

Cloud Computing & its Applications

Course Code: SWE4004

Dr Sunil Kumar Singh

Assistant Professor

School - SCOPE

VIT-AP University

sunil.singh@vitap.ac.in

Cabin - AB2 (124D)



Cloud Delivery Model Considerations

Outline

The cloud provider perspective-

- **Building IaaS environments,**
- **Equipping PaaS environments,**
- **Optimizing SaaS environments,**

The cloud consumer perspective-

- **Working with IaaS environments,**
- **Working with PaaS environments,**
- **Working with SaaS services.**

Objectives

- Cloud Delivery Models: The Cloud Provider Perspective
 - Building IaaS Environments
 - Equipping PaaS Environments
 - Optimizing SaaS Environments
- Cloud Delivery Models: The Cloud Consumer Perspective
 - Working with IaaS Environments
 - Working with PaaS Environments
 - Working with SaaS Services
- Case Study Example

Cloud Delivery Models: The Cloud Provider Perspective

Cloud Delivery Models: The Cloud Provider Perspective

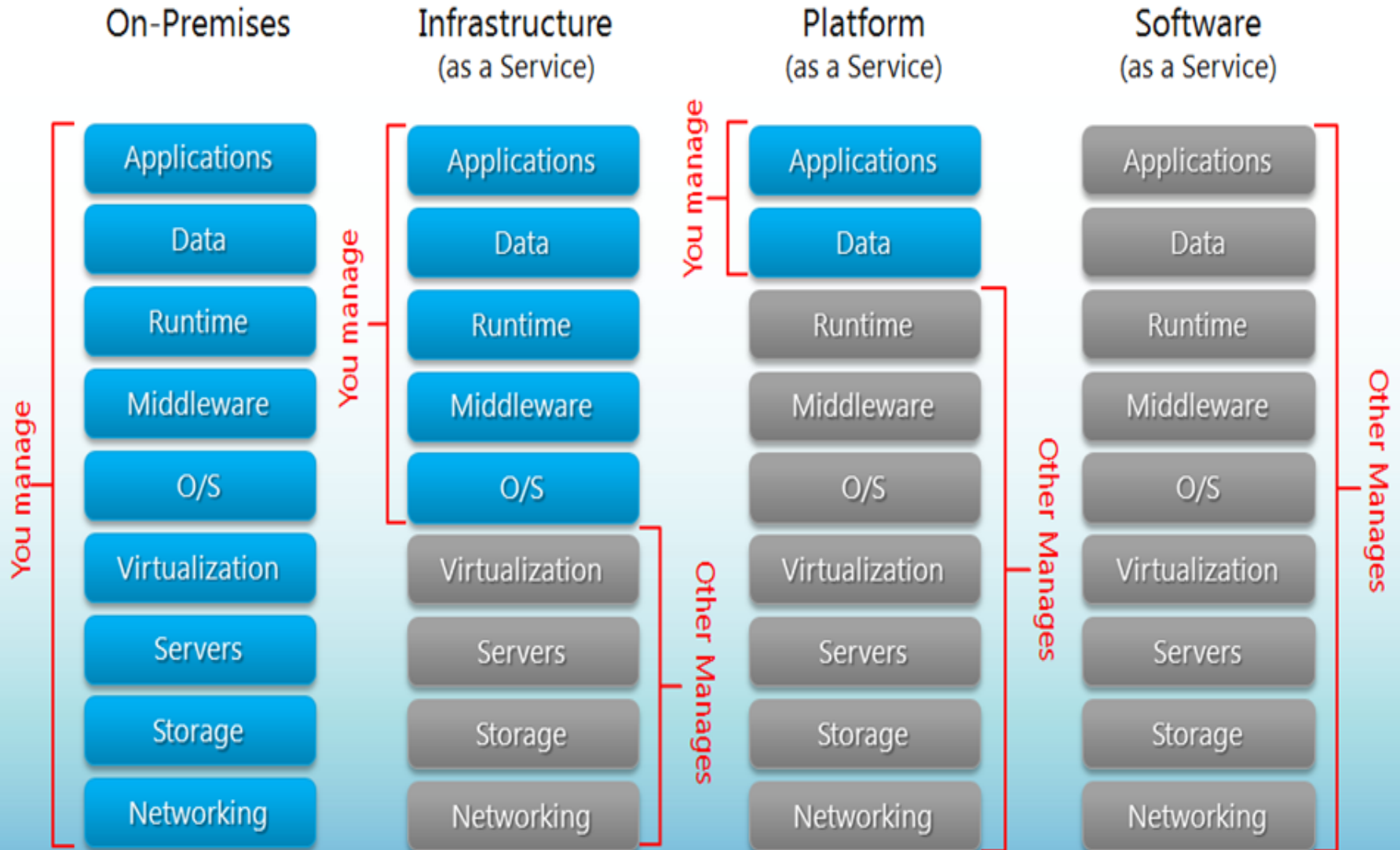
DAY-1

- Building IaaS Environments
 - ♦ Scalability and Reliability
 - ♦ Monitoring
 - ♦ security




Day-2

- Equipping PaaS Environments
 - ♦ Scalability and Reliability
 - ♦ Monitoring
 - ♦ Security
- Optimizing SaaS Environments
 - ♦ Architectural models
 - ♦ Online SaaS offerings
 - ♦ Security

Separation of Responsibilities



Building IaaS Environments

Service Model	Main Access & Management Tool	Service content
 SAAS Consume	Web Browser	Cloud Applications Social networks, Office suites, CRM, Video processing
 PAAS Build On it	Cloud Development Environment	Cloud Platform Programming languages, Frameworks, PaaS Mashups editors, structured data
 IAAS Migrate On it	Virtual Infrastructure Manager	Cloud Infrastructure Compute Servers, Data Storage, Firewall, Load Balancer

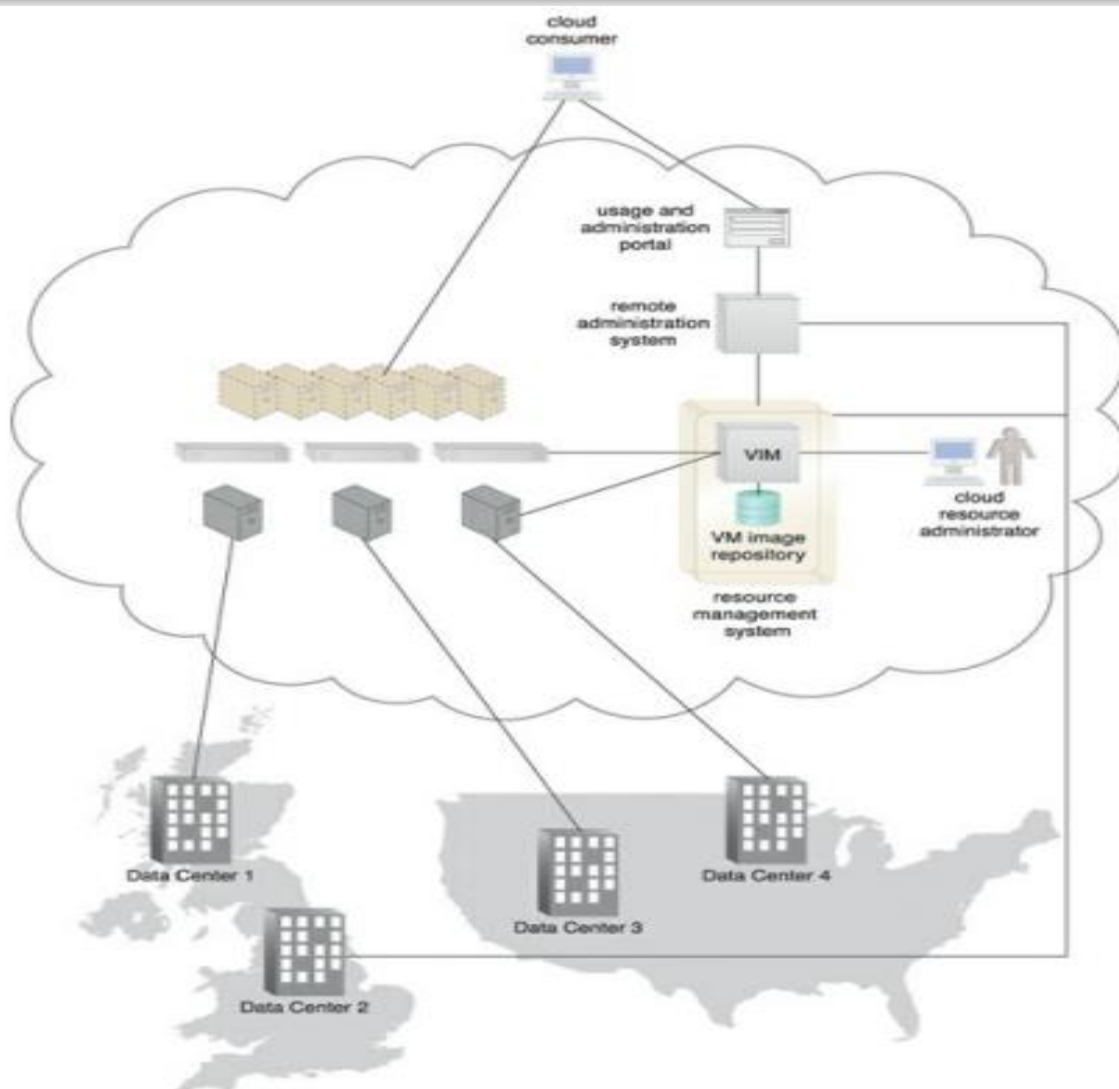
Building IaaS Environments

- Preemptively assembled by cloud providers
- Cloud consumers direct administrative access to physical IT resources
- Snapshots can be taken of a virtual server

Data Centers

- Multiple data centers can increase resiliency
- Connected through high-speed communications networks with low latency
- Data centers that are deployed in different countries(legal and regulatory requirements)

Data Centers



- A cloud provider provisioning and managing an IaaS environment with IT resources from different data centres in the United States and the United Kingdom

Scalability and Reliability

- Resource pooling architecture
- Workload distribution architecture
- Multipath resource access architecture

Monitor

- Virtual server lifecycles
- Data Storage
- Network traffic
- Failure Conditions
- Event triggers

Security

- Encryption, Integrity, PKI, digital signature
- Identity and access management (IAM) and **Single Sign-on** (SSO) mechanisms
- Cloud-based security groups
- Cloud usage monitors

Equipping PaaS Environment

- Readymade Environment
- Programming languages and programming modes, framework, editors
- Matching SDK and IDE
- Testing
- Penetration testing
- Customizing PaaS
- Multi Tenancy

Scalability and Reliability

- Dynamic Scaling
- Workload Distribution
- Non-Disruptive service relocation
- Multiple Data centers

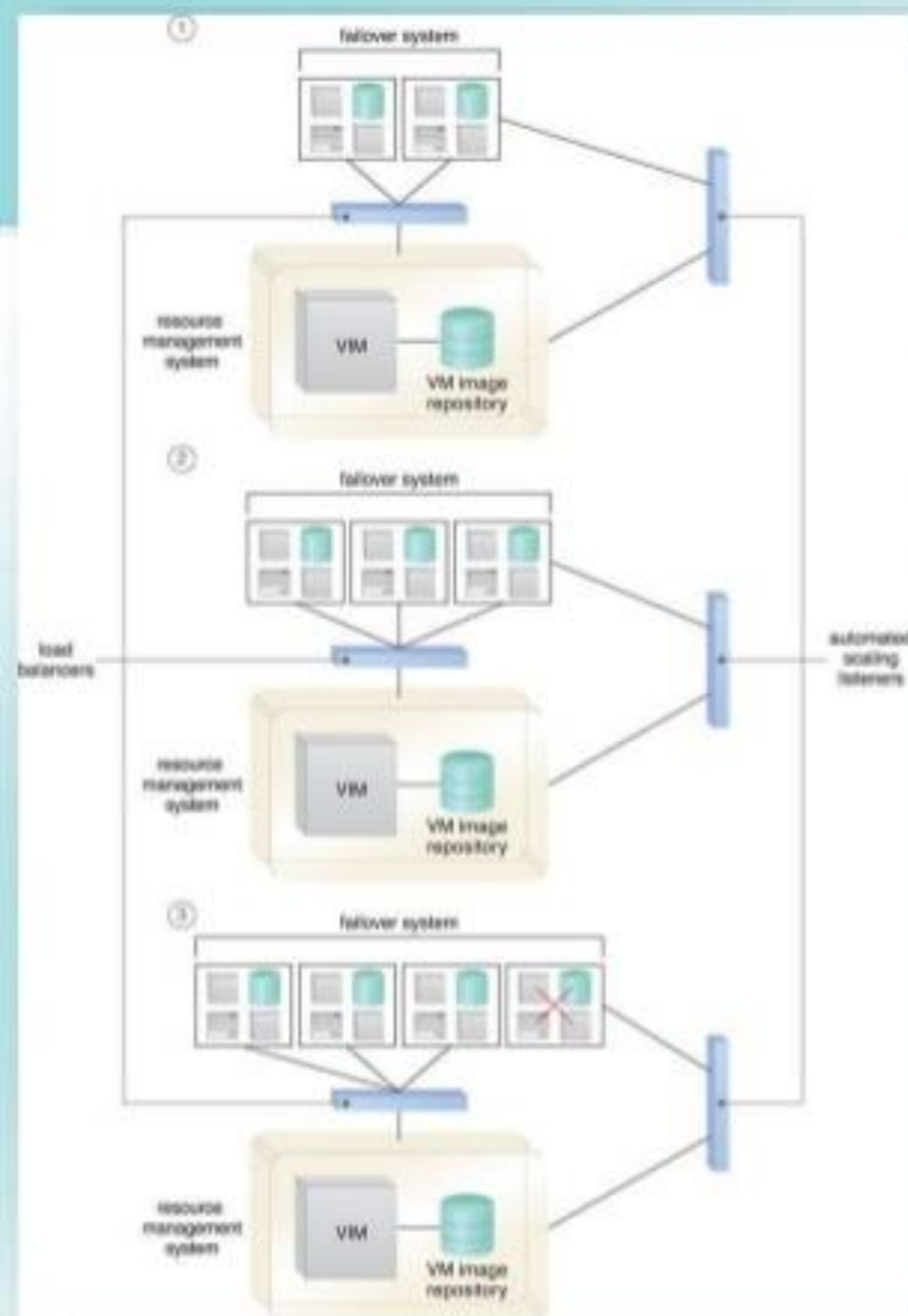


Figure 14.2. Load balancers are used to distribute ready-made environment instances that are part of a failover system, while automated scaling listeners are used to monitor the network and instance workloads,

(1). The ready-made environments are scaled out in response to an increase in workload,

(2), and the failover system detects a failure condition and stops replicating a failed ready-made environment (3).

Monitoring (pay-per-use, SLA monitor and Audit monitor)

- Readymade environment Instances
- Data persistence
- Network usage
- Failure conditions
- Event Triggers

Security : Already provisioned for IaaS

Optimizing SaaS Environments

- Multitenant environments
- IT resource segregation is not possible

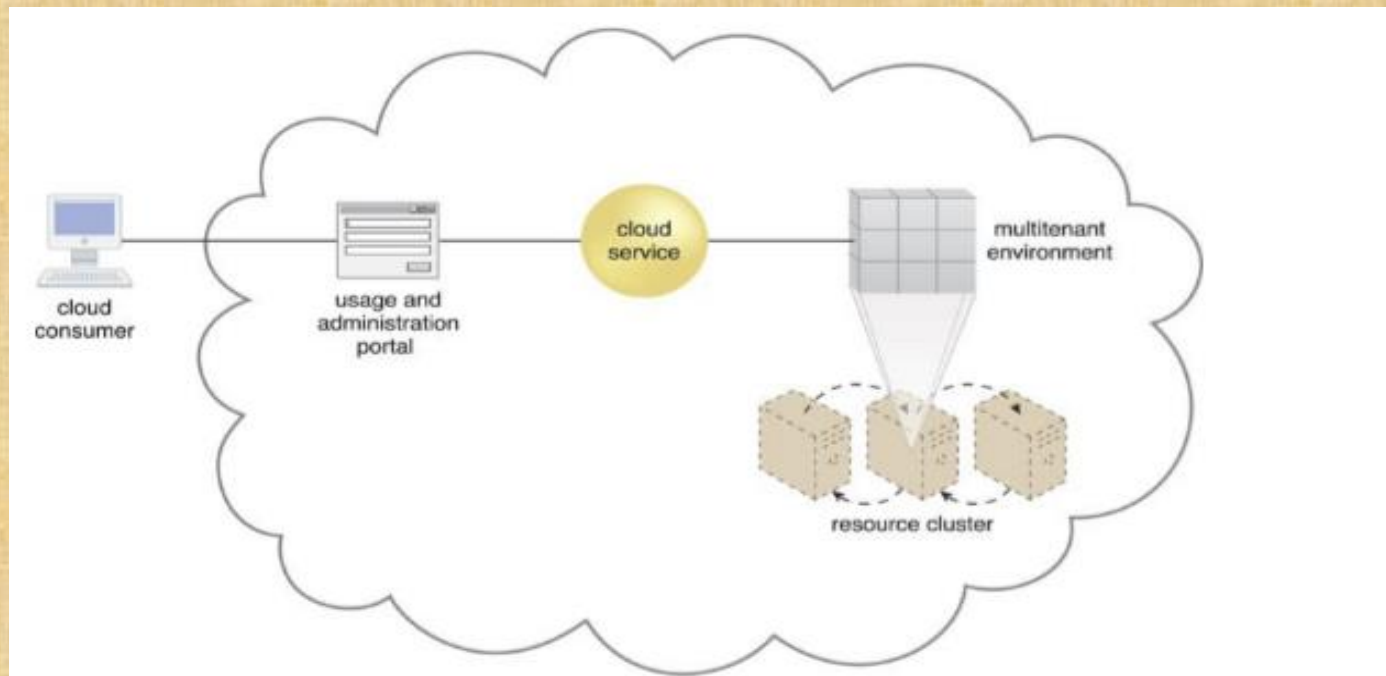


Figure 14.3. The SaaS-based cloud service is hosted by a multitenant environment deployed in a high-performance virtual server cluster. A usage and administration portal is used by the cloud consumer to access and configure the cloud service.

Cont..

Failover Condition

- dynamic scalability
- workload distribution
- non-disruptive service relocation

Cont..

Each SaaS deployment differs

- Architectural
- Functional
- Runtime requirements

Diversity in functionality and usage of recognized online SaaS offerings

- collaborative authoring and information-sharing (Wikipedia, Blogger)
- collaborative management (Zimbra, Google Apps)
- conferencing services for instant messaging, audio/video communications (Skype, Google Talk)
- enterprise management systems (ERP, CRM, CM)
- file-sharing and content distribution (YouTube, Dropbox)

Cont..

- industry-specific software (engineering, bioinformatics)
- messaging systems (e-mail, voicemail)
- mobile application marketplaces (Android Play Store, Apple App Store)
- office productivity software suites (Microsoft Office, Adobe Creative Cloud)
- search engines (Google, Yahoo)
- social networking media (Twitter, LinkedIn)

Cont..

SaaS implementation medium

- mobile application
- REST service
- Web service

SaaS-based cloud services with Web-based APIs include:

- electronic payment services (PayPal)
- mapping and routing services (Google Maps)
- publishing tools (WordPress)

Architectural Models Requirements in SaaS

- Service load balancing
- Dynamic failure detection and recovery
- Storage Maintenance window
- Elastic Resource capacity/elastic network capacity
- Cloud Balancing

Cloud Usage Monitor

Types of Metrics

- Tenant Subscription period
- Application Usage
- Tenant Application Functional Module

Cloud Delivery Models: The Cloud Consumer Perspective

This section raises various considerations concerning the different ways in which cloud delivery models are administered and utilized by cloud consumers.

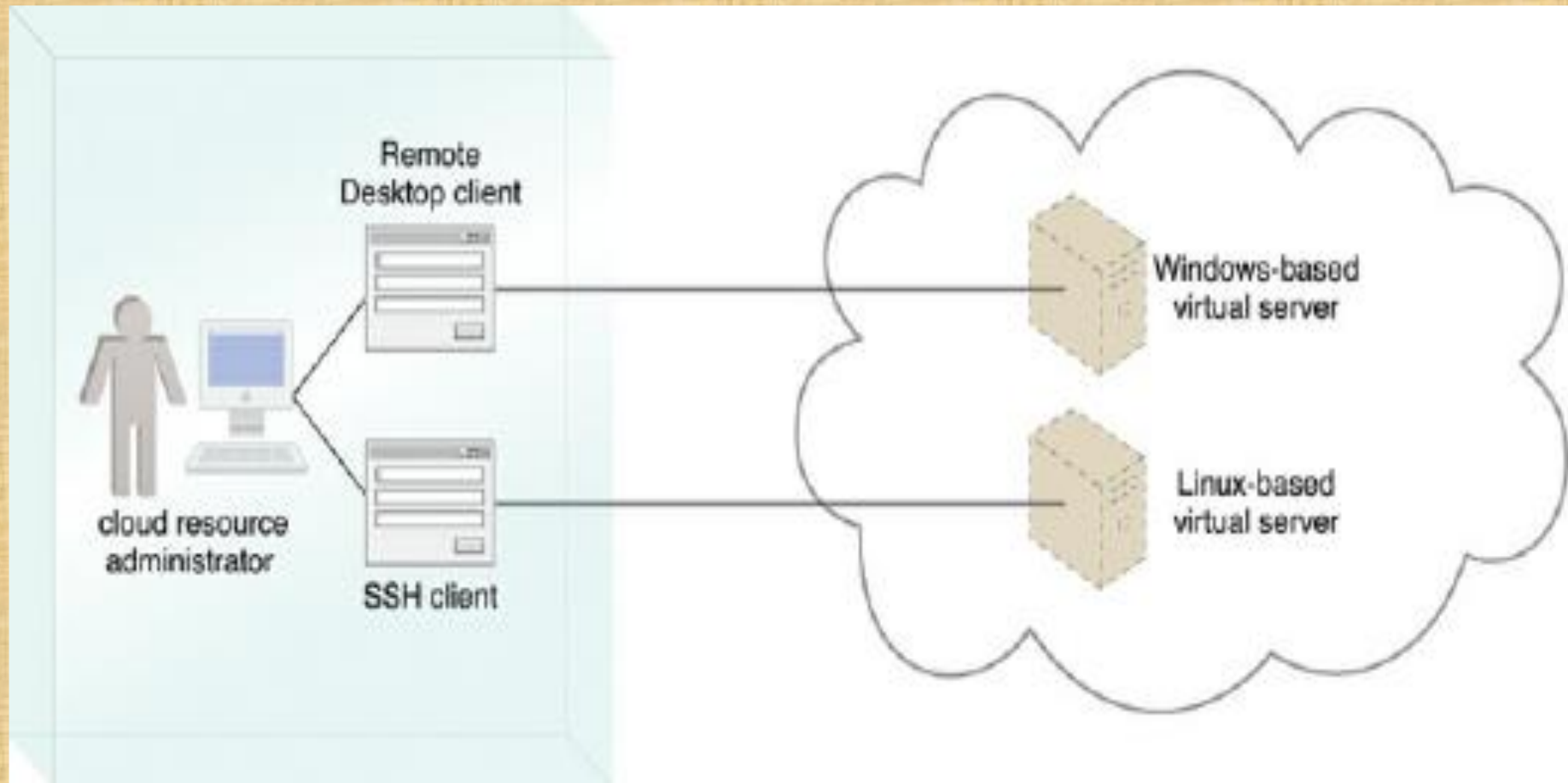
Cloud Consumer ?

- The cloud consumer is the **principal stakeholder** for the cloud computing service. A cloud consumer represents a person or organization that maintains a business relationship with, and uses the service from a cloud provider.
- The cloud consumer may be **billed for the service provisioned**, and needs to arrange payments accordingly.
- Cloud consumers **need SLAs to specify the technical performance** requirements fulfilled by a cloud provider. SLAs can cover terms regarding the quality of service, security, remedies for performance failures.

Working with IaaS Environments

- **Virtual servers** are accessed at the operating system level through the use of remote terminal applications. Accordingly, the type of client software used directly depends on the type of operating system that is running at the virtual server, of which two common options are:
- **Remote Desktop** (or Remote Desktop Connection) Client – for Windows-based environments and presents a Windows GUI desktop.
- **SSH Client** – for Mac and other Linux-based environments to allow for secure channel connections to text-based shell accounts running on the server OS

Typical usage scenario for virtual servers at are being offered as IaaS services



Virtual Server

- A cloud storage device can be attached directly to the virtual servers and accessed through the virtual servers' functional interface for management by the operating system.
- Alternatively, a cloud storage device can be attached to an IT resource that is being hosted outside of the cloud, such as an on-premise device over a WAN or VPN.
- In these cases, the following formats for the manipulation and transmission of cloud storage data are commonly used:

Virtual Server (Cont..)

- **Networked File System** – **System-based storage access**, whose rendering of files is similar to how folders are organized in operating systems (NFS, CIFS)
- **Storage Area Network Devices** – **Block-based storage access** collates and formats geographically diverse data into cohesive files for optimal network transmission (iSCSI, Fibre Channel)
- **Web-Based Resources** – **Object-based storage access** by which an interface that is not integrated into the operating system logically represents files, which can be accessed through a Web-based interface (Amazon S3)

Block v/s File v/s Object storage

➤ Interface

➤ Cost

➤ Performance

➤ Proximity

➤ Use Case

➤ Scale

Block

Operating System

\$\$-\$\$\$



Dedicated Network
Fibre Channel / 10Gb

OS, Database



File

User

\$\$ - \$\$\$



LAN / 10Gb

Sharing user data,
web content



Object

Program (API)

\$

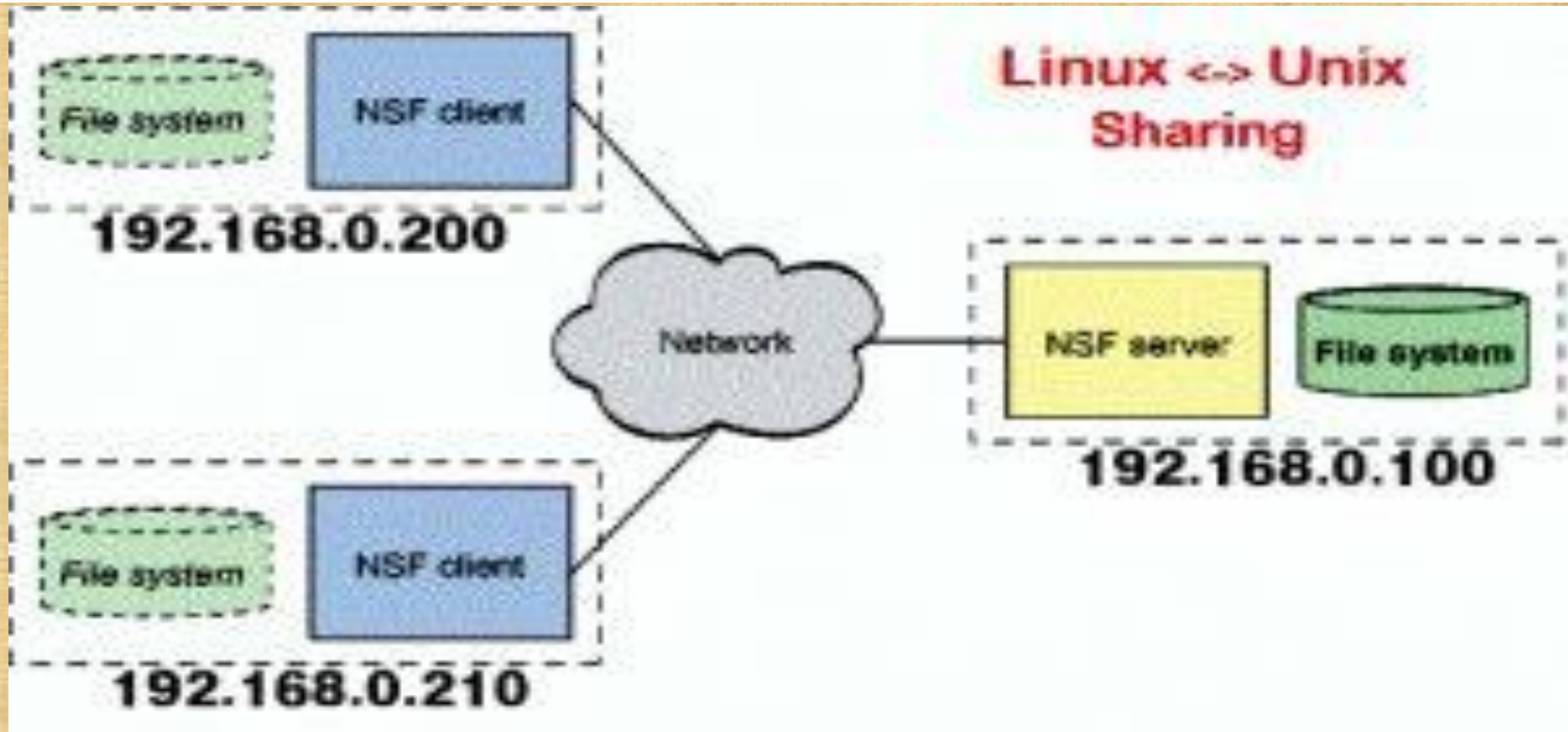


Internet

Images, PDFs, Video



Network File System



IT Resource Provisioning Considerations

- Cloud consumers have a high degree of control over how and to what extent IT resources are provisioned as part of their IaaS environments.
- *For example:*
- controlling scalability features (automated scaling, load balancing)
- controlling the lifecycle of virtual IT resources (shutting down, restarting, powering up of virtual devices)

IT Resource Provisioning Considerations

- controlling the virtual network environment and network access rules (firewalls, logical network perimeters).
- establishing and displaying service provisioning agreements (account conditions, usage terms)
- managing the attachment of cloud storage devices
- managing the pre-allocation of cloud-based IT resources (resource reservation)
- managing credentials and passwords for cloud resource administrators

IT Resource Provisioning Considerations

- managing credentials for cloud-based security groups that access virtualized IT resources through an IAM
- managing security-related configurations
- managing customized virtual server image storage (importing, exporting, backup)
- selecting high-availability options (failover, IT resource clustering)
- selecting and monitoring SLA metrics

IT Resource Provisioning Considerations

- selecting basic software configurations (operating system, pre-installed software for new virtual servers)
- selecting IaaS resource instances from a number of available hardware-related configurations and options (processing capabilities, RAM, storage)
- selecting the geographical regions in which cloud-based IT resources should be hosted
- tracking and managing costs

IT Resource Provisioning Considerations

- The management interface for these types of provisioning tasks is usually a **usage and administration portal**, but may also be offered via the use of **command line interface (CLI)** tools that can simplify the execution of many scripted administrative actions.
- Even though standardizing the presentation of administrative features and controls is typically preferred, using different tools and user-interfaces **can sometimes be justified**. For example, a script can be made to turn virtual servers on and off nightly through a CLI, while adding or removing storage capacity can be more easily carried out using a portal.

Summary

- Optimizing SaaS –Offering, Monitoring Usage and Security
- Consumer Perspective:
 - Working with IaaS
 - *IT Resource Provisioning Considerations*

Reference

- <https://timesofcloud.com/cloud-tutorial/cloud-providers/>

Working with PaaS Environments

- A typical PaaS IDE can offer a wide range of tools and programming resources, such as software libraries, class libraries, frameworks, APIs, and various runtime capabilities that emulate the intended cloud-based deployment environment.
- These features allow developers to create, test, and run application code within the cloud or locally (on-premise) while using the IDE to emulate the cloud deployment environment.
- Compiled or completed applications are then bundled and uploaded to the cloud, and deployed via the ready-made environments. This deployment process can also

Working with PaaS Environments

- PaaS also allows for applications to use cloud storage devices as independent data storing systems for holding development-specific data (for example in a repository that is available outside of the cloud environment).
- Both SQL and NoSQL database structures are generally supported.

IT Resource Provisioning Considerations

- PaaS environments provide **less administrative control than IaaS** environments, but still offer a significant range of management features.
- *For example:*
- establishing and displaying service provisioning agreements, such as account conditions and usage terms
- selecting software platform and development frameworks for ready-made environments
- selecting instance types, which are most common frontend or backend instances

IT Resource Provisioning Considerations

- selecting cloud storage devices for use in ready-made environments
- controlling the lifecycle of PaaS-developed applications (deployment, starting, shutdown, restarting, and release)
- controlling the versioning of deployed applications and modules
- configuring availability and reliability-related mechanisms
- managing credentials for developers and cloud resource administrators using IAM

IT Resource Provisioning Considerations

- managing general security settings, such as accessible network ports
- selecting and monitoring PaaS-related SLA metrics
- managing and monitoring usage and IT resource costs
- controlling scalability features such as usage quotas, active instance thresholds, and the configuration and deployment of the automated scaling listener and load balancer mechanisms

- The usage and administration portal that is used to access PaaS management features can provide the feature of pre-emptively selecting the times at which an IT resource is started and stopped.
- For example, a cloud resource administrator can set a cloud storage device to turn itself on at 9:00AM then turn off twelve hours later. Building on this system can enable the option of having the ready-made environment self-activate upon receiving data requests for a particular application and turn off after an extended period of inactivity.

Working with SaaS Environments

- Because SaaS-based cloud services are almost always **accompanied by refined and generic APIs**, they are usually designed to be incorporated as part of larger distributed solutions.
- A common example of this is **Google Maps**, which offers a comprehensive API that enables mapping information and images to be incorporated into Web sites and Web-based applications.
- Many SaaS offerings are provided **free of charge**, although these cloud services often come with data collecting sub-programs that harvest usage data for the benefit of the cloud provider.
- When using any SaaS product that is **sponsored by third parties**, there is a reasonable chance that it is performing a **form of background information gathering**. Reading the cloud provider's **agreement will** usually help shed light on any secondary activity that the cloud service is designed to perform.

Working with SaaS Environments

- Customization options are usually available to cloud consumers; however, these options are generally limited to the runtime usage control of the cloud service instances that are generated specifically by and for the cloud consumer.

For example:

- managing security-related configurations
- managing select availability and reliability options
- managing usage costs
- managing user accounts, profiles, and access authorization
- selecting and monitoring SLAs
- setting manual and automated scalability options and limitations

14.3. Case Study Example

DTGOV discovers that a number of additional mechanisms and technologies need to be assembled in order to complete its IaaS management architecture (Figure 14.5):

- Network virtualization is incorporated into logical network topologies, and logical network perimeters are established using different firewalls and virtual networks.
- The VIM is positioned as the central tool for controlling the IaaS platform and equipping it with self-provisioning capabilities.
- Additional virtual server and cloud storage device mechanisms are implemented through the virtualization platform, while several virtual server images that provide base template configurations for virtual servers are created.

- Dynamic scaling is added using the VIM's API through the use of automated scaling listeners.
- High-availability virtual server clusters are created using the resource replication, load balancer, failover system, and resource cluster mechanisms.
- A customized application that directly uses the SSO and IAM system mechanisms is built to enable interoperability between the remote administration system, network management tools, and VIM.

DTGOV uses a powerful commercial network management tool that is customized to store event information gathered by the VIM and SLA monitoring agents in an SLA measurements database. The management tool and database are used as part of a greater SLA management system. In order to enable billing processing, DTGOV expands a proprietary software tool that is based on a set of usage measurements from a database populated by pay-per-use monitors. The billing software is used as the base implementation for the billing management system mechanism.

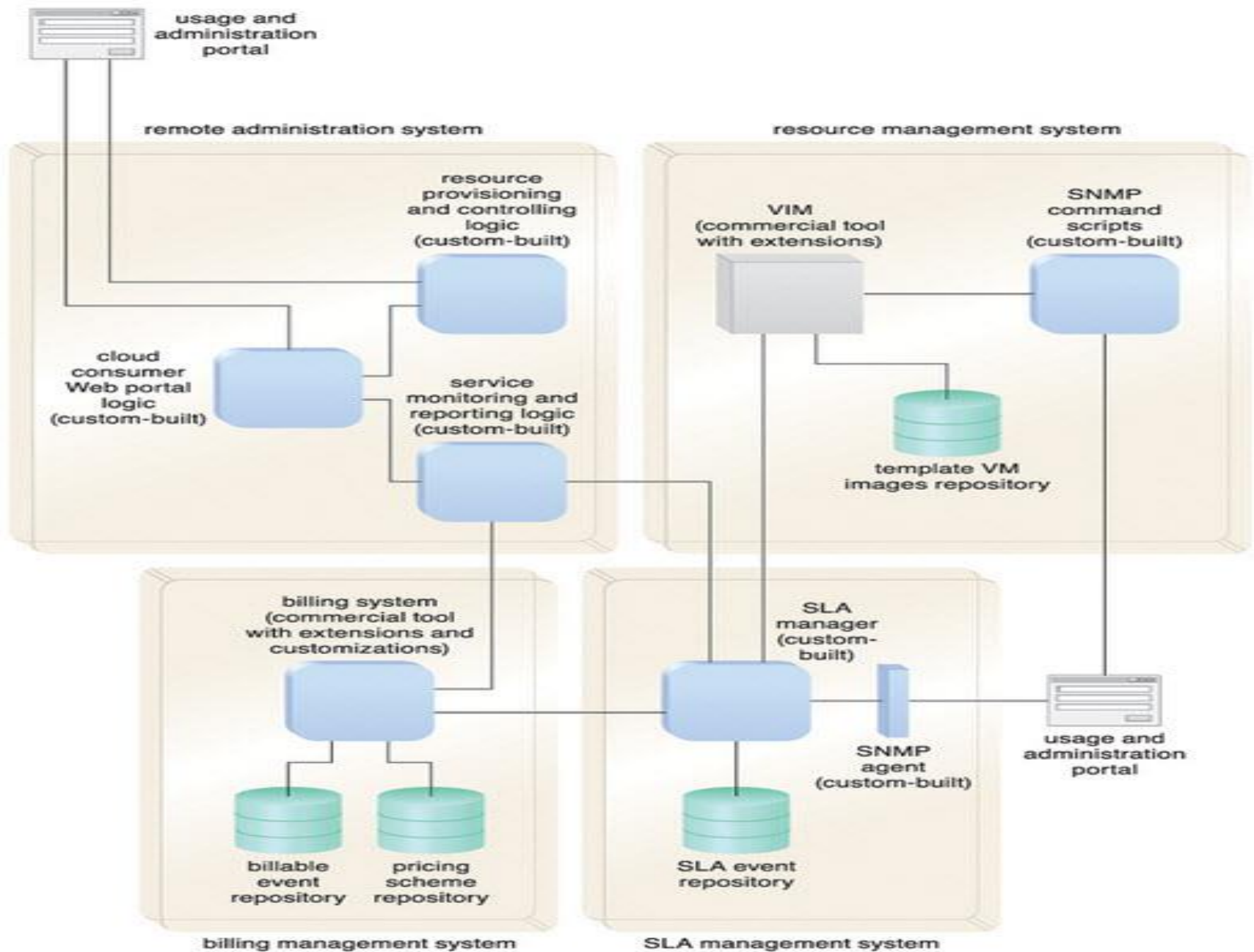


Figure 14.5 An overview of the DTGOV management architecture.

Conclusion

- Building and Launching your SaaS App

Exam: Shopify App store