalable architecture

INTRODUCTION TO SAAS

- Software as a service (SaaS) is a model of **SOFTWARE DELIVERY** where the software company provides maintenance, daily technical operation, and support for the software provided to their client.
- It assumes the software is delivered over the **internet**.
- Software delivered to home consumers, small business, medium and large business
- The web as a **platform** is the center point
- Web-browser acting as a thin-client for accessing the software **remotely across** the internet.
- Network-based access to, and management of, commercially available (i.e., not custom) **software application delivery that typically is closer to a one-to-many model** (single instance, multi-tenant architecture) than to a one-to-one model, including architecture, pricing, partnering, and management characteristics



PROS

1. Stay **focused** on business processes
2. Change software to an **Operating Expense** instead of a Capital Purchase, making better accounting and budgeting sense.
3. Create **a consistent application** environment for all users
4. No concerns for **cross platform** support
5. **Easy** Access
6. **Reduced piracy** of your software
7. **Lower Cost**
(For an affordable monthly subscription, Implementation fees are significantly lower)
8. Continuous Technology Enhancements



CONS

1. Initial time needed for **licensing and agreements**
(Trust, or the lack thereof, is the number one factor blocking the adoption of software as a service (SaaS)).
2. **Absence** of disconnected use
Centralized control
Possible erosion of customer privacy



SAAS ARCHITECTURE

| FUELED BY

1

Bandwith Technologies

2

The cost of a PC has been **reduced** significantly with more powerful computing but the cost of application software has not followed

2

Licensing **issues** for business are contributing significantly to the use of illegal software and piracy.

4

Timely and expensive **setup** and **maintenance** costs

TRADITIONAL PACKAGED SOFTWARE

- 1 Designed for customers to install, manage and maintain
- 2 Architect solutions to be run by an individual company in a dedicated instantiation of the software
- 3 Infrequent, major upgrades every 18-24 months, sold individually to each installed base customer.
- 4 Version control & Upgrade fee
- 5 Streamlined, repeatable functionality via Web services, open APIs and standard connectors

COMPARISON OF BUSINESS MODEL

SOFTWARE AS A SERVICE

- 1 Designed from the outset up for delivery as Internet-based services
- 2 Designed to run thousands of different customers on a single code
- 3 Frequent, "digestible" upgrades every 3-6 months to minimize customer disruption and enhance satisfaction.
- 4 Fixing a problem for one customer fixes it for everyone
- 5 May use open APIs and Web services to facilitate integration, but each customer must typically pay for one-off integration work.



WHAT IS INFRASTRUCTURE AS A SERVICE (IaaS)

“
IaaS is the delivery of technology infrastructure as an on demand scalable service

- ! Usually billed based on usage
- ! Usually multi tenant virtualized environment
- ! Can be coupled with Managed Services for OS and application support



PROBLEMS IN CONVENTIONAL CASE

1. Companies IT investment for peak capacity
 2. Lack of agility for IT infrastructure
 3. IT maintain cost for every company
- Usually suffered from hardware failure risk



These IT Complexities force company back!

But, who can be this “**somebody**”, and provide all these services ?

HOWTO SOLVE?

Let's consider some kind **of out-sourcing solution**

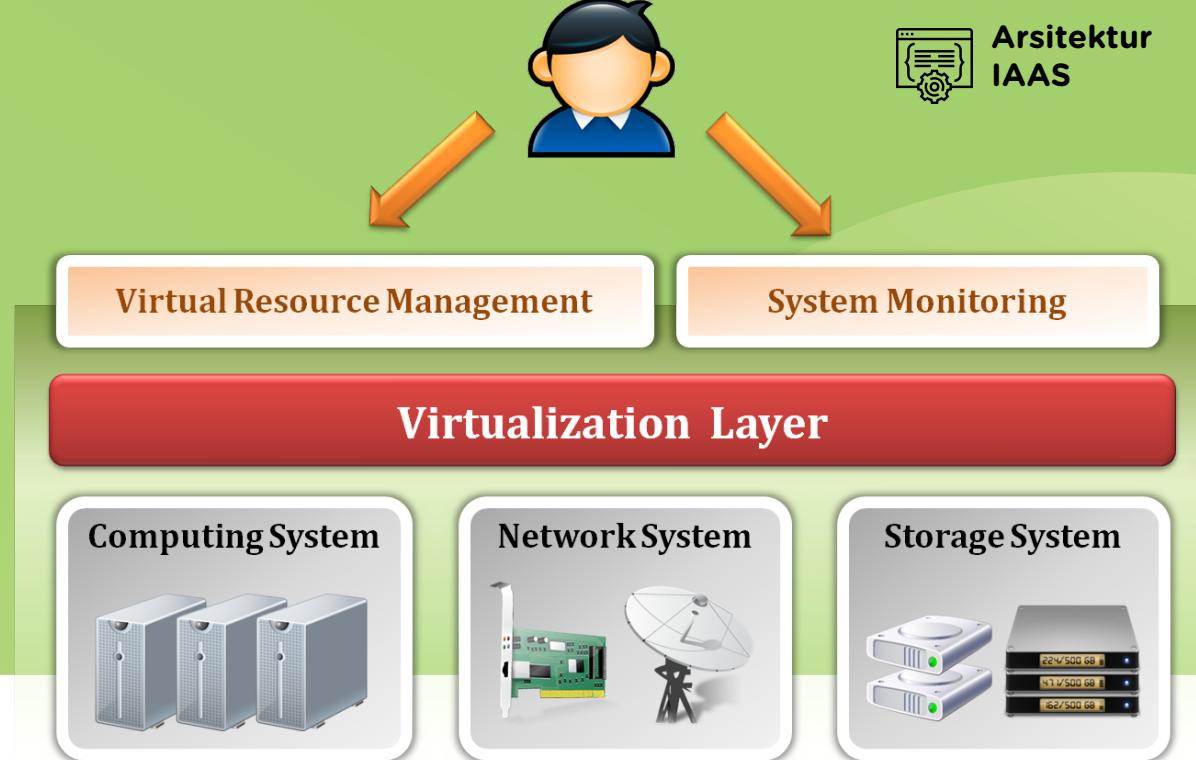
Somebody will handle on demand capacity for me

Somebody will handle high available resource for me

Somebody will handle hardware management for me

Somebody will handle system performance for me

IaaS Architecture



- **Infrastructure as a Service (IaaS)**
delivers computer infrastructure for cloud user, typically a platform virtualization environment as a services

- **Virtualization**
is an enabling technique to provide an abstraction of logical resources away from underlying physical resources.

SCALABILITY & ELASTICITY

in IAAS Dynamic Provision | Multi-Tenant Design

- Clients should be able to **dynamically increase or decrease** the amount of infrastructure resources in need.
- Large amount of resources provisioning and deployment should be **done in a short period of time**, such as several hours or days.
- System behavior **should remain identical** in small scale or large one

AVAILABILITY & RELIABILITY

in IAAS Fault Tolerance | System Resilience | System Security

- **Clients should be able to access** computation resources without considering the possibility of hardware failure.
- **Data stored in IaaS cloud** should be able to be retrieved when needed without considering any natural disaster damage.
- Communication **capability and capacity should be maintained** without considering any physical equipment shortage.

MANAGEABILITY & INTEROPERABILITY

in IAAS Control Automation | System Monitoring | Billing System

- Clients should be able **to fully control the virtualized** infrastructure resources which allocated to them.
- Virtualized resources can be **allocated by means of system control automation process** with pre-configured policy.
- States of all virtualized resource should be fully under **monitoring**.
- Usage of **infrastructure resources will be recorded** and then billing system will convert these information to user payment

PERFORMANCE & OPTIMIZATION

in IAAS Parallel Processing | Load Balancing | Job Scheduling

- Physical resources should be **highly utilized** among different clients.
- Physical resources should form a large resource pool which **provide high computing power** through parallel processing.
- Virtual infrastructure resources will be **dynamically configured** to an optimized deployment among physical resources.

ACCESSIBILITY & PROTABILITY

in IAAS Uniform Access | Thin Client

- Clients should be able **to control, manage and access infrastructure resources** in an easy way, such as the web-browser, without additional local software or hardware installation.
- Provided infrastructure resources should be able to be **reallocating or duplicated easily**.



WHAT IS PLATFORM AS A SERVICE (PaaS)

“

PaaS provides all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely from the Internet.

- !** Typically applications must be developed with a particular platform in mind
- !** Multi tenant environments
- !** Highly scalable multi tier architecture

“ PaaS PROPERTIES



PENGERTIAN PAAS

- Gives the programmer a **solution stack**
(Web server, database engine, scripting language)
- Simple deployment**, no worries about servers, storage, network, scaling, updates, ...
- Guarantees multitenancy for better security
- Users isolated by virtualization or OS means
- Accounting and billing of used resources
(Different at every vendor)
- Development tools

“ COMPARISON WITH IAAS



PENGERTIAN PAAS

- IaaS **better for migrating** existing applications
(More flexible, you install your environment)
- PaaS has **lower demands** on administration
- PaaS will take care of scaling if applications use correct frameworks, also redundancy and CDN
- > PaaS better for new applications
- BUT has dangers of vendor lock in if platform specific functions are used
(IaaS instance can be copied to your server)



“

COMPARISON WITH WEBHOSTING

- Webhosting essentially does the same - offers a platform for web sites / applications
- Minus scalability, multitenancy, accounting
- Plus personal contact - negotiation, support
- Different languages, cloud focuses on scalability

(Hosting: PHP, ASP, some Perl and Python

Cloud: Java, Ruby, PHP (due to demand), Node.js)

- Added value - e-mail and domain hosting
(vs. development tools and web services in PaaS)



“

PaaS TYPES

INSTANCES PaaS

Depends on IaaS layer for multitenancy
(Better security and performance guarantees)

Deploys applications to IaaS instances

Ex : Amazon Elastic Beanstalk | Microsoft Azure

FRAMEWORK PaaS

Uses OS capabilities for multitenancy
(Better resource utilization and accounting granularity)

Requires specific frameworks to be used

Can benefit from cloud infrastructure, but is not dependent on it

Ex : Google App Engine | VMware Cloud Foundry

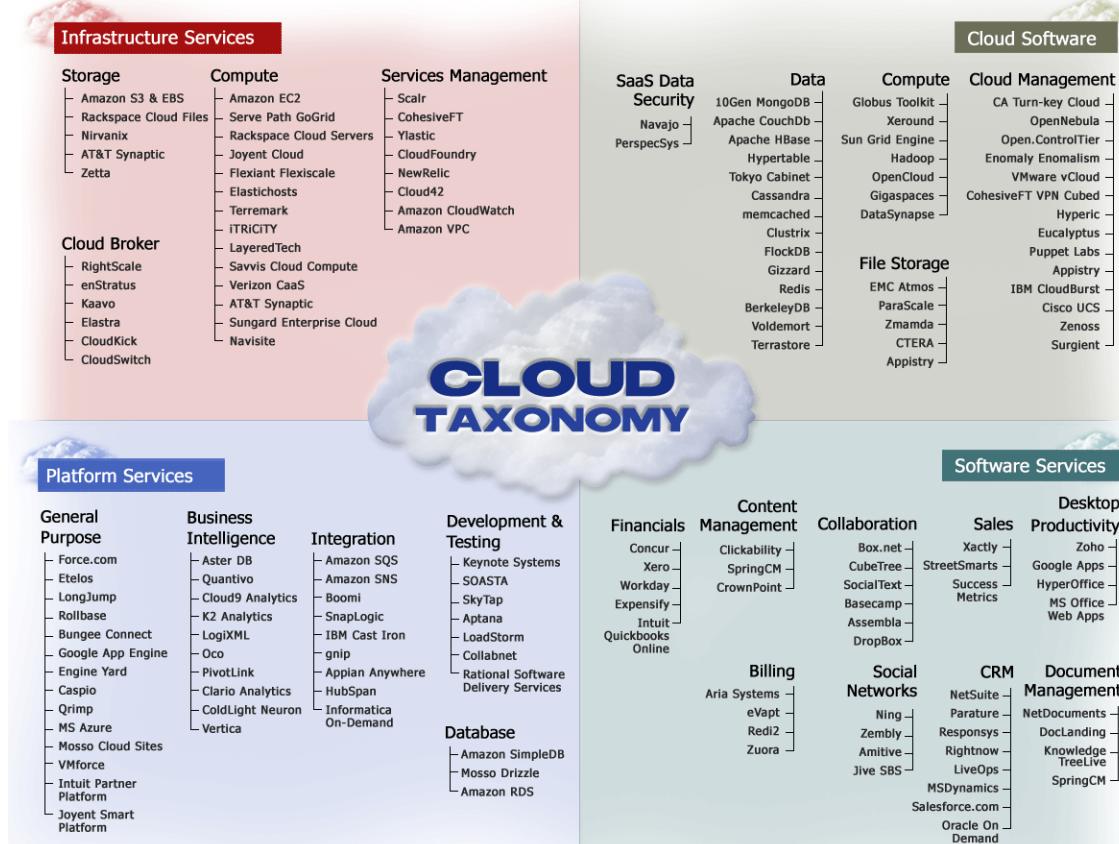
METADATA PaaS

Client configures his service through metadata

CLOUD SERVICE MODEL

Services	Description
Services	Services – Complete business services such as PayPal, OpenID, OAuth, Google Maps, Alexa
Application	Application – Cloud based software that eliminates the need for local installation such as Google Apps, Microsoft Online
Development	Development – Software development platforms used to build custom cloud based applications (PaaS & SaaS) such as SalesForce
Platform	Platform – Cloud based platforms, typically provided using virtualization, such as Amazon ECC, Sun Grid
Storage	Storage – Data storage or cloud based NAS such as CTERA, iDisk, CloudNAS
Hosting	Hosting – Physical data centers such as those run by IBM, HP, NaviSite, etc.

CLOUD TAXONOMY





DEPLOYMENT MODELS

2

PUBLIC CLOUD

Public cloud (off-site and remote) describes cloud computing where resources are dynamically provisioned on an on-demand, self-service basis over the Internet, via web applications/web services, open API, from a third-party provider who bills on a utility computing basis.

4

COMMUNITY CLOUD

A community cloud is formed when several organizations with similar requirements share common infrastructure. Costs are spread over fewer users than a public cloud but more than a single tenant

1

PRIVATE CLOUD

A *private cloud* environment is often the first step for a corporation prior to adopting a public cloud initiative. Corporations have discovered the benefits of consolidating shared services on virtualized hardware deployed from a primary datacenter to serve local and remote users.

3

HYBRID CLOUD

A *hybrid cloud* environment consists of some portion of computing resources on-site (*on premise*) and off-site (*public cloud*). By integrating public cloud services, users can leverage cloud solutions for specific functions that are too costly to maintain on-premise such as virtual server disaster recovery, backups and test/development environments.

“

CLOUD STORAGE



-Large Web companies are exploiting the fact that they have data storage capacity that can be hired out to others.

(allows data stored remotely to be temporarily cached on desktop computers, mobile phones or other Internet-linked device).

-Amazon's Elastic Compute Cloud (EC2) and Simple Storage Solution (S3) are well known examples

(Mechanical Turk)





“

UTILITY COMPUTING EC 2

Amazon Elastic Compute Cloud (EC2):

Elastic, marshal 1 to 100+ PCs via WS,
Machine Specs...,
Fairly cheap!

Powered by Xen - a Virtual Machine:

Different from Vmware and VPC as uses
“para-virtualization” where the guest OS is
modified to use special hyper-calls:

Hardware contributions by Intel (VT-
x/Vanderpool) and AMD (AMD-V).

Supports “Live Migration” of a virtual
machine between hosts.

**Linux, Windows, OpenSolaris
Management Console/AP**



“

EC 2 - THE BASICS

- Load your image onto S3 and register it.
- Boot your image from the Web Service.
- Open up required ports for your image.
- Connect to your image through SSH.
- Execute you application...



“

WINDOWS AZURE

- Enterprise-level on-demand capacity builder
- Fabric of cycles and storage available on-request for a cost
- You have to use Azure API to work with the infrastructure offered by Microsoft
- Significant features: web role, worker role , blob storage, table and drive-storage



“

GOOGLE APP ENGINE

-This is more a web interface for a development environment that offers a one stop facility for design, development and deployment Java and Python-based applications in Java, Go and Python.

-Google offers the same reliability, availability and scalability at par with Google's own applications

-Comprehensive programming platform irrespective of the size (small or large)

-Signature features: templates and.appspot, excellent monitoring and management console

TERIMAKASIH

Content By : Ardhi Putra Pratama Hartono

Design By : Annisaa Fitri Shabrina

Pusat Ilmu Komputer Universitas Indonesia

PUSILKOM UI

DIGITAL TALENT TRAINING KOMINFO

15 Oktober 2018 – 5 Desember 2018

REFERENSI:

<https://www.cse.unr.edu/~mgunes/cpe401/cpe401sp15/10-cloud.ppt>

www.uoguelph.ca/~qmahmoud/teaching/soc/students/writing/BrianMoore.ppt

<http://www.cs.nthu.edu.tw/~ychung/slides/Cloud%20Computing/Lecture%203%20-%20Introduction%20to%20IaaS.pptx>

https://cw.fel.cvut.cz/old/_media/courses/a4m39wa2/lectures/20/paas.ppt