**Features of Amazon EC2**

Amazon EC2 provides the following features:

* Virtual computing environments, known as **instances**
* Preconfigured templates for your instances, known as **Amazon Machine Images (AMIs)**, that package the bits you need for your server (including the operating system and additional software)
* Various configurations of CPU, memory, storage, and networking capacity for your instances, known as **instance types**
* Secure login information for your instances using **key pairs** (**AWS stores the public key, and you store the private key in a secure place**)
* Storage volumes for temporary data that's deleted when you stop or terminate your instance, known as **instance store volumes**
* Persistent storage volumes for your data using **Amazon Elastic Block Store** (Amazon EBS), known as Amazon EBS volumes
* Multiple physical locations for your resources, such as instances and Amazon EBS volumes, known as **regions and Availability Zones**
* A firewall that enables you to specify the protocols, ports, and source IP ranges that can reach your instances using security groups
* Static IPv4 addresses for **dynamic cloud computing**, known as **Elastic IP addresses**
* Metadata, known as **tags**, that you can **create and assign to your Amazon EC2 resources**
* Virtual networks you can create that are logically isolated from the rest of the AWS cloud, and that you can optionally connect to your own network, known as **virtual private clouds** (VPCs)

Just as Amazon Simple Storage Service (Amazon S3) enables storage in the cloud, Amazon EC2 enables “compute” in the cloud.

We can create an AMI using the EC2, which allows setting and configuring everything about your instances from the operating system up to your applications.

An AMI is a packaged-up environment that includes all the necessary details to set up and boot your instance.

EC2 provides a number of tools to make creating an AMI easy. After creating a custom AMI, you have to bundle it. If you are bundling an image with a root device backed by Amazon EBS. You can simply use the bundle command in the AWS Management console.

If you are bundling an image with a boot partition on the instance store, then you need to use the AMI Tools to upload it to the Amazon S3.

EC2 uses amazon EBS and S3 to provide reliable, scalable storage of your AMIs so that you can boot whenever you want.

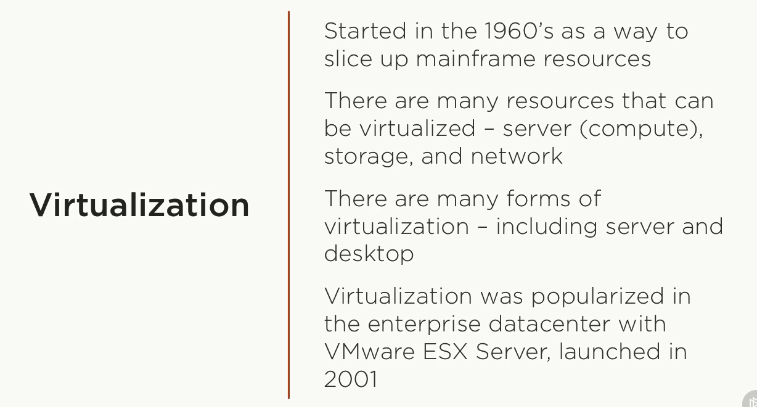
You can also use standard Linux distribution AMIs

SAAS – salesforce.com, office 365

IAAS – EC2 beta, Azure, google compute engine

PAAS – Google App Engine, heroku, AWS elastic beanstalk

**Virtualization**



Running VMware in the windows is an example

**Hypervisor** is virtual machine monitor that allows us to create a virtual machine. The system on which hypervisor runs one or more virtual machine is the Host machine and each virtual machine is the **guest machine**. Example of the hypervisors are VMware Fusion, Oracle Virtual box etc.

Hypervisors can be divided into two types:

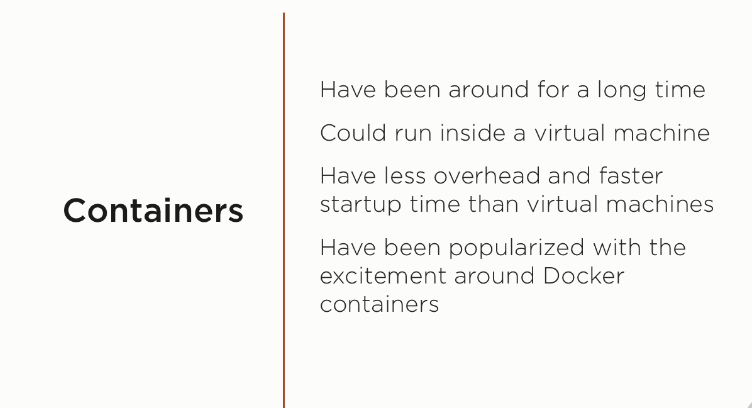
**Type 1**: Also known as native or bare-metal hypervisors, these run directly on the host computer’s hardware to control the hardware resources and to manage guest operating systems. Examples of Type 1 hypervisors include VMware ESXi, Citrix XenServer and Microsoft Hyper-V hypervisor.

**Type 2**: Also known as hosted hypervisors, these run within a formal operating system environment. In this type, the hypervisor runs as a distinct second layer while the operating system runs as a third layer above the hardware.

**Container**

A container is operating system level virtualization where the OS kernel provides isolated user spaces to run the specific applications

Can be used inside the virtual machine and have much less overhead

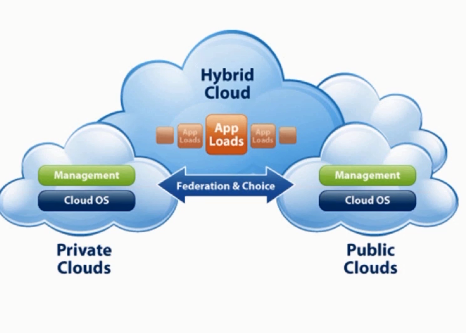


Fully self-contained applications inside a container

For example, Kitematic is a free Docker tool, running hello world Nginx, we can download the image and don’t even need a lot of command line to run this container, we can also see a web preview of the container. Runs in the secure area in a desktop. Containers are the big part of the cloud computing.

Completely portable and runs on any computer.

**Private cloud** – runs in own building, leverage own infrastructure

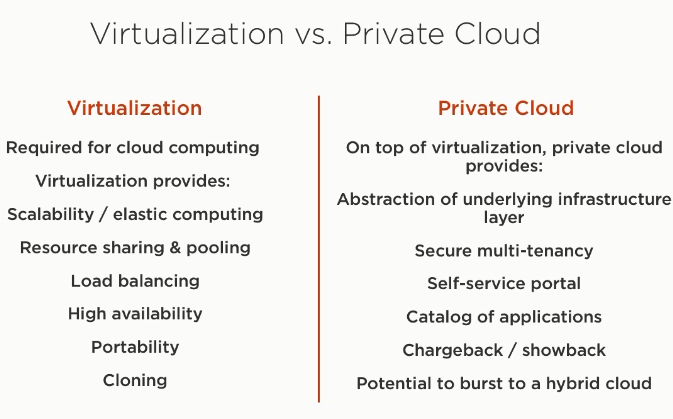


**Public clouds** - amazon EC2, Microsoft azure

**Hybrid cloud** – connected private cloud to the public cloud. Two different clouds are working together.

For example, you have your own private and suddenly due to Christmas season coming up you need more capacity and you already have a public cloud so you can scale up

Enterprise cloud is similar to the hybrid cloud.

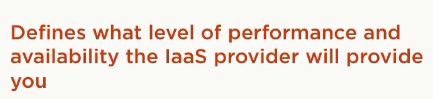


**IaaS pricing**

* On Demand
* Spot instances
* Reserved Instances
* Dedicated Hosts

Ex - Amazon EC2, google compute engine, Microsoft Azure

**SLA with your cloud provider**







**Migration To the cloud**

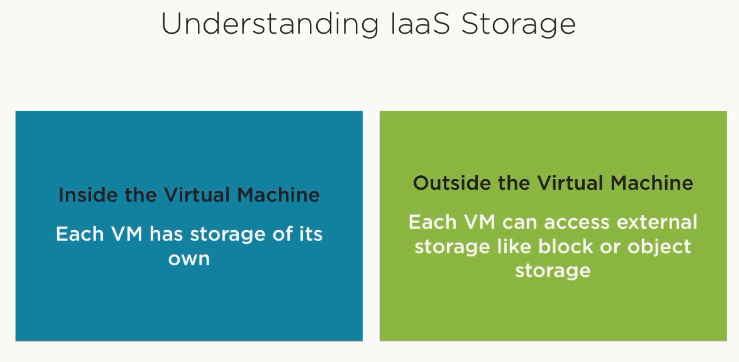
Factors to keep in mind:

* Costs
* Security, availability and performance
* Migrate vs Build
* Enterprise grade functionality
* Tools that can help

**Connecting to the cloud using a VPC (Virtual Private Network)**

We can create a virtual private network using the AWS.

**Storage in the cloud**



**Inside the VM**

Outside of the VM ex Amazon EBS allows you store data in the cloud and that can be accessed by the different Virtual Machines

**Object storage** – perfect for unstructured data for example if you have a 1000 graphics on a website that needs to be published and accessible to the other website. Those would be stored in object storage with no particular order

**Cloud storage solutions**

Dropbox, Microsoft OneDrive, Google Drive

**Object Storage**

Ex of unstructured data - Pictures, graphic files, archival data, videos

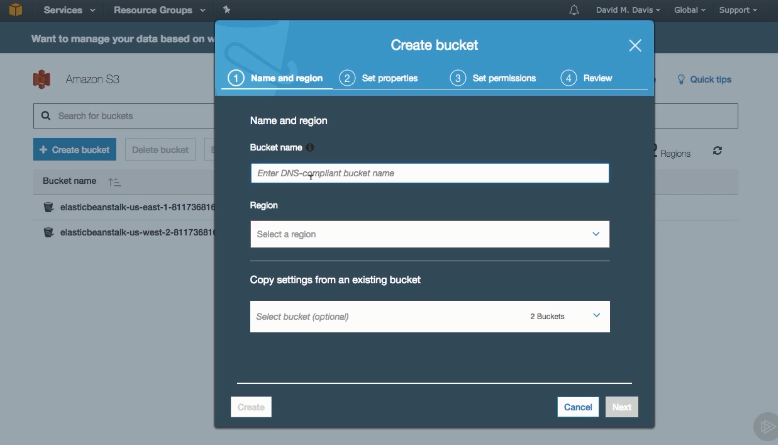
Amazon S3 - simple storage Service

Microsoft Azure Blob Storage

Great for low priority less critical data, for ex amazon s3 had a massive outage, lot of web applications went down. So the websites needed to be designed accordingly to distributed data across multiple object storage solutions out on the internet.

CEPH, SWIFT – are the private object storage solutions on premises.

We can set the Amazon S3 by creating a bucket

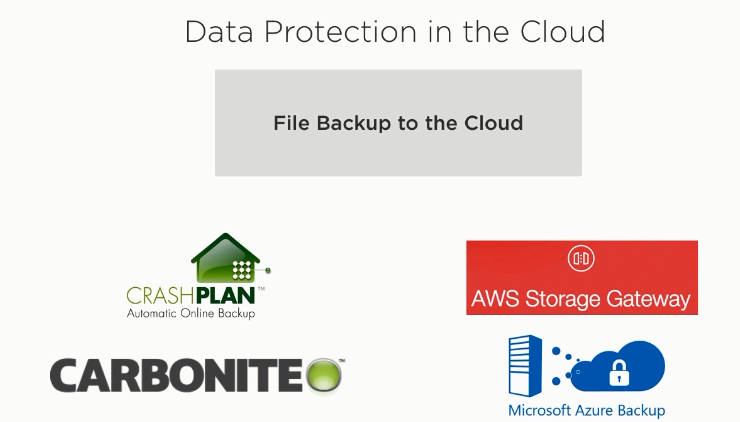


This scales virtually infinitely up and down and you only pay for the files that you actually store in amazon S3

**Data protection in the cloud**

File backup to the cloud

Many data protection solutions



But if you not just files but the virtual machines, then we can use disaster recovery in the cloud as well so in this case an entire VM with the OS applications and data will be replicated or copied in many cases with as frequent as 15 min.

In this the good thing is that **you only pay when you use the service** **in case of the disaster**



**Security in IaaS**

Cloud security concerns

Encryption

Compliance

Vulnerabilities and Mitigation



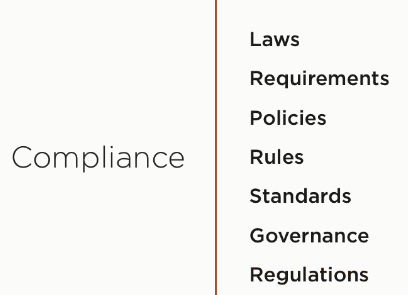
We can study resources, security bulletins

**Encryption**

Conversion of information into the cypher text so that only the party that has key can access that information

Access key public and private in AWS

**Compliance**



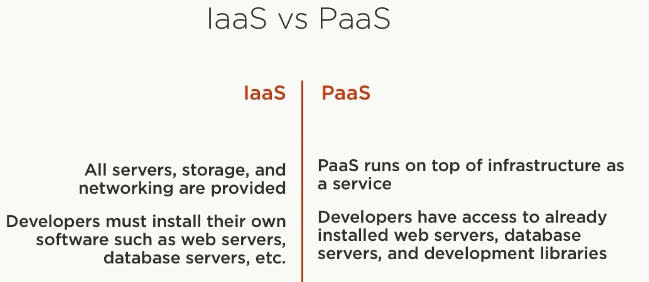
**Platform as a Service**

For developers

A cloud service for developers who want to develop, run and manage applications

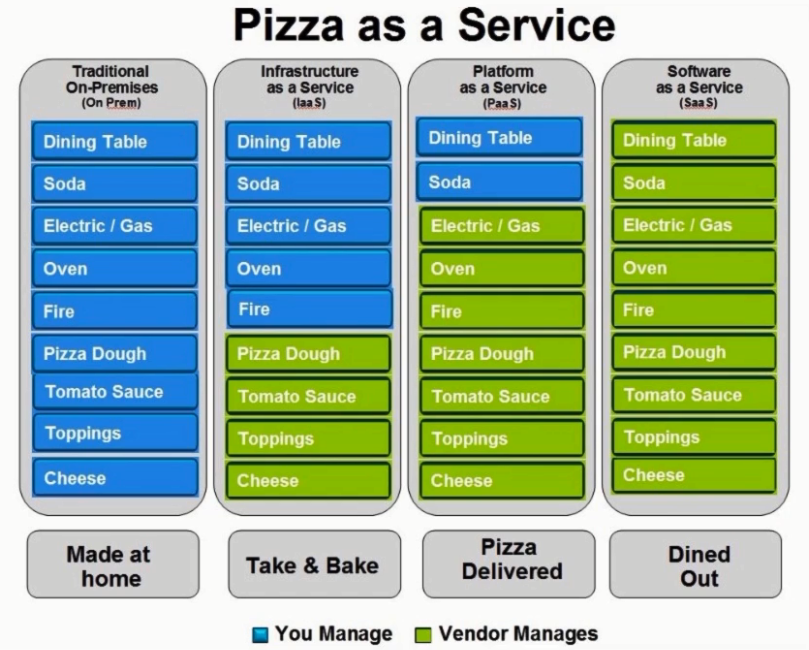
No servers, storage, network, OS, middleware, or databases are needed

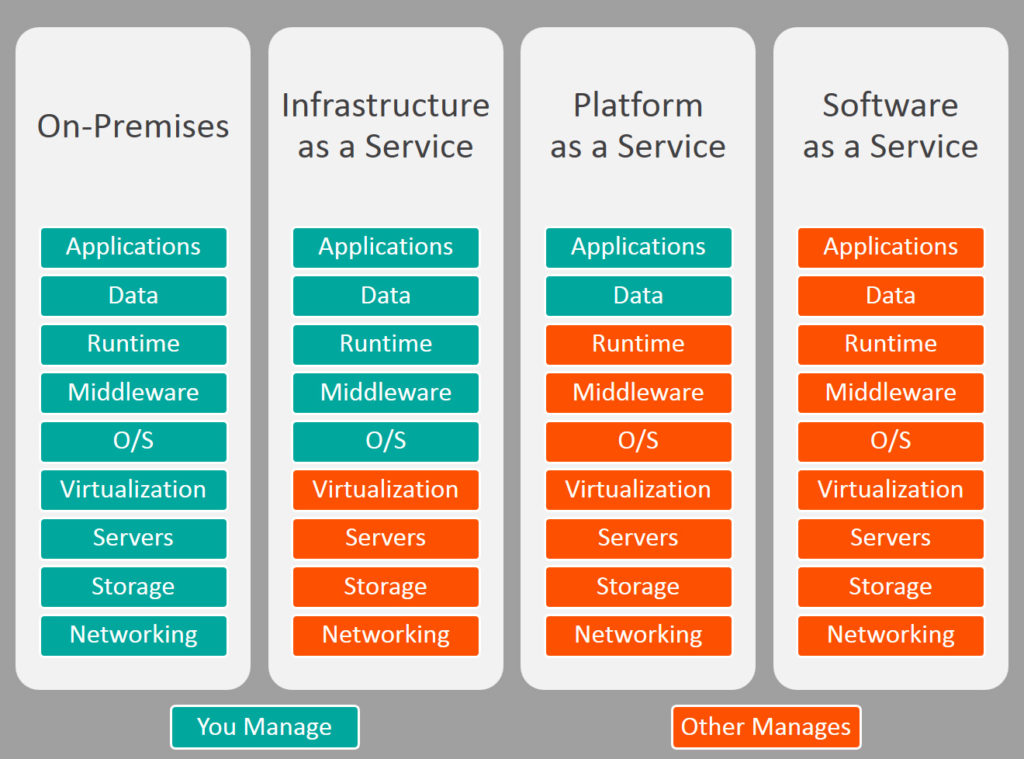




In this case, the developers don’t purchase the IaaS but purchase PaaS and the infrastructure such as web servers, database servers, and development libraries comes along.

An analogy to cloud services





**Amazon’s PaaS: Elastic Beanstalk**

We can create customized Docker Container using elastic beanstalk on your dedicated environment in the AWS cloud.

**Steps to launch an Environment**

Services > elastic beanstalk > create an environment > select one option either Web Server Environment or Worker Environment > select environment type (platform such as Docker and load balancing, auto scaling) > Application Version and Deployment Preferences > Environment Information > Additional Resources > Configuration Details (define the key pair) > Environment tags > Permissions > Review > Launch