

Chapter 1



Introduction

Cloud Computing Definition

- Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.
- This cloud model promotes availability and is composed of
- **Five** essential characteristics,
- **Three** service models, and
- **Four** deployment models.

Essential Characteristics:

- **On-demand self-service.** A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.
- **Broad network access.** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).
- **Resource pooling.** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.

Essential Characteristics:

- **Rapid elasticity.** Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.
- **Measured Service.** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

The Pros and Cons of Cloud Computing

- **Pros of cloud computing:**
- No cost on infrastructure
- Minimum management and cost (Cost Saving)
- Forget about administrative or management hassles
- Accessibility and pay per use
- Reliability
- **Cons of cloud computing:**
- Requires good speed internet with good bandwidth
- Limited control on infrastructure
- Restricted or limited flexibility
- Ongoing costs
- Security

Cloud Service Models

- **Cloud Software as a Service (SaaS).**

The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings

- **Cloud Platform as a Service (PaaS).**

The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages 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 application hosting environment configurations.

Cloud Service Models

- **Cloud Infrastructure as a Service (IaaS).**

The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).

Deployment Models (Cloud Types)

- **Private cloud.**

The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.

- **Public cloud.**

The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

- **Hybrid cloud.**

The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).

- **Community cloud.**

The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.

Deployment Models (Cloud Types)

- **Private cloud.**

The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.

- **Public cloud.**

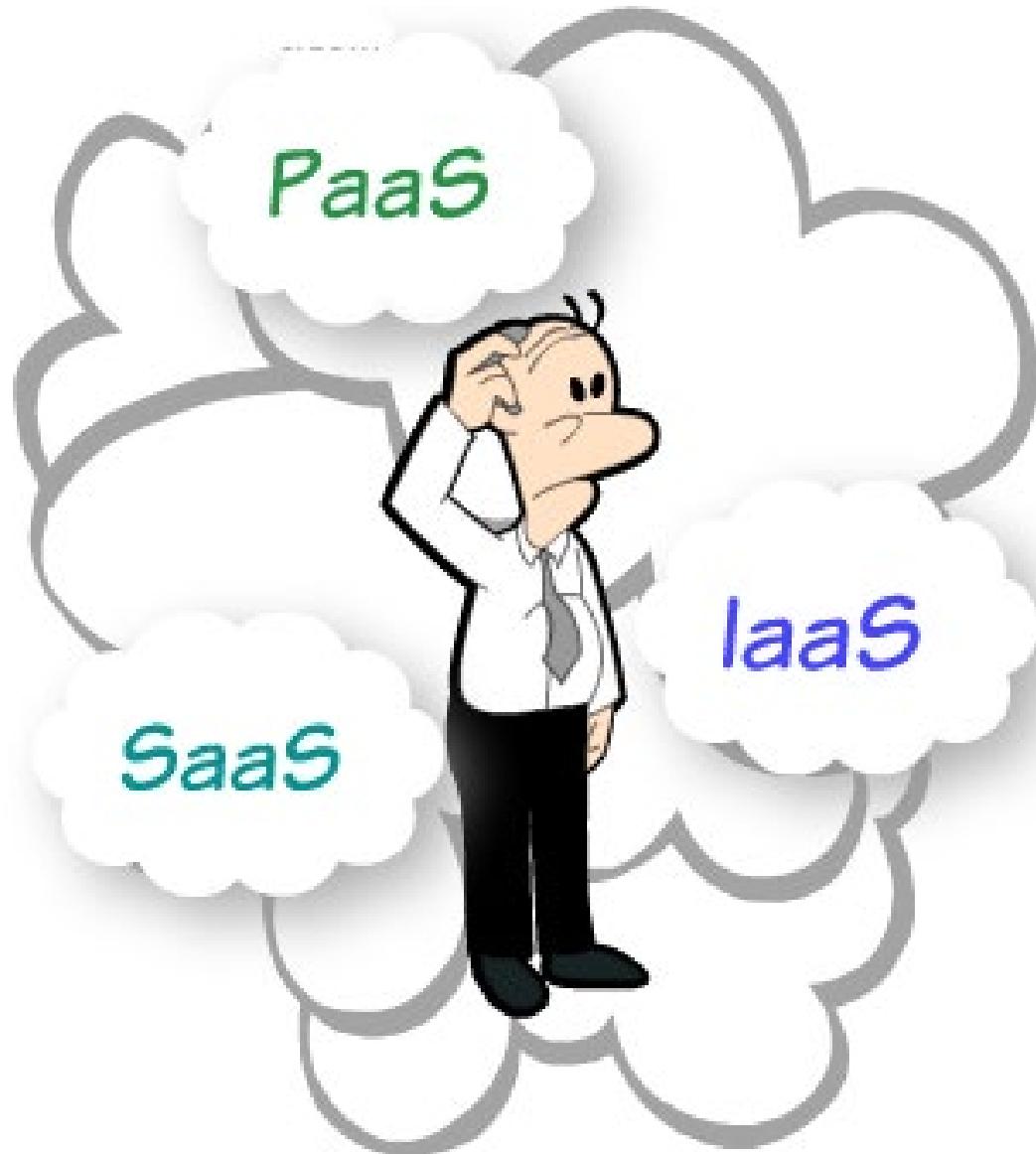
The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

- **Hybrid cloud.**

The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).

- **Community cloud.**

The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.



Comparison of SAAS PASS IAAS



Comparison of SAAS PASS IAAS

SaaS

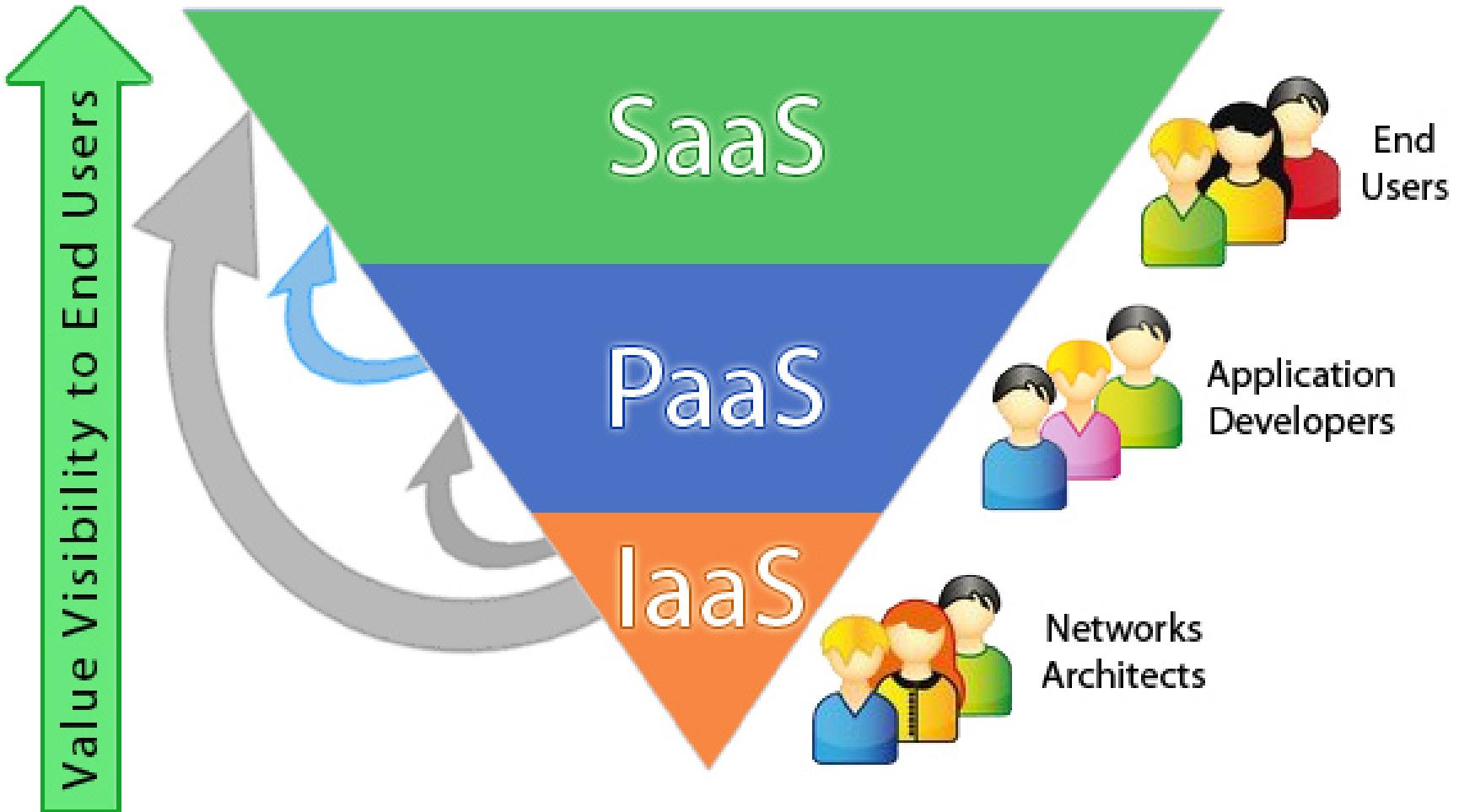
- Software as a service
- Operating environment largely irrelevant, fully functional applications provided, e.g. CRM, ERP, email

PaaS

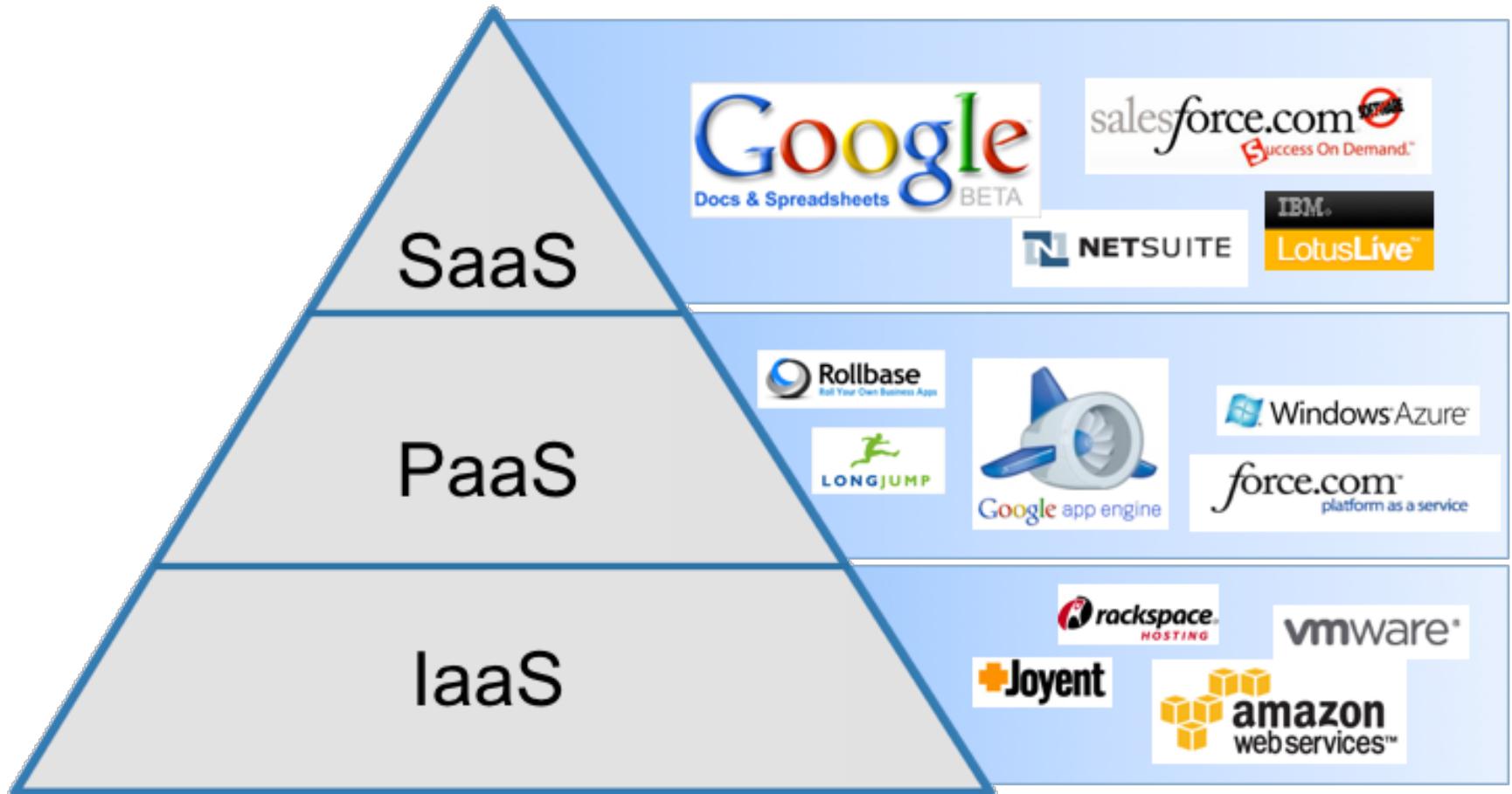
- Platform as a service
- Operating environment included, e.g. Windows/.NET, Linux/J2EE, applications of choice deployed

IaaS

- Infrastructure as a service
- Virtual platform on which required operating environment and application are deployed
- Includes storage as a service offerings



Example of vendors



Benefits of CC



Global Infrastructure Savings



Smart Resource Provisioning



Better Delivery Time



Transparent Workflow

Thank You



Cloud Computing

Concepts and Models

Concepts and Models

- [4.1 Roles and Boundaries](#)
- [4.2 Cloud Characteristics](#)
- [4.3 Cloud Delivery Models](#)
- [4.4 Cloud Deployment Models](#)

4.1. Roles and Boundaries

- Cloud Provider
- Cloud Consumer
- Cloud Service Owner
- Cloud Resource Administrator
- Organizational Boundary
- Trust Boundary

Cloud Provider

- The organization that provides cloud-based IT resources is the *cloud provider*.
- The cloud provider is further tasked with any required management and administrative duties to ensure the on-going operation of the overall cloud infrastructure.
- Cloud providers normally own the IT resources that are made available for lease by cloud consumers; however, some cloud providers also “resell” IT resources leased from other cloud providers.

Cloud Consumer

- A *cloud consumer* is an organization (or a human) that has a formal contract or arrangement with a cloud provider to use IT resources made available by the cloud provider.
- cloud consumer uses a cloud service consumer to access a cloud service.
- organizations or humans shown *remotely accessing cloud-based IT resources are considered cloud consumers*.

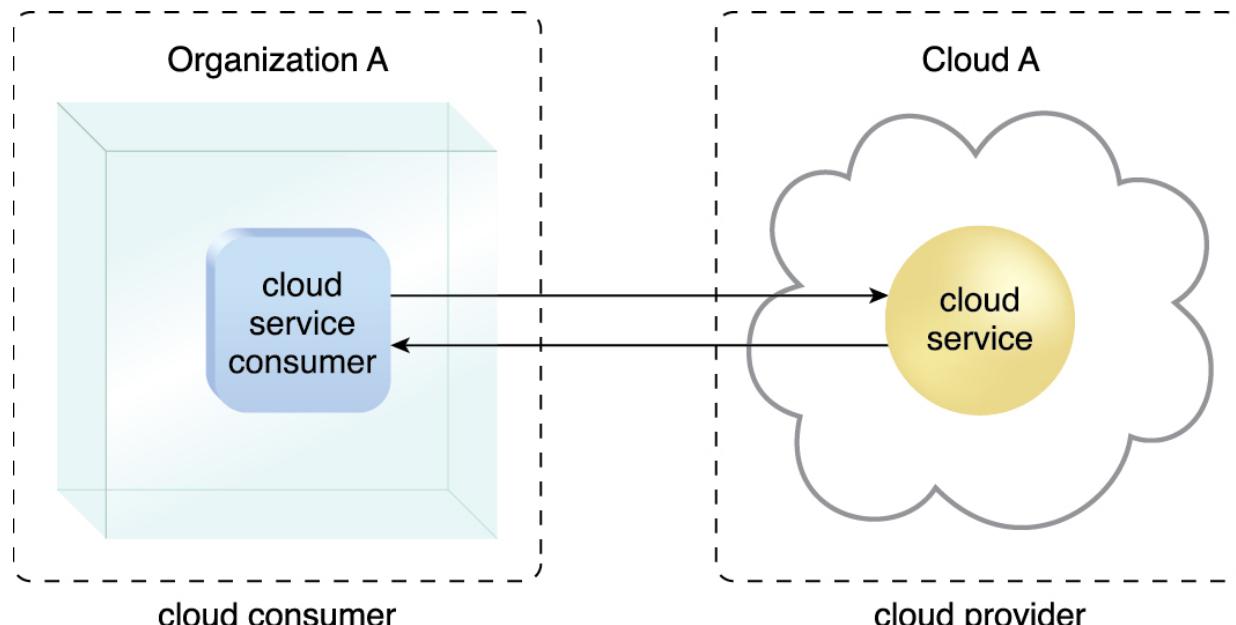


Figure 4.1 A cloud consumer (Organization A) interacts with a cloud service from a cloud provider (that owns Cloud A). Within Organization A, the cloud service consumer is being used to access the cloud service.

Cloud Service Owner

- The person or organization that legally owns a cloud service is called a *cloud service owner*.

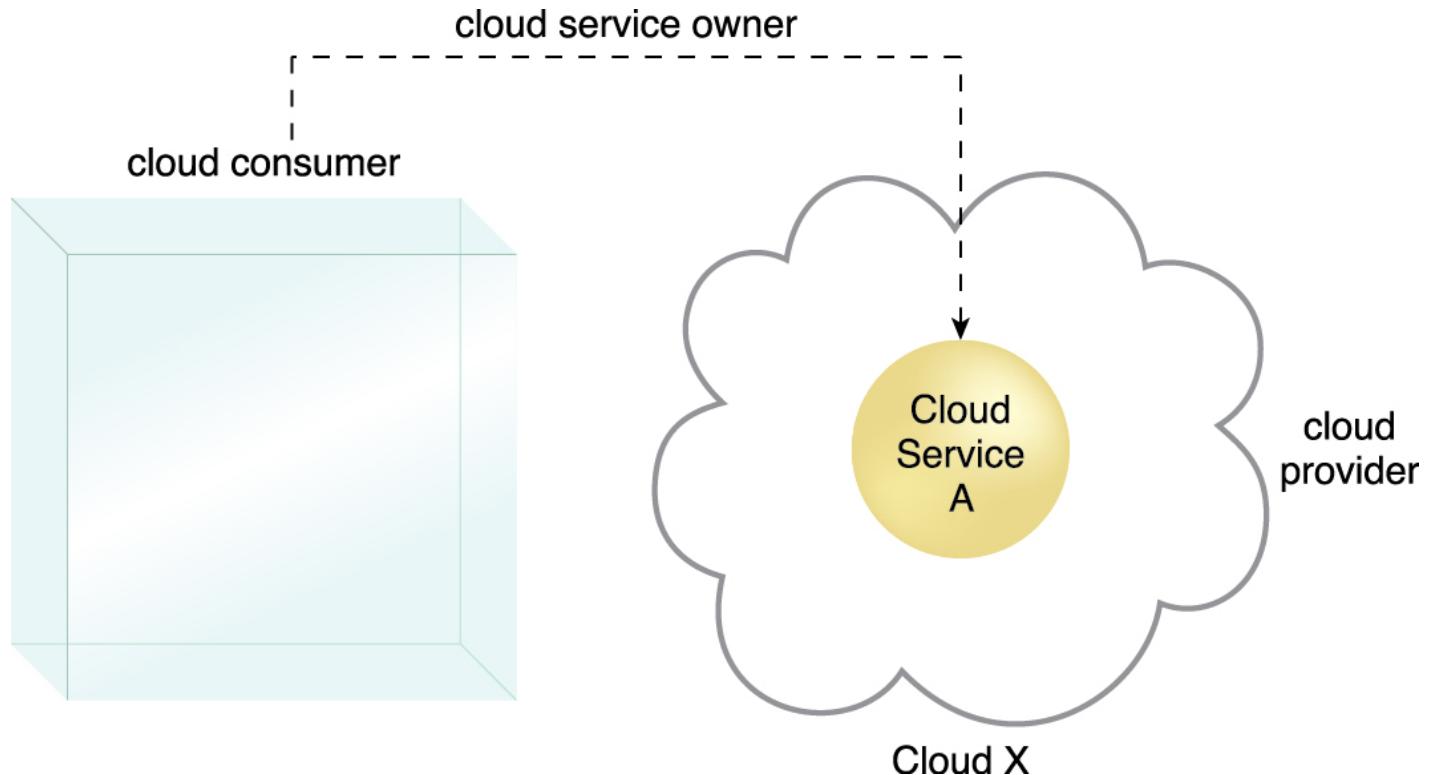


Figure 4.2 A cloud consumer can be a cloud service owner when it deploys its own service in a cloud.

Cloud Service Owner (cont..)

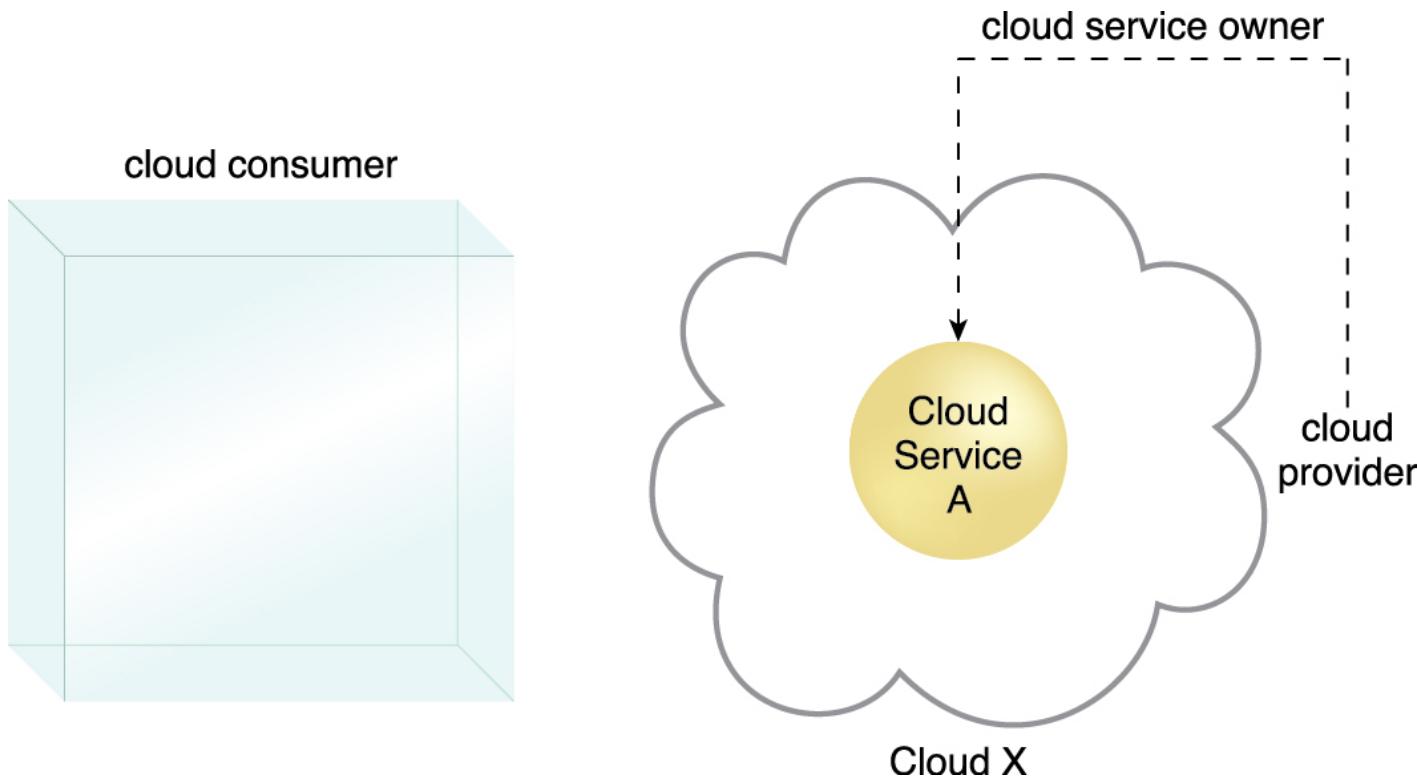


Figure 4.3 A cloud provider becomes a cloud service owner if it deploys its own cloud service, typically for other cloud consumers to use.

Cloud Resource Administrator

- A *cloud resource administrator* is the person or organization responsible for administering a cloud-based IT resource (including cloud services).

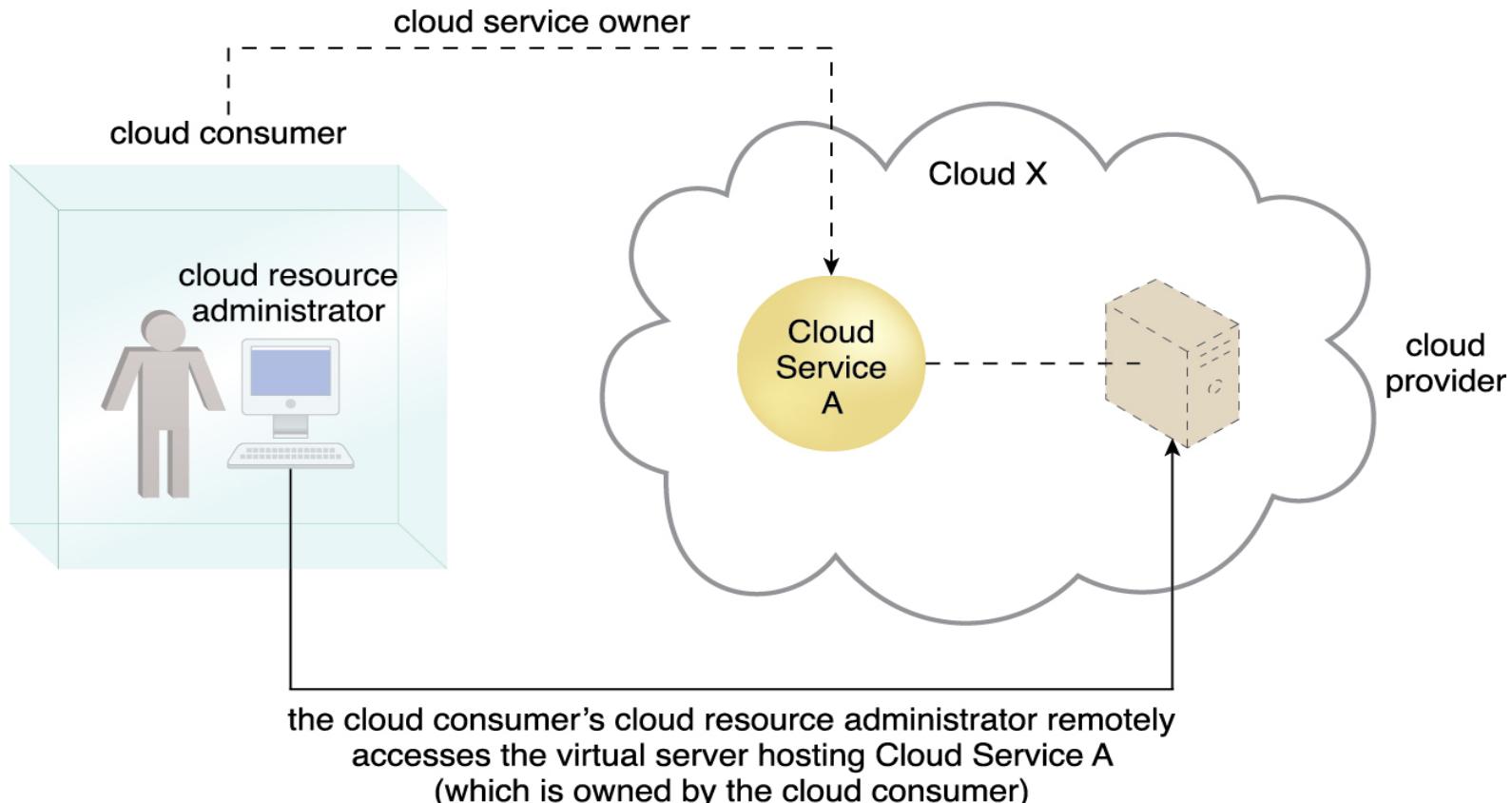


Figure 4.4 A cloud resource administrator can be with a cloud consumer organization and administer remotely accessible IT resources that belong to the cloud consumer.

Cloud Resource Administrator

- cloud resource administrator can be (or belong to) the cloud consumer or cloud provider of the cloud within which the cloud service resides.

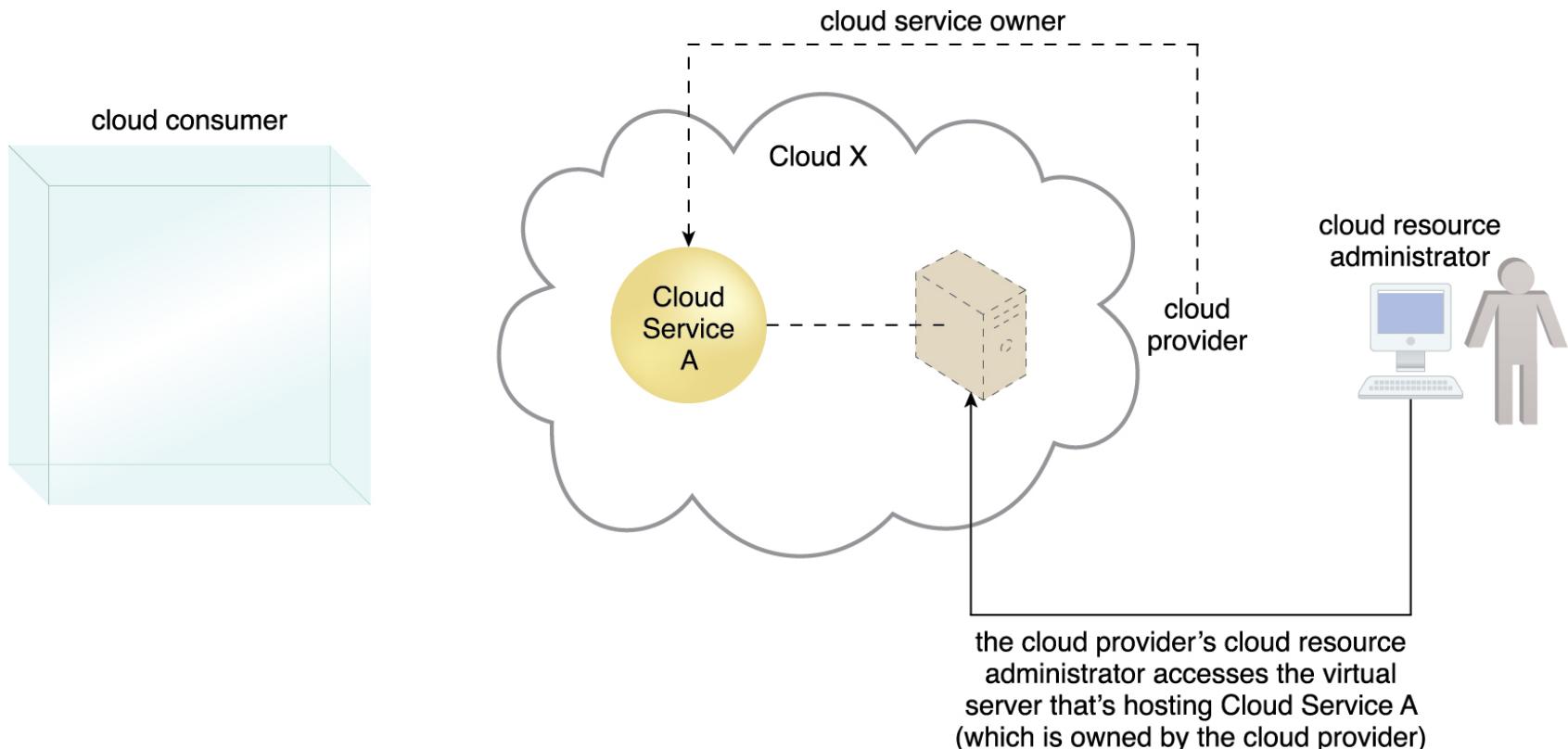


Figure 4.5 A cloud resource administrator can be with a cloud provider organization for which it can administer the cloud provider's internally and externally available IT resources.

Organizational Boundary

- An *organizational boundary* represents the physical perimeter that surrounds a set of IT resources that are owned and governed by an organization.

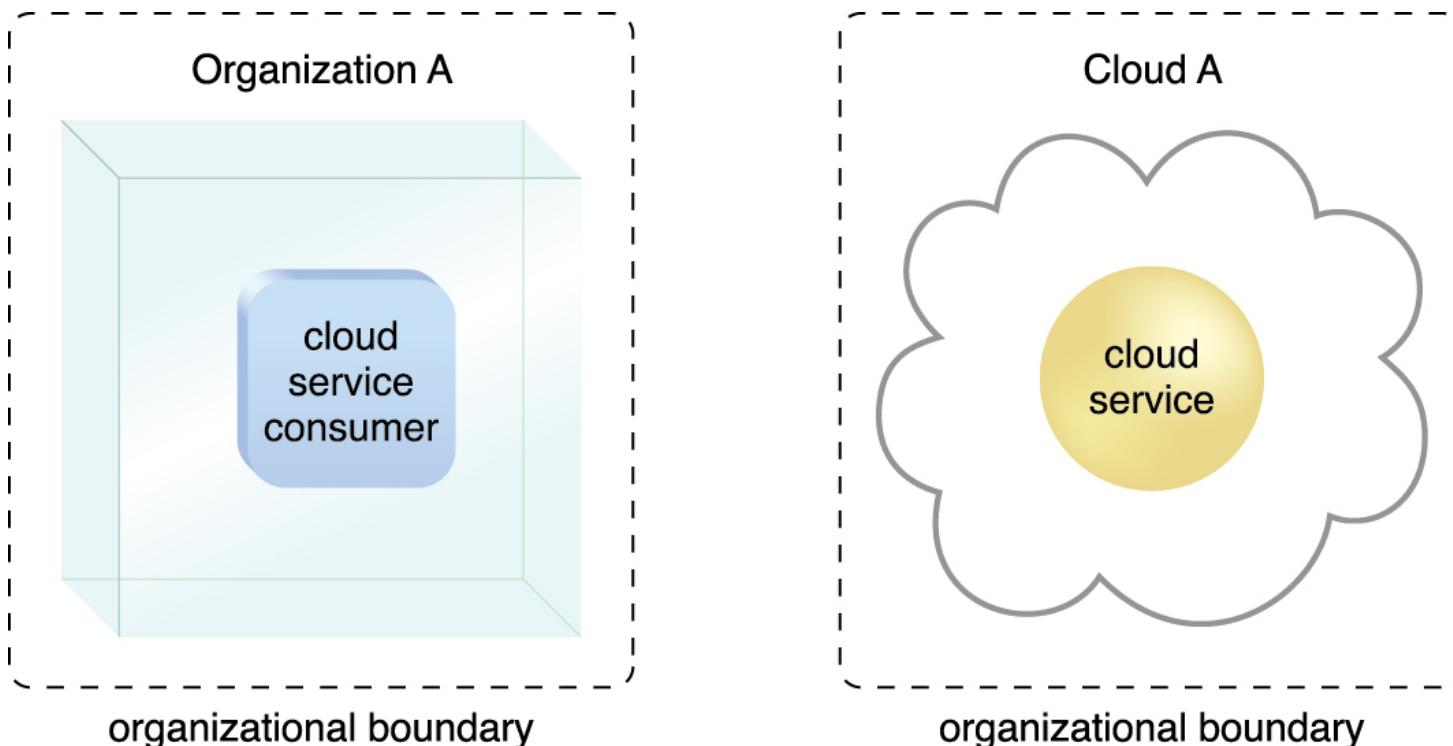


Figure 4.6 Organizational boundaries of a cloud consumer (left), and a cloud provider (right), represented by a broken line notation.

Trust Boundary

- A *trust boundary* is a logical perimeter that typically spans beyond physical boundaries to represent the extent to which IT resources are trusted

trust boundary

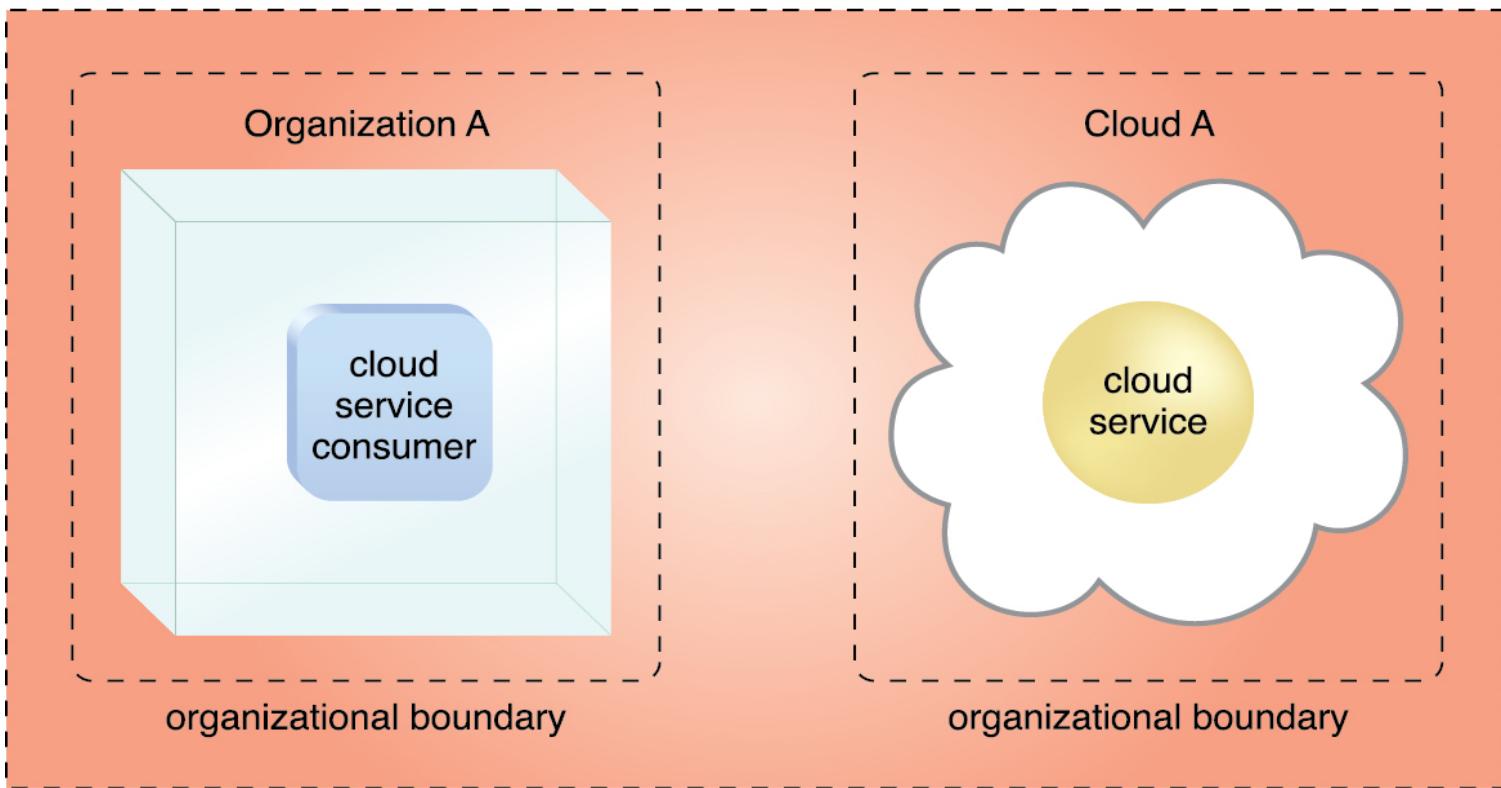


Figure 4.7 An extended trust boundary encompasses the organizational boundaries of the cloud provider and the cloud consumer.

4.2. Cloud Characteristics

- Six specific characteristics are common to the majority of cloud environments:
 - on-demand usage
 - ubiquitous access
 - multitenancy (and resource pooling)
 - elasticity
 - measured usage
 - Resiliency (NIST is excluded)

On-Demand Usage

- A cloud consumer can unilaterally access cloud-based IT resources giving the cloud consumer the freedom to **self-provision** these IT resources or *on-demand usage*.

Ubiquitous Access

- **Ubiquitous access** represents the ability for a cloud service to be widely accessible.
- Establishing ubiquitous access for a cloud service can require support for a range of devices, transport protocols, interfaces, and security technologies.

Multitenancy

- Characteristic of a software program that enables an instance of the program to serve different consumers (tenants) whereby each is isolated from the other, is referred to as *multitenancy*.

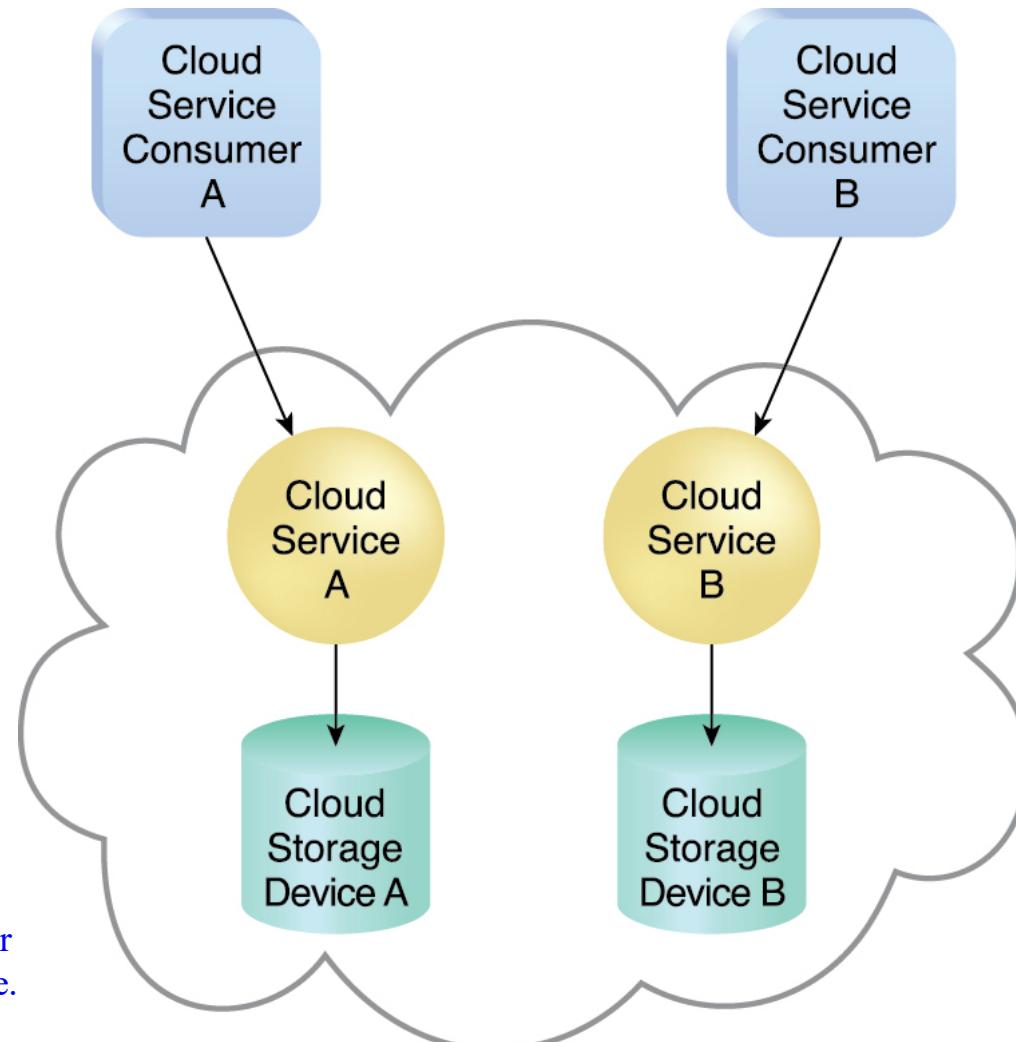


Figure 4.8 In a single-tenant environment, each cloud consumer has a separate IT resource instance.

Multitenancy (cont..)

- Multitenancy allows several cloud consumers to use the same IT resource or its instance while each remains unaware that it may be used by others.

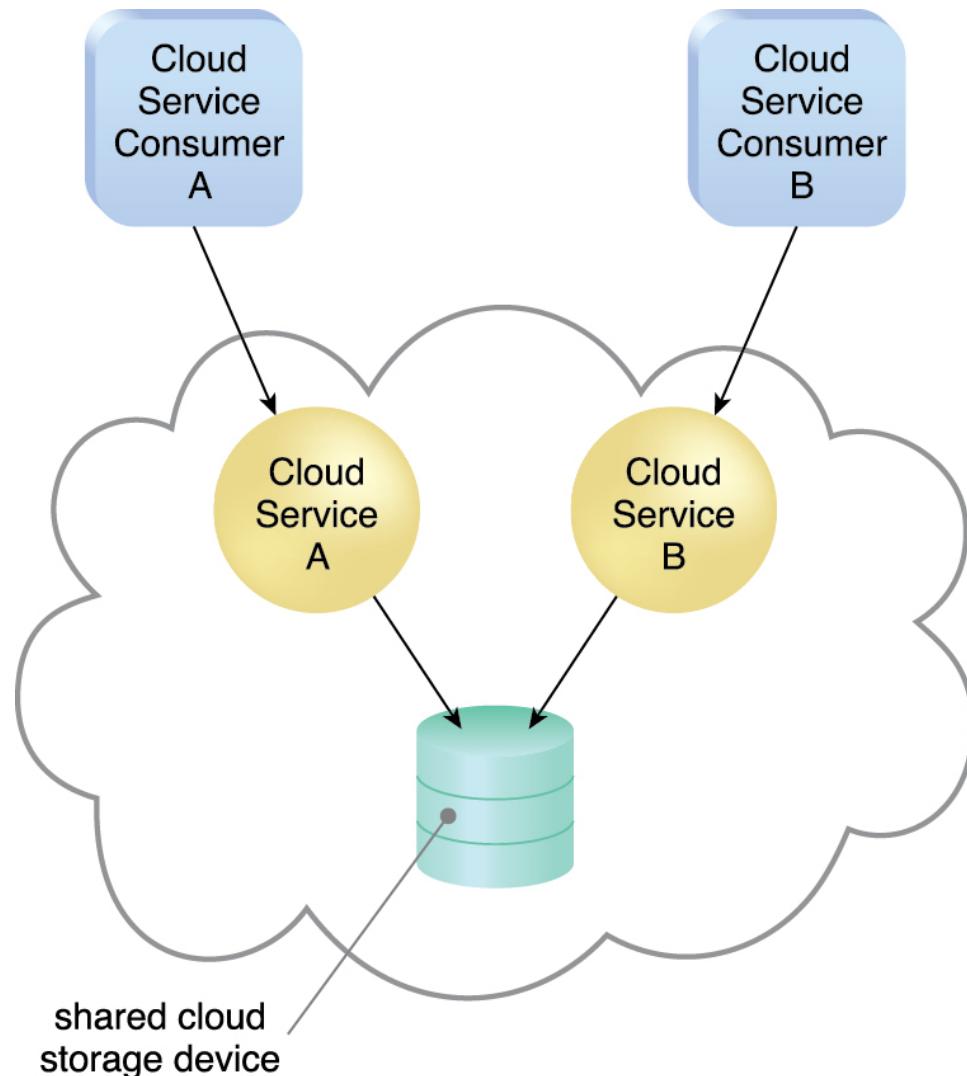


Figure 4.9 In a multitenant environment, a single instance of an IT resource, such as a cloud storage device, serves multiple consumers.

Elasticity

- *Elasticity* is the automated ability of a cloud to transparently scale IT resources, as required in response to runtime conditions or as pre-determined by the cloud consumer or cloud provider.
- Elasticity is often considered a core justification for the adoption of cloud computing.

Measured Usage

- *measured usage* characteristic represents the ability of a cloud platform to keep track of the usage of its IT resources, primarily by cloud consumers.
- Can charge a cloud consumer only for the IT resources actually used and/or for the timeframe during which access to the IT resources was granted.
- is closely related to the on-demand characteristic.
- Measured usage is not limited to tracking statistics for billing purposes. It also encompasses the general monitoring of IT resources and related usage reporting

Resiliency

- *Resilient computing* is a form of failover that distributes redundant implementations of IT resources across physical locations.
- *resiliency* can refer to redundant IT resources within the same cloud (but in different physical locations) or across multiple clouds.

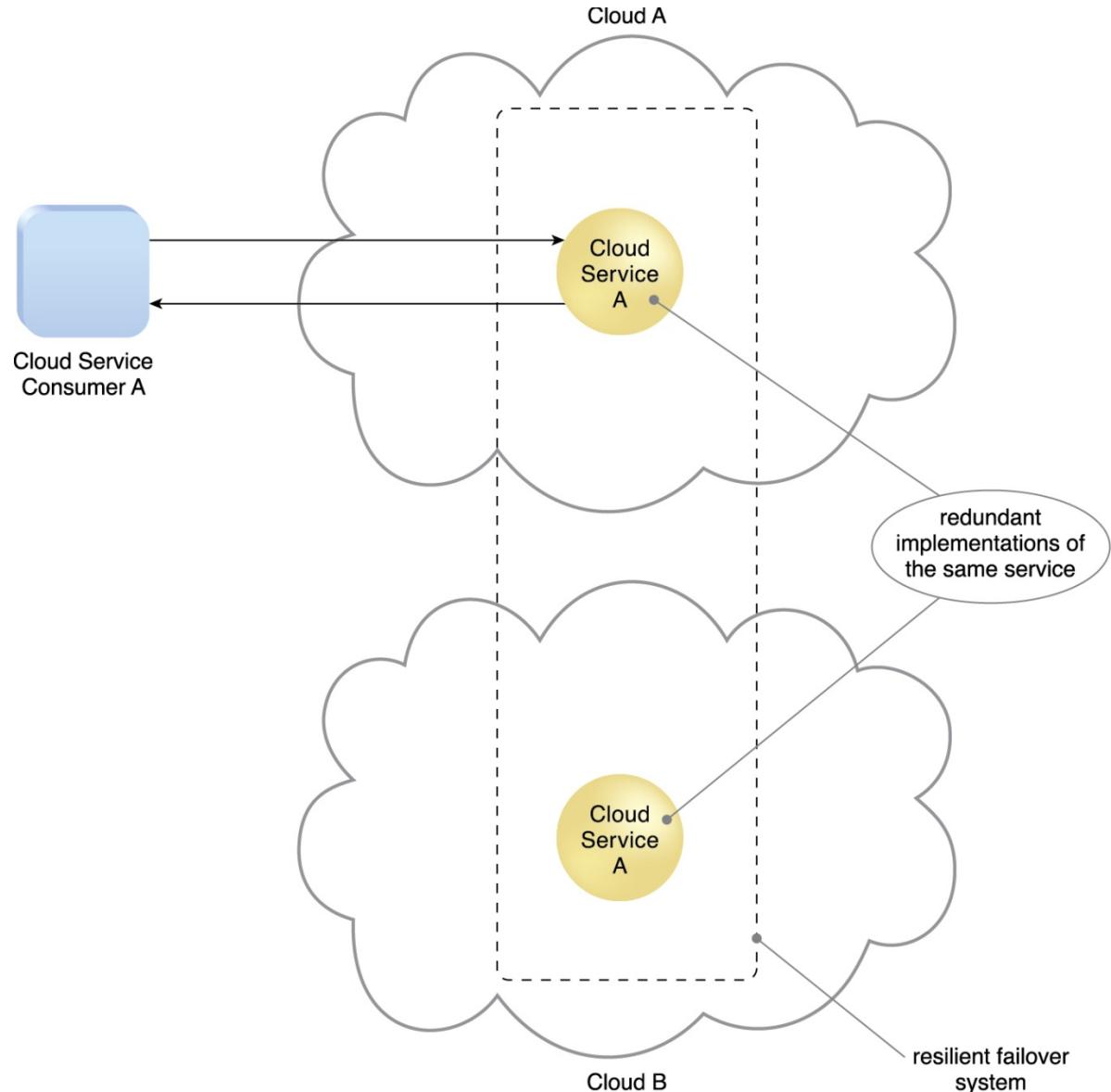


Figure 4.10 A resilient system in which Cloud B hosts a redundant implementation of Cloud Service A to provide failover in case Cloud Service A on Cloud A becomes unavailable.

4.3. Cloud Delivery Models

- A *cloud delivery model* represents a specific, pre-packaged combination of IT resources offered by a cloud provider.
- Three **common cloud delivery models** have become widely established and formalized:
 - Infrastructure-as-a-Service (**IaaS**)
 - Platform-as-a-Service (**PaaS**)
 - Software-as-a-Service (**SaaS**)

Note:

- Many specialized variations of the three base cloud delivery models have emerged, each comprised of a distinct combination of IT resources. Some examples include:
 - Storage-as-a-Service
 - Database-as-a-Service
 - Security-as-a-Service
 - Communication-as-a-Service
 - Integration-as-a-Service
 - Testing-as-a-Service
 - Process-as-a-Service

Infrastructure-as-a-Service (IaaS)

- IaaS delivery model represents a self-contained IT environment comprised of infrastructure-centric IT resources that can be accessed and managed via cloud service-based interfaces and tools.
- Can include hardware, network, connectivity, operating systems, and other “raw” IT resources.
- Are typically [virtualized and packaged into bundles that simplify up-front runtime scaling and customization of the infrastructure](#).
- General purpose of an IaaS environment is to provide cloud consumers with a high level of control and responsibility over its configuration and utilization.
- Used by cloud consumers that require a [high level of control over the cloud-based environment they intend to create](#).
- IaaS environments are generally offered as freshly initialized virtual instances.
- A central and primary IT resource within a typical IaaS environment is the virtual server.
- Virtual servers are leased by specifying server hardware requirements, such as processor capacity, memory, and local storage space.

Infrastructure(hardware)-as-a-Service (IaaS)

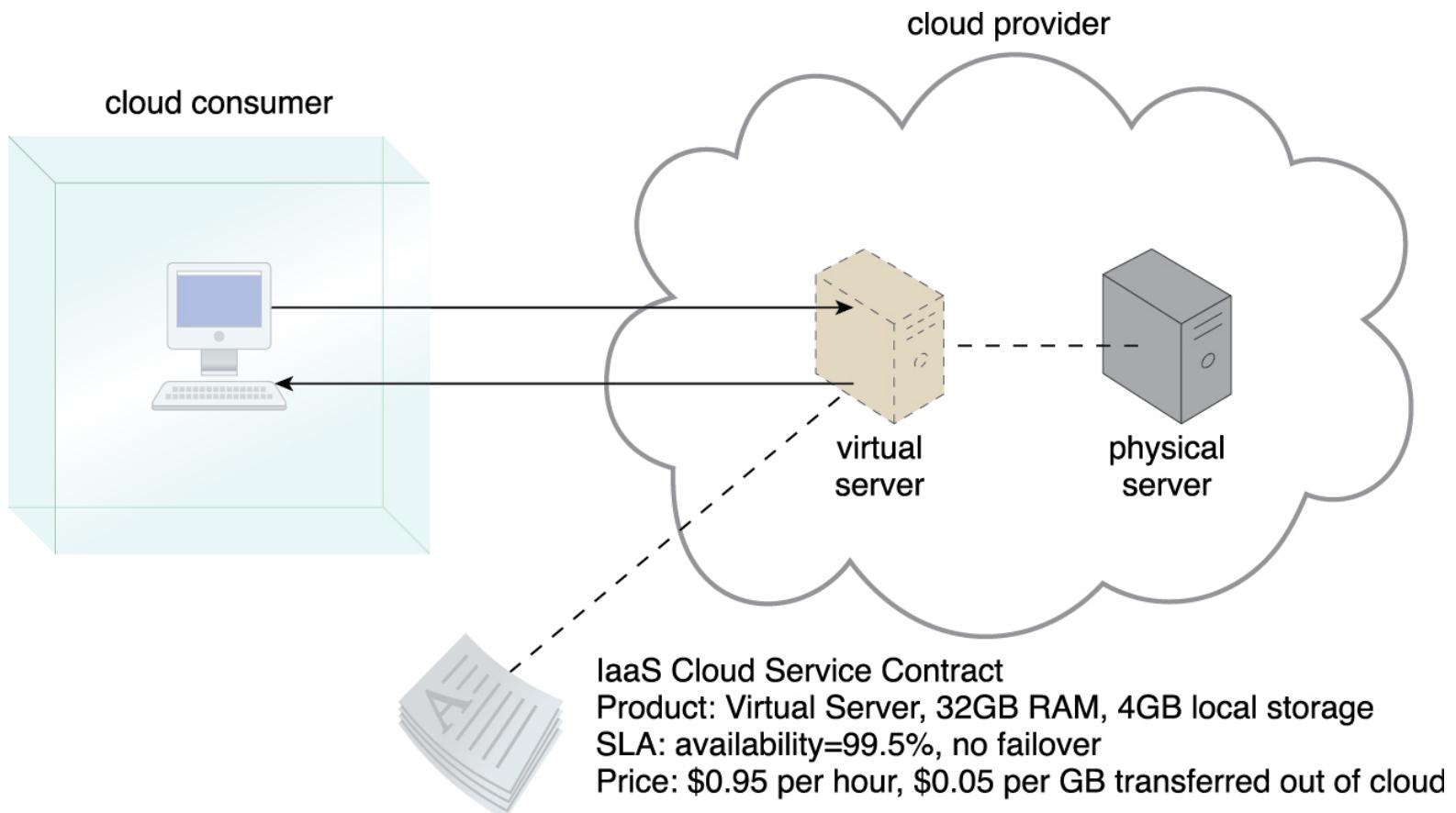


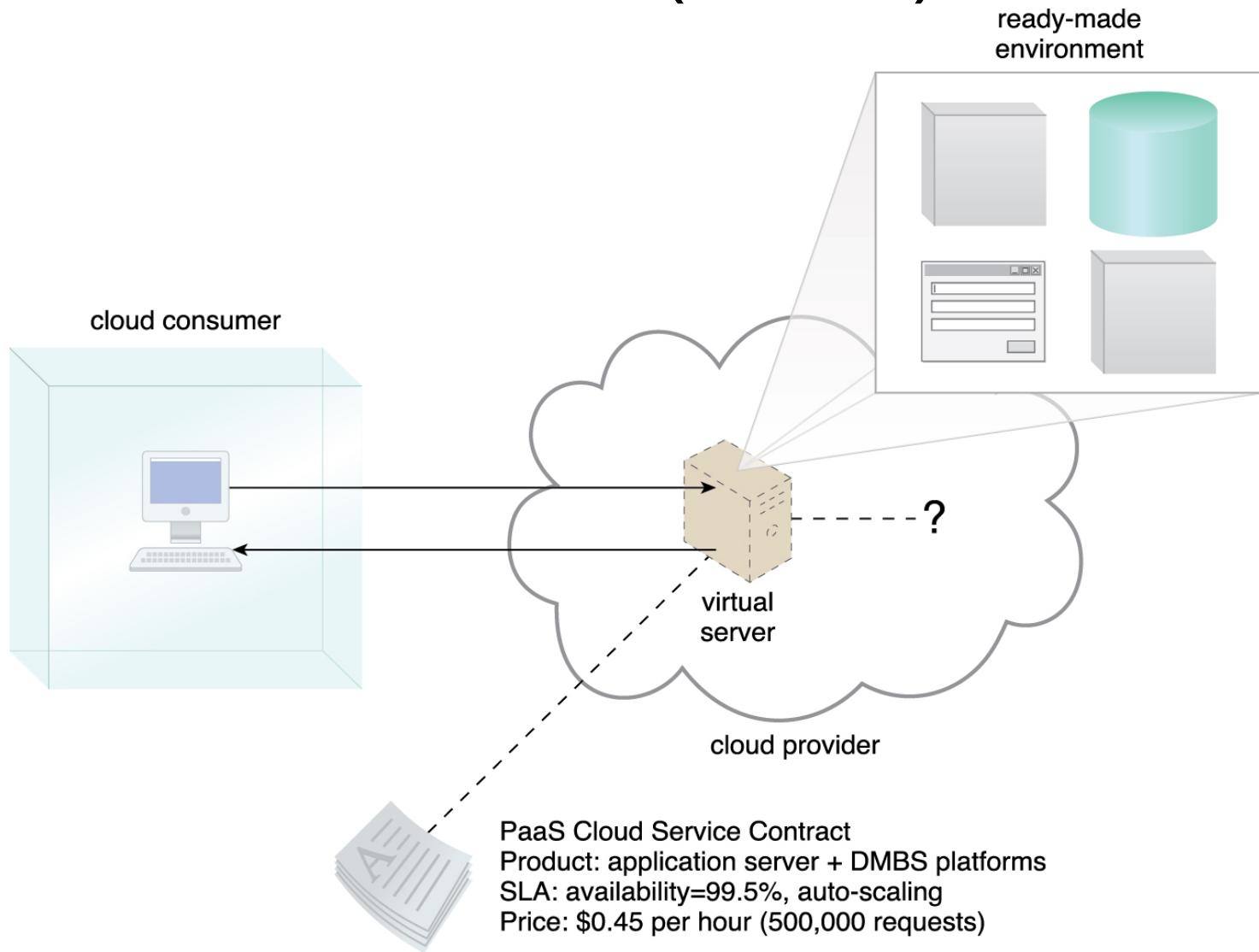
Figure 4.11 A cloud consumer is using a virtual server within an IaaS environment. Cloud consumers are provided with a range of contractual guarantees by the cloud provider, pertaining to characteristics such as capacity, performance, and availability.

Platform-as-a-Service (PaaS)

- PaaS delivery model represents a pre-defined “ready-to-use” environment typically comprised of already deployed and configured IT resources.
- Common reasons a cloud consumer would use and invest in a PaaS environment include:
 - The cloud consumer wants to extend on-premise environments into the cloud for scalability and economic purposes.
 - The cloud consumer uses the ready-made environment to entirely substitute an on-premise environment.
 - The cloud consumer wants to become a cloud provider and deploys its own cloud services to be made available to other external cloud consumers.
- By working within a ready-made platform, the cloud consumer is spared the administrative burden of setting up and maintaining the bare infrastructure IT resources provided via the IaaS model.
- The cloud consumer is granted a lower level of control over the underlying IT resources that host and provision the platform.
- *PaaS products are available with different development stacks. For example, Google App Engine offers a Java and Python-based environment.*

Platform-as-a-Service (PaaS)

Figure 4.12 A cloud consumer is accessing a ready-made PaaS environment. The question mark indicates that the cloud consumer is intentionally shielded from the implementation details of the platform.



Software-as-a-Service (SaaS)

- A software program positioned as a shared cloud service and made available as a “product” or generic utility represents the typical profile of a SaaS offering.
- The SaaS delivery model is typically used to make a reusable cloud service widely available (often commercially) to a range of cloud consumers.
- A cloud consumer is generally granted very limited administrative control over a SaaS implementation.

Software-as-a-Service (SaaS)

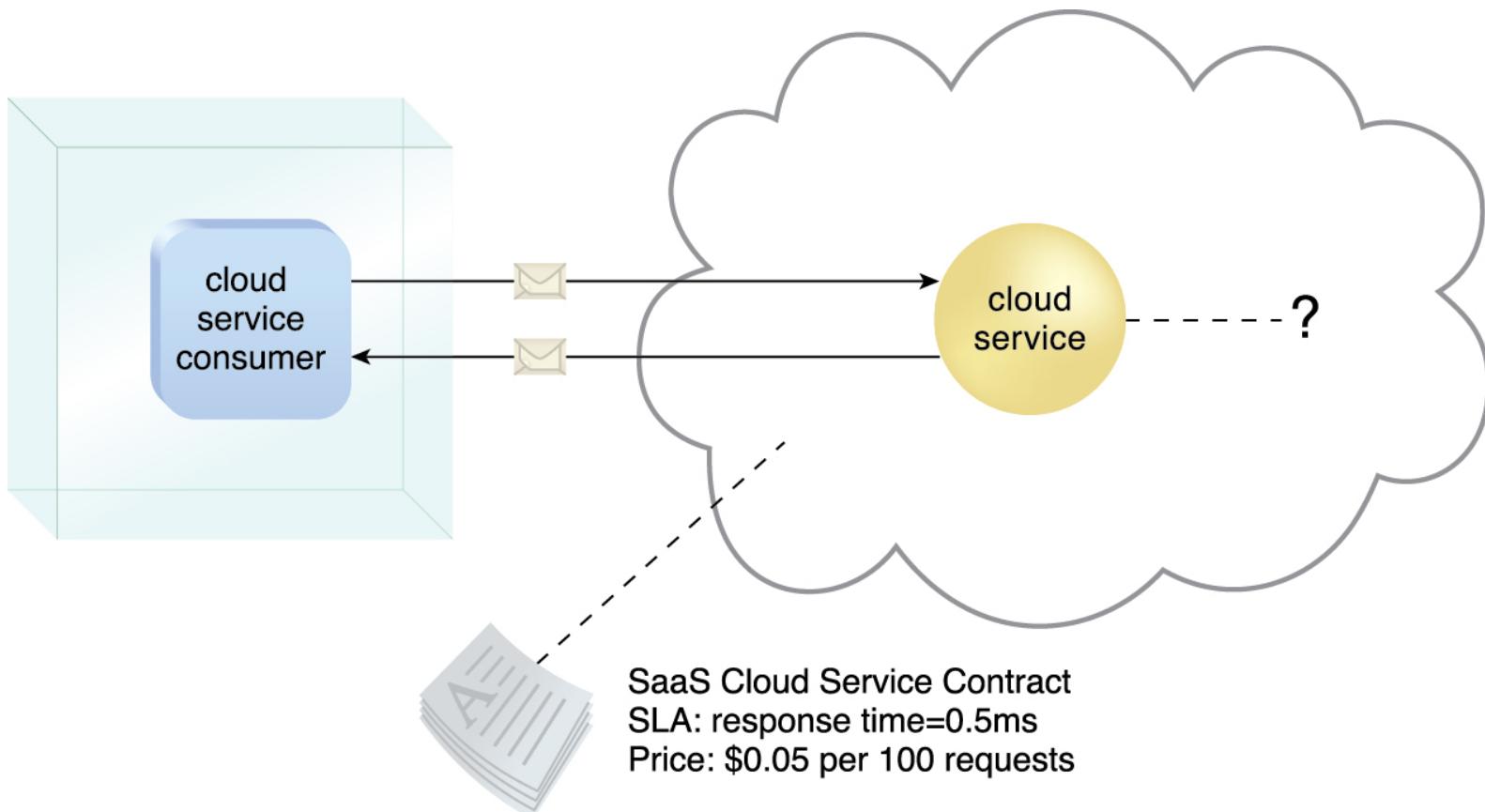


Figure 4.13 The cloud service consumer is given access the cloud service contract, but not to any underlying IT resources or implementation details.

Comparing Cloud Delivery Models

Cloud Delivery Model	Typical Level of Control Granted to Cloud Consumer	Typical Functionality Made Available to Cloud Consumer
SaaS	usage and usage-related configuration	access to front-end user-interface
PaaS	limited administrative	moderate level of administrative control over IT resources relevant to cloud consumer's usage of platform
IaaS	full administrative	full access to virtualized infrastructure-related IT resources and, possibly, to underlying physical IT resources

Table 4.1 A comparison of typical cloud delivery model control levels.

Comparing Cloud Delivery Models

Cloud Delivery Model	Common Cloud Consumer Activities	Common Cloud Provider Activities
SaaS	uses and configures cloud service	implements, manages, and maintains cloud service monitors usage by cloud consumers
PaaS	develops, tests, deploys, and manages cloud services and cloud-based solutions	pre-configures platform and provisions underlying infrastructure, middleware, and other needed IT resources, as necessary monitors usage by cloud consumers
IaaS	sets up and configures bare infrastructure, and installs, manages, and monitors any needed software	provisions and manages the physical processing, storage, networking, and hosting required monitors usage by cloud consumers

Table 4.2 Typical activities carried out by cloud consumers and cloud providers in relation to the cloud delivery models.

Combining Cloud Delivery Models

IaaS + PaaS

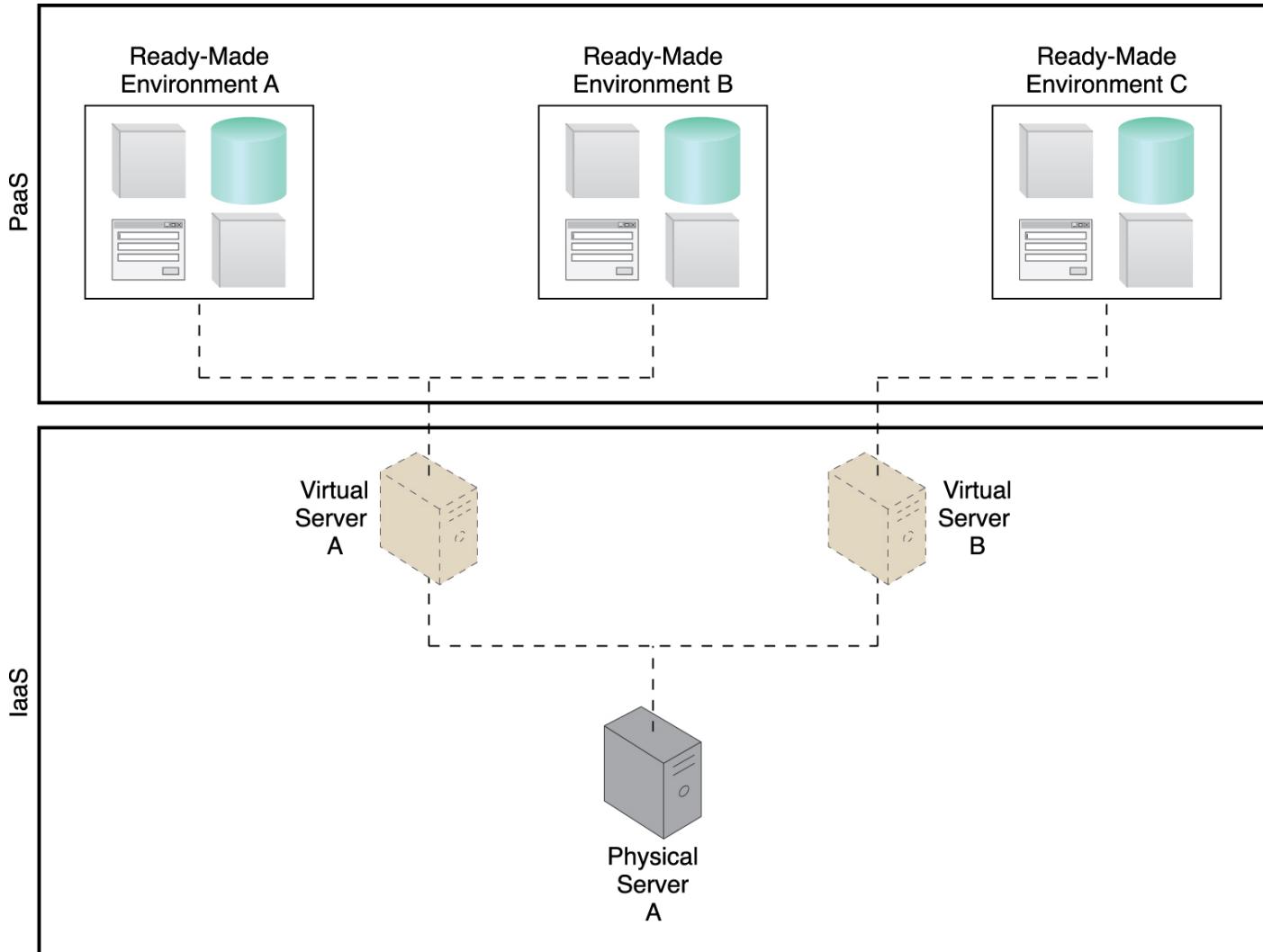


Figure 4.14 A PaaS environment based on the IT resources provided by an underlying IaaS environment.

Combining Cloud Delivery Models

IaaS + PaaS

Cloud provider offering the PaaS environment chose to lease an IaaS environment from a *different* cloud provider.

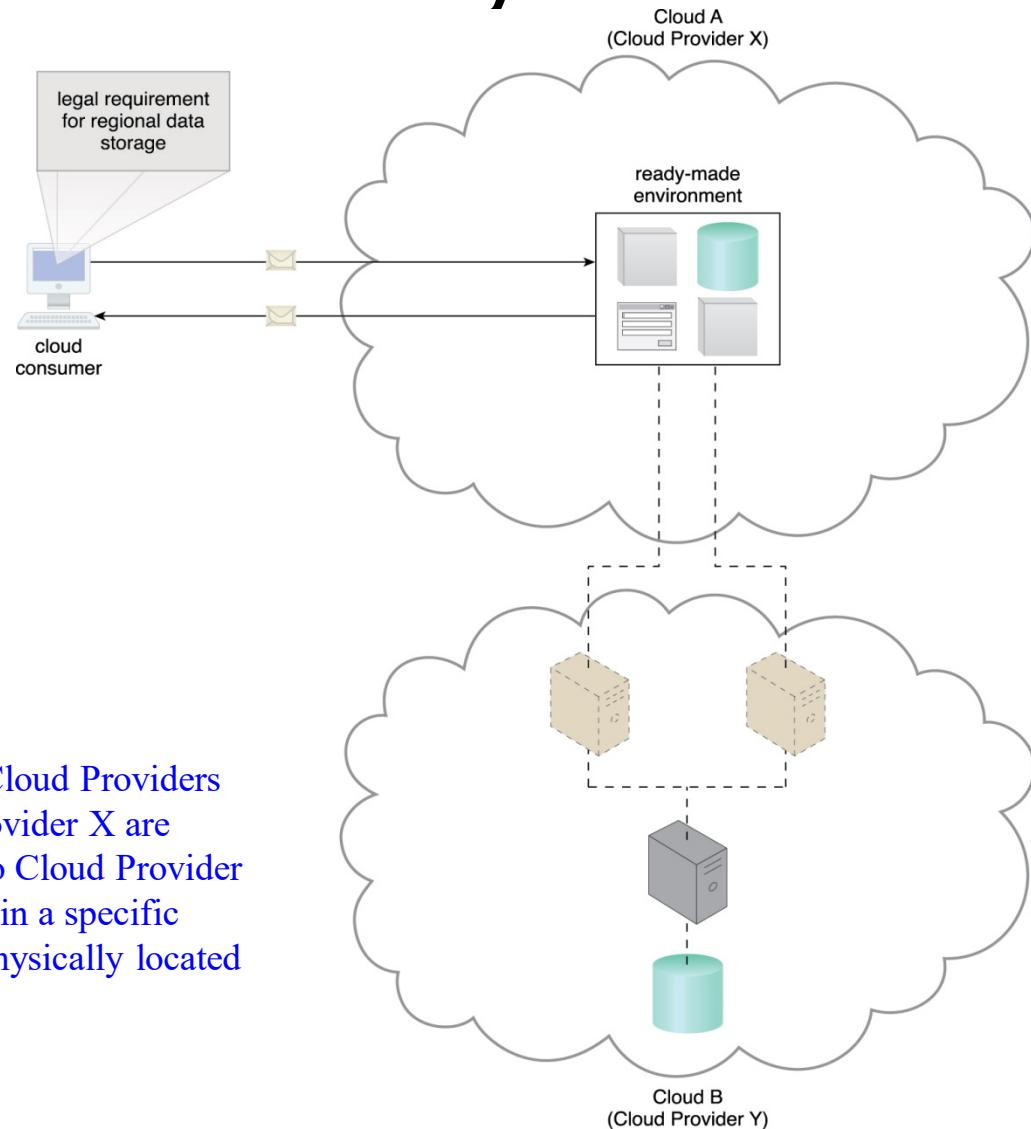


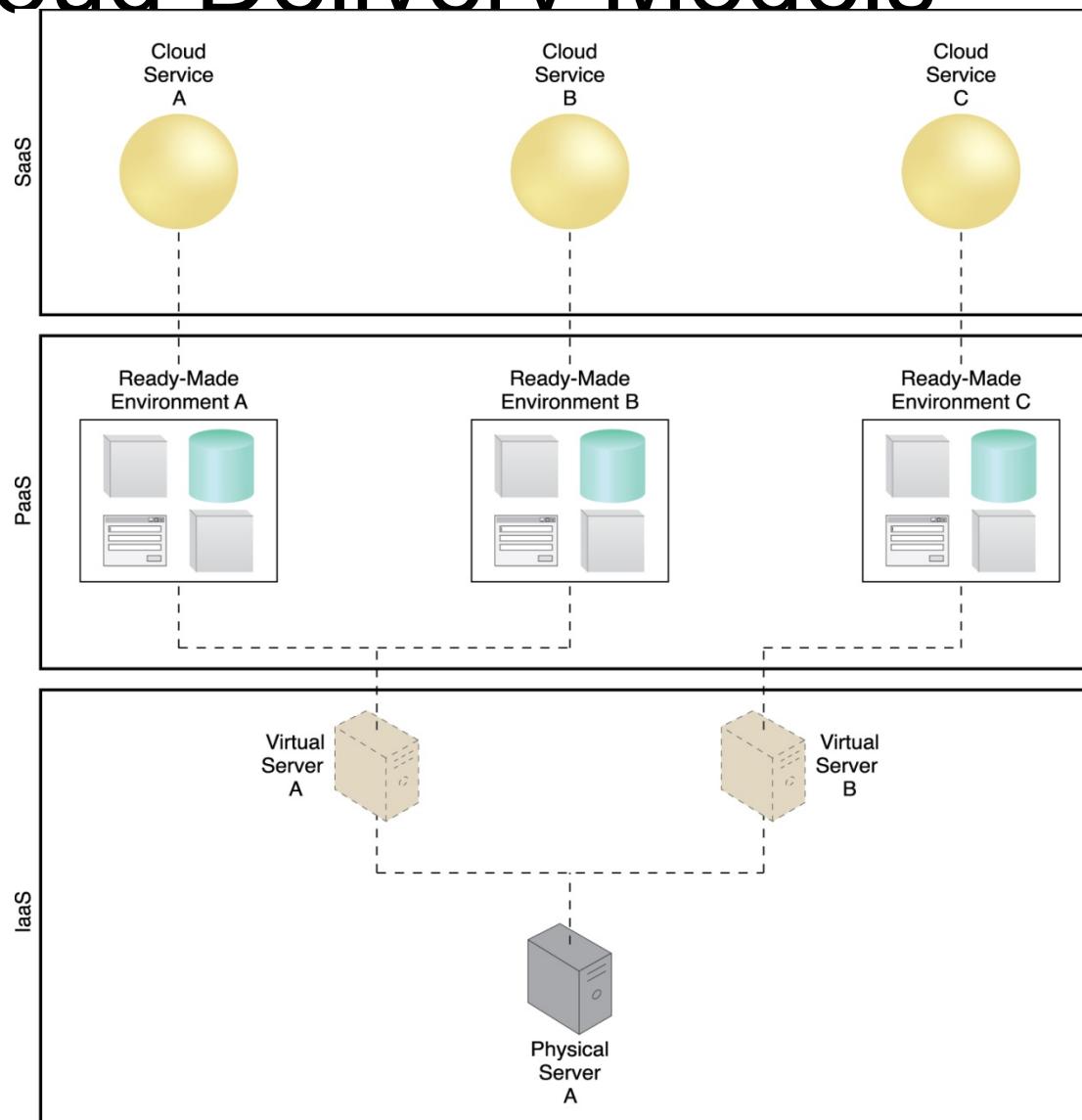
Figure 4.15 An example of a contract between Cloud Providers X and Y, in which services offered by Cloud Provider X are physically hosted on virtual servers belonging to Cloud Provider Y. Sensitive data that is legally required to stay in a specific region is physically kept in Cloud B, which is physically located in that region.

Combining Cloud Delivery Models

IaaS + PaaS + SaaS

Ready-made environment provided by the PaaS environment can be used by the cloud consumer organization to develop and deploy its own SaaS cloud services that it can then make available as commercial products

Figure 4.16 A simple layered view of an architecture comprised of IaaS and PaaS environments hosting three SaaS cloud service implementations.



4.4. Cloud Deployment Models

- A cloud deployment model represents a specific type of cloud environment, primarily distinguished by ownership, size, and access.
- There are four common cloud deployment models:
 - Public cloud
 - Community cloud
 - Private cloud
 - Hybrid cloud

Public Clouds

- A *public cloud* is a publicly accessible cloud environment owned by a third-party cloud provider. The IT resources on public clouds are usually provisioned via the previously described cloud delivery models and are generally offered to cloud consumers at a cost or are commercialized via other avenues (such as advertisement).

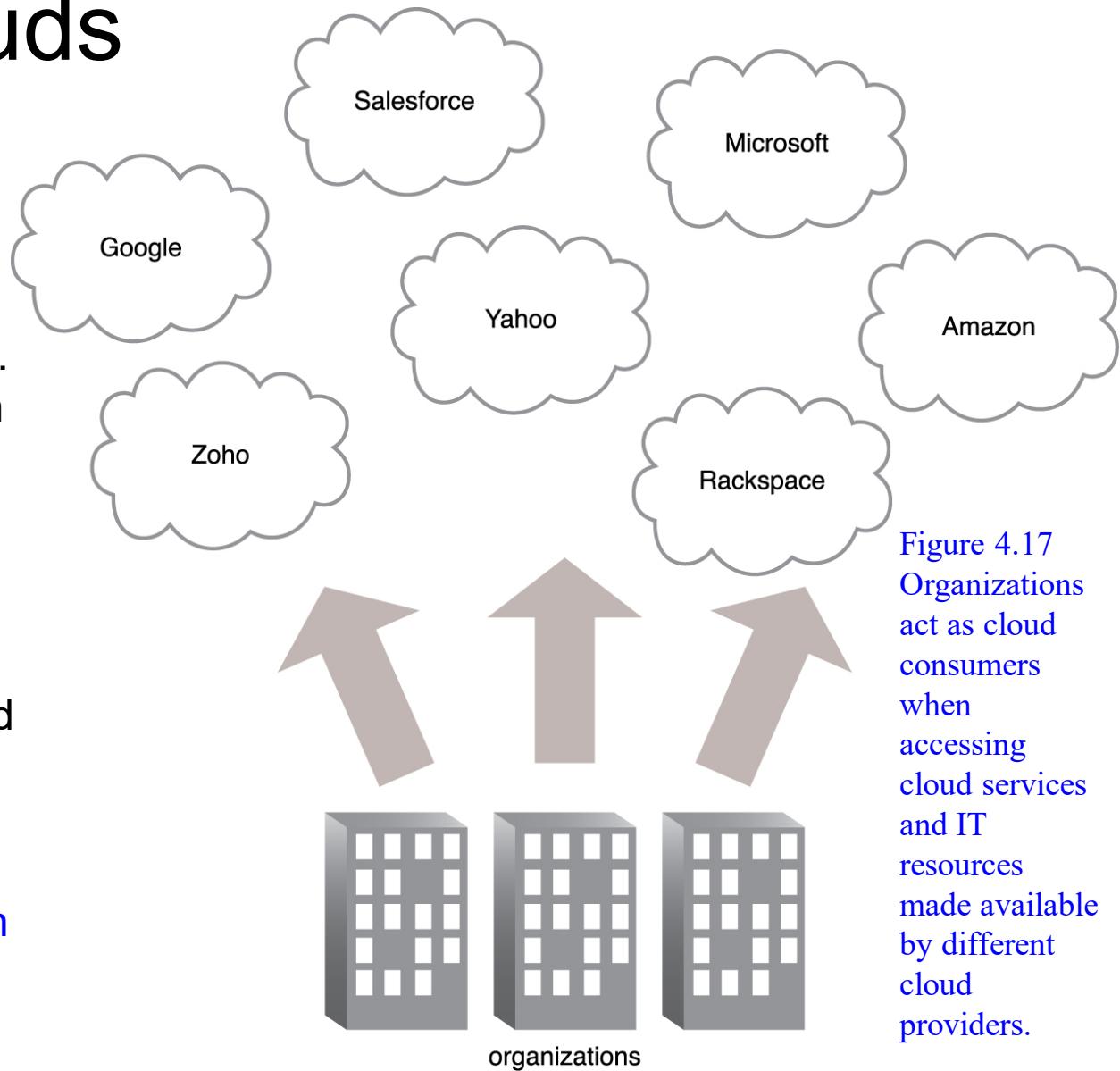


Figure 4.17
Organizations act as cloud consumers when accessing cloud services and IT resources made available by different cloud providers.

Community Clouds

- A community cloud is similar to a public cloud except that its access is limited to a specific community of cloud consumers.

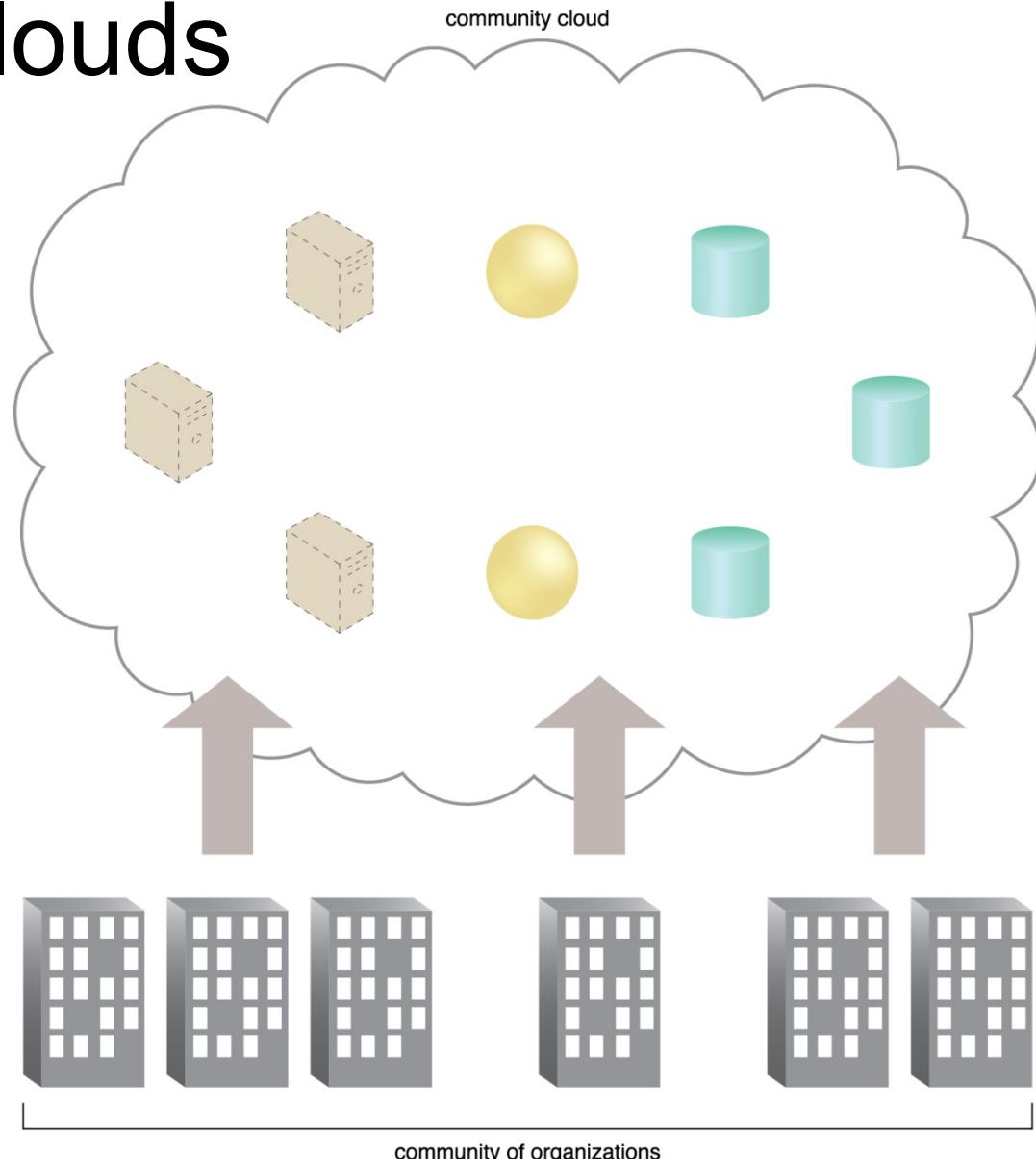


Figure 4.18 An example of a “community” of organizations accessing IT resources from a community cloud.

Private Clouds

- A private cloud is owned by a single organization. Private clouds enable an organization to use cloud computing technology as a means of centralizing access to IT resources by different parts, locations, or departments of the organization.
- Who would manage?
- is “on-premises or cloud-based?

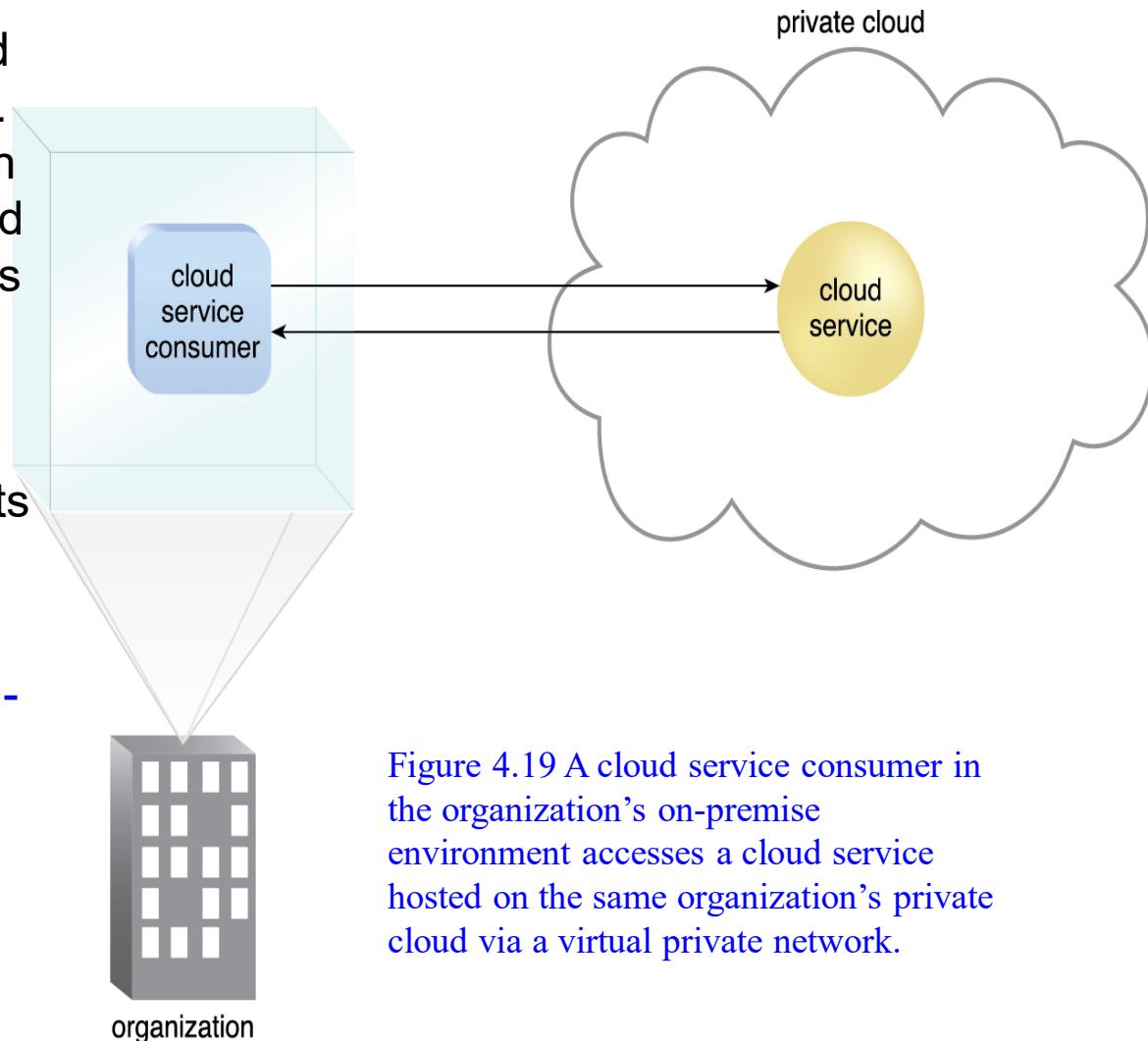
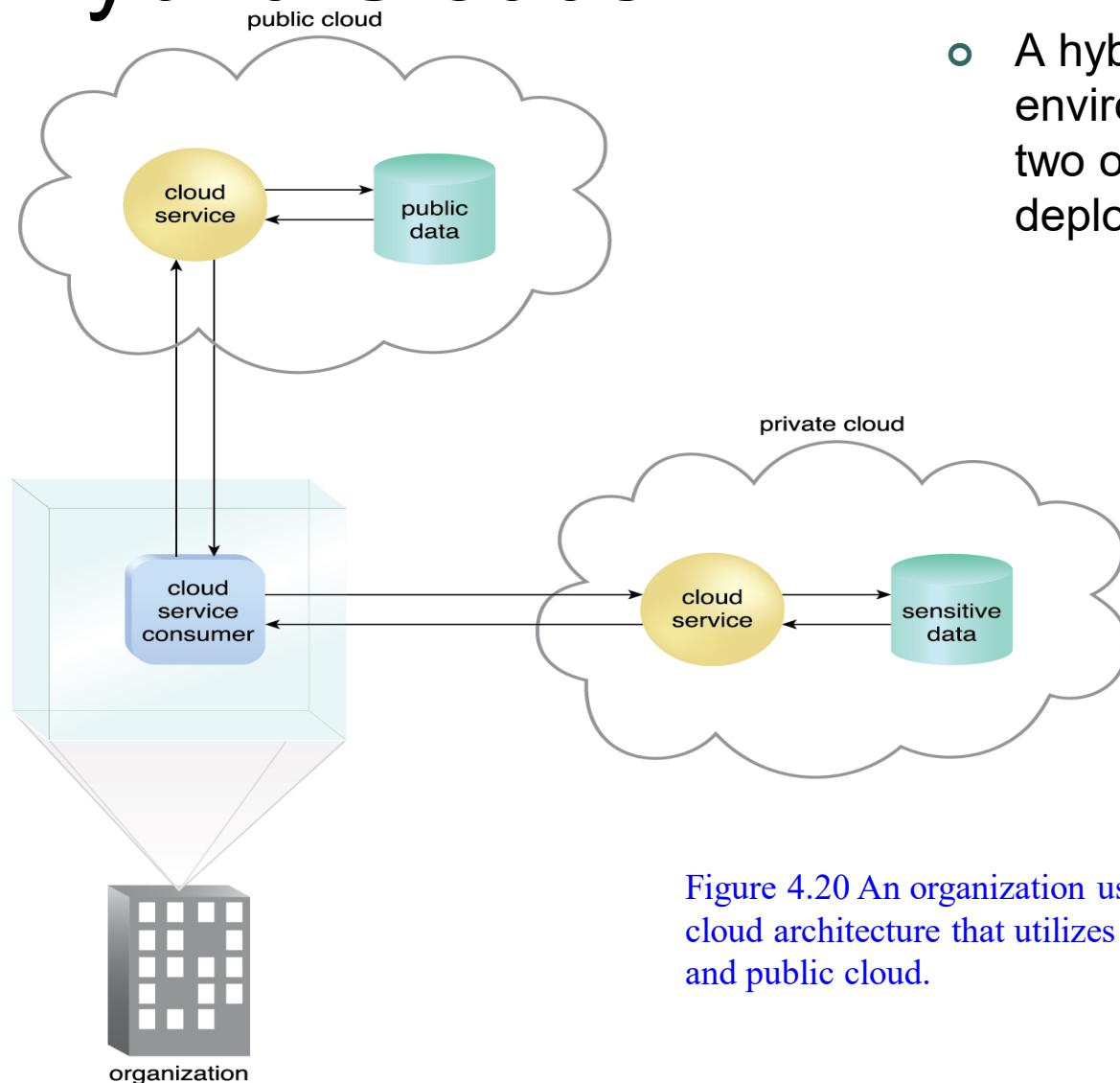


Figure 4.19 A cloud service consumer in the organization's on-premise environment accesses a cloud service hosted on the same organization's private cloud via a virtual private network.

Hybrid Clouds



- A hybrid cloud is a cloud environment comprised of two or more different cloud deployment models.

Figure 4.20 An organization using a hybrid cloud architecture that utilizes both a private and public cloud.

Other Cloud Deployment Models

- Additional variations of the four base cloud deployment models can exist. Examples include:
 - *Virtual Private Cloud* – Also known as a “**dedicated cloud**” or “**hosted cloud**,” this model results in a self-contained cloud environment hosted and managed by a public cloud provider, and made available to a cloud consumer.
 - *Inter-Cloud* – This model is based on an architecture comprised of two or more **inter-connected** clouds.

Summary

- Common roles associated with cloud-based interaction and relationships include the cloud provider, cloud consumer, cloud service owner, and cloud resource administrator.
- An organizational boundary represents the physical scope of IT resources owned and governed by an organization. A trust boundary is the logical perimeter that encompasses the IT resources trusted by an organization.
- On-demand usage is the ability of a cloud consumer to self-provision and use necessary cloud-based services without requiring cloud provider interaction. This characteristic is related to measured usage, which represents the ability of a cloud to measure the usage of its IT resources.
- Ubiquitous access allows cloud-based services to be accessed by diverse cloud service consumers, while multitenancy is the ability of a single instance of an IT resource to transparently serve multiple cloud consumers simultaneously.
- The elasticity characteristic represents the ability of a cloud to transparently and automatically scale IT resources out or in. Resiliency pertains to a cloud's inherent failover features.

Summary (cont..)

- The IaaS cloud delivery model offers cloud consumers a high level of administrative control over “raw” infrastructure-based IT resources.
- The PaaS cloud delivery model enables a cloud provider to offer a pre-configured environment that cloud consumers can use to build and deploy cloud services and solutions, albeit with decreased administrative control.
- SaaS is a cloud delivery model for shared cloud services that can be positioned as commercialized products hosted by clouds.
- Different combinations of IaaS, PaaS, and SaaS are possible, depending on how cloud consumers and cloud providers choose to leverage the natural hierarchy established by these base cloud delivery models.

Summary (cont..)

- A public cloud is owned by a third party and generally offers commercialized cloud services and IT resources to cloud consumer organizations.
- A private cloud is owned by an individual organization and resides within the organization's premises.
- A community cloud is normally limited for access by a group of cloud consumers that may also share responsibility in its ownership.
- A hybrid cloud is a combination of two or more other cloud deployment models.

Cloud-Enabling Technology

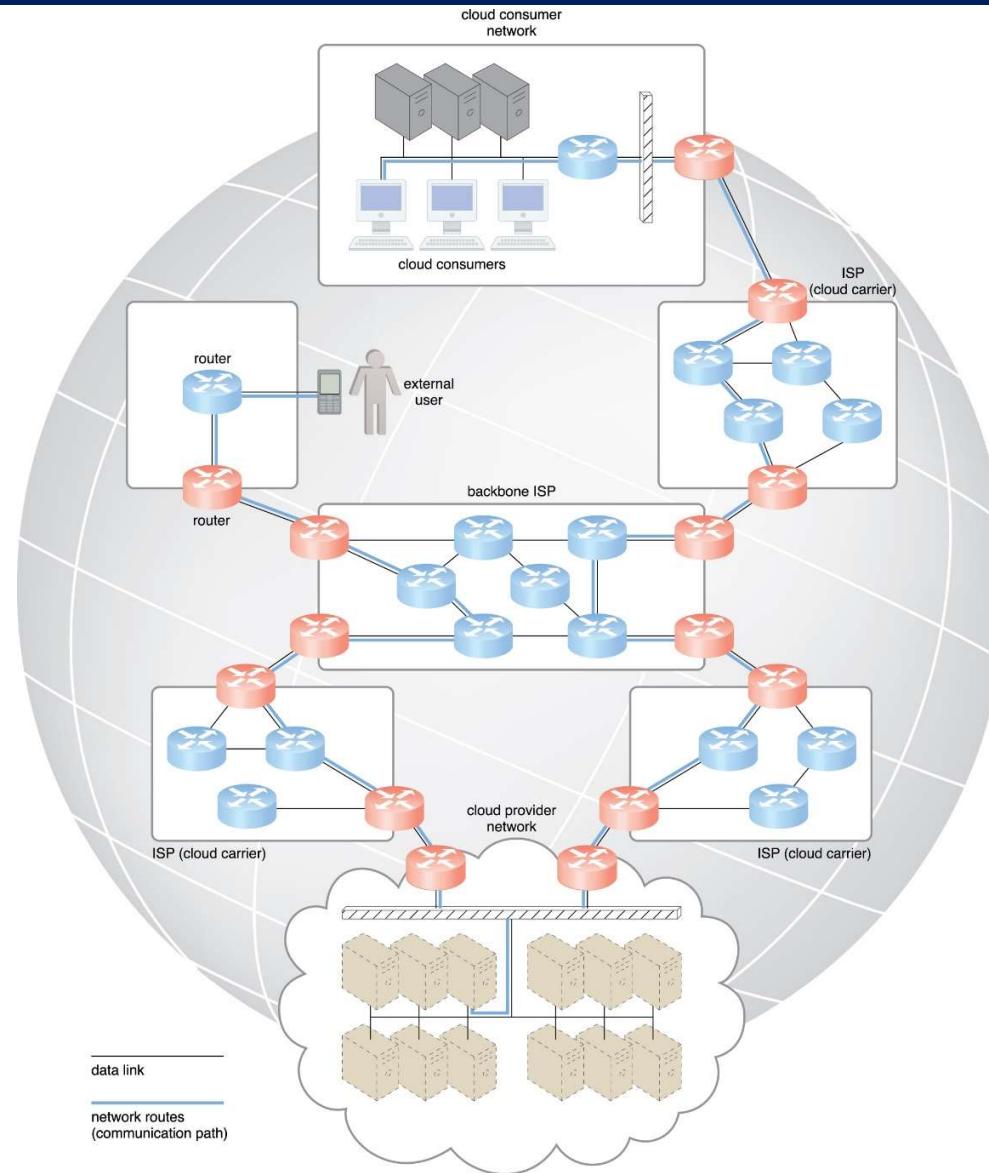
Enabling technologies

1. Broadband networks and internet architecture
2. Data center technology
3. Virtualization technology
4. Web technology
5. Multitenant technology

1. Broadband networks & Internet architecture

- All clouds must be connected to a network
- Internet's largest backbone networks, established and deployed by ISPs, are interconnected by core routers
 - ISP: internet service provider

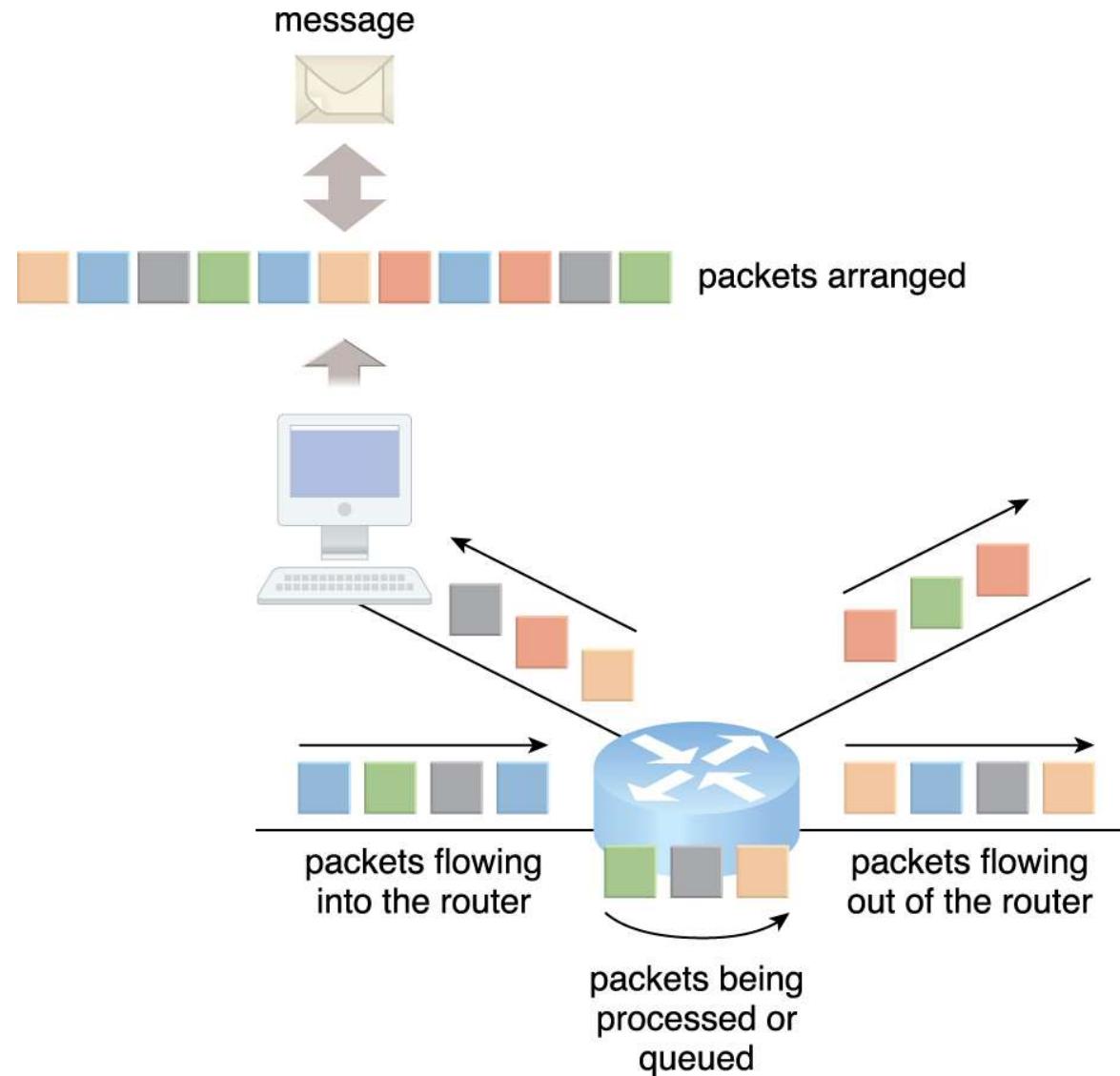
Internet connecting provider and consumer



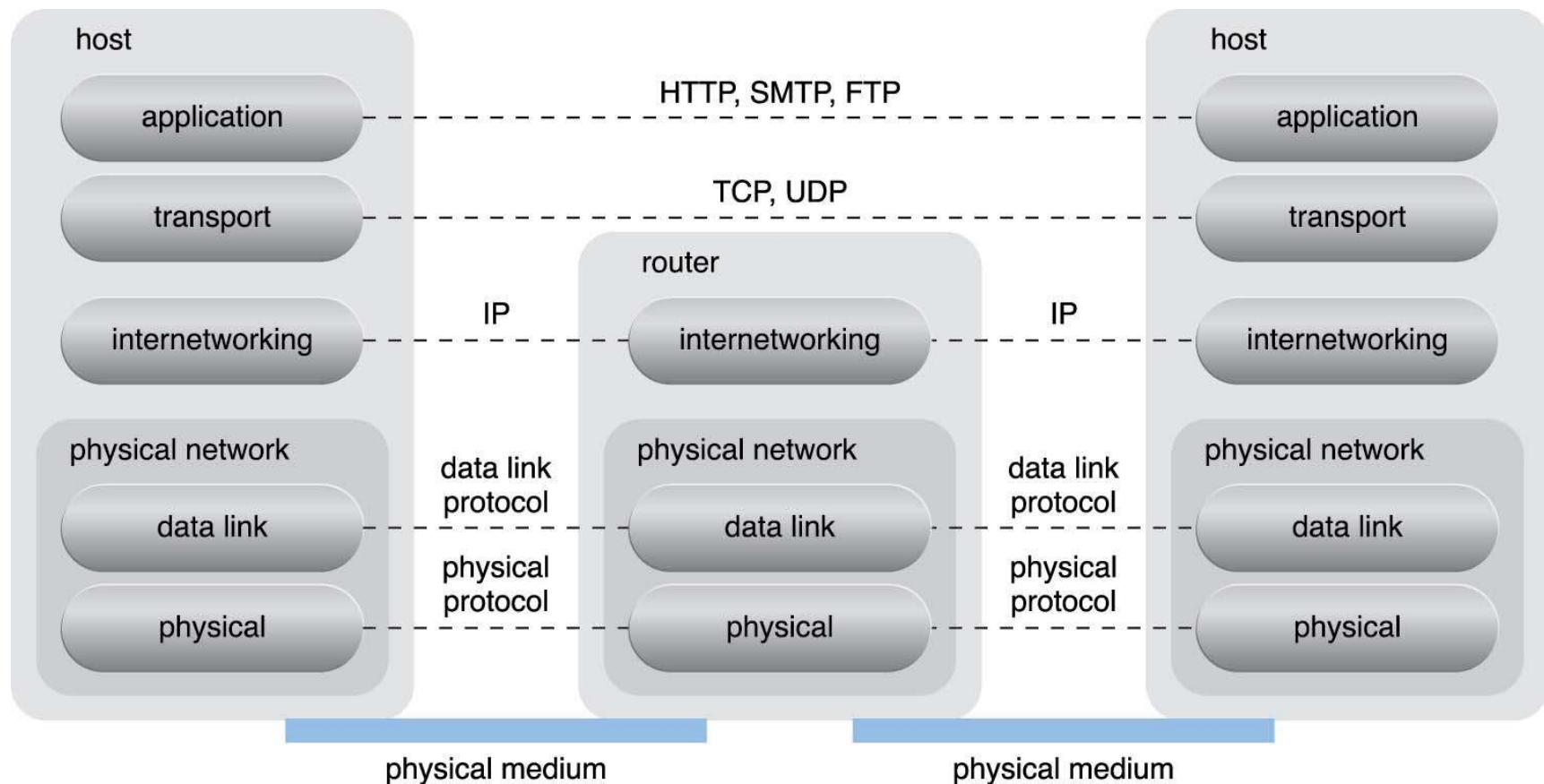
Two fundamental components

- Connectionless packet switching
 - End-to-end (sender-receiver pair) data flows are divided into packets of a limited size
 - Packets are processed through network switches and routers, then queued and forwarded from one intermediary node to the next
- Router-based interconnectivity
 - A router is a device that is connected to multiple networks through which it forwards packets
 - ❖ Each packet is individually processed
 - Use multiple alternative network routes

Packets travelling through Internet



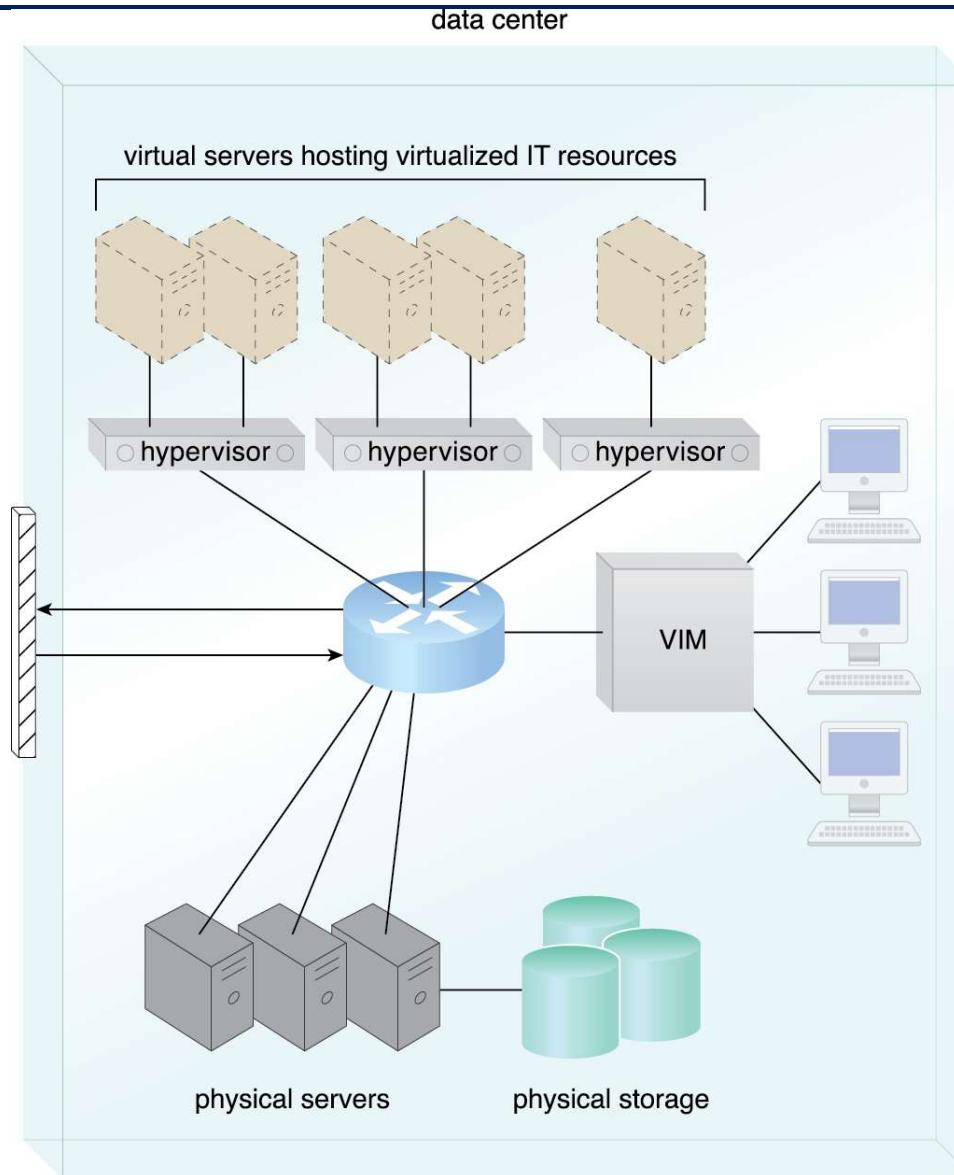
Internet reference model



2. Data Center Technology

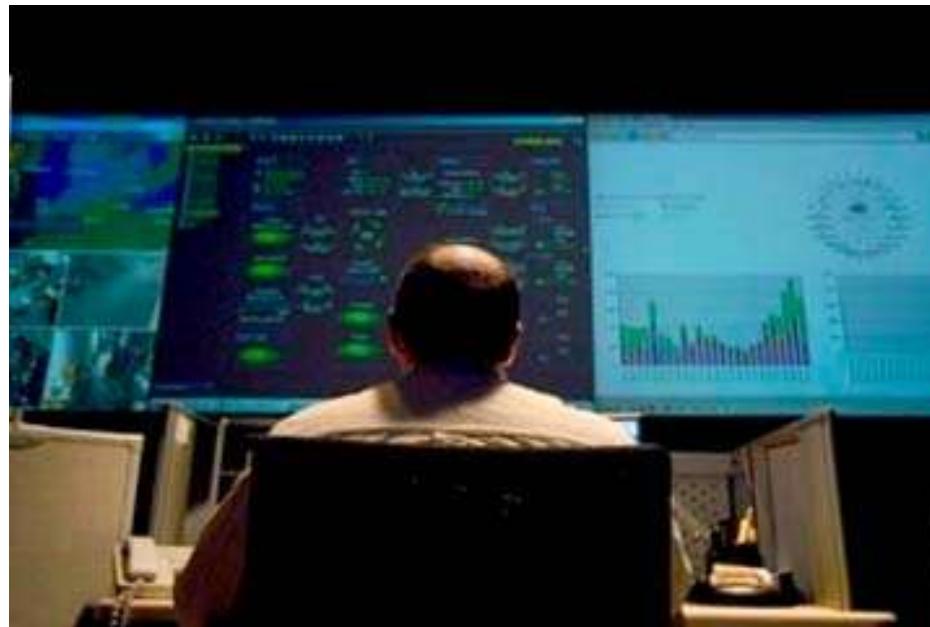
- A data center is a facility used to house computer systems and associated components, such as telecommunications and storage systems
 - Virtualization
 - Standardization and Modularity
 - Automation
 - Remote Operation and Management

Virtualization



Standardization and Modularity

- Data centers are built upon standardized commodity hardware and designed with modular architecture.



Supercomputer vs. data center

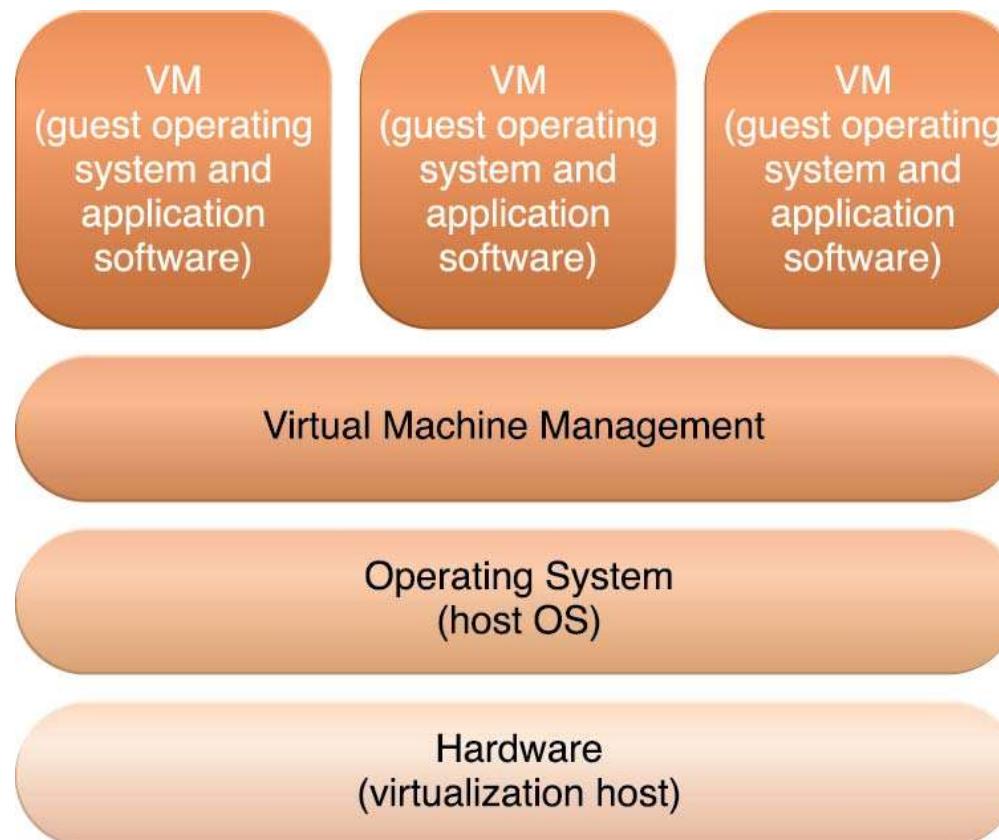
- Handouts

3. Virtualization technology

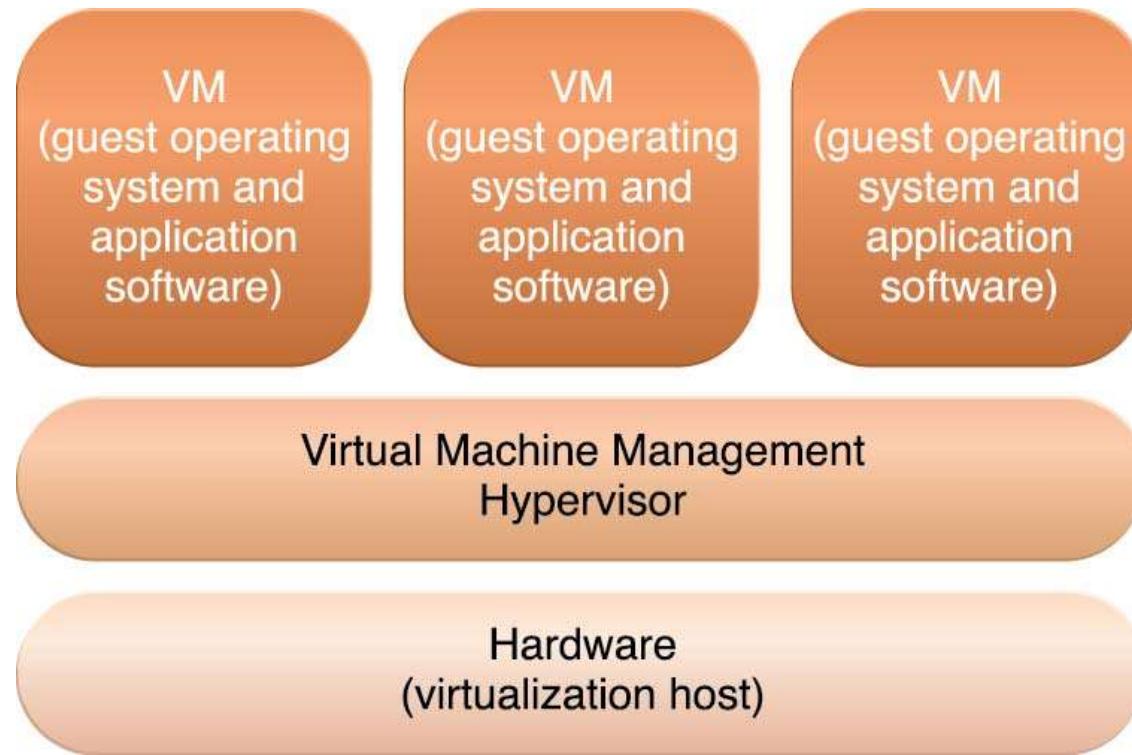
- Virtualization is a process of converting a physical IT resource into a virtual IT resource
 - Server
 - ❖ Virtual server ↔ virtual machine
 - Storage
 - Network
 - Power

Creating a new virtual server

- Allocation of physical IT resources
- Installation of an operating system, i.e., guest operating system



Hardware based virtualization



- Reduce the overhead
- May introduce compatibility issue

4. Web technology

- Cloud computing relies on internet.
- Web technology is generally used as both the implementation medium and the management interface for cloud services

Basic web technology

- Uniform resource locator (URL)
 - Commonly informally referred to as a **web address**
 - a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it
 - Example: <http://www.example.com/index.html>
- Hypertext transfer protocol (HTTP)
 - Primary communication protocol used to exchange content
- Markup languages (HTML, XML)
 - Express Web-centric data and metadata

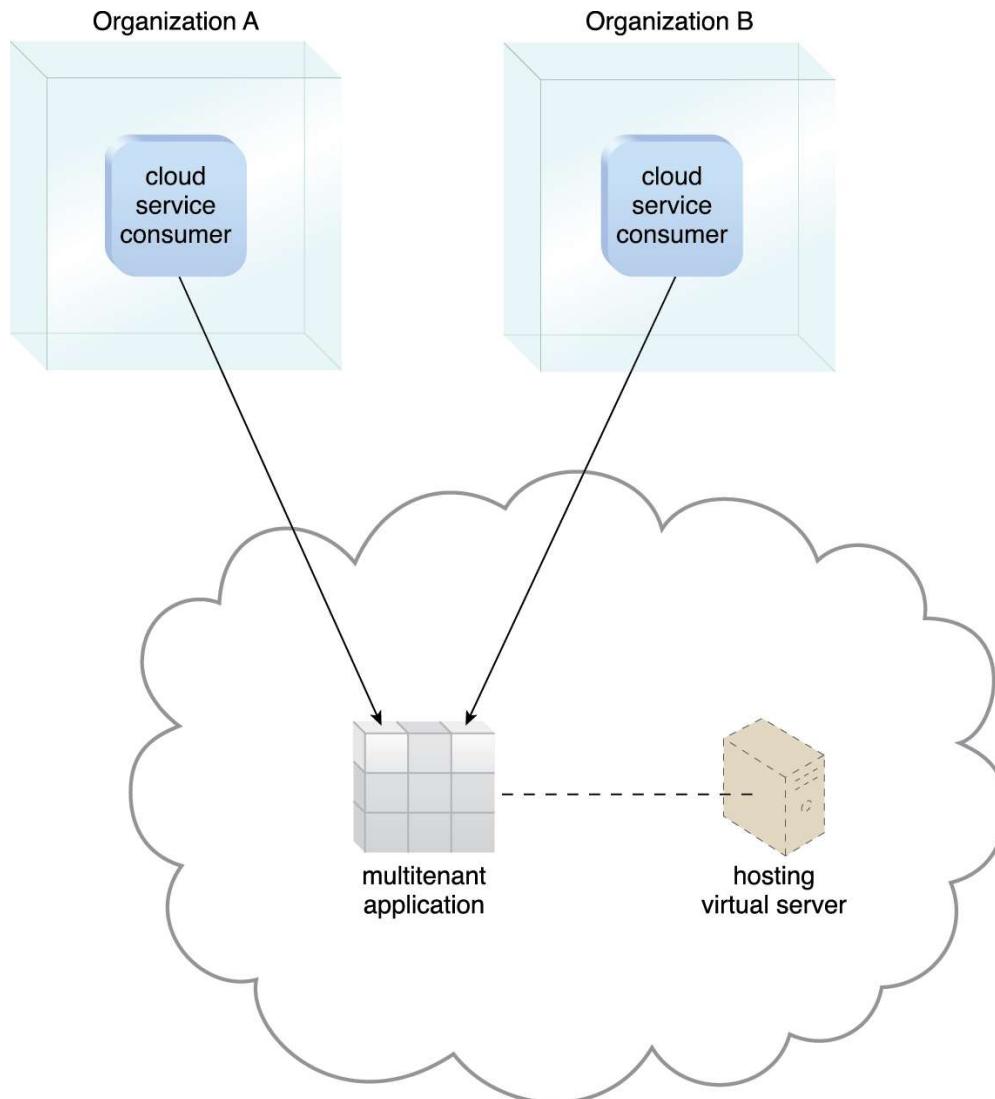
Web applications

- Applications running in a web browser
 - Rely on web browsers for the presentation of user-interfaces

5. Multitenant technology

- Enable multiple users (tenants) to access the same application simultaneously
- Multitenant applications ensure that tenants do not have access to data and configuration information that is not their own

A simple example



SERVICE ORIENTED ARCHITECTURE

-SOA-

SOA DEFINITION

SOA DEFINED

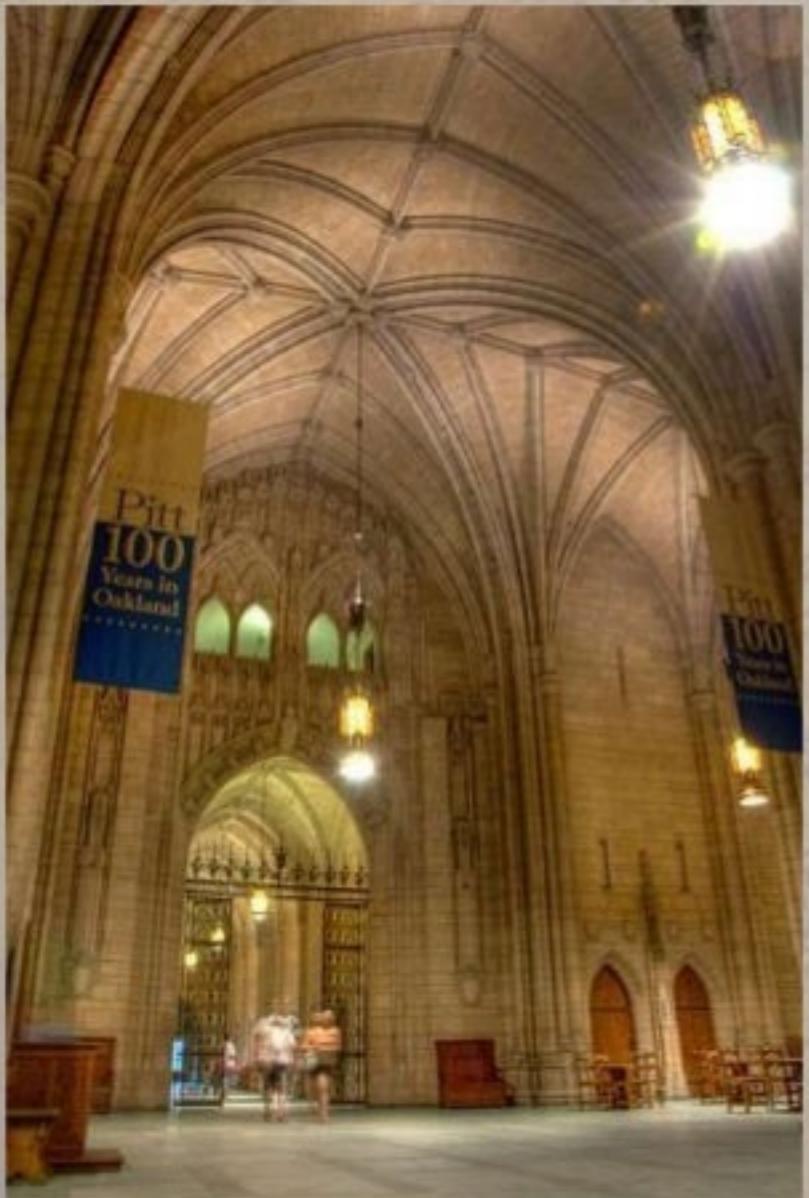
- An architecture based on **reusable**, well-defined **services** implemented by **IT components**.
- Components are **loosely coupled**.
- Provides **platform**, **technology** and **language independence**

WHAT IS AN ARCHITECTURE?

Architecture implies a consistent and coherent design approach.

Essential principles include:

- **Consistency:** The same challenges should be addressed in a uniform way.
- **Reliability:** The structures created must be fit to meet the demands for which they are designed.
- **Extensibility:** A design must provide a framework that expanded in ways both foreseen and unforeseen.
- **Scalability:** The implementation must be capable of scaled to accommodate increasing load by adding hardware to the solution.

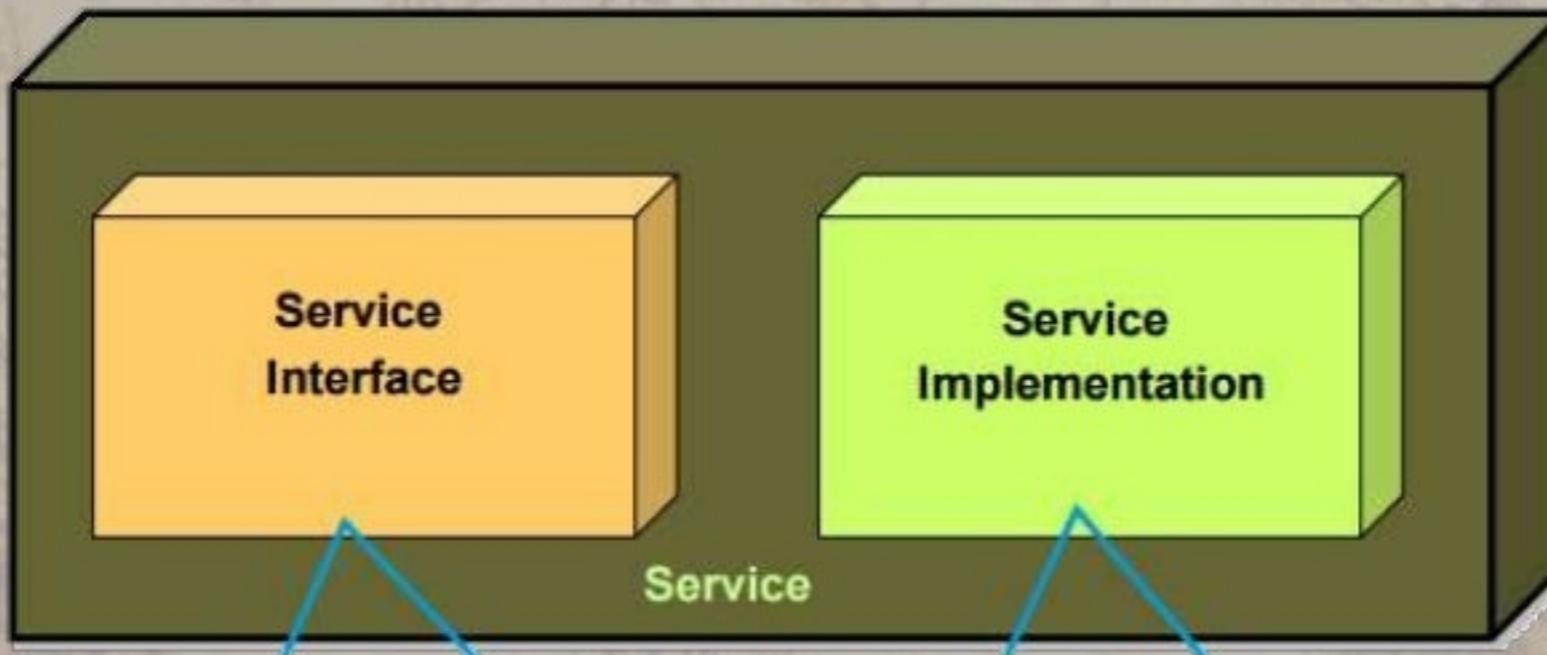




WHAT IS A SERVICE?

- A **service** is a package of closely related standardized functions, which are called repeatedly in a similar fashion, and should therefore be implemented by a dedicated facility, which can be specialized to perform them.
- A service can be partitioned and have multiple **service functions**.
- The smallest subunits within service functions are called **service primitives**.

ANATOMY OF A SERVICE



Access layer between the service consumer and service provider.
It contains,

- Service Identity
- Service Input & Output data information
- Service Purpose & Function Metadata

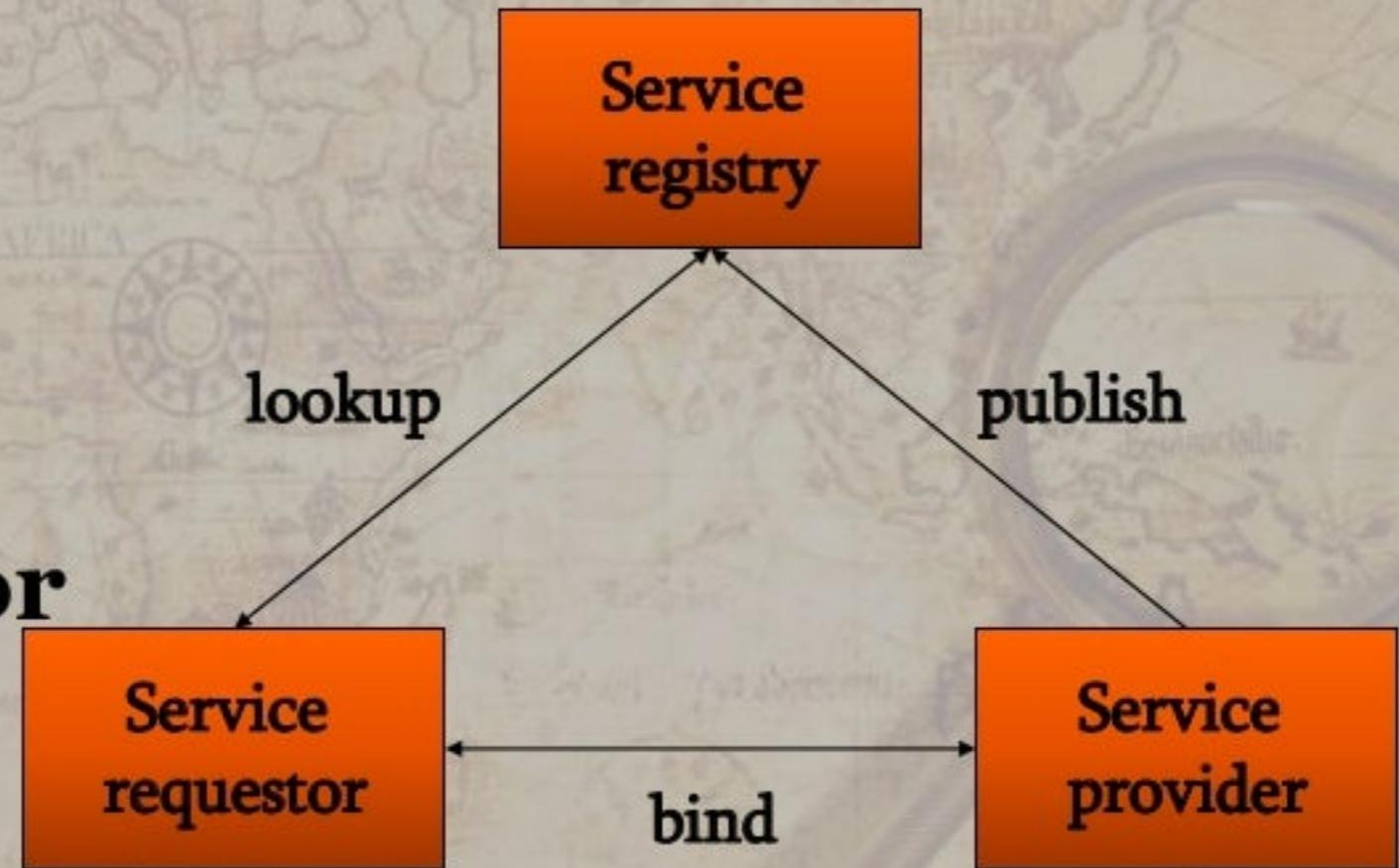
- Contains the core functional or business logic of the service.
- The implementation should be totally transparent to the service consumer, with no knowledge necessary about the implementation specifics.

SERVICE DISCOVERY

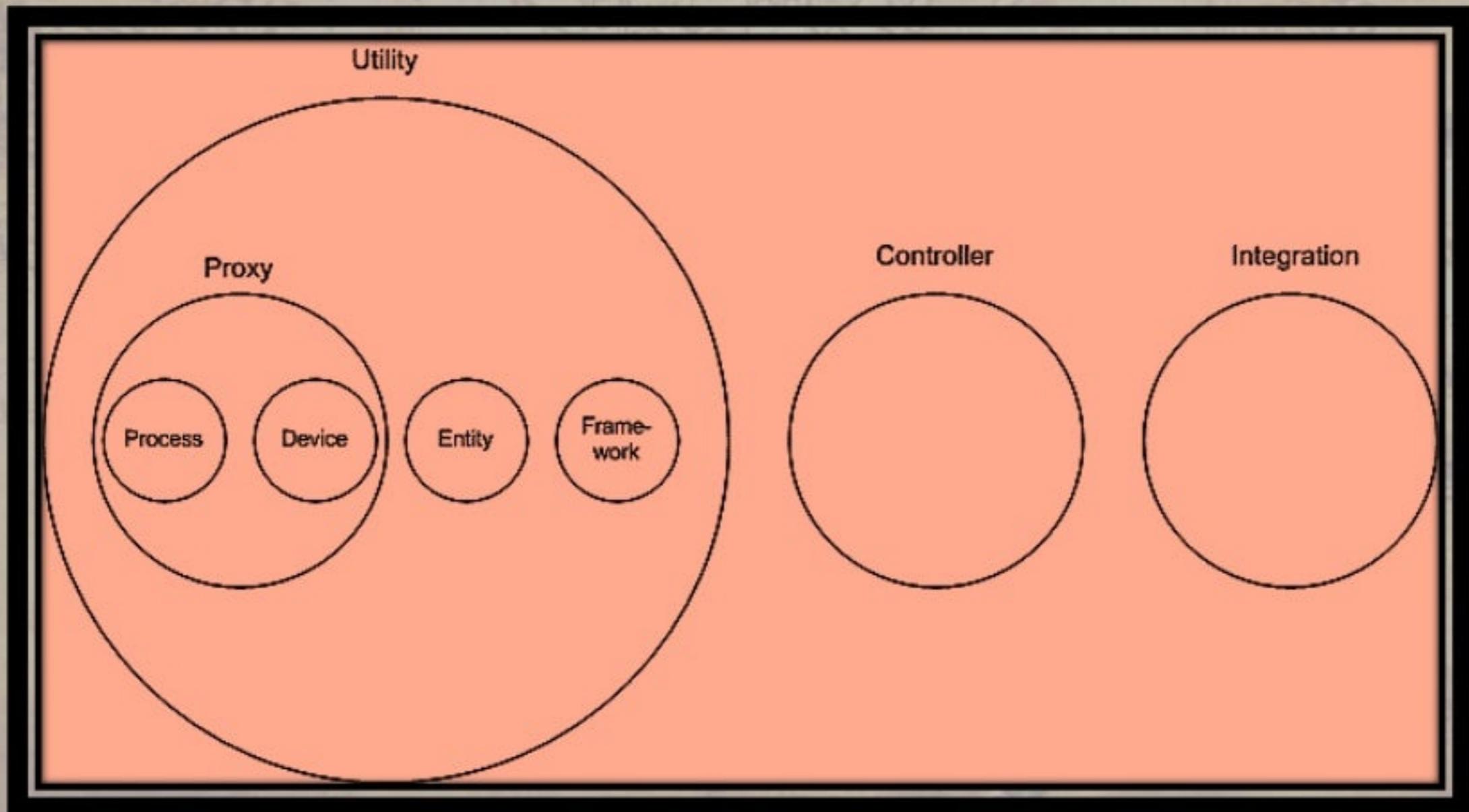
Consist of 3

components:

- **Service Registry**
- **Service Requestor**
- **Service Provider**



TYPES OF SERVICES THAT MAKE UP A SOA

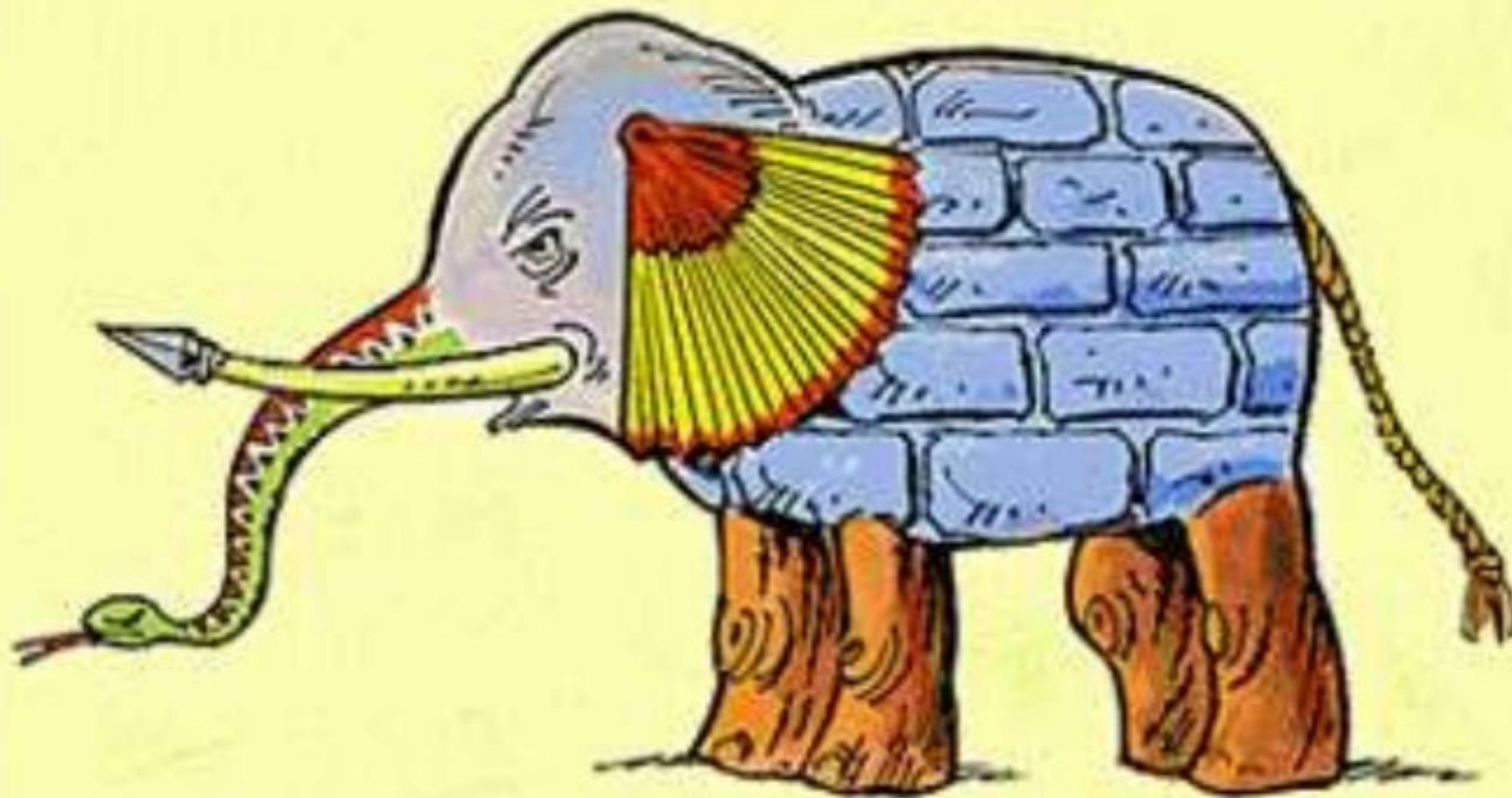


WHAT CAN SERVICES DO?

- Perform business logic
- Transform data
- Route messages
- Query databases
- Apply business policy
- Handle business exceptions
- Prepare information for use by a user interface
- Orchestrate conversations between multiple services

SERVICES OR COMPONENTS?

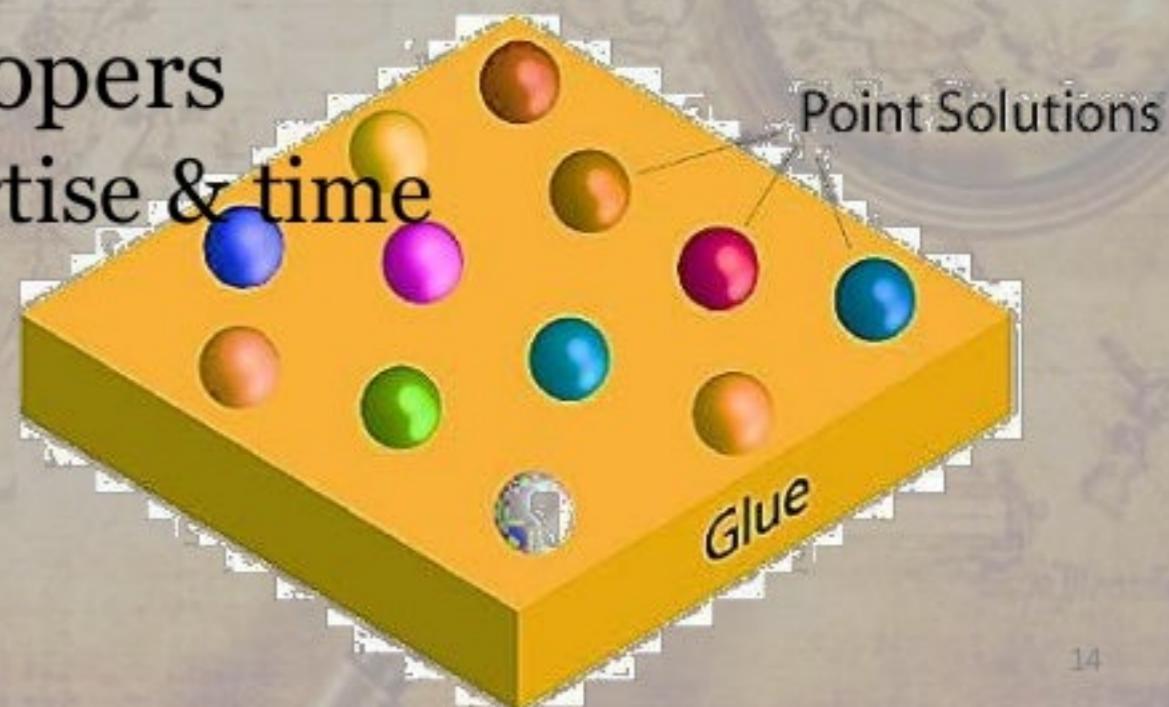
- A service can be defined as:
 - *A loosely-coupled, reusable software component that encapsulates discrete functionality which may be distributed and accessed. A web service is a service that is accessed using standard Internet and XML-based protocols.*
- A critical distinction between a service and a component as defined in component-based software engineering is that services are independent.
 - *Services do not have a ‘requires’ interface.*
 - *Services rely on message-based communication with messages in XML.*



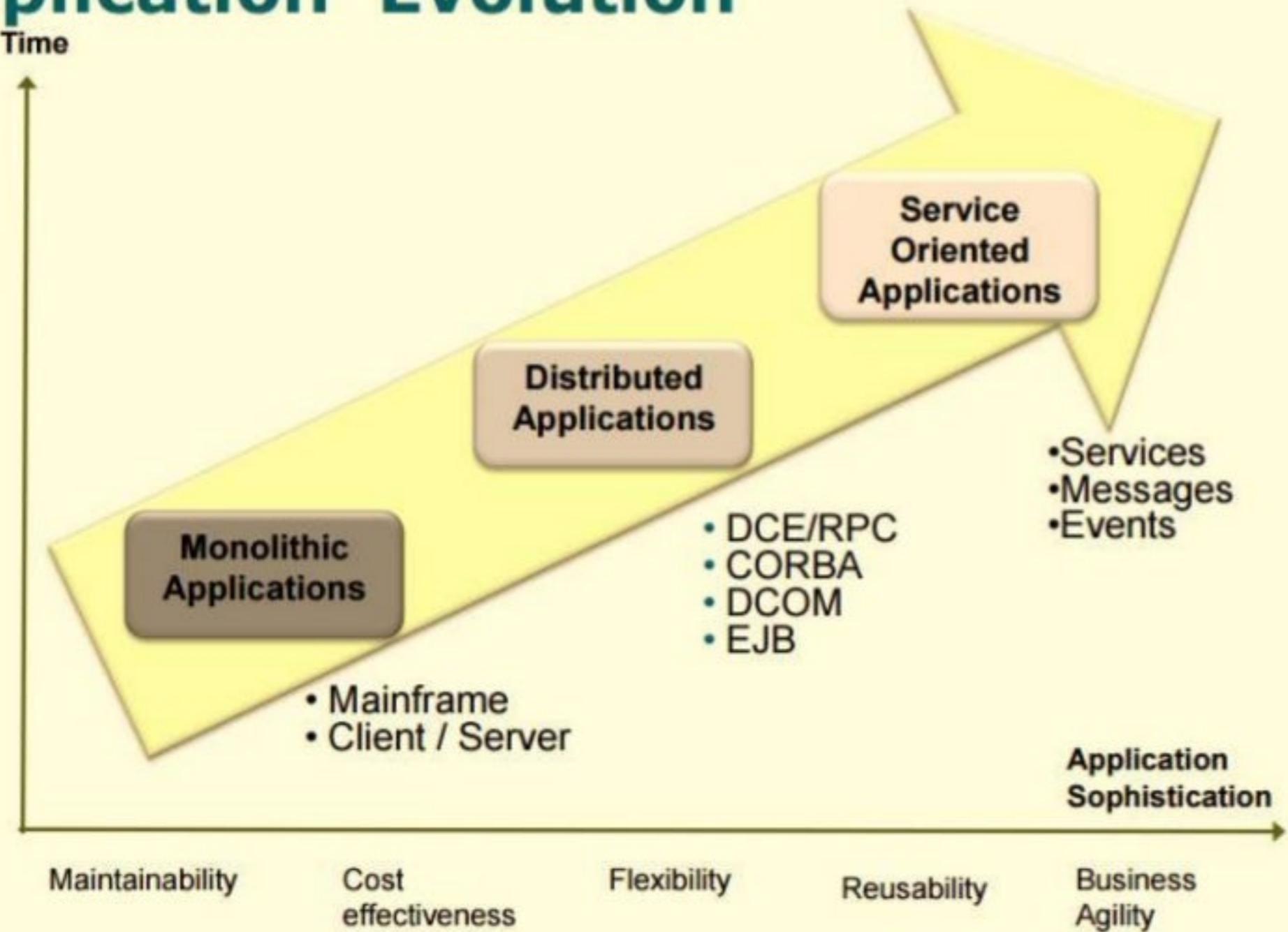
SOA

TRADITIONAL APPLICATION DEVELOPMENT

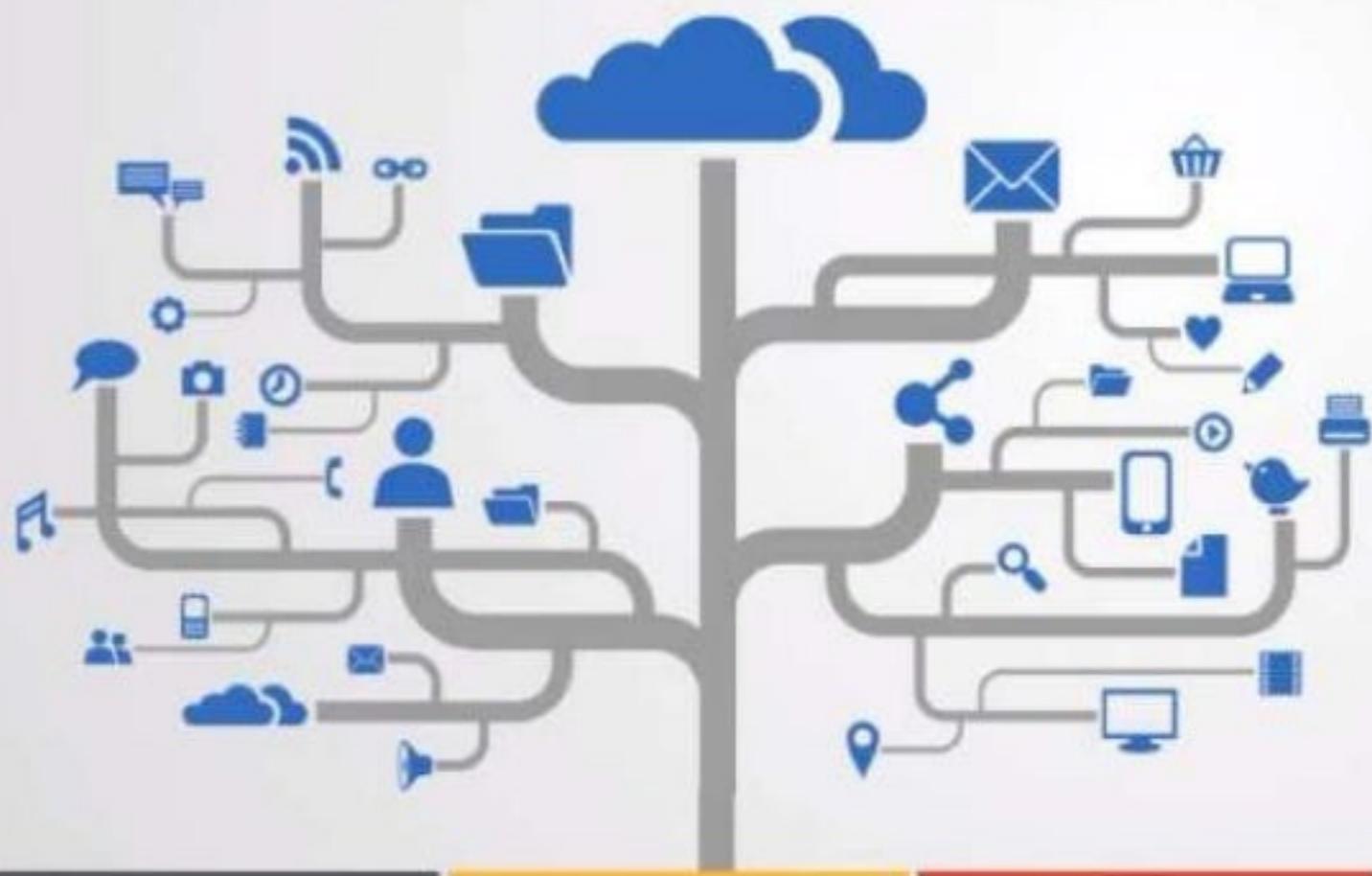
- Point technologies, products, and APIs
 - For example: EJB, Spring, Hibernate, JSF, Servlets, Struts, etc.
- Lots of glue written by developers
 - Requires a great deal of expertise & time
 - Inflexible



Application Evolution



Services orientated architecture



1970

1980

1990

2000

2010

WHAT IS SOA?

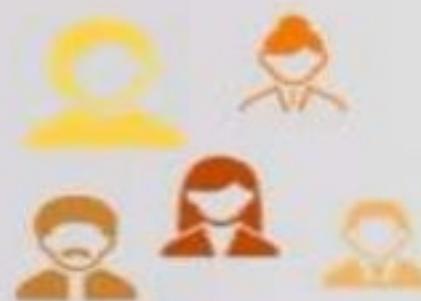
- **Gartner** - A software architecture that starts with an interface definition and builds the entire application topology as a topology of interfaces, interface implementations and interface calls.
- **IBM** - An application framework that takes everyday business applications and breaks them down into individual business functions and processes, processes, called services.
- **Microsoft** - A world-wide mesh of collaborating services that are published and available for invocation on a Service Bus.
- **BearingPoint** - A software design & implementation approach ("Architecture") of loosely coupled, coarse grained, reusable artifacts ("Services"), which can be integrated with each other, through a wide variety of platform independent service interfaces.

- An architectural framework that enables us to:
 - Create reusable services that are highly interoperable through the use of broadly-supported standards.
 - Support a new generation of agile composite applications assembled from business, application, and technical services.
 - Create external / business partner interfaces to streamline inter-enterprise integration, replacing EDI, VANs, etc.
 - Aligns business requirements with IT technology assets





Services over components



Interoperability & cross platform



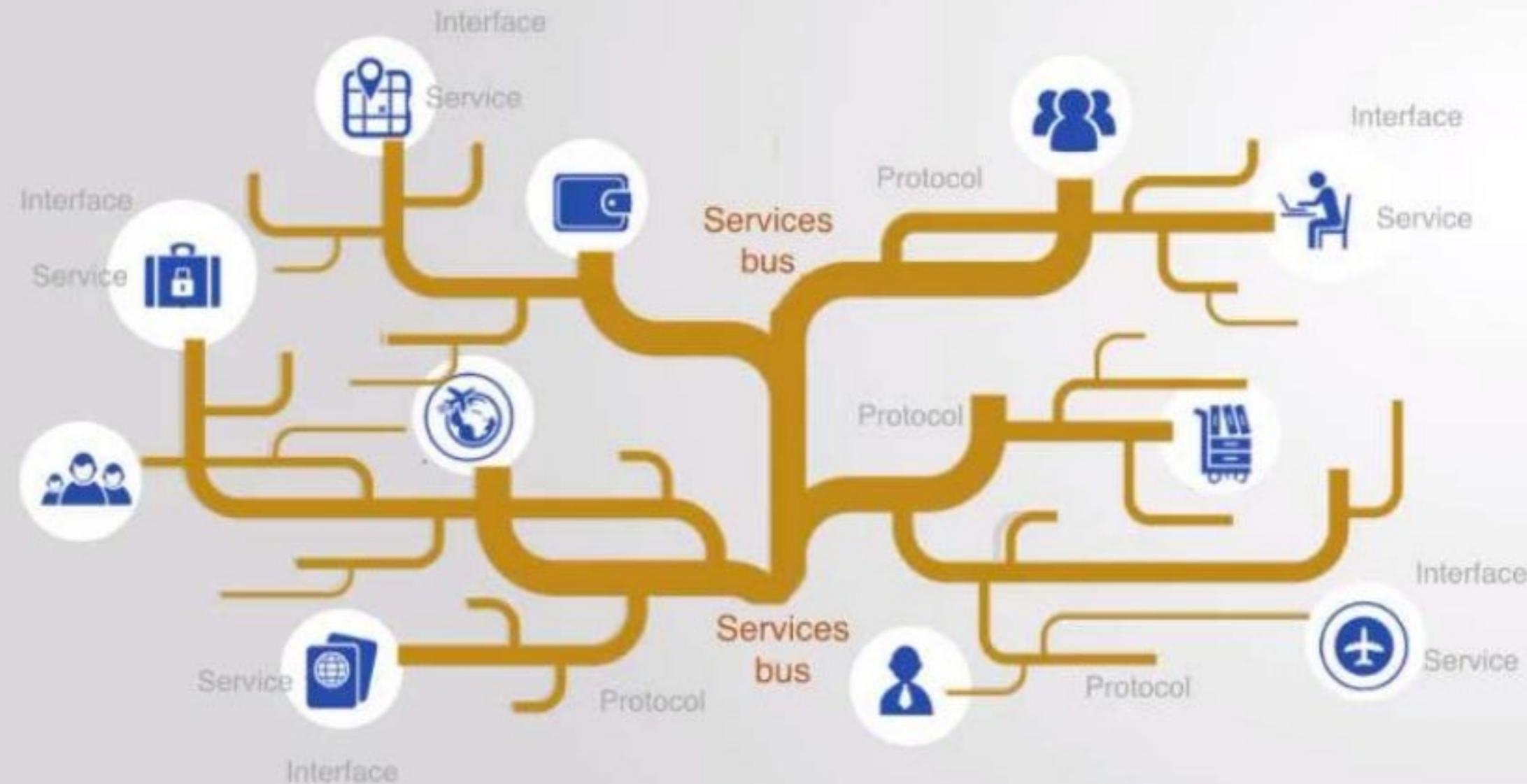
Loose coupling & distributed



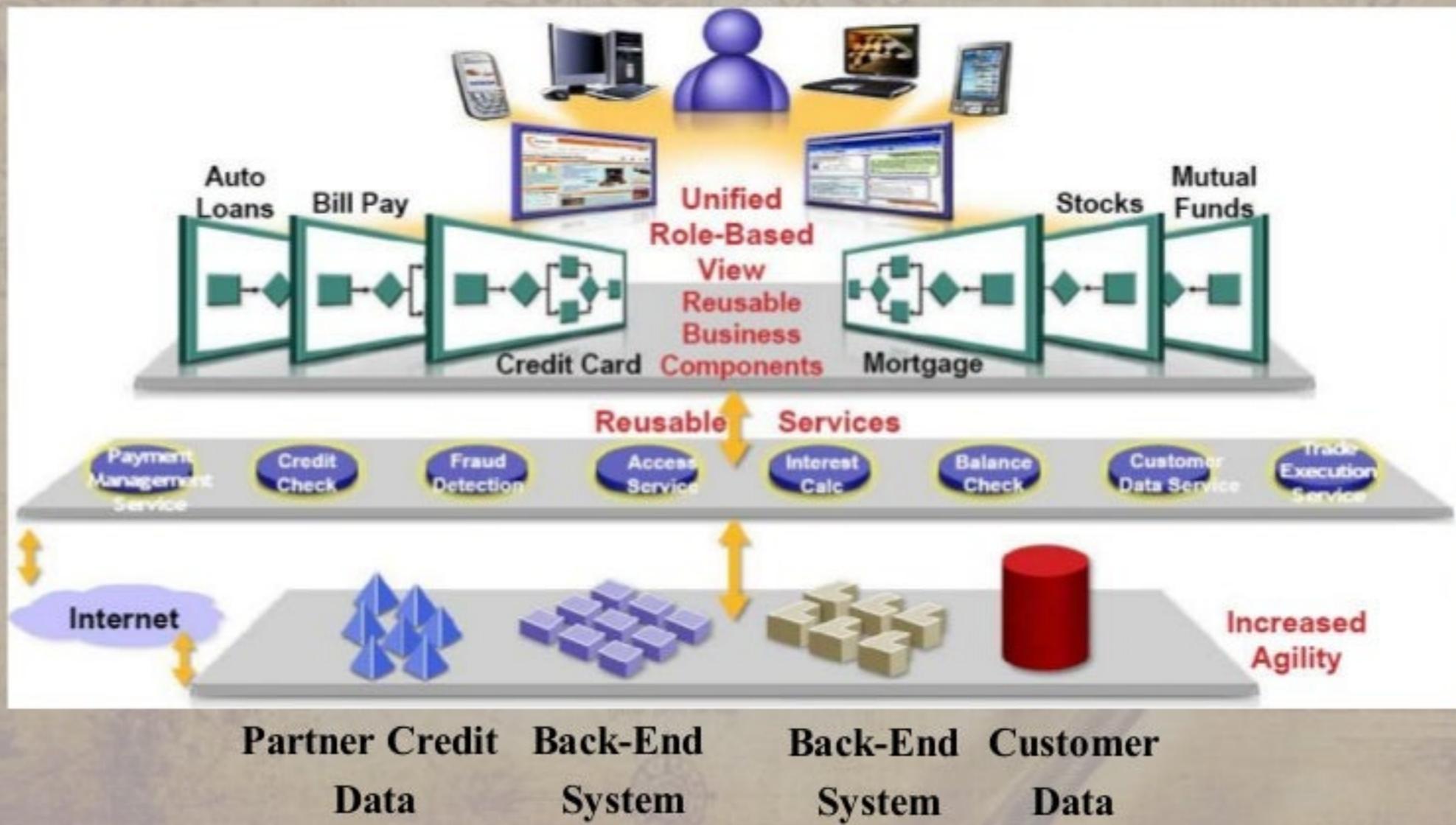
Abstraction again complexity

WHY IS SOA DIFFERENT?

- **Terminology:** Both IT people and business people know what a service is.
- **Interoperability:** The interfaces and the wire protocols are based on standards.
- **Extension** and **Evolution** not rip and replace.
- **Reuse** of both functionality and machine resources.



SOA-ENABLED SCENARIO



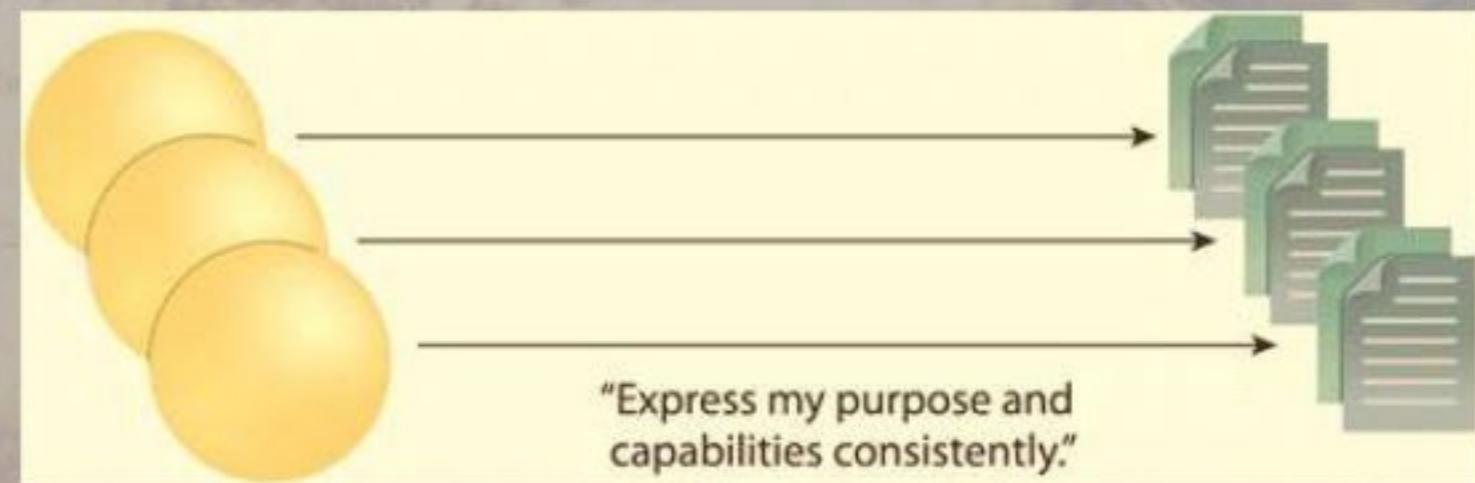
SOA ARCHITECTURE CHARACTERISTICS

SOA ARCHITECTURAL CHARACTERISTICS/PRINCIPLES

- Standardized Service Contracts
- Loose Coupling
- Abstraction
- Reusability
- Autonomy
- Statelessness
- Discoverability
- Composability

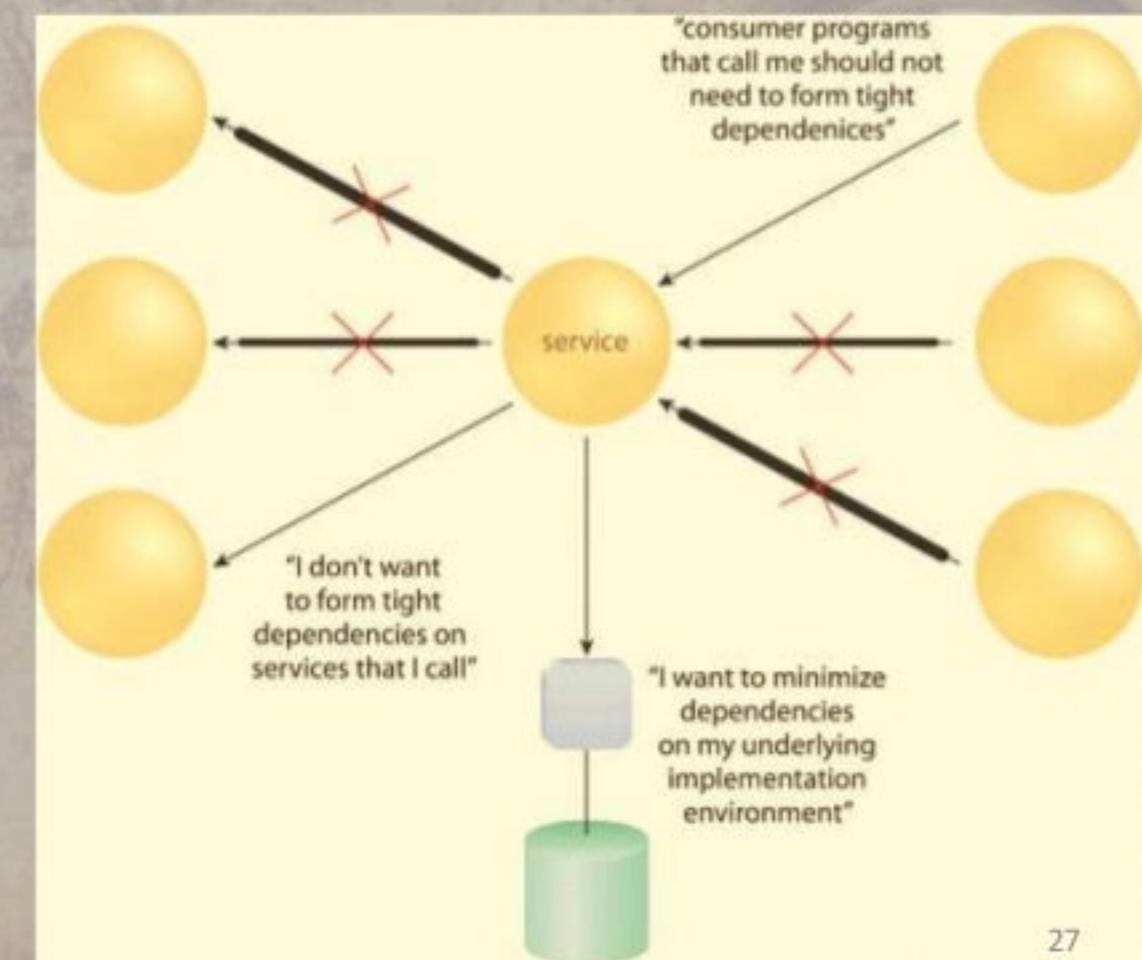
Standardized Service Contracts

- Services adhere to a service-description.
- Services use service contract to
 - Express their purpose
 - Express their capabilities
- Use formal, standardized service contracts



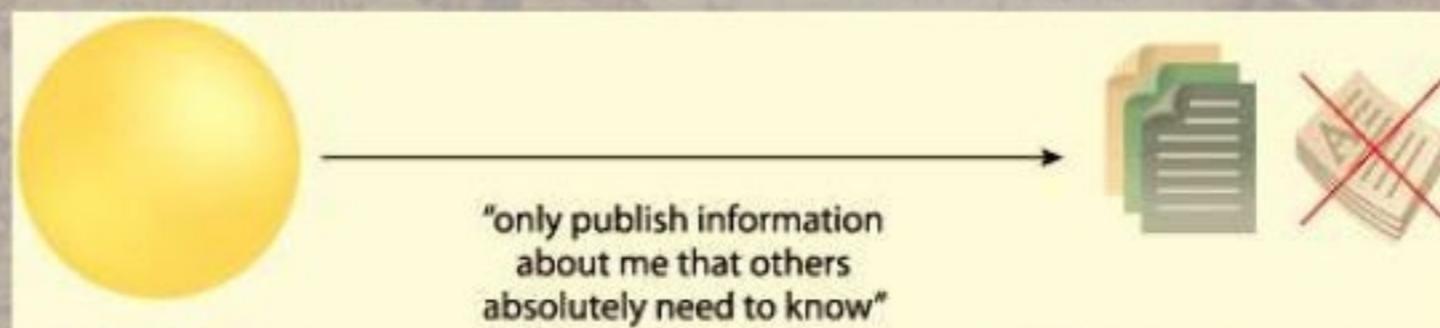
Loose Coupling

- Services minimize dependencies on each other.
- Create specific types of relationships within and outside of service boundaries with a constant emphasis on reducing (“loosening”) dependencies between
 - Service contract
 - Service implementation
 - Service consumers



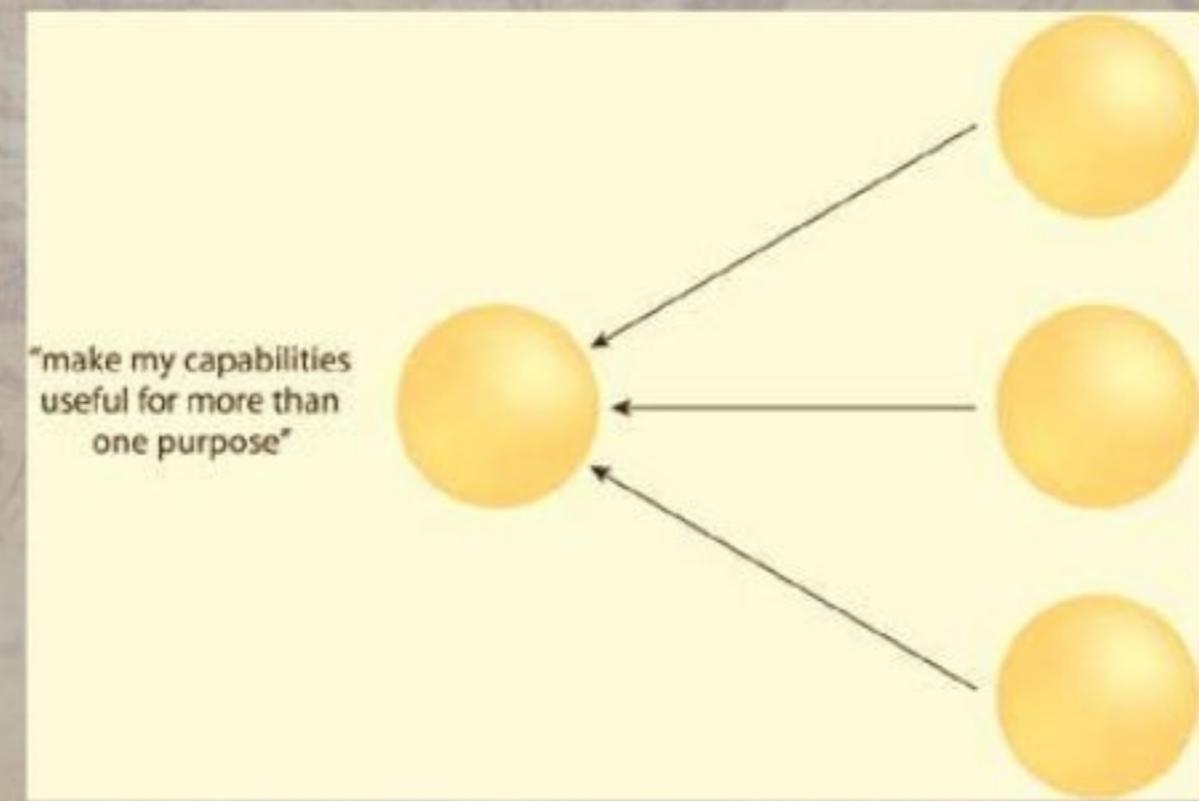
Abstraction

- Services hide the logic they encapsulate from the outside world.
- Avoid the proliferation of unnecessary service information, meta-data.
- Hide as much of the underlying details of a service as possible.
 - Enables and preserves the loosely coupled relationships
 - Plays a significant role in the positioning and design of service compositions



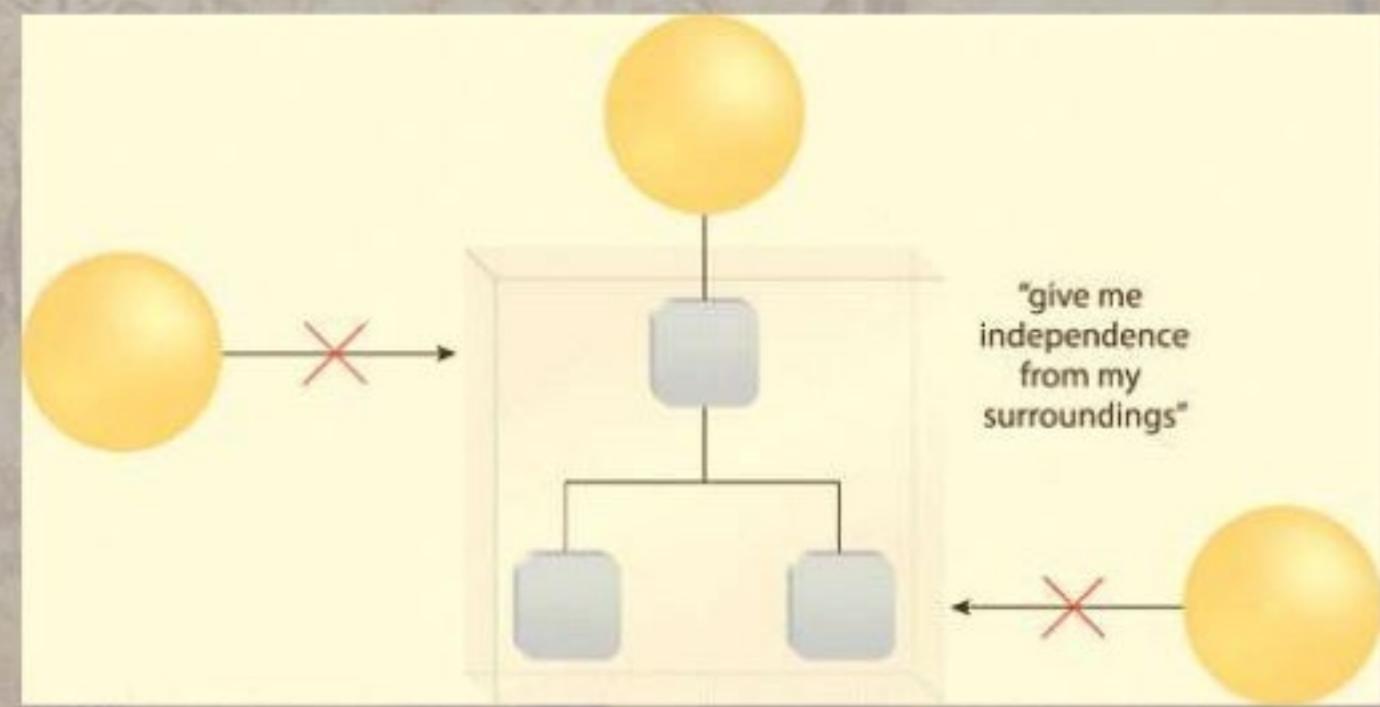
Service Reusability

- Logic is divided into services with the intent of maximizing reuse



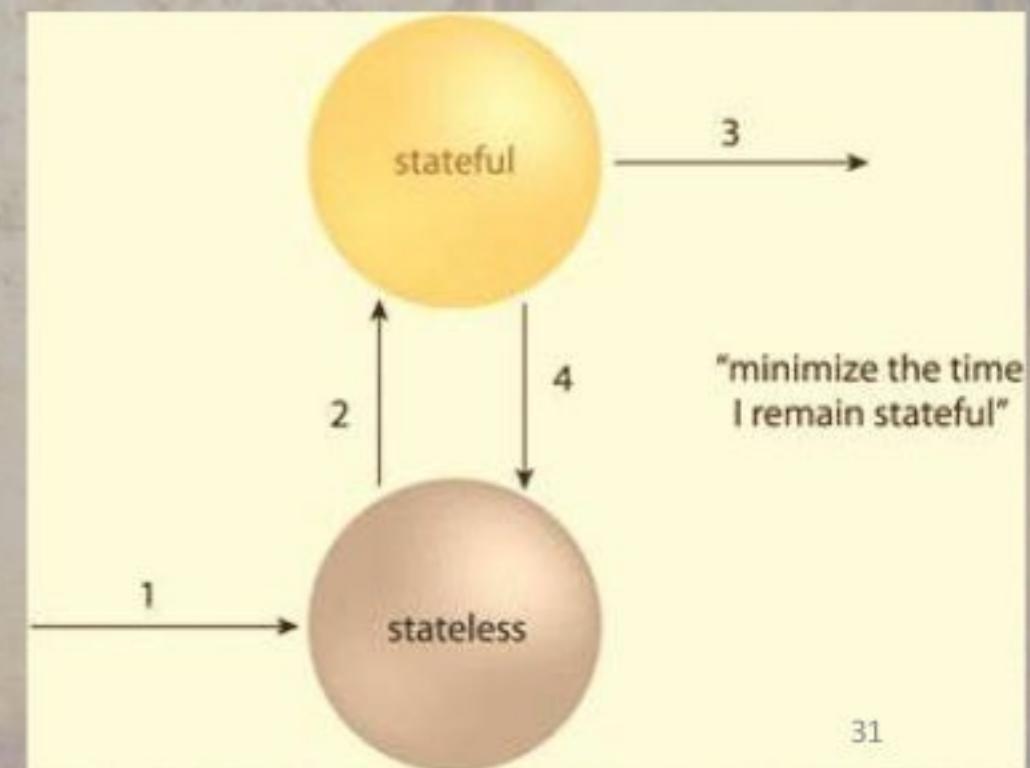
Autonomy

- Services should have control over the logic they encapsulate
- Represents the ability of a service to carry out its logic independently independently of outside influences
- To achieve this, services must be more isolated
- Primary benefits
 - Increased reliability
 - Behavioral predictability



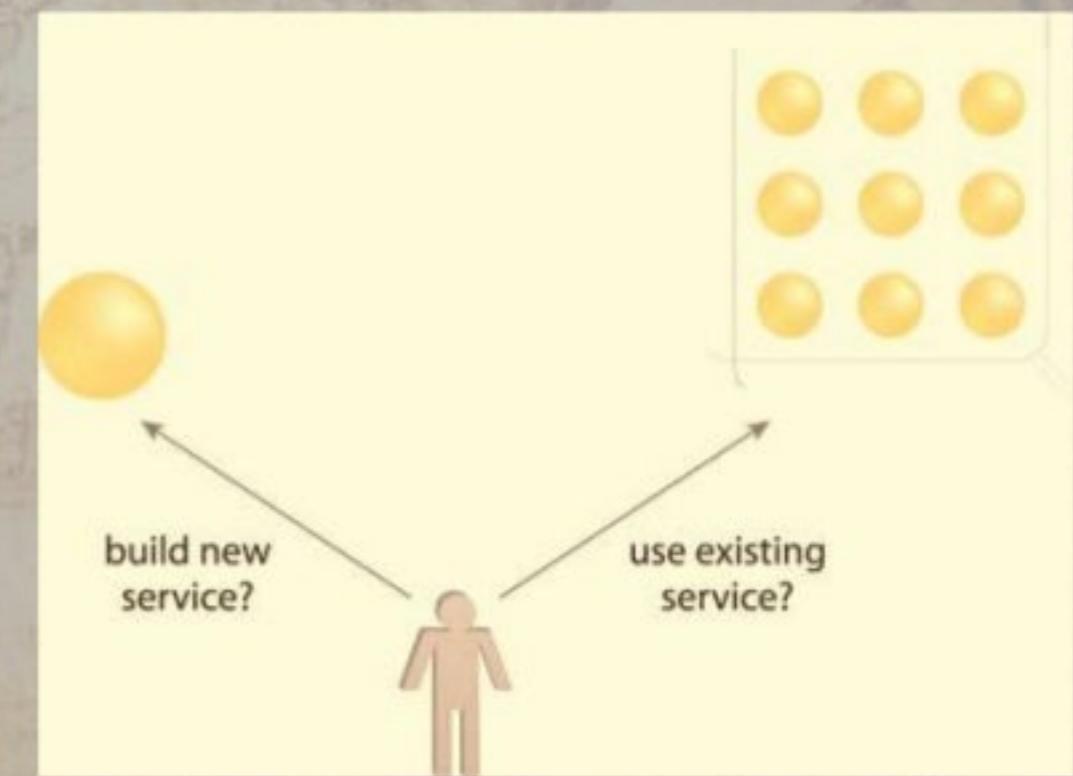
Statelessness

- Ideally, services should be stateless.
- Incorporate state management deferral extensions within a service design Goals
 - Increase service scalability
 - Support design of agnostic logic and improve service reuse



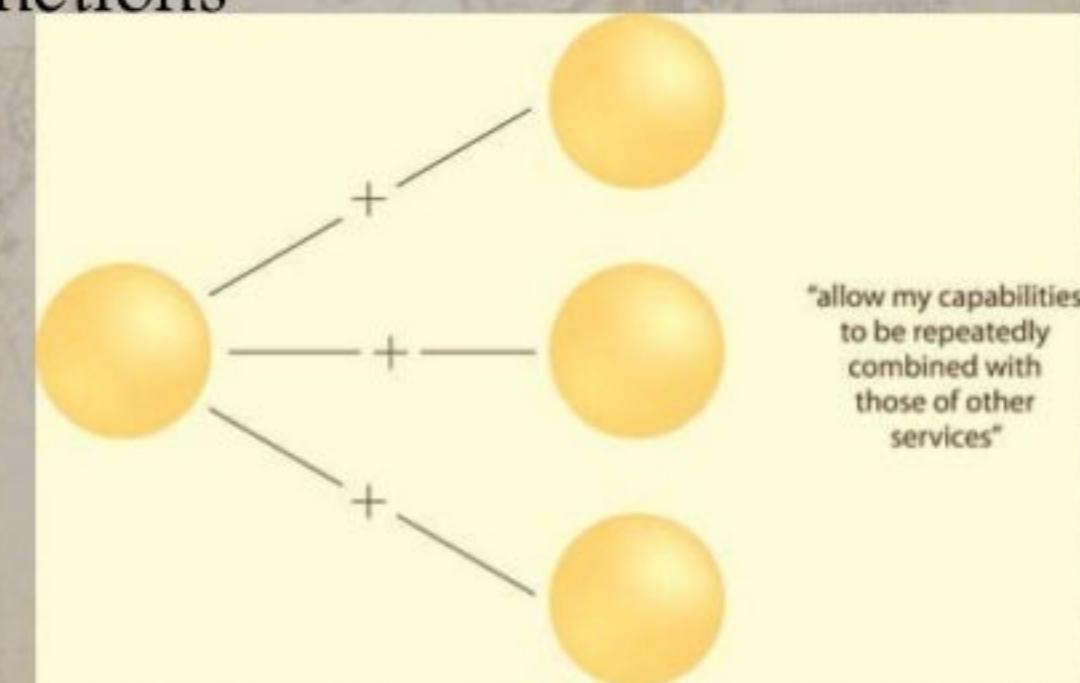
Discoverability

- Services can be discovered (usually in a service registry).
- Service contracts contain appropriate meta data for discovery which which also communicates purpose and capabilities to humans
- Store meta data in a service registry or profile documents



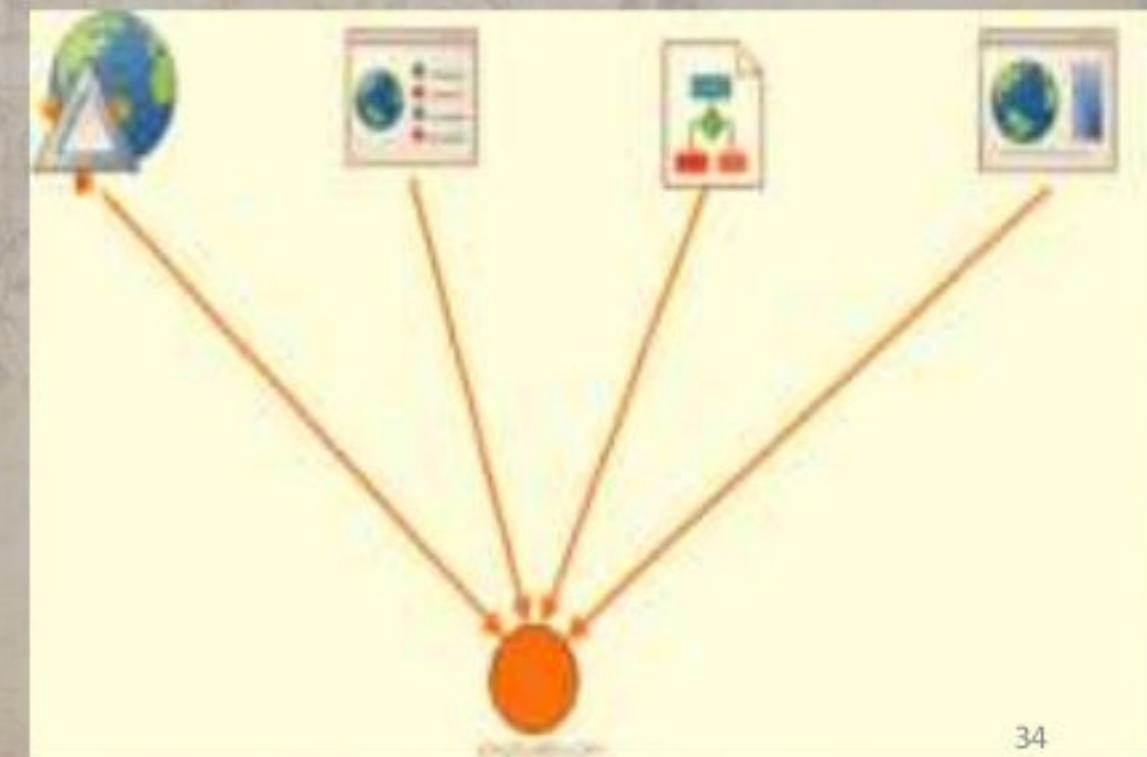
Composability

- Services break big problems into little problems.
 - Related to Reusability principle
- Service execution should efficient in that individual processing should be highly tuned
- Flexible service contracts to allow different types of data exchange requirements for similar functions

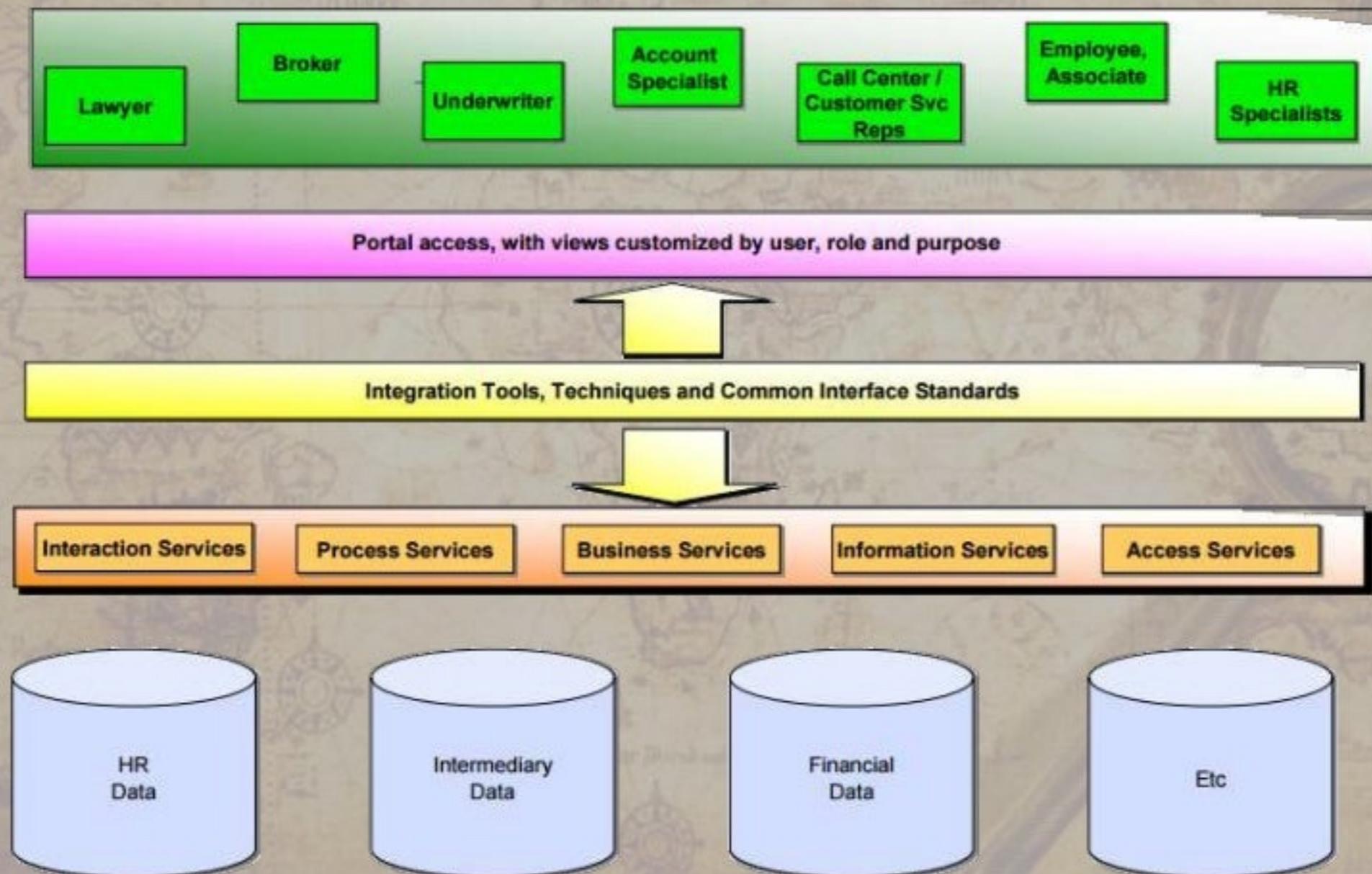


Interoperability

- Services should use standards that allow diverse subscribers to use the service.
- This is considered so obvious these days that it is often dropped as a principle.



SOA CONCEPTUAL MODEL



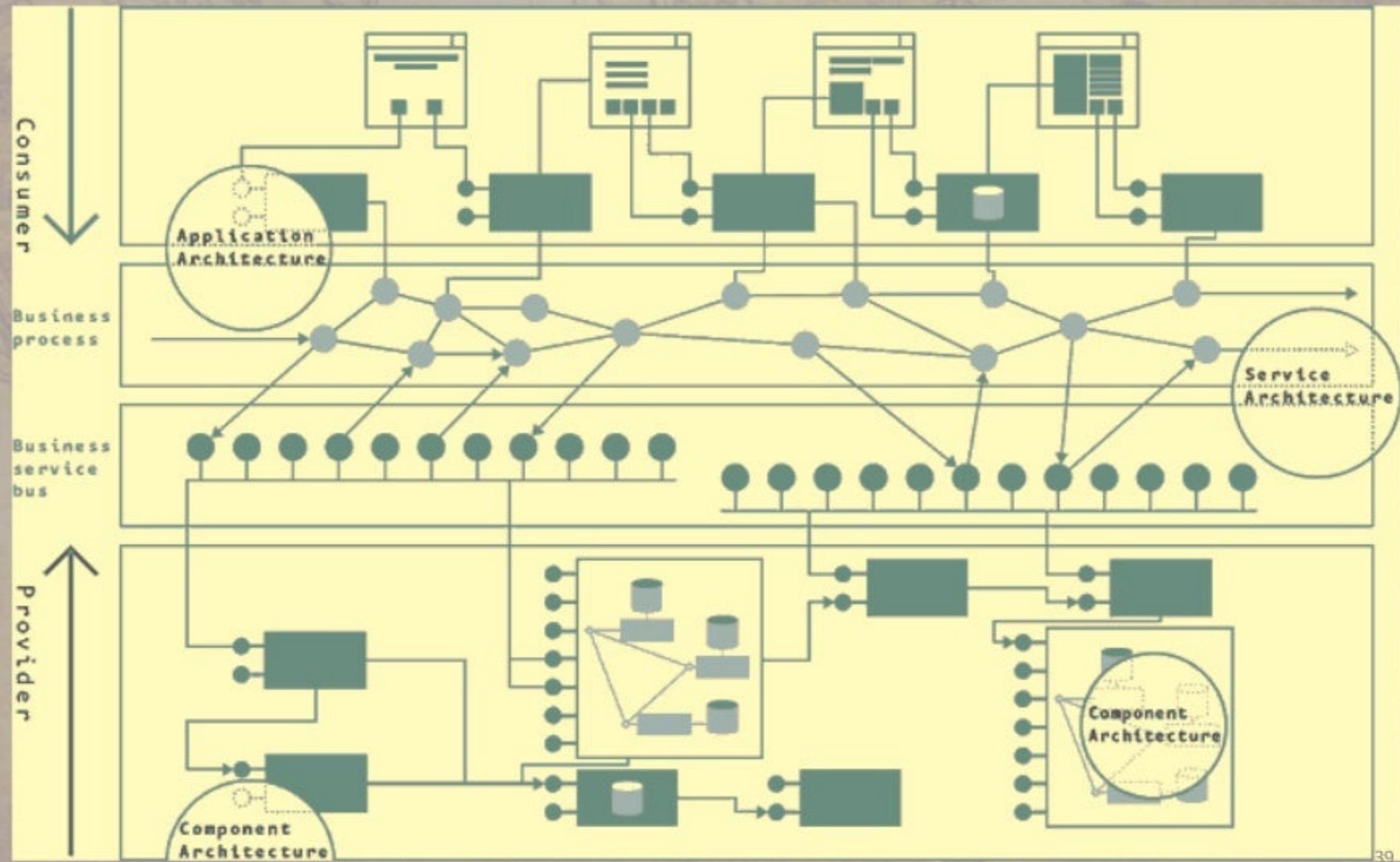
SOA ARCHITECTURE

SOA ARCHITECTURE

For SOA there are three important architectural perspectives:

- The Application Architecture
- The Service Architecture
- The Component Architecture

- These architectures can be viewed from either the consumer or provider perspective.
- Key to the architecture is that the consumer of a service should not be interested in the implementation detail of the service—just the service provided.
- The consumer is focused on their application architecture, the services used, but not the detail of the component architecture.
- Similarly, the provider is focused on the component architecture, the service architecture, but not on the



SERVICE ARCHITECTURE

- At the core of the SOA is the need to be able to manage services as first order deliverables.
- It is the service that we have constantly emphasized that is the key to communication between the provider and consumer.
- So we need a Service Architecture that ensures that services don't get reduced to the status of interfaces, rather they have an identity of their own, and can be managed individually and in sets.
- CBDI developed the concept of the Business Service Bus (BSB)

BUSINESS SERVICE BUS

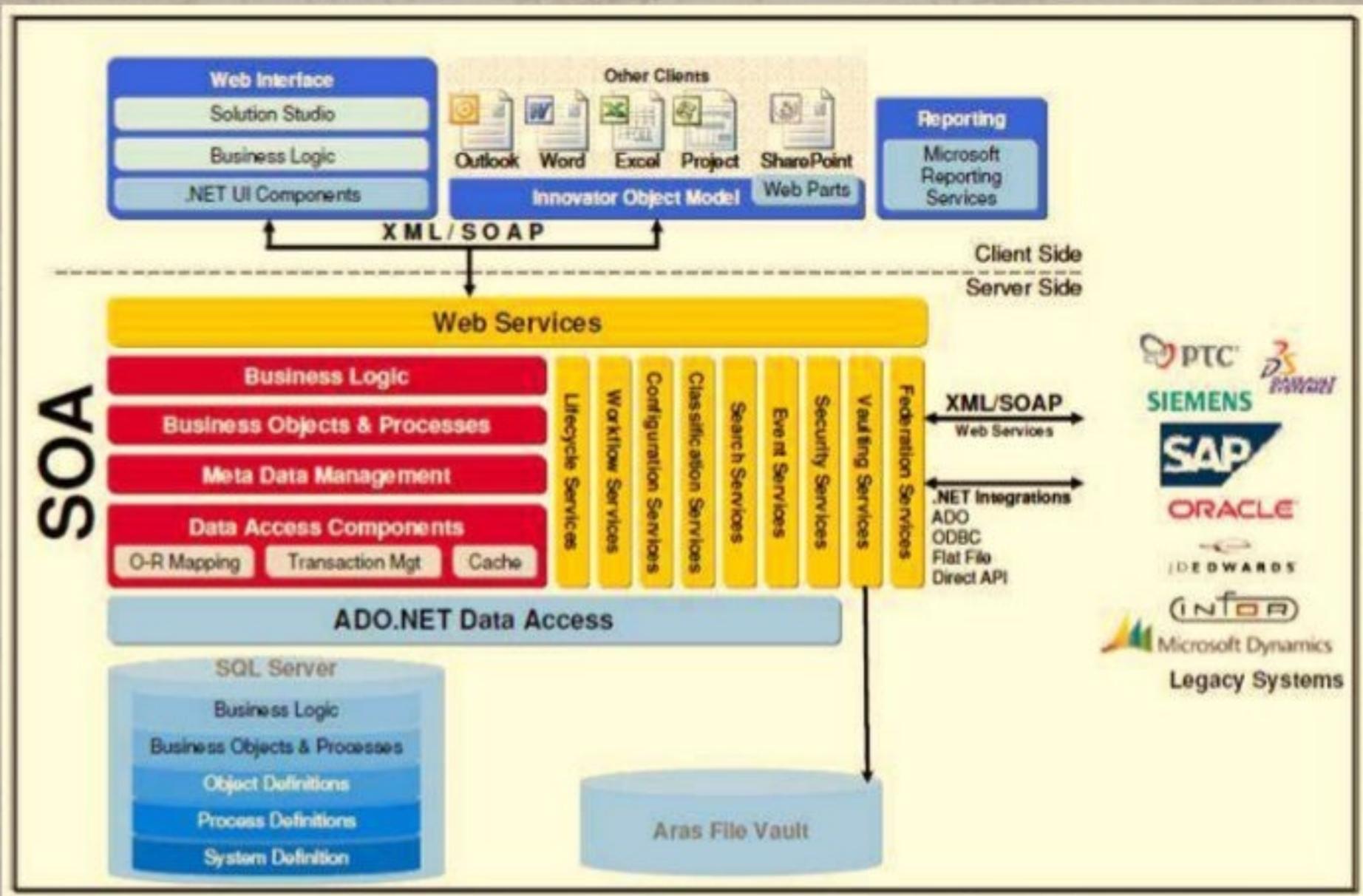
- The BSB is a logical view of the available and used services for a particular business domain, such as Human Resources or Logistics.
- The purpose of the BSB is so that common specifications, policies, etc can be made at the bus level, rather than for each individual service
- It also facilitates the implementation of a number of common, lower-level business infrastructure services that can be aggregated into other higher level business services on the same bus.

- A key question for the Service Architecture is
'What is the scope of the service that is published to the Business Service Bus?'
- A simplistic answer is
'At a business level of abstraction'

- In principle, the level of abstraction will be developed such that services are at a level that is relevant and appropriate to the consumer.
- The level might be one or all of the following:
 - Business Services
 - Service Consumer Oriented
 - Agreed by both Provider and Consumer
 - Combine low-level implementation-based services into something meaningful to business
 - Coarser Grained
 - Suitable for External Use
 - Conforms to pre-existing connection design

THE SOA PLATFORM

THE SOA PLATFORM



- The key to separation is to define a **virtual platform** that is equally relevant to a number of real platforms.
- The objective of the virtual platform is to enable the separation of services from the implementation to be as complete as possible and allow components built on various implementation platforms to offer services which have no implementation dependency.
- The virtual SOA platform comprises a blueprint which covers the development and implementation platforms.
- The blueprint provides guidance on the development and implementation of applications to ensure that the published services conform to the same set of structural principles that are relevant to the management and consumer view of the services.
- When a number of different applications can all share the same structure, and where the relationships between the parts of the structure are the same, then have what might be called a common architectural style.

- The style may be implemented in various ways;
 - it might be a common technical environment
 - a set of policies, frameworks or practices
- Example platform components of a virtual platform include:
 - Host environment
 - Consumer environment
 - Middleware
 - Integration and assembly environment
 - Development environment
 - Asset management
 - Publishing & Discovery
 - Service level management
 - Security infrastructure
 - Monitoring & measurement
 - Diagnostics & failure
 - Consumer/Subscriber management
 - Web service protocols
 - Identity management
 - Certification
 - Deployment & Versioning

THE ENTERPRISE SOA

SOA Enterprise

Enterprise SOA Vision

SOA Mission

SOA Scope

Enterprise
Business
and
Technology
Alignment

SOA
Processes

SOA
Principles

SOA
Patterns

SOA
Standards

Enterprise
Integration

SOA Technology Capabilities

SOA Services

THE SOA PLATFORM

SOA Platform



Partner
Services



Application
Services



Development
Services



Integration
Services



Business
Services



Interaction
Services



Legacy
Services



SOA Infrastructure:



Security



Management



Mediation



Governance



Enterprise Infrastructure:



Identity Management



Systems Management



Resource Management

- The optimum implementation architecture for SOA is a component-based architecture.
- Many will be familiar with the concepts of process and entity component, and will understand the inherent stability and flexibility of this component architecture, which provide a one to one mapping between business entities and component implementations.
- Enterprise SOA (ESOA) brings the two main threads
 - Web services
 - CBD (or CBSE)—together
- The result is an enterprise SOA that applies to both Web services made available externally and also to core business component services built or specified for internal use.

Before SOA

Siloed · Closed · Monolithic · Brittle

Application Dependent Business Functions



After SOA

Shared services · Collaborative · Interoperable · Integrated

Composite Applications

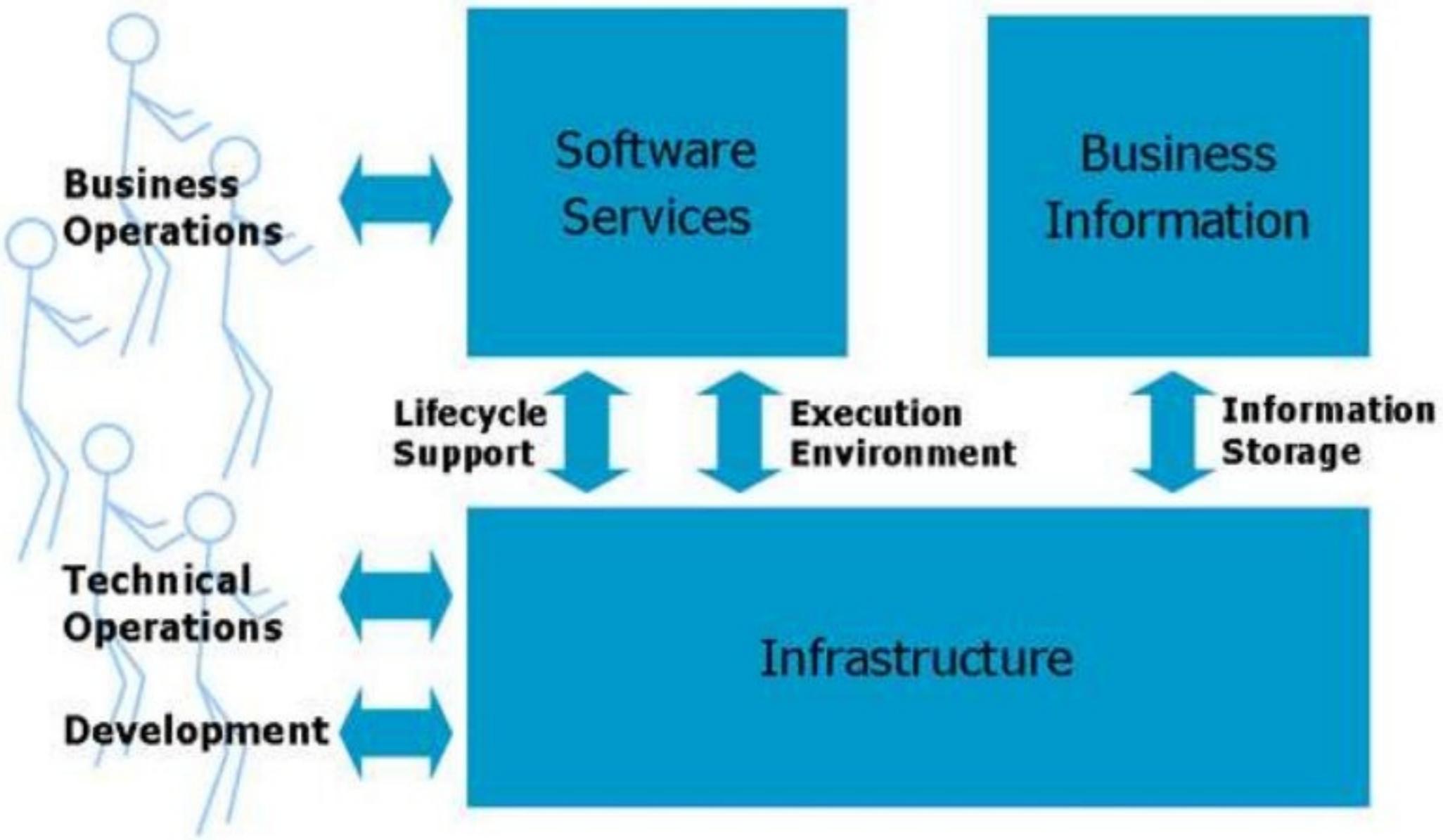


Reusable Business Services



Data Repository





SOA IMPLEMENTATION STEPS

SOA IMPLEMENTATION STEPS

Step 1: Establish an SOA Vision.

Step 2: Building an SOA Roadmap

- Services needed
- What their interfaces should look like
- Scope of each service
- Granularity of each service

Step 3: Establish an SOA Methodology

For instance, exposing a specific function in an application as a Web service may be possible. But, will it serve the needs of the overall SOA? Your IT organization needs to establish core principles surrounding the SOA then consider potential applications. It may be useful to build a set of best practices over a period of time. These practices will become the core of a proven methodology.

Step 4: Link SOA with Key Business Initiatives

- ❖ **Efficiency:** Business processes can be transformed from isolated silo and replicated processes into highly leveraged, shared services that cost less to maintain.
- ❖ **Responsiveness:** Rapid adaptation and delivery of key business services can meet market demands for increased service levels to customers, employees and partners.
- ❖ **Adaptability:** Changes can be made throughout the business with minimal complexity and effort, saving time and money.

Step 5: Create Architectural Blueprints

Blueprints may include:

- Common security model
- Service orchestration model
- Metadata management
- Process integration model
- Web services compliance model

Step 6: Assess Risks

Key risk of SOA,

- Security
- Interoperability
- Approaches to failure

Step 7: Create an SOA Risk Mitigation Strategy

Step 8: Process-Driven Integration

Patterns for Aligning Business and IT supplies detailed guidance on how to design and build software architectures that follow the principles of business-IT alignment. It illustrates the design process using proven patterns that address complex business/technical scenarios, where integrated concepts of service-oriented architecture (SOA), Business Process Management (BPM), and Event-Driven Architecture (EDA) are required.

WEB SERVICES AND SOA

- *Web Services are the set of protocols by which Services can be published, discovered and used in a technology neutral, standard form.*
- In fact Web services are not a mandatory component of a SOA, although increasingly they will become so.
- SOA is potentially much wider in its scope than simply defining service implementation, addressing the quality of the service from the perspective of the provider and the

- Web services are purely the implementation. SOA is the approach, not just the service equivalent of a UML component packaging diagram.
- *SOA is not just an architecture of services seen from a perspective, but the policies, practices, and frameworks by which we ensure the right services are provided and consumed.*
- It is important that if a service is to be used by multiple consumers, the specification needs to be generalized, the service needs to be abstracted from the implementation.

ENABLED BY WEB	Technology neutral	Endpoint platform independence.
SERVICES	Standardized	Standards-based protocols.
	Consumable	Enabling automated discovery and usage.
	Reusable	Use of Service, not reuse by copying of code/implementation.
	Abstracted	Service is abstracted from the implementation.
ENABLED BY SOA	Published	Precise, published specification functionality of service interface, not implementation.
	Formal	Formal contract between endpoints places obligations on provider and consumer.
	Relevant	Functionality presented at a granularity recognized by the user as a meaningful service.

VALUE PROPOSITION - BUSINESS BENEFITS

VALUE PROPOSITION - BUSINESS BENEFITS

- Agile business and technology environment
- Reduced business, IT, and maintenance costs
- Manageable project size
- Technology autonomy and platform-agnostic systems
- Facilitates automated business process
- Evolutionary approach

VALUE PROPOSITION - BUSINESS BENEFITS

TODAY

Business Process Friction

New supplier relationship

Re-architect business process

Implement process randomly

IT application and infrastructure change

TIME
→

Business objective

DESIRED

Business Process Agility

New supplier relationship

Leverage supplier on-ramp process

Integrate supplier info SOA

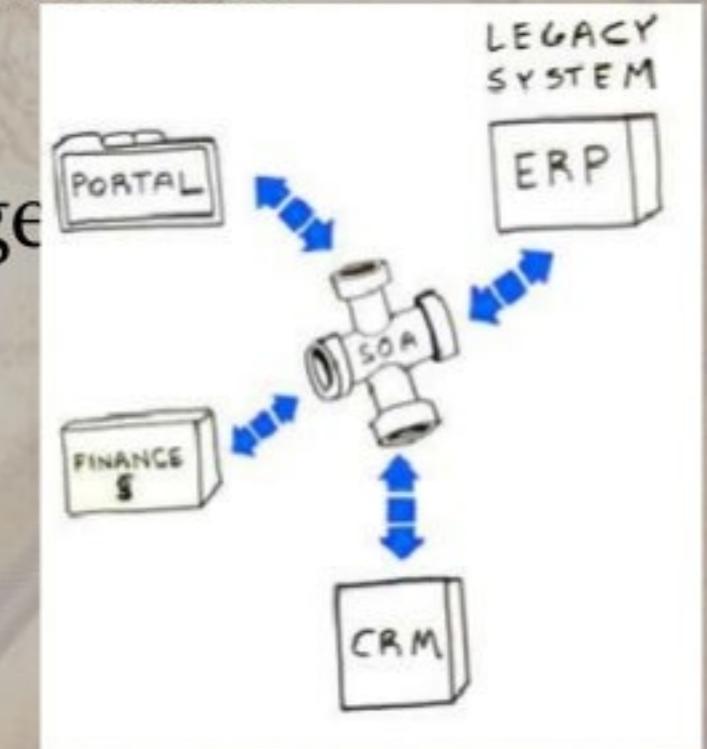
TIME
→

Business objective

TIME SAVED
→

VALUE PROPOSITION - TECHNOLOGY BENEFITS

- Efficient development process
- Reduced number of interfaces and development time
- Platform-agnostic applications
- Design time discovery, introspection and usage services
- Standards-based architectures
- Several enablers - Portals, BPM, BI, ERP



SOA –BENEFITS PERSPECTIVE

CEO Perspective

- Budget Strategy
Agile & Reactive IT environments.
- Short Term Planning
SOA enables step-by-step approaches
- Budget Reduction
- Technology & Vendor Agnostic

CIO Perspective

- Independence from technology
- Positive Role of IT Department
IT – Business bridging
- Cost Reduction
- Increase of Influence
- Manageable Project Size

IT Architect Perspective

- Disentanglement
- Loose Coupling
- Code Reuse

Project Manager Perspective

- Smaller & Shorter Projects
- Technology Independence
- Parallel Development
- Reduced Project Risk
- Easier Testing & Integration

Developer Perspective

- Reduction of Dependency
- Rapid Prototyping
- Better Define Requirements
- Simplified Testing

Business Department Perspective

- Independence from Technology
- Shorter Time to Market
- Reduction of Development Costs

ADVANTAGES OF SOA

- Service Reusability
- Easy Maintainability
- Greater Reliability
- Location Independence
- Improved Scalability & Availability
- Improved Software Quality
- Platform Independence
- Increased Productivity

DISADVANTAGES OF SOA

- Increased Overhead
- Complex Service Management
- High Investment Cost

SOA IS NOT RECOMMENDED FOR THE FOLLOWING TYPE OF APPLICATIONS.



1) Homogenous:

- Implementing SOA for applications that use the technologies of a single vendor will not be cost-effective.
e.g. - if an application is built in Java, then it would be better to use component methods of Java rather than using communications.

2) GUI-Based:

- SOA would not be suitable for applications with GUI functionality,
e.g. a map manipulation application.
- Such applications require heavy data exchange, which in turn would increase the complexity of the application if SOA is used.

Cont...

Real-time:

- SOA is not desirable to be used with strictly-enforced response times since the services communicate asynchronously.

Stand-alone:

- It would be futile to invest in SOA for stand-alone non-distributed applications, which do not require request and response-response-based calls.

SUMMARY

- The goal for a SOA is a **world wide mesh of collaborating services**, which are published and available for invocation on the Bus.
- Adopting SOA is essential to deliver the **business agility** and **IT flexibility** promised by Web Services.
- These benefits are delivered not by just viewing service architecture from a technology perspective and the adoption of Web Service protocols, but **require the creation of a Service Oriented Environment that is based on the key principals**.

- Service is the important concept. Web Services are the set of protocols by which Services can be published, discovered and used in a technology neutral, standard form.
- SOA is not just an architecture of services seen from a technology perspective, but the **policies, practices, and frameworks by which ensure the right services are provided and consumed.**
- With SOA it is critical to implement processes that ensure that there are at least two different and separate processes — for **provider** and **consumer**.
- Rather than leaving developers to discover individual services and put them into context, the Business Service Bus is instead their starting point that guides them to a coherent set that has been assembled for their domain.

THANK YOU !

Chapter 6. Fundamental Cloud Security



6.1 Basic Terms and Concepts

6.2 Threat Agents

6.3 Cloud Security Threats

6.4 Additional Considerations

6.5 Case Study Example

This chapter introduces terms and concepts that address basic information security within clouds, and then concludes by defining a set of threats and attacks common to public cloud environments. The cloud security mechanisms covered in [Chapter 10](#) establish the security controls used to counter these threats.

6.1. Basic Terms and Concepts

Information security is a complex ensemble of techniques, technologies, regulations, and behaviors that collaboratively protect the integrity of and access to computer systems and data. IT security measures aim to

defend against threats and interference that arise from both malicious intent and unintentional user error.

The upcoming sections define fundamental security terms relevant to cloud computing and describe associated concepts.

Confidentiality

Confidentiality is the characteristic of something being made accessible only to authorized parties (Figure 6.1). Within cloud environments, confidentiality primarily pertains to restricting access to data in transit and storage.

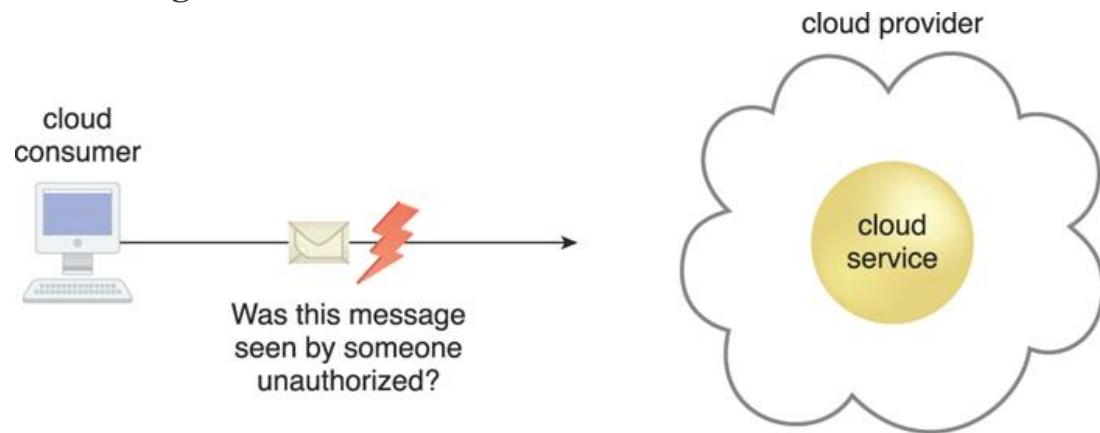


Figure 6.1. The message issued by the cloud consumer to the cloud service is considered confidential only if it is not accessed or read by an unauthorized party.

Integrity

Integrity is the characteristic of not having been altered by an unauthorized party (Figure 6.2). An important issue that concerns data integrity in the cloud is whether a cloud consumer can be guaranteed that the data it transmits to a cloud service matches the data received by that cloud service. Integrity can extend to how data is stored, processed, and retrieved by cloud services and cloud-based IT resources.

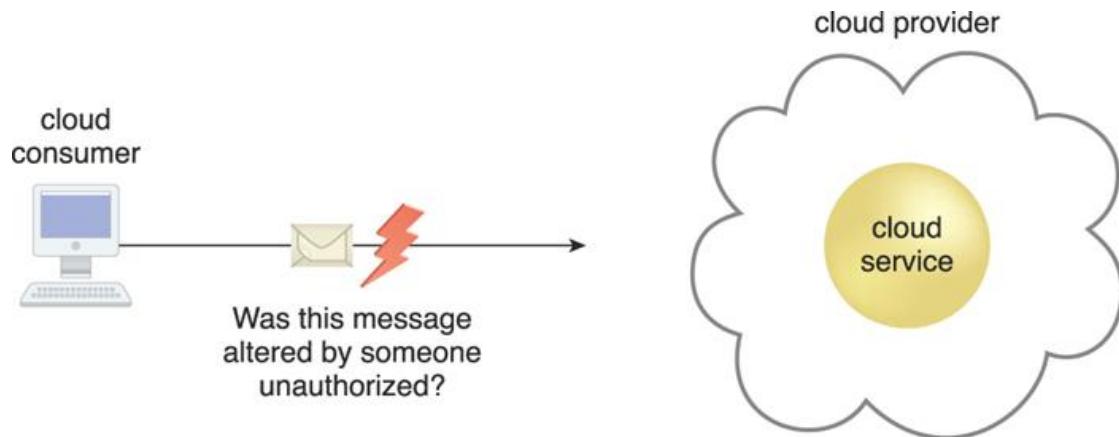


Figure 6.2. The message issued by the cloud consumer to the cloud service is considered to have integrity if it has not been altered.

Authenticity

Authenticity is the characteristic of something having been provided by an authorized source. This concept encompasses non-repudiation, which is the inability of a party to deny or challenge the authentication of an interaction. Authentication in non-repudiable interactions provides proof that these interactions are uniquely linked to an authorized source. For example, a user may not be able to access a non-repudiable file after its receipt without also generating a record of this access.

Availability

Availability is the characteristic of being accessible and usable during a specified time period. In typical cloud environments, the availability of cloud services can be a responsibility that is shared by the cloud provider and the cloud carrier. The availability of a cloud-based solution that extends to cloud service consumers is further shared by the cloud consumer.

Threat

A *threat* is a potential security violation that can challenge defenses in an attempt to breach privacy and/or cause harm. Both manually and automatically instigated threats are designed to exploit known weaknesses, also referred to as vulnerabilities. A threat that is carried out results in an *attack*.

Vulnerability

A *vulnerability* is a weakness that can be exploited either because it is protected by insufficient security controls, or because existing security controls are overcome by an attack. IT resource vulnerabilities can have a range of causes, including configuration deficiencies, security policy weaknesses, user errors, hardware or firmware flaws, software bugs, and poor security architecture.

Risk

Risk is the possibility of loss or harm arising from performing an activity. Risk is typically measured according to its threat level and the number of possible or known vulnerabilities. Two metrics that can be used to determine risk for an IT resource are:

- the probability of a threat occurring to exploit vulnerabilities in the IT resource
 - the expectation of loss upon the IT resource being compromised
- Details regarding risk management are covered later in this chapter.

Security Controls

Security controls are countermeasures used to prevent or respond to security threats and to reduce or avoid risk. Details on how to use security countermeasures are typically outlined in the security policy, which contains a set of rules and practices specifying how to implement a system, service, or security plan for maximum protection of sensitive and critical IT resources.

Security Mechanisms

Countermeasures are typically described in terms of security mechanisms, which are components comprising a defensive framework that protects IT resources, information, and services.

Security Policies

A security policy establishes a set of security rules and regulations. Often, security policies will further define how these rules and regulations are implemented and enforced. For example, the positioning and usage of security controls and mechanisms can be determined by security policies.

Summary of Key Points

- Confidentiality, integrity, authenticity, and availability are characteristics that can be associated with measuring security.
- Threats, vulnerabilities, and risks are associated with measuring and assessing insecurity, or the lack of security.
- Security controls, mechanisms, and policies are associated with establishing countermeasures and safeguards in support of improving security.

6.2. Threat Agents

A *threat agent* is an entity that poses a threat because it is capable of carrying out an attack. Cloud security threats can originate either internally or externally, from humans or software programs. Corresponding threat agents are described in the upcoming sections. [Figure 6.3](#) illustrates the role a threat agent assumes in relation to vulnerabilities, threats, and risks, and the safeguards established by security policies and security mechanisms.

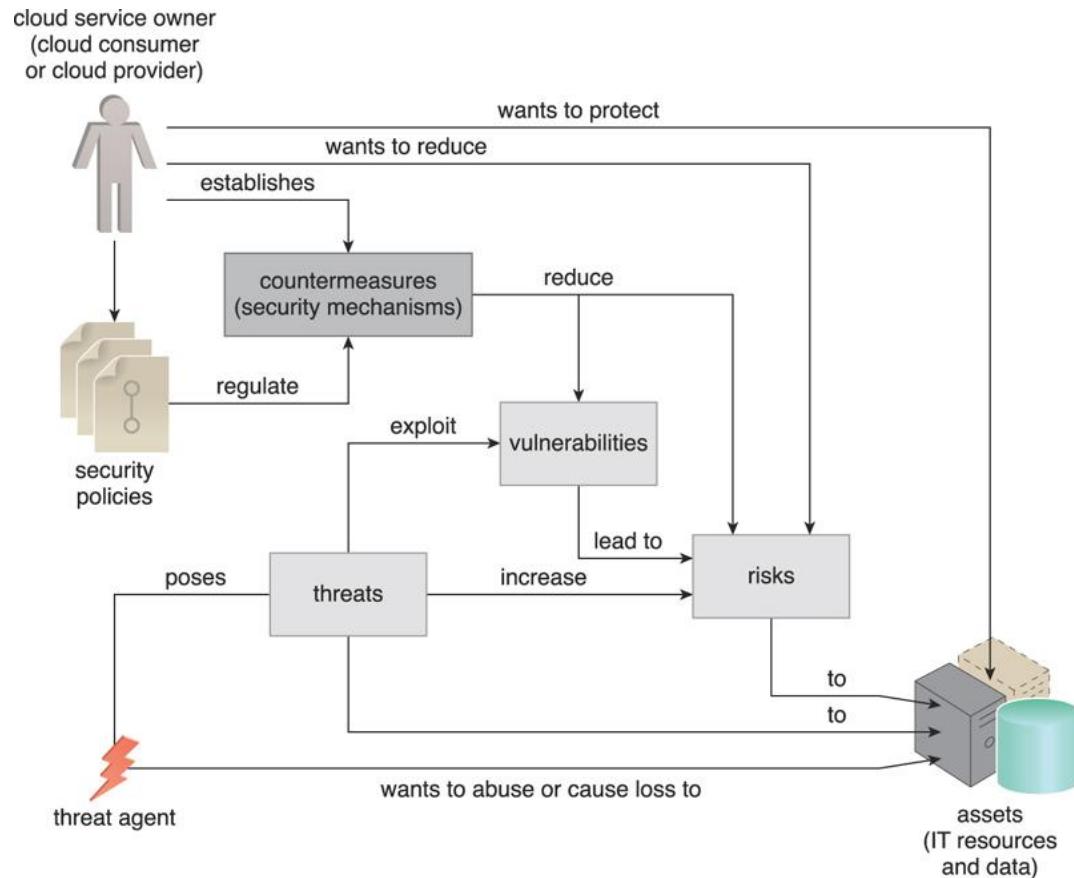


Figure 6.3. How security policies and security mechanisms are used to counter threats, vulnerabilities, and risks caused by threat agents.

Anonymous Attacker

An *anonymous attacker* is a non-trusted cloud service consumer without permissions in the cloud ([Figure 6.4](#)). It typically exists as an external software program that launches network-level attacks through public networks. When anonymous attackers have limited information on security policies and defenses, it can inhibit their ability to formulate effective attacks. Therefore, anonymous attackers often resort to committing acts like bypassing user accounts or stealing user credentials, while using methods that either ensure anonymity or require substantial resources for prosecution.



Figure 6.4. The notation used for an anonymous attacker.

Malicious Service Agent

A *malicious service agent* is able to intercept and forward the network traffic that flows within a cloud ([Figure 6.5](#)). It typically exists as a service agent (or a program pretending to be a service agent) with compromised or malicious logic. It may also exist as an external program able to remotely intercept and potentially corrupt message contents.



Figure 6.5. The notation used for a malicious service agent.

Trusted Attacker

A *trusted attacker* shares IT resources in the same cloud environment as the cloud consumer and attempts to exploit legitimate credentials to target cloud providers and the cloud tenants with whom they share IT resources ([Figure 6.6](#)). Unlike anonymous attackers (which are non-trusted), trusted attackers usually launch their attacks from within a cloud's trust boundaries by abusing legitimate credentials or via the appropriation of sensitive and confidential information.



Figure 6.6. The notation that is used for a trusted attacker.

Trusted attackers (also known as *malicious tenants*) can use cloud-based IT resources for a wide range of exploitations, including the hacking of weak authentication processes, the breaking of encryption, the spamming of e-mail accounts, or to launch common attacks, such as denial of service campaigns.

Malicious Insider

Malicious insiders are human threat agents acting on behalf of or in relation to the cloud provider. They are typically current or former employees or third parties with access to the cloud provider's premises. This type of threat agent carries tremendous damage potential, as the malicious insider may have administrative privileges for accessing cloud consumer IT resources.

Note

A notation used to represent a general form of human-driven attack is the workstation combined with a lightning bolt (Figure 6.7). This generic symbol does not imply a specific threat agent, only that an attack was initiated via a workstation.



Figure 6.7. The notation used for an attack originating from a workstation. The human symbol is optional.

Summary of Key Points

- An anonymous attacker is a non-trusted threat agent that usually attempts attacks from outside of a cloud's boundary.
- A malicious service agent intercepts network communication in an attempt to maliciously use or augment the data.
- A trusted attacker exists as an authorized cloud service consumer with legitimate credentials that it uses to exploit access to cloud-based IT resources.
- A malicious insider is a human that attempts to abuse access privileges to cloud premises.

6.3. Cloud Security Threats

This section introduces several common threats and vulnerabilities in cloud-based environments and describes the roles of the aforementioned

threat agents. Security mechanisms that are used to counter these threats are covered in [Chapter 10](#).

Traffic Eavesdropping

Traffic eavesdropping occurs when data being transferred to or within a cloud (usually from the cloud consumer to the cloud provider) is passively intercepted by a malicious service agent for illegitimate information gathering purposes ([Figure 6.8](#)). The aim of this attack is to directly compromise the confidentiality of the data and, possibly, the confidentiality of the relationship between the cloud consumer and cloud provider. Because of the passive nature of the attack, it can more easily go undetected for extended periods of time.

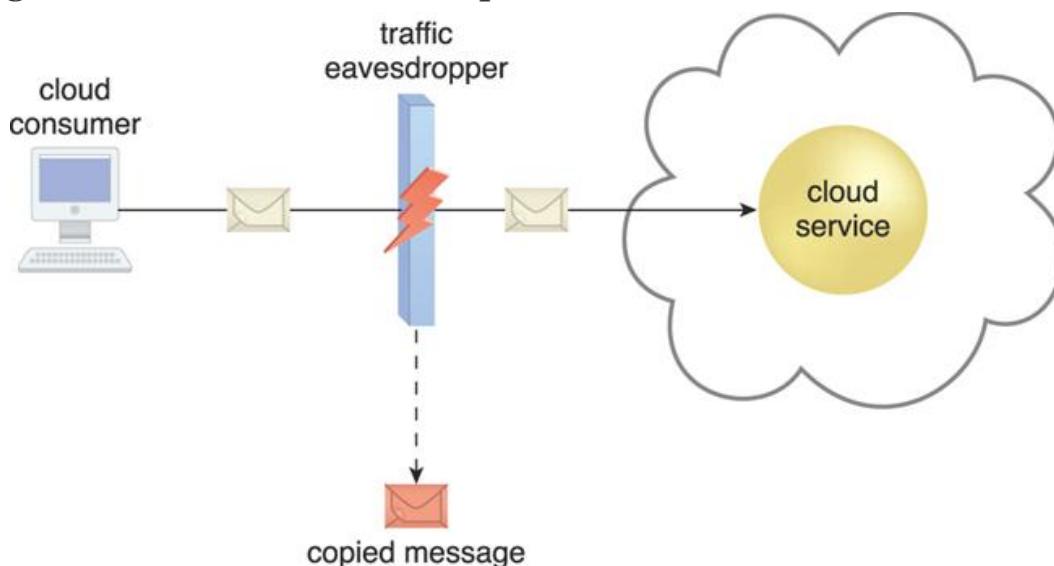


Figure 6.8. An externally positioned malicious service agent carries out a traffic eavesdropping attack by intercepting a message sent by the cloud service consumer to the cloud service. The service agent makes an unauthorized copy of the message before it is sent along its original path to the cloud service.

Malicious Intermediary

The *malicious intermediary* threat arises when messages are intercepted and altered by a malicious service agent, thereby potentially compromising the message's confidentiality and/or integrity. It may also insert harmful data into the message before forwarding it to its destination. [Figure 6.9](#) illustrates a common example of the malicious intermediary attack.

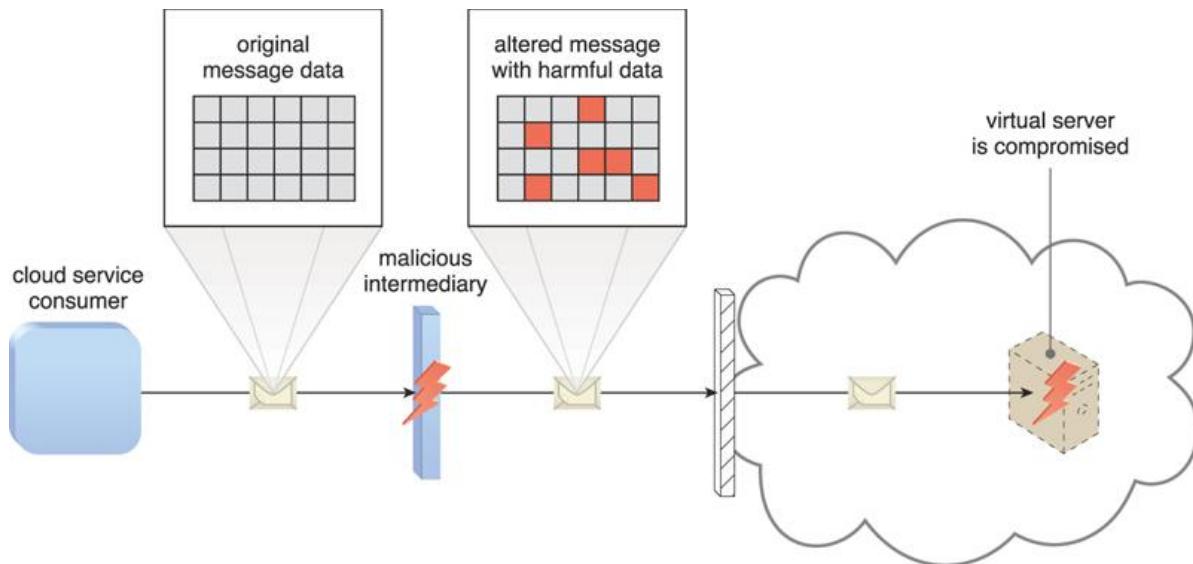


Figure 6.9. The malicious service agent intercepts and modifies a message sent by a cloud service consumer to a cloud service (not shown) being hosted on a virtual server. Because harmful data is packaged into the message, the virtual server is compromised.

Note

While not as common, the malicious intermediary attack can also be carried out by a malicious cloud service consumer program.

Denial of Service

The objective of the denial of service (DoS) attack is to overload IT resources to the point where they cannot function properly. This form of attack is commonly launched in one of the following ways:

- The workload on cloud services is artificially increased with imitation messages or repeated communication requests.
- The network is overloaded with traffic to reduce its responsiveness and cripple its performance.
- Multiple cloud service requests are sent, each of which is designed to consume excessive memory and processing resources.

Successful DoS attacks produce server degradation and/or failure, as illustrated in [Figure 6.10](#).

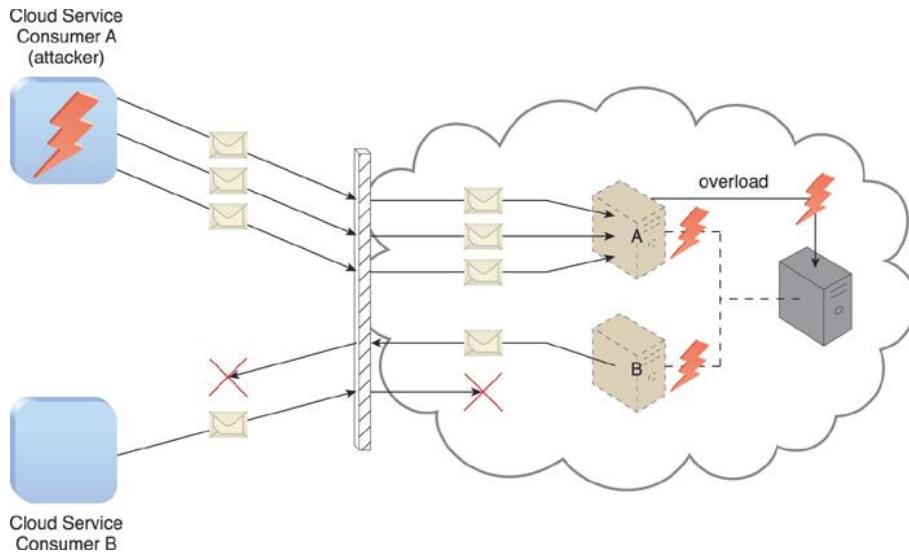


Figure 6.10. Cloud Service Consumer A sends multiple messages to a cloud service (not shown) hosted on Virtual Server A. This overloads the capacity of the underlying physical server, which causes outages with Virtual Servers A and B. As a result, legitimate cloud service consumers, such as Cloud Service Consumer B, become unable to communicate with any cloud services hosted on Virtual Servers A and B.

Insufficient Authorization

The insufficient authorization attack occurs when access is granted to an attacker erroneously or too broadly, resulting in the attacker getting access to IT resources that are normally protected. This is often a result of the attacker gaining direct access to IT resources that were implemented under the assumption that they would only be accessed by trusted consumer programs ([Figure 6.11](#)).

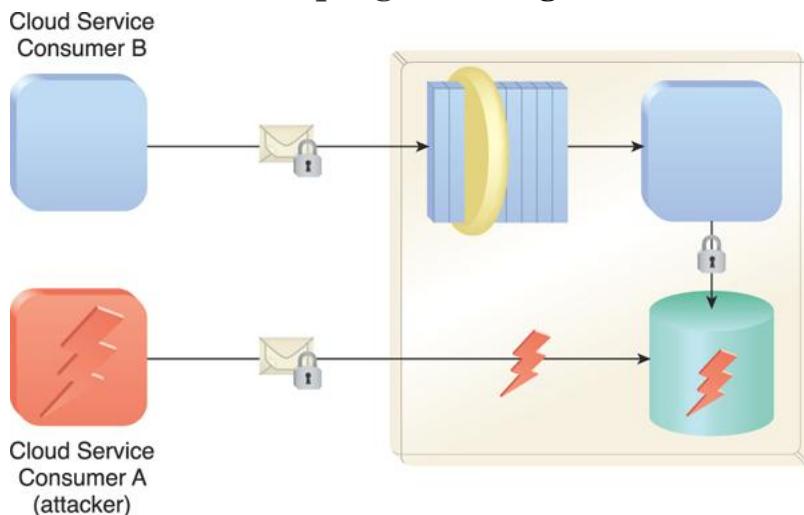


Figure 6.11. Cloud Service Consumer A gains access to a database that was implemented under the assumption that it would only be accessed through a Web service with a published service contract (as per Cloud Service Consumer B).

A variation of this attack, known as *weak authentication*, can result when weak passwords or shared accounts are used to protect IT resources. Within cloud environments, these types of attacks can lead to significant impacts depending on the range of IT resources and the range of access to those IT resources the attacker gains ([Figure 6.12](#)).

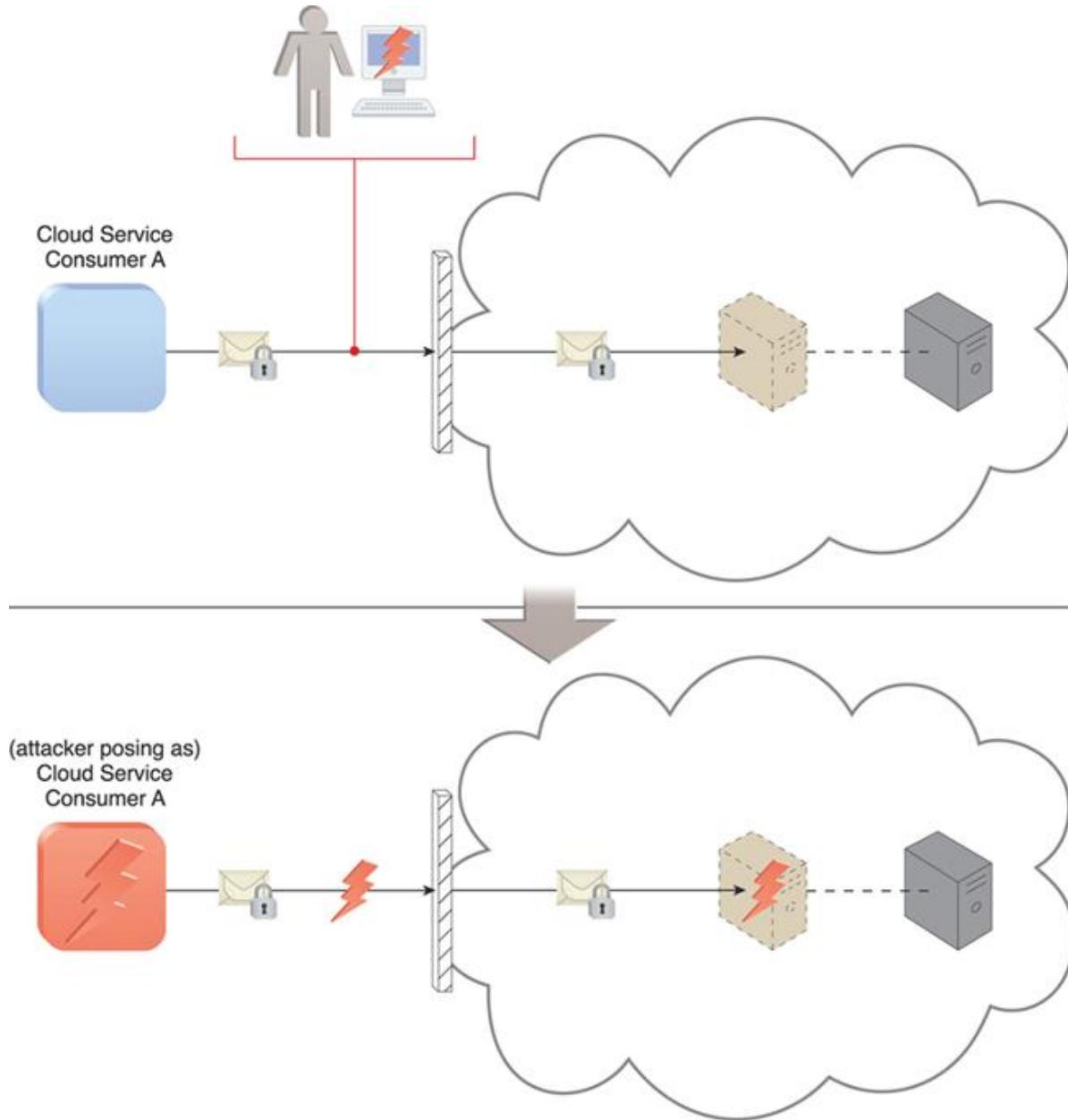


Figure 6.12. An attacker has cracked a weak password used by Cloud Service Consumer A. As a result, a malicious cloud service consumer (owned by the attacker) is designed to pose as Cloud Service Consumer A in order to gain access to the cloud-based virtual server.

Virtualization Attack

Virtualization provides multiple cloud consumers with access to IT resources that share underlying hardware but are logically isolated from each other. Because cloud providers grant cloud consumers administrative access to virtualized IT resources (such as virtual servers), there is an inherent risk that cloud consumers could abuse this access to attack the underlying physical IT resources.

A *virtualization attack* exploits vulnerabilities in the virtualization platform to jeopardize its confidentiality, integrity, and/or availability. This threat is illustrated in [Figure 6.13](#), where a trusted attacker successfully accesses a virtual server to compromise its underlying physical server. With public clouds, where a single physical IT resource may be providing virtualized IT resources to multiple cloud consumers, such an attack can have significant repercussions.

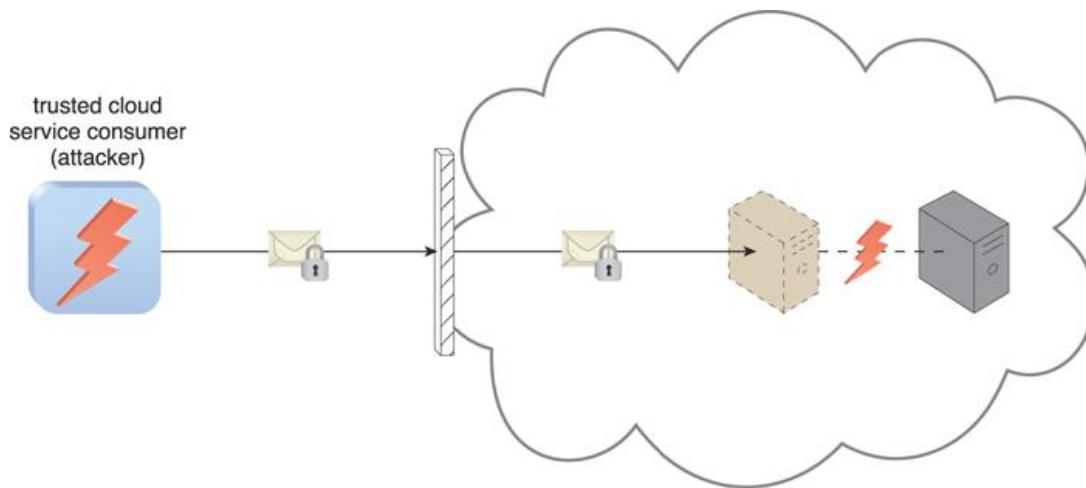


Figure 6.13. An authorized cloud service consumer carries out a virtualization attack by abusing its administrative access to a virtual server to exploit the underlying hardware.

Overlapping Trust Boundaries

If physical IT resources within a cloud are shared by different cloud service consumers, these cloud service consumers have overlapping trust boundaries. Malicious cloud service consumers can target shared IT resources with the intention of compromising cloud consumers or other IT resources that share the same trust boundary. The consequence is that some or all of the other cloud service consumers could be impacted by the attack and/or the attacker could use virtual IT resources against others that happen to also share the same trust boundary.

Figure 6.14 illustrates an example in which two cloud service consumers share virtual servers hosted by the same physical server and, resultantly, their respective trust boundaries overlap.

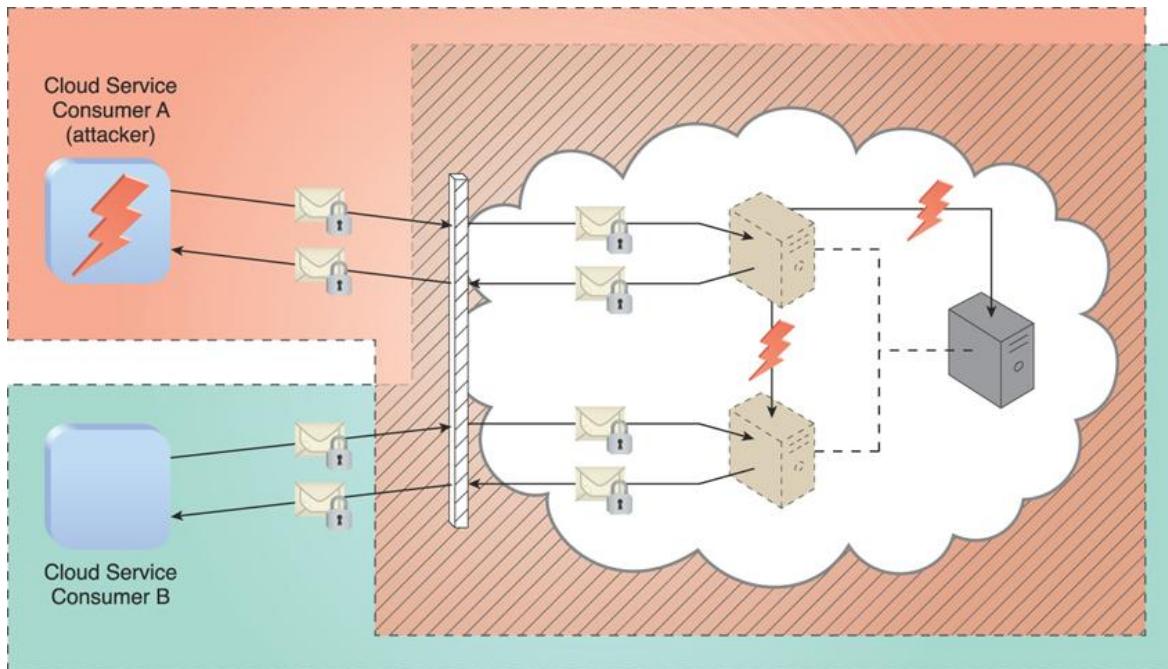


Figure 6.14. Cloud Service Consumer A is trusted by the cloud and therefore gains access to a virtual server, which it then attacks with the intention of attacking the underlying physical server and the virtual server used by Cloud Service Consumer B.

Summary of Key Points

- Traffic eavesdropping and malicious intermediary attacks are usually carried out by malicious service agents that intercept network traffic.
- A denial of service attack occurs when a targeted IT resource is overloaded with requests in an attempt to cripple or render it unavailable. The insufficient authorization attack occurs when access is granted to an attacker erroneously or too broadly, or when weak passwords are used.
- A virtualization attack exploits vulnerabilities within virtualized environments to gain unauthorized access to underlying physical hardware. Overlapping trust boundaries represent a threat whereby attackers can exploit cloud-based IT resources shared by multiple cloud consumers.

6.4. Additional Considerations

This section provides a diverse checklist of issues and guidelines that relate to cloud security. The listed considerations are in no particular order.

Flawed Implementations

The substandard design, implementation, or configuration of cloud service deployments can have undesirable consequences, beyond runtime exceptions and failures. If the cloud provider's software and/or hardware have inherent security flaws or operational weaknesses, attackers can exploit these vulnerabilities to impair the integrity, confidentiality, and/or availability of cloud provider IT resources and cloud consumer IT resources hosted by the cloud provider.

Figure 6.15 depicts a poorly implemented cloud service that results in a server shutdown. Although in this scenario the flaw is exposed accidentally by a legitimate cloud service consumer, it could have easily been discovered and exploited by an attacker.

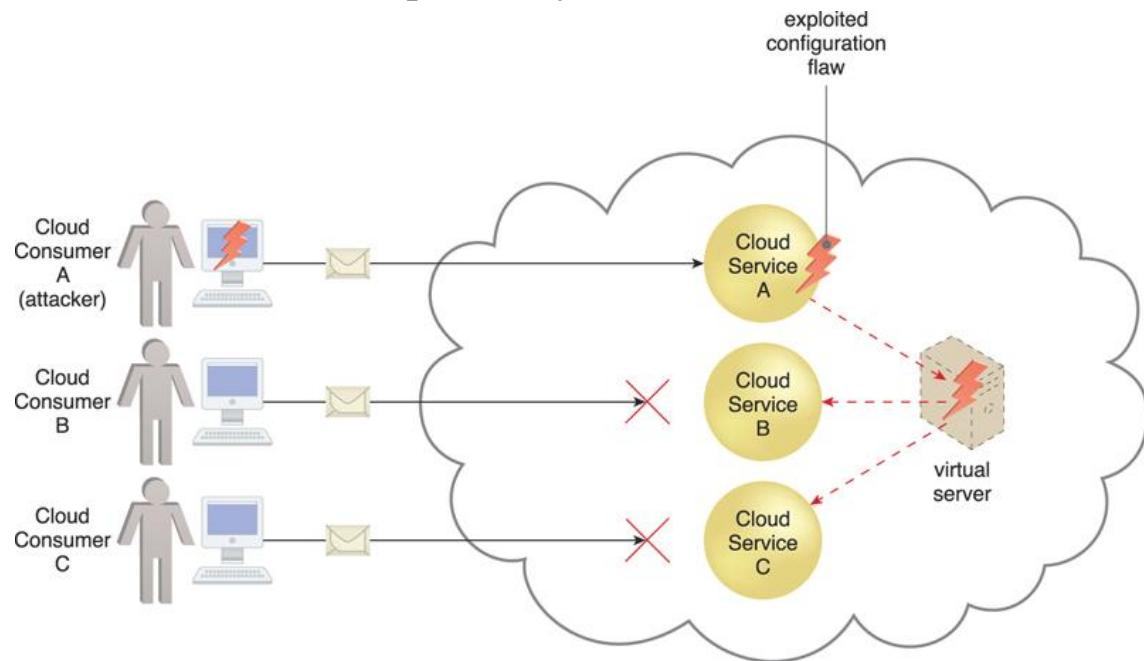


Figure 6.15. Cloud Service Consumer A's message triggers a configuration flaw in Cloud Service A, which in turn causes the virtual server that is also hosting Cloud Services B and C to crash.

Security Policy Disparity

When a cloud consumer places IT resources with a public cloud provider, it may need to accept that its traditional information security approach may not be identical or even similar to that of the cloud provider. This incompatibility needs to be assessed to ensure that any data or other IT assets being relocated to a public cloud are adequately protected. Even when leasing raw infrastructure-based IT resources, the cloud consumer may not be granted sufficient administrative control or influence over

security policies that apply to the IT resources leased from the cloud provider. This is primarily because those IT resources are still legally owned by the cloud provider and continue to fall under its responsibility. Furthermore, with some public clouds, additional third parties, such as security brokers and certificate authorities, may introduce their own distinct set of security policies and practices, further complicating any attempts to standardize the protection of cloud consumer assets.

Contracts

Cloud consumers need to carefully examine contracts and SLAs put forth by cloud providers to ensure that security policies, and other relevant guarantees, are satisfactory when it comes to asset security. There needs to be clear language that indicates the amount of liability assumed by the cloud provider and/or the level of indemnity the cloud provider may ask for. The greater the assumed liability by the cloud provider, the lower the risk to the cloud consumer.

Another aspect to contractual obligations is where the lines are drawn between cloud consumer and cloud provider assets. A cloud consumer that deploys its own solution upon infrastructure supplied by the cloud provider will produce a technology architecture comprised of artifacts owned by both the cloud consumer and cloud provider. If a security breach (or other type of runtime failure) occurs, how is blame determined? Furthermore, if the cloud consumer can apply its own security policies to its solution, but the cloud provider insists that its supporting infrastructure be governed by different (and perhaps incompatible) security policies, how can the resulting disparity be overcome?

Sometimes the best solution is to look for a different cloud provider with more compatible contractual terms.

Risk Management

When assessing the potential impacts and challenges pertaining to cloud adoption, cloud consumers are encouraged to perform a formal risk assessment as part of a risk management strategy. A cyclically executed process used to enhance strategic and tactical security, risk management is comprised of a set of coordinated activities for overseeing and controlling risks. The main activities are generally defined as risk assessment, risk treatment, and risk control ([Figure 6.16](#)).

- *Risk Assessment* – In the risk assessment stage, the cloud environment is analyzed to identify potential vulnerabilities and shortcomings that threats can exploit. The cloud provider can be asked to produce statistics and other information about past attacks (successful and unsuccessful) carried out in its cloud. The identified risks are quantified and qualified according to the probability of occurrence and the degree of impact in relation to how the cloud consumer plans to utilize cloud-based IT resources.

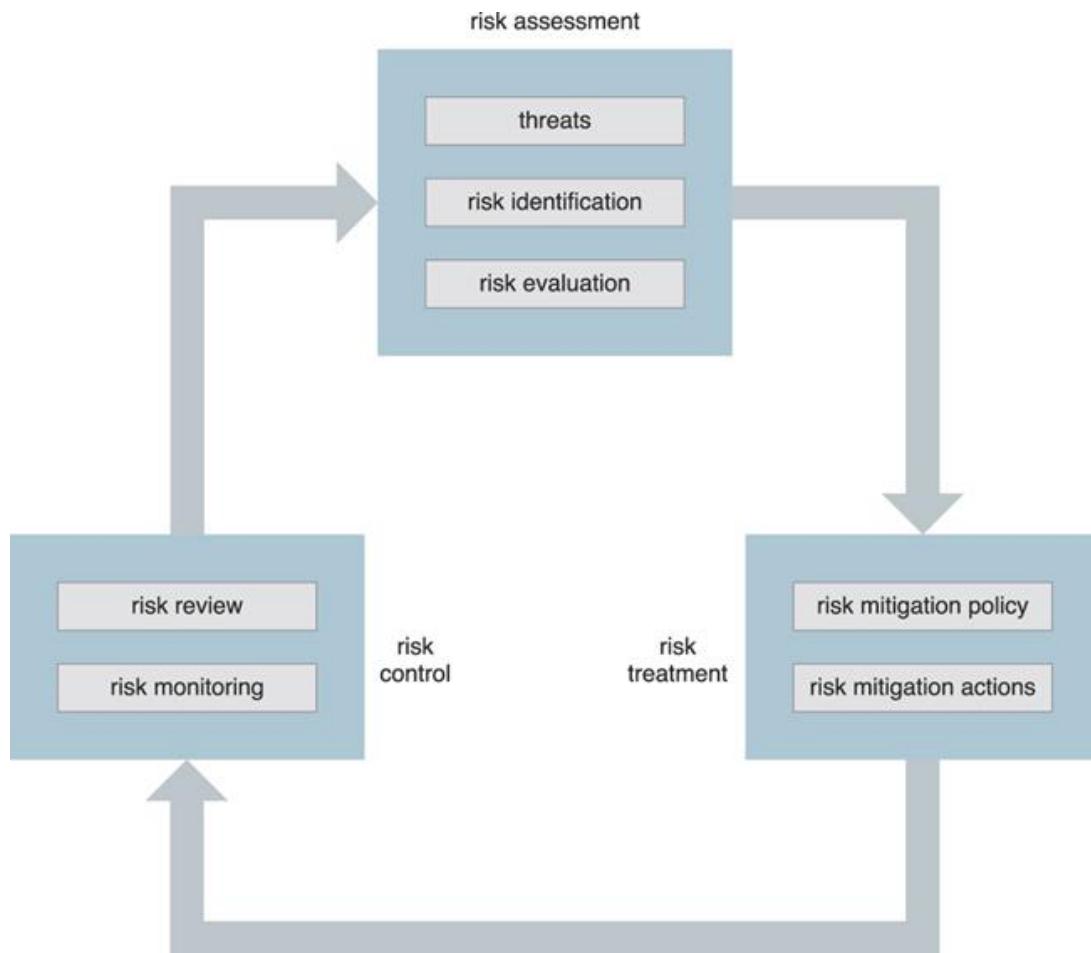


Figure 6.16. The on-going risk management process, which can be initiated from any of the three stages.

- *Risk Treatment* – Mitigation policies and plans are designed during the risk treatment stage with the intent of successfully treating the risks that were discovered during risk assessment. Some risks can be eliminated, others can be mitigated, while others can be dealt with via outsourcing or even incorporated into the insurance and/or operating loss budgets. The cloud provider itself may agree to assume responsibility as part of its contractual obligations.

- *Risk Control* – The risk control stage is related to risk monitoring, a three-step process that is comprised of surveying related events, reviewing these events to determine the effectiveness of previous assessments and treatments, and identifying any policy adjustment needs. Depending on the nature of the monitoring required, this stage may be carried out or shared by the cloud provider.

The threat agents and cloud security threats covered in this chapter (as well as others that may surface) can be identified and documented as part of the risk assessment stage. The cloud security mechanisms covered in [Chapter 10](#) can be documented and referenced as part of the corresponding risk treatment.

Summary of Key Points

- Cloud consumers need to be aware that they may be introducing security risks by deploying flawed cloud-based solutions.
- An understanding of how a cloud provider defines and imposes proprietary, and possibly incompatible, cloud security policies is a critical part of forming assessment criteria when choosing a cloud provider vendor.
- Liability, indemnity, and blame for potential security breaches need to be clearly defined and mutually understood in the legal agreements signed by cloud consumers and cloud providers.
- It is important for cloud consumers, subsequent to gaining an understanding of the potential security-related issues specific to a given cloud environment, to perform a corresponding assessment of the identified risks.

6.5. Case Study Example

Based on an assessment of its internal applications, ATN analysts identify a set of risks. One such risk is associated with the myTrendek application that was adopted from OTC, a company ATN recently acquired. This application includes a feature that analyzes telephone and Internet usage, and enables a multi-user mode that grants varying access rights. Administrators, supervisors, auditors, and regular users can therefore be assigned different privileges. The application's user-base encompasses internal users and external users, such as business partners and contractors.

The myTrendek application poses a number of security challenges pertaining to usage by internal staff:

- authentication does not require or enforce complex passwords
- communication with the application is not encrypted

- European regulations (ETelReg) require that certain types of data collected by the application be deleted after six months

ATN is planning to migrate this application to a cloud via a PaaS environment, but the weak authentication threat and the lack of confidentiality supported by the application make them reconsider. A subsequent risk assessment further reveals that if the application is migrated to a PaaS environment hosted by a cloud that resides outside of Europe, local regulations may be in conflict with ETelReg. Given that the cloud provider is not concerned with ETelReg compliance, this could easily result in monetary penalties being assessed to ATN. Based on the results of the risk assessment, ATN decides not to proceed with its cloud migration plan.

Chapter 1



Introduction

Cloud Computing Definition

- Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.
- This cloud model promotes availability and is composed of
- **Five** essential characteristics,
- **Three** service models, and
- **Four** deployment models.

Essential Characteristics:

- **On-demand self-service.** A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.
- **Broad network access.** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).
- **Resource pooling.** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.

Essential Characteristics:

- **Rapid elasticity.** Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.
- **Measured Service.** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

The Pros and Cons of Cloud Computing

- **Pros of cloud computing:**
- No cost on infrastructure
- Minimum management and cost (Cost Saving)
- Forget about administrative or management hassles
- Accessibility and pay per use
- Reliability
- **Cons of cloud computing:**
- Requires good speed internet with good bandwidth
- Limited control on infrastructure
- Restricted or limited flexibility
- Ongoing costs
- Security

Cloud Service Models

- **Cloud Software as a Service (SaaS).**

The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings

- **Cloud Platform as a Service (PaaS).**

The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages 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 application hosting environment configurations.

Cloud Service Models

- **Cloud Infrastructure as a Service (IaaS).**

The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).

Deployment Models (Cloud Types)

- **Private cloud.**

The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.

- **Public cloud.**

The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

- **Hybrid cloud.**

The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).

- **Community cloud.**

The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.

Deployment Models (Cloud Types)

- **Private cloud.**

The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.

- **Public cloud.**

The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

- **Hybrid cloud.**

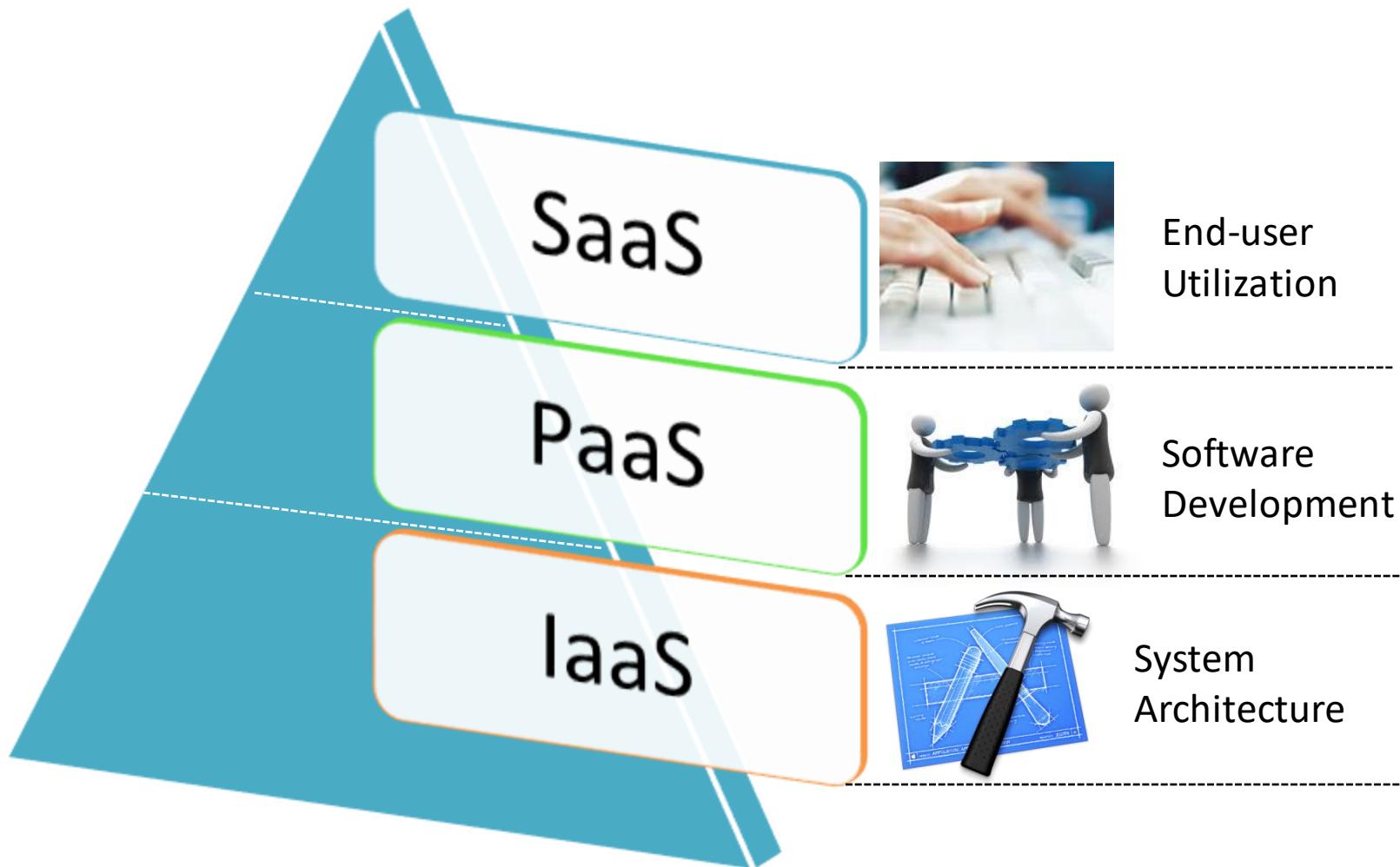
The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).

- **Community cloud.**

The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.



Comparison of SAAS PASS IAAS



Comparison of SAAS PASS IAAS

SaaS

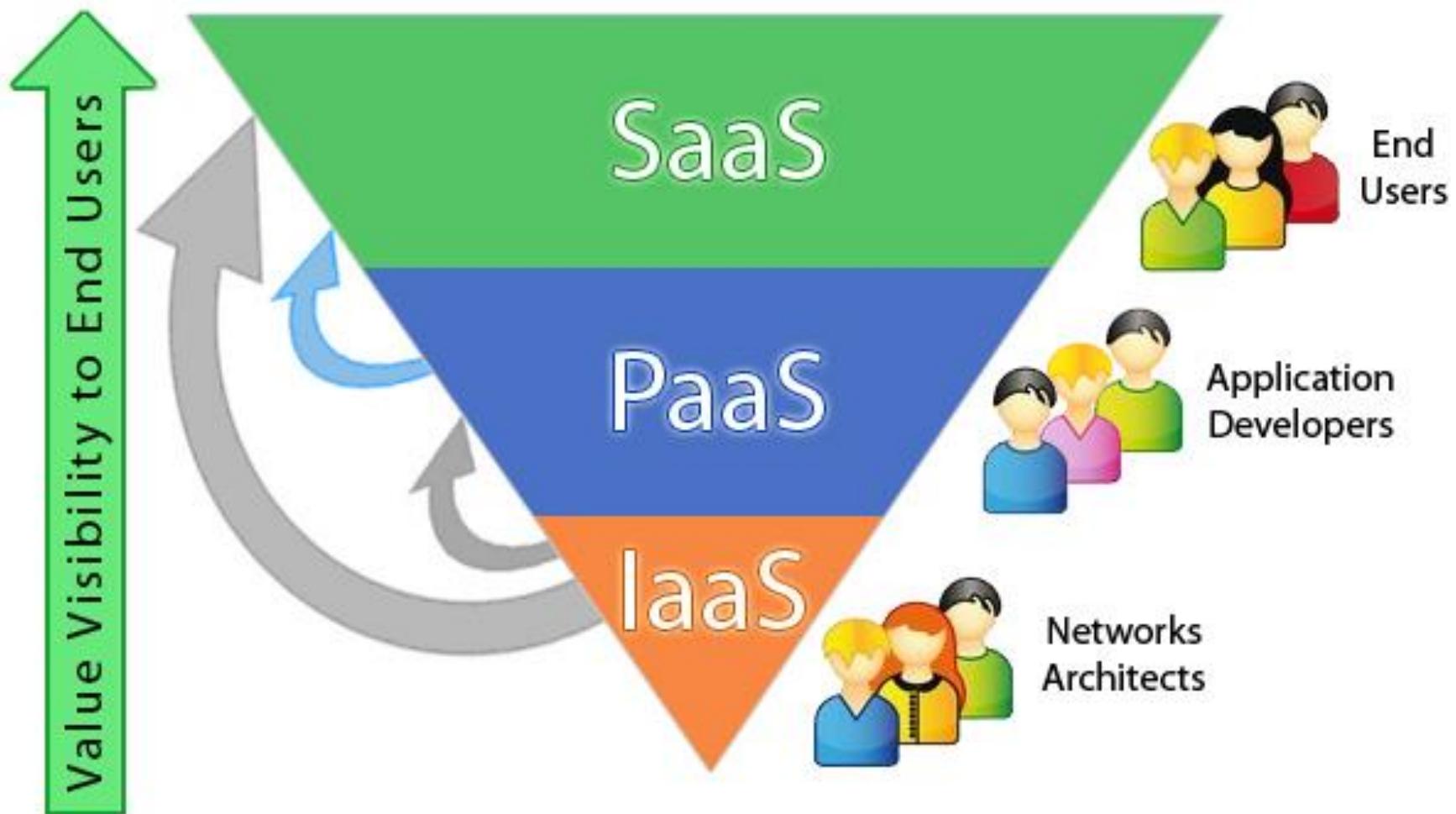
- Software as a service
- Operating environment largely irrelevant, fully functional applications provided, e.g. CRM, ERP, email

PaaS

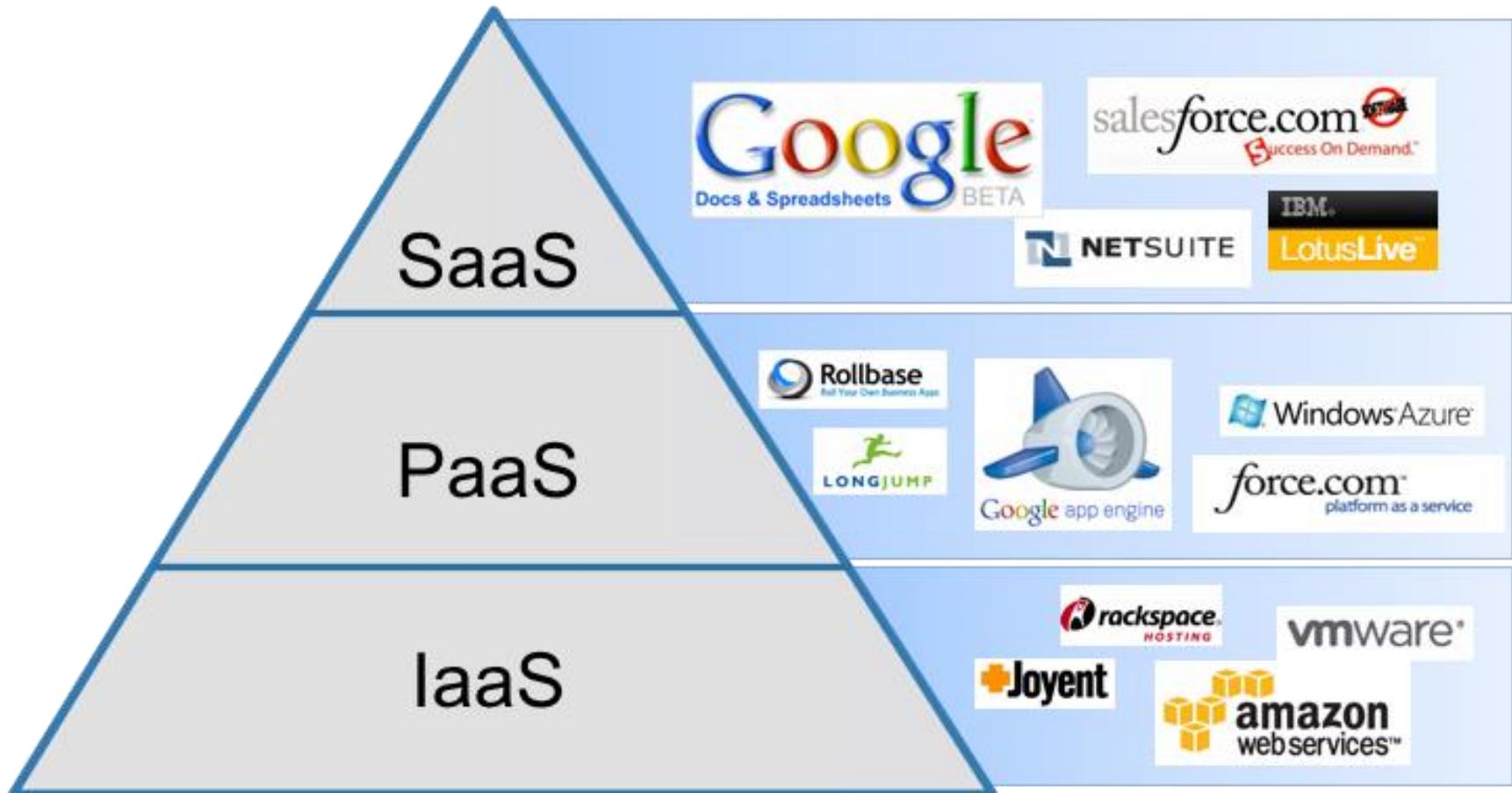
- Platform as a service
- Operating environment included, e.g. Windows/.NET, Linux/J2EE, applications of choice deployed

IaaS

- Infrastructure as a service
- Virtual platform on which required operating environment and application are deployed
- Includes storage as a service offerings



Example of vendors



Benefits of CC



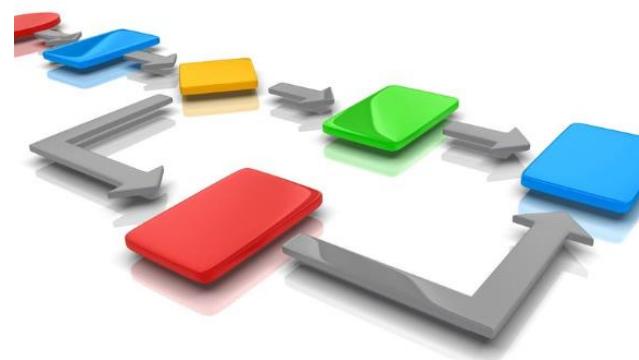
Global Infrastructure Savings



Smart Resource Provisioning



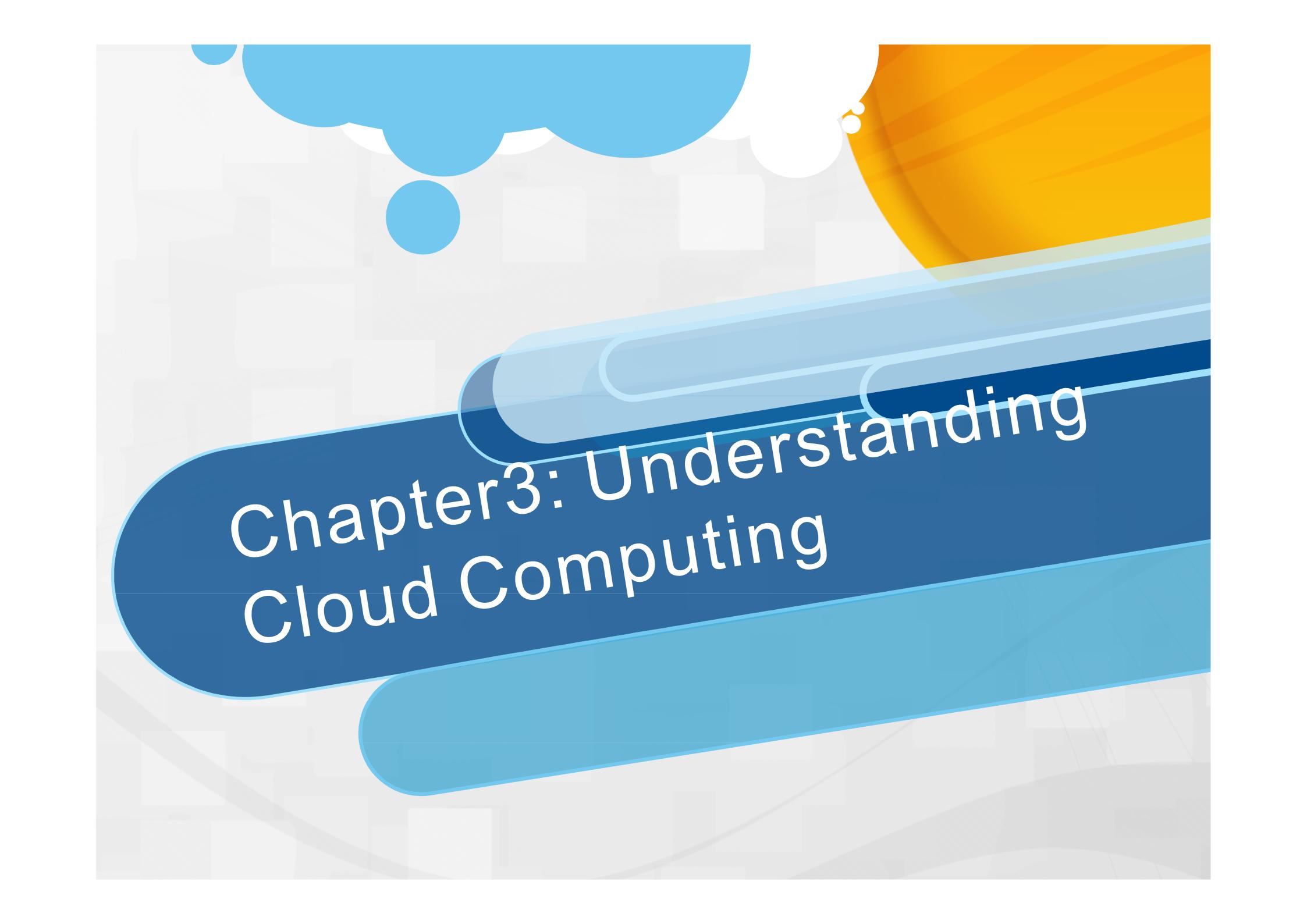
Better Delivery Time



Transparent Workflow

Thank You





Chapter3: Understanding Cloud Computing



A Brief History

- The general public has been leveraging forms of Internet-based computer utilities since the mid-1990s .
- In the late 1990s, Salesforce.com pioneered the notion of bringing remotely provisioned services into the enterprise.
- In 2002, Amazon.com launched the Amazon Web Services (AWS) platform, a suite of enterprise-oriented services that provide remotely provisioned storage, computing resources, and business functionality.



Definitions

the National Institute of Standards and Technology (NIST)

- “*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models.*”



Definitions

- “*Cloud computing is a specialized form of distributed computing that introduces utilization models for remotely provisioning scalable and measured resources.*”



Basic Concepts and Terminology

- **Cloud:**

a distinct IT environment that is designed for the purpose of remotely provisioning scalable and measured IT resources.



Cloud

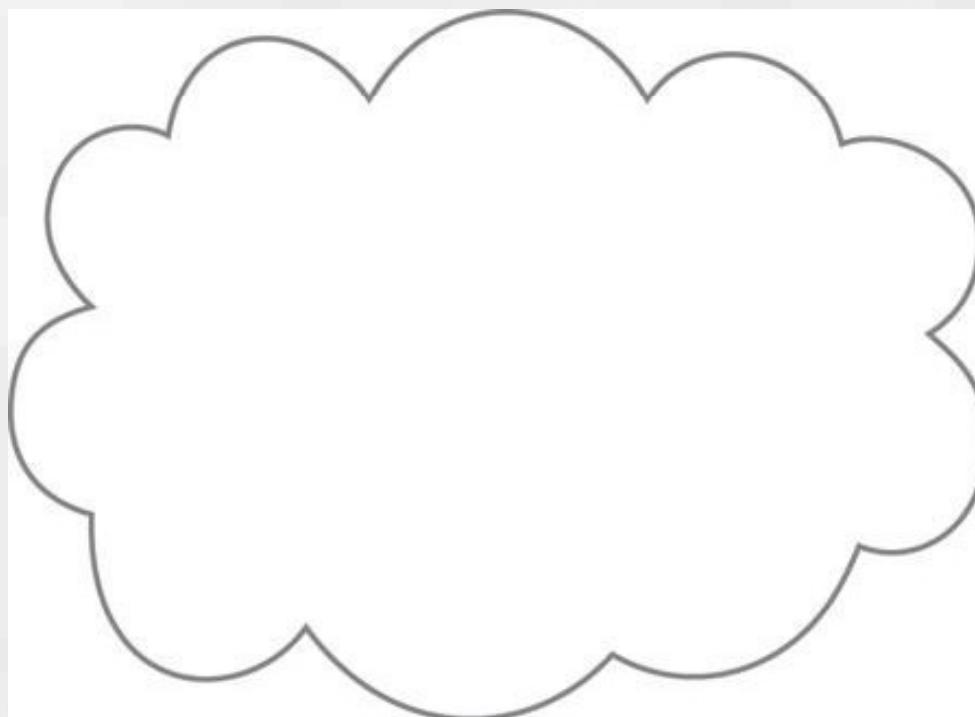


Figure 3.1 The symbol used to denote the boundary of a cloud environment.



IT Resource

- a physical or virtual IT-related artifact that can be either software-based, such as a virtual server or a custom software program, or hardware-based, such as a physical server or a network device.

IT Resource

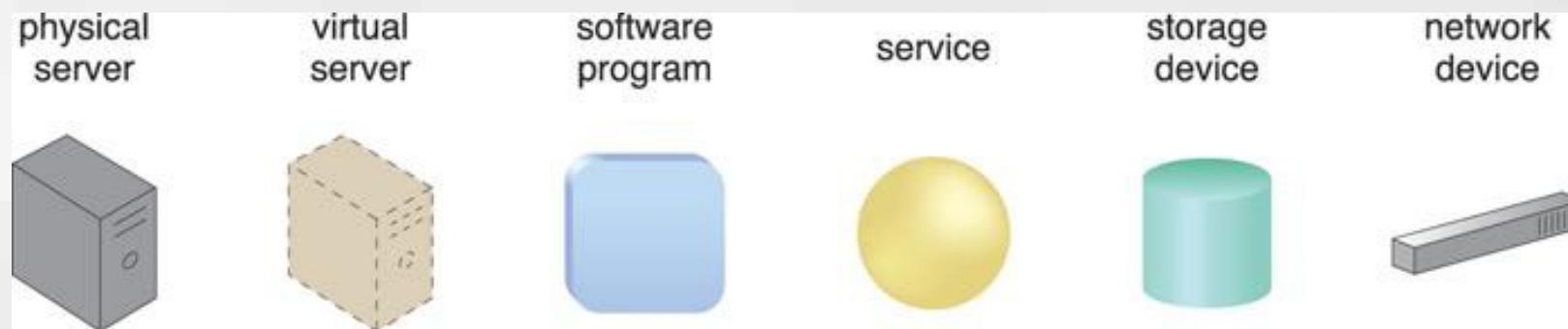


Figure 3.2 Examples of common IT resources and their corresponding symbols.

IT Resource

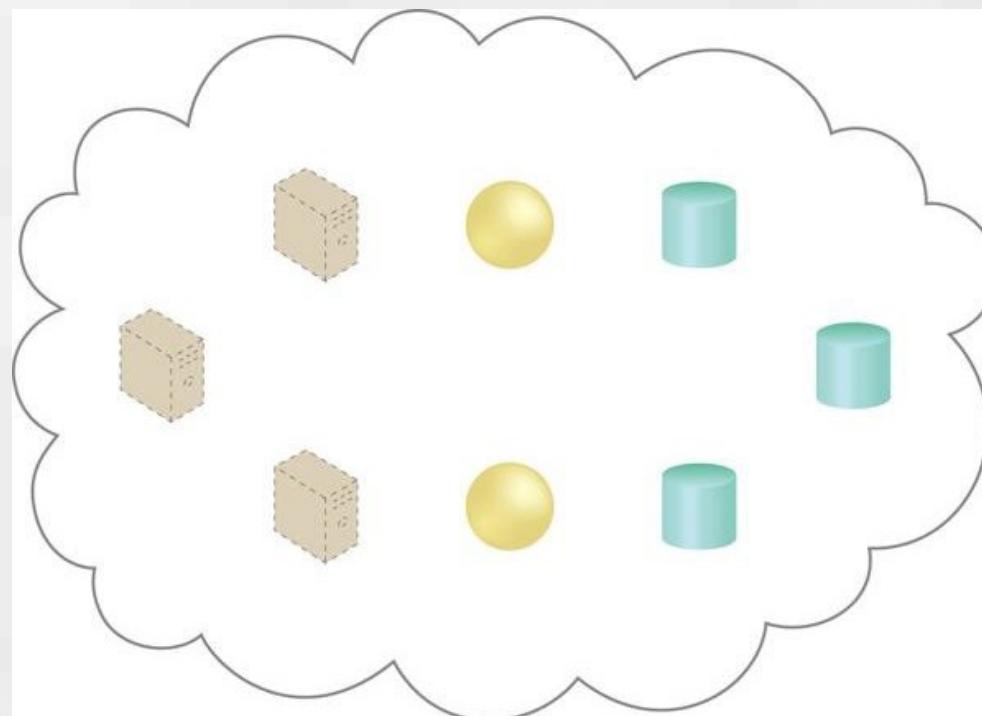


Figure 3.3 A cloud is hosting eight IT resources: three virtual servers, two cloud services, and three storage devices.



On-Premise

- An IT resource that is hosted in a conventional IT enterprise within an organizational boundary (that does not specifically represent a cloud) is considered to be located on the premises of the IT enterprise, or *on-premise* for short.
- An IT resource that is on-premise cannot be cloud-based, and vice-versa.



On-Premise

- An on-premise IT resource can access and interact with a cloud-based IT resource.
- An on-premise IT resource can be moved to a cloud, thereby changing it to a cloud-based IT resource.
- Redundant deployments of an IT resource can exist in both on-premise and cloud-based environments.



Cloud Consumers and Cloud Providers

- The party that provides cloud-based IT resources is the cloud provider.
- The party that uses cloud-based IT resources is the cloud consumer.



Scaling

- Scaling, from an IT resource perspective, represents the ability of the IT resource to handle increased or decreased usage demands.
- Horizontal Scaling* – scaling out and scaling in
- Vertical Scaling* – scaling up and scaling down



Horizontal Scaling

- The allocating or releasing of IT resources that are of the same type .
- The horizontal allocation of resources is referred to as scaling out .
- the horizontal releasing of resources is referred to as scaling in .

Horizontal Scaling

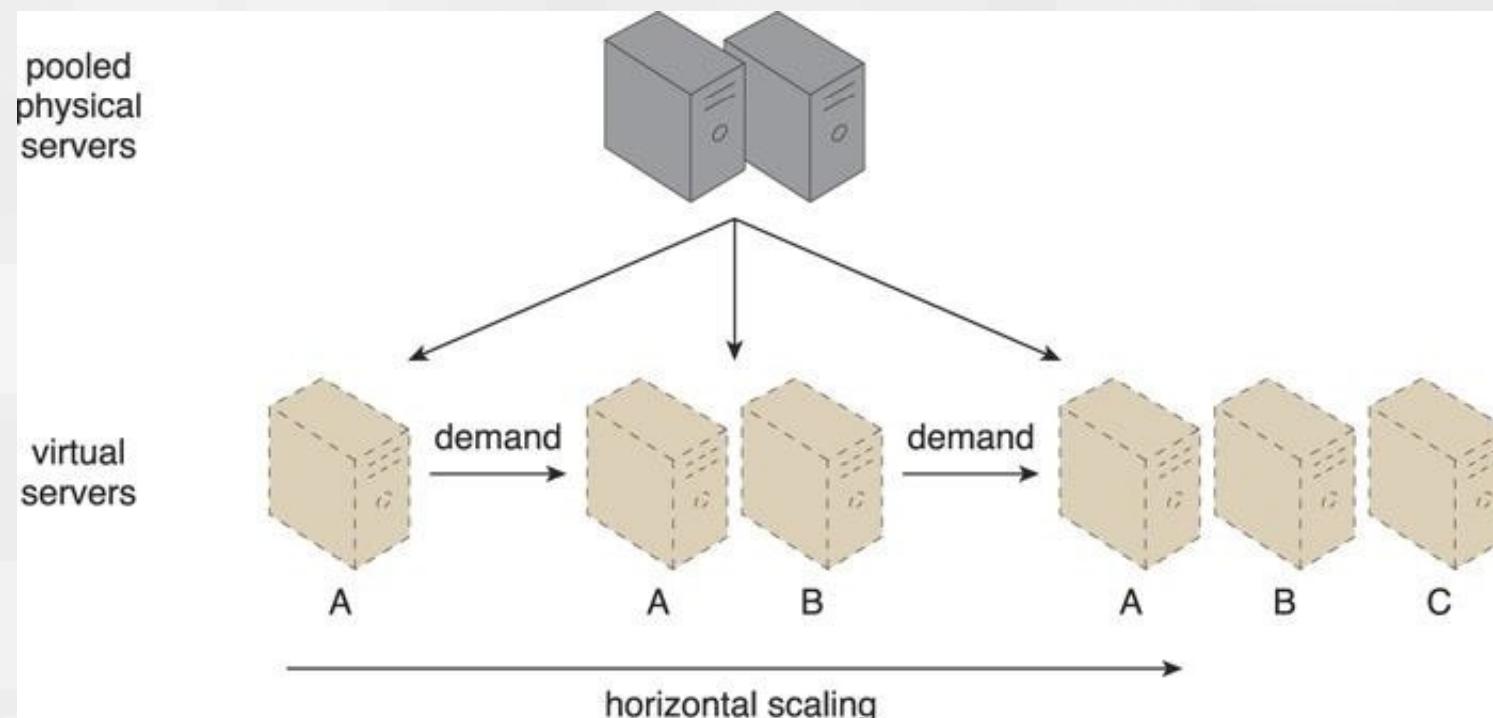


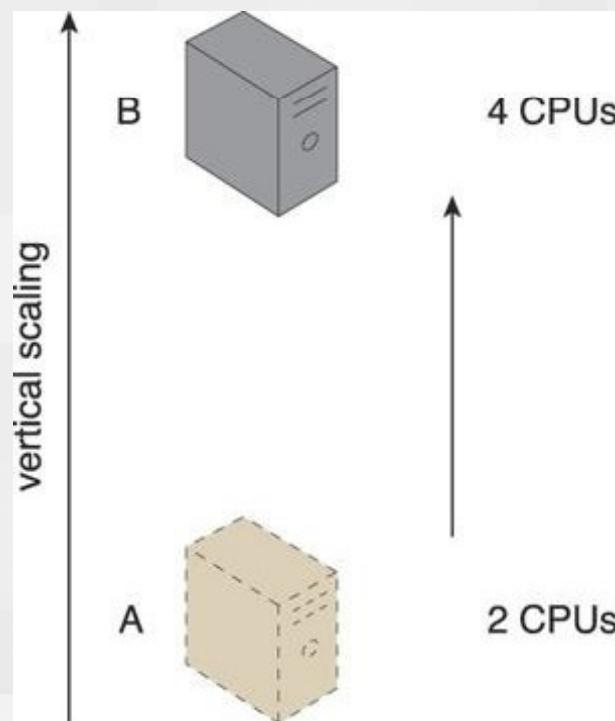
Figure 3.4 An IT resource (Virtual Server A) is scaled out by adding more of the same IT resources (Virtual Servers B and C).

Vertical Scaling

- When an existing IT resource is replaced by another with higher or lower capacity .
- The replacing of an IT resource with another that has a higher capacity is referred to as scaling up.
- The replacing an IT resource with another that has a lower capacity is considered scaling down.

Vertical Scaling

Figure 3.5 An IT resource (a virtual server with two CPUs) is scaled up by replacing it with a more powerful IT resource with increased capacity for data storage (a physical server with four CPUs).



A comparison of horizontal and vertical scaling

Horizontal Scaling	Vertical Scaling
less expensive (through commodity hardware components)	more expensive (specialized servers)
IT resources instantly available	IT resources normally instantly available
resource replication and automated scaling	additional setup is normally needed
additional IT resources needed	no additional IT resources needed
not limited by hardware capacity	limited by maximum hardware capacity



Cloud Service

- A *cloud service* is any IT resource that is made remotely accessible via a cloud.
- Cloud service usage conditions are typically expressed in a service-level agreement (SLA) that is the human-readable part of a service contract between a cloud provider and cloud consumer that describes QoS features, behaviors, and limitations of a cloud-based service or other provisions.

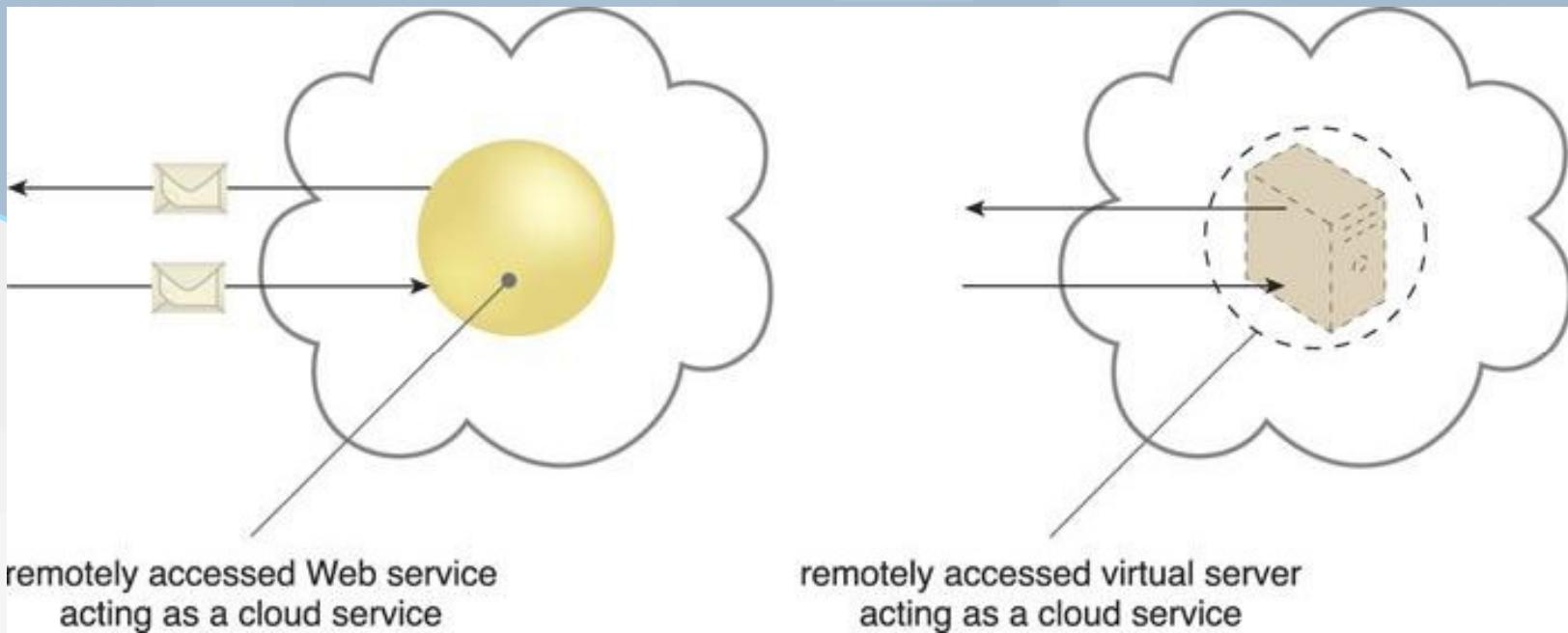


Figure 3.6 A cloud service with a published technical interface is being accessed by a consumer outside of the cloud (left). A cloud service that exists as a virtual server is also being accessed from outside of the cloud's boundary (right). The cloud service on the left is likely being invoked by a consumer program that was designed to access the cloud service's published technical interface. The cloud service on the right may be accessed by a human user that has remotely logged on to the virtual server.

Cloud Service Consumer

- The *cloud service consumer* is a temporary runtime role assumed by a software program when it accesses a cloud service.



Figure 3.7 Examples of cloud service consumers. Depending on the nature of a given diagram, an artifact labeled as a cloud service consumer may be a software program or a hardware device (in which case it is implied that it is running a software program capable of acting as a cloud service consumer).



Goals and Benefits

- **Reduced Investments and Proportional Costs**
- **Increased Scalability**
- **Increased Availability and Reliability**



Risks and Challenges

- Increased Security Vulnerabilities
- Reduced Operational Governance Control
- Limited Portability Between Cloud Providers
- Multi-Regional Compliance and Legal Issues

Chapter 4.

Fundamental Concepts and Models

Roles and Boundaries

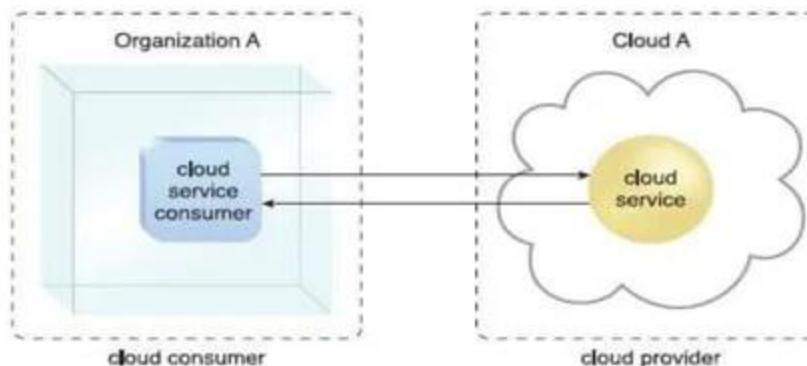
Cloud Provider

The organization that provides cloud-based IT resources is the *cloud provider*. When assuming the role of cloud provider, an organization is responsible for making cloud services available to cloud consumers, as per agreed upon SLA guarantees.

Cloud providers normally own the IT resources that are made available for lease by cloud consumers; however, some cloud providers also “resell” IT resources leased from other cloud providers.

Cloud Consumer

A *cloud consumer* is an organization (or a human) that has a formal contract or arrangement with a cloud provider to use IT resources made available by the cloud provider



Cloud Service Owner

- The person or organization that legally owns a cloud service is called a *cloud service owner*.
- The cloud service owner can be the cloud consumer, or the cloud provider that owns the cloud within which the cloud service resides.
- For example, either the cloud consumer of Cloud X or the cloud provider of Cloud X could own Cloud Service A (Figures 2.2 and 2.3).

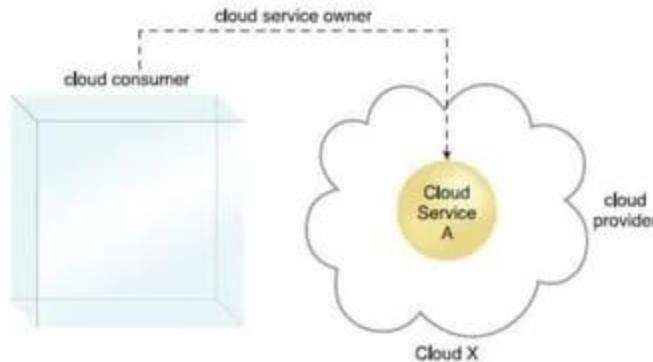
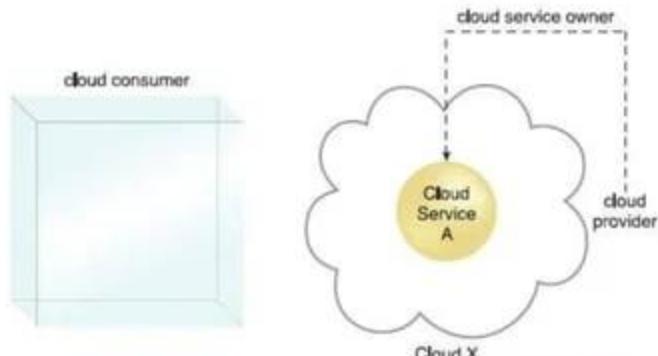


Figure 2.2

A cloud consumer can be a cloud service owner when it deploys its own service in a cloud.

Figure 2.3

A cloud provider becomes a cloud service owner if it deploys its own cloud service, typically for other cloud consumers to use.



Cloud Resource Administrator

A *cloud resource administrator* is the person or organization responsible for administering a cloud-based IT resource (including cloud services).

The cloud resource administrator can be (or belong to) the cloud consumer or cloud provider of the cloud within which the cloud service resides.

Alternatively, it can be (or belong to) a third-party organization contracted to administer the cloud-based IT resource.

For example, a cloud service owner can contract a cloud resource administrator to administer a cloud service (Figures 2.4 and 2.5).

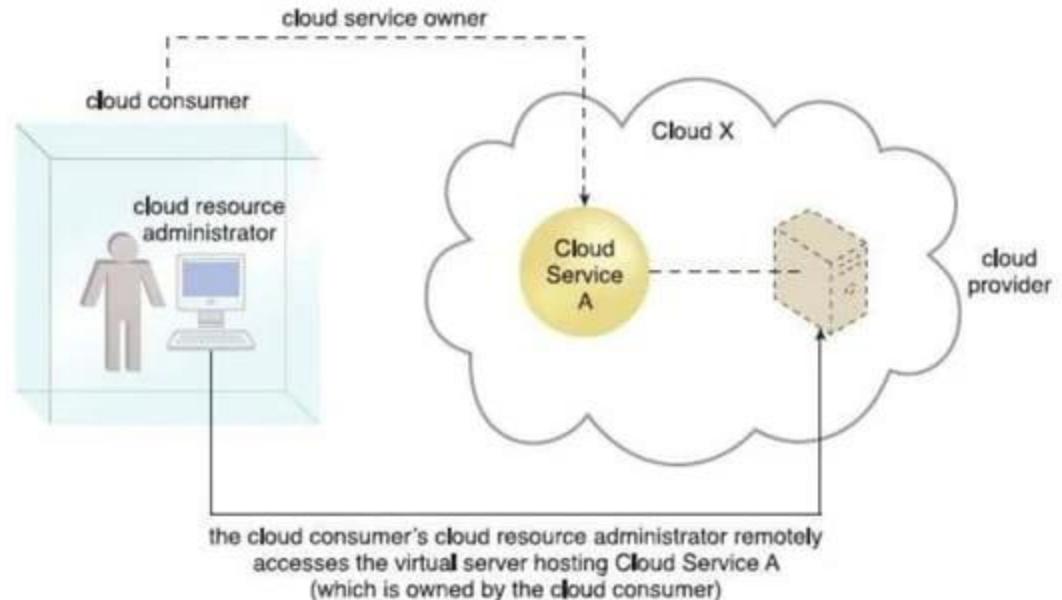


Figure 2.4

A cloud resource administrator can be with a cloud consumer organization and administer remotely accessible IT resources that belong to the cloud consumer.

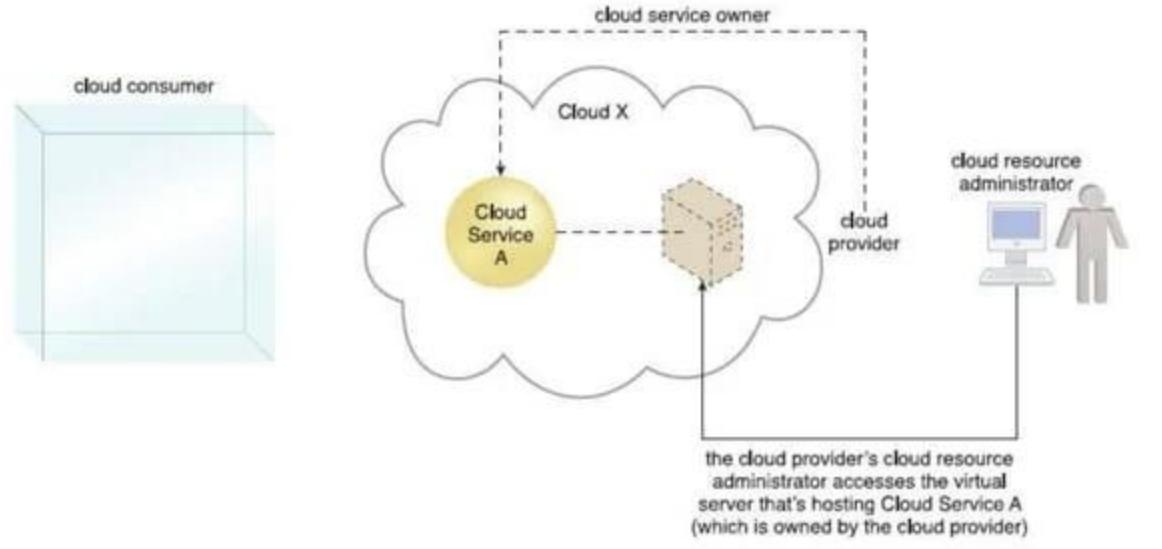


Figure 2.5

A cloud resource administrator can be with a cloud provider organization for which it can administer the cloud provider's internally and externally available IT resources.

Additional Roles

Cloud Auditor

- A **third-party** (often accredited) that conducts independent assessments of cloud environments assumes the role of the cloud auditor..
- The typical responsibilities associated with this role include **the evaluation of security controls, privacy impacts, and performance..**
- **The main purpose** of the cloud auditor role is to provide an unbiased assessment (and possible endorsement) of a cloud environment to help strengthen the trust relationship between cloud consumers and cloud providers.

Cloud Broker

- This role is assumed by a party that assumes the responsibility of managing and negotiating the usage of cloud services between cloud consumers and cloud providers.
- Mediation services provided by cloud brokers include service intermediation, aggregation, and arbitrage.

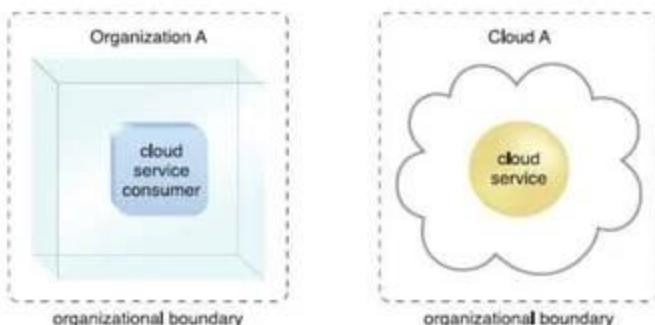
Cloud Carrier

- The party responsible for providing the wire-level connectivity between cloud consumers and cloud providers assumes the role of the cloud carrier. This role is often assumed by network and telecommunication providers.

Boundaries

Organizational Boundary

- An organizational boundary represents the **physical perimeter** that surrounds a set of IT resources that are owned and governed by an organization.
- The organizational boundary does not represent the boundary of an actual organization, only an organizational set of IT assets and IT resources. Similarly, clouds have an organizational boundary (Figure 2.6).



Trust Boundary

- A *trust boundary* is a **logical** perimeter that typically spans beyond physical boundaries to represent the extent to which IT resources are trusted (Figure 2.7).

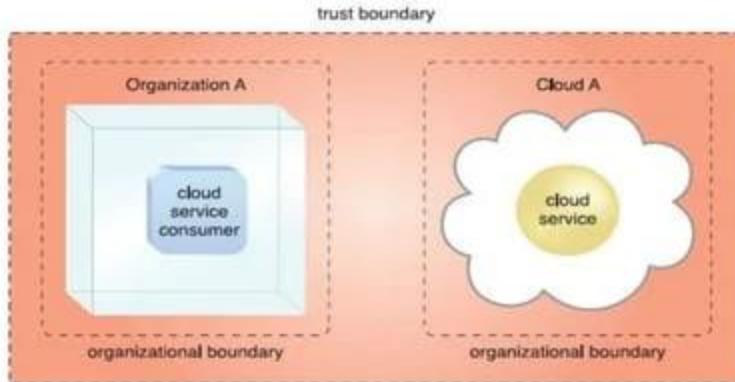


Figure 4.7

An extended trust boundary encompasses the organizational boundaries of the cloud provider and the cloud consumer.

3/17/2014

Cloud Characteristics

- On-demand usage
- Ubiquitous access
- Multitenancy (and resource pooling)
- Elasticity
- Measured usage
- Resiliency

Multitenancy

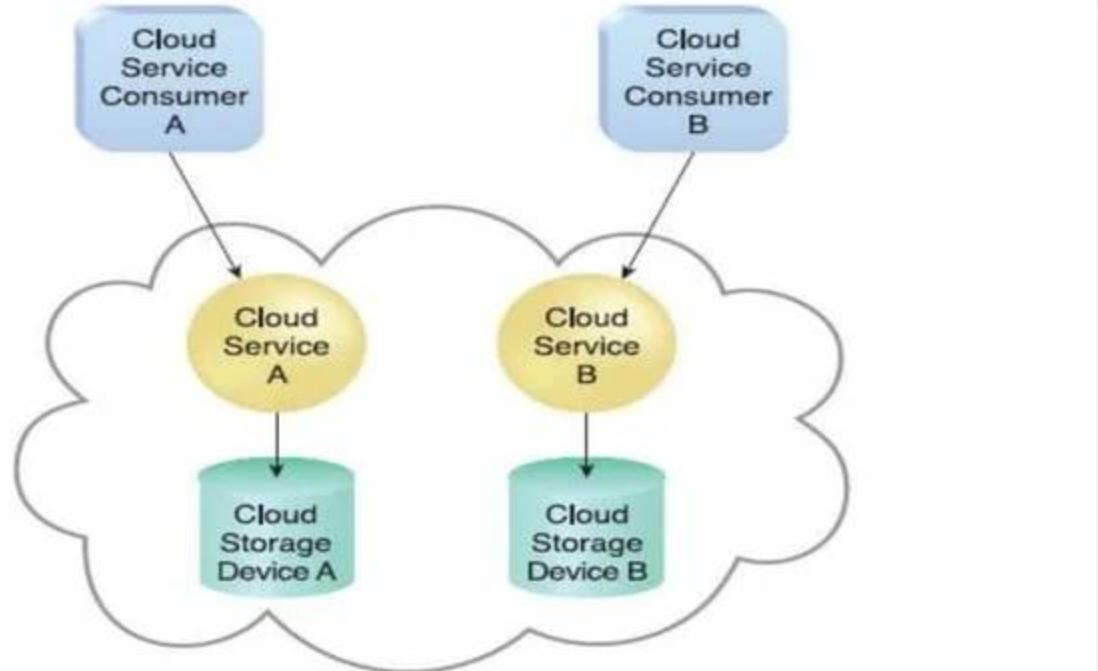


Figure 2.8

In a single-tenant environment, each cloud consumer has a separate IT resource instance.

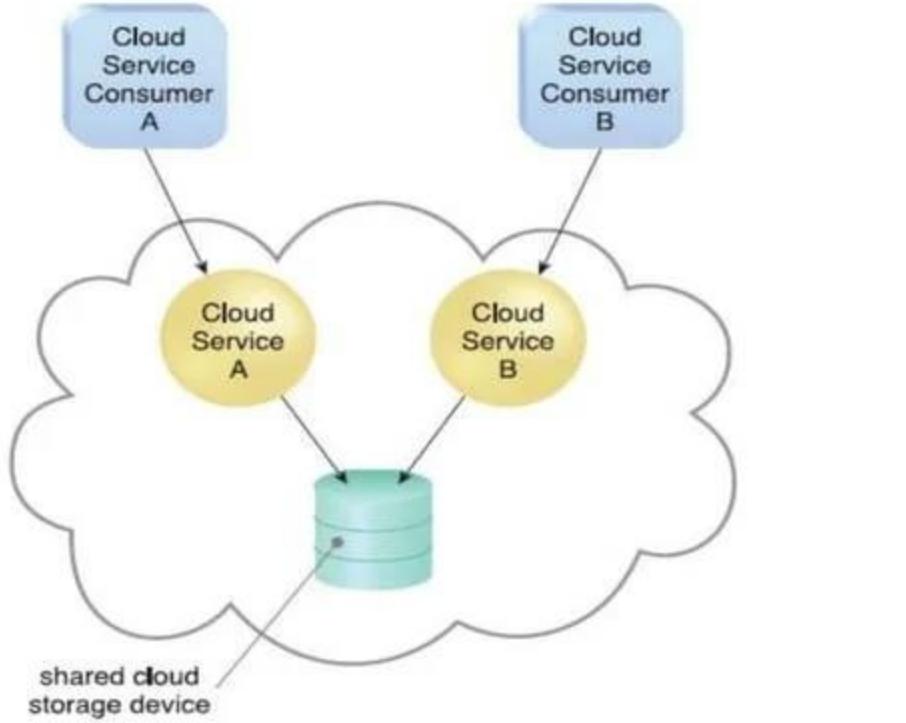


Figure 2.9

In a multitenant environment, a single instance of an IT resource, such as a cloud storage device, serves multiple consumers.

Resiliency

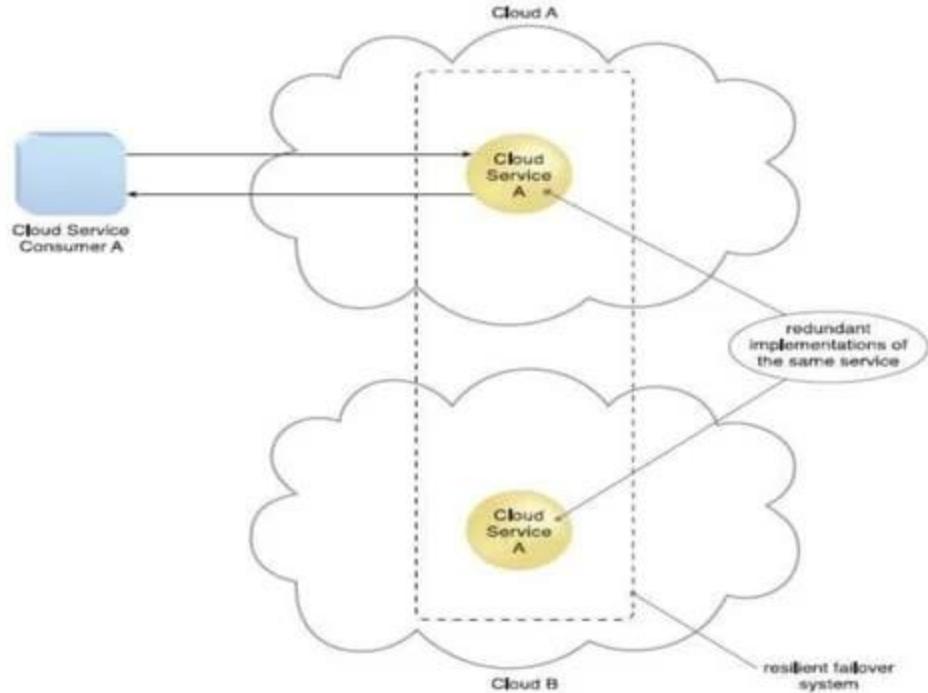


Figure 2.10

A resilient system in which Cloud B hosts a redundant implementation of Cloud Service A to provide failover in case Cloud Service A on Cloud A becomes unavailable.

3/17/2014

Cloud Delivery Models

Cloud Delivery Models

Infrastructure-as-a-Service (IaaS) → Amazon EC2
and S3, Terremark Enterprise Cloud, Windows Live
Skydrive and Rackspace Cloud

Platform-as-a-Service (PaaS) → Microsoft Azure,
Force and Google App engine

Software-as-a-Service (SaaS) → (Salesforce CRM,
Google Docs, etc).

Note

- Many specialized variations of the three base cloud delivery models have emerged, each comprised of a distinct combination of IT resources. Some examples include
 - Storage-as-a-Service
 - Database-as-a-Service
 - Security-as-a-Service
 - Communication-as-a-Service
 - Integration-as-a-Service
 - Testing-as-a-Service
 - Process-as-a-Service

Infrastructure-as-a-Service (IaaS)

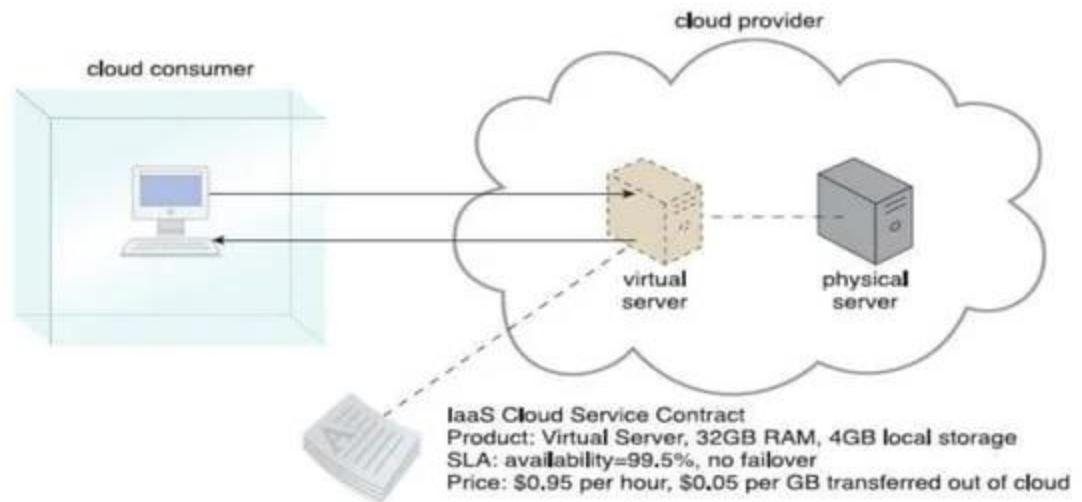


Figure 2.11

A cloud consumer is using a virtual server within an IaaS environment. Cloud consumers are provided with a range of contractual guarantees by the cloud provider, pertaining to characteristics such as capacity, performance, and availability.

Platform-as-a-Service (PaaS)

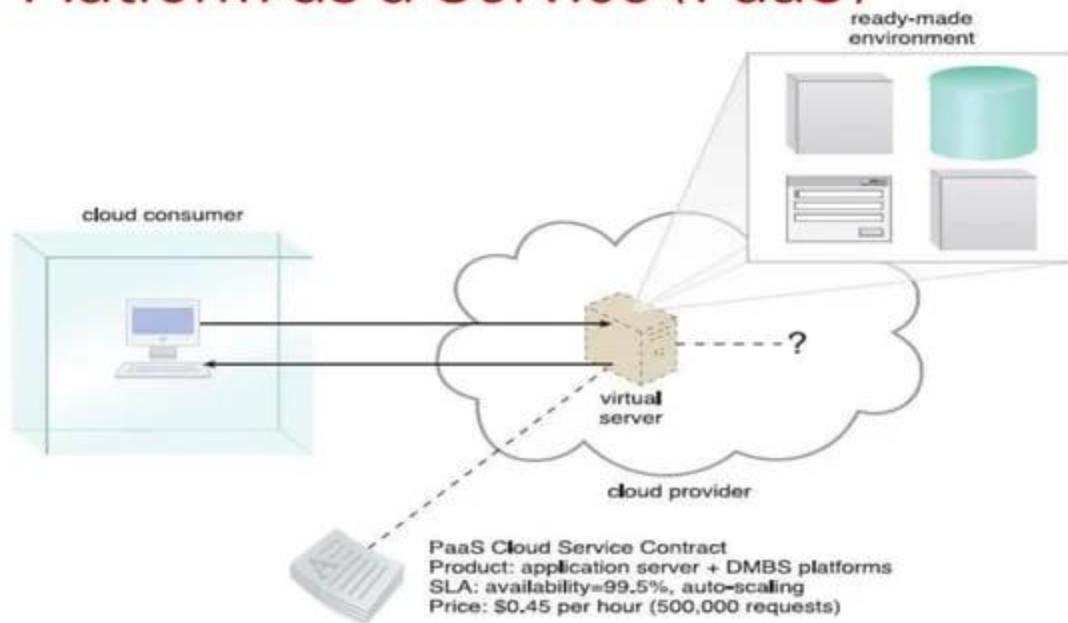


Figure 212

A cloud consumer is accessing a ready-made PaaS environment. The question mark indicates that the cloud consumer is intentionally shielded from the implementation details of the platform.

3/17/2014

Software-as-a-Service (SaaS)



Figure 2.13

The cloud service consumer is given access the cloud service contract, but not to any underlying IT resources or implementation details.

3/17/2014

Cloud Delivery Model	Typical Level of Control Granted to Cloud Consumer	Typical Functionality Made Available to Cloud Consumer
SaaS	usage and usage-related configuration	access to front-end user-interface
PaaS	limited administrative	moderate level of administrative control over IT resources relevant to cloud consumer's usage of platform
IaaS	full administrative	full access to virtualized infrastructure-related IT resources and, possibly, to underlying physical IT resources

Table 2.1

A comparison of typical cloud delivery model control levels.

Cloud Delivery Model	Common Cloud Consumer Activities	Common Cloud Provider Activities
SaaS	uses and configures cloud service	implements, manages, and maintains cloud service monitors usage by cloud consumers
PaaS	develops, tests, deploys, and manages cloud services and cloud-based solutions	pre-configures platform and provisions underlying infrastructure, middleware, and other needed IT resources, as necessary monitors usage by cloud consumers
IaaS	sets up and configures bare infrastructure, and installs, manages, and monitors any needed software	provisions and manages the physical processing, storage, networking, and hosting required monitors usage by cloud consumers

Table 2.2

Typical activities carried out by cloud consumers and cloud providers in relation to the cloud delivery models.

3/17/2014

Combining Cloud Delivery Models

IaaS + PaaS

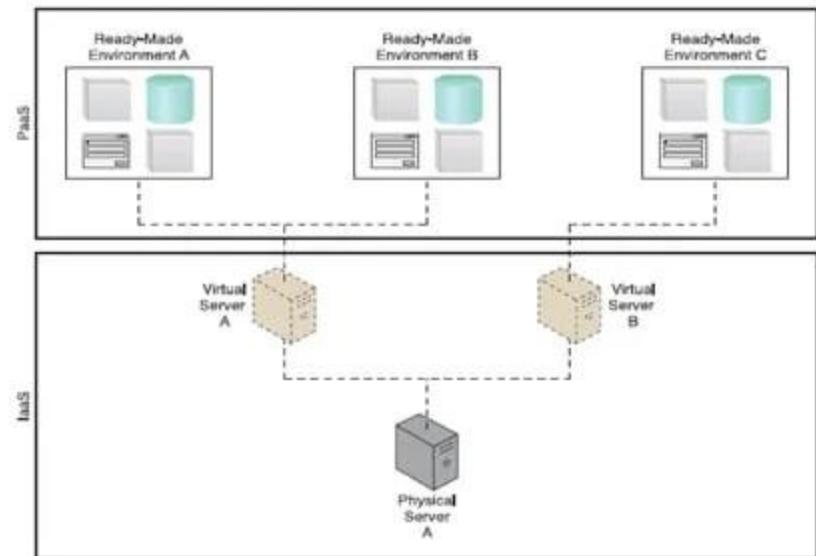
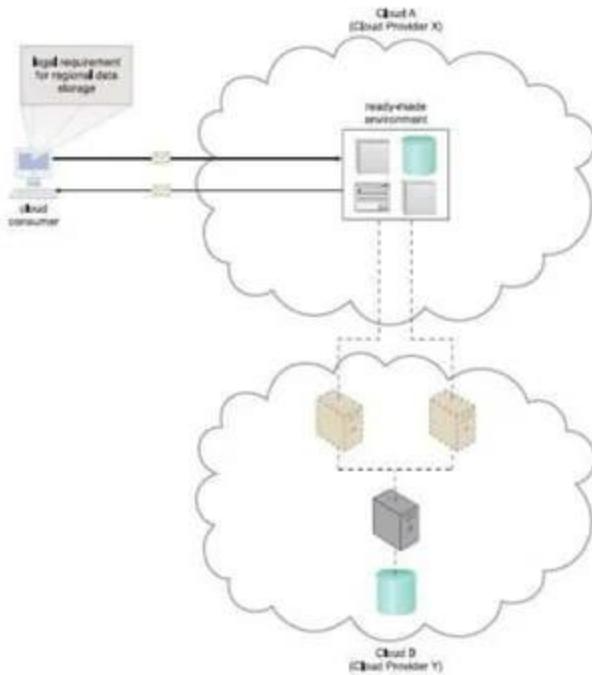


Figure 2.14

A PaaS environment based on the IT resources provided by an underlying IaaS environment.

Figure 2.15

An example of a contract between Cloud Providers X and Y, in which services offered by Cloud Provider X are physically hosted on virtual servers belonging to Cloud Provider Y. Sensitive data that is legally required to stay in a specific region is physically kept in Cloud B, which is physically located in that region.



IaaS + PaaS + SaaS

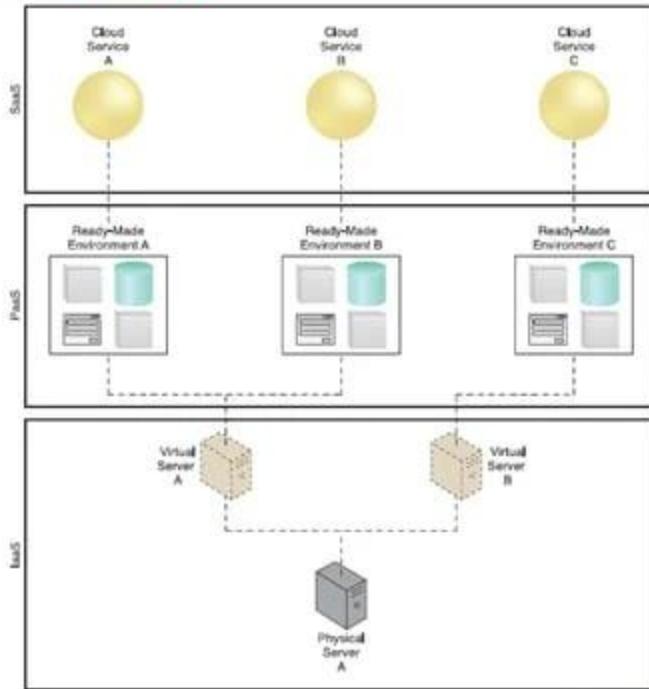


Figure 2.16

A simple layered view of an architecture comprised of IaaS and PaaS environments hosting three SaaS cloud service implementations.

Cloud Deployment Models

Cloud Deployment Models

- A cloud deployment model represents a specific type of cloud environment, primarily distinguished by ownership, size, and access.
- There are four common cloud deployment models:
 - Public cloud
 - Community cloud
 - Private cloud
 - Hybrid cloud

Public Clouds

- A *public cloud* is a publicly accessible cloud environment owned by a third-party cloud provider.
- The IT resources on public clouds are usually provisioned via the previously described cloud delivery models and are generally offered to cloud consumers at a cost or are commercialized via other avenues (such as advertisement).
- The cloud provider is responsible for the creation and on-going maintenance of the public cloud and its IT resources.

A partial view of the public cloud landscape, highlighting some of the primary vendors in the marketplace.

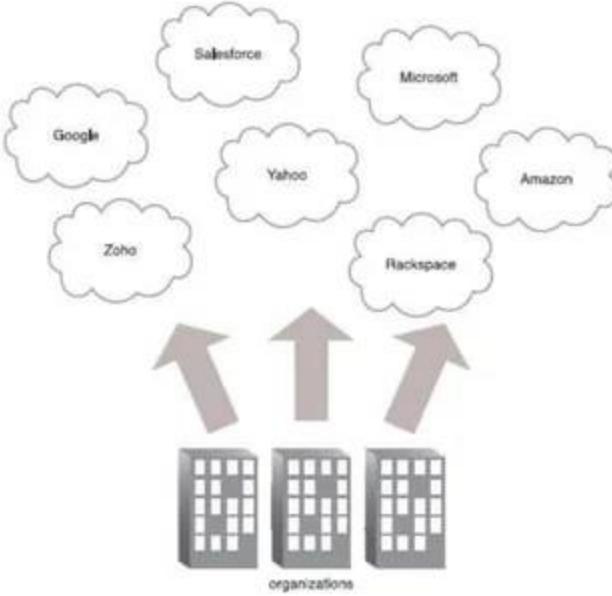


Figure 2.17

Organizations act as cloud consumers when accessing cloud services and IT resources made available by different cloud providers.

3/17/2014

Community Clouds

- A community cloud is similar to a public cloud except that its access is limited to a specific community of cloud consumers.
- The community cloud may be jointly owned by the community members or by a third-party cloud provider that provisions a public cloud with limited access.
- The member cloud consumers of the community typically share the responsibility for defining and evolving the community cloud

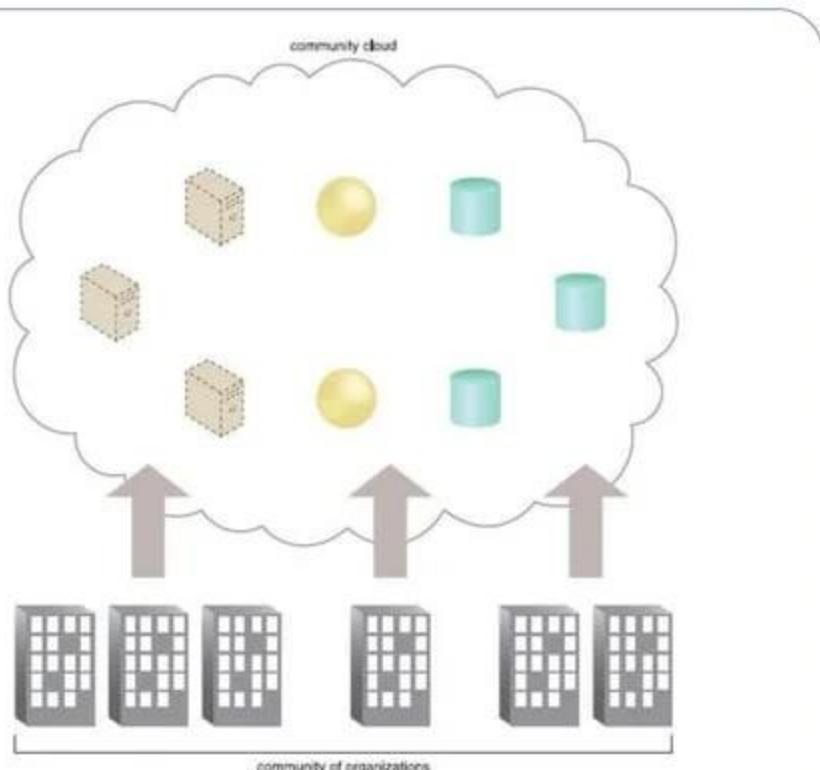


Figure 2.18

An example of a “community” of organizations accessing IT resources from a community cloud.

Private Clouds

- A private cloud is owned by a single organization.
- Private clouds enable an organization to use cloud computing technology as a means of centralizing access to IT resources by different parts, locations, or departments of the organization

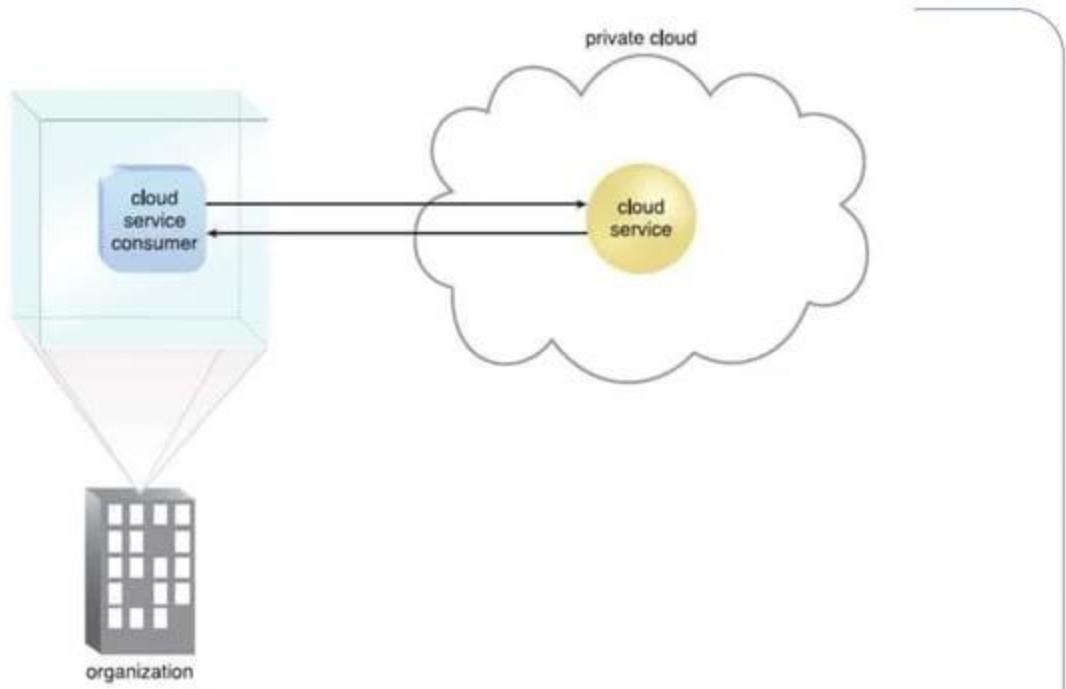


Figure 2.19

A cloud service consumer in the organization's on-premise environment accesses a cloud service hosted on the same organization's private cloud via a virtual private network.

Hybrid Clouds

- A hybrid cloud is a cloud environment comprised of two or more different cloud deployment models.
- For example, a cloud consumer may choose to deploy cloud services processing sensitive data to a private cloud and other, less sensitive cloud services to a public cloud. The result of this combination is a hybrid deployment model

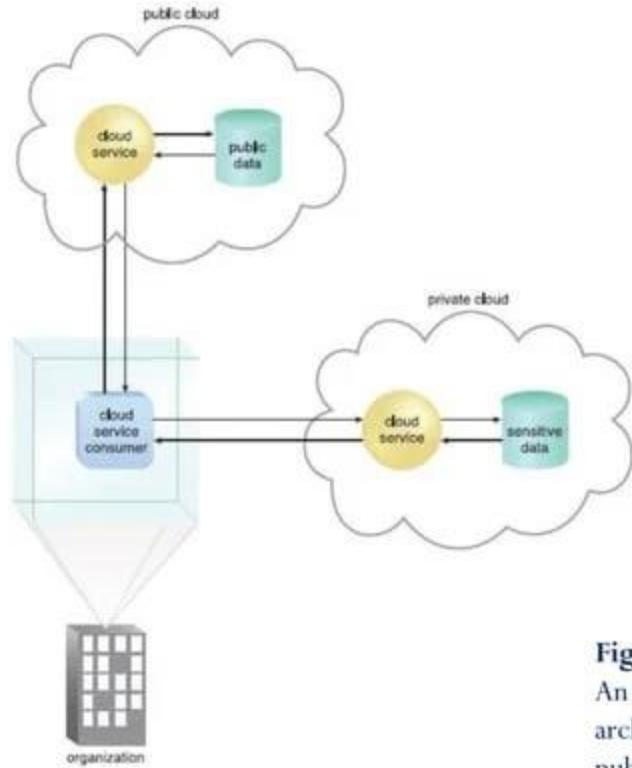


Figure 2.20

An organization using a hybrid cloud architecture that utilizes both a private and public cloud.

Other Cloud Deployment Models

- Additional variations of the four base cloud deployment models can exist. Examples include:
 - ***Virtual Private Cloud*** – Also known as a “dedicated cloud” or “hosted cloud,” this model results in a self-contained cloud environment hosted and managed by a public cloud provider, and made available to a cloud consumer.
 - ***Inter-Cloud*** – This model is based on an architecture comprised of two or more inter-connected clouds.

Cloud-Enabling Technology

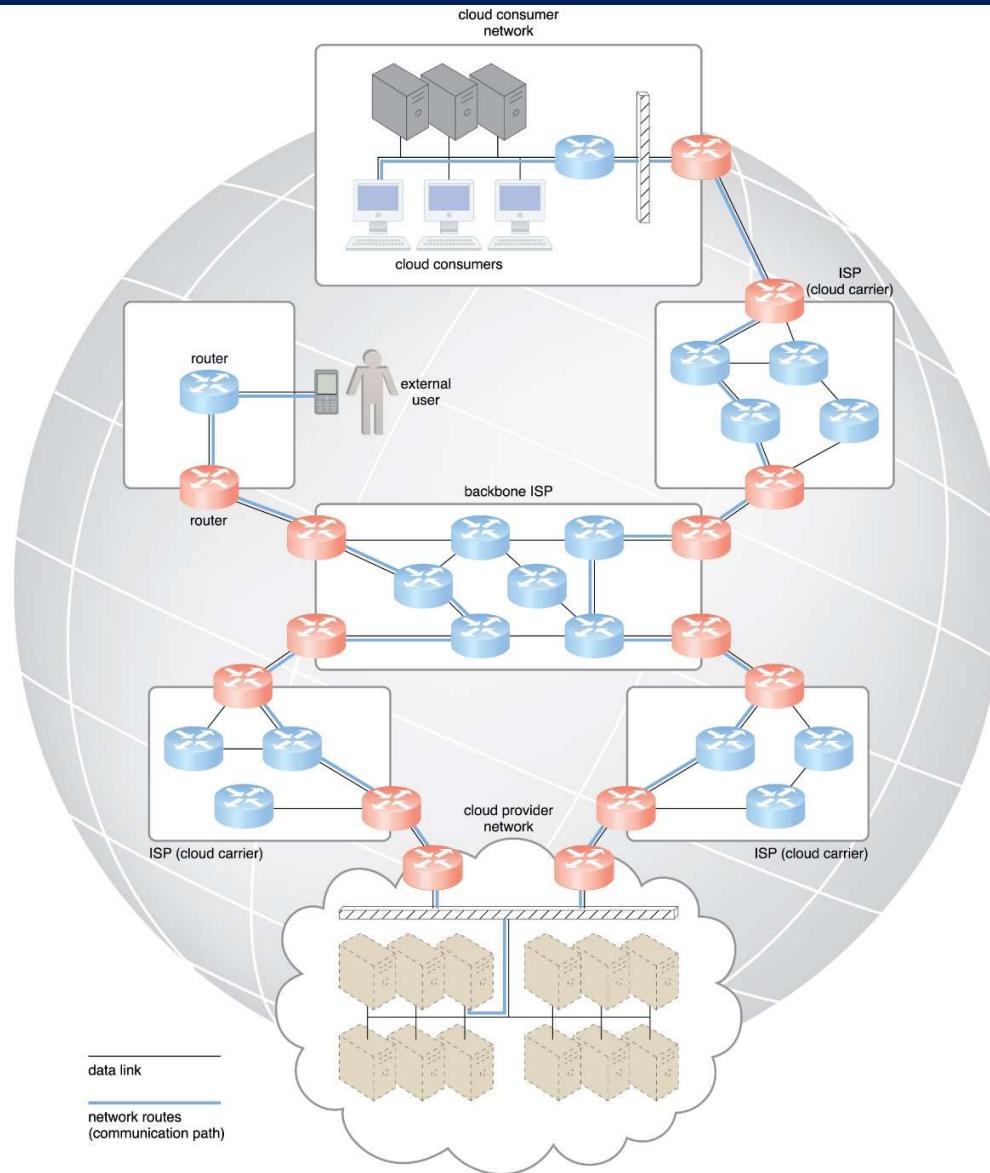
Enabling technologies

1. Broadband networks and internet architecture
2. Data center technology
3. Virtualization technology
4. Web technology
5. Multitenant technology

1. Broadband networks & Internet architecture

- All clouds must be connected to a network
- Internet's largest backbone networks, established and deployed by ISPs, are interconnected by core routers
 - ISP: internet service provider

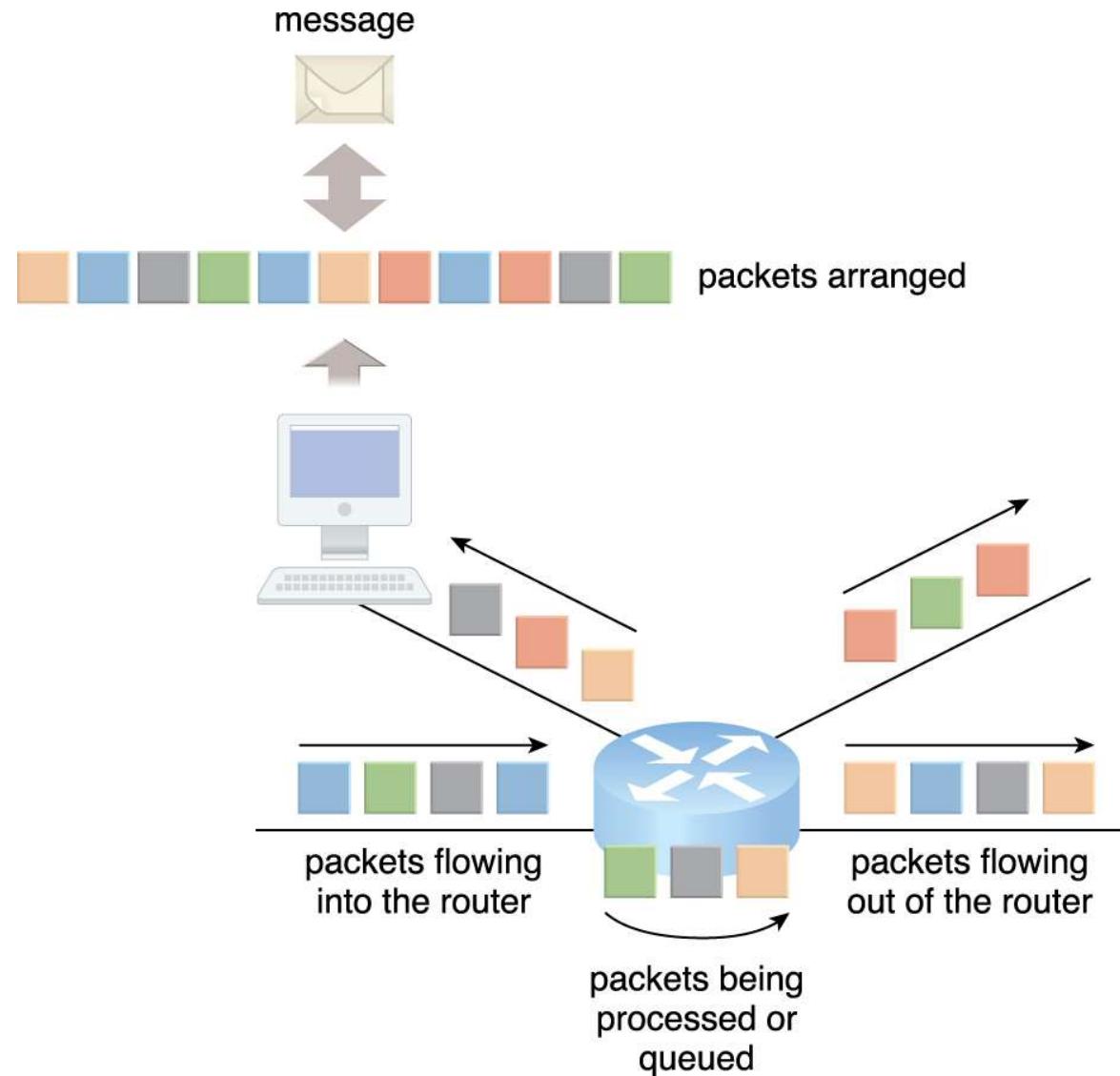
Internet connecting provider and consumer



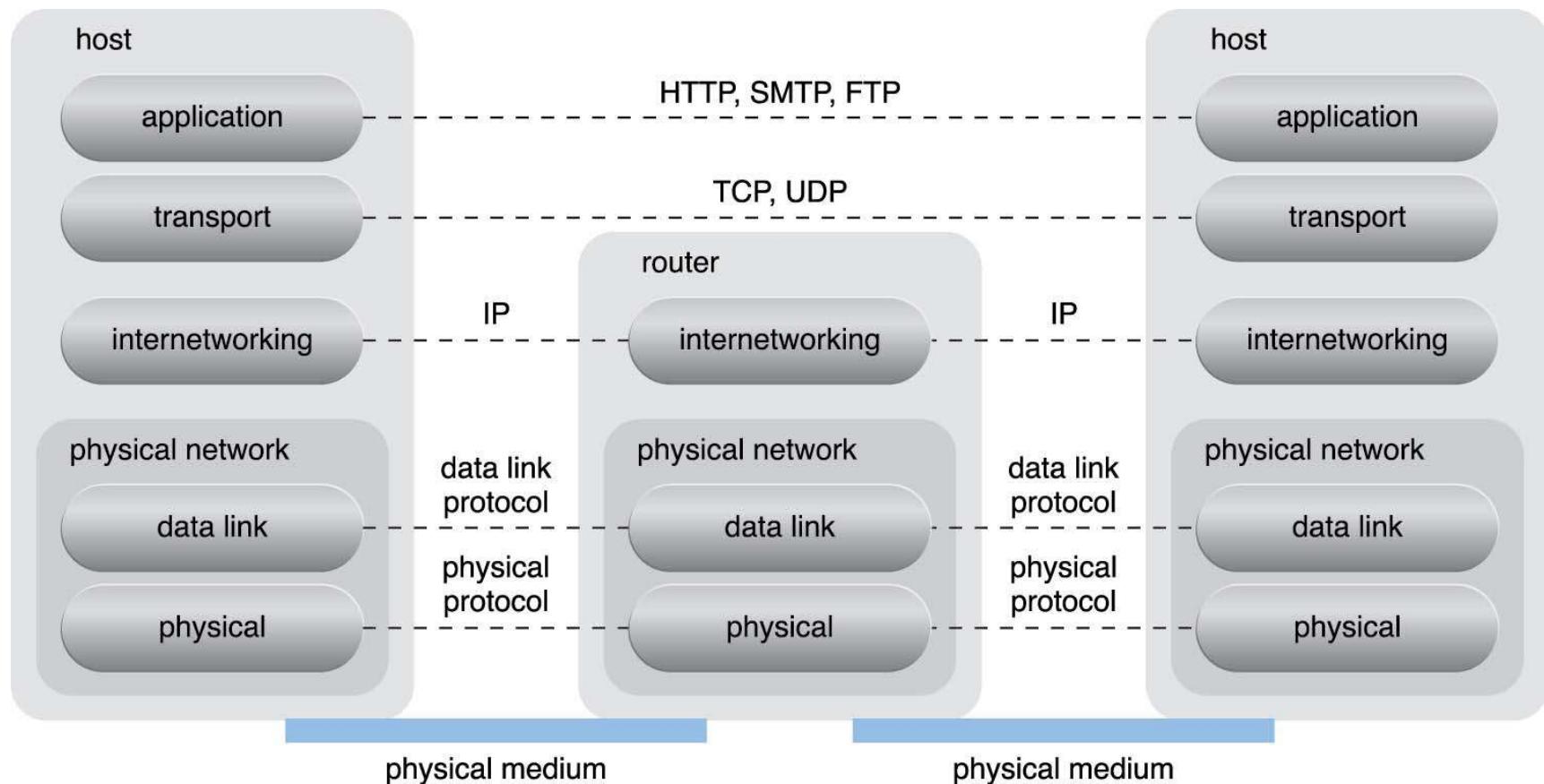
Two fundamental components

- Connectionless packet switching
 - End-to-end (sender-receiver pair) data flows are divided into packets of a limited size
 - Packets are processed through network switches and routers, then queued and forwarded from one intermediary node to the next
- Router-based interconnectivity
 - A router is a device that is connected to multiple networks through which it forwards packets
 - ❖ Each packet is individually processed
 - Use multiple alternative network routes

Packets travelling through Internet



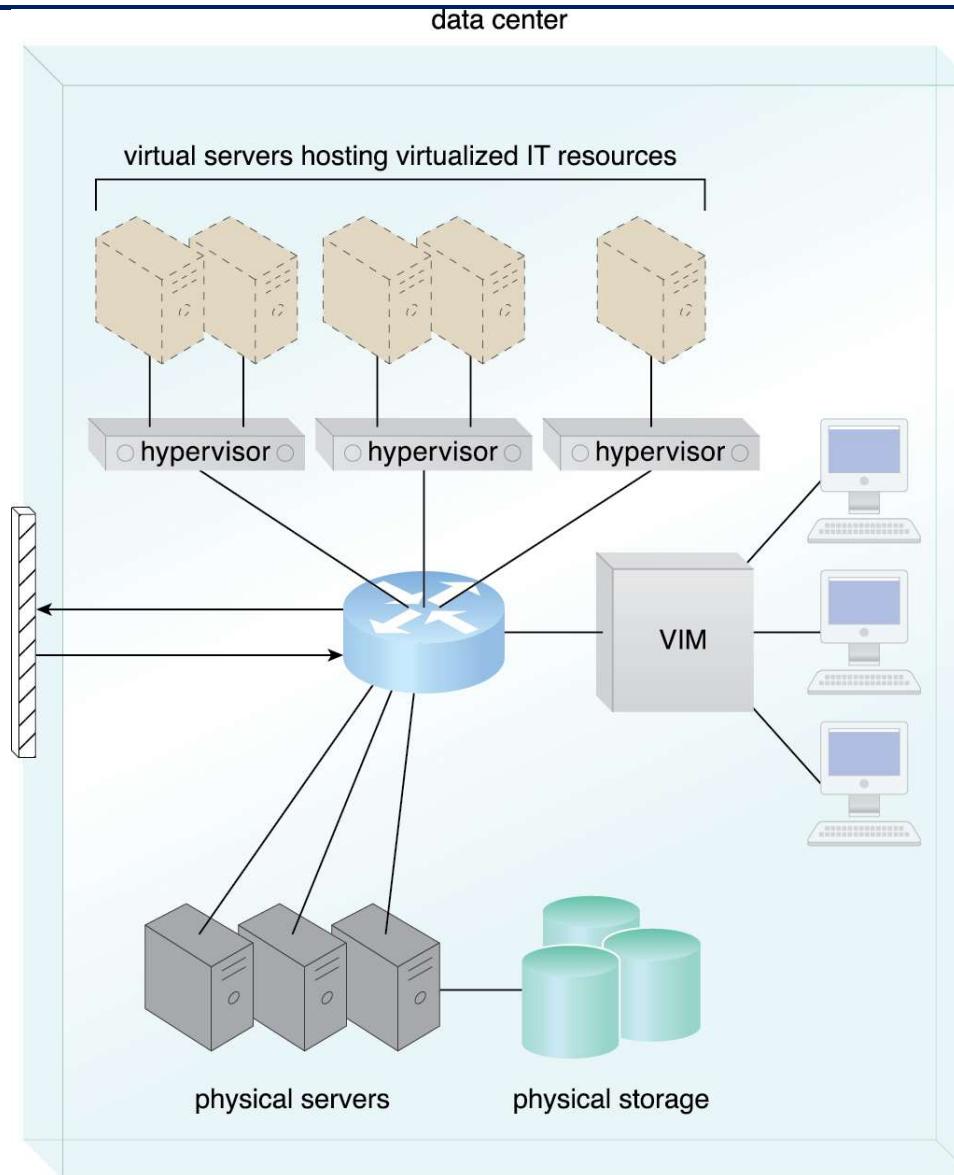
Internet reference model



2. Data Center Technology

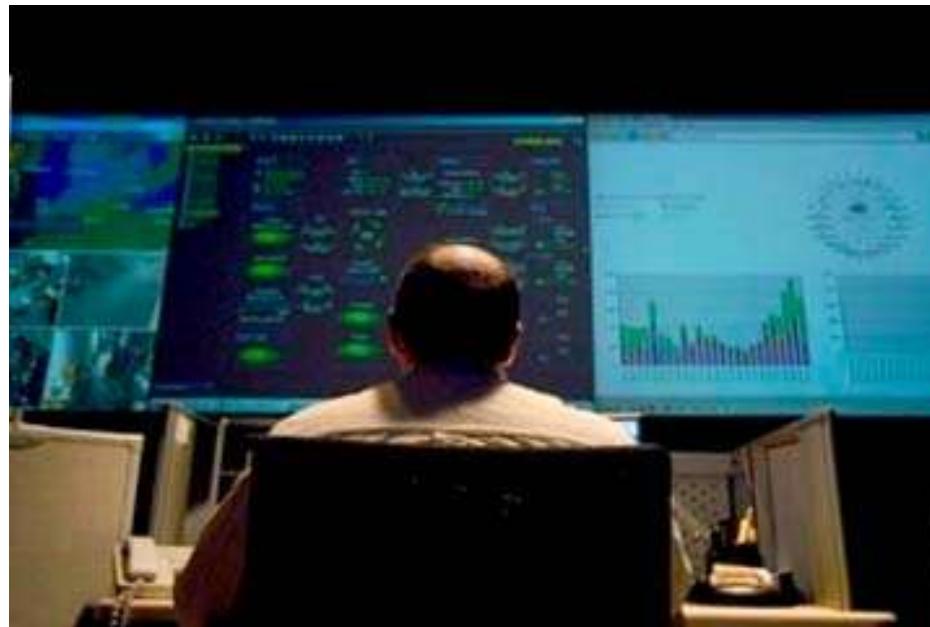
- A data center is a facility used to house computer systems and associated components, such as telecommunications and storage systems
 - Virtualization
 - Standardization and Modularity
 - Automation
 - Remote Operation and Management

Virtualization



Standardization and Modularity

- Data centers are built upon standardized commodity hardware and designed with modular architecture.



Supercomputer vs. data center

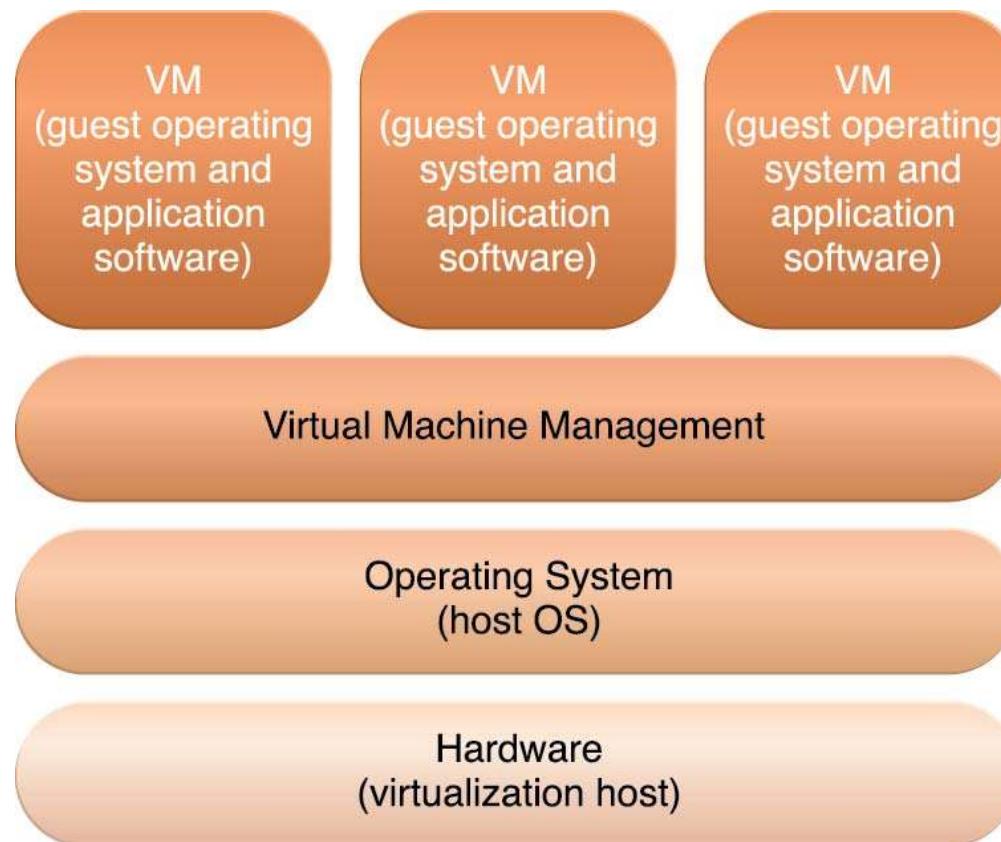
- Handouts

3. Virtualization technology

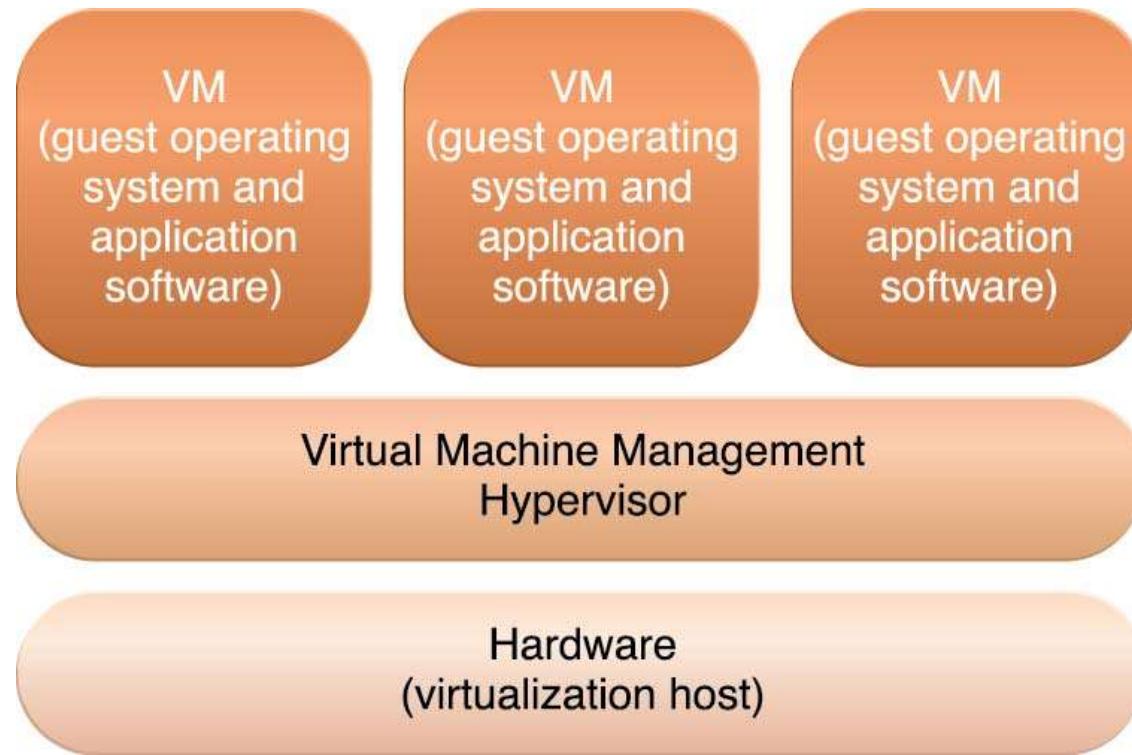
- Virtualization is a process of converting a physical IT resource into a virtual IT resource
 - Server
 - ❖ Virtual server ↔ virtual machine
 - Storage
 - Network
 - Power

Creating a new virtual server

- Allocation of physical IT resources
- Installation of an operating system, i.e., guest operating system



Hardware based virtualization



- Reduce the overhead
- May introduce compatibility issue

4. Web technology

- Cloud computing relies on internet.
- Web technology is generally used as both the implementation medium and the management interface for cloud services

Basic web technology

- Uniform resource locator (URL)
 - Commonly informally referred to as a **web address**
 - a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it
 - Example: <http://www.example.com/index.html>
- Hypertext transfer protocol (HTTP)
 - Primary communication protocol used to exchange content
- Markup languages (HTML, XML)
 - Express Web-centric data and metadata

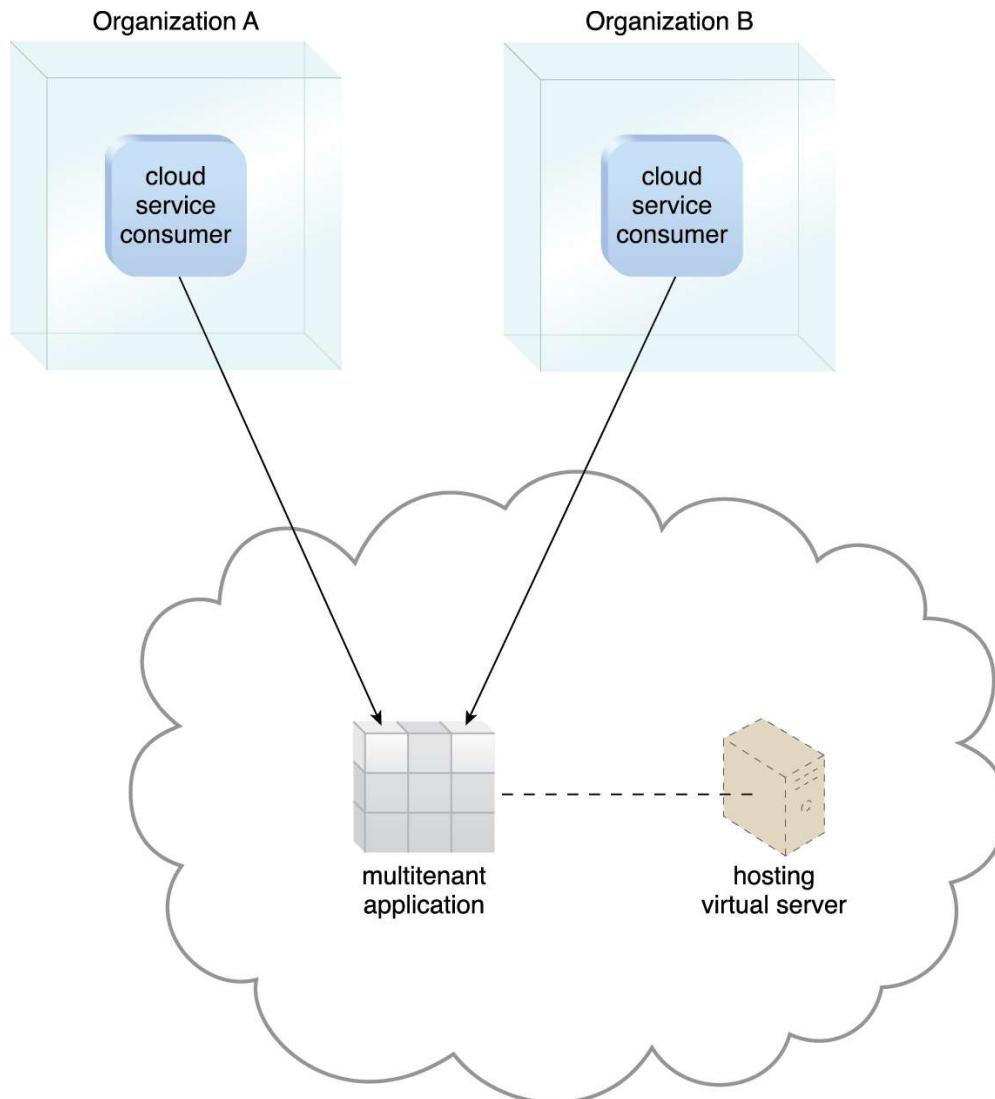
Web applications

- Applications running in a web browser
 - Rely on web browsers for the presentation of user-interfaces

5. Multitenant technology

- Enable multiple users (tenants) to access the same application simultaneously
- Multitenant applications ensure that tenants do not have access to data and configuration information that is not their own

A simple example



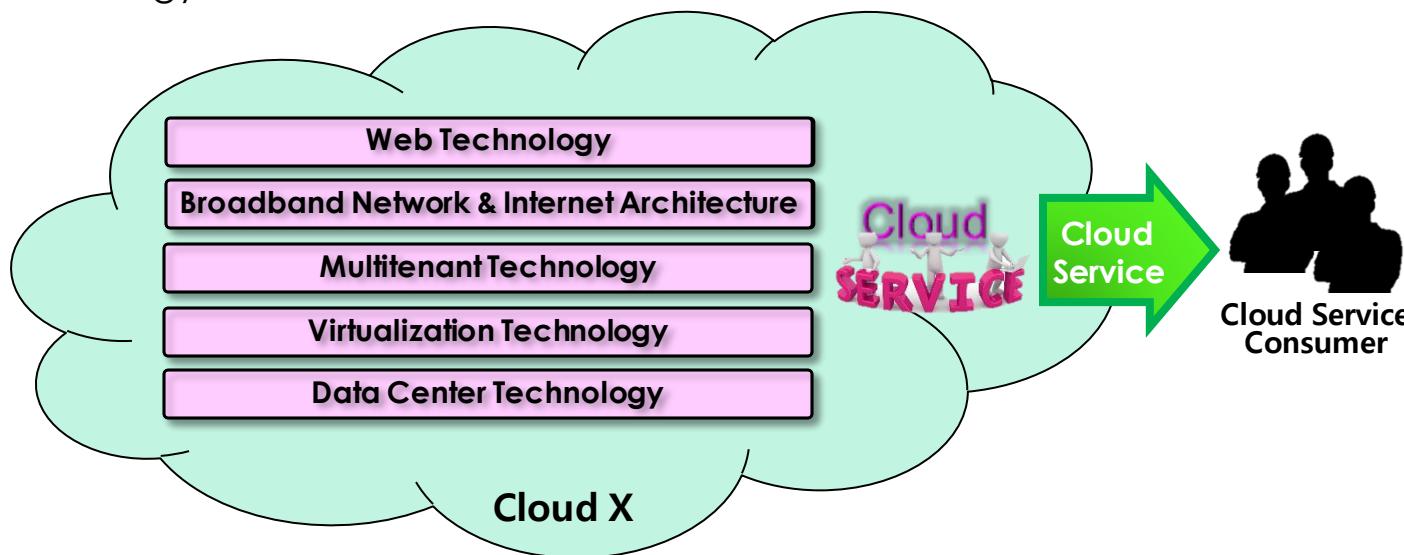
Cloud Enabling Technology

□ Integrated technology

- Not something entirely new – combined of & integrated from a number of existing technologies
- Integrating a number of existing core technologies into a single service – already matured and some of them more evolved on the way

□ Existing technologies enabled cloud computing include:

- Broadband networks & internet architecture
- Data center technology
- Virtualization technology
- Web technology
- Multitenant technology
- Service technology



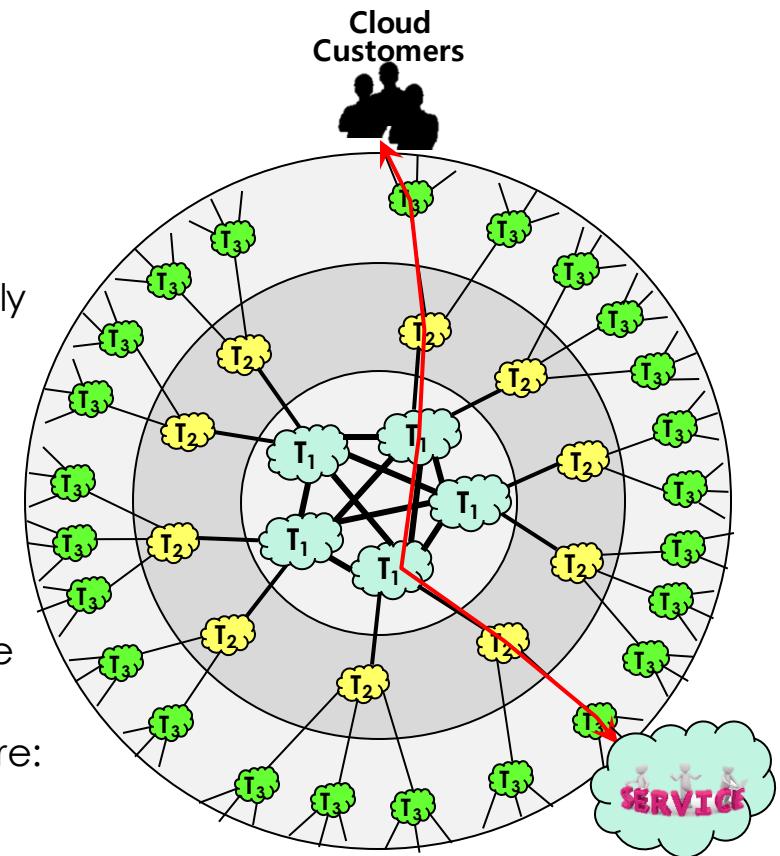
Broadband Networks & Internet Architecture – 1/11

□ Cloud service

- Requires remotely accessible service by definition – network connections are inevitable
- Implies inherent dependency on internet technology
- Enables remote provisioning of IT resources via ubiquitous network access (VPN or public network)
- Advances in accordance with the advancements of internet technology and QoS

□ Internet Service Providers (ISPs)

- An organization providing national-wide or world-wide internet access service
- Governed by Internet Corporations for Assigned Names and Numbers (ICANN)
- No comprehensive governing by ICANN – ISPs freely deploys, operates and manages their own networks based on basically decentralized provisioning and management models
- Fundamental governmental and regulatory laws applied within national borders
- Internet topology – a dynamic and complex aggregate of ISPs highly interconnected via its core protocols
- Worldwide connectivity via a hierarchical topology composed of Tier 1 (large-scale international ISPs), Tier 2 (large regional ISPs) and Tier 3 (local ISPs)
- Two fundamental components of internetworking architecture: connectionless packet switching v.s. router-based interconnectivity



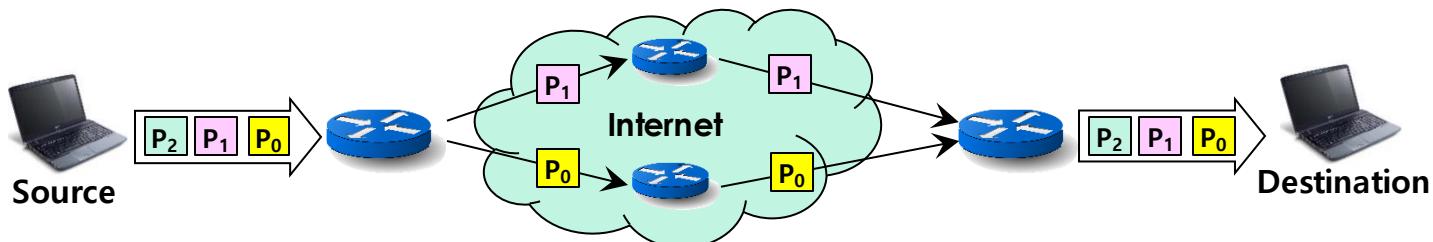
Broadband Networks & Internet Architecture – 2/11

□ Connectionless packet switching (datagram network)

- End-to-end (sender-receiver pair) data message divided into packets of limited size
- Each packet processed through network switches and routers, queued and forwarded from one intermediary node to the next
- Necessary transfer information carried by each packet in accordance with corresponding protocols such as Internet Protocol (**IP**) address or Media Access Control (**MAC**) address

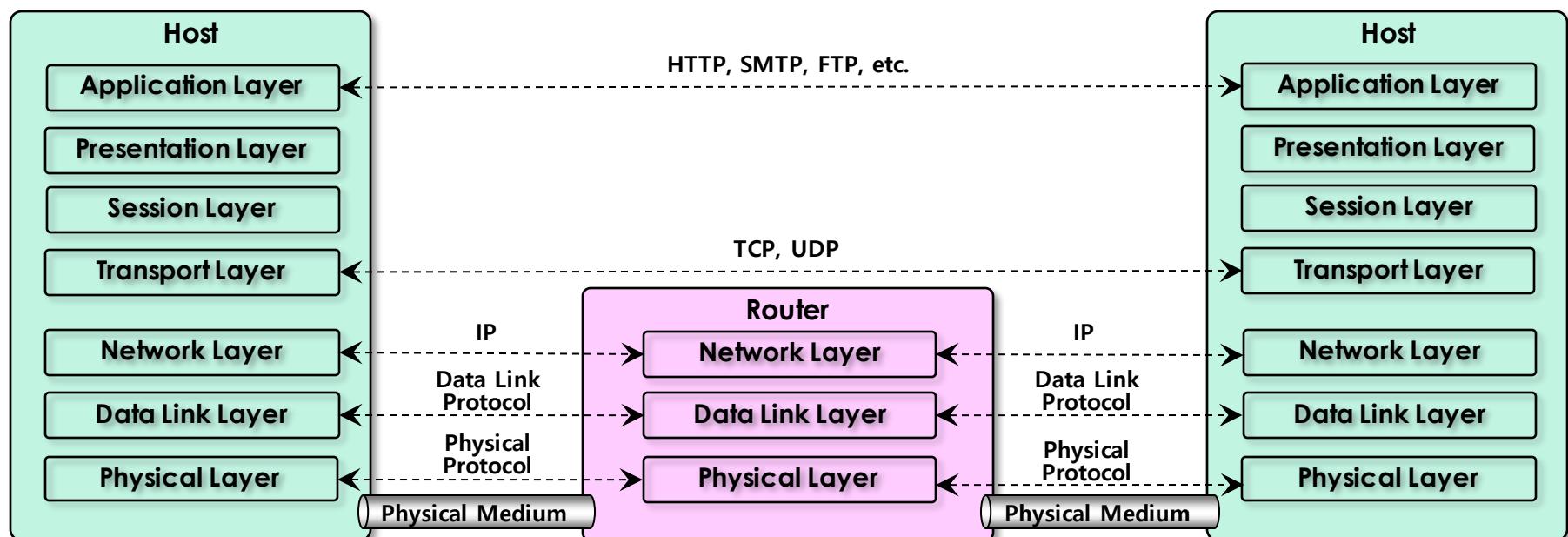
□ Router-based interconnectivity

- A router – a device connected to multiple networks through which it forwards packets
- Each packet transferred (stored & forwarded at each router) to destination individually via possibly different routes from each other ⇒ routing information (IP addresses of the source & the destination, sequential number, etc.) included in each packet
- Packets reassembled into a message on the destination node (at the network layer)
- Each router responsible for finding the most efficient hop for packet delivery at runtime
- Possibly multiple ISP networks between a cloud customer and its cloud provider
- 7 abstraction layer model defined in **OSI** (Open Systems Interconnection) project by **ISO/IEC 7498-1**
 - physical layer (1), data link layer (2), network layer (3), transport layer (4), session layer (5), presentation layer (6), application layer (7)



Broadband Networks & Internet Architecture – 3/11

Layer	Protocol Data Unit (PDU)	Function
7. Application	Data	High-level APIs, including resource sharing, remote file access – HTTP, FTP etc.
6. Presentation		Translation of data between a networking service and an application; including character encoding, data compression and encryption/decryption
5. Session		Managing communication sessions, i.e. continuous exchange of information in the form of multiple back-and-forth transmissions between two nodes
4. Transport	Segment (TCP) / Datagram (UDP)	Reliable transmission of data segments between points on a network, including segmentation, acknowledgement and multiplexing
3. Network	Packet	Structuring and managing a multi-node network, including addressing, routing and traffic control
2. Data Link	Frame	Reliable transmission of data frames between two nodes connected by a physical layer
1. Physical	Bit	Transmission and reception of raw bit streams over a physical medium



Broadband Networks & Internet Architecture – 4/11

□ OSI 7 layer model

- Physical layer (Layer 1)
 - Defines for the electrical and physical specifications of the data connection
 - Defines the relationship between a device and a physical transmission medium (e.g., a copper or fiber optical cable, radio frequency) including the layout of pins, voltages, line impedance, cable specifications, signal timing and similar characteristics for connected devices and frequency (5 GHz or 2.4 GHz etc.) for wireless devices
 - Responsible for transmission and reception of unstructured raw data in a physical medium
 - Defines the network topology as bus, mesh, or ring being some of the most common
 - Includes **Parallel SCSI**, **Ethernet** & other local-area networks such as **token ring**, **FDDI**, **ITU-T G.hn**, and **IEEE 802.11** (Wi-Fi)
 - Defines personal area networks such as **Bluetooth** and **IEEE 802.15.4** as well
 - Defines low-level networking equipment, such as network adapters, repeaters, network hubs, modems, and fiber media converters
 - Protocol independent - never concerned with protocols or other such higher-layer items
- Data link layer (layer 2)
 - Provides node-to-node data transfer – a link between two directly connected nodes
 - Detects and possibly corrects errors that may occur in the physical layer
 - Defines the protocol to establish and terminate a connection between two physically connected devices as well as the protocol for flow control between them
 - High-speed local area networking over existing wires (power lines, phone lines and coaxial cables) defined by The **ITU-T G.hn** standard in this data link layer, providing both error correction and flow control by means of a selective-repeat sliding-window protocol

Broadband Networks & Internet Architecture – 5/11

- Divided into two sublayers by IEEE 802:
 - **Media Access Control (MAC)** layer - responsible for controlling how devices in a network gain access to medium and permission to transmit it
 - **Logical Link Control (LLC)** layer - responsible for identifying Network layer protocols and then encapsulating them and controls error checking and frame synchronization
- Includes the MAC and LLC layers of IEEE 802 networks such as **802.3 Ethernet**, **802.11 Wi-Fi**, and **802.15.4 ZigBee**
- Defines the Point-to-Point Protocol (**PPP**) that can operate over several different physical layers, such as synchronous and asynchronous serial lines
- Network layer (Layer 3)
 - Provides the functional and procedural means of transferring variable length data sequences (called **datagrams**) from one node to another connected to the same "network"
 - A network – a communication medium to which many nodes with **addresses** (e.g., IP) can be connected, allowing each member node to transfer a message to any other member nodes via **address resolution** or **routing** through intermediate nodes
 - Large messages divided into several fragments before sending and reassembled again upon receiving at the network layer
 - May report delivery errors – message delivery at the network layer is not necessarily guaranteed to be reliable; a network layer protocol may provide reliable message delivery, but it need not do so.
 - Defines a number of layer-management protocols (a function defined in the management annex, ISO 7498/4) including routing protocols, multicast group management, network-layer information and error, and network-layer address assignment – determined by the payload that makes these belong to the network layer, not the protocol that carries them

Broadband Networks & Internet Architecture – 6/11

- Transport layer (Layer 4)
 - Provides the functional and procedural means of transferring variable-length data sequences from a source to a destination host via one or more networks, while maintaining the quality of service functions - Transmission Control Protocol (TCP) usually built on top of the Internet Protocol (IP) is an example of a transport-layer protocol in the standard Internet stack
 - Controls the reliability of a given link through flow control, segmentation/desegmentation, and error control
 - Some protocols are state- and connection-oriented implying that the transport layer can keep track of the segments and re-transmit those that fail.
 - Also provides the acknowledgement of the successful data transmission and sends the next data if no errors occurred
 - Creates packets out of the message received from the application layer. Packetizing is a process of dividing the long message into smaller messages.
 - Five classes of connection-mode transport protocols defined by OSI, ranging from class 0 (which is also known as TP0 and provides the fewest features) to class 4 (TP4, designed for less reliable networks, similar to the Internet)
 - Class 0: contains no error recovery and designed for use on network layers that provide error-free connections
 - Class 4: closest to TCP, although TCP contains functions, such as the graceful close, which OSI assigns to the session layer
 - All OSI TP connection-mode protocol classes provide expedited data and preservation of record boundaries.
 - Similar to a post office which deals with the dispatch and classification of mail and parcels sent

Broadband Networks & Internet Architecture – 7/11

- Packets are then encapsulated into higher level protocols, such as cryptographic presentation services that can be read by the addressee only.
- Non-IP tunneling protocols operating at the transport layer: IBM's **SNA**, Novell's **IPX** over an IP network, or end-to-end encryption with **IPsec**
- While Generic Routing Encapsulation (GRE) might seem to be a network-layer protocol, if the encapsulation of the payload takes place only at endpoint, GRE becomes closer to a transport protocol that uses IP headers but contains complete frames or packets to deliver to an endpoint.
- L2TP carries PPP frames inside transport packet.
- Although not developed under the OSI Reference Model and not strictly conforming to the OSI definition of the transport layer, the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP) of the Internet Protocol Suite are commonly categorized as layer-4 protocols within OSI.

Feature Name	TP0	TP1	TP2	TP3	TP4
Connection-oriented Network	Yes	Yes	Yes	Yes	Yes
Connectionless Network	No	No	No	No	Yes
Concatenation and Separation	No	Yes	Yes	Yes	Yes
Segmentation and Reassembly	Yes	Yes	Yes	Yes	Yes
Error Recovery	No	Yes	Yes	Yes	Yes
Reinitiate Connection*	No	Yes	No	Yes	No
Multiplexing / Demultiplexing over Single Virtual Circuit	No	No	Yes	Yes	Yes
Explicit Flow Control	No	No	Yes	Yes	Yes
Retransmission on Timeout	No	No	No	No	Yes
Reliable Transport Service	No	Yes	No	Yes	Yes

* If an excessive number of PDUs are unacknowledged

Broadband Networks & Internet Architecture – 8/11

- Session layer (Layer 5)
 - Controls the dialogues (connections) between computers - establishing, managing and terminating the connections between the local and remote application
 - Provides for full-duplex, half-duplex, or simplex operation, and establishes checkpointing, adjournment, termination, and restart procedures
 - Also provides graceful close of sessions which is a property of the Transmission Control Protocol and session checkpointing and recovery which is not usually used in the Internet Protocol Suite
 - Commonly implemented explicitly in application environments that use remote procedure calls
- Presentation layer (Layer 6)
 - Establishes context between application-layer entities, in which the application-layer entities may use different syntax and semantics if the presentation service provides a mapping between them
 - Encapsulates presentation service data units into session protocol data units that are then passed down the protocol stack If a mapping is available
 - Provides independence from data representation (e.g., encryption) by translating between application and network formats
 - Transforms data into the form that the application accepts - formatting and encrypting data to be sent across a network (sometimes called the syntax layer)
 - The original presentation structure used the Basic Encoding Rules of Abstract Syntax Notation One (ASN.1) with capabilities such as converting an EBCDIC-coded text file to an ASCII-coded file, or serialization of objects and other data structures from and to XML.

Broadband Networks & Internet Architecture – 9/11

- Application layer (Layer 7)
 - The OSI layer closest to the end user which means both the OSI application layer and the user interact directly with the software application
 - Interacts with software applications (outside the scope of the OSI model) that implement a communicating component
 - Includes functions such as identifying communication partners, determining resource availability, and synchronizing communication
 - Determines the identity and availability of communication partners for an application with data to transmit when identifying communication partners
 - Must decide whether sufficient network resources for the requested communication are available when determining resource availability

Broadband Networks & Internet Architecture – 10/11

❑ Technical and business considerations

▪ Connectivity issues

➤ Traditional deployment model

- Via the corporate network (VPN) which provide uninterrupted Internet connectivity
- Completely controlled by the organizations with their own safeguard based on firewalls and various monitoring tools
- Each organization responsible for deploying, operating and managing their IT resources and Internet connectivity

➤ Cloud deployment model

- Continuous access to centralized servers and applications granted to end-user devices as long as they are connected to the network through the Internet in the cloud
- Centralized IT resources accessible using the same network protocols regardless of whether users reside inside or outside of a corporate network
- Cloud IT resources configured by cloud providers to be accessible for both external and internal users through an Internet connection and for cloud consumers to provide Internet-based services to external users

On-premise IT Resources	Cloud-based IT Resources
Internal end-user devices access corporate IT services through the corporate network.	Internal end-user devices access corporate IT services through an Internet connection.
Internal users access corporate IT services through the corporate Internet connection while roaming in external networks.	Internal users access corporate IT services while roaming in external networks through the cloud provider's Internet connection.
External users access corporate IT services through the corporate Internet connection.	External users access corporate IT services through the cloud provider's Internet connection.

Broadband Networks & Internet Architecture – 11/11

- Network bandwidth and latency issues
 - Network QoS: bandwidth, latency, jitter
 - **Bandwidth** – how much data can be transferred within a unit time
 - End-to-end bandwidth determined by the transmission capacity of the shared data links that connect intermediary nodes
 - Attempt to improve end-to-end bandwidth by ISPs with technologies such as broadband network technology & web acceleration technologies – dynamic caching, compression, pre-fetching, etc.
 - Critical for applications requiring substantial amount of data transfer
 - **Latency** – how fast a request can be satisfied (time for a packet to travel from one node to another)
 - The longer a packet travels the larger the latency is – web caching technology can apply
 - The more network traffic is the larger the latency is – more queuing delay at each hop
 - Critical for applications with a business requirement of fast response time
 - **Jitter** – how consistent the given latency is
 - A gap between the smallest latency and the largest latency
 - Response time less than a millisecond in general, but frequently more than several seconds
 - Internet-wide QoS control required to guarantee small jitter
 - QoS of the underlying network inherited to QoS of the given cloud service
- Cloud carrier and cloud provider selection
 - Involves multiple cloud carriers to achieve the necessary level of connectivity and reliability for the given cloud applications resulting in additional costs
 - QoS determined by multiple ISPs involved & required collaboration of the cloud carriers
 - Wise to adopt more relaxed latency and bandwidth requirements

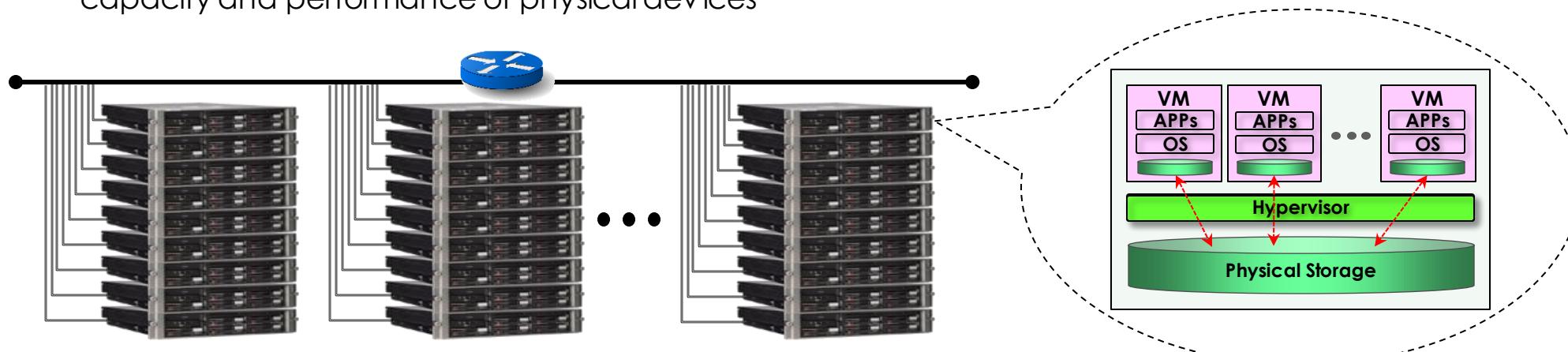
Data Center Technology – 1/6

□ Data center

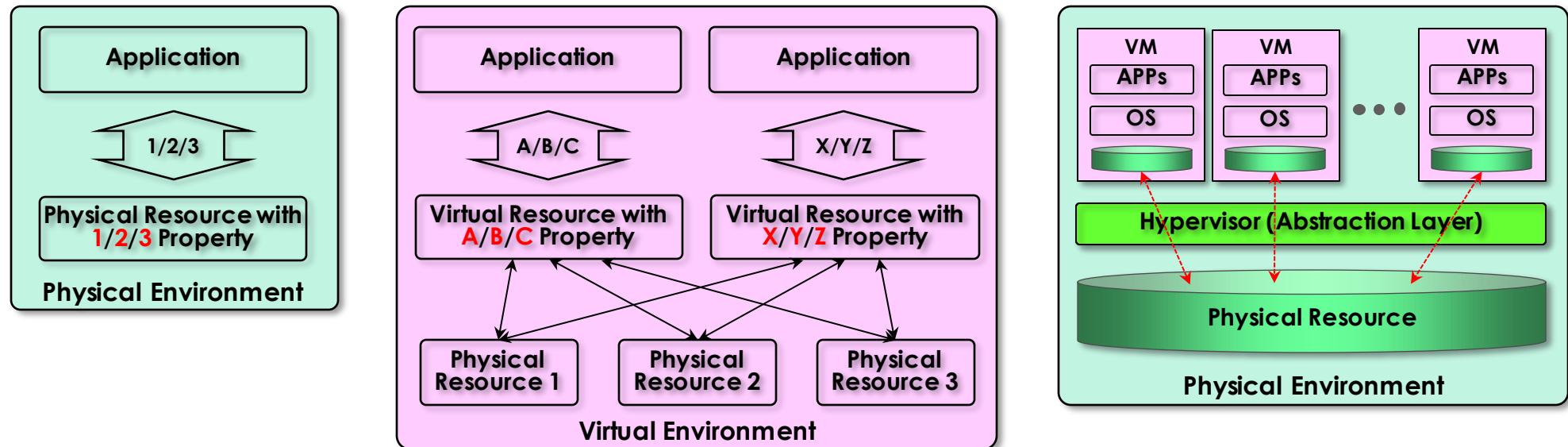
- Grouping IT resources in close proximity with one another (rather than having them geographically dispersed) for power sharing, higher efficiency in shared IT resource usage and improved accessibility for IT personnel – reason for popularizing data center concept
- Characterized for centralized IT resources such as servers, storages, databases, networking & telecommunication devices and software solutions via applying a number of technologies

□ Standardization and modularity

- Built upon standardized commodity hardware and designed with modular architecture
- Aggregating multiple identical building blocks of facility infrastructure and equipment to support scalability, growth and speedy hardware replacements
- Reduces investment and operational costs as they enable economies of scale for the procurement, acquisition, deployment, operation and maintenance processes
- IT resource consolidation favored by common virtualization strategies and the constantly improving capacity and performance of physical devices



Data Center Technology – 2/6



❑ Virtualization

- Virtualization: an abstraction layer with mapping or redirection capability
- Physical IT resources: the facility infrastructure that houses computing/networking systems and equipment, together with hardware systems and their operating systems
- Virtual IT resources: comprised of operational and management tools that are often based on virtualization platforms that abstract the physical computing and networking IT resources as virtualized components that are easier to allocate, operate, release, monitor and control (more details later)

❑ Automation

- Reduces operational costs and the error rate in data center via automated management without human supervision – provisioning, configuration, patching and monitoring
- Enables self-configuration and self-recovery – basis of automatic computing technology

Data Center Technology – 3/6

Remote operation and management

- Most operational and administrative tasks of IT resources in data center can be commanded through the network's remote (within data center boundary in general) consoles and management systems.
- Most operational and administrative tasks carried out from the control room in data center except for those requiring physical operations such as hardware jobs or cabling
- Remote operation from outside of data center boundary strictly prohibited in general.

High availability

- All resources in data center are subject to fail anytime based on current hardware and software technologies.
- Most resource failures affect service continuity and underlying business as well.
- In general, data center provides fail-safe technologies mainly based on redundancy in every possible layer – fault-tolerant or fault-resilient technologies on top of redundant resources: power supply, cabling, networking, servers, storages and software licenses.
- Fault-avoidance technologies: load balancing, scaling-up/down, etc.

Security-aware design, operation and management

- The level of security determines the credibility of the given data center.
- Security issue is the main concern that prohibits many organizations from migrating their IT resources from on-premise to cloud-based.
- Security threats that make organizations hesitate to outsource IT environment are two-fold: possible malicious attack from outside and anxiety about keeping data outside of organization's physical boundary (not only business-wise but also legality-wise).
- Various levels of protection and security mechanisms: network isolation, firewalls and monitoring tools – big data analysis recently

Data Center Technology – 4/6

❑ Facilities

- Typically custom-designed computing resources, storages and network equipment for the given purpose
- Several functional layout areas based on power supplies, cabling, environmental control stations that regulate heating, ventilation, air conditioning, fire protection, (physical) security & access control system, monitoring system, etc.



❑ Computing hardware

- Mainly composed of standardized commodity servers with a number of computing hardware technologies such as:
 - Rack technology – standardized rack with interconnects for power, network, and internal cooling
 - CPU architecture – support for various CPU types: x86-32bits, x86-64bits, RISC, CISC, etc.
 - Multi-core CPU architecture – hundreds of physical & logical processing cores in single unit of standardized racks
 - Redundancy & hot-swap technology – hard disks, power supplies, network interfaces, storage controller cards, etc.
- Blade server technologies with rack-embedded physical interconnections (blade enclosures), fabrics (switches), power supply units, cooling fans, etc.
- Maximizes and enhances inter-component networking and management while optimizing physical space & power via individual server hot-swapping, scaling, replacement and maintenance
- Benefits the deployment of fault-resilient (tolerant) systems based on cluster technology



Data Center Technology – 5/6

- Several industry-standard and proprietary operational and management software tools that configure, monitor, and control hardware IT resources from remote & centralized consoles – self-provisioning
- Hundreds or even thousands of physical or virtual servers (IT resources) operated by a single operator

□ Storage hardware

- Needs to deal with tons of data created every day – easily reaching PBs of total scale in general
- One of the most difficult tasks to deal with in data center and many different levels of technologies for fast access, data availability, massive data accommodation, etc.:
 - RAID (Redundant Array of Independent/Inexpensive Disks) – integrating hundreds of individual HDD to provide fast, reliable, massive storage space
 - IO caching – at different layers: storage controllers, each physical/virtual servers, separate caching servers
 - Hot-swapping – replacing faulty HDD without requiring prior power down (a part of RAID technology)
 - Storage virtualization – abstracted storage layer creating virtual storage device free from the physical property of member storage devices
 - Data replication – memory snapshot, volume cloning, mirroring, DR, CDP, etc.
 - Distributed storage – file, block, object-level distributed storage: HDFS, Ceph, etc.
- Storage Topology
 - DAS (Direct Attached Storage): storages directly attached to a host system via block-level channel protocol such as SCSI/FC
 - NAS (Network Attached Storage): storages attached to a number of host systems via file-level network protocols such as NFS/CIFS/SMB – while providing file-level data sharing among multiple hosts
 - SAN (Storage Area Network): storages attached to multiple hosts via block-level network protocols such as Fibre Channel, Infiniband, iSCSI, etc.
- Data backup issues in data center

Data Center Technology – 6/6

❑ Network hardware

- One of the most important IT capabilities for data center to support remote IT access – broken down into five network subsystems in general
 - Carrier & external network interconnection ⇒ internetworking infrastructure comprised of backbone routers that provide routing between external WAN connections and LANs in the given data center including firewalls and VPN gateways
 - Web-tier load balancing and acceleration ⇒ for even distribution of web traffics and acceleration of web protocols comprised of XML pre-processors, encryption/decryption appliances (web acceleration), layer 7 switching devices (content-aware load balancing), etc.
 - LAN fabric ⇒ intranetworking infrastructure comprised of multiple layer 4 or lower switching devices up to ~10G bandwidth providing several virtualization functions such as LAN segregation into VLANs, link aggregation, control routing between networks, load balancing, failover (redundant connectivity), etc.
 - SAN fabric ⇒ data networking infrastructure composed of multiple SAN switching devices based on data networking protocols such as Fibre Channel (FC), Fibre Channel over Ethernet (FCoE), Infiniband (IB), Internet Small Computer Systems Interface (iSCSI)
 - NAS gateway ⇒ shared file-transfer networking infrastructure composed of a number of NAS-based storage devices based on file-transfer protocols such as NFS and SMB/CIFS (Samba)
- Basically redundant and/or fault-tolerant networking configurations for scalability and high availability
- DWDM (Dense Wavelength Division Multiplexing) devices for ultra high-speed networking and improved resiliency ⇒ in general for the purpose of high-speed real-time data replication between data centers

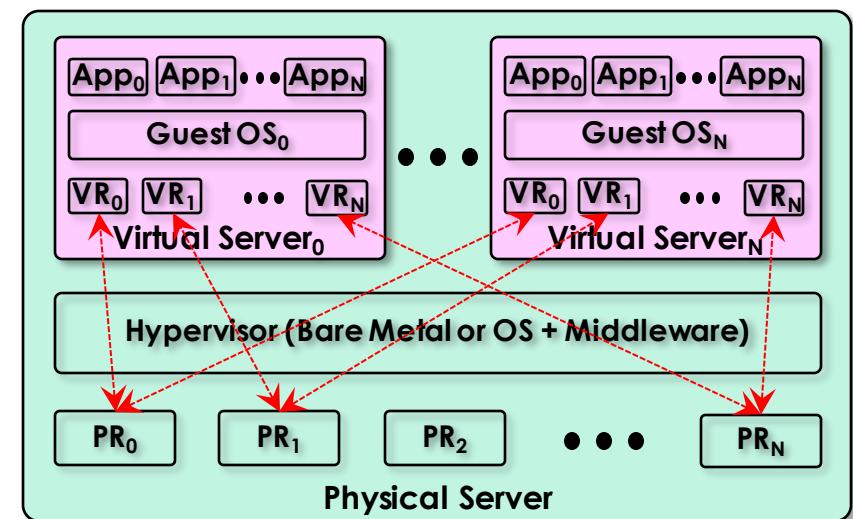
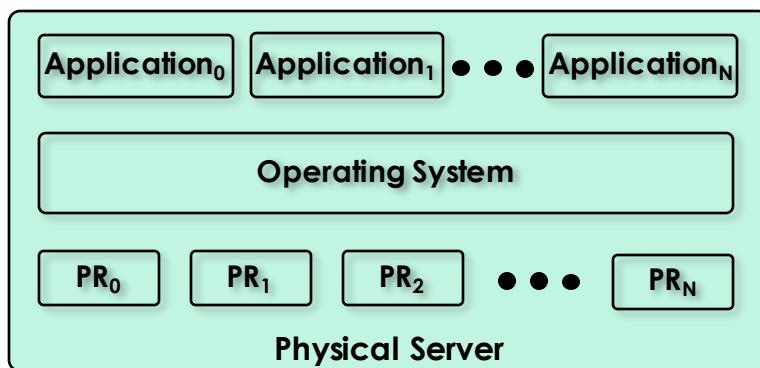
❑ Other consideration

- Technological obsolescence, heterogeneity, security, vast quantities of data and their backup, etc.

Virtualization Technology – 1/5

□ Basic concepts

- A process to convert a physical IT resource into a virtual IT resource
- Typical IT resources that can be virtualized: memory, CPUs, network, storage, power, **server**
- Server virtualization software: middleware vs. bare metal
 - Create a set of virtual IT resources on top of physical IT resources and maintain mapping between a set of virtual resources and physical resources
 - Create guest (user) operating systems
 - Guest OSs and application programs running on both physical or virtual environments **without any modification** ⇒ **a vital characteristic of virtualization**
 - Virtual machine manager, virtual machine monitor or **hypervisor**



Virtualization Technology – 2/5

❑ Hardware independence

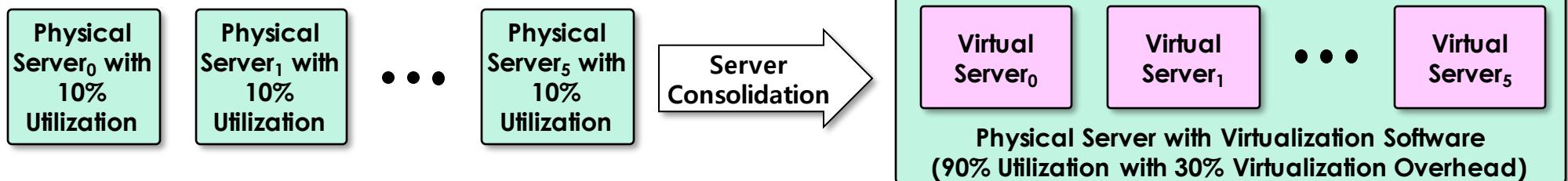
- Creating standardized soft (virtual) copies of physical IT resources ⇒ eliminating hardware dependency
- Easy automated VM migration or failover between different physical servers

❑ Server consolidation

- Creating different multiple virtual servers on a single physical server
- Mainly for Increasing server utilization, load balancing or optimizing IT resource utilization
- Support for common cloud features such as on-demand usage, resource pooling, elasticity, scalability and resiliency

❑ Resource replication

- Virtual servers are implemented as virtual disk images (configuration, memory state, etc.) containing binary file copies of hard disk content and being accessible via simple file operations such as copy and move host OS.
- Easy to be replicated, migrated, backed up and manipulated enabling:
 - Easy creation of standardized VM images with guest OS and pre-packaged application software in virtual disk images for instantaneous deployment
 - Increased agility in the migration and deployment of a virtual machine's new instance by being able to rapidly scale out and up

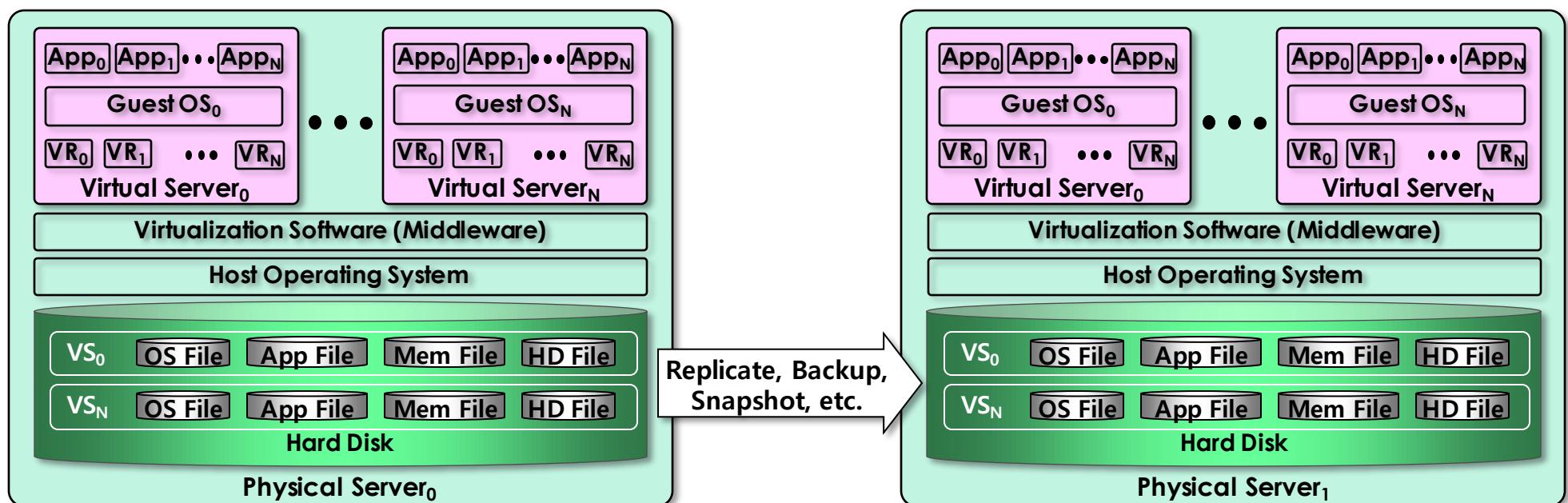


Virtualization Technology – 3/5

- Ability to roll back for instantaneous creation of VM snapshot by saving the state of the virtual server's memory and hard disk image to a host-based file
- Easy implementation of business continuity with efficient backup and restoration

❑ Operating system-based virtualization

- Install virtualization software in a pre-existing operating system (host vs. guest)
- Act as an application or more precisely as a middleware
- Easy to deal with hardware compatibility issues even with absence of a specific hardware driver
- Host OS services to be utilized: backup/recovery, integration to directory service, security management
- Performance degradation due to:

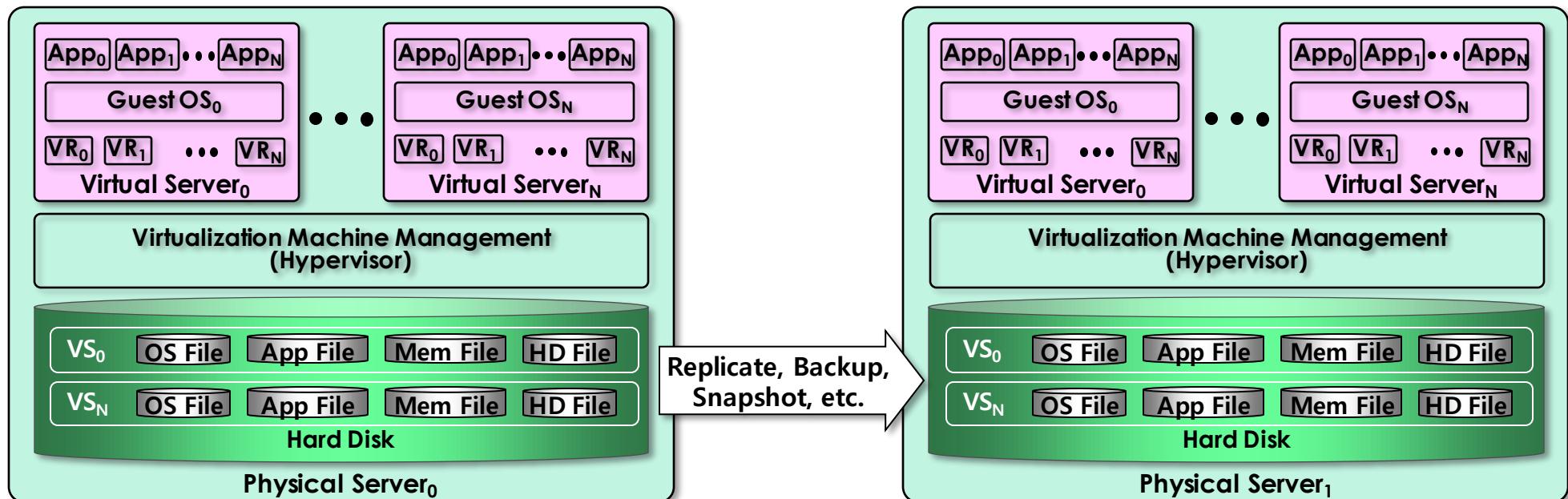


Virtualization Technology – 4/5

- IT resource (CPU, Memory, etc.) sharing with host and guest OSs
- Several additional traverse for each system call
- Additional license cost for host OS (Windows license or Linux subscription)

□ Hardware-based virtualization (Bare Metal)

- Install virtualization software (**Hypervisor**) on physical (bare metal) host hardware directly (no host OS)
- Eliminating one layer between host hardware and virtual servers reducing performance overhead
- Optimized and minimized thin software layer that handles hardware management functions to provide virtualization management - a sort of designated system, not a general purpose operating system
- Hardware compatibility issues – should provide OS-level of compatibility (device drivers, software...)
- No need for hypervisor device drivers to be as fully functional as those of general purpose OS



Virtualization Technology – 5/5

❑ Virtualization management

- Easier to administrate virtual servers than physical servers
- Many administration tasks automated by virtualization software
- VIM (Virtualization Infrastructure Management) tools – collectively manage virtual IT resources from a centralized & dedicated management computer (controller)

❑ Other consideration

- Performance overhead
 - Not ideal for complex systems with heavy workload
 - Excessive or unnecessary performance overhead with poorly formulated virtualization plan
 - Para-virtualization APIs – modified to reduce the guest OS's processing overhead ⇒ need to customize guest OSs to adapt them at the cost of sacrificing portability
- Special hardware compatibility
 - There are many vendors supplying specialized hardware devices and not all of them are compatible with the given virtualization software.
 - Old and existing software may not support those hardware recently released.
 - Solution: standardization, commoditization and frequent virtualization software update/upgrade
- Portability
 - Poor portability due to automated & programmatic VS management interfaces for their own
 - Demand for international standard such as OVF (Open Virtualization Format) for standardization of virtual disk formats in order to insure a wide range of VS management portability

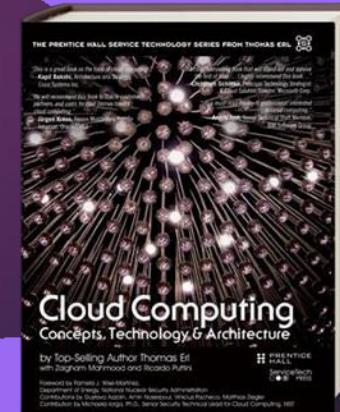
Cloud Computing

End of Lecture Note

Cloud Computing

Concept, Technology & Architecture

CHAPTER 06 FUNDAMENTAL CLOUD SECURITY



Contents

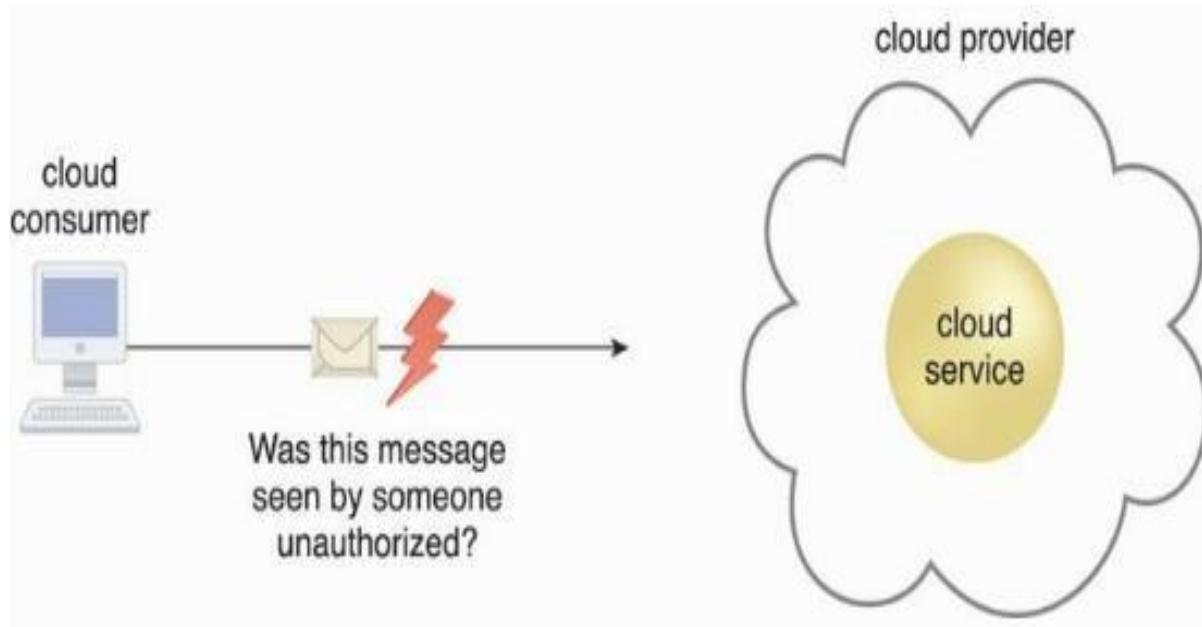
- This chapter introduces terms and concepts that address basic information security within clouds, and then concludes by defining a set of threats and attacks common to public cloud environments.
- 6.1 Basic Terms and Concepts
- 6.2 Threat Agents
- 6.3 Cloud Security Threats
- 6.4 Additional Considerations

6.1. Basic Terms and Concepts

- ❑ Information security is a complex ensemble of
 - ❑ techniques,
 - ❑ technologies, regulations, and
 - ❑ Behaviors
- ❑ that collaboratively protect the integrity of and access to computer systems and data.
- ❑ IT security measures aim to defend against threats and interference that arise from both malicious intent and unintentional user error.

Confidentiality

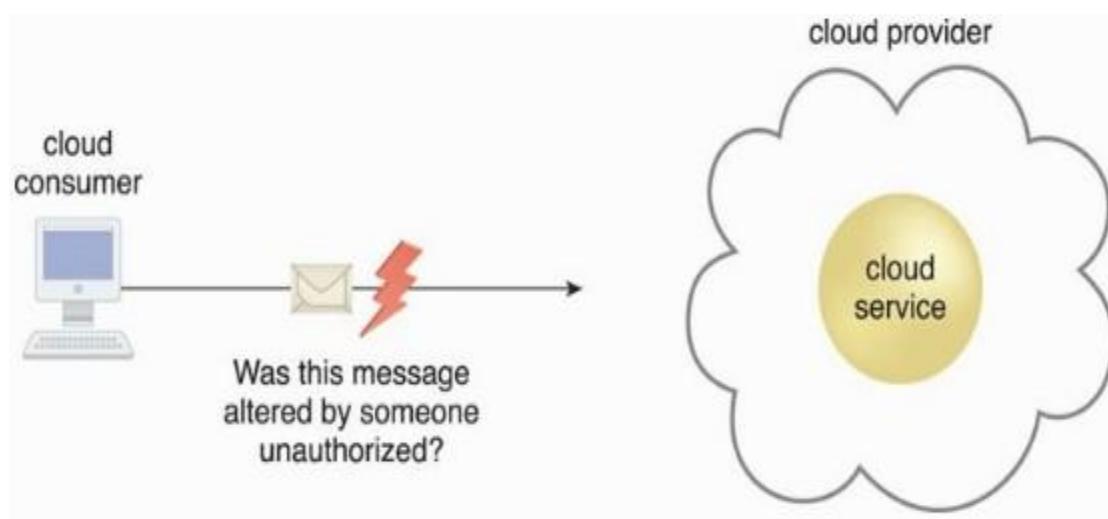
- ❑ The characteristic of something being made accessible only to authorized parties.



Within cloud environments, confidentiality primarily pertains to restricting access to data in transit and storage

Integrity

- ❑ The characteristic of not having been altered by an unauthorized party.



Important issue that concerns data integrity in the cloud is whether a cloud consumer can be guaranteed that the data it transmits to a cloud service matches the data received by that cloud service.

Integrity can extend to how data is stored, processed, and retrieved by cloud services and cloud-based IT resources.

Authenticity

- ❑ The characteristic of something having been provided by an authorized source. This also encompasses non-repudiation.
- ❑ Non-repudiation?
 - ❑ The inability of a party to deny or challenge the authentication of an interaction.

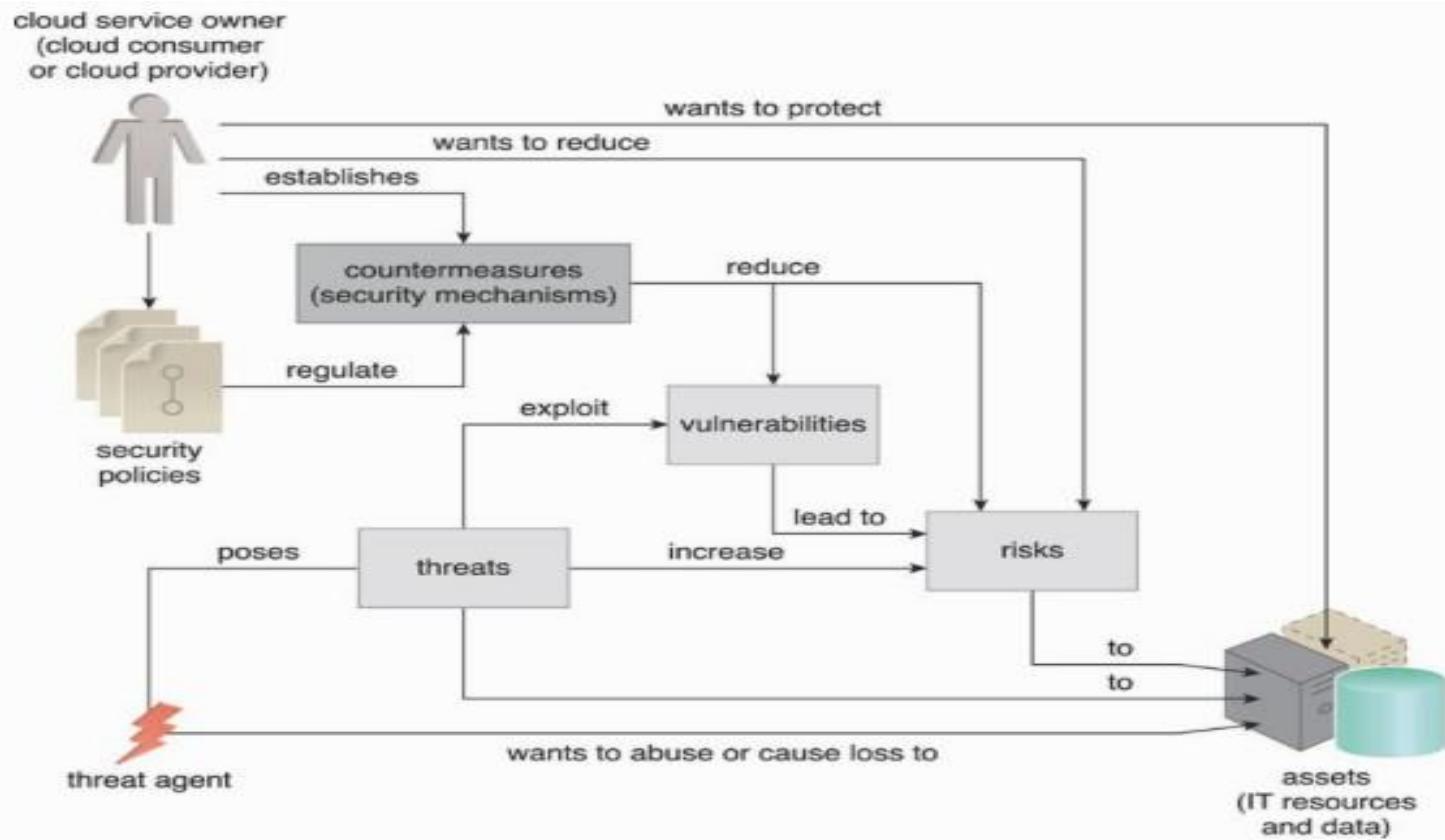
Other Relevant Terms

- ❑ **Availability** - being accessible and usable during a specified time period.
- ❑ **Threat** - a potential security violation that can challenge defenses in an attempt to breach privacy and/or cause harm.
- ❑ **Vulnerabilities** - a weakness that can be exploited either because it is protected by insufficient security controls.
- ❑ **Risk** - the possibility of loss or harm arising from performing an activity.

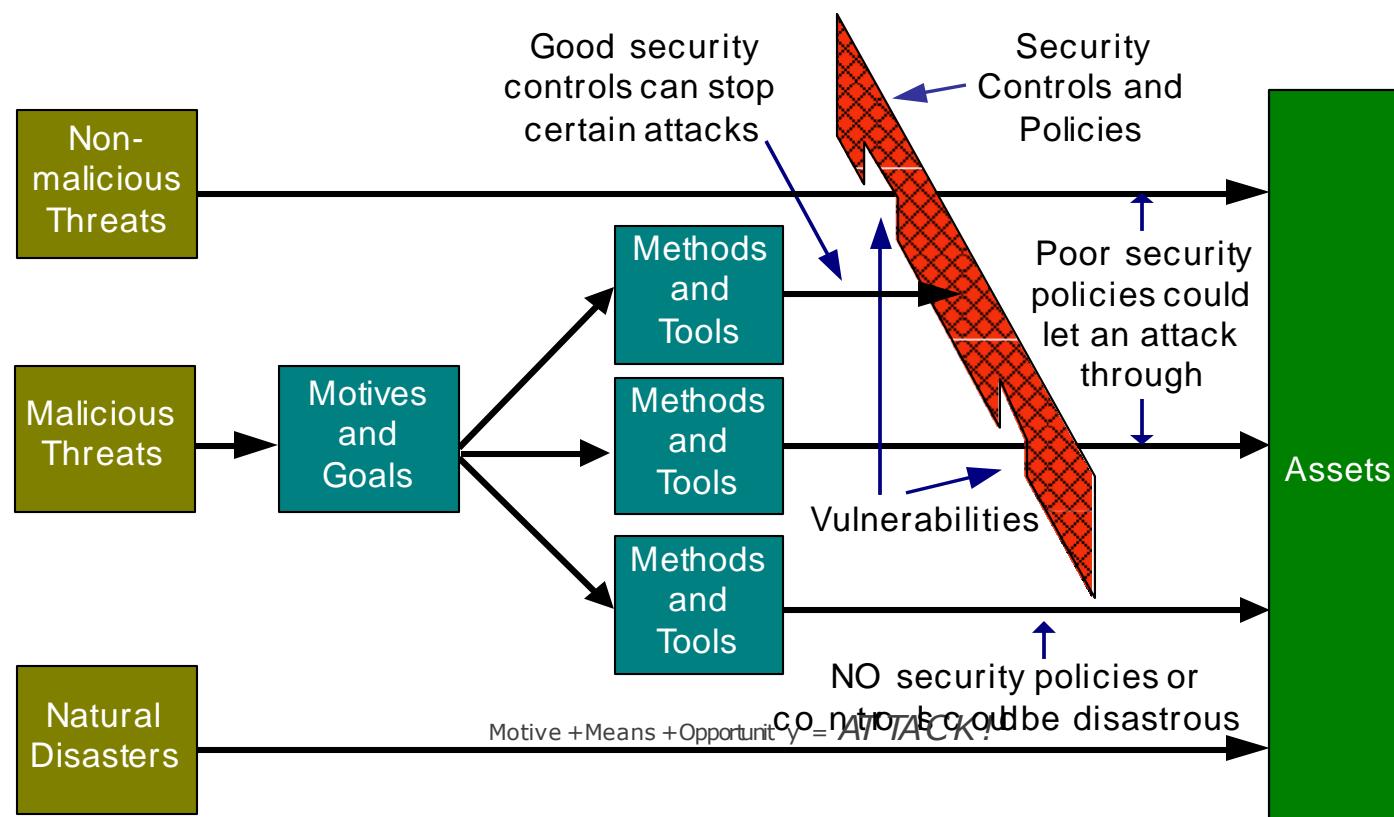
Other Relevant Terms(2)

- ❑ **Security Controls** - countermeasures used to prevent or respond to security threats and to reduce or avoid risk.
- ❑ **Security Mechanisms** - components comprising a defensive framework that protects IT resources, information, and services.
- ❑ **Security Policies** - a set of security rules and regulations that enforce security controls and mechanisms.

Figure 6.3. How security policies and security mechanisms are used to counter threats, vulnerabilities, and risks caused by threat agents.



The Ingredients of an Attack



6.2. Threat Agents

Common Threat Agents

- ❑ A threat agent?
 - ❑ an entity that poses a threat because it is capable of carrying out an attack.
- ❑ Originated from?
 - ❑ either internally or externally, from humans or software programs.
- ❑ Relationship?

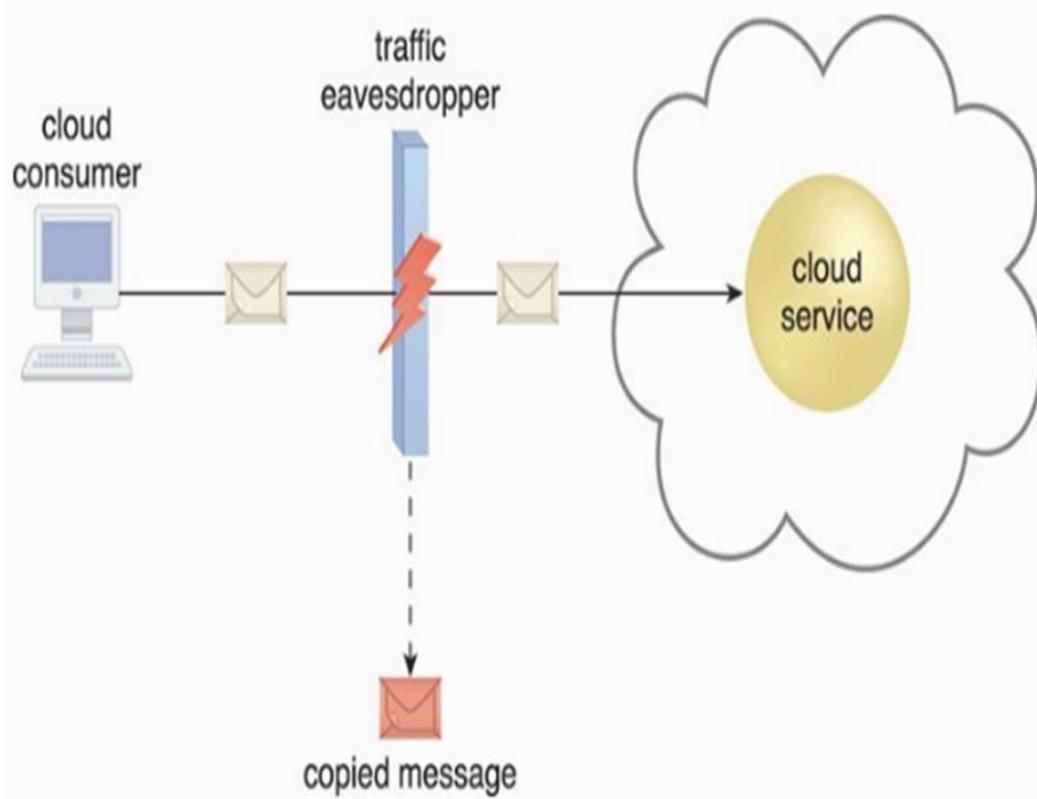
Types of Threat Agents/Attackers

- ❑ **Anonymous Attackers** - a non-trusted cloud service consumer without permissions in the cloud (typically external software programs).
- ❑ **Malicious Service Agent** - a service agent (or a program pretending to be a service agent) with compromised or malicious logic.
- ❑ **Trusted Attacker** - attacks from within a cloud's trust boundaries by abusing legitimate credentials.
- ❑ **Malicious Insider** - threat agents acting on behalf of or in relation to the cloud provider. They are typically current or former employees or third parties with access to the cloud provider's premises.

Anonymous Attacker

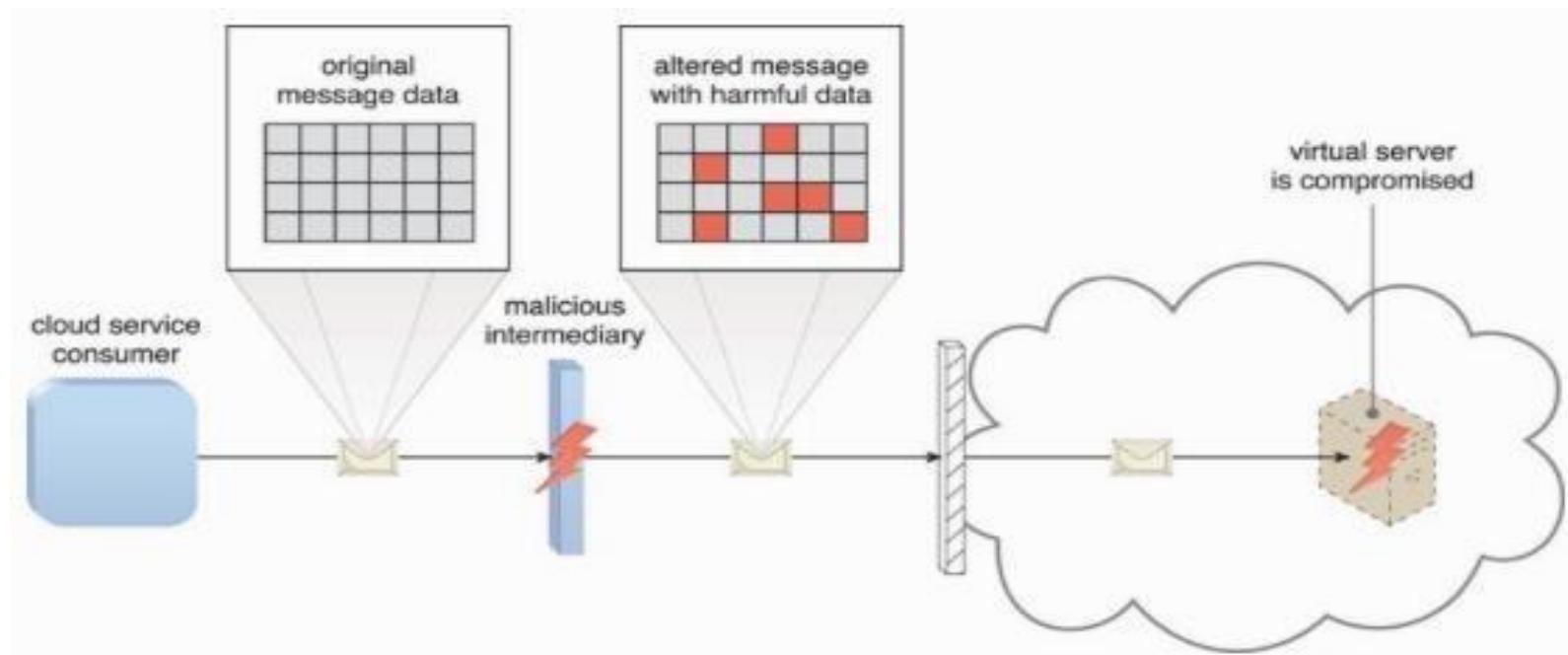
- ❑ An anonymous attacker is a non-trusted cloud service consumer without permissions in the cloud .
- ❑ It typically exists as an external software program that launches network-level attacks through public networks. When anonymous attackers have limited information on security policies and defenses, it can inhibit their ability to formulate effective attacks.
- ❑ Therefore, anonymous attackers often resort to committing acts like bypassing user accounts or stealing user credentials, while using methods that either ensure anonymity or require substantial resources for prosecution.

6.3. Cloud Security Threats



- ❑ **Traffic Eavesdropping**
 - data is passively intercepted by malicious service agents.
- ❑ Gather information to directly compromise confidentiality, e.g., username and password.

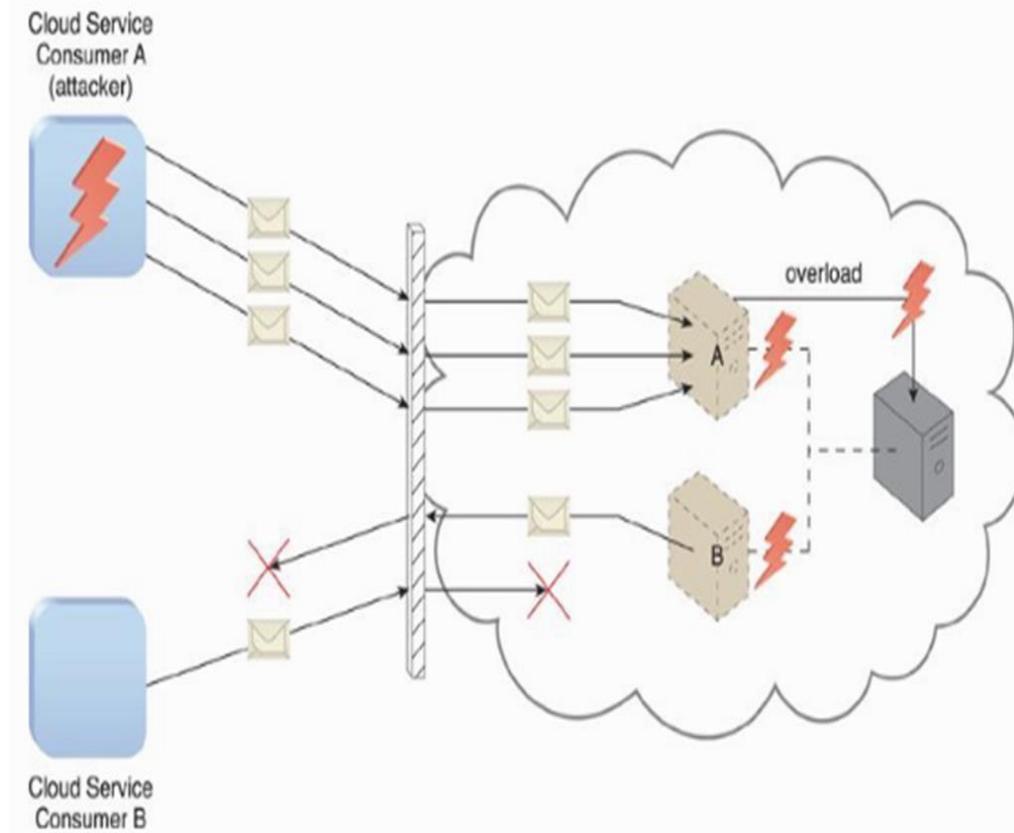
Malicious Intermediary



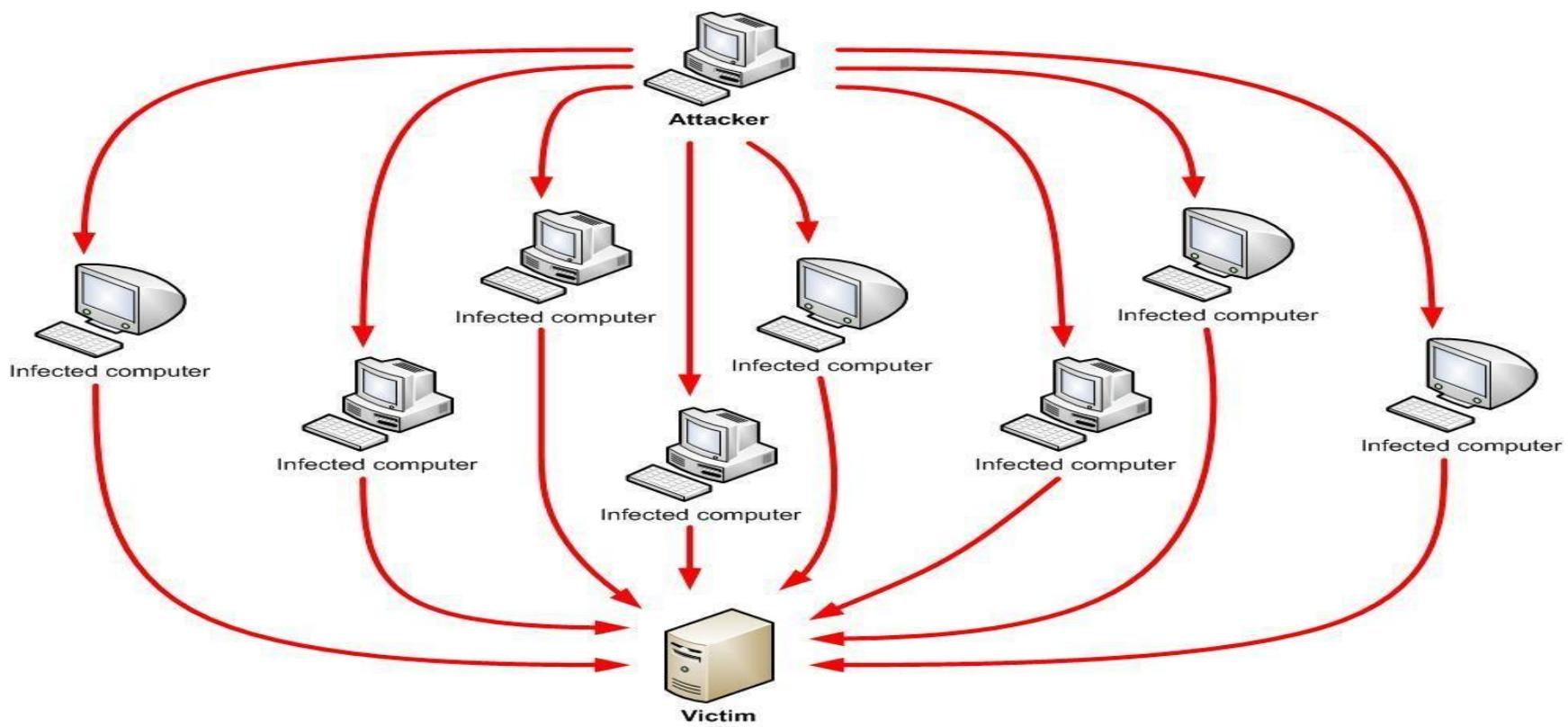
- ❑ This attack arises when messages are intercepted and altered by a malicious service agent.

Denial of Service

- ❑ To overload IT resources to the point where they cannot function properly.
 - ❑ Workload increased (CPU, memory loads)
 - ❑ Network traffic increased
- ❑ Successful DoS attacks produce server degradation and/or failure.

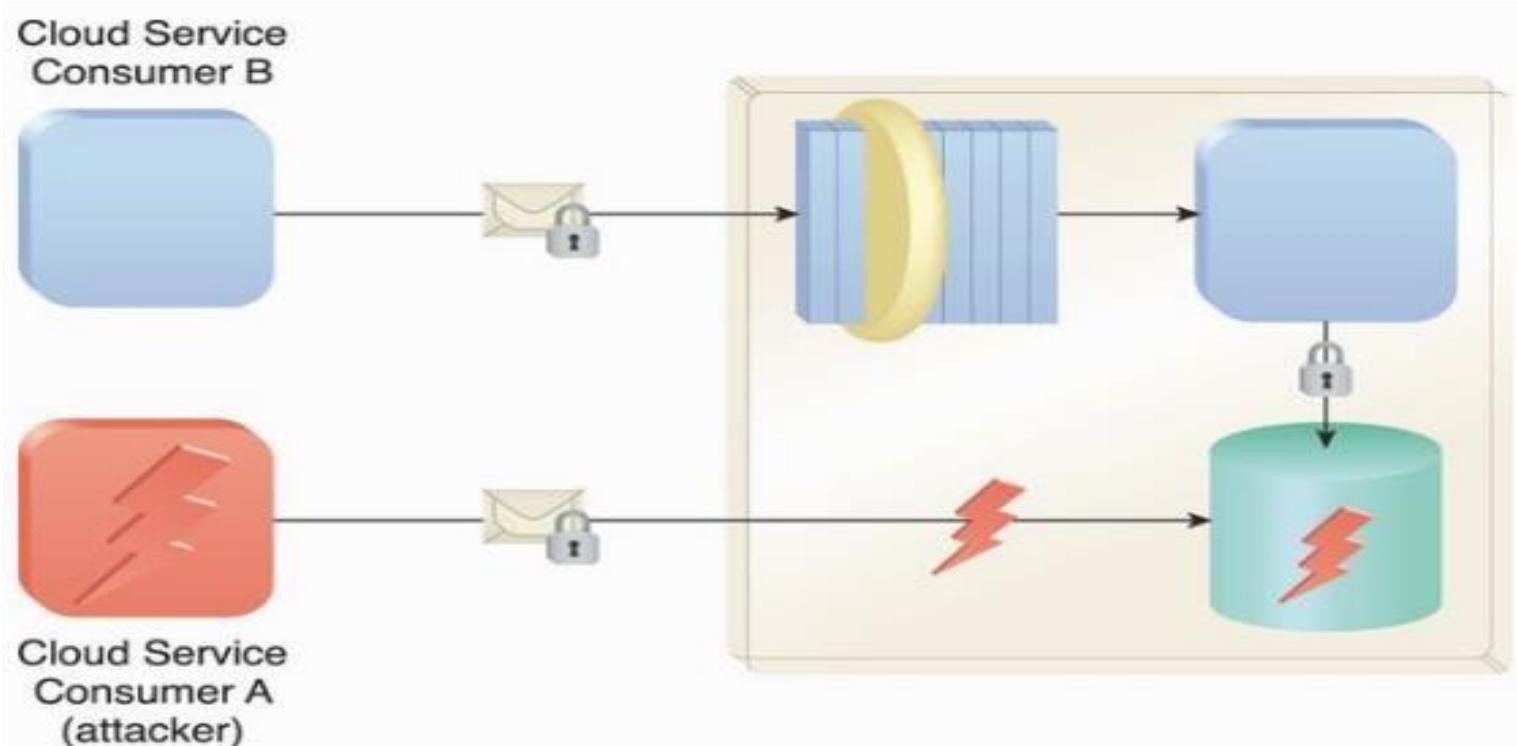


Distributed DoS (DDoS)



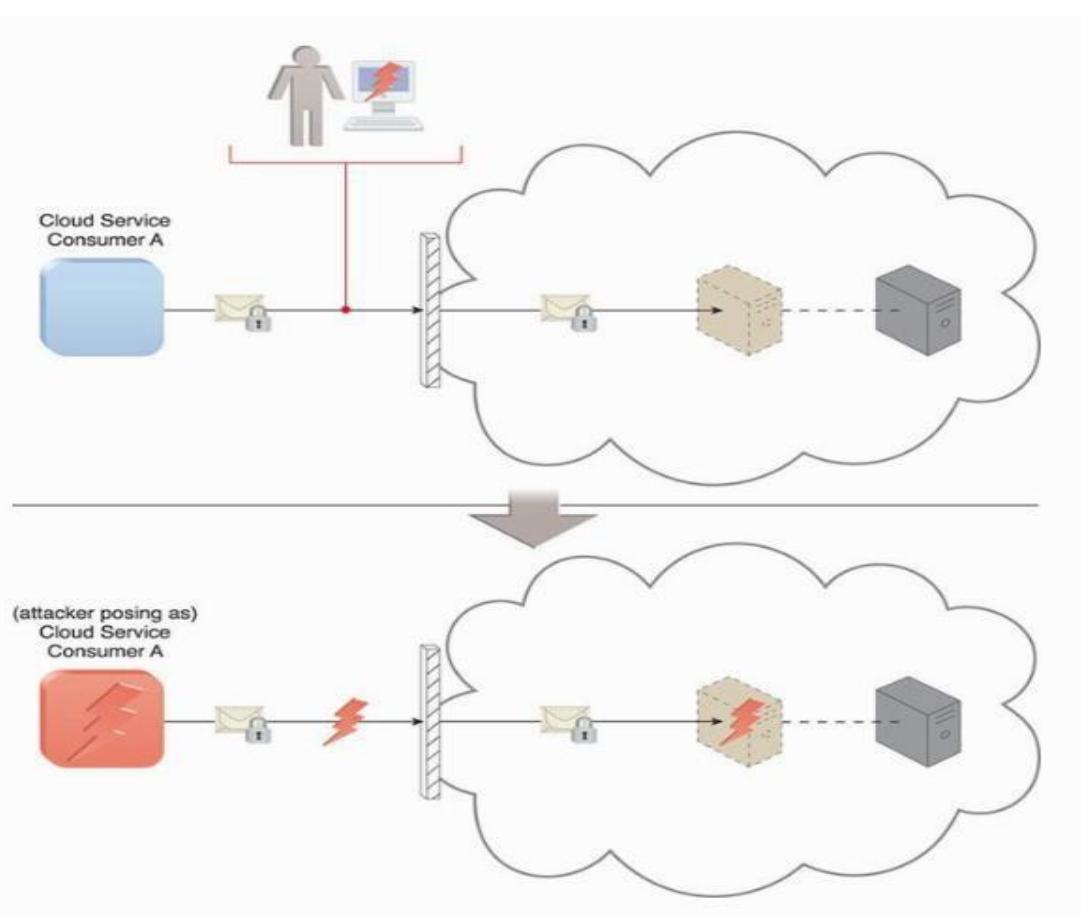
- DoS is easy to detect (trace back) and mitigate.

Insufficient Authorization



- ▶ Attackers gain direct access to IT resources through poorly managed cloud API.

Weak Authentication

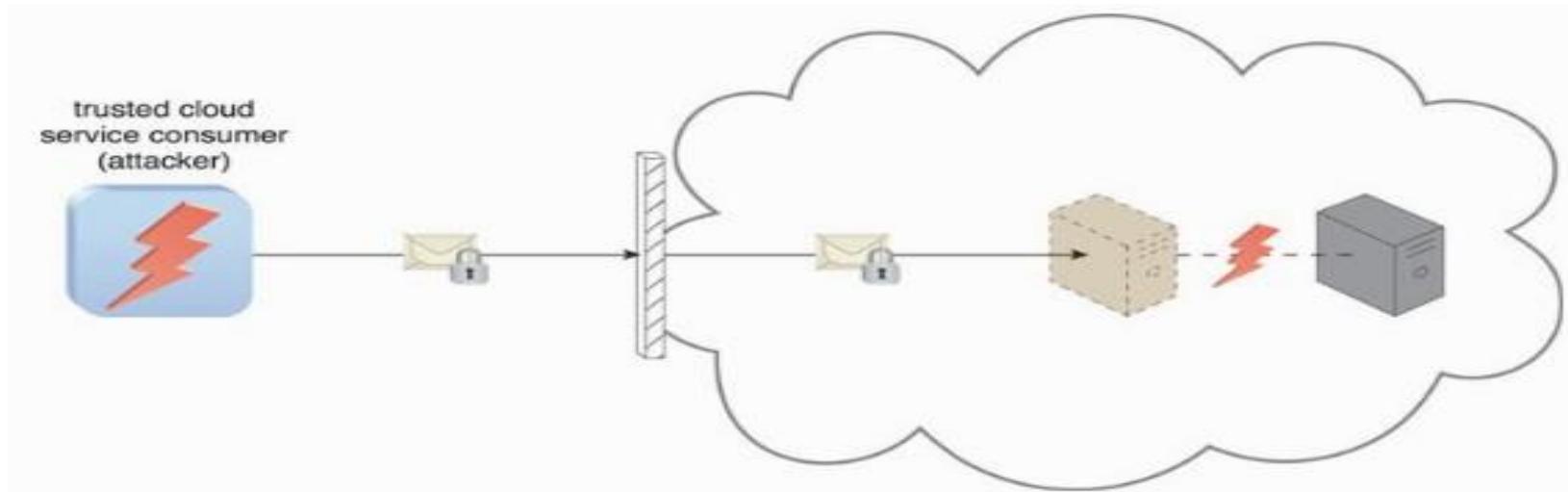


- Cloud consumer A uses a weak password enabling an attacker to easily crack it.

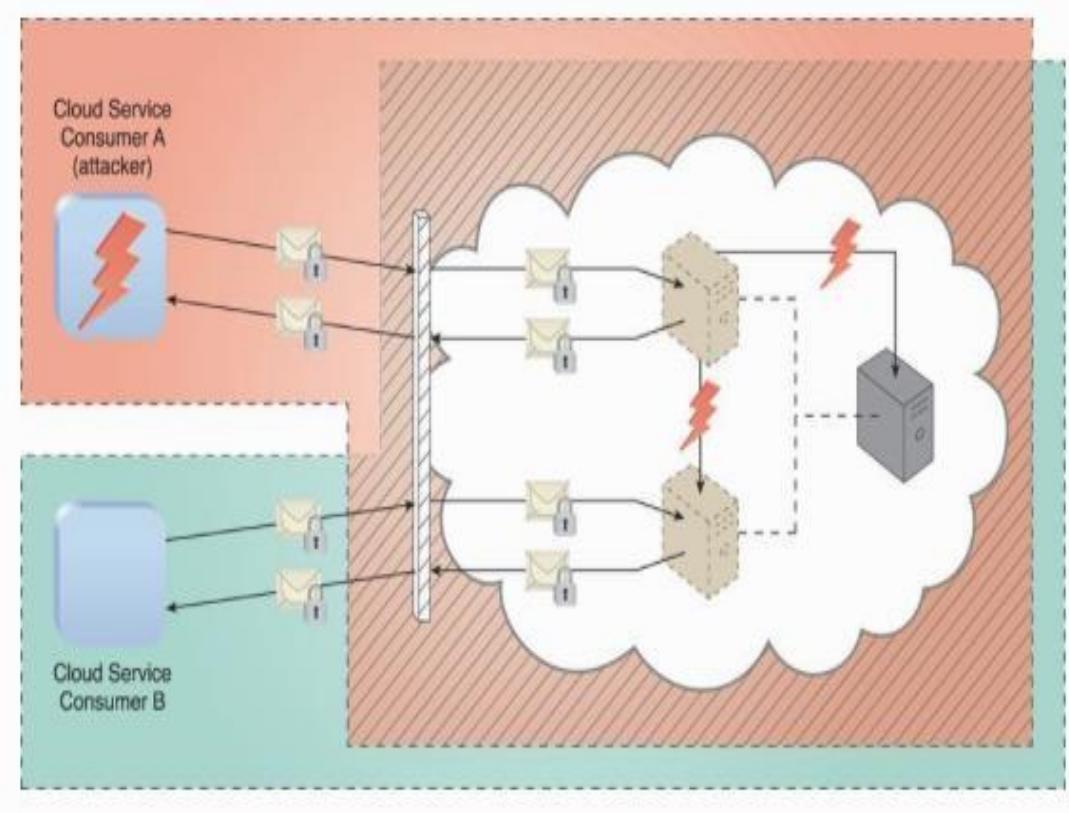
Virtualization Attack

to jeopardize its confidentiality, integrity, and/or availability.

- ❑ Accesses a virtual server to compromise its underlying physical server.
- ❑ This attack exploits vulnerabilities in the virtualization platform



Overlapping Trusted Boundaries



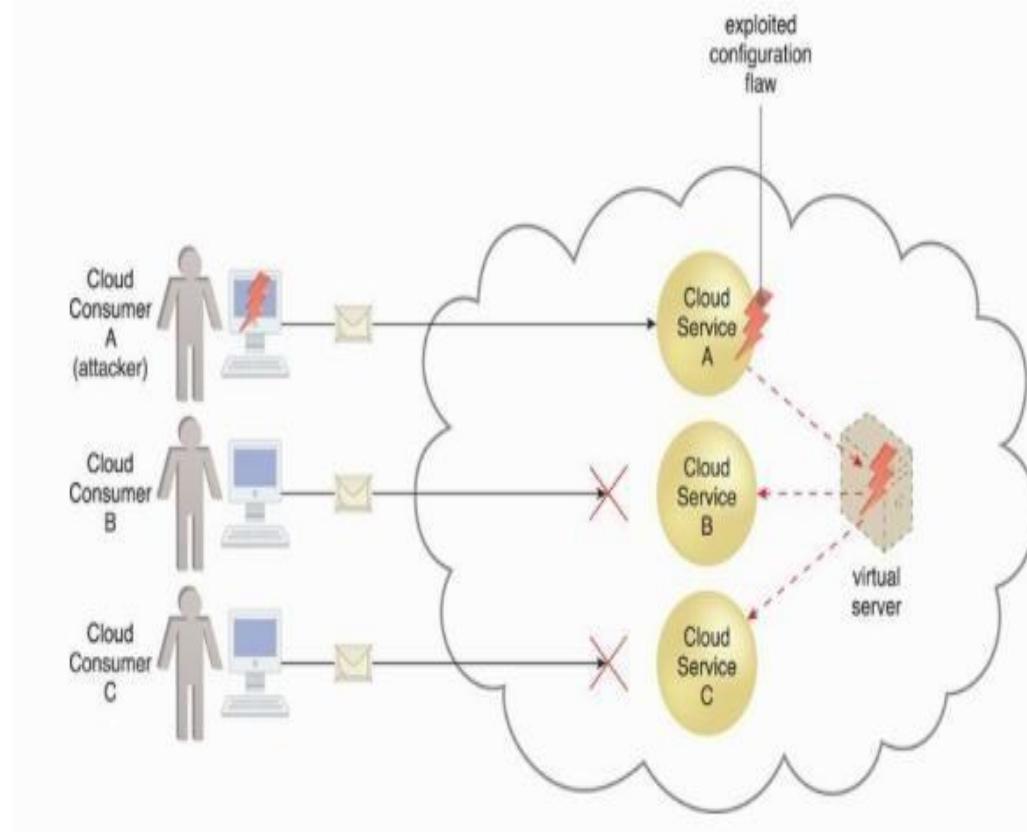
- ▶ Physical IT resources shared by multiple cloud consumers, resulting in overlapping trusted boundaries.
- ▶ Malicious cloud consumers target shared IT resources with the intention of compromising cloud consumers or other IT resources that share the same trust boundary.

6.4. Additional Considerations

- ❑ Flawed Implementation
- ❑ Security Policy Disparity
- ❑ Contracts
- ❑ Risk Management

Flawed Implementation

- ❑ Substandard design, implementation, or configuration of cloud service deployments may lead to undesirable consequences.
- ❑ Attackers can exploit these vulnerabilities to impair the integrity, confidentiality, and/or availability of cloud provider IT resources.



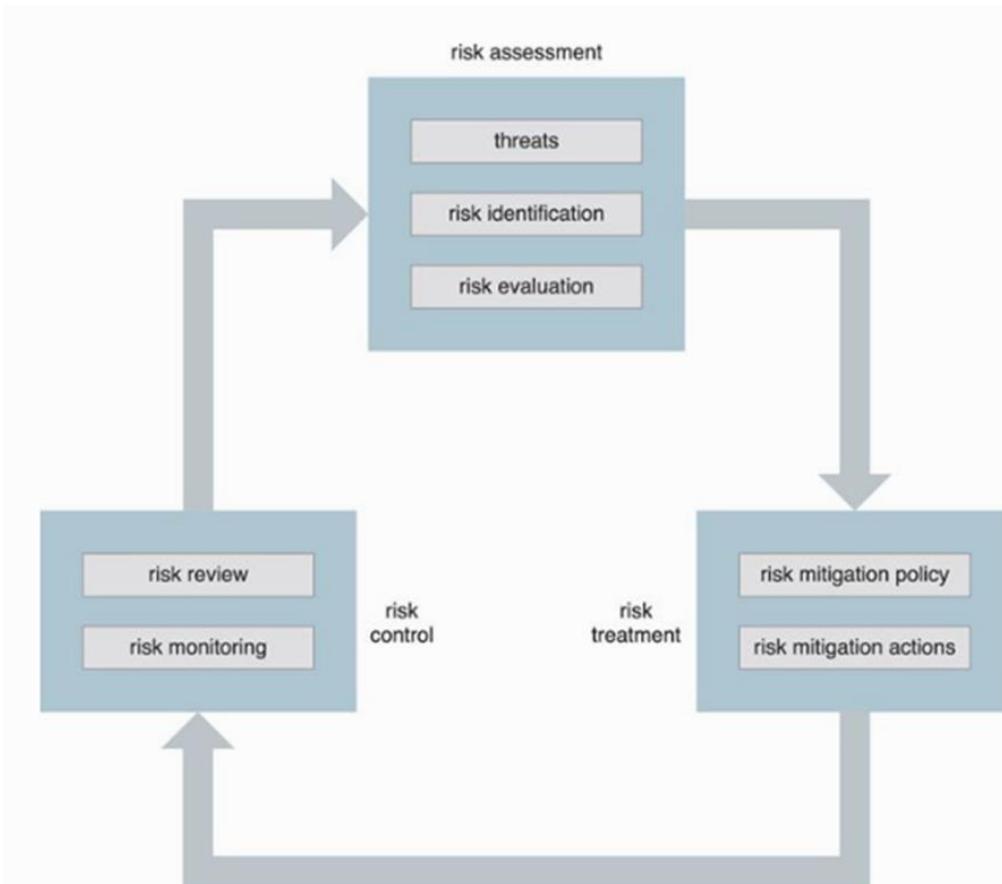
Security Policy Disparity

- ❑ Our own implemented security mechanisms are different from that provided by cloud providers.
- ❑ Assessments are needed to ensure our IT resources being migrated to a cloud are sufficiently protected.
- ❑ Cloud consumers may not be granted sufficient administrative control (of course we are not the owner of the cloud infra).
- ❑ Some public clouds, additional third parties, such as security brokers and certificate authorities, may introduce their own distinct set of security policies and practices (make things more complicated).

Contracts

- ❑ Examine contract and SLA.
- ❑ Use clear language that indicates the amount of liability assumed by the cloud provider and/or the level of indemnity the cloud provider may ask for.
- ❑ Contractual obligations is where the lines are drawn between cloud consumer and cloud provider assets. In case of security breach, who to be blamed (us or cloud provider).

Risk Management



Risk Management

- ❑ When assessing the potential impacts and challenges pertaining to cloud adoption, cloud consumers are encouraged to perform a formal risk assessment as part of a risk management strategy. Process comprises
 - ❑ Risk assessment – to identify potential vulnerabilities and shortcomings.
 - ❑ Risk treatment – mitigation policies and plans to treat risks.
 - ❑ Risk control – risk monitoring
- ❑ Risk management is an on-going process.

THANK YOU

Cloud computing mechanisms

Cloud infrastructure mechanisms

Cloud Infrastructure Mechanisms

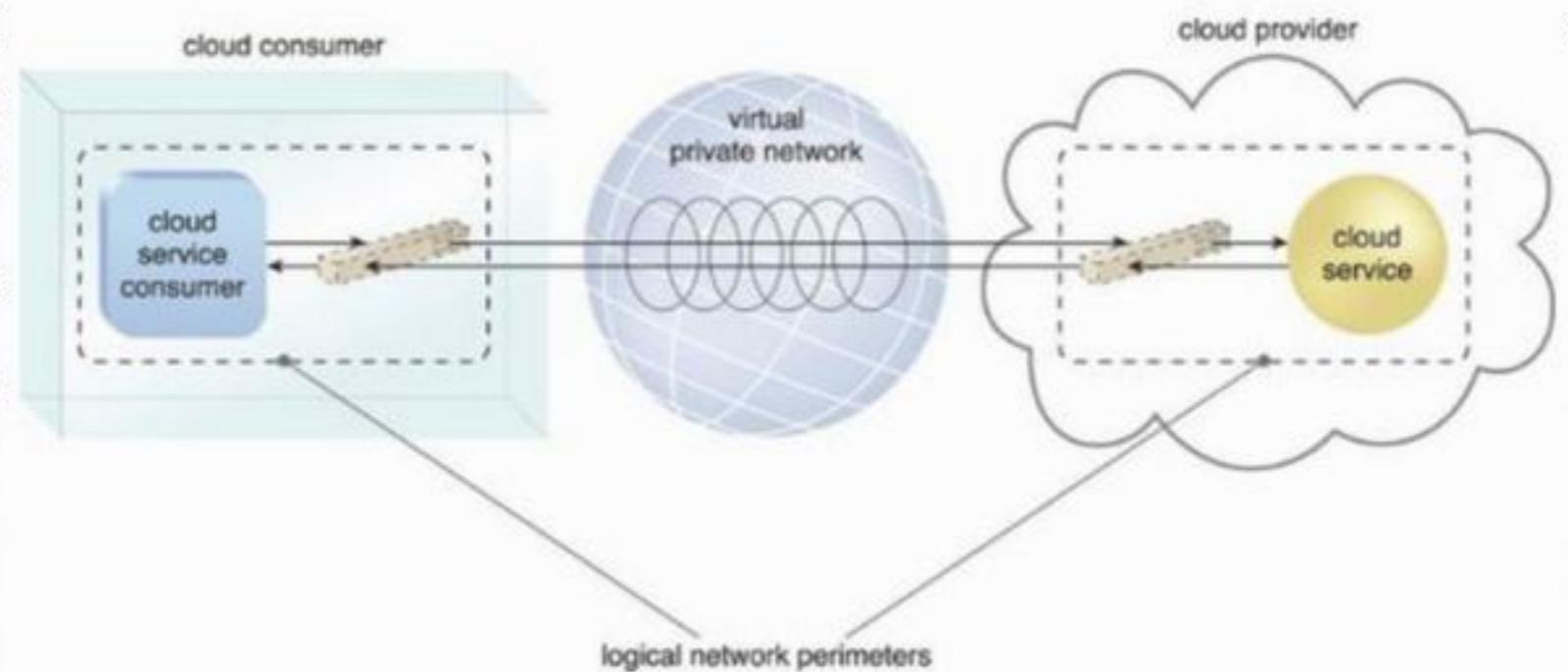
- Foundational building blocks of cloud environments, which comprises
 - Logical Network Perimeter
 - Virtual Server
 - Cloud Storage Device
 - Cloud Usage Monitor
 - Resource Replication
 - Read-Made Environment

Logical Network Perimeter

- An isolation of network environment establishing a virtual network boundary.
- Purposes?
 - isolate IT resources in a cloud from non-authorized users,
 - isolate IT resources in a cloud from non-users,
 - isolate IT resources in a cloud from cloud consumers, and
 - control the bandwidth that is available to isolated IT resources.

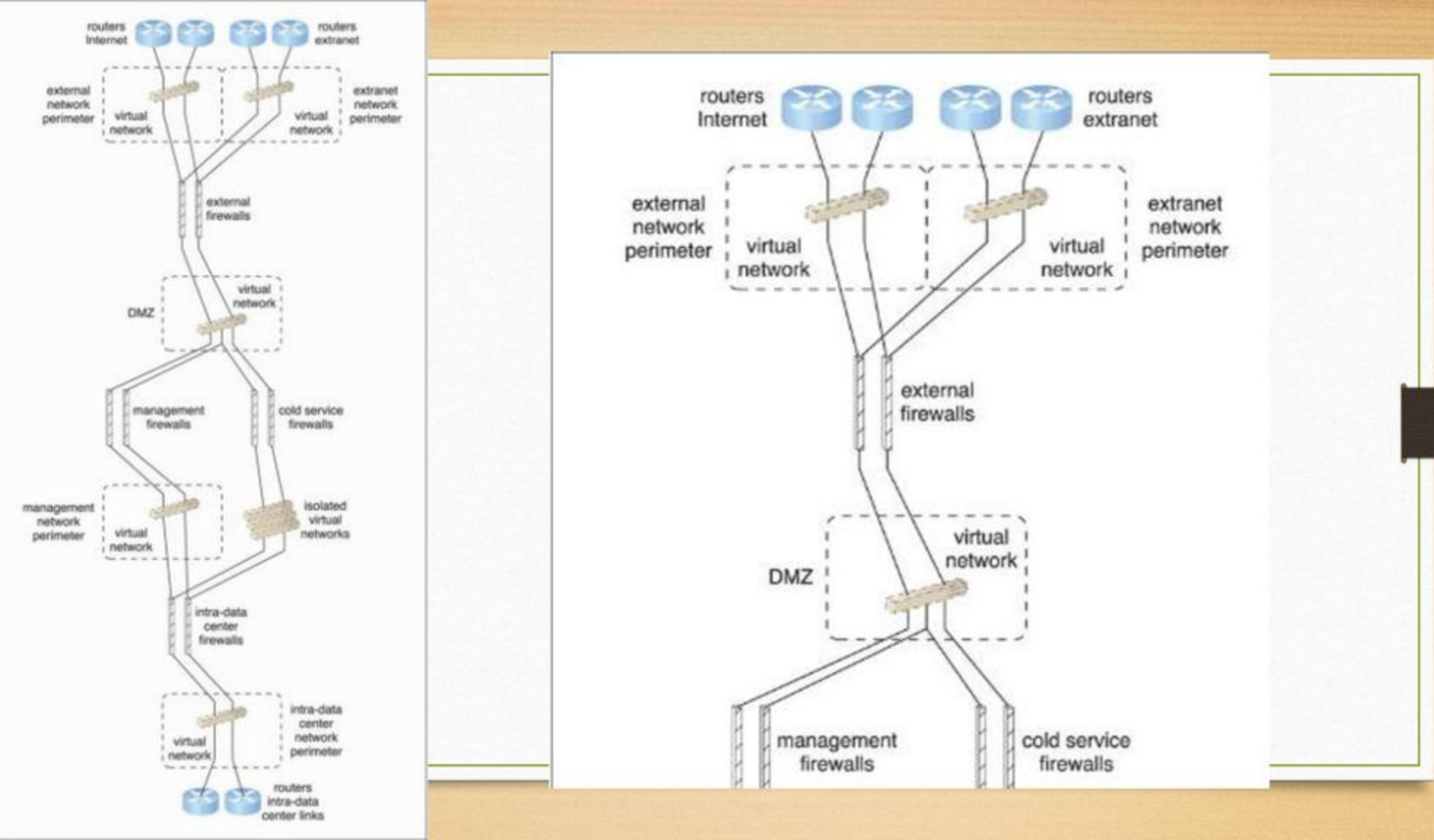
Logical Network Perimeter (2)

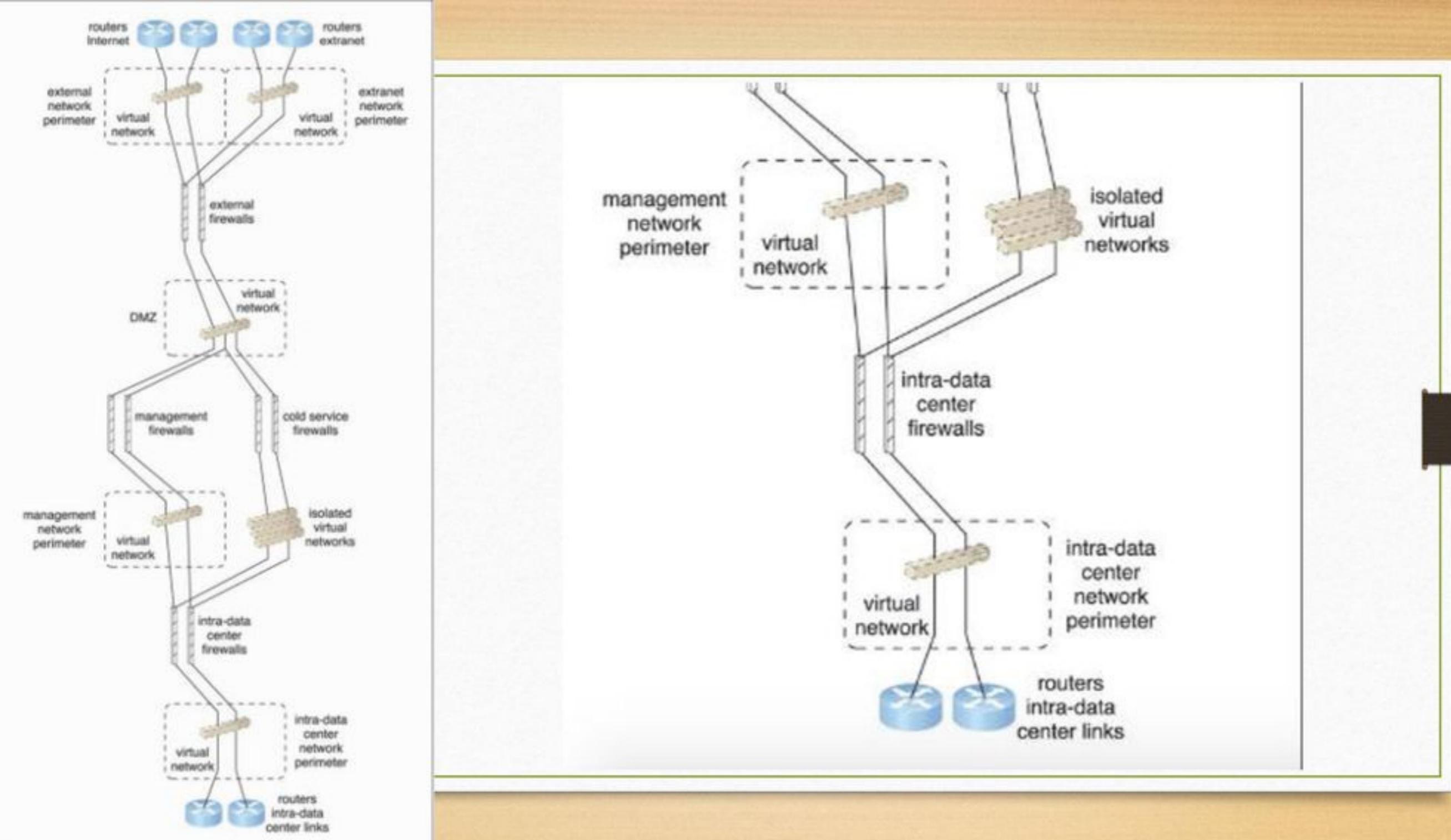
- Typical
connec
environ
 - Virt
 - Virt



Case Study (DTGOV)

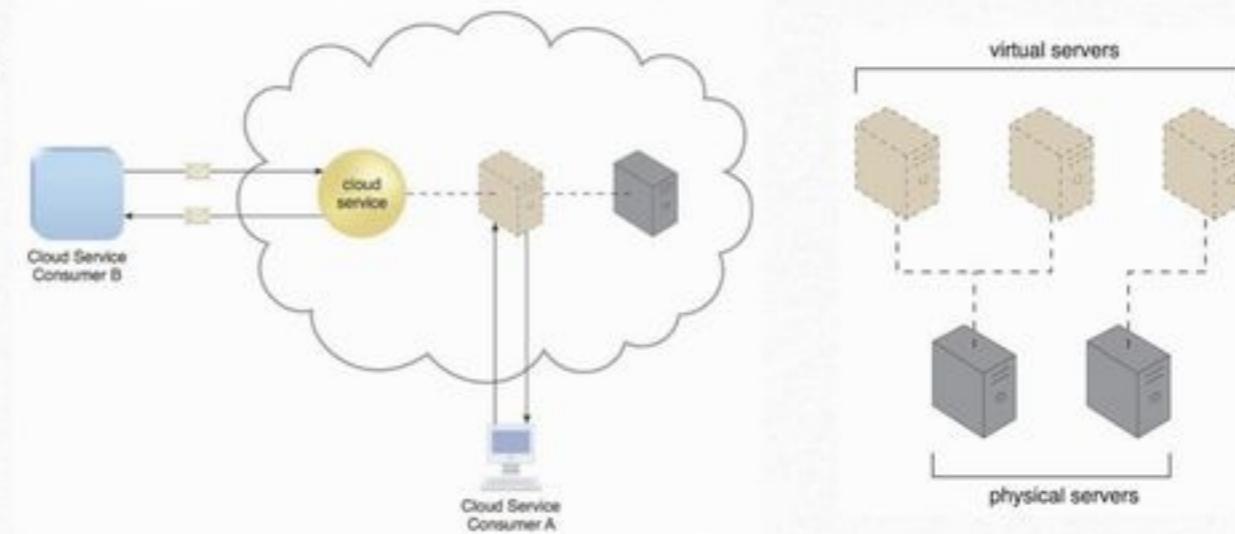
- Routers – connect the Internet and the extranet.
- DMZ zone – virtual network hosting the proxy servers.
- Management firewalls – isolate the management perimeter, providing management services.
- Cold service firewalls – isolate traffic to cloud-based IT resources.
- Intra-data center firewalls – filter network traffic to and from other data centers via routers.





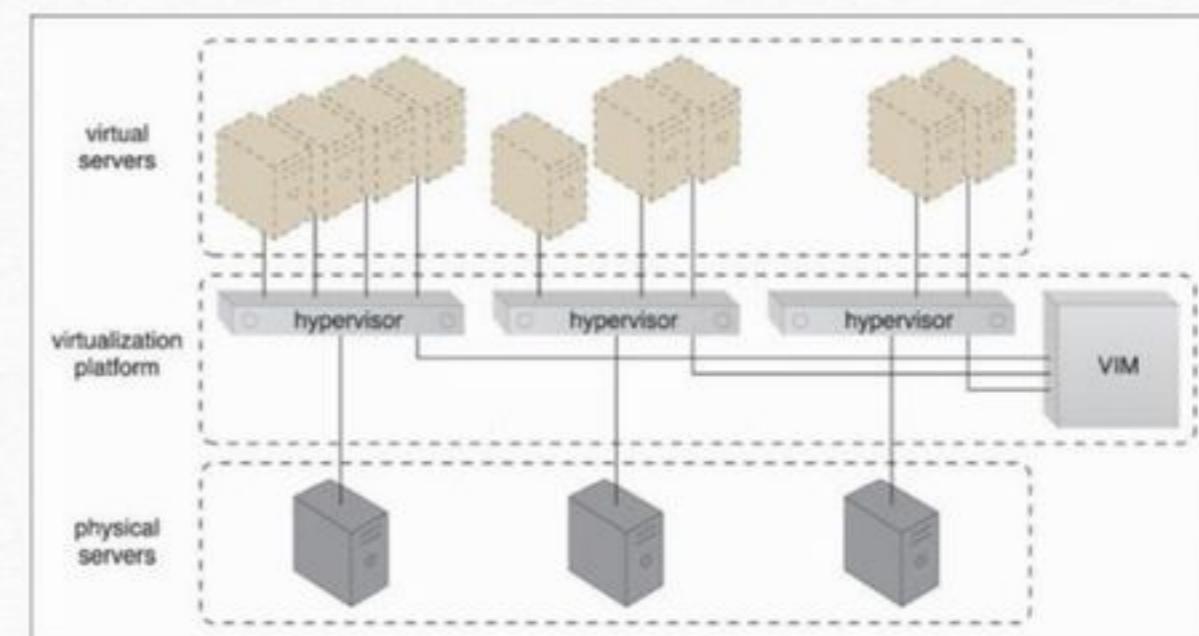
Virtual Servers

- A form of virtualization software that emulates a physical server.
- Used by a cloud provider for resources sharing.
- Virtual server = virtual machine



Case Study (DTGOV) Continued.

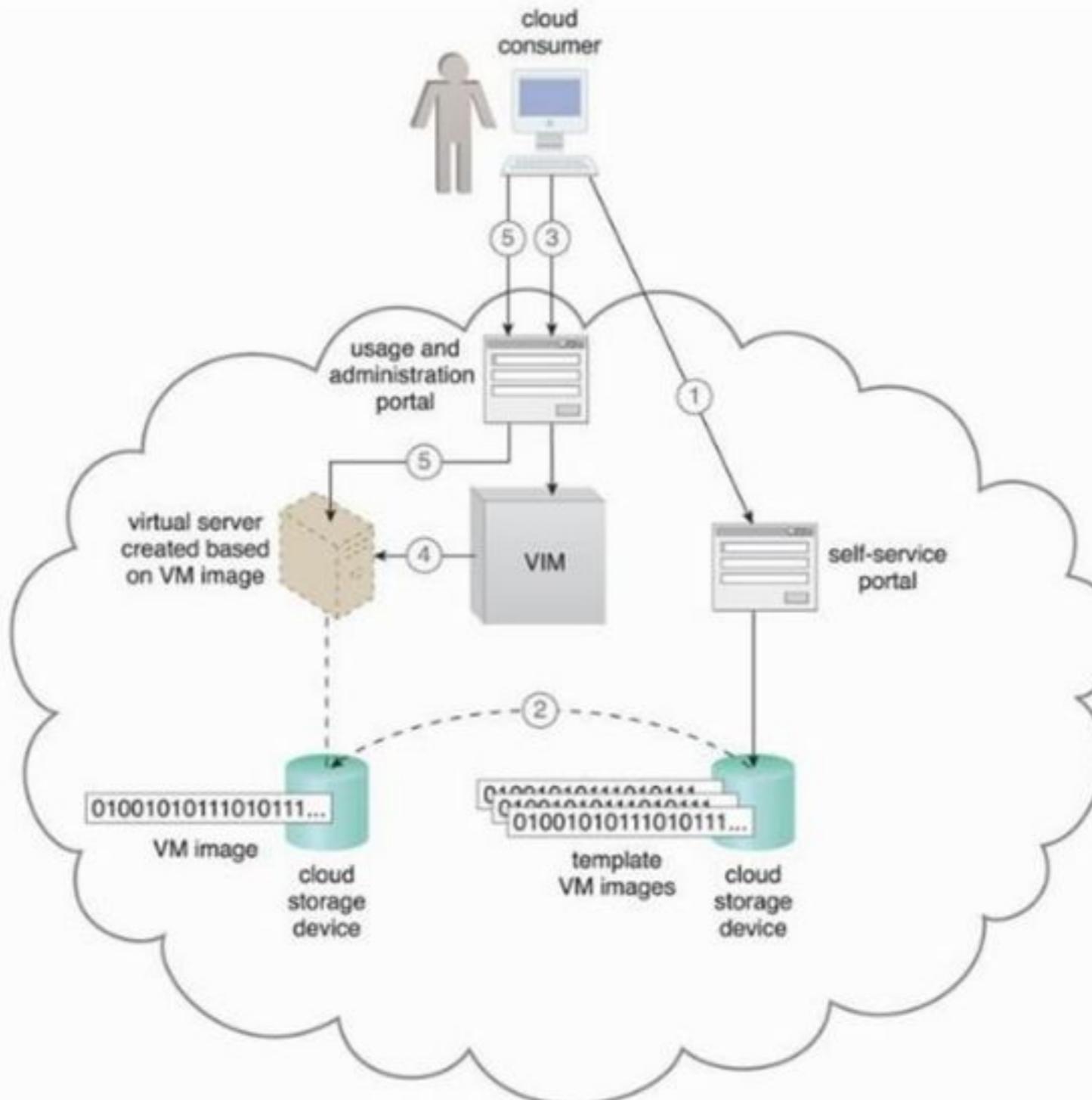
- DTGOV offers several types of pre-made VM images for its customers.
- VM images = virtual disk images used by a hypervisor to boot virtual servers.
- Template virtual servers.



Case Study (DTGOV) Continued

- Template Virtual Servers (may include pre-installed software/applications) examples:
- **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
- **Ultra-Large Virtual Server Instance** – 128 virtual processor cores, 512 GB of virtual RAM, 40 GB of storage space in the root file system

- Additional virtual machines
- Cloud consumer can see the list of VMs
- The allocation is customized
- VIM creates VMs

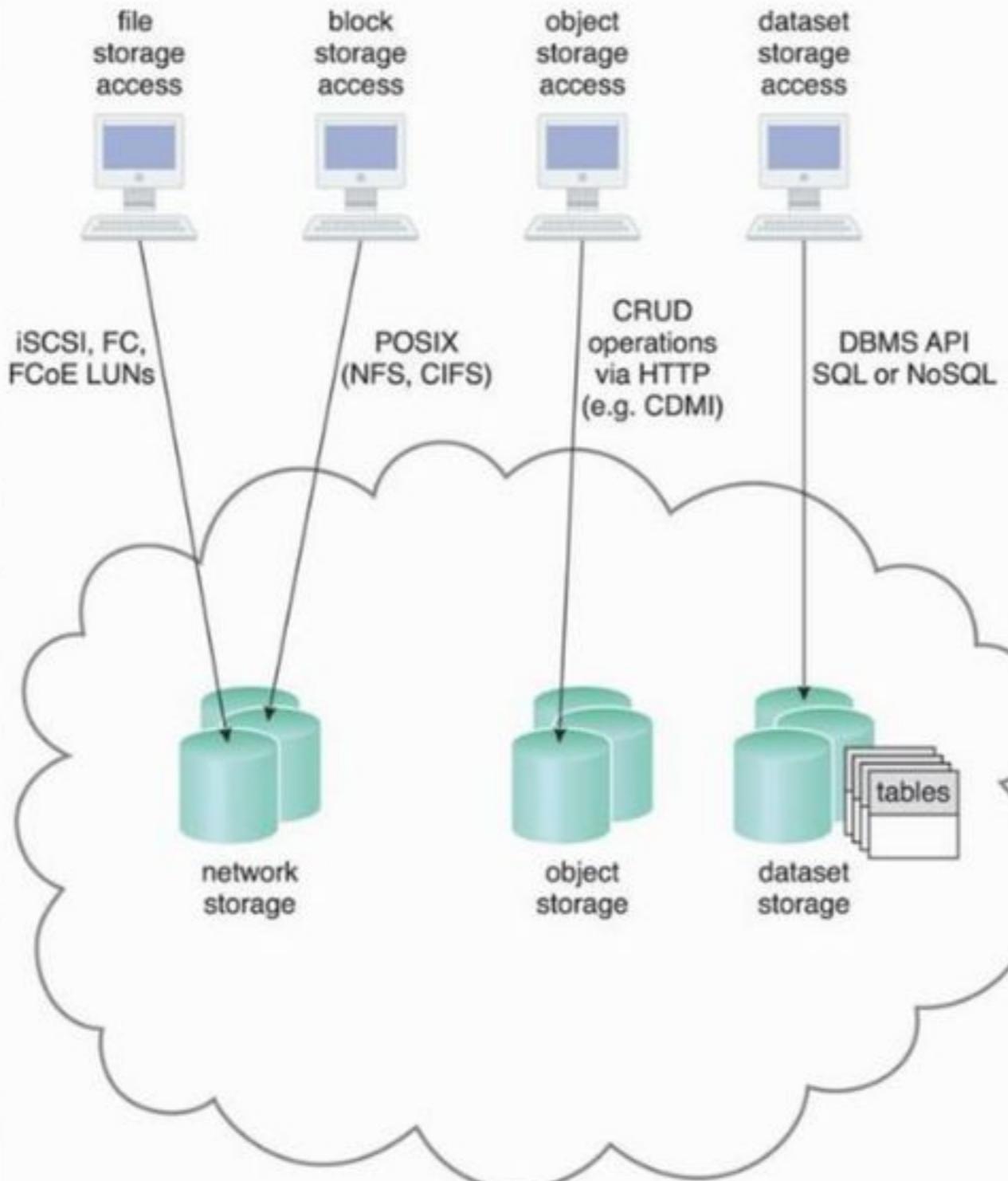


1
ching a
ite from
cal server.

Cloud Storage Devices Mechanism

- Storage devices designed specifically for cloud-based environment.
- Instances of these storage could be virtualized.
- Able to provide fix-increment capacity allocation in support of pay-per-use mechanism.
- Primary concern - CIA

- Files – Collection of data in folders.
- Blocks – The smallest unit of storage.
- Datasets – Large files in a specific format.
- Objects – Individual resources.



lware, a block-based, or record-based

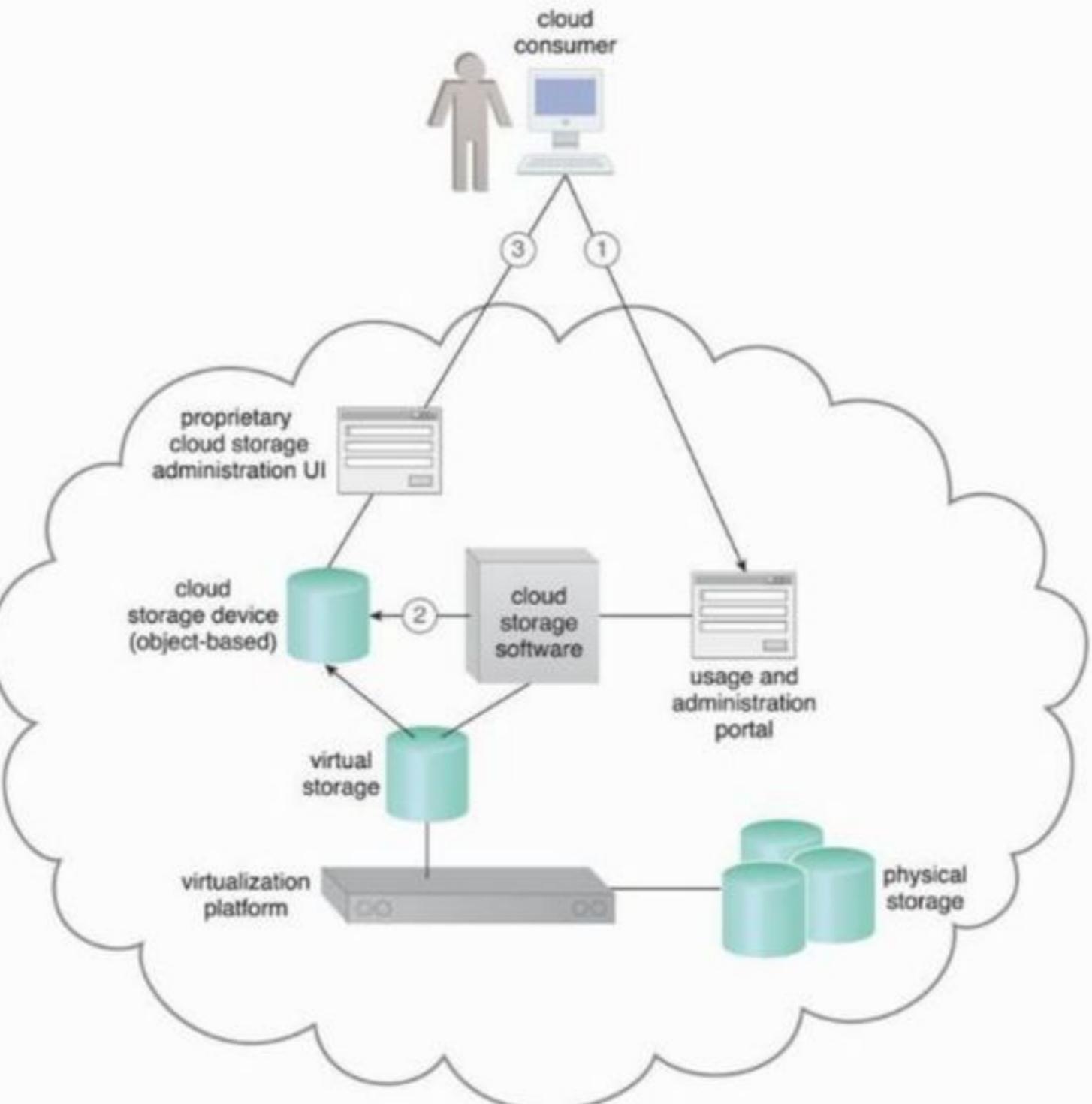
Technical Interfaces to Storage

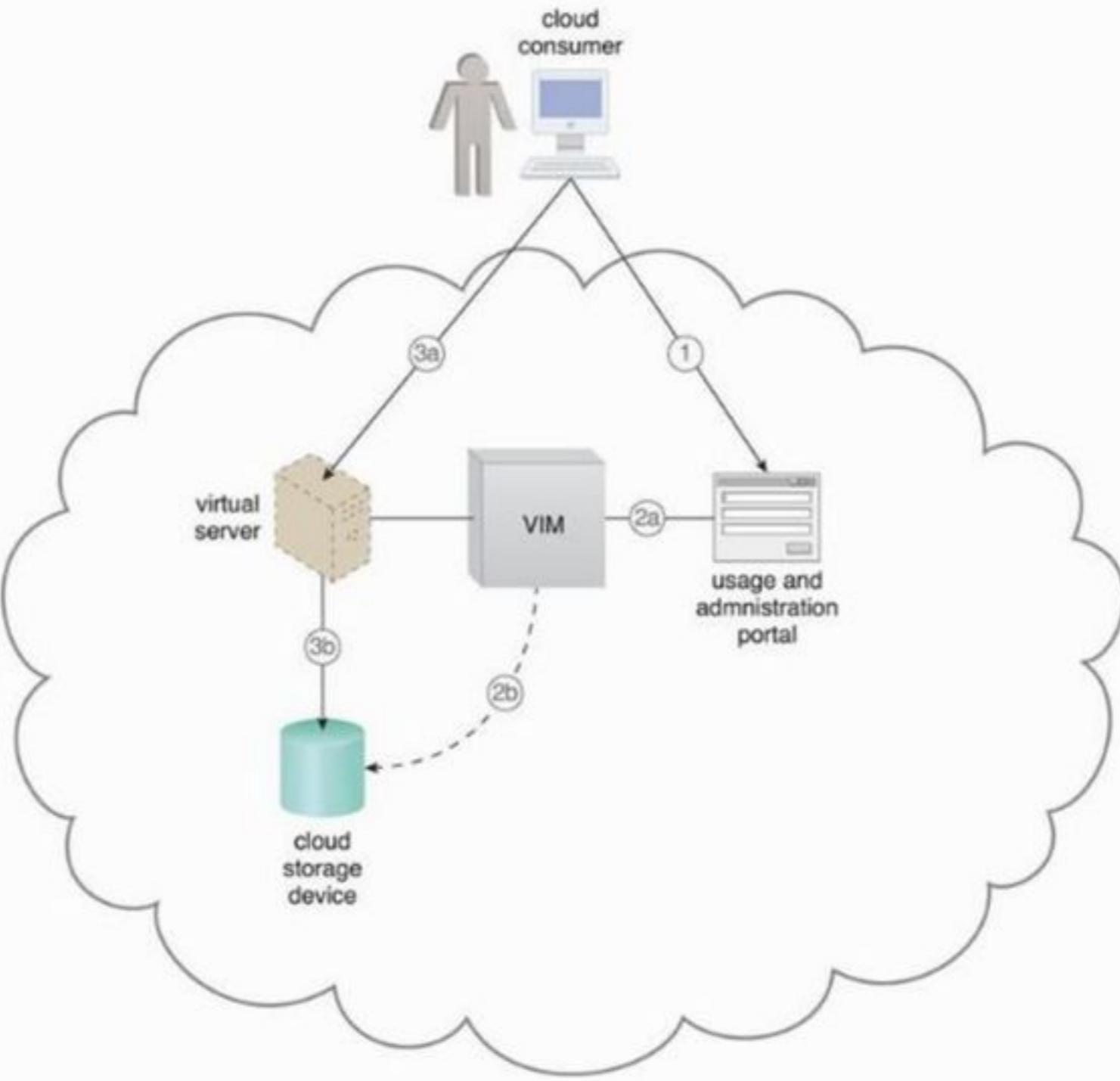
- Network Storage Interfaces – Most legacy network storage falls under this category, e.g., SCSI for storage blocks, NFS for network storage.
 - Storage processing levels and thresholds for file allocation are usually determined by the file system itself (tend to be suboptimal)
- Object Storage Interfaces - Various types of data can be referenced and stored as Web resources. This is referred to as object storage.
 - REST protocol, Web service-based cloud services as examples

Technical Interfaces to Storage (2)

- Database Storage Interfaces – support a query language in addition to basic storage operations.
 - Relational Data Storage – relies on table to organize similar data into rows and columns. Use of the industry standard Structured Query Language (SQL). Examples include IBM DB2, Oracle database, Microsoft SQL and MySQL.
 - Complex relational database designs can impose higher processing overhead and latency
 - Non-relational Data Storage – aims at reducing processing overhead of relational databases.
 - Drawback – tend to not support relational database functions such as transactions or joins.

		Relational	Non-Relational
Analytics	Proprietary Storage	Amazon Redshift EMC Greenplum HP Vertica	IBM Netezza Oracle Teradata MPP
	Hadoop Storage	Cloudera Impala Presto	Hive SQL-on-Hadoop
Operational	Proprietary Storage	Traditional SQL	NewSQL
	Hadoop Storage		Splice Machine On-Hadoop
		<u>Traditional SQL</u> Oracle DB2 SQL Server MySQL	
		<u>NewSQL</u> <u>User-Sharded MySQL</u> NuoDB Clustrix On-Disk	
		<u>NoSQL</u> <u>Key Value:</u> Aerospike, Riak	
		<u>Column Family:</u> Cassandra	
		<u>Document:</u> MongoDB	
		<u>Graph:</u> Neo4j, InfiniteGraph	
		<u>Column Family:</u> HBase	

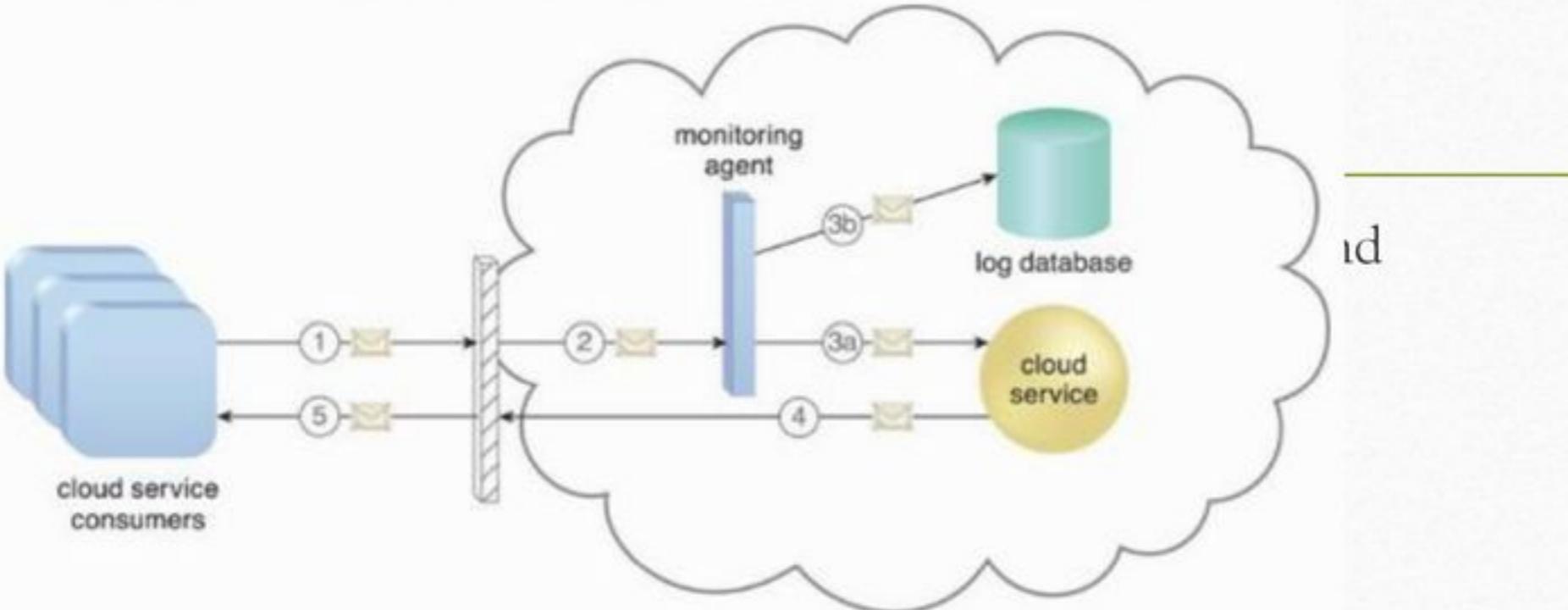




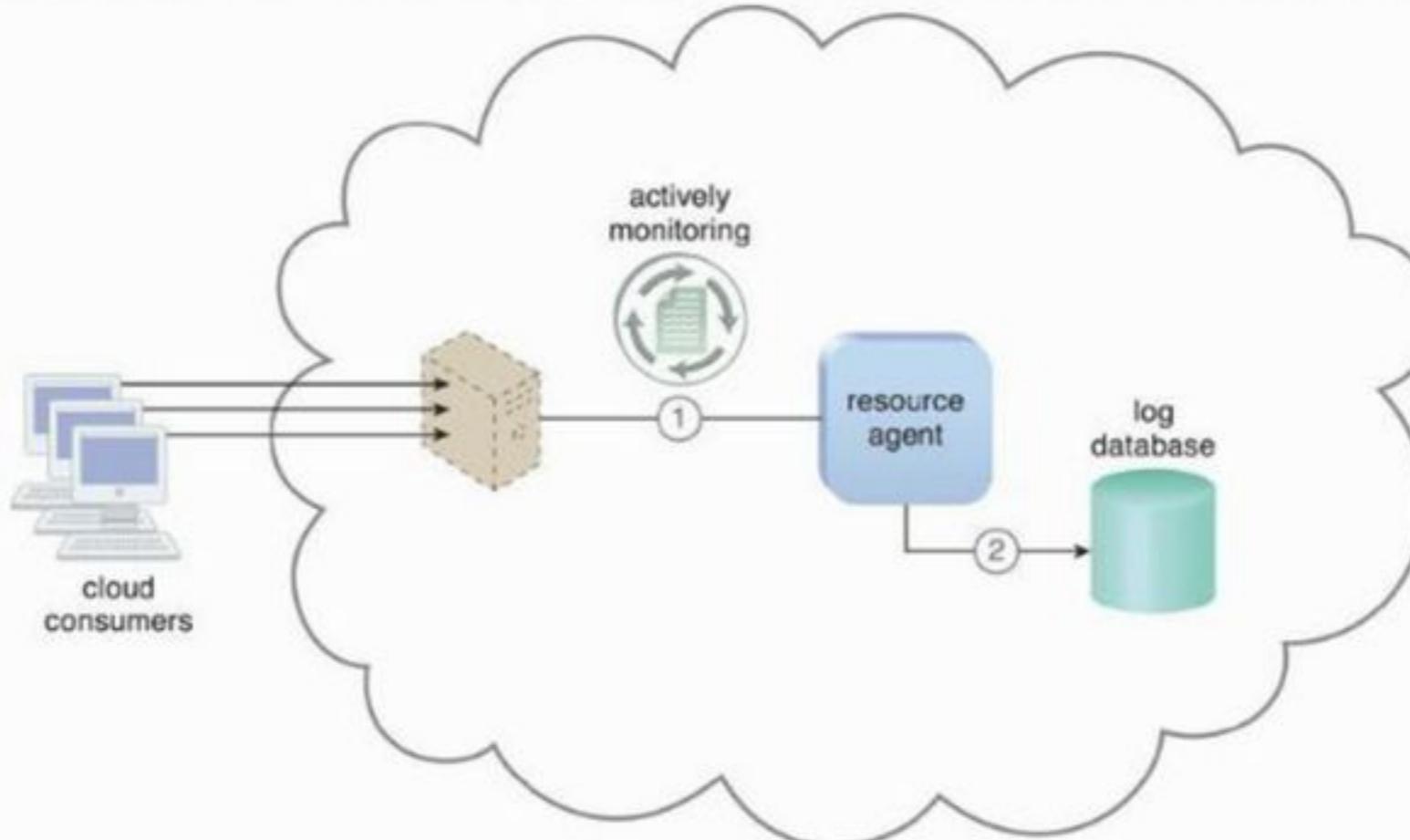
Cloud Usage Monitor Mechanism

- A lightweight and autonomous software program responsible for collecting and processing IT resource usage data.
- Metrics – amount of data, number of transactions, usage time, etc.
- Three common agent-based implementation formats:
 - Monitoring agent
 - Resource agent
 - Polling agent

- A serv
- analyz
- Measu



- Even-
observ
resumi

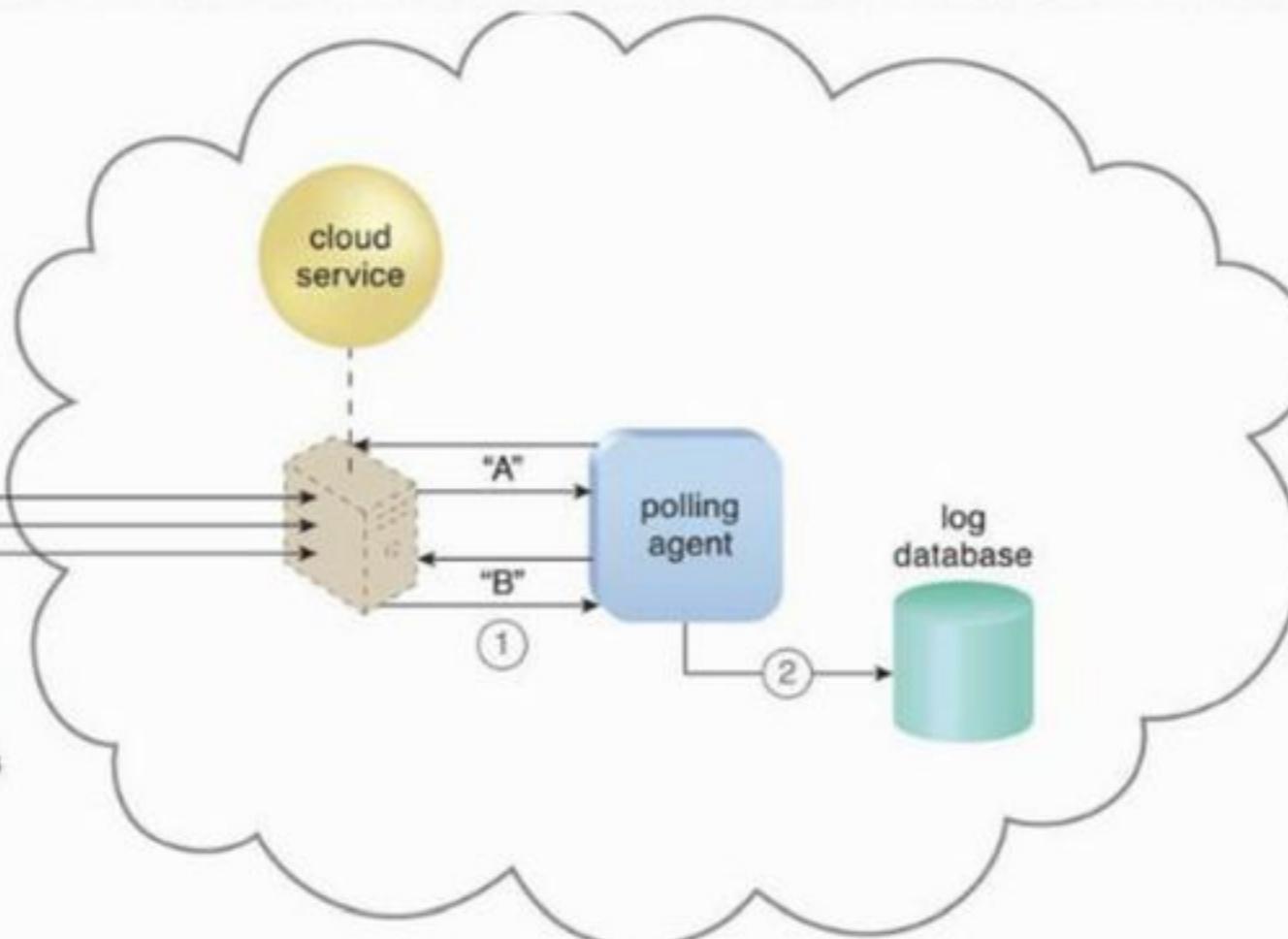


ig,

- A process resource
- Common and do



cloud consumers



ing IT
uptime

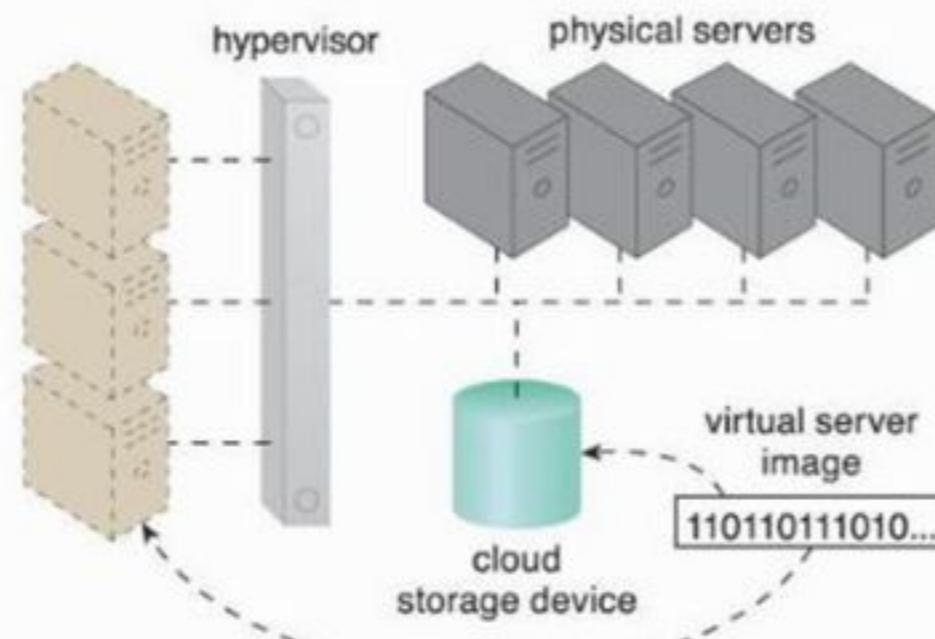
Case Study (DTGOV) Continued

- Needs to define a model that allows virtual servers of varying performance levels to be leased and billed hourly.
- Each resource usage event that is generated by VIM contains the following data:
 - Event Type (starting, started, scaled, stopping, stopped), VM Type – pre-defined VM configurations, VM ID, Cloud Consumer ID, Timestamp.
 - Usage measurements – for every VM, a measurement period (in a scale of minute usage).
 - VM can be started, scaled and stopped multiple times (e.g., started and scaled, or scaled and scaled).

$$U_{\text{total_VM_type_j}} = \sum_{t_{\text{start}}}^{t_{\text{end}}} T_{\text{cycle}_i}$$

- The cr
- Replicat

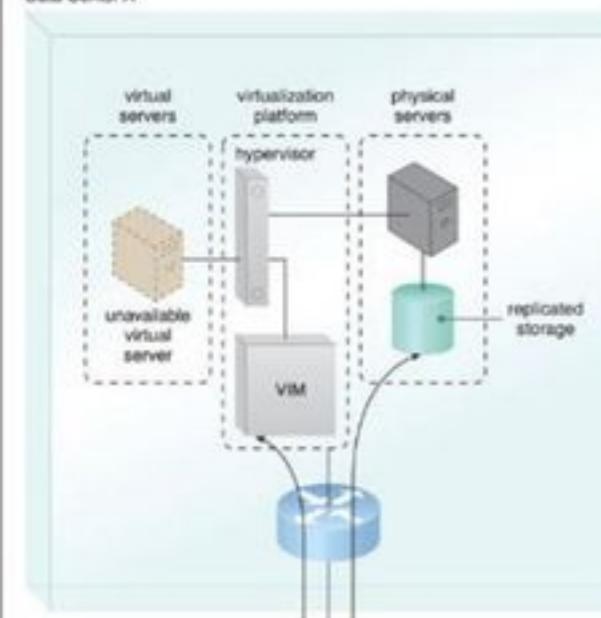
and



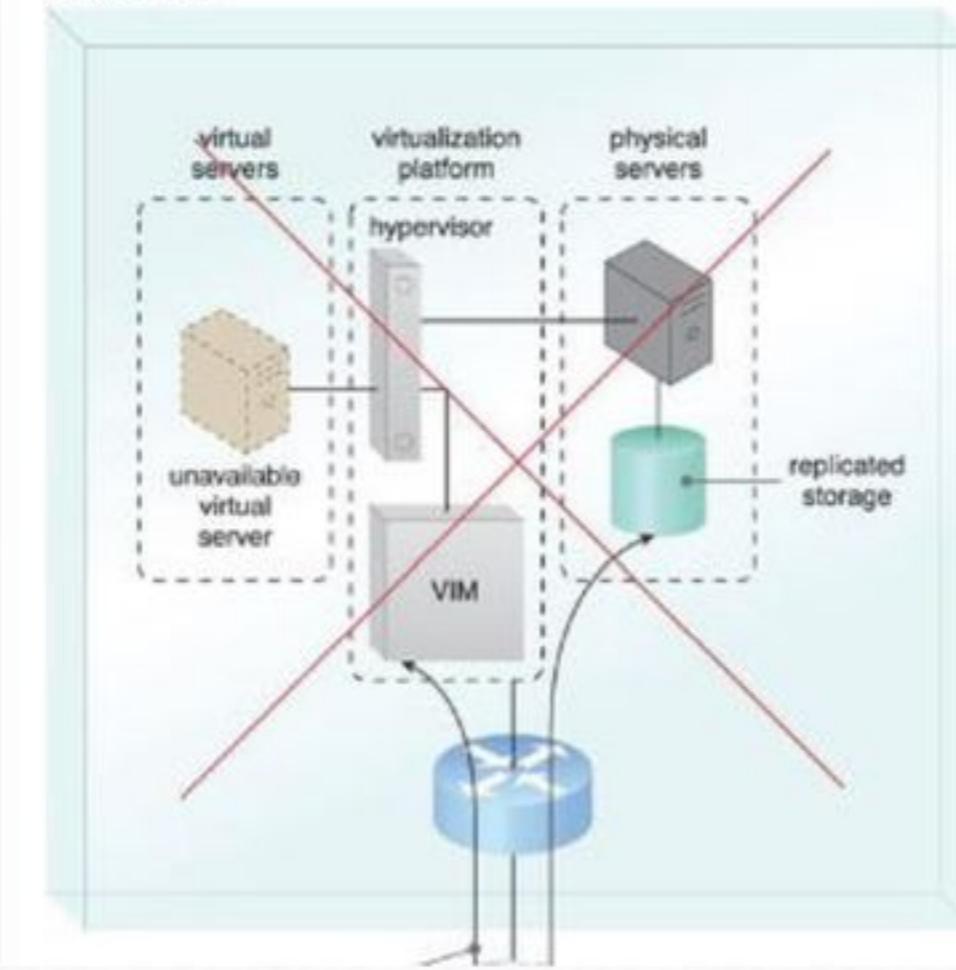
Case Study (DTGOV) Continued.

- 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.

Data Center A



Data Center A



Data Center B

