



INFORMATIQUE MOBILE et NUAGIQUE

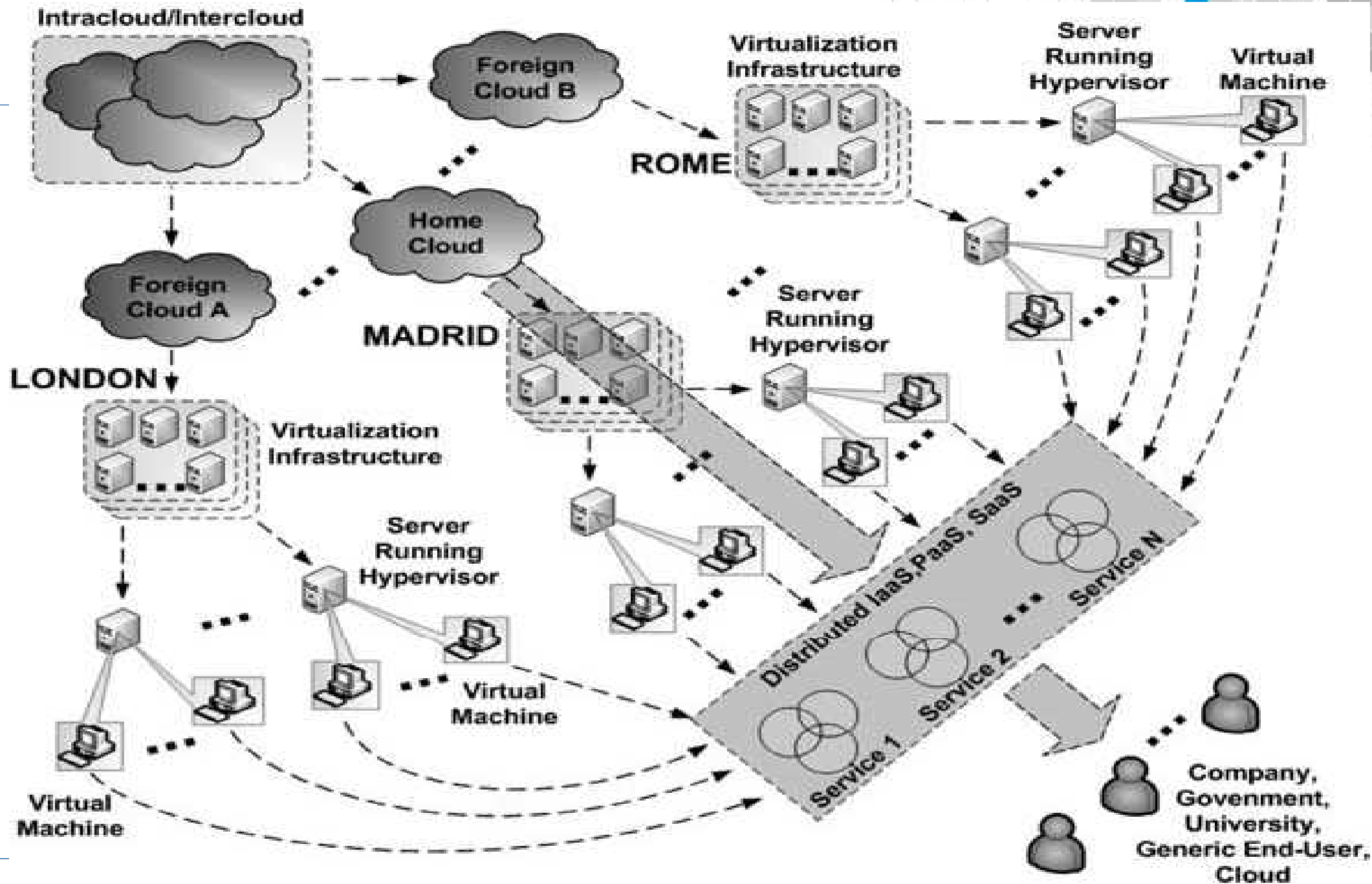
MASTER I : Réseaux et Technologies de l'information et de la communication

Labib TERRISSA

Université de Biskra - FSES NV
Département Informatique
Laboratoire d'informatique Intelligente (LINFI)



2014 - 2015



Plan du cours



I. Open Source Cloud

1. Introduction
2. Components of an Open-Source Cloud
3. Recurring Considerations
4. Open-Source Cloud Implementations
5. A Cloud Builder's Checklist
6. The Cloud Computing Software Stack

INTRODUCTION

- Open-source frameworks have a unique function in that they allow an organization to construct a “private” cloud environment.
- IaaS “Infrastructure as a Service”, in which a user requests some number of virtual machines (VM), which are hosted somewhere on the cloud’s own resources.

Then, What’s an OSC?

The key difference is that open-source cloud middleware allows any individual or organization to build their own cloud to host their own VM, without relying on an external, commercial cloud provider.

Components of an Open-Source Cloud

- The Underlying Physical Hardware

To build a cloud, some physical hardware is needed. It is important that this hardware have the processor and memory capacity to host numerous virtual machines. Also, network access and disk space must be accounted for.

- The Underlying Operating System

All of the components to build a private cloud must run on some operating system. The selection of the operating system will be critical for determining the compatibility issues that a private cloud will face between its various components.

Components of an Open-Source Cloud

- The Hypervisor

Also known as a Virtual Machine Monitor, or VMM, the choice of underlying hypervisor plays a huge role in the stability and performance of a cloud system.

- Disk Storage

In addition to storing the binaries for the running software, disk storage must be allocated for the disk image files that will form the basis for the virtual hard disks of the VM. This includes storing both the template disk images for future VM as well as the current disk images for running VM.

Components of an Open-Source

Cloud

- User Front-End

In some way, users must be able to request and access VM. Moreover, the front-end must be tuned to the usability and security requirements of the cloud provider and user.

- The Cloud Controlling Software

This component ties all of the parts together and coordinates them. Open-source options for this part include Eucalyptus, OpenNebula and Nimbus.

Recurring Considerations



An open-source cloud cannot be regarded as a single component. Rather, a private cloud is constructed out of a **Complete Cloud Computing Software Stack**.

In order to develop and maintain an open-source cloud, **Compatibility** must be constantly kept in mind. All of the hardware and software components must be able to talk to each other. In practice, we found that insuring such compatibility required numerous configuration tweaks to get a working system.

The cloud middleware of Eucalyptus, OpenNebula and Nimbus are **Active Projects**.

Recurring Considerations



Since this is open-source software, administrators can **Customize**. The ability to mix and match components, tweak configurations, or even write new code, allows private cloud providers to tune their systems to their own need.

If one is part of an organization, one might encounter a certain level of **System Limitations** determining some software configurations. A private cloud provider must conform their system to their organization's expectations regarding network configuration, available hardware, and underlying software. Maintaining working ties with other system administrators in the organization is critical to a successful cloud system.

Recurring Considerations



Organizations wishing to construct a private cloud must be aware of **User Requirements**. Fortunately, since private clouds are customizable, there is a great deal of opportunity to conform the cloud to what users want. Some considerations are security, ease-of-use and the ability to interface with previously used systems.

The construction and maintenance of a private cloud is an exercise in **Systems Administration**. All of the best practices for administering robust, stable systems apply in this setting as well.

Open-Source Cloud Implementations



There are several open-source projects available for constructing an Infrastructure as a service private cloud. These include

- Eucalyptus,
- OpenNebula,
- Nimbus.

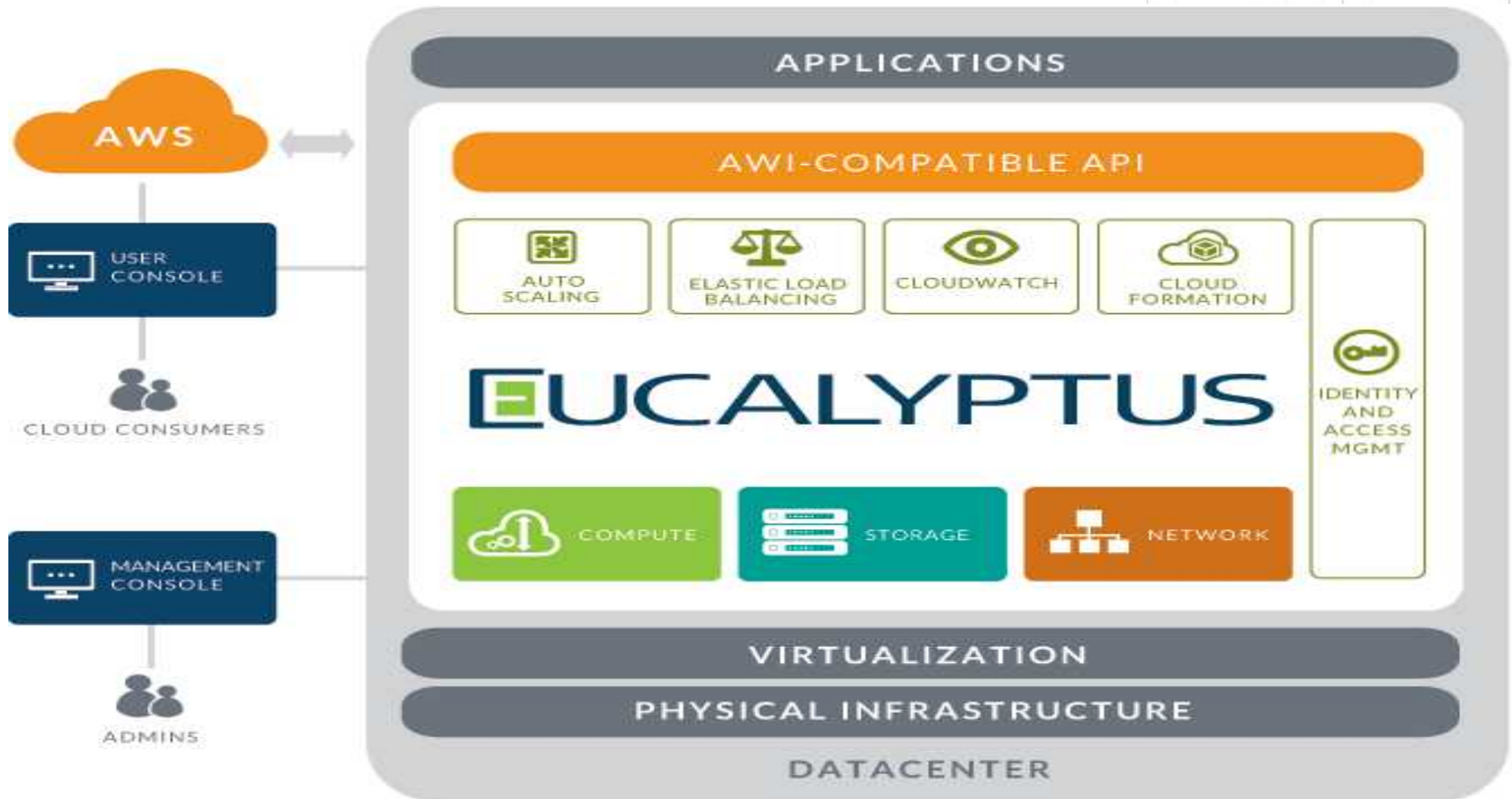
Since these are active projects under consistent development, the features and performance of these projects change over time.

Open-Source Cloud Implementations

EUCALYPTUS (Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems)

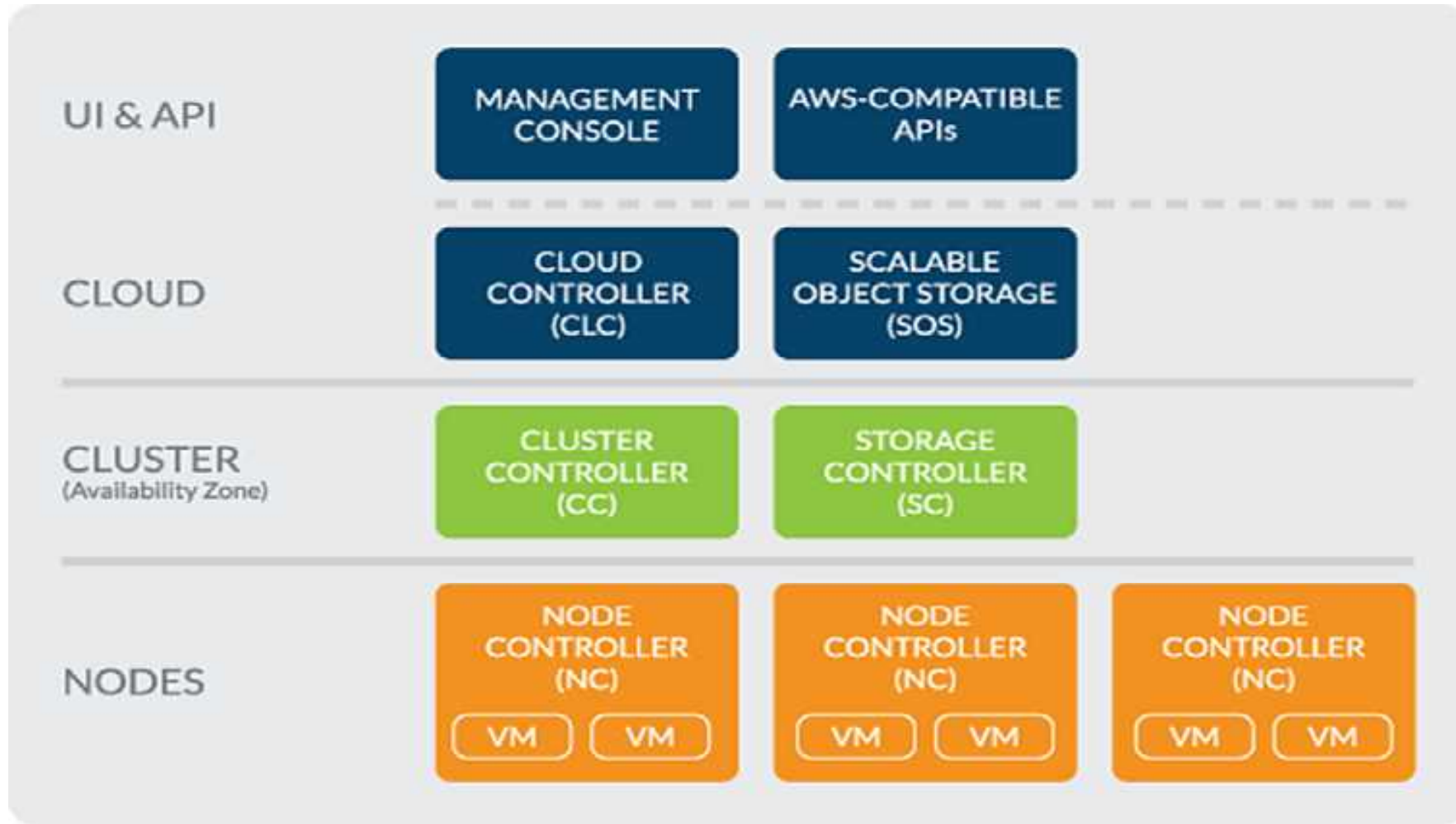
- An open-source imitation of Amazon's EC2.
- The front-end provided by Eucalyptus, called euca2ools, is designed to share compatibility with Amazon's EC2 front-end tools and the EC2 API.
- Eucalyptus implements a distributed storage system called "Walrus" which is designed to be compatible with Amazon's S3 storage.
- Internally, Eucalyptus is designed to be extensible enough to support large numbers of machines and users.
- Eucalyptus has a particularly decentralized design, permitting multiple "clusters" of machines in a single cloud.
- Eucalyptus provides a user-management web interface, allowing the cloud administrator to keep track of a large database of users and giving new users the ability to "signup" for the cloud.
- Eucalyptus very cleanly hides the internal working of the cloud from users.

Open-Source Cloud Implementations



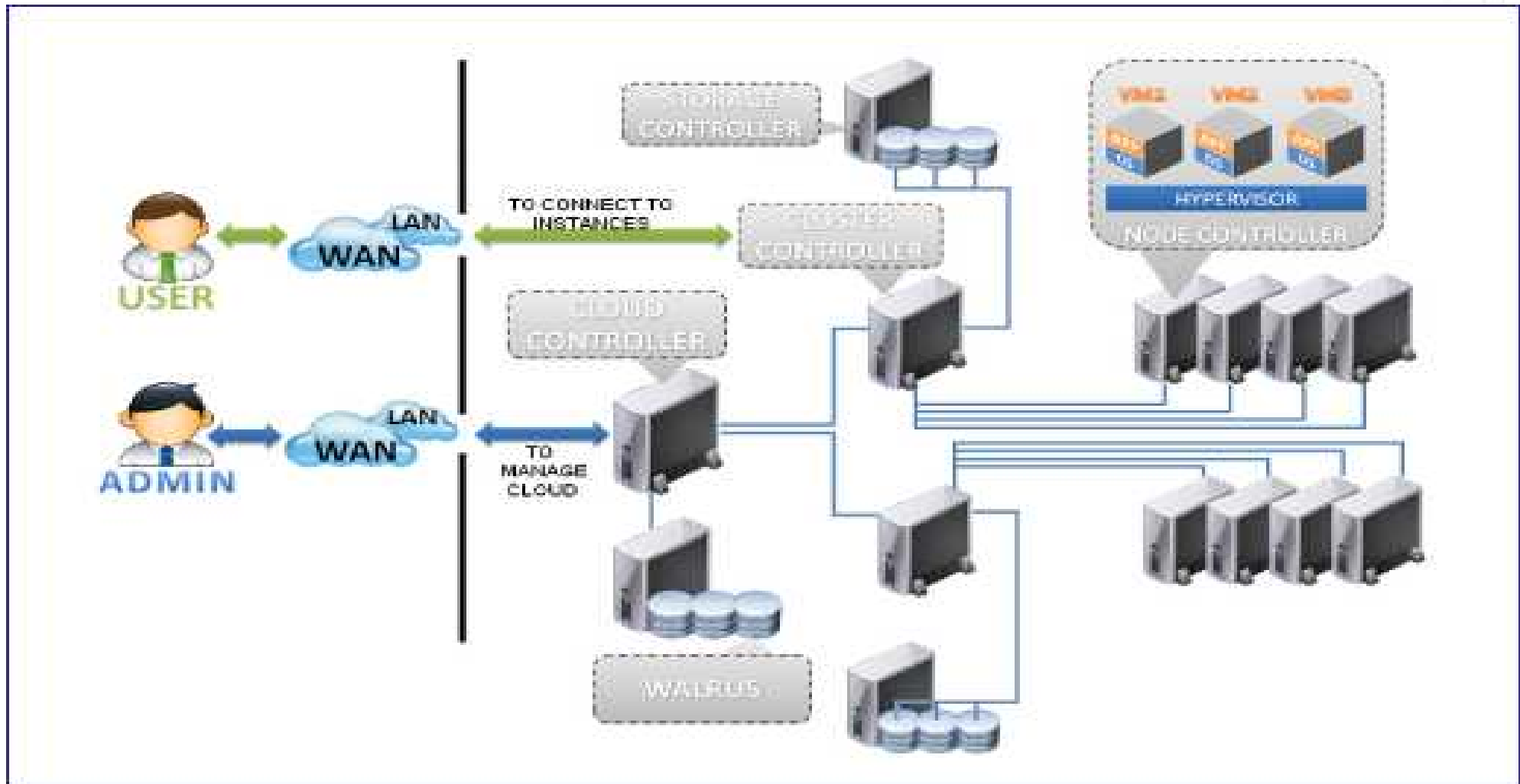
Eucalyptus cloud platform

Open-Source Cloud Implementations



Eucalyptus architecture diagram

Open-Source Cloud Implementations



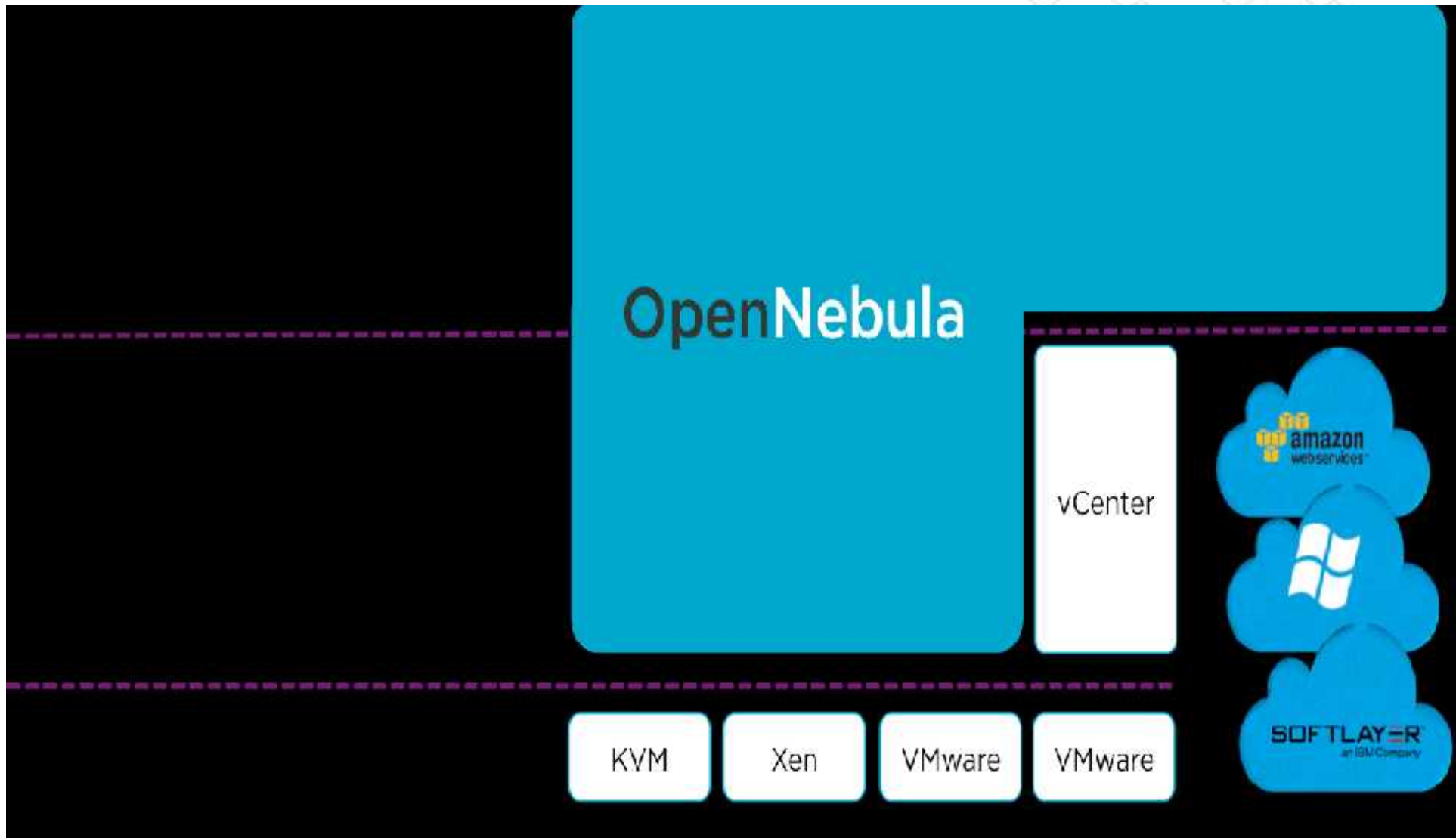
Eucalyptus architecture

Open-Source Cloud Implementations

OpenNebula

- OpenNebula's most distinctive feature is the very high degree of customizability that is given to administrators and users.
 - OpenNebula's available interfaces include a command line on the cloud's head node that users login to, an XML-RPC API.
 - If the cloud administrator wishes to use the default, command-line interface on the head node, users specify the VM configuration in a formatted file. This file allows for almost any configuration of memory, disk, network and more.
 - Internally, a great deal of customization is built around the underlying file system. OpenNebula can employ one shared file system (such as NFS or any distributed file system) for all files
- OpenNebula

Open-Source Cloud Implementations

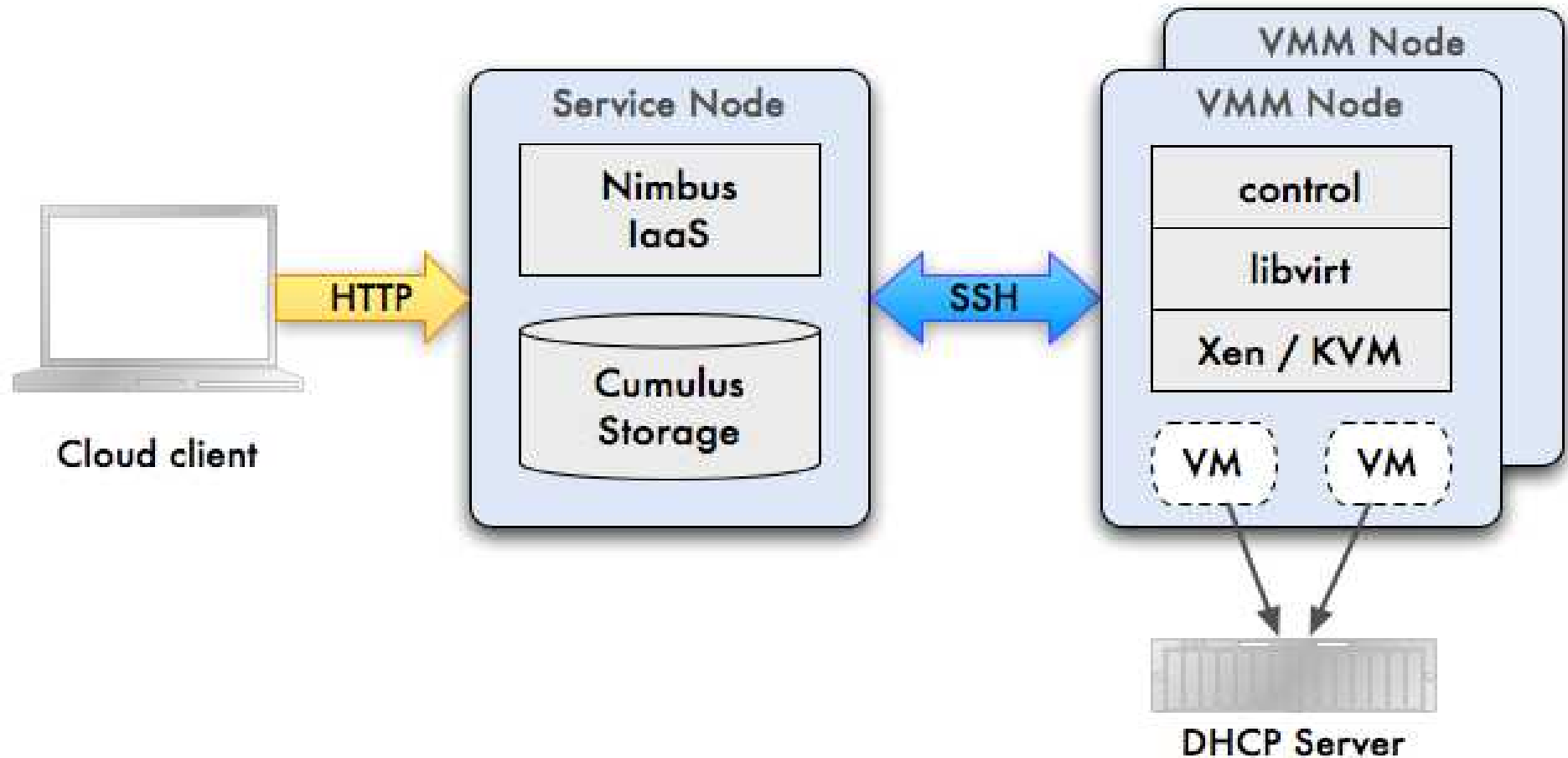


Open-Source Cloud Implementations

Nimbus

- The Nimbus project describes itself in terms of “cloud computing for science”.
- Nimbus allows for interaction with other clouds API, such as interfaces for the WSRF and EC2 protocols.
- Nimbus has recently implemented (version 2.5) an Amazon S3 compatible storage called Cumulus.
- Nimbus also has its own client front-end interface.

Open-Source Cloud Implementations



A Cloud Builder's Checklist

There are several practical issues that a builder of a private cloud should be aware of before beginning

- How is the Network set up?

First, whoever is building and maintaining a private cloud should clearly understand the various components of a TCP/IP network, including DHCP, DNS, virtual NIC and similar components.

Second, the cloud controller should be very clear on what limitations his or her organization places on the network setup. This includes, in particular, an awareness of the firewall setup and a clear understanding of the available IP address ranges of the organization.

Third, the administrator of the private cloud and the administrator of the organization's network should be working close together.

A Cloud Builder's Checklist



- Hardware Inventory and Expectation

The person building the cloud should have a clear notion of what hardware is currently available and what hardware will be made available in the future. In particular, the disk space, memory, network bandwidth and number of cores should all be noted. One of the key things to avoid is a mismatch in capacity.

- User Privileges

Many of the functions needed to perform a successful install of an open-source cloud require a careful understanding of the UNIX permissions used to perform the various tasks. Root privileges are typically needed for at least some parts of the install on all machines. However, not all tasks performed by various private clouds use root. Being aware of what privilege level should be used to run what commands can prevent frustrating errors.

A Cloud Builder's Checklist



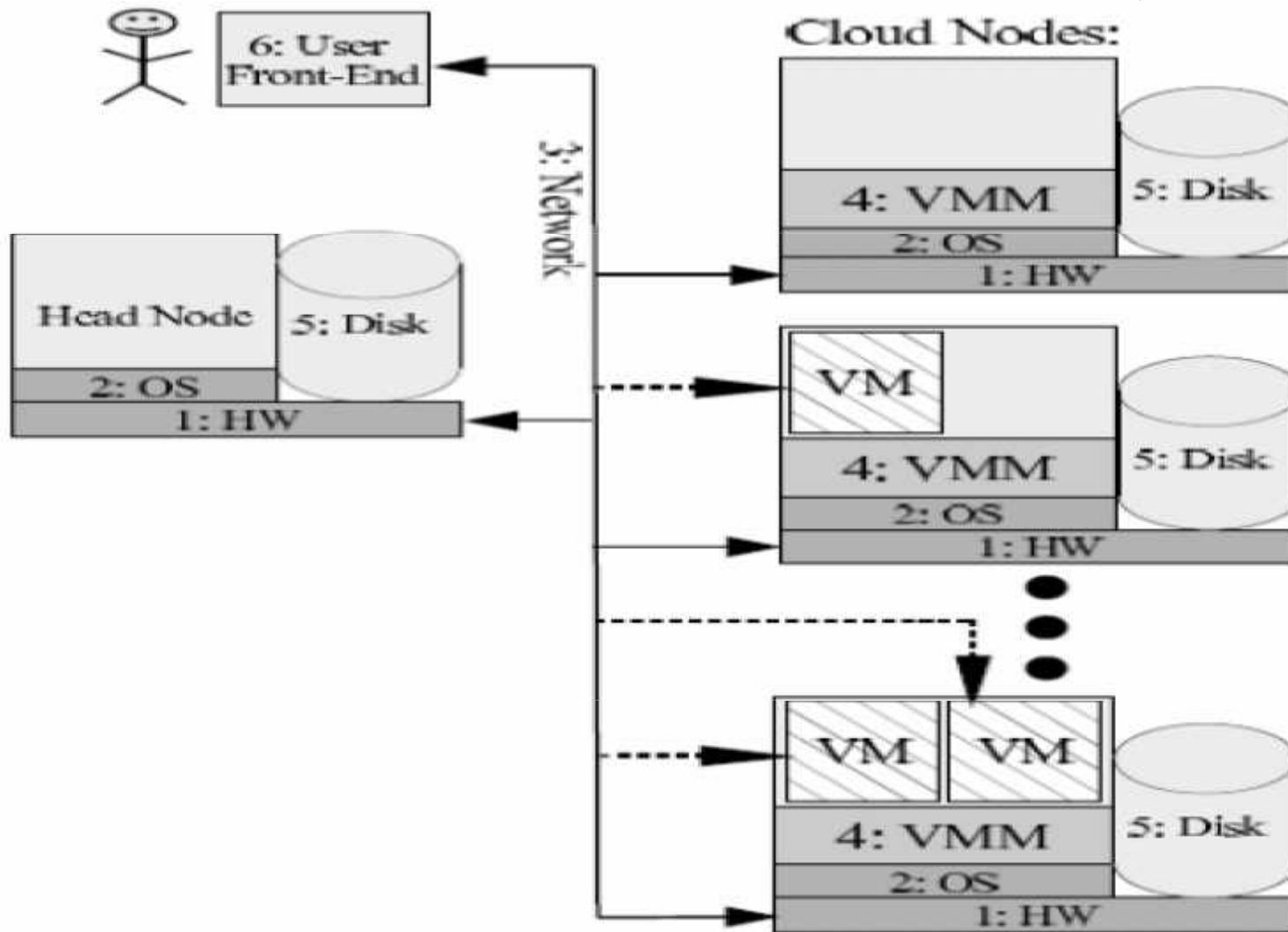
- User Expectations

The selection of Eucalyptus, OpenNebula and Nimbus probably depends on user expectations more than any other factor. In any case, being aware of what users can and can't do is critical to a successful private cloud setup.

- Be Ready To Tinker

before getting a successful setup with any of the common open-source clouds, it was required to tinker with configurations throughout the software stack. The best advice we can give is to locate all of the relevant log files for virtualization, user permissions, and the cloud software itself and check them frequently.

The Cloud Computing Software Stack







- S3
- EC2
- VPC
- Elastic MapReduce
- CloudFront
- RDS
- SNS

Navigation

Region: US East

- EC2 Dashboard
- INSTANCES
 - Instances
 - Spot Requests
- IMAGES
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE
 - Volumes
 - Snapshots
- NETWORKING & SECURITY
 - Elastic IPs
 - Security Groups
 - Placement Groups
 - Load Balancers
 - Key Pairs

My Instances

[Launch Instance](#)
Instance Actions
Reserved Instances
Show/Hide Refresh Help

Viewing: All Instances All Instance Types

| | Name | Instance | AMI ID | Root Device | Type | Status | Security Groups | Key Pair |
|-------------------------------------|------|------------|--------------|----------------|----------|------------|-----------------|----------|
| <input type="checkbox"/> | | i-bbc205d7 | ami-2d4aa444 | instance-store | m1.small | running | default | geheim_ |
| <input type="checkbox"/> | | i-9dc205f1 | ami-86db39ef | ebs | m1.large | running | default | new_key |
| <input type="checkbox"/> | | i-7ee299b | ami-fb16f92 | instance-store | m1.large | rebooting | default | geheim_ |
| <input checked="" type="checkbox"/> | | i-5fed2a33 | ami-a403f7cd | instance-store | m1.small | running | default | keypair_ |
| <input type="checkbox"/> | | i-59ed2a35 | ami-a403f7cd | instance-store | m1.small | running | default | keypair_ |
| <input type="checkbox"/> | | i-5ee2999 | ami-fb16f92 | instance-store | m1.large | terminated | default | geheim_ |

1 EC2 Instance selected

EC2 Instance: i-5fed2a33

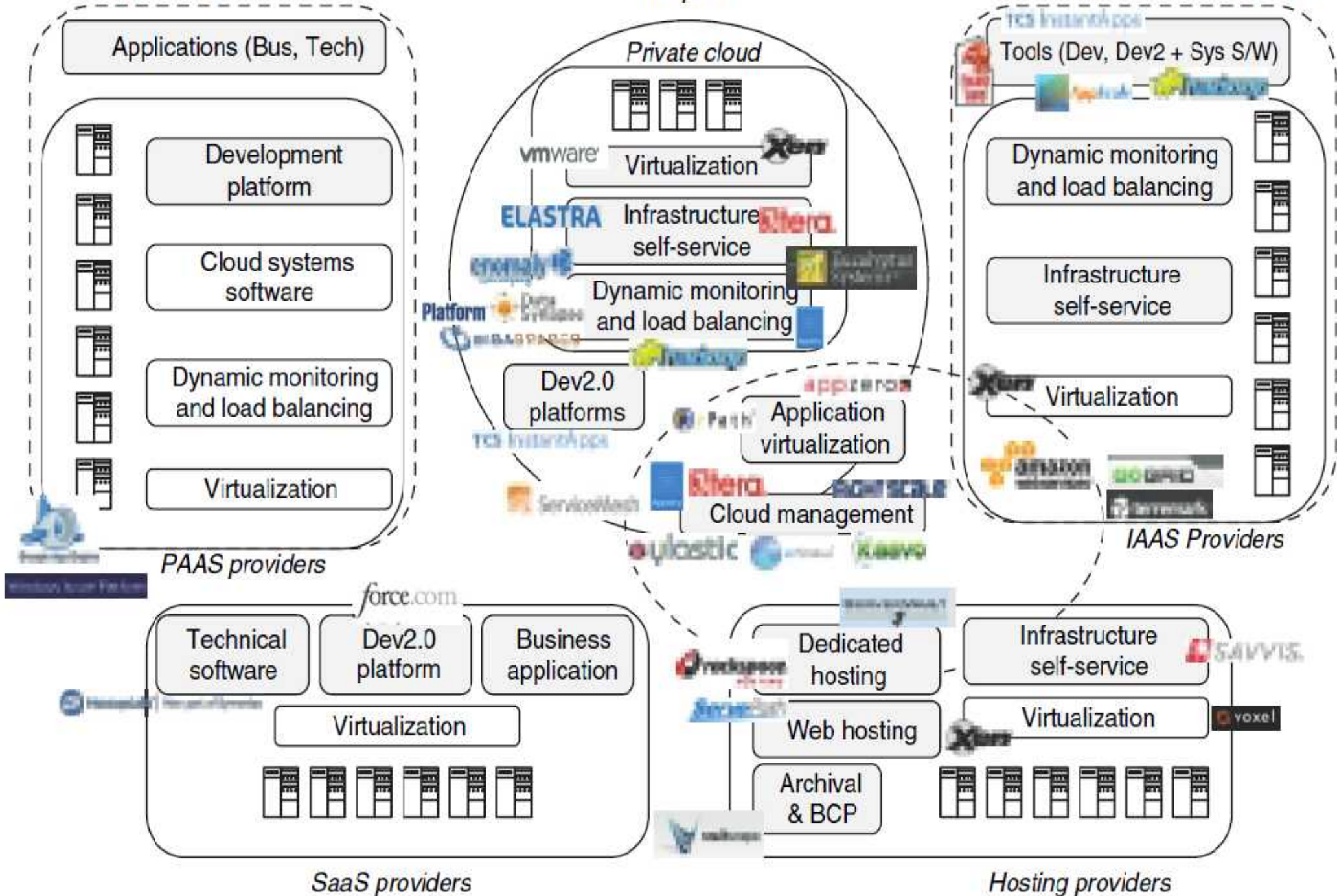
[Description](#)
[Monitoring](#)
[Tags](#)

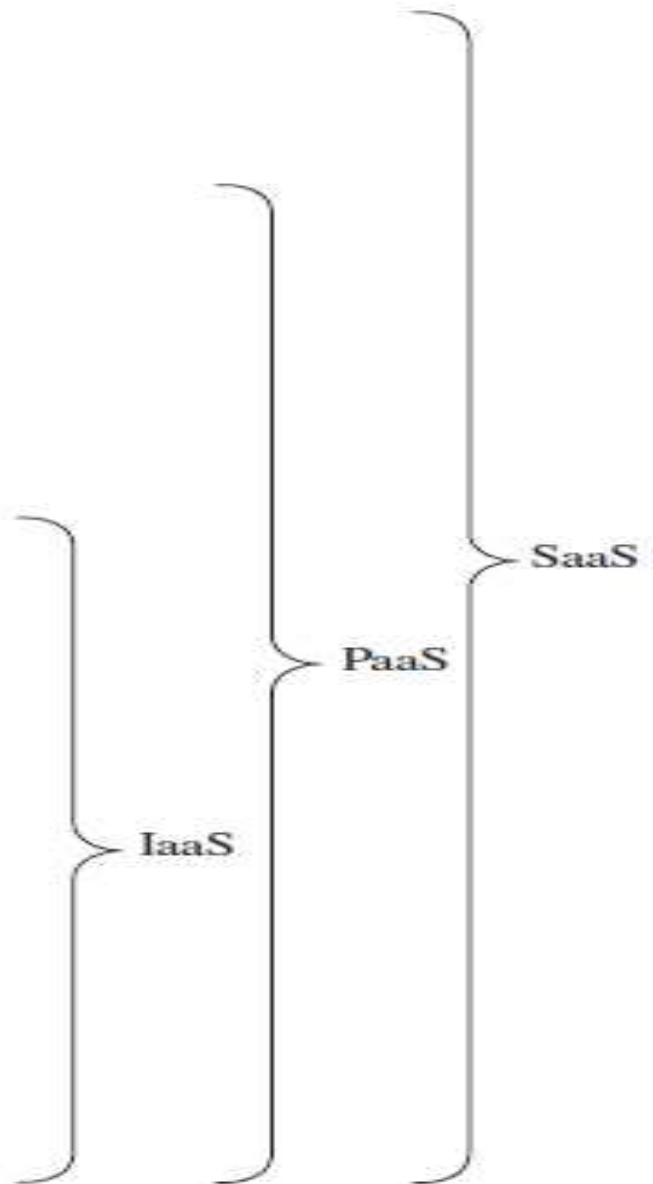
| | | | |
|------------------|--------------|--------|--------------|
| AMI ID: | ami-a403f7cd | Zone: | us-east-1a |
| Security Groups: | default | Type: | m1.small |
| Status: | running | Owner: | 178412210831 |

SaaS on PaaS

Enterprise IT

PaaS on IaaS





Questions ?



Merci de votre attention