Week1

Keh-Harng Feng

January 3, 2018

Contents

1 Cloud Computing

Why Cloud?

Using your own information systems:

1. Need capital investment (to buy hardware)
2. Heavy fixed costs (housing, air-conditioning etc)
3. Redundant expenditures (pay for admins, ITs, programmers, etc)
4. Energy costs.
5. Unreliable hardware (frequent failures)
6. High levels of overcapacity (not utilizing the hardware/labor all the time).

Overall, not sustainable.

1.1 Cloud Service Delivery Models

- Infrastructure as Service (IaaS)
  - Rent processing, storage, network (ie: cloud provides computer hardware)
  - Capacity and other fundamental computing resources.
  - Ex: Amazon EC2, S3
- Platform as a Service (PaaS)
  - Deploy customer-created applications to a cloud.
  - Cloud provides operating system (ie: Google's App Engine)
- Software as a Service (SaaS)

    - Cloud provides software application over a network to users.
    - Ex: Gmail, Google Docs.

1.2 Synergy

The rise of virtualization and big data are synergizing with cloud computing.

Big data:

- Data collection too large to transmit economically over Internet (petabyte range)
- Computation is data intensive.
    - Lots of Disks, Networks and CPUS.
    - Overhead of maintaining cyber infrastructure is expensive.
    - Users buy Big Data services from Clouds to share overhead.
- Goals: Easy to write programs, fast turn around.
- MapReduce - Hadoop, PIG, HDFS, HBase

1.3 Cloudonomics

- Cloud services don't need to be cheaper to be economic!
- Consider a car
    - Buy or lease for $10 per day.
    - Rent a car for $45 a day.
    - If you need a car for 2 days in a trip, buying would be much more costly than renting.
    - It depends on the **demand**.

When is cloud cheaper?

If the demand is highly variable (ie: peak demand is much larger than average demand), then cloud is often more economical.

In practice demands are often highly spiky (ie: news stories, marketing promotions, product launches, Internet flash floods, tax season, Christmas shopping, etc).

Often a hybrid model is the best:

- You own a car for daily commute, and rent a car when traveling or when you need a van to move.
- Key factor is again the ratio of peak to average demand.
- But we should also consider other costs.
    - Network cost (both fixed costs and usage costs).
    - Interoperability overhead.
    - Consider Reliability, accessibility etc.

So for a company, own part of the infrastructure for the average demand, and be prepared to utilize cloud for peak demand.

1.3.1 Common Infrastructure

Check 1.1.4 again.

- For infrastructure built to peak requirements: try to allow multiplex (ie: use the infrastructure for other tasks if not at peak demand). This allows lower cost per delivered resource than unconsolidated workloads.
- For infrastructure built to less than peak: multiplexing reduce unserved demand over the capacity.

1.4 Big Data

- A collection of data sets so large and complex, it's impossible to process it on one computer with the usual databases and tools.
- Because of its size and complexity, Big Data is hard to capture, store, copy, delete (privacy issues), search, share, analyze and visualize.

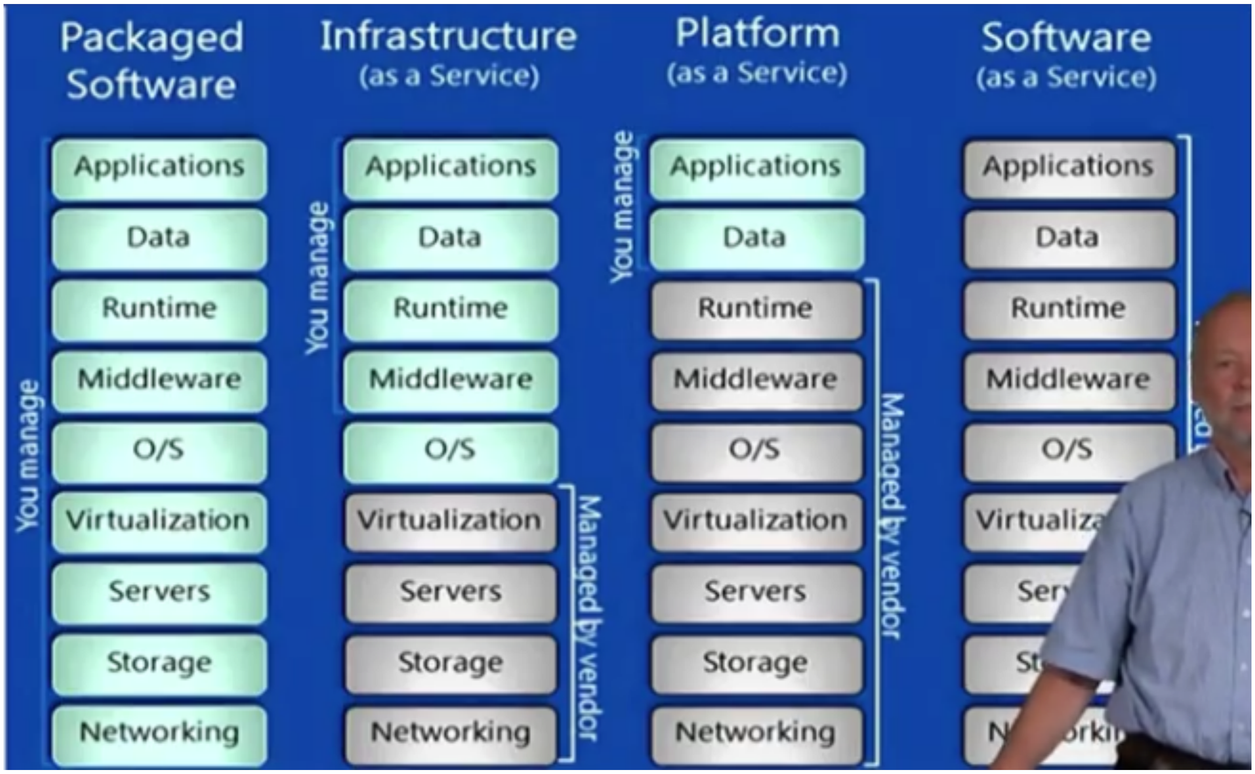
Big Data represents information assets characterized by

1. High Volume
2. High Velocity
3. High Variety

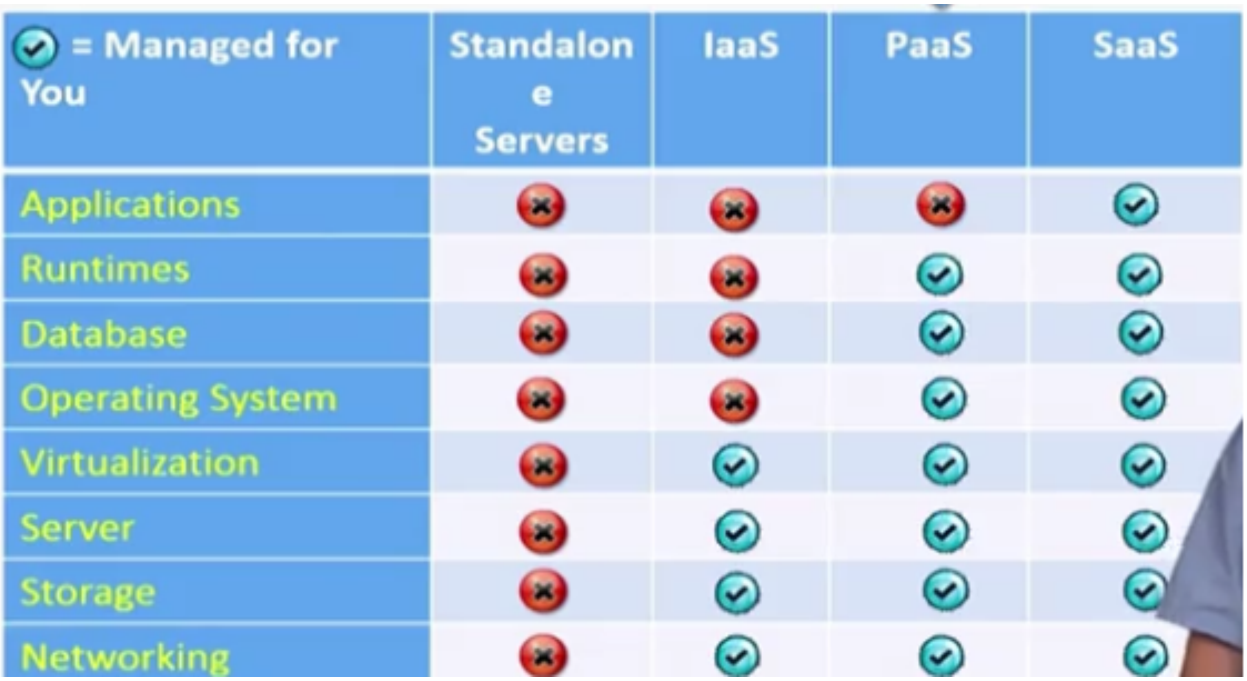
2 Software Defined Architecture

- Cloud privides services, service orchestration and provisioning.
- A cloud may provide IAAS, PAAS, SAAS and have both internal and external API.
- The mechanisms and concept of providing services, orchestration and provisioning is called a Software Defined Architecture
- A cloud may contain other software defined entities:
  - software defined network
  - software defined storage
  - software defined compute
- The point of SDA is to increase flexibility and job multiplexing -> dynamic load balancing etc.
- Virtulization (ie: containers) can be used to achieve this.

3 Cloud Services



Packaged Software	Infrastructure (as a Service)	Platform (as a Service)	Software (as a Service)
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualiza
Servers	Servers	Servers	Ser
Storage	Storage	Storage	St
Networking	Networking	Networking	N orki



✓ = Managed for You	Standalon e Servers	IaaS	PaaS	SaaS
Applications	✗	✗	✗	✓
Runtimes	✗	✗	✓	✓
Database	✗	✗	✓	✓
Operating System	✗	✗	✓	✓
Virtualization	✗	✓	✓	✓
Server	✗	✓	✓	✓
Storage	✗	✓	✓	✓
Networking	✗	✓	✓	✓

- Infrastructure as a Service (IaaS): basic compute and storage resources
  - On-demand servers
  - Amazon EC2, VMWare vCloud
- Platform as a Service (PaaS): cloud application infrastructure
  - On-demand application-hosting environment
  - Eg: Google AppEngine, Salesforce.come, Windows Azure, Amazon
- Software as a Service (SaaS): cloud applications
  - On-demand applications
  - Eg: GMail, Microsoft Office Web Companions

3.1 Platform as a Service (PaaS)

- PaaS is a cloud computing service that offers a platform for users to run applications on the cloud.
- It is a level above Infrastructure as a service (IaaS) because unlike IaaS, PaaS does not require users to develop their own operating system environment.
- Middle ground between SaaS and IaaS
- Development platform: customers use to develop applications that benefit from the scalability of the cloud without fully developing their own solution using an IaaS provider.
- Offers an application development platform that will automatically scale with demand.

NIST Definition of PaaS:

“The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming langauges, libraries, services and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.”

Examples: Google AppEngine, Windows Azure

Windows Azure:

- Provides
  - Compute (web, worker and VM role)
  - Storage (Blob, Table, Queue and Azure SQL server)
  - Application fabric (service bus, access control, in the future -> cache, integration and composite)

Most Cost Effective!

PaaS can be better for costs than IaaS, because systems are optimized to run applications efficiently.

IaaS may only provide hardware and thus, clients must be in charge of load balancing, networking etc.

3.1.1 Multi-tenancy

PaaS is better suited for multi-tenancy (many users may share the same physical computer and database). This is because

1. PaaS provider optimizes its infrastructure for use by many providers.
2. An IaaS may provide each user with his own VM, this creates a clear separation of resources. In PaaS users may share the same machine, database etc.

3.1.2 Vendor Lock-in

- PaaS may lock in applications by requiring users to develop apps using proprietary interfaces and languages.
- This means that it may be difficult for users to go to another vendor to host their app.
- Businesses may risk their future on the dependability of the PaaS.

3.1.3 Development Tools

- Often a PaaS will offer browser-based development tools
- In this way, developers can create their own applications online.
- Ease of deployment: the platform takes care of the scaling for you.
- As a developer, your objective is to create an application in the quickest, most effective way possible -> don’t create applications using convoluted methods that might take a long time to complete.
- The user sees the end product, not the development process.

3.1.4 PaaS vs IaaS

- When you use the Cloud, remember that your decisions have long-term consequences
- If you choose to use a PaaS and get your application vendor locked in, then your business may fail if the PaaS greatly increases the vendor's prices (you will not be able to move to another Cloud since your app cannot be easily migrated to somewhere else).
- An app that is used to fulfill a temporary need may be handled by a PaaS solution quickly.
- An app that needs to be deployed quickly may be developed faster by a PaaS
- If your software team is small, it may be better to develop a PaaS and let the PaaS provider handle the OS and networking for your team.

3.2 Infrastructure as a Service (IaaS)

Examples:

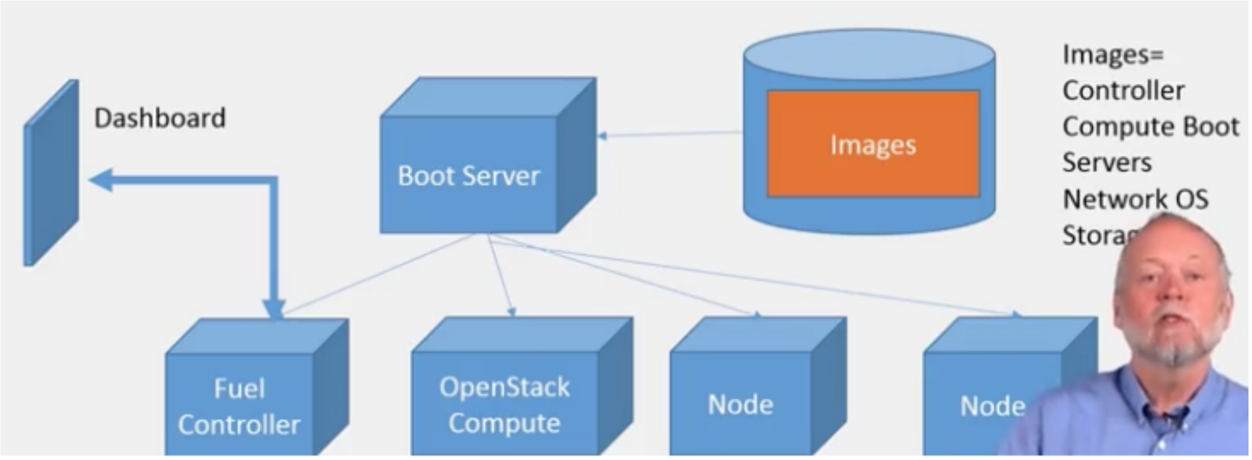
- Xen
- XenCloud
- OpenStack

Xen and the Linux Kernel Distributions

- Xen was initially a University research project.
- Invasive changes to the kernel to run Linux as a paravirtualized guest.
- Even more changes to run Linux as Dom0 -> required to make sure VM IOs are independent of each other.
- Xen support in the Linux kernel; no upstream.
- Great maintenance effort on distributions.
- Risk of distributions dropping Xen support.
- Hard to use.

Mirantis Fuel and OpenStack Architecture

Mirantis:

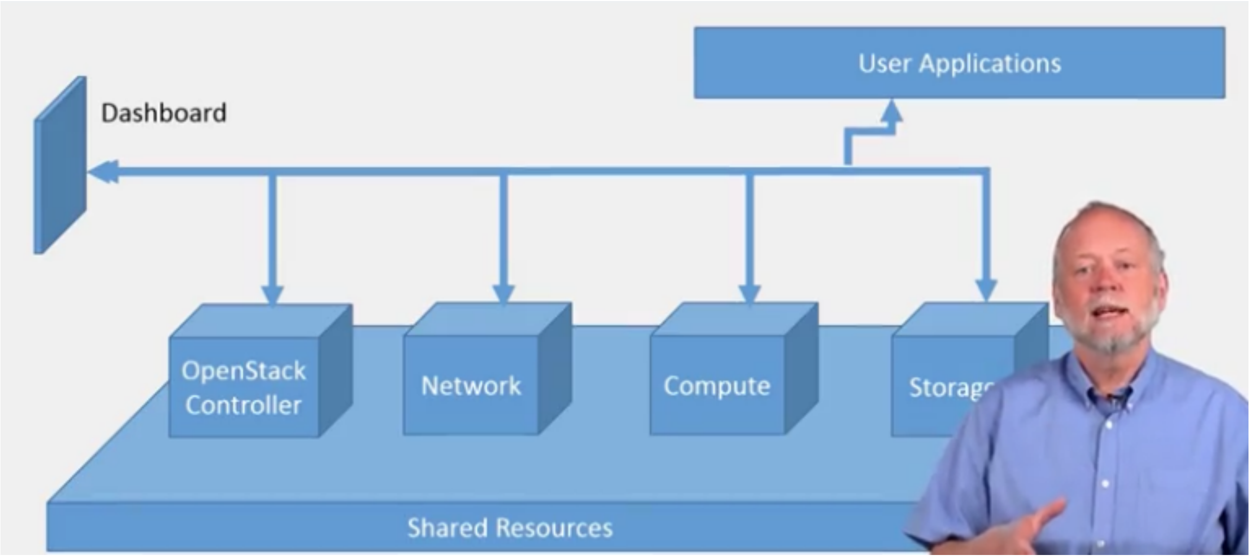


Dashboard: user control interface.

Mirantis brings up a Fuel controller.

The Fuel controller allows one to bring up OpenStack controller, networks, compute servers and storage.

OpenStack:


User Applications
Dashboard
OpenStack Controller
Network
Compute
Storag
Shared Resources