고려대학교
KOREA UNIVERSITY

KU-The Future

Cloud computing – Chapter 7

Heonchang Yu

Distributed and Cloud Computing Lab.

Logical Network Perimeter

- The 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.
- It is defined as the isolation of a network environment from the rest of a communications network

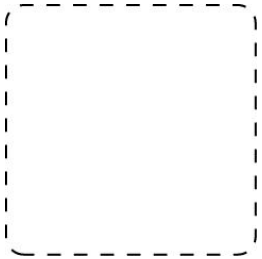


Figure 7.1 The dashed line notation used to indicate the boundary of a logical network perimeter.

Logical Network Perimeter

- This mechanism can be implemented to :
  - ✓isolate IT resources in a cloud from non-authorized users
  - ✓isolate IT resources in a cloud from non-users
  - ✓isolate IT resourced in a cloud from cloud consumers
  - ✓control the bandwidth that is available to isolated IT resources

Logical Network Perimeter

- Logical network perimeters are typically established via network devices that supply and control the connectivity of a data center and are commonly deployed as virtualized IT environments that include:
  - **Virtual Firewall** – An IT resource that actively filters network traffic to and from the isolated network while controlling its interactions with the Internet.
  - **Virtual Network** – Usually acquired through VLANs, this IT resource isolates the network environment within the data center infrastructure

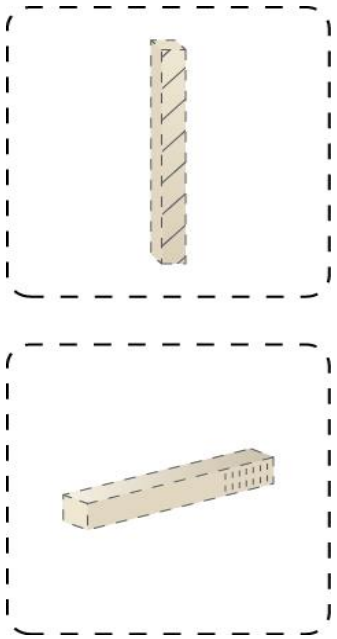


Figure 7.2 The symbols used to represent a virtual firewall (top) and a virtual network (bottom).

Logical Network Perimeter

- One logical network perimeter contains a cloud consumer's on-premise environment while another contains a cloud provider's cloud-based environment.
- These perimeters are connected through a VPN that protects communications, since the VPN is typically implemented by point-to-point encryption of the data packets sent between the communicating endpoints.

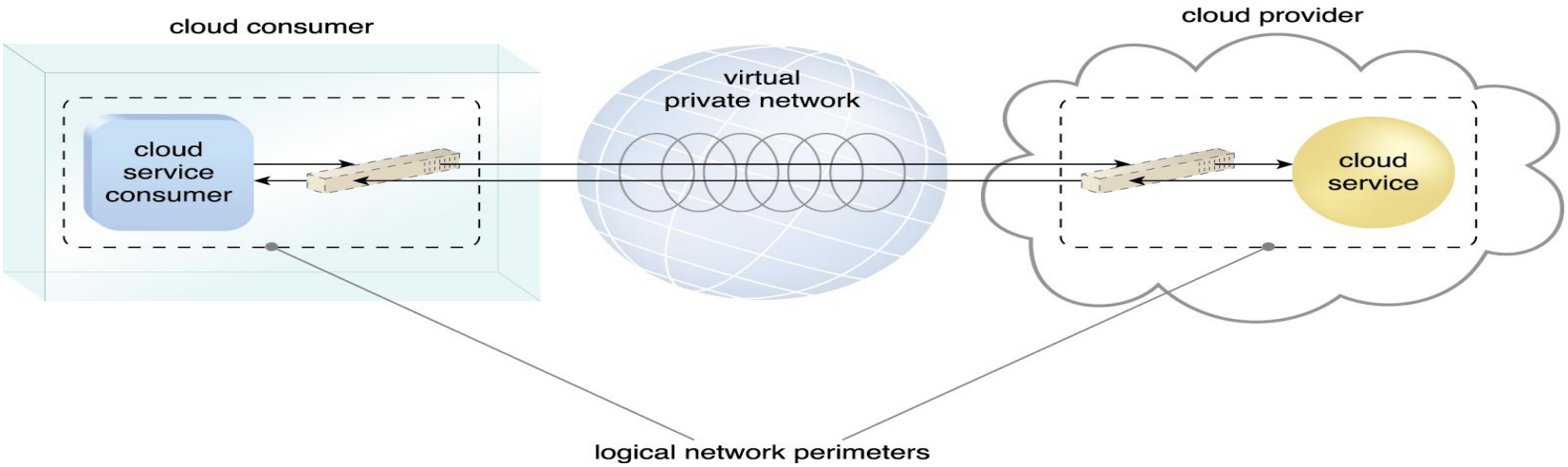


Figure 7.3 Two logical network perimeters surround the cloud consumer and cloud provider environments.

Logical Network Perimeter

- **Case Study Example**
  - DTGOV has virtualized its network infrastructure to produce a logical network layout favoring network segmentation and isolation.
  - The virtual firewall and the isolated virtual network jointly form the cloud consumer's logical network perimeter.
  - Figure 7.4 depicts the logical network perimeter implemented at each DTGOV data center.

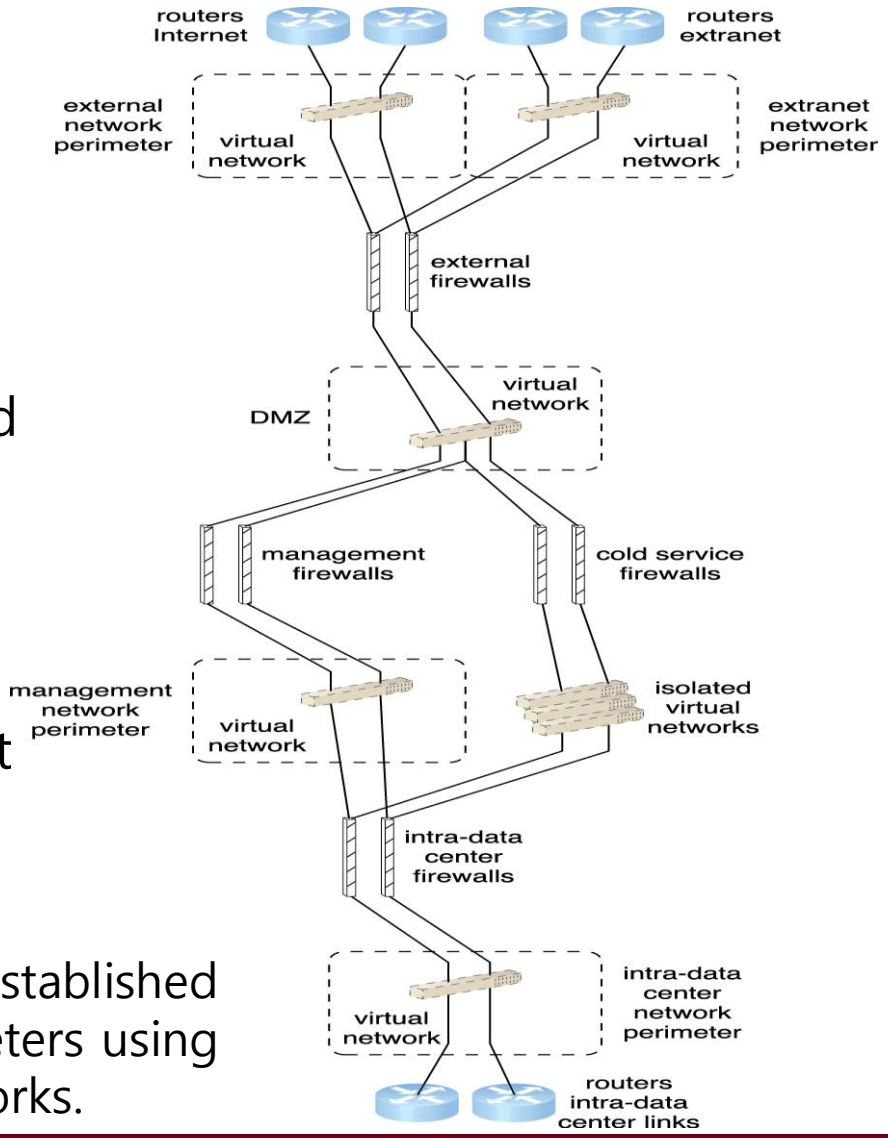


Figure 7.4 A logical network layout is established through a set of logical network perimeters using various firewalls and virtual networks.

Virtual Server

- A virtual server is a form of virtualization software that emulates a physical server.
- Virtual servers are used by cloud providers to share the same physical server with multiple cloud consumers by providing cloud consumers with individual virtual server instances.
- Figure 7.5 shows three virtual servers being hosted by two physical servers. The number of instances a given physical server can share is limited by its capacity.

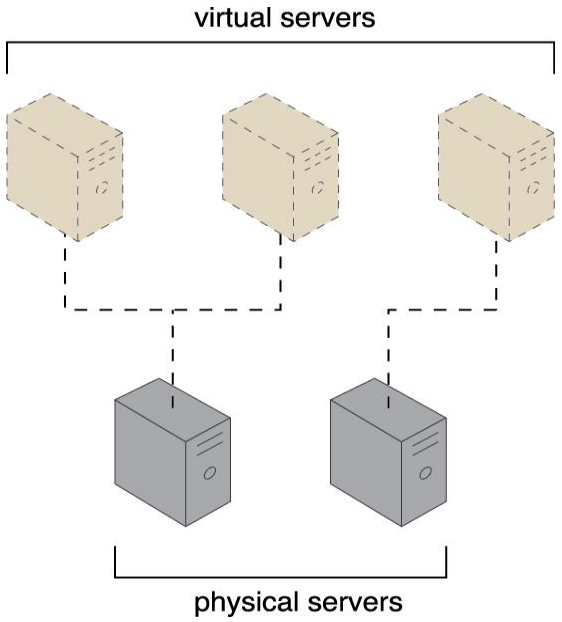


Figure 7.5 The first physical server hosts two virtual servers, while the second physical server hosts one virtual server.

Virtual Server

- As a commodity mechanism, the virtual server represents the most foundational building block of cloud environments.
- Each virtual server can host numerous IT resources, cloud-based solutions, and various other cloud computing mechanisms.
- The instantiation of virtual servers from image files is a resource allocation process that can be completed rapidly and on-demand.
- Figure 7.6 depicts a virtual server that hosts a cloud service being accessed by Cloud Service Consumer B, while Cloud Service Consumer A accesses the virtual server directly to perform an administration task.

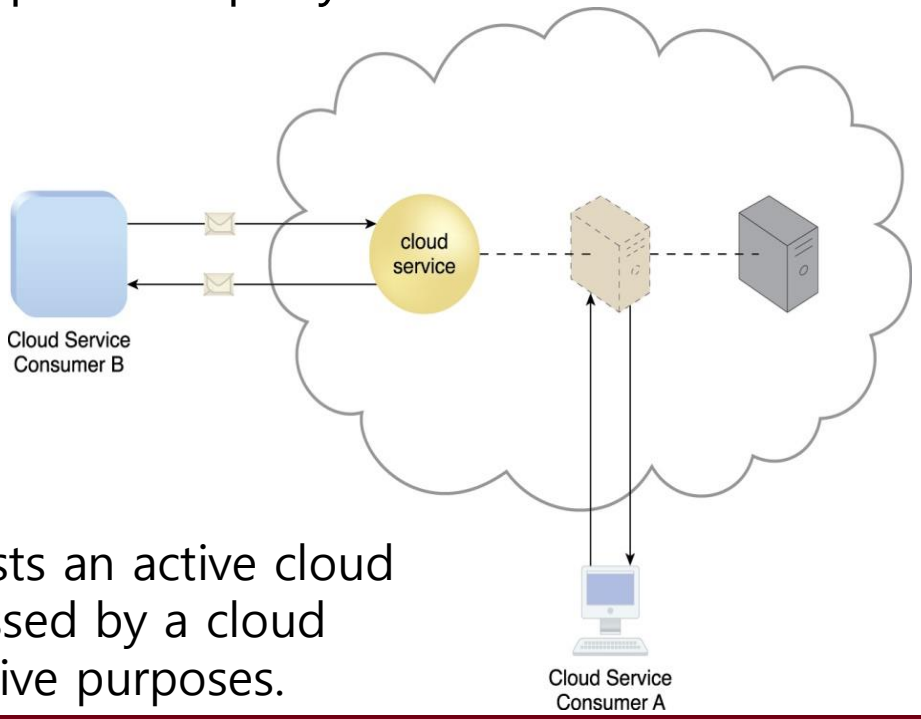


Figure 7.6 A virtual server hosts an active cloud service and is further accessed by a cloud consumer for administrative purposes.

Virtual Server

- **Case Study Example**
  - DTGOV's IaaS environment contains hosted virtual servers that were instantiated on physical servers running the same hypervisor software that controls the virtual servers.
  - Their VIM(Virtual Infrastructure Management) is used to coordinate the physical servers in relation to the creation of virtual server instances. This approach is used at each data center to apply a uniform implementation of the virtualization layer.

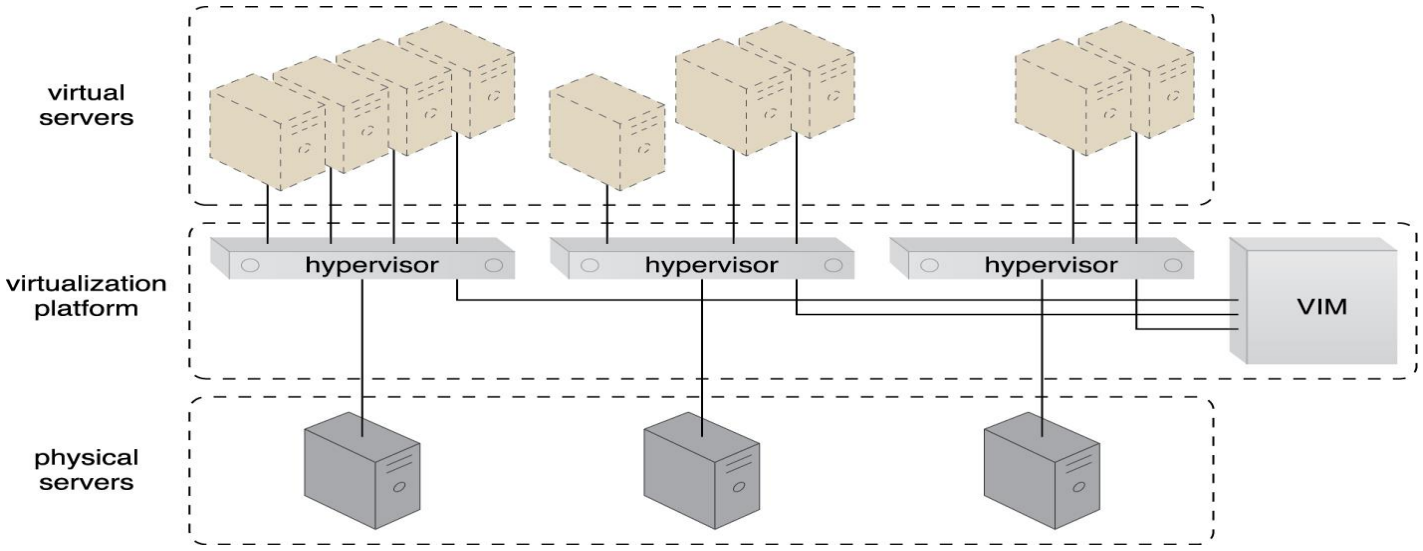


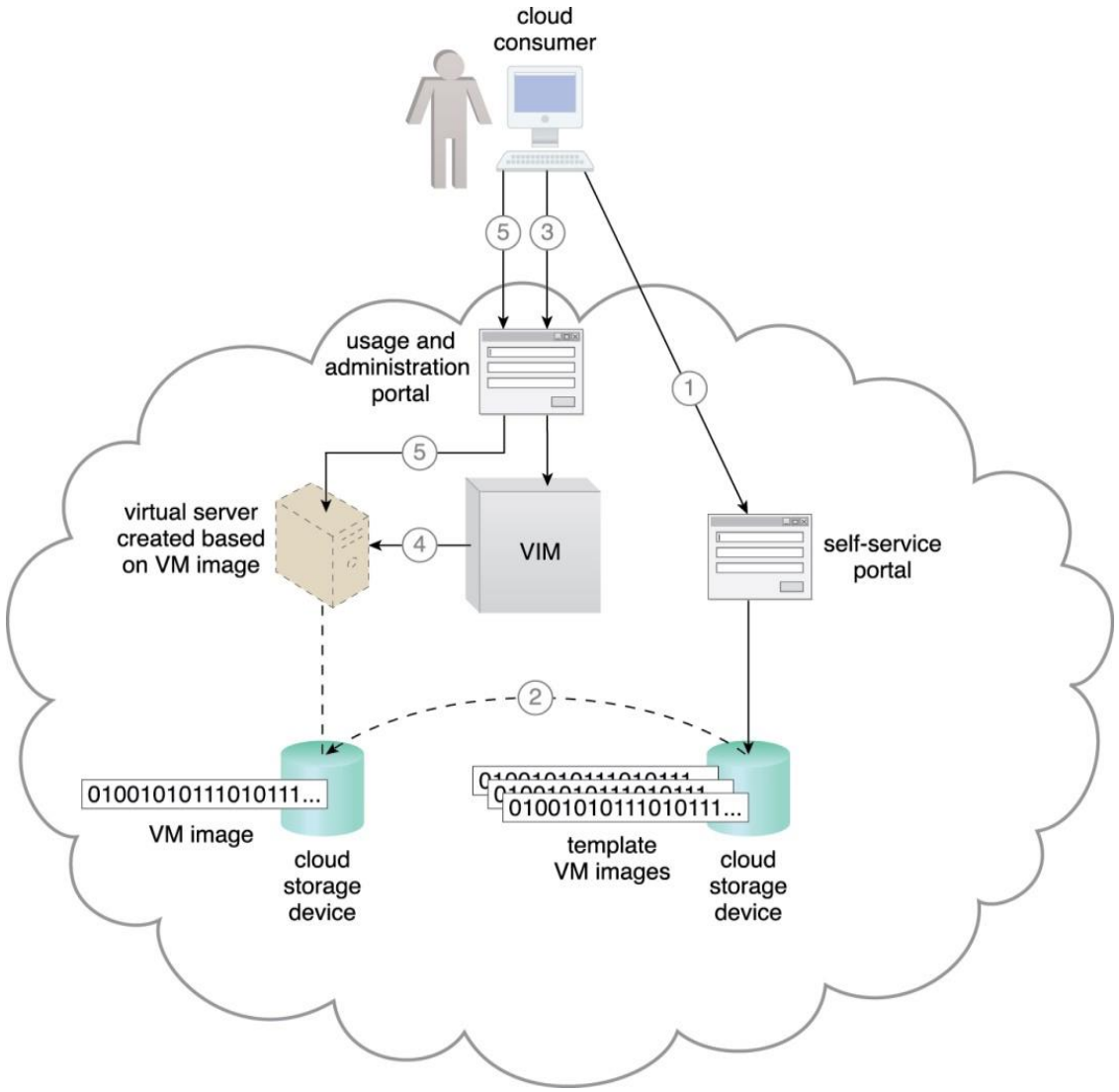
Figure 7.7 Virtual servers are created via the physical servers' hypervisors and a central VIM.

Virtual Server

- **Case Study Example** (Continued)
  - Virtual server packages
    - *Small Virtual Server Instance* – 1 virtual processor core, 4 GB of virtual RAM, 20 GB of storage space in the root file system
    - *Medium Virtual Server Instance* – 2 virtual processor cores, 8 GB of virtual RAM, 20 GB of storage space in the root file system
    - Large Virtual Server Instance – 8 virtual processor cores, 16 GB of virtual RAM, 20 GB of storage space in the root file system
    - *Memory Large Virtual Server Instance* – 8 virtual processor cores, 64 GB of virtual RAM, 20 GB of storage space in the root file system
    - Processor Large Virtual Server Instance – 32 virtual processor cores, 16 GB of virtual RAM, 20 GB of storage space in the root file system
  - DTGOV uses the process described in Figure 7.8 to support the creation and management of virtual servers that have different initial software configurations and performance characteristics.

Virtual Server

Figure 7.8 (1) The cloud consumer uses the self-service portal to select a template virtual server for creation. (2) A copy of the corresponding VM image is created in a cloud consumer-controlled cloud storage device. (3)The cloud consumer initiates the virtual server using the usage and administration portal, (4) which interacts with the VIM to create the virtual server instance via the underlying hardware (5) The cloud consumer is able to use and customize the virtual server via other features on the usage and administration portal.



Cloud Storage Device

- The cloud storage device mechanism represents storage devices that are designed specifically for cloud-based provisioning.
- Instances of these devices can be virtualized, similar to how physical servers can spawn virtual server images.
- Cloud storage devices are commonly able to provide fixed-increment capacity allocation in support of the pay-per-use mechanism.
- Cloud storage devices can be exposed for remote access via cloud storage services.

- **Primary Concern**
  - ✓the security, integrity, and confidentiality of data
  - ✓legal and regulatory implications.
  - ✓performance of large databases.

Cloud Storage Device

- **Cloud Storage Levels**

  - provide common logical units of data storage, such as:

    ✓**Files** – Collections of data are grouped into files that are located in folders.

    ✓**Blocks** – The lowest level of storage and the closest to the hardware, a block is the smallest unit of data that is still individually accessible.

    ✓**Datasets** – Sets of data are organized into a table-based, delimited, or record format.

    ✓**Objects** – Data and its associated metadata are organized as Web-based resources.

Cloud Storage Device

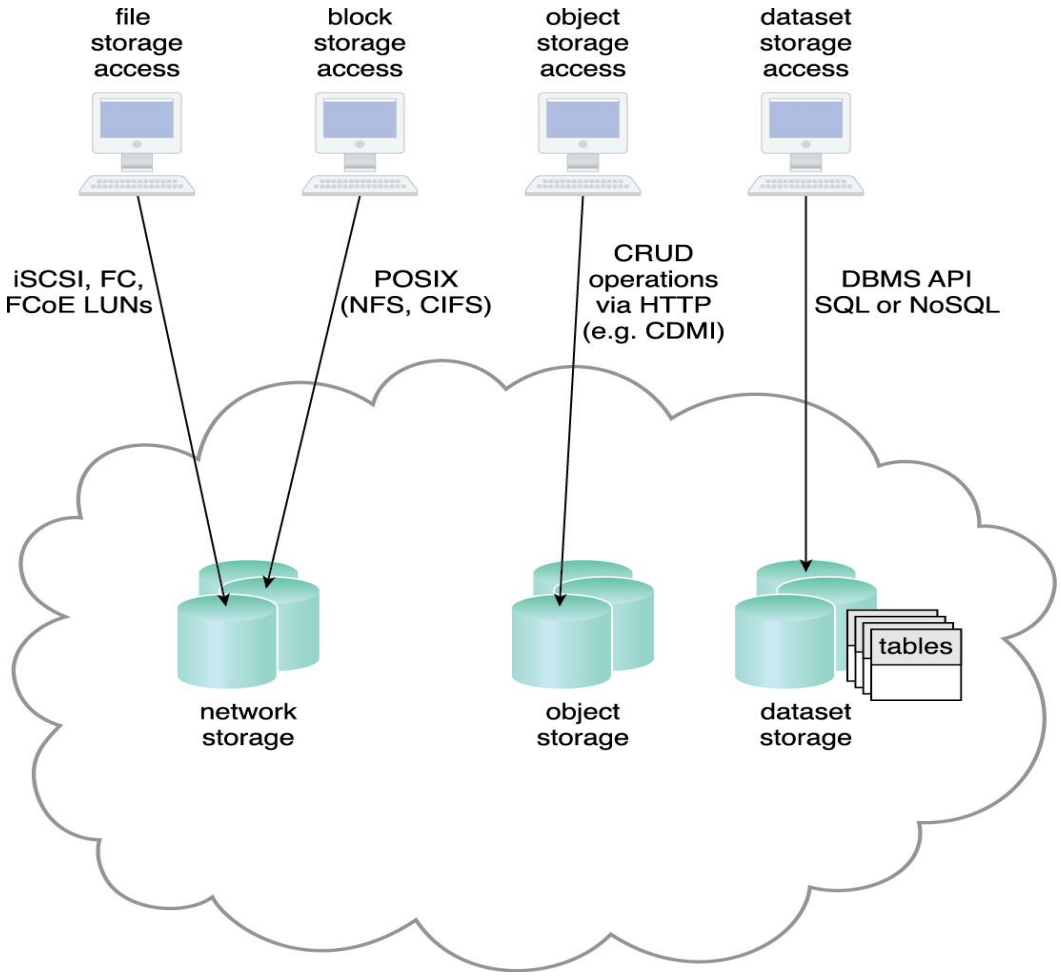


Figure 7.9 Different cloud service consumers utilize different technologies to interface with virtualized cloud storage devices.

Cloud Storage Device

- **Network Storage Interfaces**
  - Legacy network storage most commonly falls under the category of network storage interfaces. It includes storage devices in compliance with industry standard protocols, such as:
    - ✓ SCSI for storage blocks and the server message block (SMB), common Internet file system (CIFS), and network file system (NFS) for file and network storage.
  - When a cloud storage device mechanism is based on this type of interface, its data searching and extraction performance will tend to be suboptimal. Storage processing levels and thresholds for file allocation are usually determined by the file system itself.
  - Block storage requires data to be in a fixed format (known as a data block), which is the smallest unit that can be stored and accessed and the storage format closest to hardware.
    - ✓ Using either the logical unit number (LUN) or virtual volume block-level storage will typically have better performance than file-level storage.

Cloud Storage Device

- **Object Storage Interfaces**
  - ✓Various types of data can be referenced and stored as Web resources. This is referred to as object storage, which is based on technologies that can support a range of data and media types.
  - ✓Cloud Storage Device mechanisms that implement this interface can typically be accessed via REST or Web service-based cloud services using HTTP as the prime protocol.
  - ✓The Storage Networking Industry Association's Cloud Data Management Interface (SNIA's CDMI) supports the use of object storage interfaces.

Cloud Storage Device

- **Database Storage Interfaces**
  - ✓ Cloud storage device mechanisms based on database storage interfaces typically support a query language in addition to basic storage operations.
  - ✓ Storage management is carried out using a standard API or an administrative user-interface.
  - ✓ This classification of storage interface is divided into two main categories according to storage structure, as follows.
    - **Relational Data Storage -** using SQL, rely on tables, data normalization
    - **Non-Relational Data Storage -** NoSQL storage, establishes a “looser” structure, data denormalization

Cloud Storage Device

▪ 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).
- ✓ Using the industry standard Structured Query Language (SQL).
- ✓ It could be based on commercially available database products, such as IBM DB2, Oracle Database, Microsoft SQL Server, and MySQL.

- **Challenges** : scaling and performance.
  - ✓ Scaling a relational cloud storage device vertically can be more complex and cost-ineffective than horizontal scaling.
  - ✓ Databases with complex relationships and/or containing large volumes of data can be afflicted with higher processing overhead and latency, especially when accessed remotely via cloud services.

Cloud Storage Device

▪ Non-Relational Data Storage

- ✓ Non-relational storage (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.
- ✓ More horizontally scalable than relational storage.

- **Characteristics**
  - ✓ The data loses much of the native form and validation due to limited or primitive schemas or data models.
  - ✓ Not support relational database functions, such as transactions or joins.
  - ✓ Normalized data will usually become denormalized, meaning that the size of the data will typically grow.
  - ✓ Many non-relational storage mechanisms are proprietary and therefore can severely limit data portability.

Cloud Storage Device

- **Case Study Example**
  - DTGOV provides cloud consumers access to a cloud storage device based on an object storage interface. The cloud service that exposes this API offers basic functions on stored objects, such as search, create, delete, and update. The search function uses a hierarchical object arrangement that resembles a file system.
  - DTGOV further offers a cloud service that is used exclusively with virtual servers and enables the creation of cloud storage devices via a block storage network interface.

    - **Object-based cloud storage interface**
    - **Block-based cloud storage interface**

Cloud Storage Device

- **Case Study Example** (Continued)
  - **Object-based cloud storage interface**

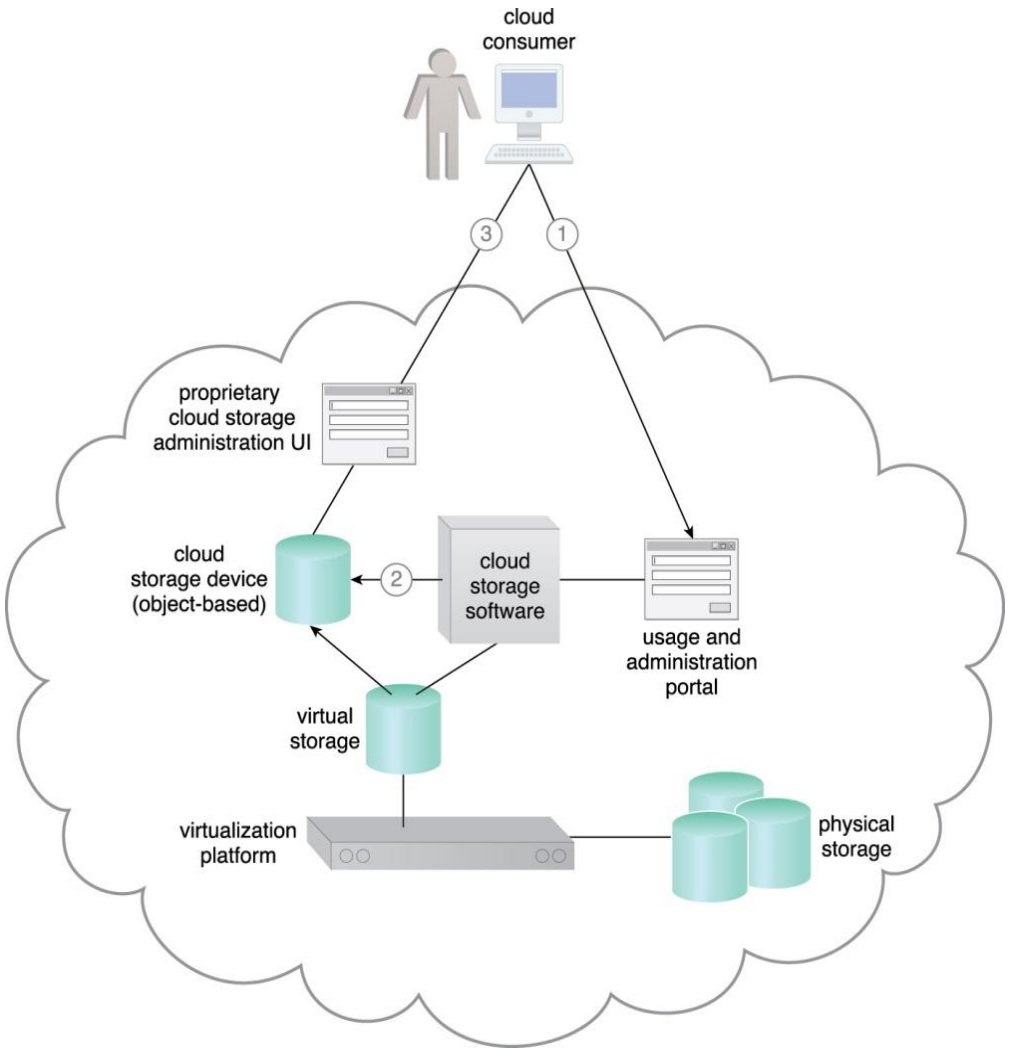
✓ The object-based cloud storage device has an underlying storage system with variable storage capacity, which is directly controlled by a software component that also exposes the interface.

✓ This software enables the creation of isolated cloud storage devices that are allocated to cloud consumers. The storage system uses a security credential management system to administer user-based access control to the device's data objects (Figure 7.10)

✓ Access control is granted on a per-object basis and uses separate access policies for creating, reading from, and writing to each data object. Public access permissions are allowed, although they are read-only. Access groups are formed by nominated users that must be previously registered via the credential management system. Data objects can be accessed from both Web applications and Web service interfaces, which are implemented by the cloud storage software.

Cloud Storage Device

Figure 7.10 (1) The cloud consumer interacts with the usage and administration portal to create a cloud storage device and define access control policies. (2) The usage and administration portal interact with the cloud storage software to create the cloud storage device instance and apply the required access policy to its data objects. (3) Each data object is assigned to a cloud storage device and all of the data objects are stored in the same virtual storage volume. The cloud consumer uses the proprietary cloud storage device UI to interact directly with the data objects .

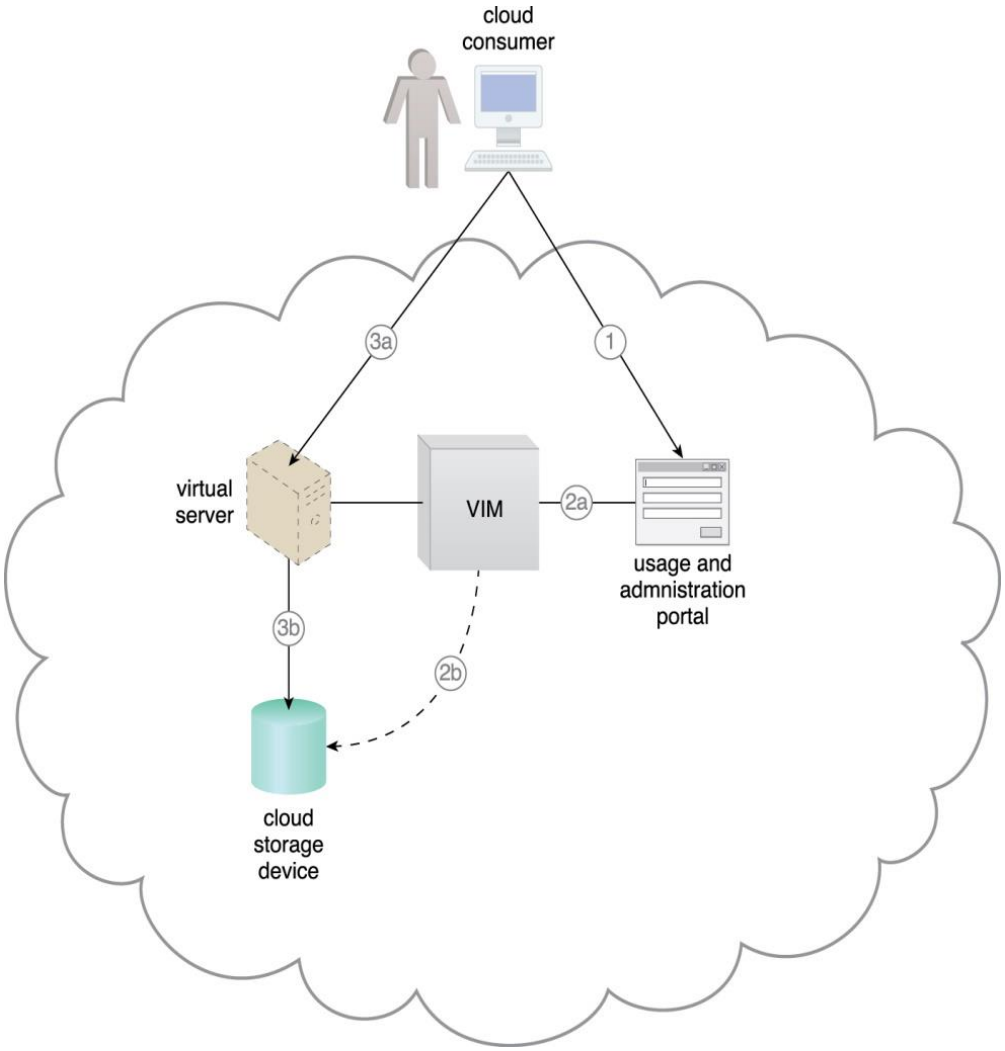


Cloud Storage Device

- **Case Study Example** (Continued)
  - **Block-based cloud storage interface**
    - ✓ The creation of the cloud consumers' block-based cloud storage devices is managed by the virtualization platform, which instantiates the LUN's implementation of the virtual storage (Figure 7.11).
    - ✓ The cloud storage device (or the LUN) must be assigned by the VIM to an existing virtual server before it can be used.
    - ✓ The capacity of block-based cloud storage devices is expressed by one GB increments. It can be created as fixed storage that cloud consumers can modify administratively or as variable size storage that has an initial 5 GB capacity that automatically increases and decreases by 5 GB increments according to usage demands.

Cloud Storage Device

Figure 7.11 (1) The cloud consumer uses the usage and administration portal to create and assign a cloud storage device to an existing virtual server. (2a) The usage and administration portal interacts with the VIM software, (2b) which creates and configures the appropriate LUN. (3a) Each cloud storage device uses a separate LUN controlled by the virtualization platform. The cloud consumer remotely logs into the virtual server directly (3b) to access the cloud storage device.



Cloud Usage Monitor

- **Cloud Usage Monitor**
  - ✓A lightweight and autonomous software program responsible for collecting and processing IT resource usage data
  - ✓Designed to forward collected usage data to log database for post-processing and reporting purpose.
  - ✓Three common agent-based implementation formats.
    - Monitoring Agent
    - Resource Agent
    - Polling Agent

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 data flows.

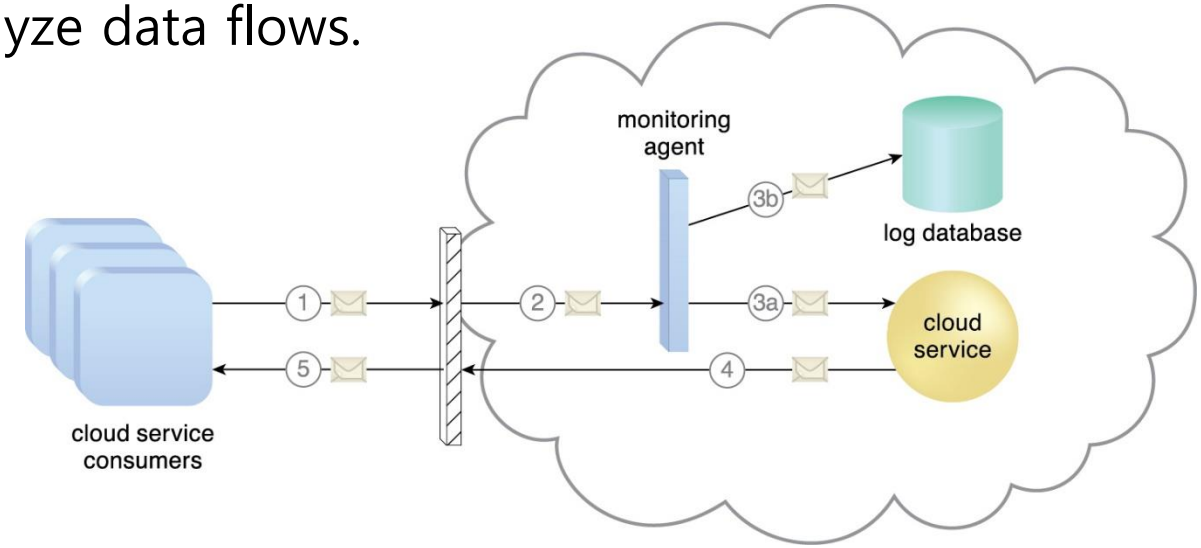


Figure 7.12 (1) A cloud service consumer sends a request message to a cloud service. (2) The monitoring agent intercepts the message to collect relevant usage data (3a) before allowing it to continue to the cloud service. (3b) The monitoring agent stores the collected usage data in a log database. (4) The cloud service replies with a response message (5) that is sent back to the cloud service consumer without being intercepted by the monitoring agent.

Cloud Usage Monitor

- Resource Agent
  - ✓A processing module that collects usage data by having event-driven interactions with specialized resource software.
  - ✓Monitoring usage metrics (initiating, suspending, resuming and vertical scaling)

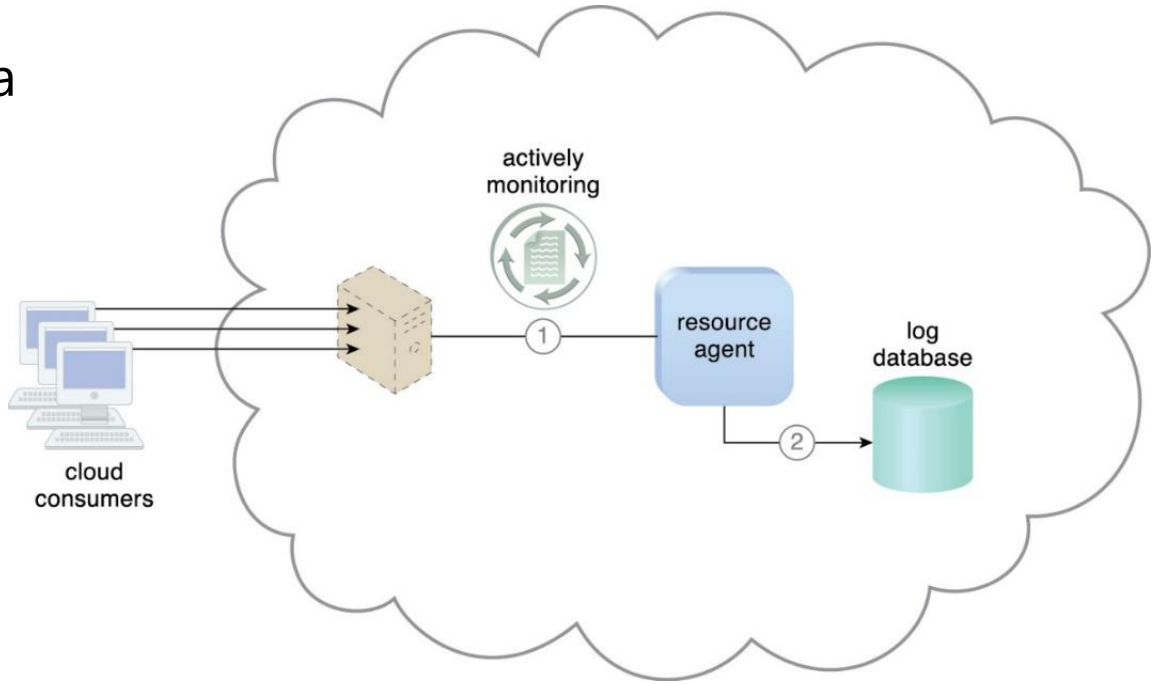


Figure 7.13 (1) The resource agent is actively monitoring a virtual server and detects an increase in usage. (2) The resource agent receives a notification from the underlying resource management program that the virtual server is being scaled up and stores the collected usage data in a log database, as per its monitoring metrics.

Cloud Usage Monitor

- Polling Agent
  - ✓A processing module that collects cloud service usage data by polling IT resources.
  - ✓Is to periodically monitor IT resource status. (uptime and downtime)

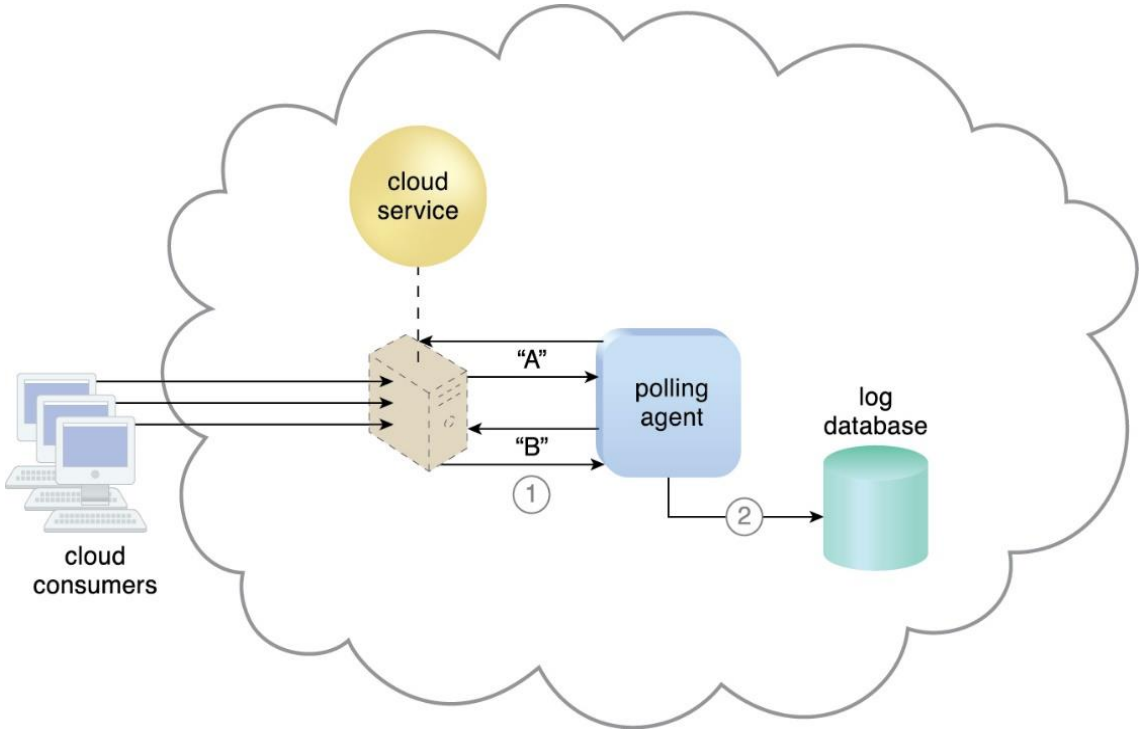
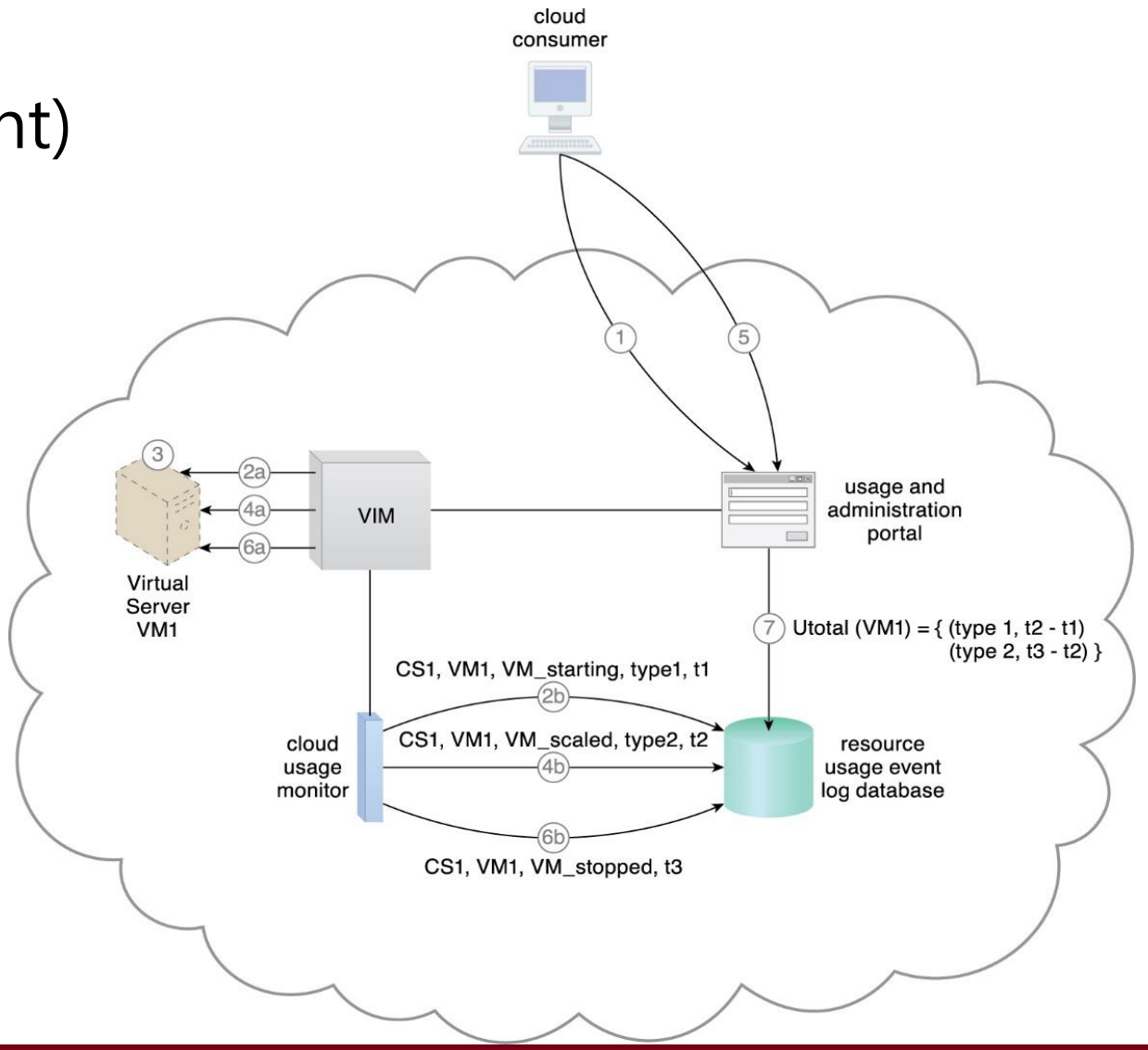


Figure 7.14 (1) A polling agent monitors the status of a cloud service hosted by a virtual server by sending periodic polling request messages and receiving polling response messages that report usage status "A" after a number of polling cycles, until it receives a usage status of "B", (2) upon which the polling agent records the new usage status in the log database.

Cloud Usage Monitor

- Case Study (resource agent)



Cloud Usage Monitor

- Case Study

Figure 7.15
(1) The cloud consumer (CS_ID = CS1) requests the creation of a virtual server (VM_ID = VM1) of configuration size type 1 (VM_TYPE = type1). (2a) The VIM creates the virtual server. (2b) The VIM's event-driven API generates a resource usage event with timestamp = t1, which the cloud usage monitor software agent captures and records in the resource usage event log database. (3) Virtual server usage increases and reaches the auto-scaling threshold. (4a) The VIM scales up Virtual Server VM1 from configuration type 1 to type 2 (VM_TYPE = type2). (4b) The VIM's event-driven API generates a resource usage event with timestamp = t2, which is captured and recorded at the resource usage event log database by the cloud usage monitor software agent. (5) The cloud consumer shuts down the virtual server. (6a) The VIM stops Virtual Server VM1 and (6b) its event-driven API generates a resource usage event with timestamp = t3, which the cloud usage monitor software agent captures and records at the log database. (7) The usage and administration portal accesses the log database and calculates the total usage (Utotal) for Virtual Server Utotal VM1.

Resource Replication

- Resource Replication
  - ✓ The creation of multiple instances of the same IT resource
  - ✓ Replication is typically performed when an IT resource's availability and performance need to be enhanced.

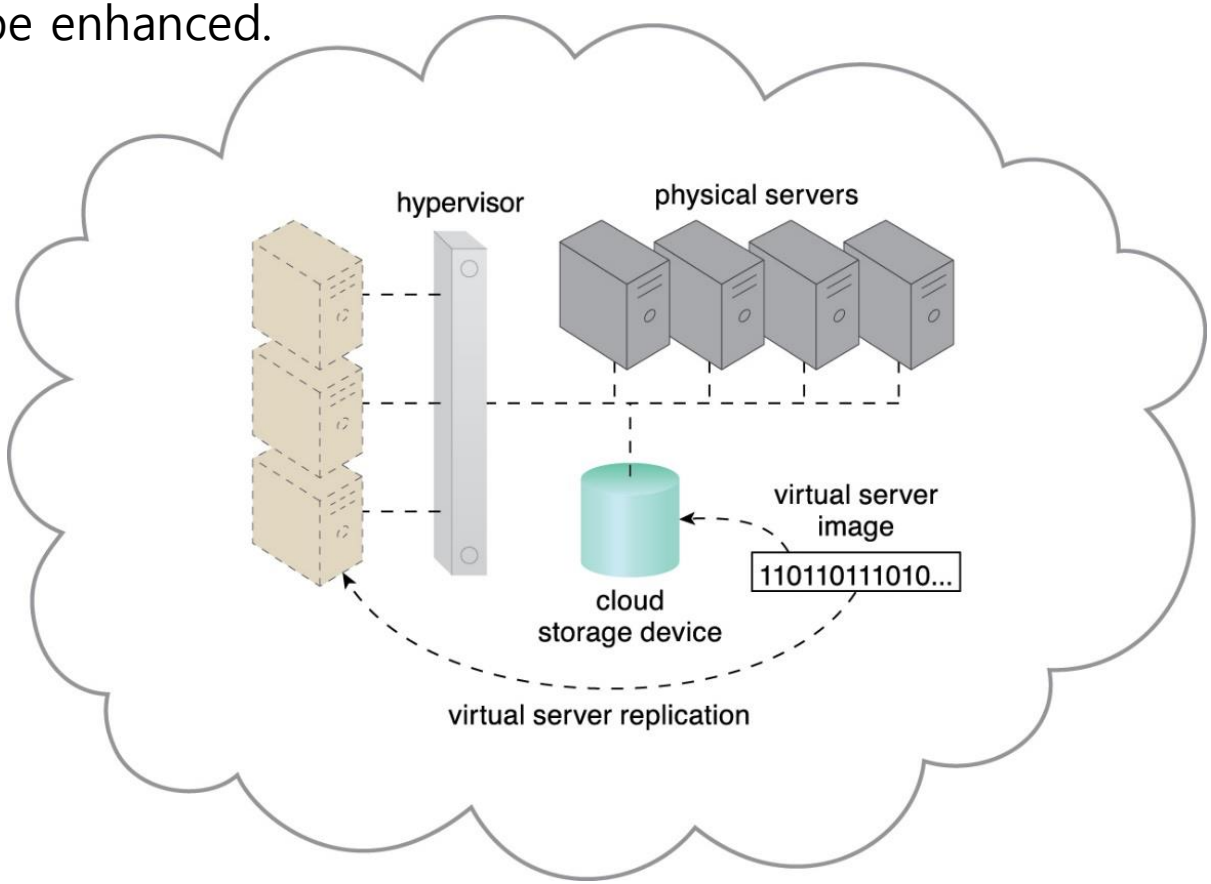


Figure 7.16
The hypervisor replicates several instances of a virtual server, using a stored virtual server image.

Resource Replication

- Case Study
  - ✓DTGOV establishes a set of high-availability virtual servers that can be automatically relocated to physical servers running in different data centers in response to severe failure conditions.

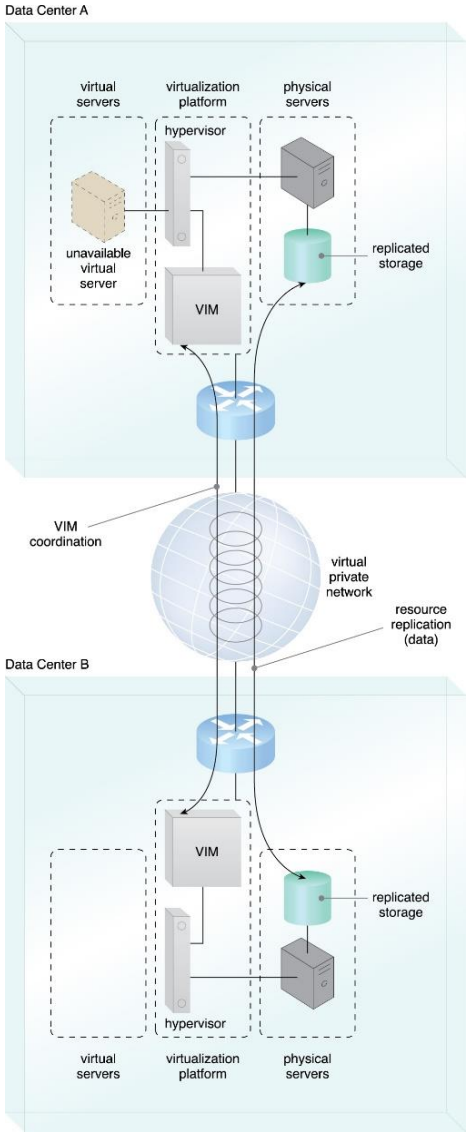


Figure 7.17
A high-availability virtual server is running in Data Center A. VIM instances in Data Centers A and B are executing a coordination function that allows detection of failure conditions. Stored VM images are replicated between data centers as a result of the high-availability architecture.

Resource Replication

- Case Study

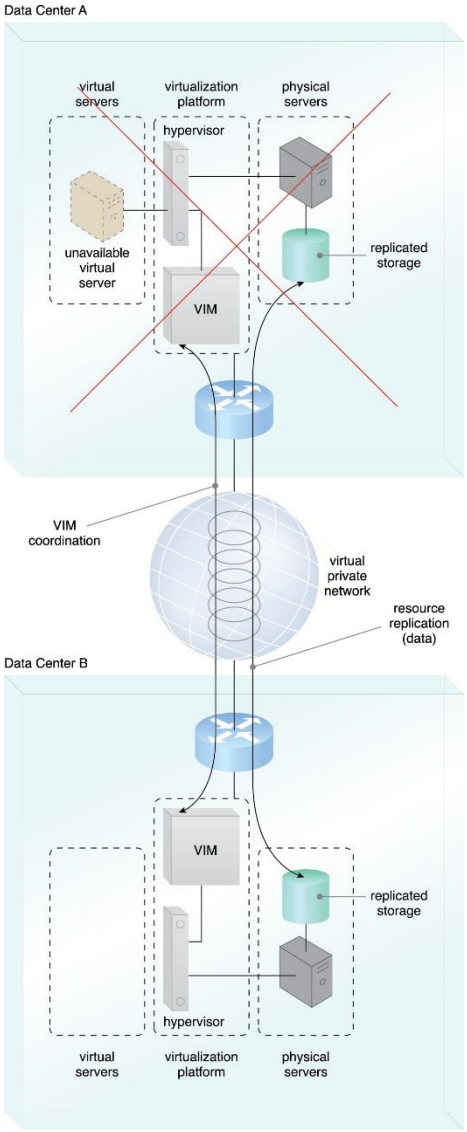


Figure 7.18
The virtual server becomes unavailable in Data Center A. The VIM in Data Center B detects the failure condition and starts to reallocate the high-availability server from Data Center A to Data Center B.

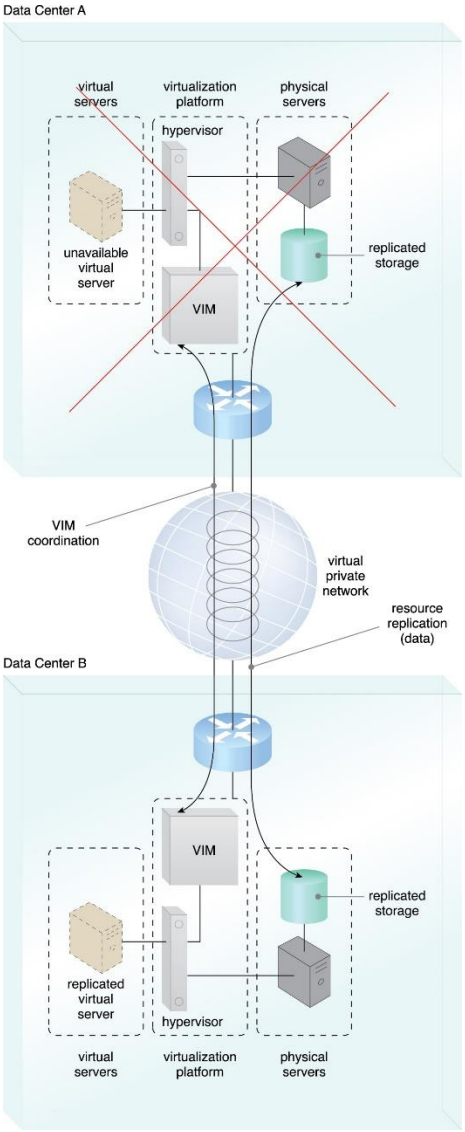


Figure 7.19
A new instance of the virtual server is created and made available in Data Center B.

Ready-Made Environment

- Ready-Made Environment
  - ✓ 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
  - ✓ Are utilized by cloud consumers to remotely develop and deploy their own services and applications within a cloud

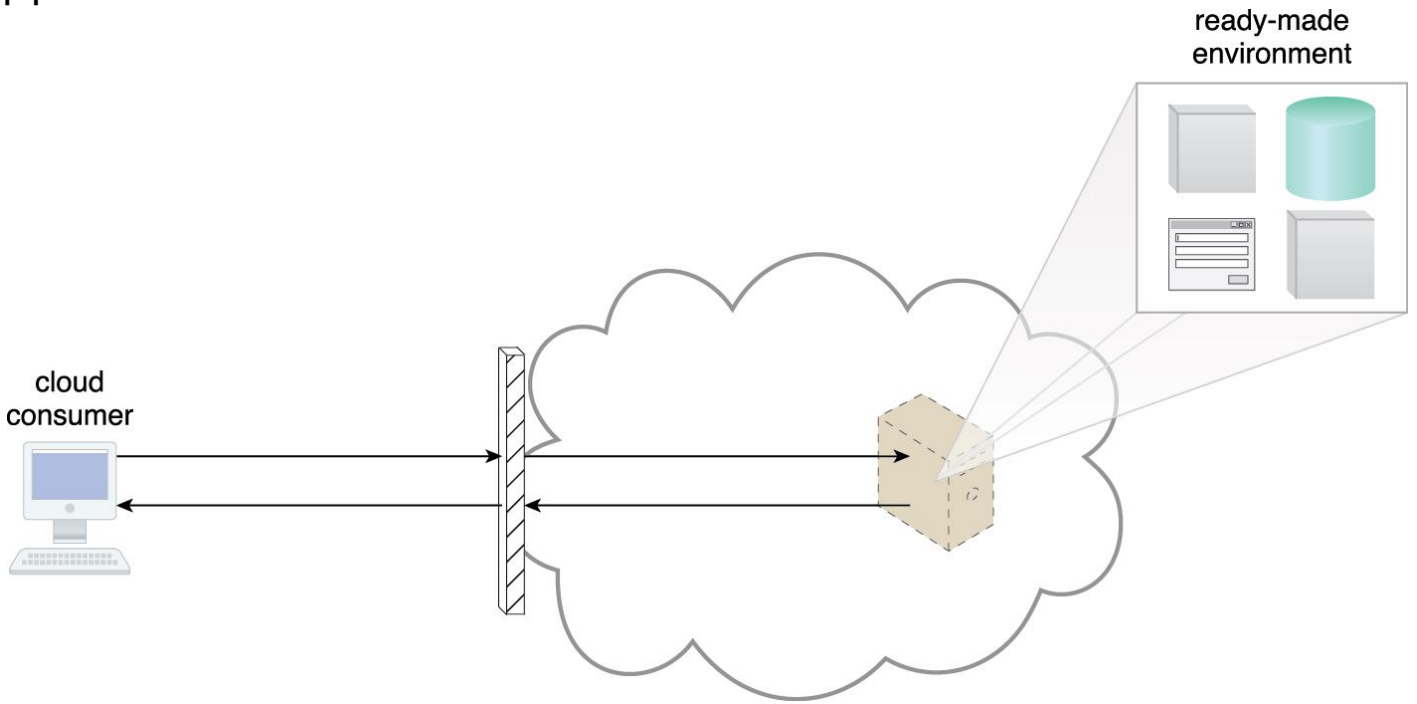


Figure 7.20
A cloud consumer accesses a ready-made environment hosted on a virtual server.

Ready-Made Environment

- Ready-Made Environment
  - ✓Typical ready-made environments include pre-installed IT resources, such as database, middleware, development tools, and governance tools.
  - ✓ A ready-made environment is generally equipped with a complete software development kit(SDK) that provides cloud consumers with programmatic access to the development technologies that comprise their preferred programming stack.

Ready-Made Environment

Figure 7.21
(1) The developer uses the provided SDK to develop the Part Number Catalog Web application. (2a) The application software is deployed on a Web platform that was established by two ready-made environments called the front-end instance. (2b) and the back-end instance. (3) The application is made available for usage and one end-user accesses its front-end instance. (4) The software running in the front-end instance invokes a long-running task at the back-end instance that corresponds to the processing required by the end-user. (5) The application software deployed at both the front-end and back-end instances is backed by a cloud storage device that provides persistent storage of the application data.