Cloud Computing (ECS781P)

Week 2: Virtualization, Mechanisms

Dr Arman Khouzani, Dr Felix Cuadrado

Electronic Engineering and Computer Science
Queen Mary University of London

January 19, 2018

Virtualisation

Virtualisation

Virtualization is a core enabling technology for many different cloud computing services. So we dig into it a bit deeper.

- *virtualization* is not a “new” concept, or practice! Has been around since 70s!

- 1950s, the era of **mainframes**[1]
  - big schools and corporations.
  - installed in a server “room” (literally!)
  - multiple users could access it via “dumb terminals”
  - main motivation was its huge capex and opex costs

[1] http://blog.softlayer.com/tag/mainframe

Virtualisation

- In the 1970s, IBM released VM OS that allowed admins to have multiple virtual systems on their mainframes
- it allowed multiple distinct compute environments to live in the same physical environment.
- Every VM could run custom operating systems or guest operating systems that had their “own” memory, CPU, and hard drives along with CD-ROMs, keyboards and networking, despite the fact that all of those resources would be shared.

Virtualisation

- In the 1990s, telecommunications companies started offering “virtualized private network” connections with the same service quality as their dedicated services at a reduced cost.
- Shared access to the same physical infrastructure instead of allocating dedicated point-to-point infrastructure It also allowed shifting traffic as necessary for better network balance and more control over bandwidth usage.

Virtualisation

- Virtualisation was also adopted by many IT managers:
  - Efficient resource utilization and increased ROI;
  - Ease of administration;
  - Green IT support;
  - Better disaster recovery, . . .

Virtualisation

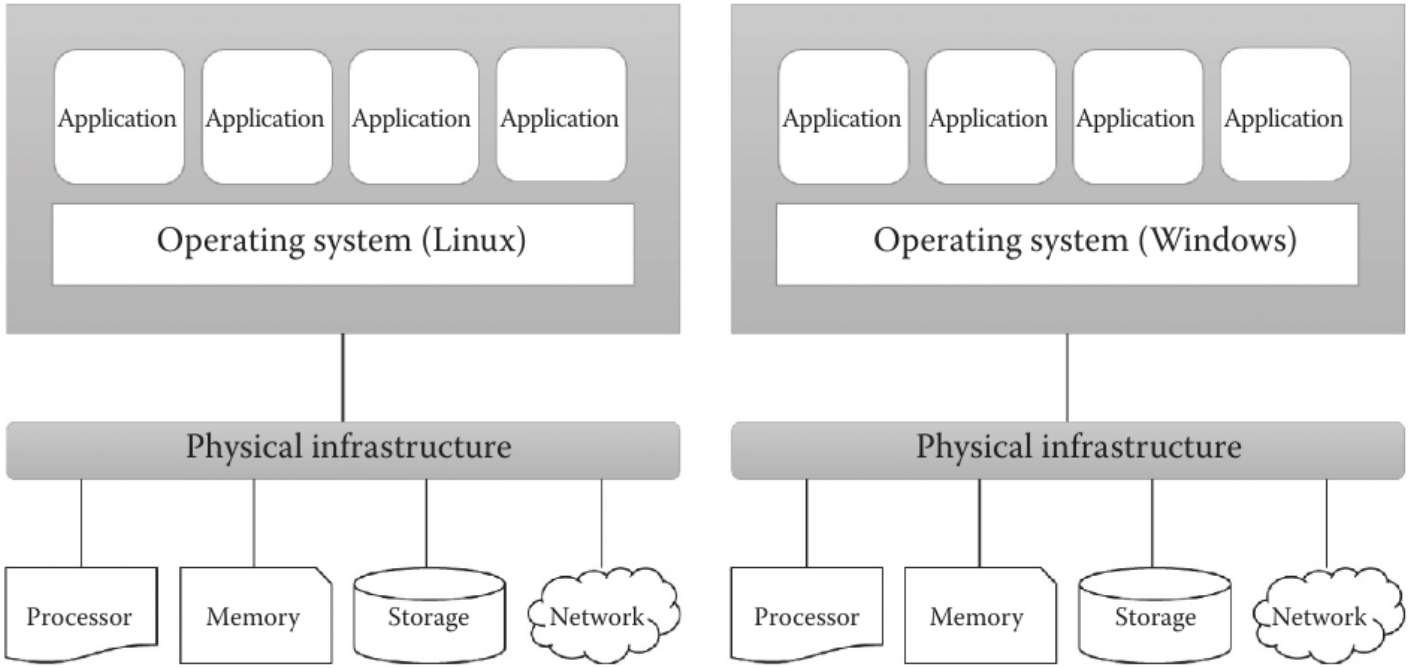
- A new problem arised: One server is not enough to provide the necessary resources I need
- solution: let's combine them, and emulate them as if they combined resources are in a single physical node.
- that was the birth of "cloud computing": the virtualized pooled resource can now be sliced up as needed and used like a "utility".
- Adding resources to this "cloud" is easy: just buy another server and and put it in the rack, install the hypervisor on it, it's now part of the cloud!

Before Virtualisation

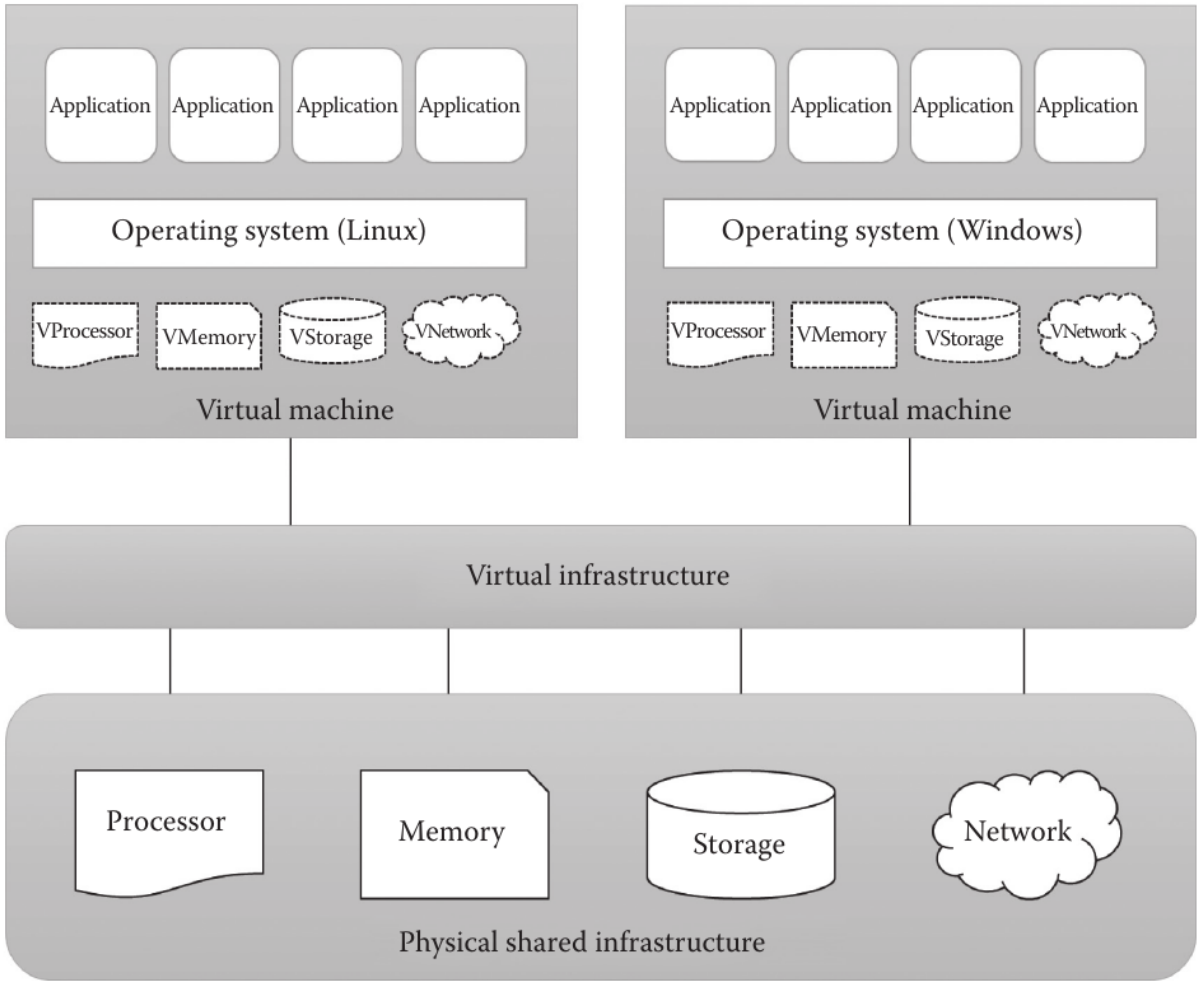
Before virtualization, the single physical infrastructure would run a single OS and its applications

- The organizations has buy a new hardware to meet an additional computing need. $\rightarrow$ underutilization of resources, bad ROI.

No Virtualisation



Using Virtualisation



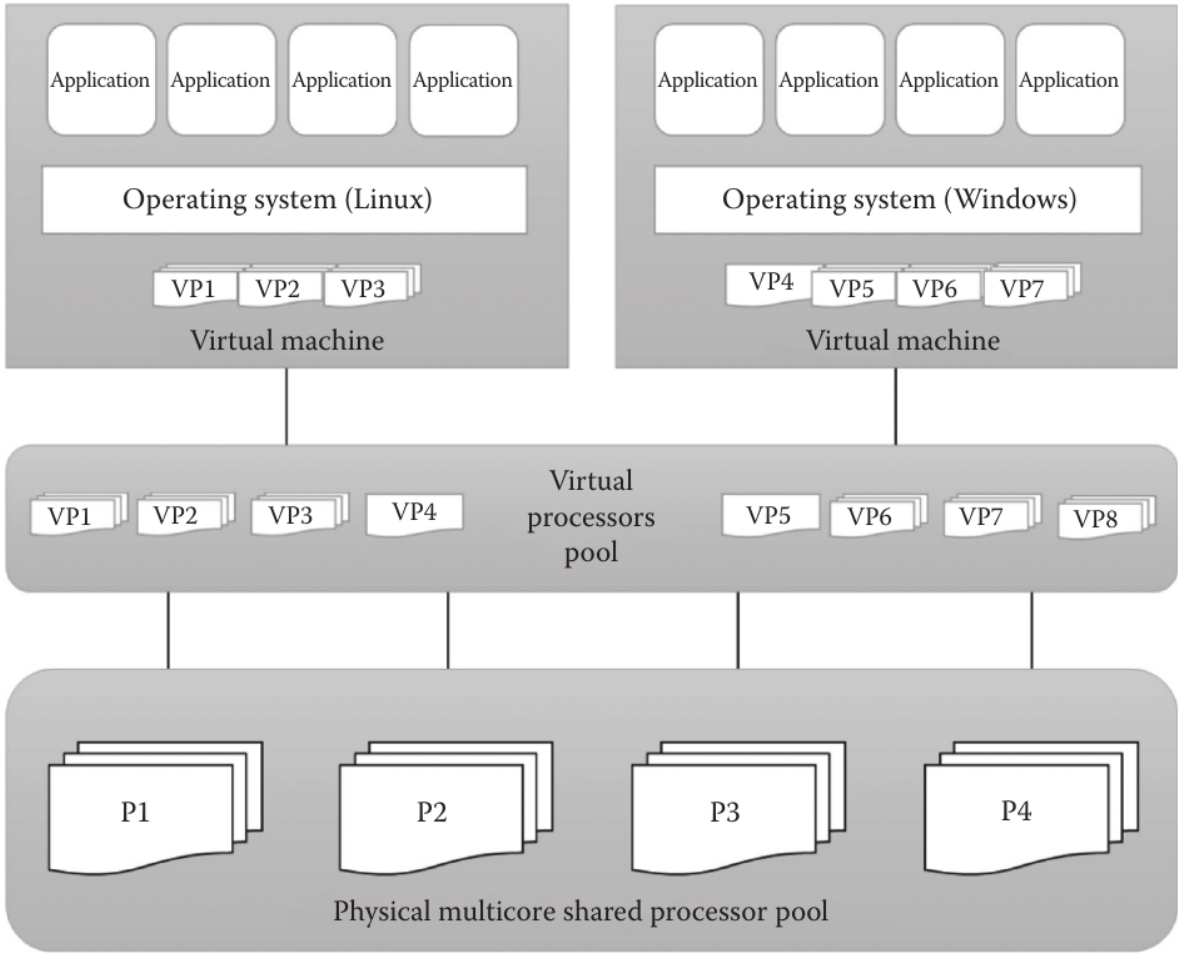
Using virtualisation, different OSs and applications can share a single physical infrastructure.

virtualisation

Virtualization “abstracts” the physical resources to a pool of virtual resources that can then be segmented as wished.

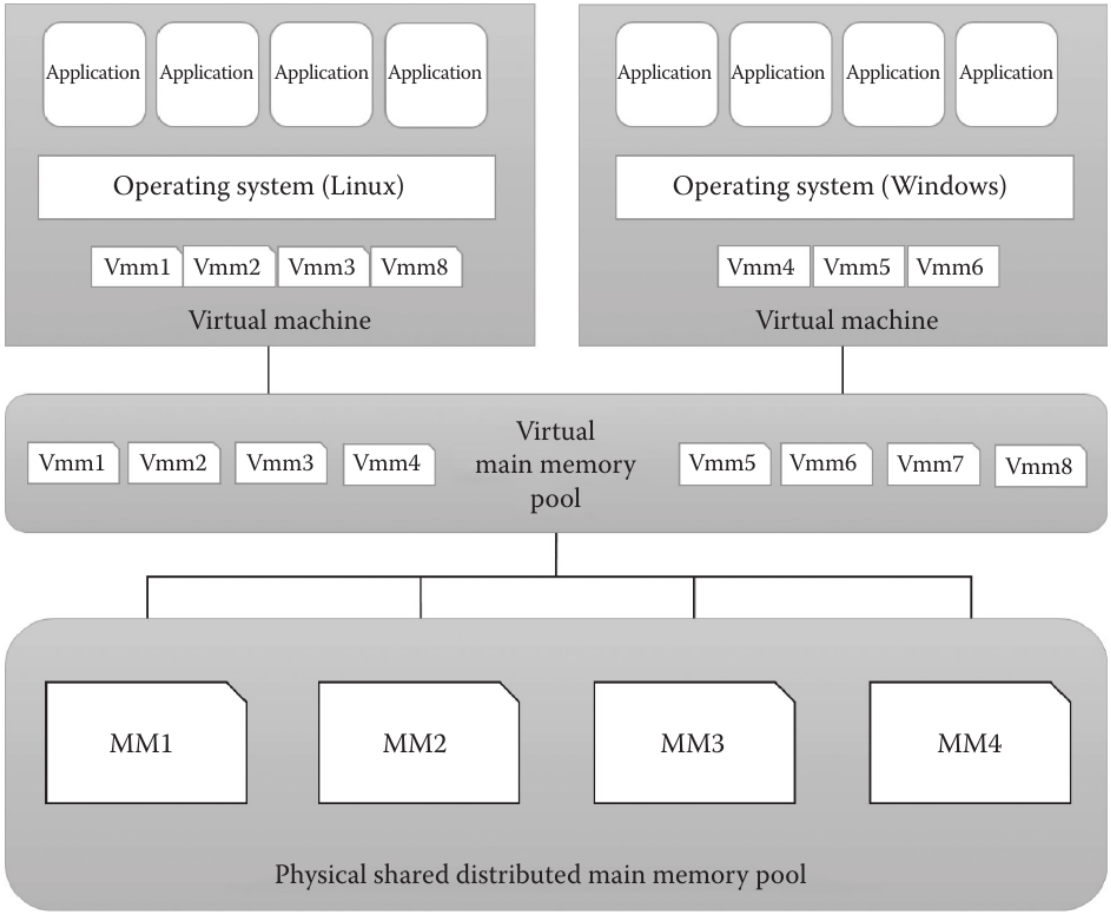
- Any resource can be virtualized with appropriate virtulization technology: memory, processors, storage, network, . . .

Processor Virtualisation



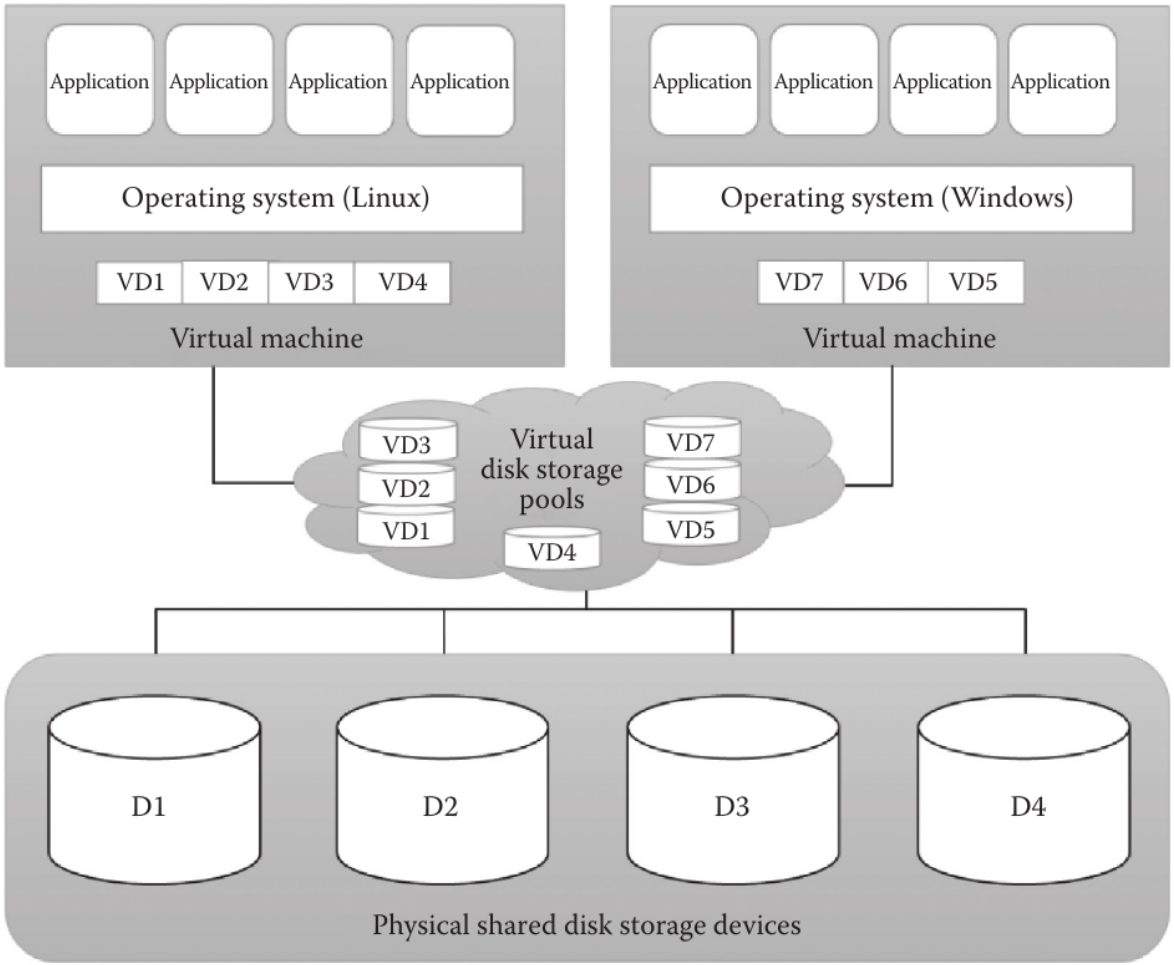
Virtual processors that are abstracted from the physical processors available at the underlying infrastructure can be shared among VMs.

(Main) Memory Virtualisation



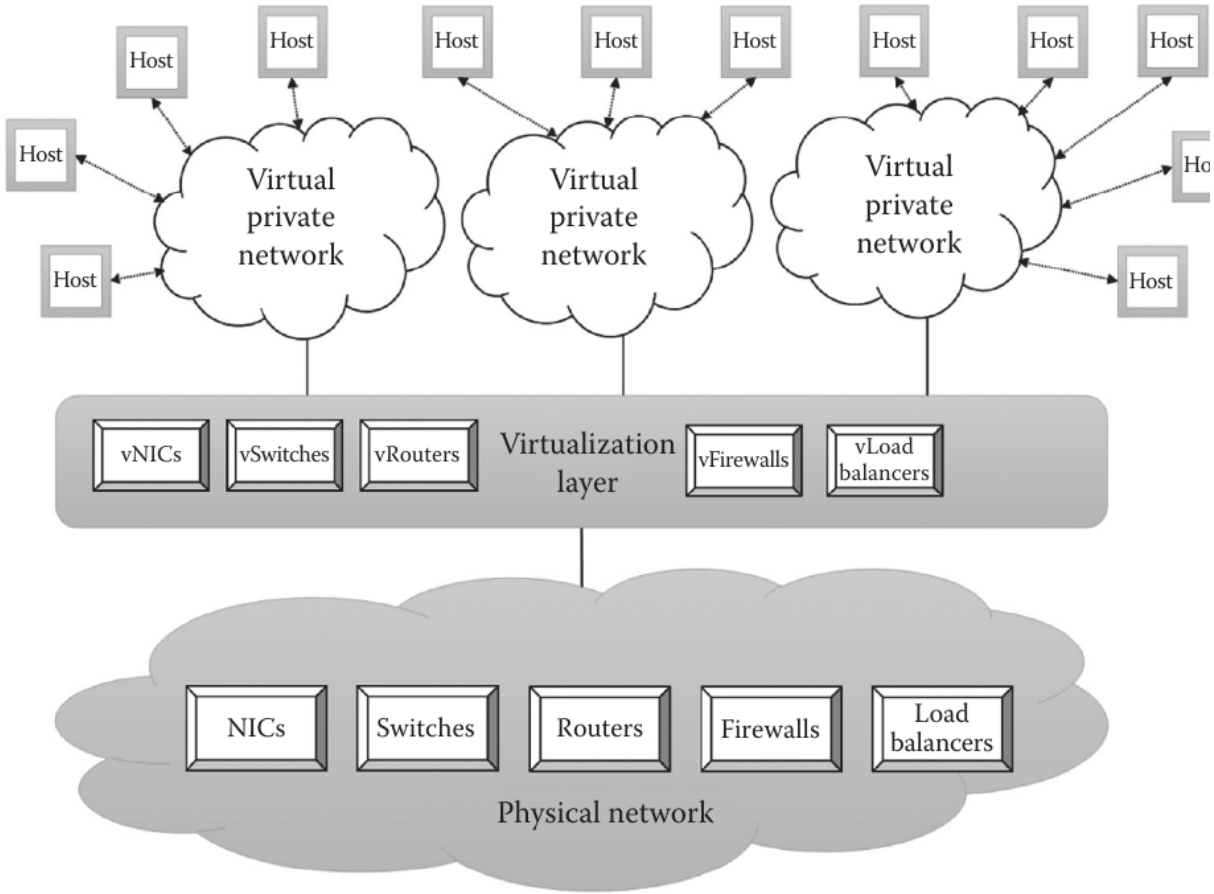
The physical main memory is mapped to the virtual main memory as in the virtual memory concepts in most of the OSs. The main idea of main memory virtualization is to map the virtual page numbers to the physical page numbers.

Storage Virtualisation



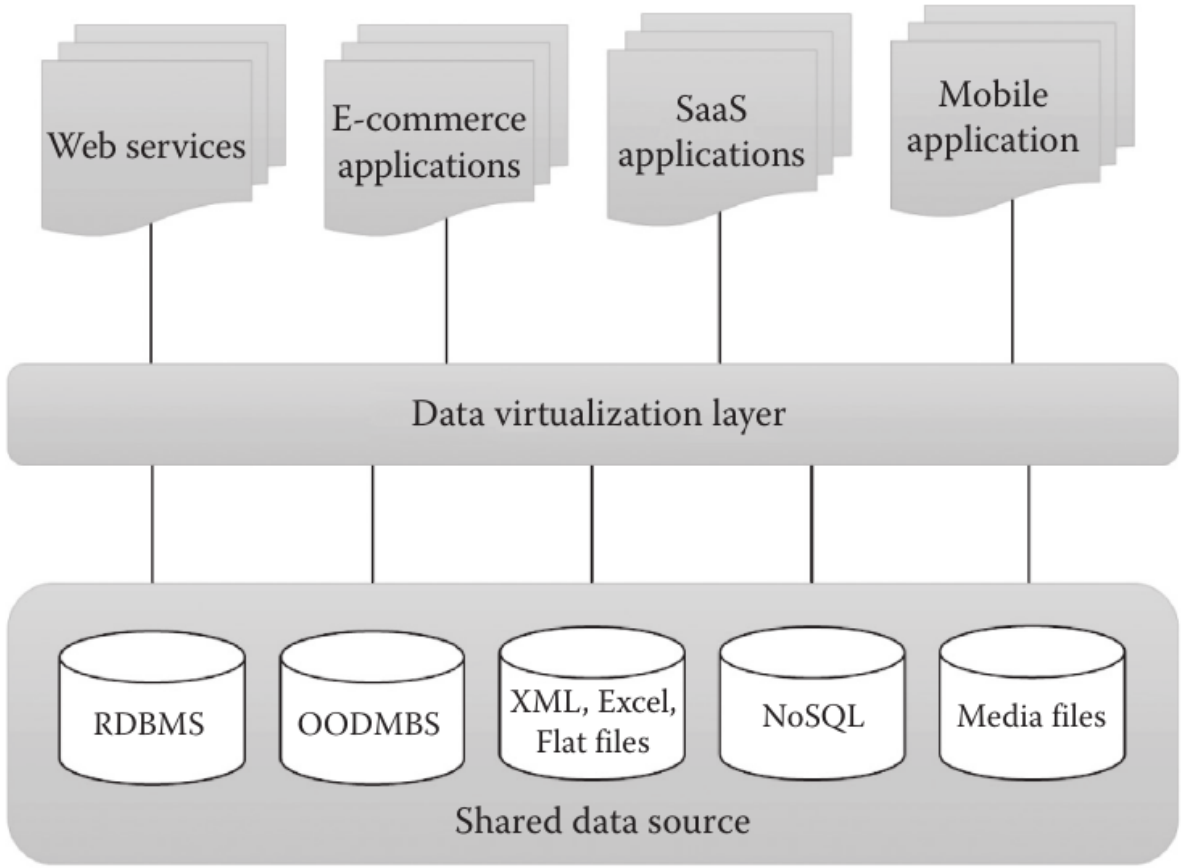
multiple physical storage disks are abstracted as a pool of virtual storage disks (logical storage). Also extensively used by SANs and NAS.

Network Virtualisation



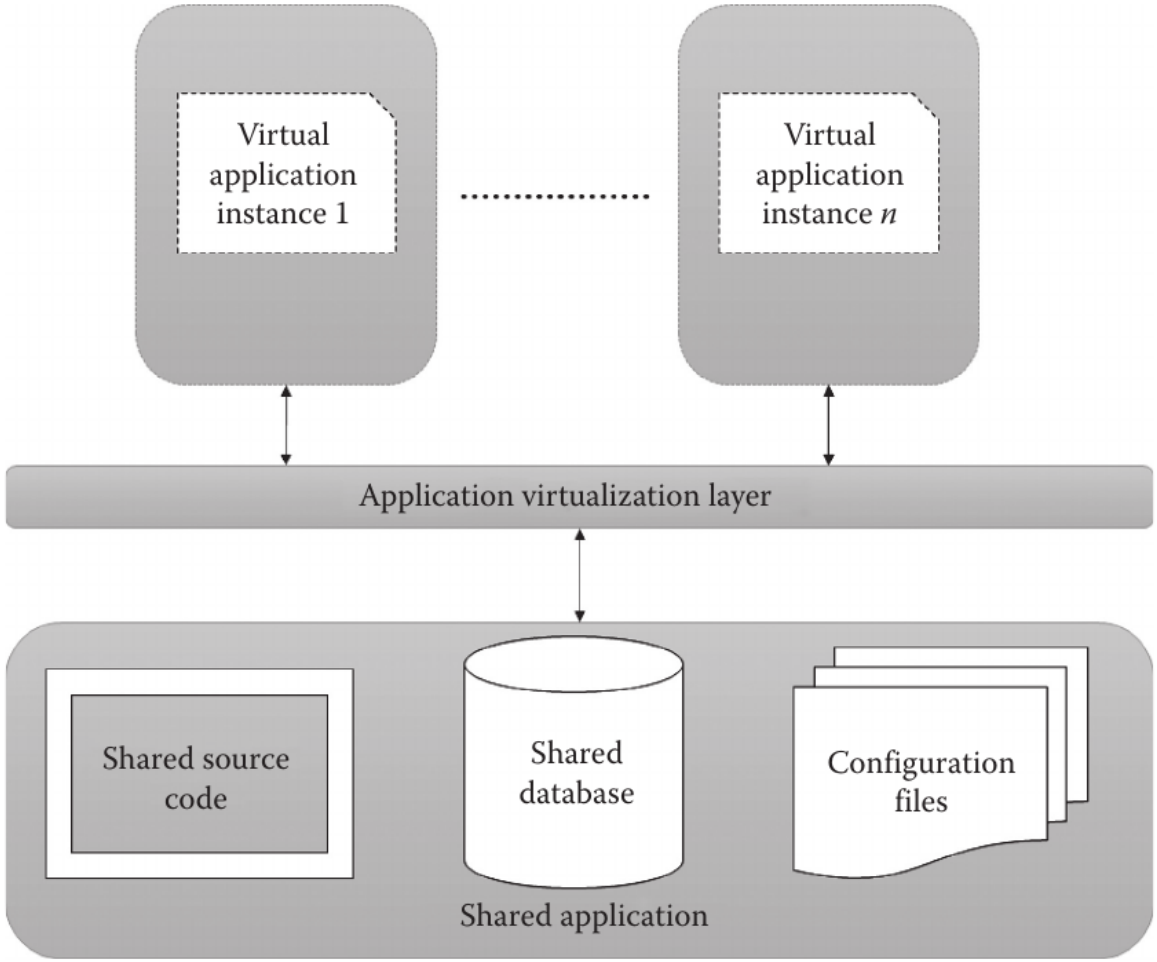
the physical network can be abstracted to create a virtual network. This involves router, switch, and Network Interface Card (NIC) being controlled by the virtualising software.

Data Virtualisation



Data virtualization hides the type of the data and the location of the data for the application that access it. It also ensures the single point access to data by aggregating data from different sources.

Application Virtualisation



The application virtualization offers the ability to the user to use the application without the need to install any software or tools in the machine (for SaaS services)

Cloud Mechanisms

Cloud Mechanisms

- Cloud mechanisms are the building blocks of how cloud services are built.
  - Each mechanism serves a well-defined cloud-related functionality.
  - Specific cloud services are built by putting these mechanisms together depending on the service, as per an “architecture”.
  - An architecture describes which mechanisms should be combined, how should each behave, and how they interact with each other.

Cloud Mechanisms: Logical Network Perimeter

- **Logical Network Perimeter:** establishes a virtual network boundary that can encompass and isolate a group of related cloud-based IT resources that may be physically distributed. This mechanism:
  - isolates IT-resources in a cloud from non-authorized users;
    - $\sim$ from non-users;
    - $\sim$ from cloud consumers;
    - also controls the bandwidth available to isolated resources.

Cloud Mechanisms: Logical Network Perimeter

- logical network perimeters are typically established via network devices that supply and control connectivity, and are usually deployed as virtualized. Examples are:
  - *Virtual Firewall*: actively filters network traffic to and from the a section of the infrastructure that it isolates.
    - *Virtual Networks* further isolates the network environments within the isolated infrastructure e.g. allowing only certain links between components (virtual links) e.t.c.
    - implemented usually through VLAN methods, or by the hypervisor itself.

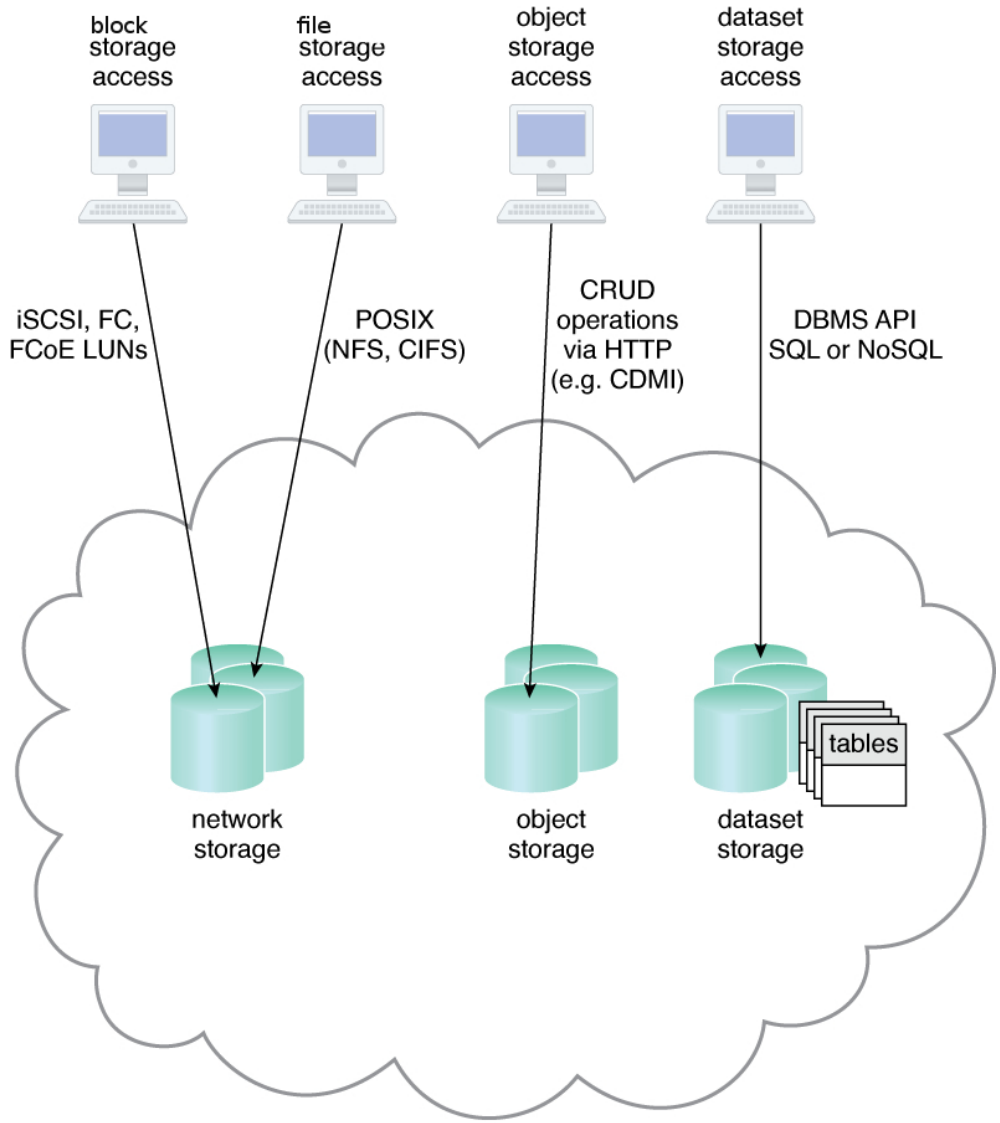
Cloud Mechanisms: Cloud Storage Device

- Storage units that are specialised for cloud-based provisioning.
  - typically, instances are virtualized and, similar to virtual servers, spawned as needed.
    - Typically are able to provice fixed increment capacity allocation (according to the pay-per-use mechanism)
    - A primary concern is security, integrity, and confidentiality of data (CIA).
    - there can also be legal and regulatory implications regarding the geographical relation of data outside of a region (e.g. a nation's boundary)
    - Performance is also a concern when dealing with large databases
    - This is both in terms of the elasticity, as well as latency and bandwidth

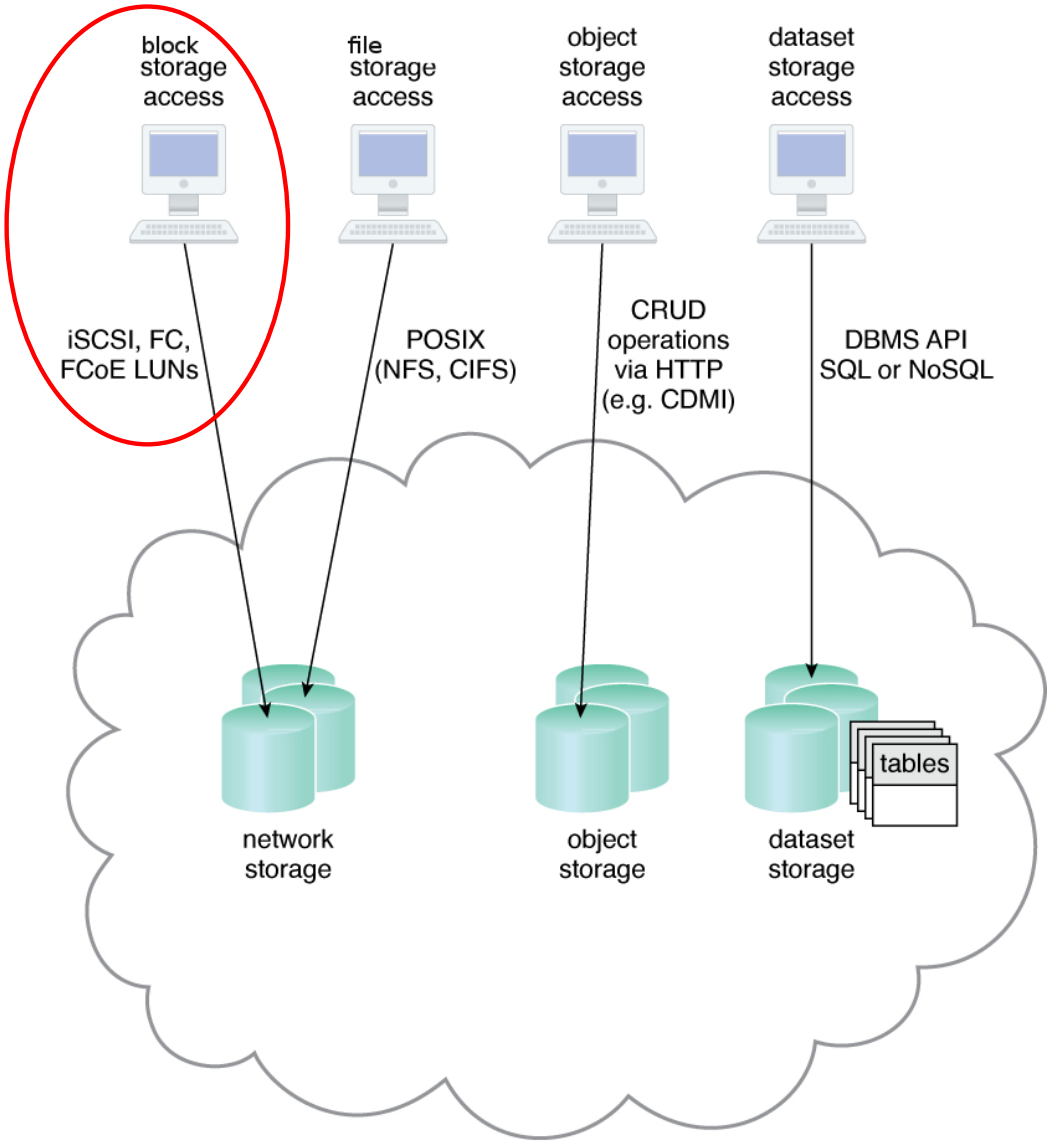
Cloud Mechanisms: Cloud Storage Device

- Cloud storage service allows remote access to a cloud storage device
  - Storage level can be Files, Blocks, Datasets, or Objects.
  - Each of them are associated with a certain type of technical interface.

Cloud Mechanisms: Cloud Storage Device



Cloud Mechanisms: Cloud Storage Device



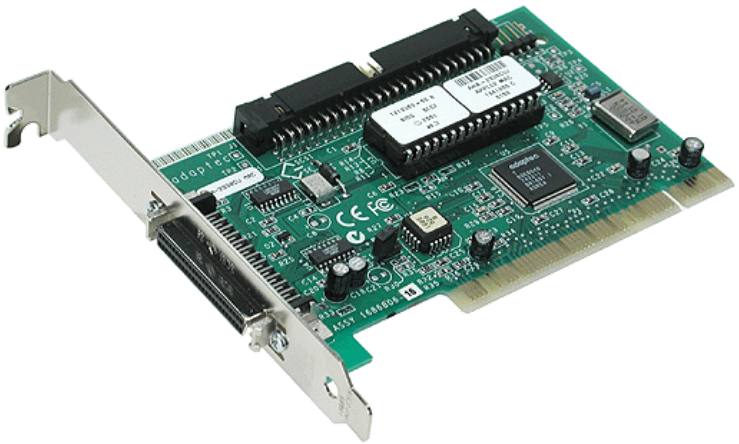
Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks**
  - A *block* is the smallest amount of data (as an addressable set of bits) that can be accessed (for reading/writing) from storage or memory.
    - The exact size varies depending on CPU/OS/disk/memory type, or application.
    - This is the lowest level of access to data (the blocks can be directly accessed without a need for a file system or a database, etc. in place), and is conceptually closest to the hardware.
    - They just provide the bare minimum functionality: e.g. `check the status of the device, read, write` (and that's almost it!)
    - So they are very *flexible* and provide *better performance*, but flexibility means that the application has to take care of everything else.

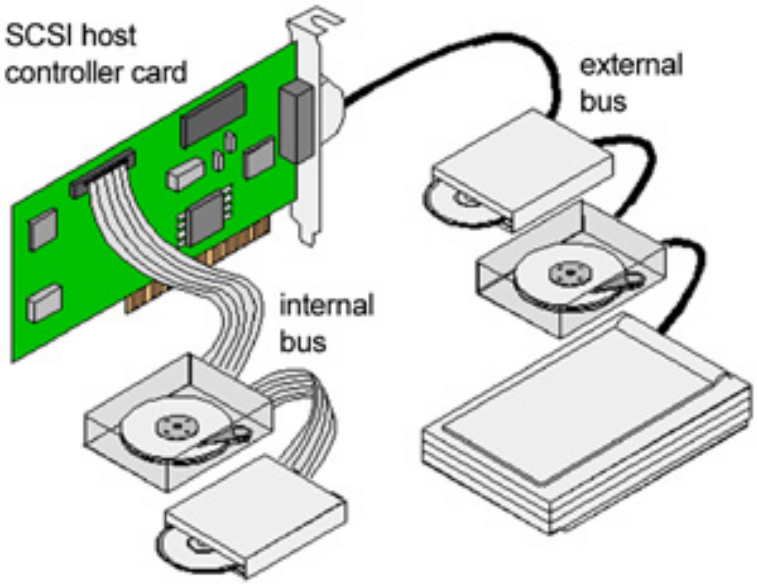
Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks** (continue. . .)
  - The main interface protocol for “block” level access to data is **SCSI**
    - stands for *Small Computer System Interface*
    - Has been in use since the early 1980’s
    - Within a single server, the OS handles writing data using the SCSI protocol to a *SCSI drive controller* which manages one or more devices on a SCSI cable within the chassis.
    - The *SCSI controller* ensures that only one device is active on the cable (*SCSI Bus*) at any time preventing *contention* and *data loss*.
    - In a networked protocol, preventing contention and data loss is more challenging.

Cloud Mechanisms: Cloud Storage Device



SCSI Controller



SCSI bus and “chain”

Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks** (continue. . .)
  - SCSI uses a low-voltage differential bus: comparing the difference in voltage between a pair of wires represent the data.
    - $\Rightarrow$ its speed and cable lengths are limited The first version was SCSI-1, in 1986: 8-bit, 5 MBps transfer rate, up to eight devices, maximum cable length of six meters
    - The latest version is Ultra-640 SCSI, in 2003: 16-bit, 640 MBps transfer rate, up to 16 devices, with a 12 meter cable length.

Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks** (continue. . .)
  - A serial encapsulation of SCSI is *Serial-attached SCSI* (**SAS**). Note: (serial: one bit at a time)
    - Advantage over (parallel) SCI: Thinner cables and less bulky the connectors, less crosstalk effect (because of electromagnetic interference) which means can be used over longer distance (longer cables), cheaper adapters.
    - It is compatible with **SATA** and backward compatible with (parallel) SCSI.
    - Supports up to 128 point-to-point connections, and up to 12 Gbps transfer rate.
    - The dominant interface for accessing direct-attached-storage devices (like external hard drives and magnetic tape drives) for large-scale storage, backup, archiving, etc.

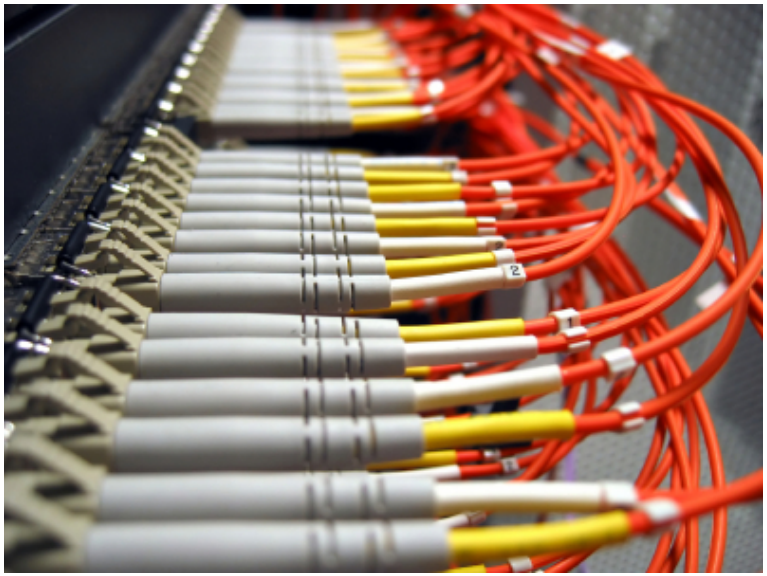
Cloud Mechanisms: Cloud Storage Device

(Mini) SAS cables.

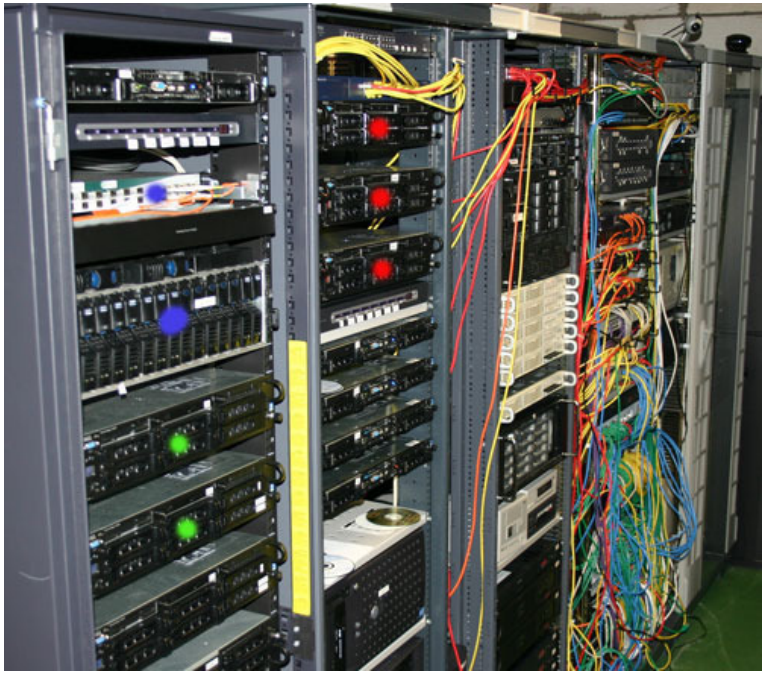
Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks** (continue. . .)
  - **Fibre Channel (FC)**: extended the functionality of SCSI to point-to-point, loop, switched topologies.
    - Typically carried over fibre-optic links, providing high bandwidth and low-latency.
    - Data rate of up to 128 Gbps (& full duplex)!
    - Devices can be up to 10 km apart (if optical fibre)
    - It is replacing the SCSI as the transmission interface between servers and clustered storage devices, in Storage Area Networks (SANs) in commercial data centres.
    - Note: Fibre Channel (FC) can work with other physical mediums like coaxial (or twisted) cable!

Cloud Mechanisms: Cloud Storage Device



Optical fibre connections.



Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks** (continue. . .)
  - **internet/IP Small Computer System Interface (iSCSI)**: encapsulation of SCSI into IP packets, allowing SCSI to be extended across existing IP infrastructures (which “routable”) with minimal additional cost.
    - One problem with iSCSI from a protocol standpoint is that it takes the SCSI protocol which expects lossless, in-order delivery, and places it in TCP/IP packets which are designed to support heterogeneous WAN networks and experience packet loss and out-of-order delivery frequently.[2]
    - Adoption has been patchy (compared to FC).

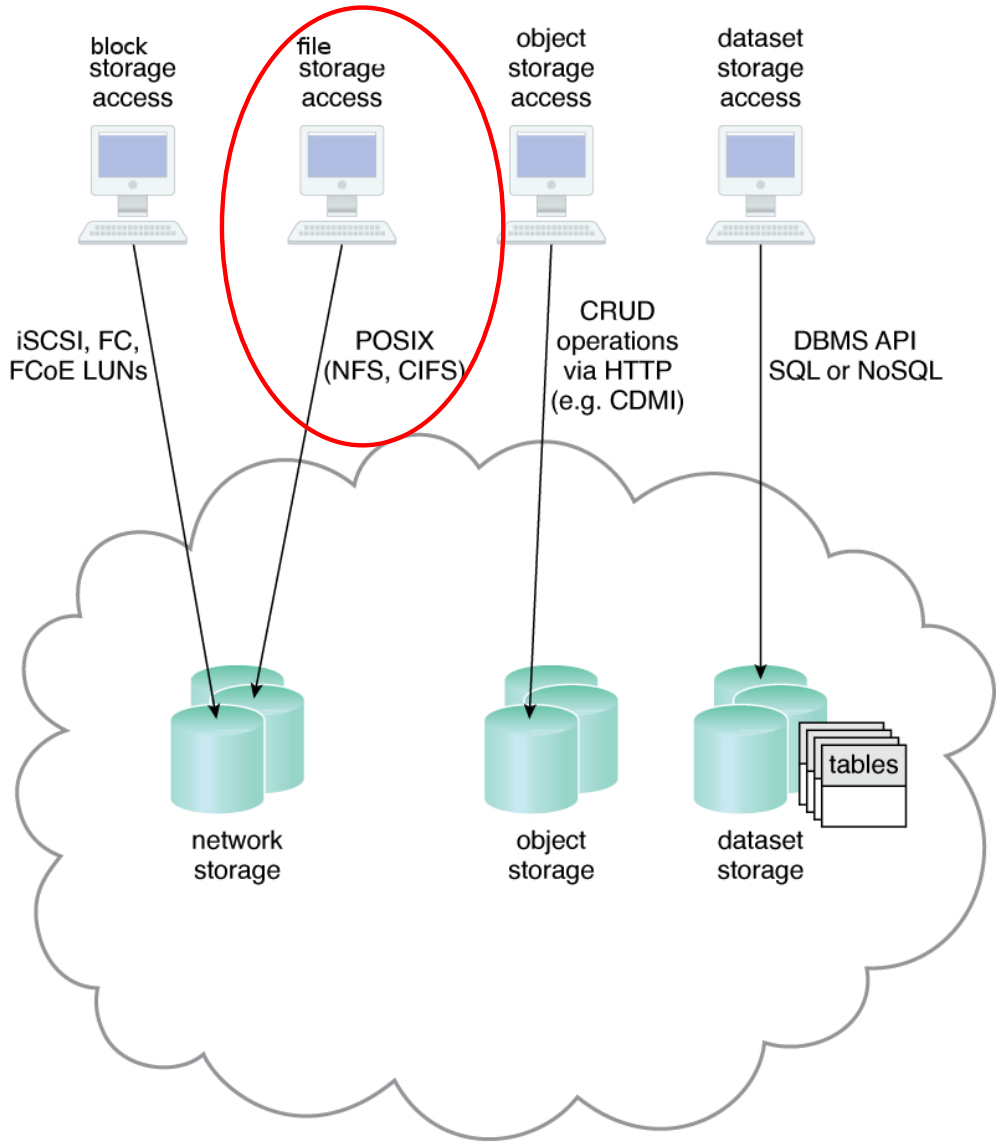
[2] http://www.definethecloud.net/tag/scsi/

Cloud Mechanisms: Cloud Storage Device

- 1- Storage level: **Blocks** (continue. . .)
  - **Fibre Channel over Ethernet (FCoE)**: provides the functionality for moving native Fibre Channel across consolidated Ethernet networks (encapsulates full Fibre Channel frames inside Ethernet Jumbo Frame payloads)
    - FCoE does not modify the existing Fibre Channel protocol suite and allows for the same management model including zoning, LUN[3] masking, etc.
    - Other notable Block-level interfaces over network (Storage Area Network) are: **ATA over Ethernet (AoE)**, **HyperSCSI**, **InfiniBand (IB)**.

[3]LUN: Logical Unit Number: a number used to identify a logical unit, which is a device addressed by the SCSI protocol or Storage Area Network protocols which encapsulate SCSI, e.g. FC or iSCSI

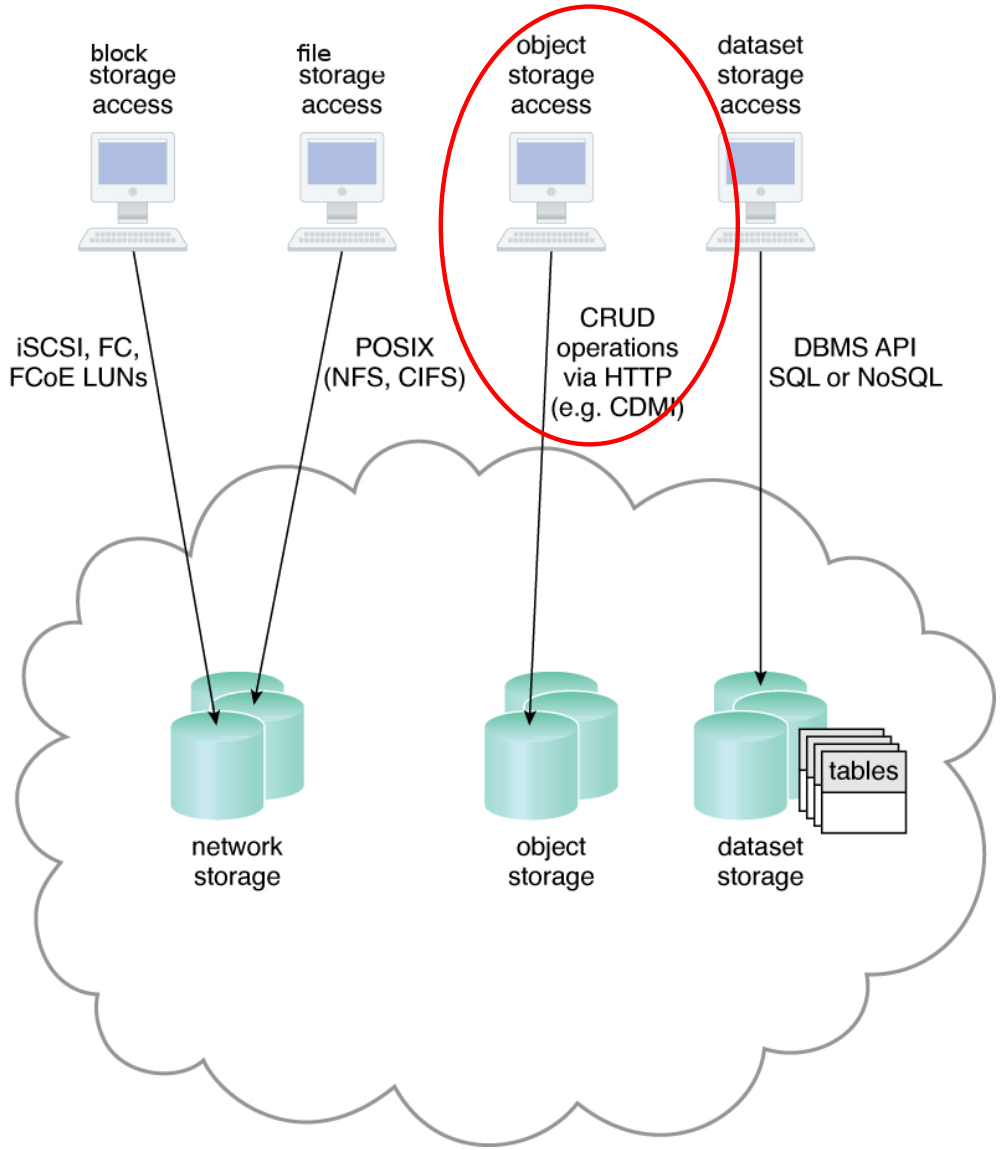
Cloud Mechanisms: Cloud Storage Device



Cloud Mechanisms: Cloud Storage Device

- 2- Storage level: **Files**
  - Collections of data are grouped into files that are located in folders.
    - File storage entails storing individual data in separate files that can be different sizes and formats and organized into folders and subfolders. Original files are often replaced by the new files that are created when data has been modified.
    - Most well-known interfaces at file level for networked storage devices are:
      - **Server Message Block (SMB)**
        - **Common Internet File System (CIFS)**
        - **Network File System (NFS)**

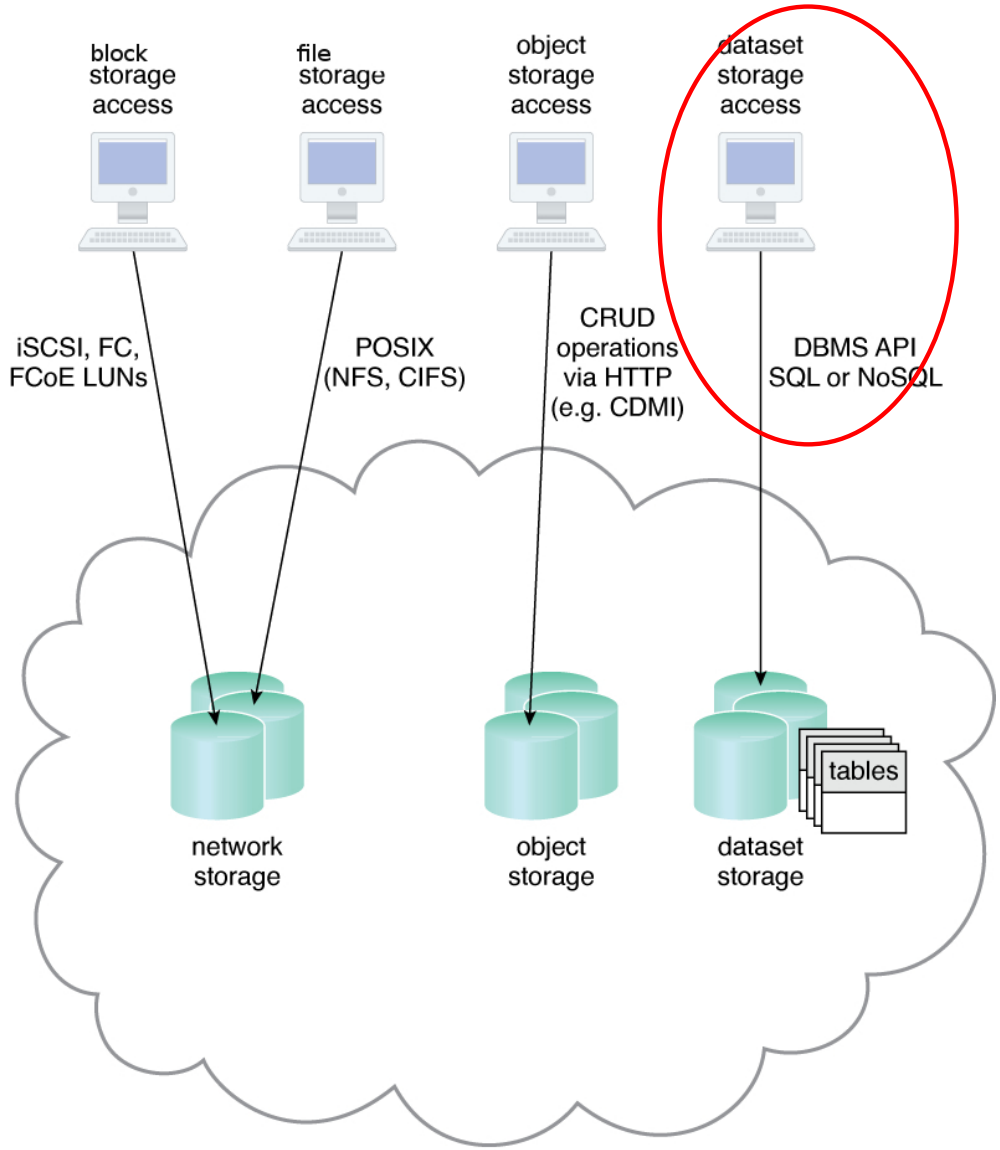
Cloud Mechanisms: Cloud Storage Device



Cloud Mechanisms: Cloud Storage Device

- 3- Storage level: **Objects**
  - Data and its associated metadata are organizes as Web-based resources.
    - The Create, Read, Update and Delete (**CRUD**) operations are typically done over the HTTP.
    - Example of an interface for object-level cloud storage device is **CDMI** (Cloud Data Management Interface) standard
    - Besides CRUD operations, CDMI provides managing domains, users and groups, implementing access control, attaching metadata, making arbitrary queries, using persistent queues, using a logging facility, billing, moving data between cloud systems, and exporting data via other protocols such as iSCSI and NFS.

Cloud Mechanisms: Cloud Storage Device



Cloud Mechanisms: Cloud Storage Device

- 4- Storage level: **Database Storage**
  - Sets of data are organized into a table-based, delimited, or record format.
    - Cloud storage device mechanisms based on database storage interfaces typically support a query language in addition to basic storage operations.
    - They are divided into two main categories according to storage structure:
      - **Relational**
        - **Non-Relational**

Cloud Mechanisms: Cloud Storage Device

- 4- Storage level: **Database Storage**
  - **Relational Data Storage**
    - Relational databases rely on tables to organize similar data into rows and columns. Tables can have relationships with each other to give the data increased structure, to protect data integrity, and to avoid data redundancy (which is referred to as data normalization).
      - Working with relational storage typically involves using the Structured Query Language (SQL).
      - Examples of commercially available database products include IBM DB2, Oracle Database, Microsoft SQL Server, and MySQL.
      - Challenges with cloud-based relational databases pertain to scaling and performance.

Cloud Mechanisms: Cloud Storage Device

- 4- Storage level: **Database Storage**
  - **Non-Relational Data Storage** (a.k.a. NoSQL storage) establishes a “looser” structure for stored data with less emphasis on defining relationships and realizing data normalization.
    - The primary motivation is to avoid the potential complexity and processing overhead that can be imposed by relational databases.
    - The trade-off is that the data loses much of the native form and validation. Also some of relational database functions are typically not supported, e.g. transactions or joins.
    - Also, there is no or partial data normalization, so the storage efficiency is much lower.
    - Many non-relational storage mechanisms are proprietary and can severely limit data portability.

Cloud Mechanisms: Cloud Usage Monitor

- **cloud usage monitor** mechanism: a lightweight and autonomous software program responsible for *collecting and processing IT resource usage data*.
  - The collected usage data is sent to a *log database* for post-processing and reporting purposes.
    - Depending on the type of usage metrics they are designed to collect and the manner in which usage data needs to be collected, cloud usage monitors can exist in different formats:
      - **Monitoring Agent**
        - **Resource Agent**
        - **Polling Agent**

Cloud Mechanisms: Cloud Usage Monitor

- **Monitoring Agent**: an intermediary, event-driven program that exists as a service agent and resides along existing communication paths to transparently monitor and analyze dataflows
  - commonly used to measure network traffic and message metrics.

Cloud Mechanisms: Cloud Usage Monitor

- **Resource Agent**: a processing module that collects usage data by having event-driven interactions with specialized resource software
  - monitors usage metrics based on pre-defined, observable events at the resource software level, such as initiating, suspending, resuming, and vertical scaling.

Cloud Mechanisms: Cloud Usage Monitor

- **Polling Agent**: a processing module that collects cloud service usage data by polling IT resources.
  - used to periodically monitor IT resource status, such as uptime and downtime.

Cloud Mechanisms: Resource Replication

- **Resource Replication**: the creation of multiple instances of the same IT resource.
  - Typically performed when an IT resource's availability and performance need to be enhanced
    - Virtualization technology is used to implement the resource replication mechanism to replicate cloud-based IT resources
    - commonly implemented as a hypervisor, that accesses a virtual server image to create several instances, or to deploy and replicate ready-made environments and entire applications.

Cloud Mechanisms: Ready-Made Environment

- **Ready-Made Environment**: a defining component of the PaaS cloud delivery model that represents a pre-defined, cloud-based platform comprised of a set of already installed IT resources, ready to be used and customized by a cloud consumer.
  - These environments are utilized by cloud consumers to remotely develop and deploy their own services and applications within a cloud.
  - Typical ready-made environments include pre-installed IT resources, such as databases, middleware, development, and governance tools.

Cloud Mechanisms: Automated Scaling Listener

- **Automated Scaling Listener**: a service agent that monitors and tracks communications between cloud service consumers and cloud services to track workload status information for dynamic scaling.
  - Workloads can be determined by the volume of consumer-generated requests or via back-end processing demands triggered by certain types of requests (a small amount of incoming data can result in a large amount of processing).
  - Responses to workload fluctuation include:
    - Automatically scaling IT resources out or in based on parameters previously defined by the cloud consumer (auto-scaling).
    - Sending automatic notification to the cloud consumer when workloads exceed current thresholds or fall below allocated resources.

Cloud Mechanisms: Load Balancer

- **Load Balancer**: a runtime agent to balance a workload across multiple IT resources to increase performance and capacity.
  - Load balancer is programmed with a set of performance and Quality-of-Service (QoS) rules and parameters with the general objectives of optimizing IT resource usage, avoiding overloads, and maximizing throughput.
  - Some specialised workload distribution functions:
    - *Asymmetric Distribution* – larger workloads are issued to higher capacity IT resources.
      - *Workload Prioritization* – workloads are scheduled, queued, discarded, and distributed according to their priority levels.
      - *Content-Aware Distribution* – assigning requests to different IT resources based on their content.

Cloud Mechanisms: SLA Monitor

- **SLA Monitor**: used to observe the runtime performance of cloud services to ensure that they meet the contractual Quality-of-Service (QoS) metrics requirements in the SLA.
  - The data collected by the SLA monitor is processed by an SLA management system to be aggregated into SLA reporting metrics.
    - This system can proactively repair or failover cloud services when exception conditions occur, such as when the SLA monitor reports a cloud service as “down”.

Cloud Mechanisms: Pay-Per-Use Monitor

- **Pay-Per-Use Monitor**: measures cloud-based IT resource usage (e.g., request/response message quantity, transmitted data volume, bandwidth consumption, etc.) in accordance with predefined pricing parameters and generates usage logs for fee calculations and billing purposes.
  - data collected by the pay-per-use monitor is processed by a billing management system that calculates the payment fees.

Cloud Mechanisms: Fail-over System

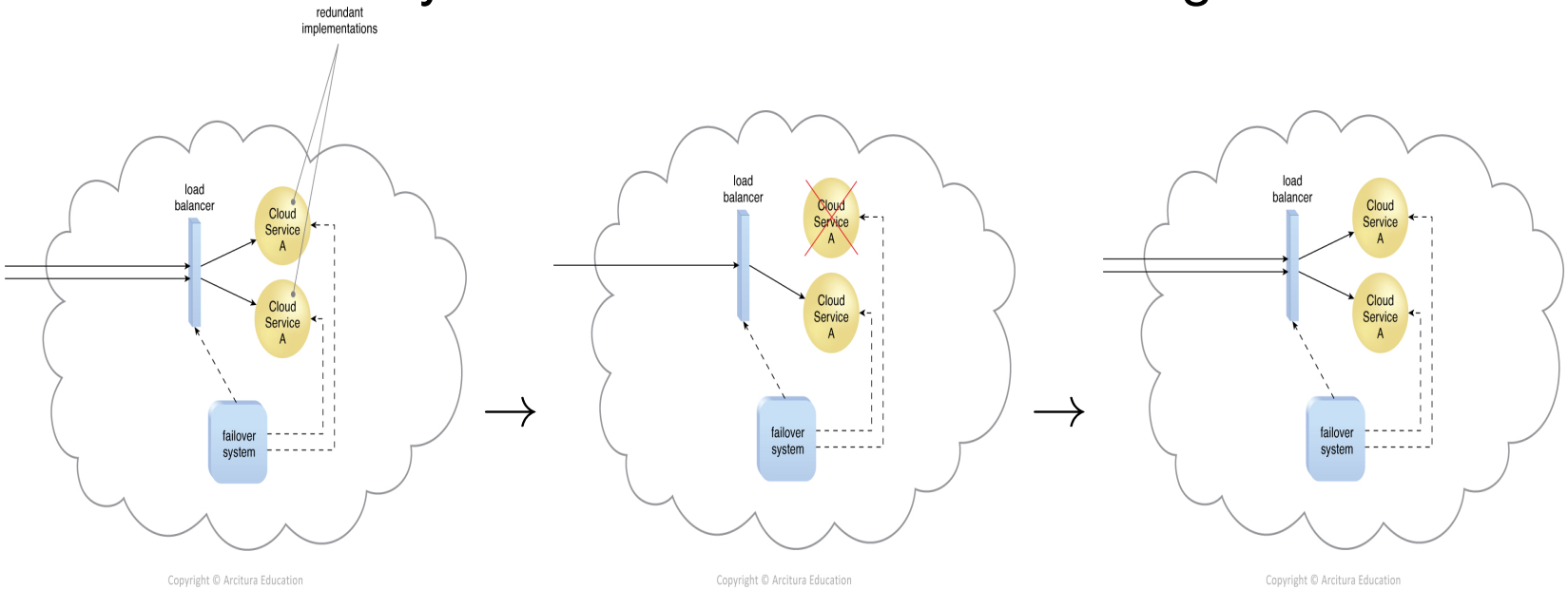
- **Fail-over System**: used to increase the reliability and availability of IT resources by using established clustering technology to provide redundant implementations.
  - A failover system is configured to automatically switch over to a redundant or standby IT resource instance whenever the currently active IT resource becomes unavailable.
    - Commonly used for mission-critical programs or for reusable services that can introduce a single point of failure for multiple applications.
    - Can span more than one geographical region so that each location hosts one or more redundant implementations of the same IT resource.

Cloud Mechanisms: Fail-over System

- Failover systems come in two basic configurations:
  - **Active-Active Configuration**: Redundant implementations of the IT resource actively serve the workload synchronously through a load-balancer. When a failure is detected, the failed instance is removed from the load balancing scheduler, and the workload is distributed among the remaining operational instances.
  - **Active-Passive Configuration**: A stand-by or inactive implementation is activated to take over the processing from the IT resource that becomes unavailable, and the corresponding workload is redirected to it.

Cloud Mechanisms: Fail-over System

Failover System with Active-Active Configuration:

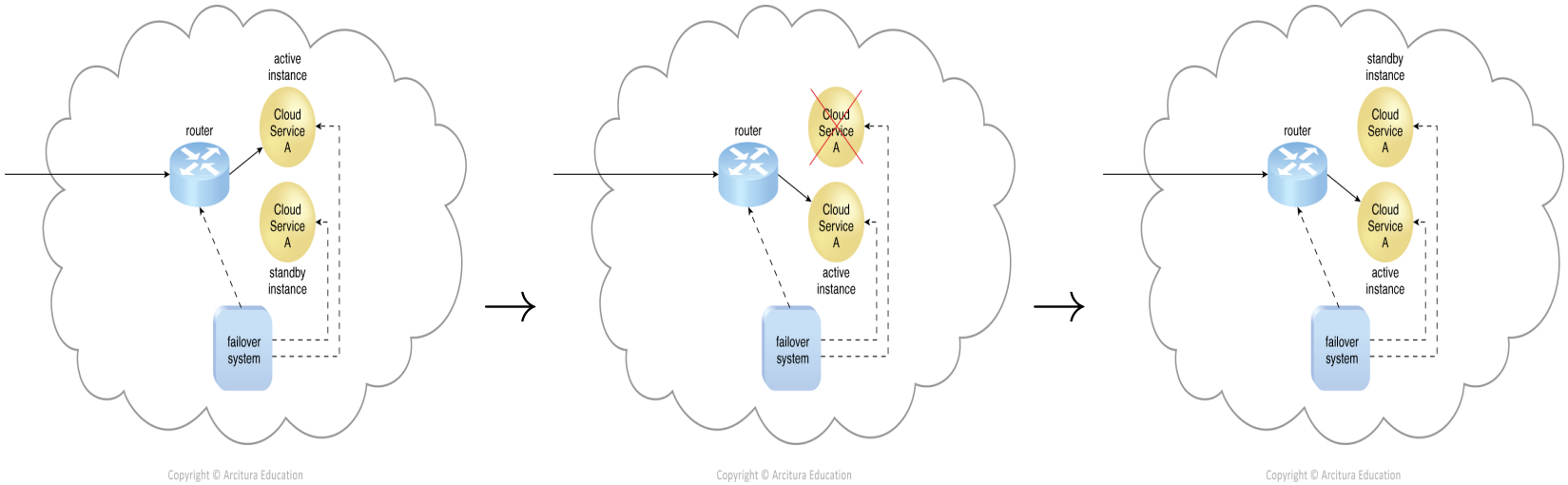


(A) (B) (C)

(A) The failover system monitors the operational status of the Cloud Service. (B) When a failure is detected, the failover system commands the load balancer to distribute the load to remaining implementations. (C) The failed cloud service implementation is recovered or replicated. The failover system

Cloud Mechanisms: Fail-over System

Failover System with Active-Passive Configuration:



(A') (B') (C')

(A') The (one) active instance receives the cloud service consumer requests. (B') The active instance encounters a failure that is detected by the failover system, which activates the inactive implementation and redirects the workload toward it. (C') The failed implementation is recovered or replicated

Cloud Mechanisms: Fail-over System

- For IT resources that do not require execution state management and provide stateless processing services, the failover system can be simply implemented by a load balancer that detects failure conditions and exclude failed IT resource instances from the workload distribution.
- If the service is not stateless, then the redundant or standby IT resource implementations are also required to share their state and execution context, so that the tasks on the failed resource can remain operational in one if the redundant implementations. Clustering and virtualization technologies exist to achieve this rather complex task.

Cloud Mechanisms: Hypervisor

- **Hypervisor**: a fundamental part of virtualization infrastructure that is primarily used to generate virtual server instances of a physical server concurrently running on the same (host) physical infrastructure.
  - Hypervisor provides features for controlling, sharing and scheduling the usage of hardware resources, e.g. processor power, memory, I/O.
    - These resources can appear to each virtual server's operating system as dedicated resources.
    - Importantly, the controlling/scheduling features cannot be accessed from inside the virtual machines!

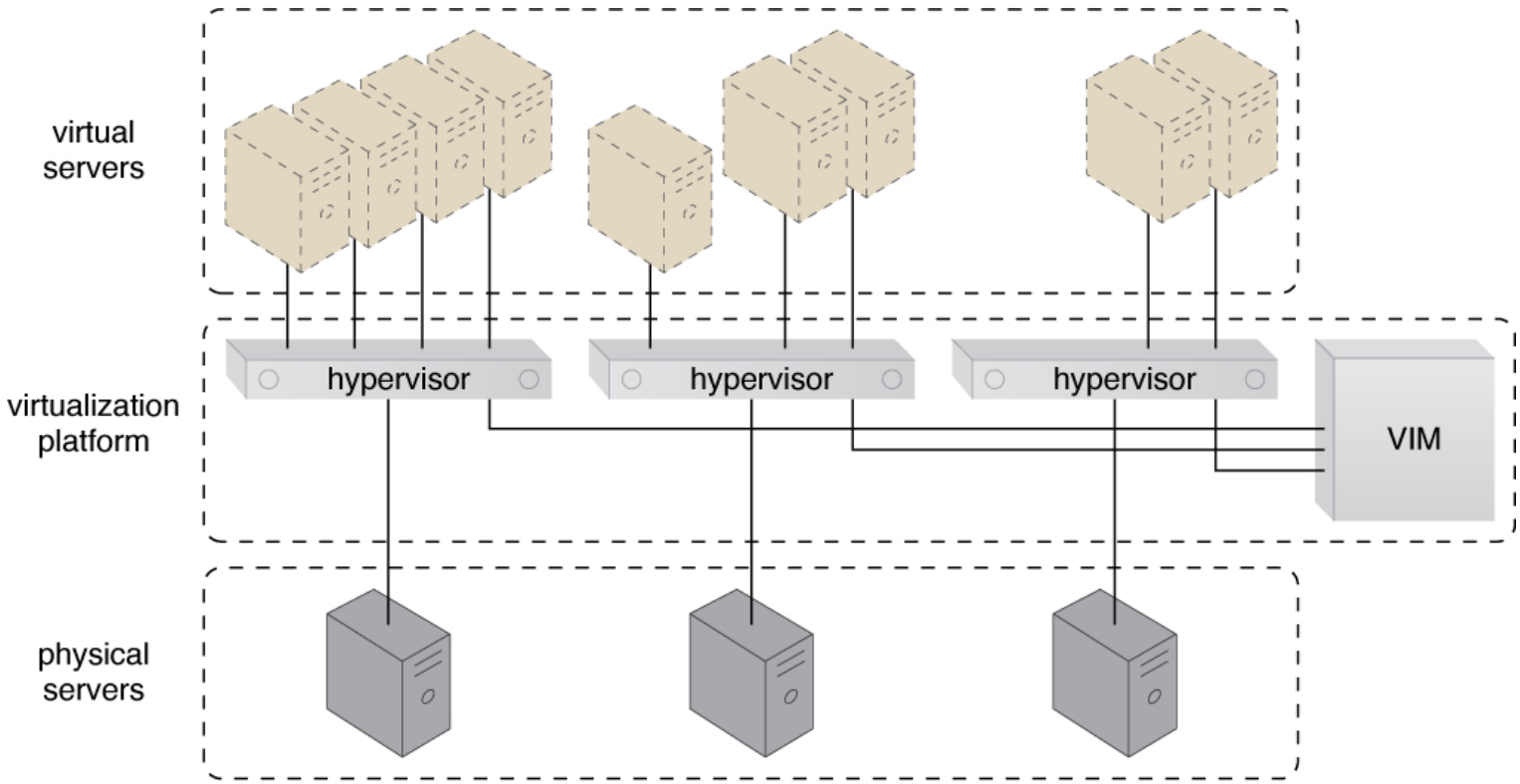
Cloud Mechanisms: Hypervisor

- There are two types of hypervisors:
  - **Type-1**: Runs directly on the hardware (also called "*native*", "*bare metal*" or "*embedded*", or "*hardware virtualization engine*".
    - **Type-2**: Runs on top of a host operating system.
- Type-1 hypervisors are more efficient: they need a lower overhead to run the hypervisor itself and no overhead will be wasted for running the operating system, therefore, almost all of the underlying resources can be devoted to virtual machines.
- As a rule-of-thumb, Type-1 hypervisors provide higher performance, availability, and security.

Cloud Mechanisms: VIM

- A hypervisor can only assign the virtual servers it generates to resource pools that reside on the same underlying physical server.
- A hypervisor has limited virtual server management features, e.g.: initiating a virtual server, increasing its capacity, shutting it down.
- The **virtualization infrastructure management (VIM)** provides a range of features for administering multiple hypervisors across physical servers.
  - The VIM coordinates the server hardware so that virtual server instances can be created from the most expedient underlying physical server.
    - The VIM can create and manage multiple instances of a hypervisor across different physical servers or move a virtual machine across physical servers.

Cloud Mechanisms: VIM



Each hypervisor can create multiple virtual machines on a physical server that it is running on. The hypervisors and their virtual machines are managed centrally by the virtualization infrastructure management (VIM).

Cloud Mechanisms: Resource Cluster

- **Resource Cluster**: used to group multiple IT resource instances so that they can be operated as a single IT resource.
  - This increases the combined computing capacity, load balancing, and availability of the clustered IT resources.
    - The IT resources can be geographically diverse, but are “logically” combined via the resource cluster mechanism.
    - Resource cluster architectures rely on high-speed dedicated network connections, or *cluster nodes*, between IT resource instances to communicate about workload distribution, task scheduling, data sharing, and system synchronization.

Cloud Mechanisms: Resource Cluster

- A cluster management platform that is running as distributed middleware in all of the cluster nodes is responsible for such activities (message passing for workload distribution, task scheduling, data sharing, and system synchronization)
- This *clustering middleware* is a software layer that sits on top of the *nodes* and *orchestrates* the activities of the computing nodes such that the whole cluster can effectively be treated and behave as one cohesive unit.

Cloud Mechanisms: Resource Cluster

- There are two basic types of resource clusters:
  - **Load Balanced Cluster**: specializes in distributing workloads among cluster nodes to increase IT resource capacity while preserving the centralization of IT resource management.
    - It usually implements a load balancer mechanism (either as part of the cluster management platform or set up separately).
  - **High-Availability (HA) Cluster**: maintains system availability in the event of multiple node failures, and has redundant implementations of most or all of the clustered IT resources.
    - It implements a failover system mechanism that monitors failure conditions and automatically redirects the workload away from failed nodes.

Cloud Mechanisms: Resource Cluster

- Common resource clusters include:
  - **Server Cluster**
    - **Database Cluster**
    - **Large Dataset Cluster**
- **Server Cluster**: Physical or virtual servers are clustered to increase performance and availability.
  - Hypervisors running on each physical server can be configured to share virtual server execution state (memory pages, processor register state, etc)
    - Then virtual servers can live-migrate: virtualization platform suspends the execution of a VM at one physical server and resumes it on another one.
    - This can be used to increase scalability by live-migrating a VM that is running at an overloaded physical server to one that has suitable capacity.

Cloud Mechanisms: Resource Cluster

- **Database Cluster**: Designed to improve data availability, this *high-availability* (HA) resource cluster has a synchronization feature that maintains the consistency of data being stored at different storage devices used in the cluster.
  - Implementation is usually based on an active-active or active-passive failover system with a mechanism that enforces the synchronization conditions.
- **Large Dataset Cluster**: A mechanism that efficiently performs data partitioning and distribution, such that data integrity and computing accuracy is maintained.

Cloud Mechanisms: Resource Cluster

- Many resource clusters require cluster nodes to have almost identical characteristics to simplify the design & maintain consistency within the cluster architecture.
- The cluster nodes in high-availability cluster architectures need to access and share common storage IT resources.
- This can require two layers of communication between the nodes: one for accessing the storage device, and another to execute IT resource orchestration.
- Some resource clusters have more loosely coupled IT resources that only require the network layer.

Cloud Mechanisms: Multi-Device Broker

- **Multi-Device Broker**: used to facilitate runtime data transformation so as to make a cloud service accessible by a wider range of cloud service consumer programs and devices.
  - A cloud service may need to be accessed by different types of cloud service consumers with distinct hosting hardware devices and/or different types of communication requirements, some of which may be incompatible with the cloud service's published service contract.
    - To overcome incompatibilities between a cloud service and a cloud service consumer, the information that is exchanged at runtime need to be transformed/converted, which is done by the multi-device broker mechanism.

Cloud Mechanisms: Multi-Device Broker

- Multi-device brokers commonly incorporate *gateway* components, e.g.:
  - *XML Gateway*: transmits and validates XML data.
  - *Cloud Storage Gateway*: transforms cloud storage protocols and encodes storage devices to facilitate data transfer and storage.
  - *Mobile Device Gateway*: transforms the communication protocols used by mobile devices.
- Transformation can be created at different levels:
  - transport protocols
  - messaging protocols
  - storage device protocols
  - data schemas/data models
- e.g., a multi-device broker may covert both transport and messaging protocols for a cloud service consumer with a mobile device.

Cloud Mechanisms: State Management Database

- **State Management Database**: a storage device that is used to temporarily save state information for software programs.
  - Software programs can offload state information to the database instead of caching them in memory, in order to reduce the amount of run-time memory they consume.
  - By doing so, the software programs and the surrounding infrastructure are more scalable.
  - State management databases are especially used by cloud services that are involved in long-running runtime activities.

Cloud Mechanisms: State Management Database

- During the lifespan of a cloud service instance, it may be required to remain stateful and keep state data cached in memory, even when idle.
- By saving the state data in a state repository, the cloud service can transition to a stateless condition and in the meantime freeing system resources.

Cloud Management Mechanisms: Remote Administration System

- **Remote Administration System**: provides tools and interfaces for external cloud resource administrators to configure & administer cloud-based IT resources.
- The remote administration system exposes and centralizes administration controls to external cloud resource administrators.

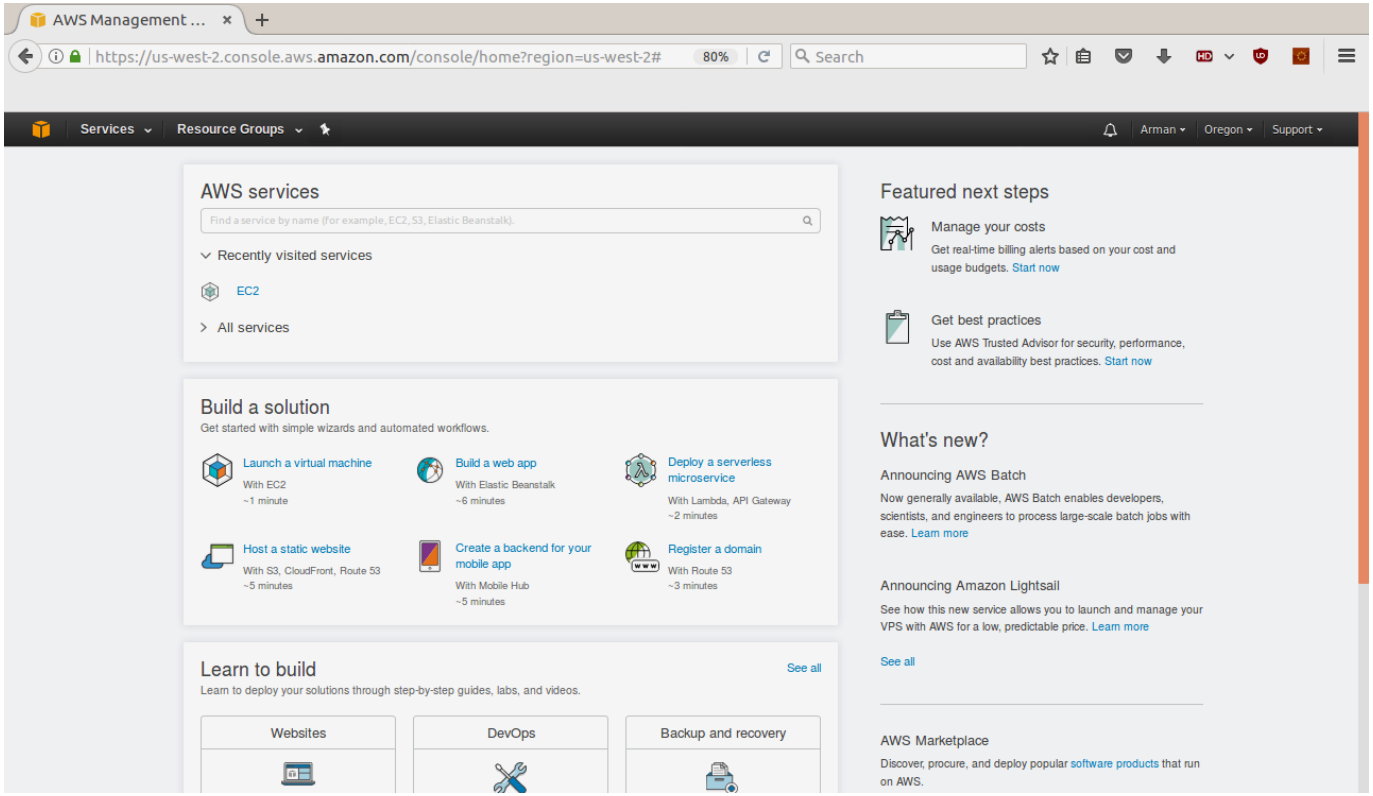
Cloud Management Mechanisms: Remote Administration System

- Two primary types of portals created with the remote administration system:
  - *Usage and Administration Portal* – A general purpose management portal that can also provide IT resource usage reports
  - *Self-Service Portal* – A portal allowing cloud consumers to see the latest list of available cloud services and IT resources to request for provisioning.

Cloud Management Mechanisms: Remote Administration System

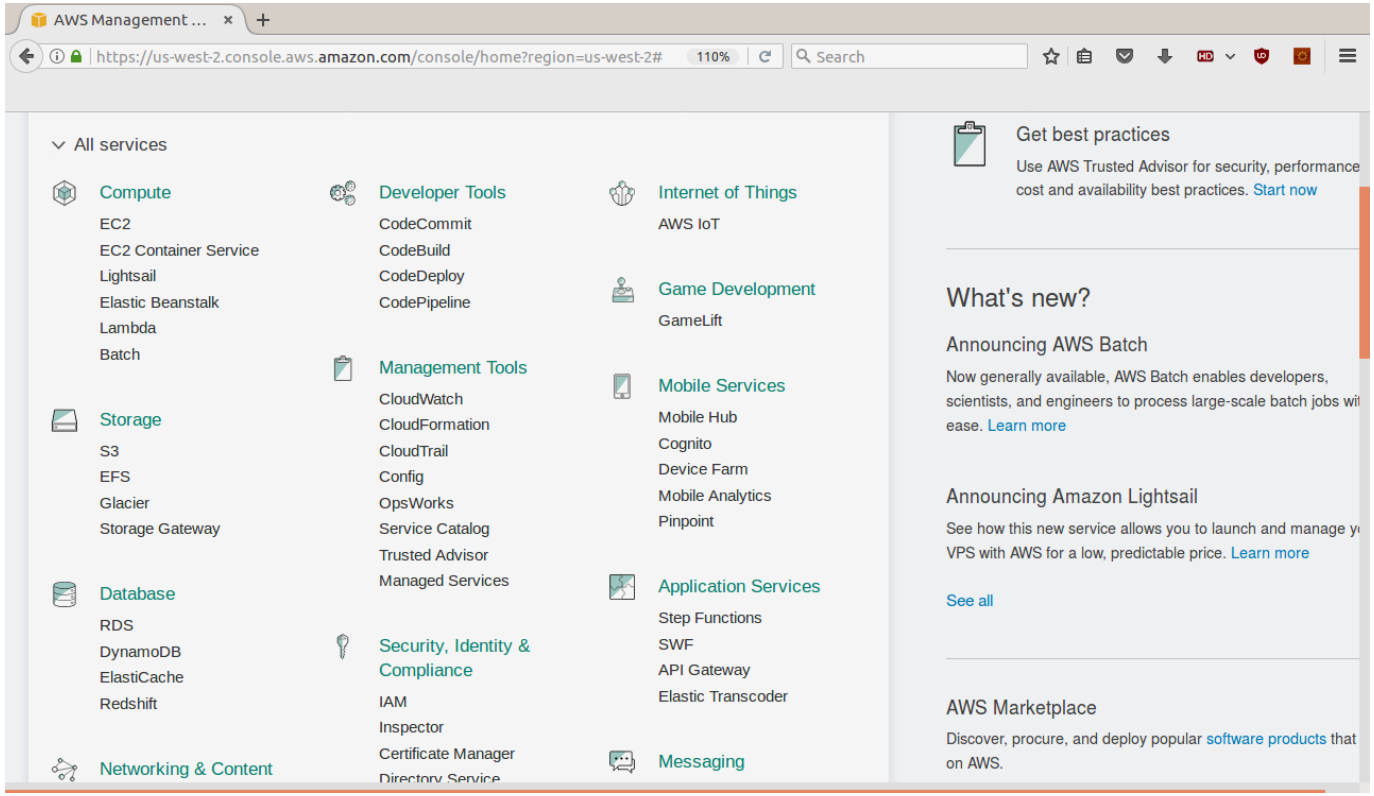
- Example tasks that can be performed by cloud consumers via a remote administration console:
  - configuring cloud services
    - provisioning and releasing IT resource for on-demand cloud services
    - monitoring service status, usage, and performance
    - monitoring QoS and SLA fulfilment
    - managing leasing costs and usage fees
    - managing user accounts, security credentials, authorization, and access control
    - tracking access to leased services
    - planning and assessing IT resource provisioning
    - capacity planning

Cloud Management Mechanisms: Remote Administration System



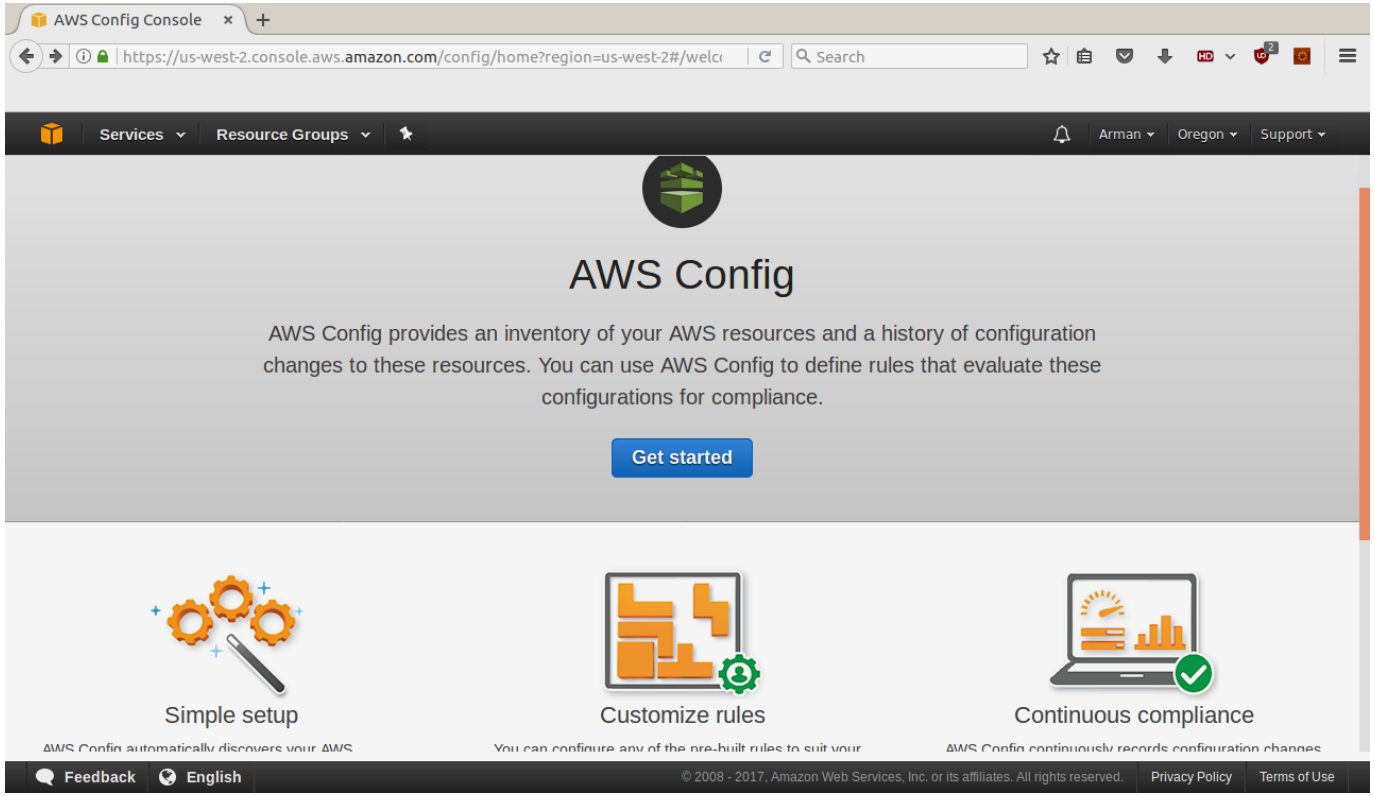
Amazon's AWS console.

Cloud Management Mechanisms: Remote Administration System



Amazon's AWS console.

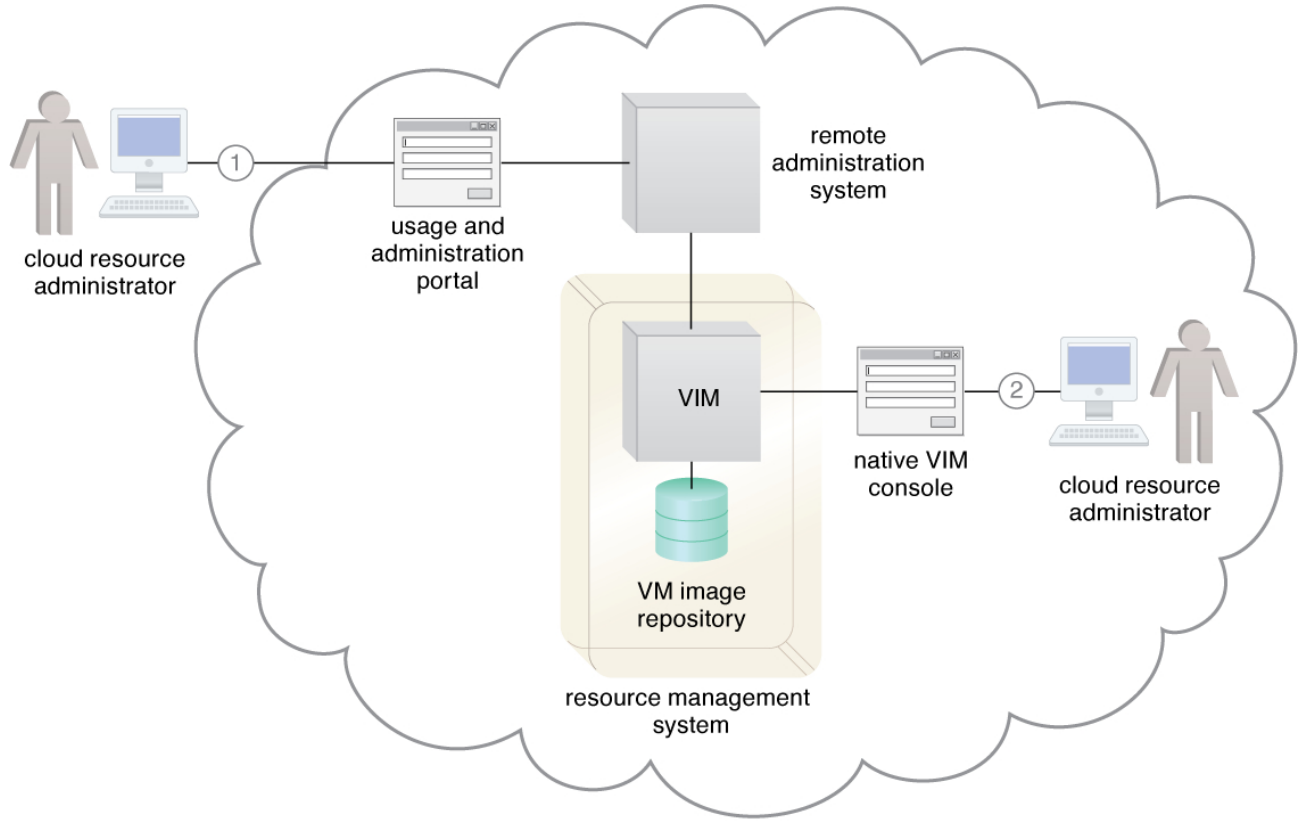
Cloud Management Mechanisms: Remote Administration System



Amazon's AWS console.

Cloud Management Mechanisms: Resource Management System

- **Resource Management System**: coordinates IT resources in response to management actions performed by both cloud consumers and cloud providers (mainly through using the virtual infrastructure manager (VIM) mechanism)



Cloud Management Mechanisms: Resource Management System

- Typical tasks that are automated and implemented through the resource management system:
  - Managing virtual IT resource templates that are used to create pre-built instances, e.g. VM images
  - Allocating & releasing virtual IT resources in response to the starting, pausing, resuming, and termination of virtual instances
  - Coordinating IT resources in relation to the involvement of other mechanisms, e.g. resource replication, load balancer, and failover system
  - Enforcing usage and security policies throughout the lifecycle of cloud service instances
  - Monitoring operational conditions of IT resources

Cloud Management Mechanisms: SLA management System

- **SLA Management System**: provide features pertaining to the administration, collection, storage, reporting, and runtime notification of SLA data.
  - Generally includes a repository used to store and retrieve collected SLA data based on pre-defined metrics and reporting parameters.
    - Relies on one or more SLA monitor mechanisms to collect the SLA data that can then be made available in near-realtime to usage and administration portals to provide ongoing feedback regarding active cloud services.
    - The metrics monitored for individual cloud services are aligned with the SLA guarantees in corresponding cloud provisioning contracts.

Cloud Management Mechanisms: Billing Management System

- **Billing Management System** dedicated to the collection & processing of usage data related to cloud provider accounting & cloud consumer billing.
  - Relies on pay-per-use monitors to gather runtime usage data that is stored in a repository that the system components then draw from for billing reporting and invoicing purposes.
    - Allows for the definition of different pricing policies as well as custom pricing models on a per-cloud consumer and/or per-IT resource basis.
    - Pricing models can vary from the pay-per-use models to flat-rate or pay-per-allocation models, or combinations thereof.

Cloud Management Mechanisms: Billing Management System

- Billing arrangements can be based on pre-usage and post-usage payments. The latter type can include pre-defined limits or can be set up (with the mutual agreement of the cloud consumer) to allow for unlimited usage (and, consequently, no limit on subsequent billing).
- When limits are established, they are usually in the form of usage quotas. When quotas are exceeded, the billing management system can block further usage requests by cloud consumers.