



Platform as a Service (PaaS)

Roch Glitho, PhD

Professor and Canada Research Chair

My URL - <http://users.encs.concordia.ca/~glitho/>



Platform as a Service (PaaS)

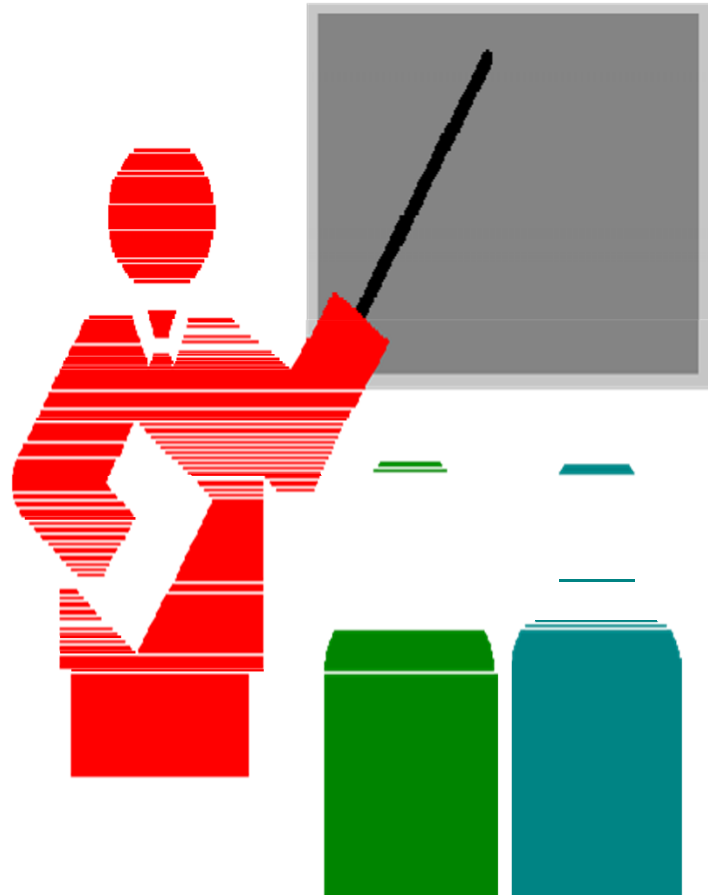


References

1. L.M. Vaquero et al., A Break in the Clouds: Towards a Cloud Definition, ACM SIGCOM Computer Review, January 2009
- 2.Q. Zhang et al., Cloud Computing: State-of-the-Art and Research Challenges, Journal of Internet Services Applications (2010)
- 3.C. Vechiola et al., Aneka – Integration of Private and Public Cloud, Chapter 9, in “Cloud Computing: Principles and Paradigms”, (eds: R. Buyya et al.) Wiley 2011



PaaS



- What is PaaS?
- Case studies
 - Aneka
 - Cloud Foundry



What is a PaaS?

- Platforms used for the development and management of the applications offered as SaaS to end-users (and other applications)
 - Examples:
 - Google Apps Engine
 - Microsoft Azur
 - Cloud Foundry
-

What is a PaaS?

- Platforms used for the development and management of the applications offered as SaaS to end-users (and other applications)
 - Examples:
 - Microsoft Azure (Microsoft IaaS)
 - Cloud Foundry (OpenStack, AWS, vSphere)
 - Heroku (runs on AWS)
 - Digital Ocean (Your own IaaS)
-

What is a PaaS?

``Cloud systems can offer an additional abstraction level: instead of supplying a virtualized infrastructure, they can provide the software platform where systems run on. The sizing of the hardware resources demanded by the execution of the services is made in a transparent manner. This is denoted as Platform as a Service (PaaS)``

Reference 1.

What is a PaaS?

``The platform layer: Built on top of the infrastructure layer, the platform layer consists of operating systems and application frameworks. The purpose of the platform layer is to minimize the burden of deploying applications directly into VM containers. For example, Google App Engine operates at the platform layer to provide API support for implementing storage, database and business logic of typical web applications``

Reference 2.

What is a PaaS?

“They provide enterprises with a platform for creating, deploying and managing distributed applications on top of existing cloud infrastructures. They are in charge of monitoring and managing the infrastructure and acquiring new nodes and they rely on virtualization technologies in order to scale applications on demand”

Reference 3.



What is a PaaS?

PaaS handle application / service life cycle

- 4 phases in the early service life cycle models

1. Phase 1: Development –

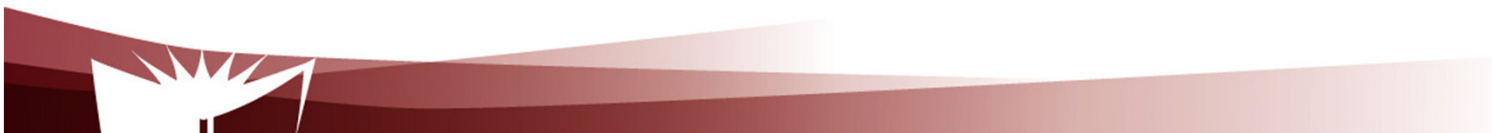
- Includes design, testing ...

2. Phase 2: Deployment

3. Phase 3: Usage

- Includes activation, execution ..

4. Phase 4: Removal



What is a PaaS?

On application / service life cycle

The 4 phases are sometimes collapsed in two phases:

1. Phase 1: Development –

- Includes design, testing ...
-

2. Phase 2: Management (i.e. everything that is not development)

- Deployment
- Usage
- Removal



What is a PaaS?

A PaaS might re-use existing application / service life cycle frameworks / tools for some of the phases:

- Microsoft Azure re-uses .NET for development phase
 - Cloud Foundry does not come with frameworks for development phase (The user can choose)
-



What is a PaaS?

A PaaS might be bound to a given IaaS or allow the user to select within a pre-defined set:

- Google Apps Engine comes with Google IaaS
 - The infrastructure can be bypassed to some extent
- Cloud Foundry allows the user to select within a pre-defined set (e.g. Openstack, Amazon WS)



What is a PaaS?

A PaaS might allow or not allow auto-scaling:

- Horizontal and/or
- Vertical

Note: Aneka allows horizontal auto-scaling

What is a PaaS?

A PaaS might allow or not allow auto-scaling:

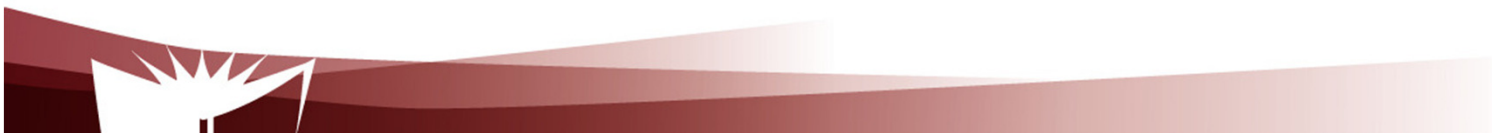
- Horizontal and/or
- Vertical

Note: Aneka allows horizontal auto-scaling

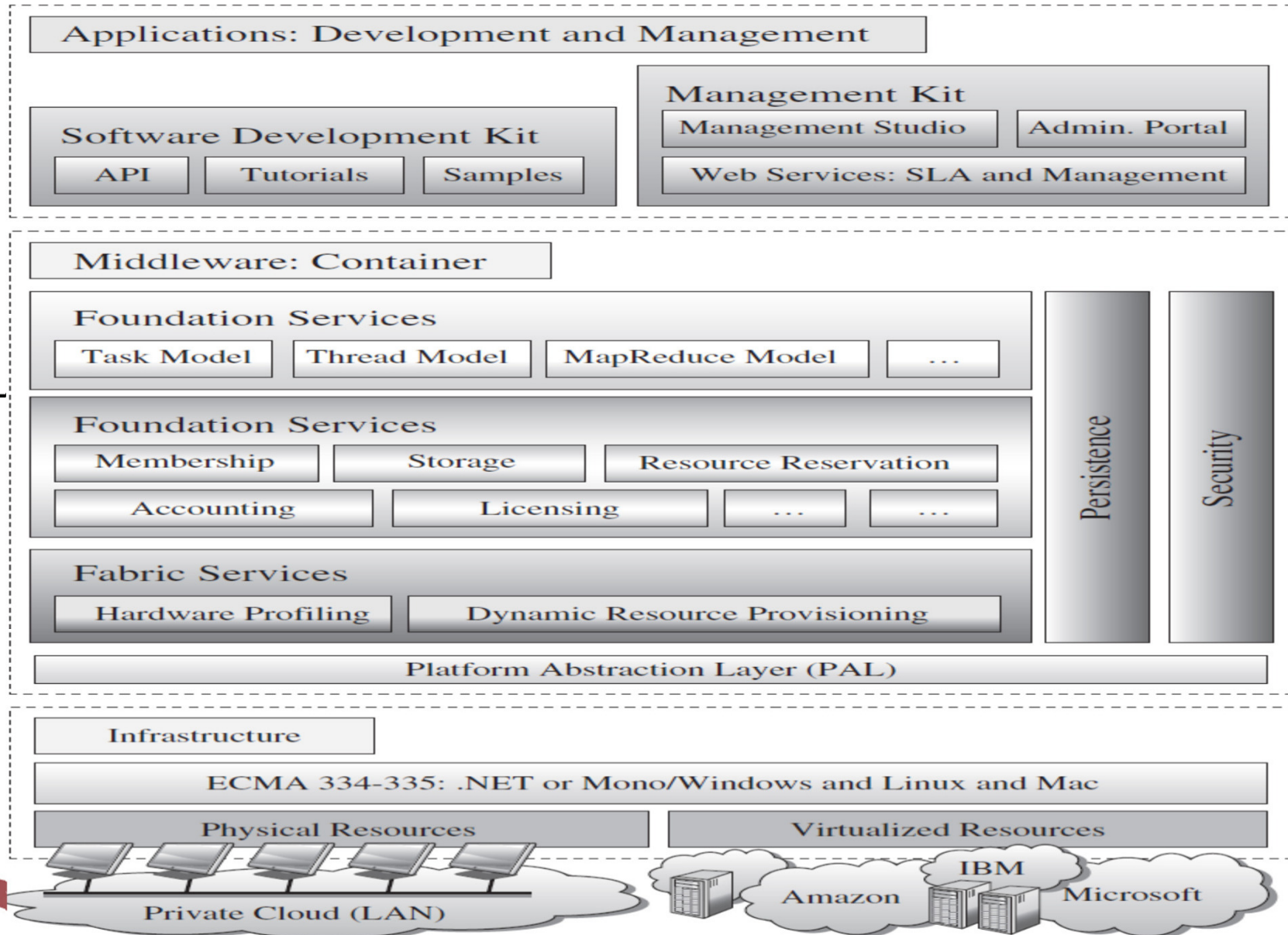


A Case Study: Aneka (Ref 3)

- Distributed applications development and management on clouds
 - Provision of resources in both private and public clouds
-
- Horizontal auto-scaling



A Case Study: Aneka (Ref. 3)



A Case Study: Aneka (Ref 3)

- Development services
 - Foundation services: Core management functions, e.g.
 - Metering
 - Resource allocation for execution
-
- Fabric services: Provide access to resources managed by the cloud, e.g.
 - dynamic resource provisioning



A Case Study: Aneka (Ref 3)

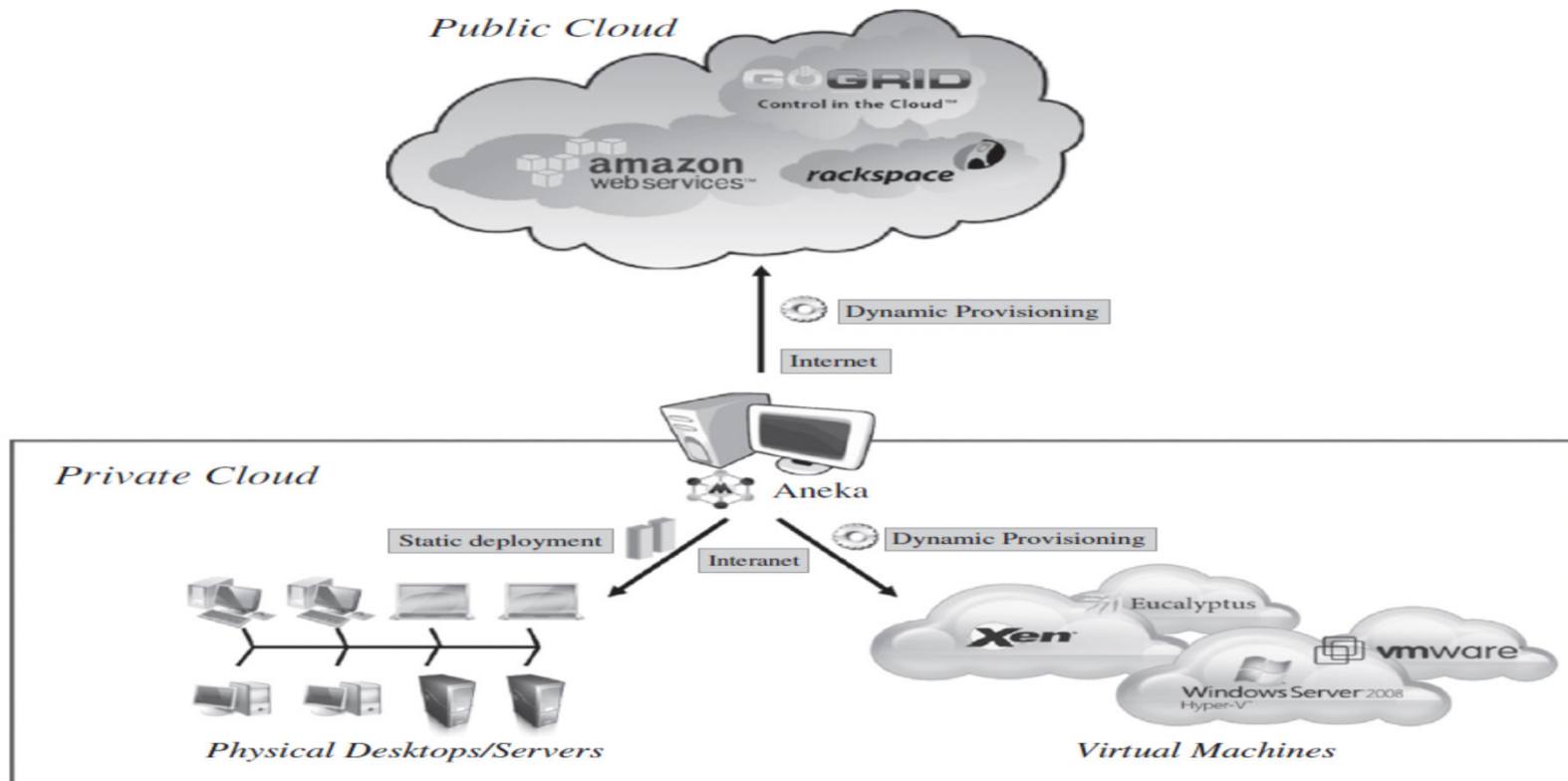


FIGURE 9.2. Aneka resource provisioning over private and public clouds.

A Case Study: Aneka (Ref 3)

Resource provisioning (Mission critical application that needs 30 VM at a maximum cost of 5 dollars and which should be completed within 1 hour)

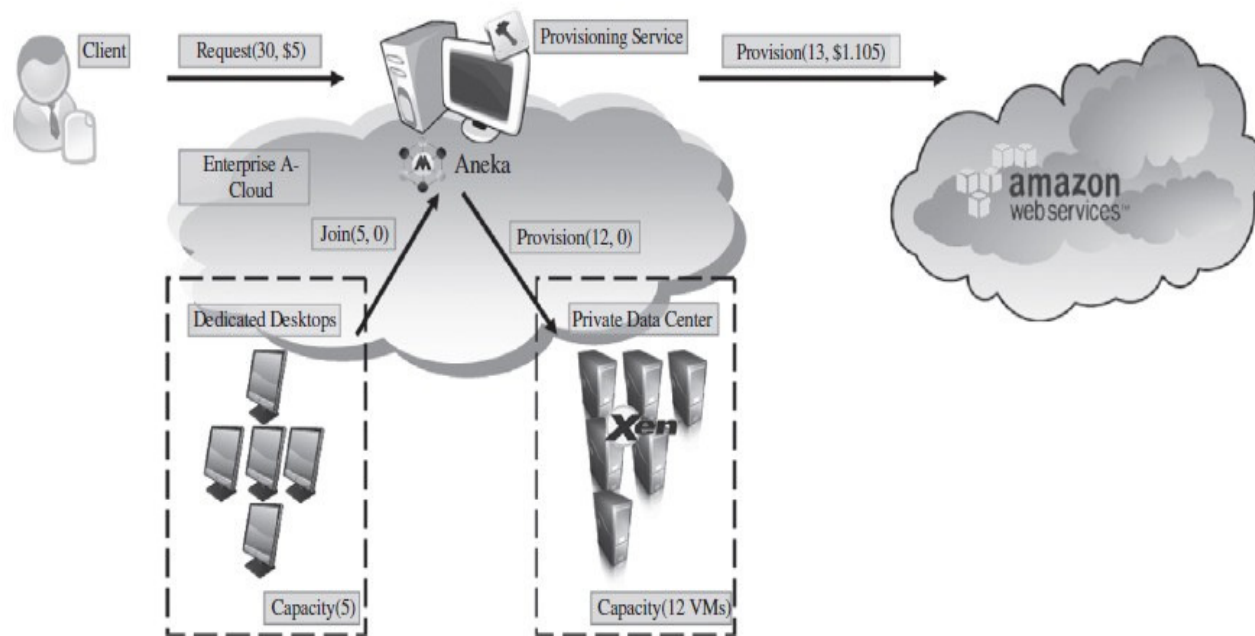


FIGURE 9.3. Use case of resource provisioning under Aneka.

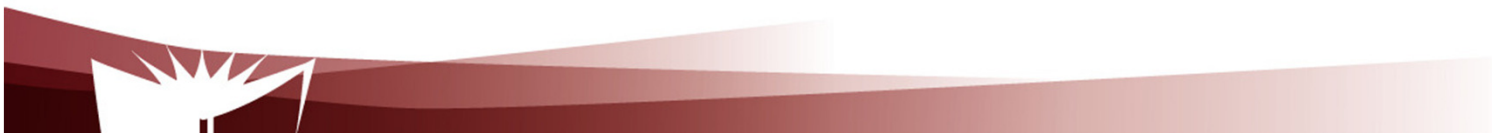
A Case Study: A Cloud Foundry (A Few Words)

- Open source endorsed by several large companies (e.g. IBM)
- Cater to multiple IaaS (e.g. Openstack, Amazon AWS)
- No auto-scaling (application / cloud user responsibility)
 - Horizontal scaling
 - Vertical scaling



A Case Study: A Cloud Foundry (A Few Words)

- An important concept: Droplet execution agent
 - Responsible for running applications in containers, called Warden containers
 - Warden containers ensure that applications run in isolation environment in terms of CPU usage, memory usage, disk usage, and network access
 - Starts or stops applications as required



The End

