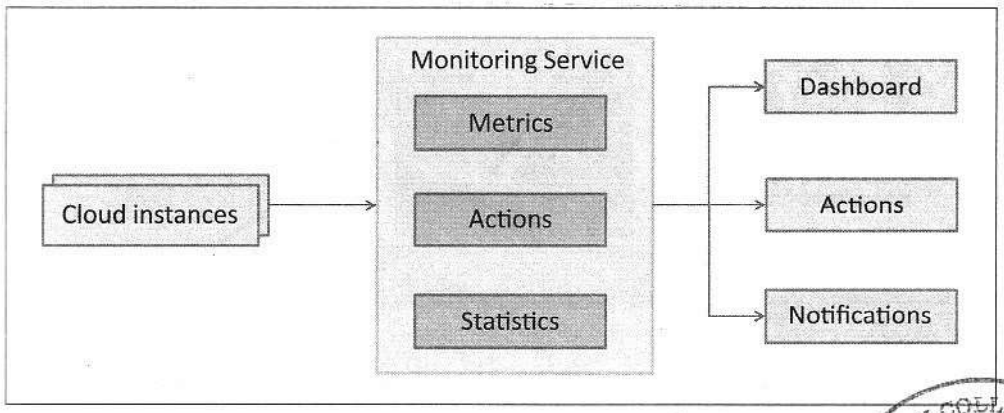


Figure 2.10: Typical cloud monitoring service architecture

Type	Metrics
CPU	CPU-Usage, CPU-Idle
Disk	Disk-Usage, Bytes/sec (read/write), Operations/sec
Memory	Memory-Used, Memory-Free, Page-Cache
Interface	Packets/sec (incoming/outgoing), Octets/sec(incoming/outgoing)

Table 2.4: Typical monitoring metrics

2.7 Software Defined Networking

Software-Defined Networking (SDN) is a networking architecture that separates the control plane from the data plane and centralizes the network controller. Figure 2.11 shows the conventional network architecture built with specialized hardware (switches, routers, etc.). Network devices in conventional network architectures are getting exceedingly complex with the increasing number of distributed protocols being implemented and the use of proprietary hardware and interfaces. In the conventional network architecture the control plane and data plane are coupled. Control plane is the part of the network that carries the signaling and routing message traffic while the data plane is the part of the network that carries the payload data traffic.

The limitations of the conventional network architectures are as follows:

- **Complex Network Devices**: Conventional networks are getting increasingly complex with more and more protocols being implemented to improve link speeds and reliability. Interoperability is limited due to the lack of standard and open interfaces. Network devices use proprietary hardware and software and have slow product lifecycles limiting innovation. The conventional networks were well suited for static traffic patterns and had a large number of protocols designed for specific applications. With the emergence of cloud computing and proliferation of internet access devices, the traffic patterns are becoming more and more dynamic. Due to the complexity of conventional network devices, making changes in the networks to meet the dynamic traffic patterns has

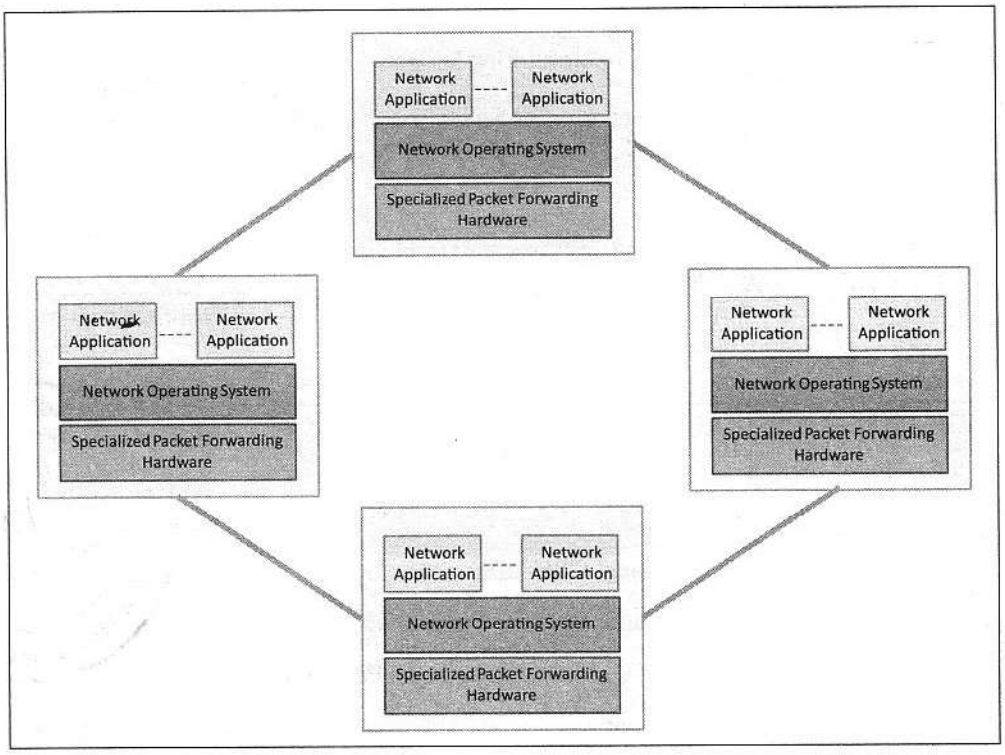


Figure 2.11: Conventional network architecture

become increasingly difficult.

- **Management Overhead**: Conventional networks involve significant management overhead. Network managers find it increasingly difficult to manage multiple network devices and interfaces from multiple vendors. Upgradation of network requires configuration changes in multiple devices (switches, routers, firewalls, etc.)
- **Limited Scalability**: The virtualization technologies used in cloud computing environments has increased the number of virtual hosts requiring network access. Multi-tenanted applications hosted in the cloud are distributed across multiple virtual machines that require exchange of traffic. Big data applications run distributed algorithms on a large number of virtual machines that require huge amounts of data exchange between virtual machines. Such computing environments require highly scalable and easy to manage network architectures with minimal manual configurations, which is becoming increasingly difficult with conventional networks.

SDN attempts to create network architectures that are simpler, inexpensive, scalable, agile and easy to manage. Figures 2.12 and 2.13 show the SDN architecture and the SDN layers in which the control and data planes are decoupled and the network controller is centralized. Software-based SDN controllers maintain a unified view of the network and make configuration, management and provisioning simpler. The underlying infrastructure in SDN uses simple packet forwarding hardware as opposed to specialized hardware in conventional networks. The underlying network infrastructure is abstracted from the applications. Network devices become simple with SDN as they do not require implementations of a large number of

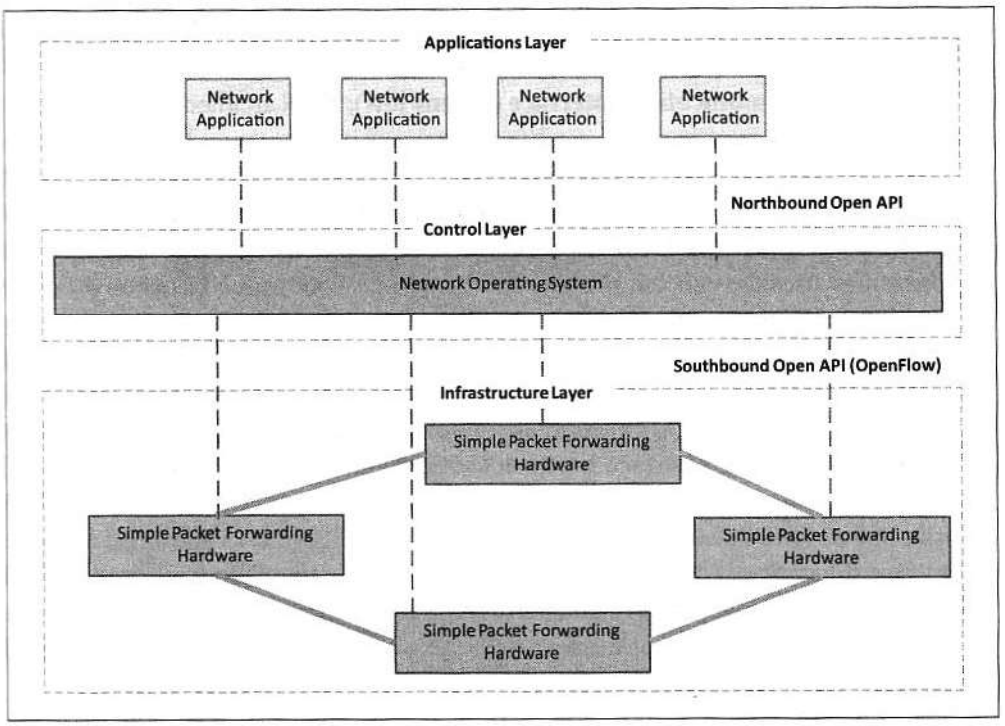


Figure 2.12: SDN architecture

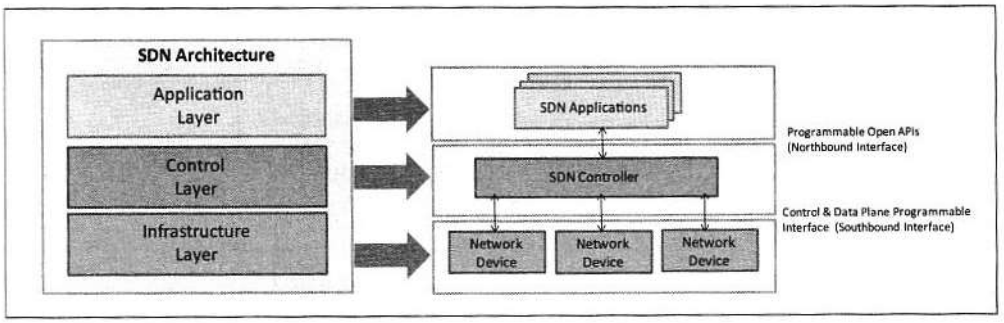


Figure 2.13: SDN layers

protocols. Network devices receive instructions from the SDN controller on how to forward the packets. These devices can be simpler and cost less as they can be built from standard hardware and software components.

Key elements of SDN are as follows:

- **Centralized Network Controller**: With decoupled the control and data planes and centralized network controller, the network administrators can rapidly configure the network. SDN applications can be deployed through programmable open APIs. This speeds up innovation as the network administrators no longer need to wait for the device vendors to embed new features in their proprietary hardware.
- **Programmable Open APIs**: SDN architecture supports programmable open APIs for interface between the SDN application and control layers (Northbound interface).

These open APIs that allow implementing various network services such as routing, quality of service (QoS), access control, etc.

- **Standard Communication Interface (OpenFlow)**: SDN architecture uses a standard communication interface between the control and infrastructure layers (Southbound interface). OpenFlow, which is defined by the Open Networking Foundation (ONF) is the broadly accepted SDN protocol for the Southbound interface. With OpenFlow, the forwarding plane of the network devices can be directly accessed and manipulated. OpenFlow uses the concept of flows to identify network traffic based on pre-defined match rules. Flows can be programmed statically or dynamically by the SDN control software. Figure 2.14 shows the components of an OpenFlow switch comprising of one or more flow tables and a group table, which perform packet lookups and forwarding, and OpenFlow channel to an external controller. OpenFlow protocol is implemented on both sides of the interface between the controller and the network devices. The controller manages the switch via the OpenFlow switch protocol. The controller can add, update, and delete flow entries in flow tables. Figure 2.15 shows an example of an OpenFlow flow table. Each flow table contains a set of flow entries. Each flow entry consists of match fields, counters, and a set of instructions to apply to matching packets. Matching starts at the first flow table and may continue to additional flow tables of the pipeline [12].

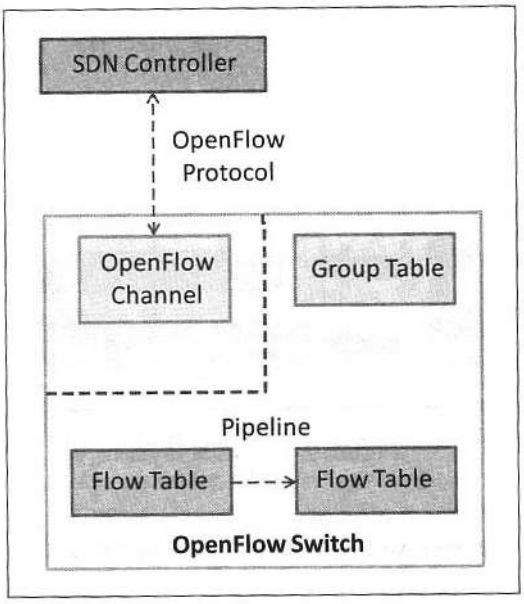


Figure 2.14: OpenFlow switch

2.8 Network Function Virtualization

Network Function Virtualization (NFV) is a technology that leverages virtualization to consolidate the heterogeneous network devices onto industry standard high volume servers, switches and storage. NFV is complementary to SDN as NFV can provide the infrastructure on which SDN can run. NFV and SDN are mutually beneficial to each other but not dependent.

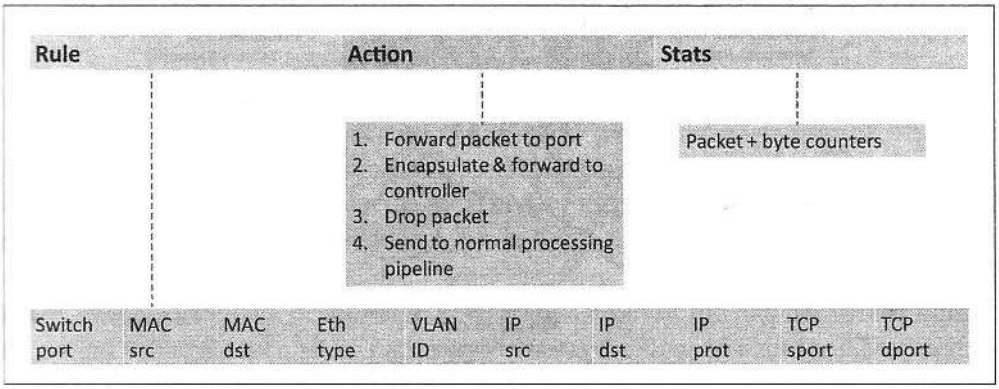


Figure 2.15: OpenFlow flow table

Network functions can be virtualized without SDN, similarly, SDN can run without NFV.

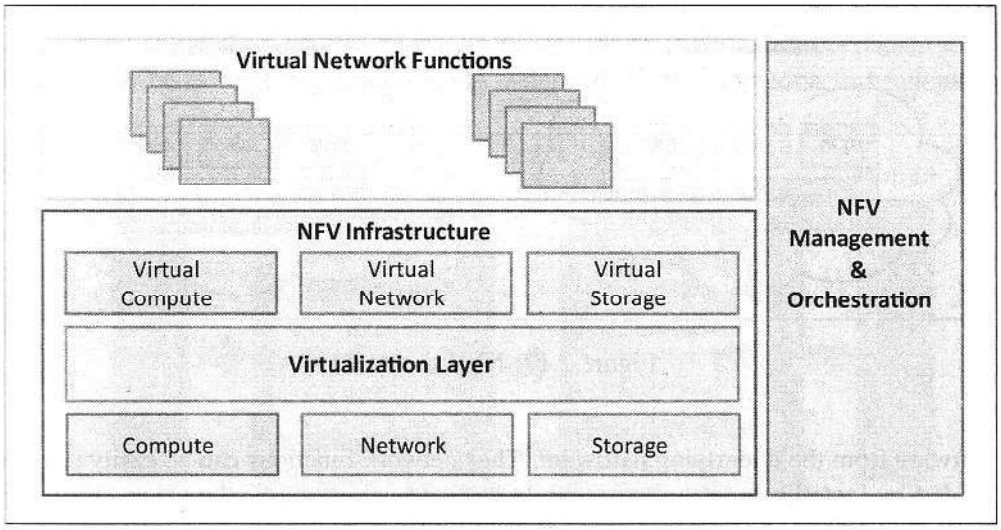


Figure 2.16: NFV architecture

Figure 2.16 shows the NFV architecture, as being standardized by the European Telecommunications Standards Institute (ETSI) [11]. Key elements of the NFV architecture are as follows:

- **Virtualized Network Function (VNF)**: VNF is a software implementation of a network function which is capable of running over the NFV Infrastructure (NFVI).
- **NFV Infrastructure (NFVI)**: NFVI includes compute, network and storage resources that are virtualized.
- **NFV Management and Orchestration**: NFV Management and Orchestration focuses on all virtualization-specific management tasks and covers the orchestration and lifecycle management of physical and/or software resources that support the infrastructure virtualization, and the lifecycle management of VNFs.

NFV comprises of network functions implemented in software that run on virtualized resources in the cloud. NFV enables a separation the network functions which are implemented

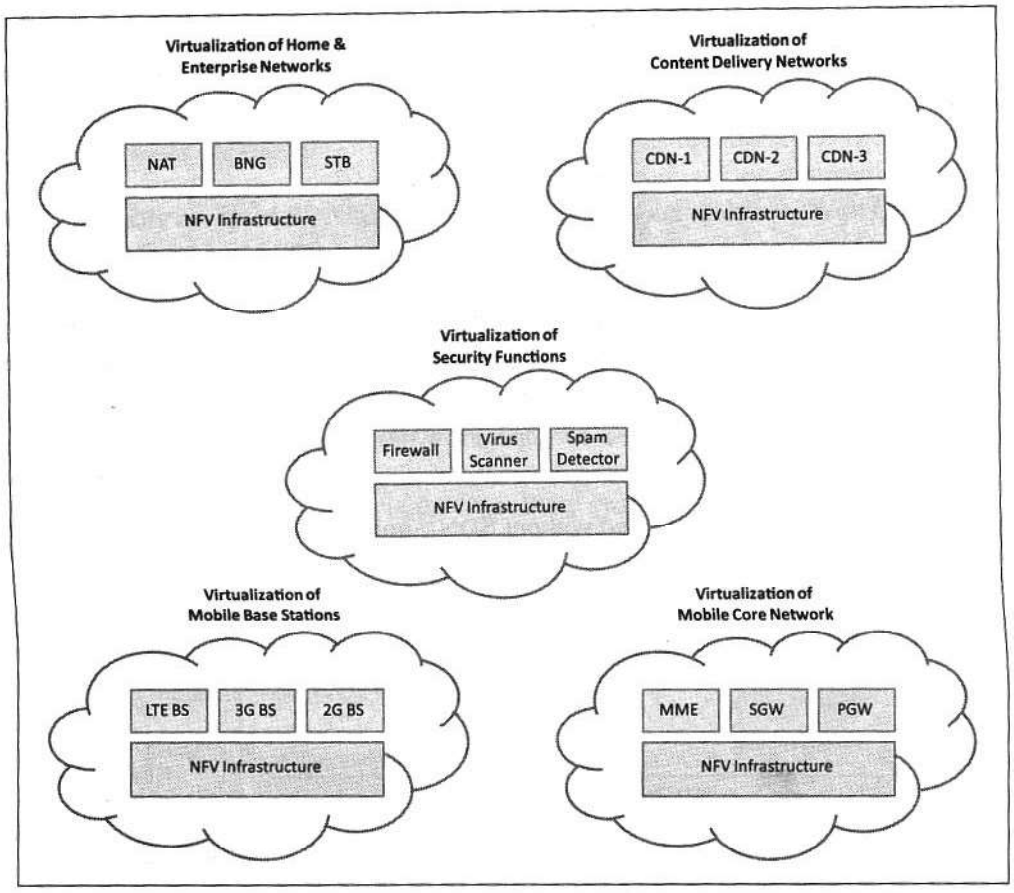


Figure 2.17: NFV use cases

in software from the underlying hardware. Thus network functions can be easily tested and upgraded by installing new software while the hardware remains the same. Virtualizing network functions reduces the equipment costs and also reduces power consumption. The multi-tenanted nature of the cloud allows virtualized network functions to be shared for multiple network services. NFV is applicable only to data plane and control plane functions in fixed and mobile networks. Figure 2.17 shows use cases of NFV for home and enterprise networks, content delivery networks, mobile base stations, mobile core network and security functions.

2.9 MapReduce

MapReduce is a parallel data processing model for processing and analysis of massive scale data [14]. MapReduce model has two phases: Map and Reduce. MapReduce programs are written in a functional programming style to create Map and Reduce functions. The input data to the map and reduce phases is in the form of key-value pairs. Run-time systems for MapReduce are typically large clusters built of commodity hardware. The MapReduce run-time systems take care of tasks such partitioning the data, scheduling of jobs and communication between nodes in the cluster. This makes it easier for programmers

to analyze massive scale data without worrying about tasks such as data partitioning and scheduling. *Figure 2.18 shows the workflow* of MapReduce. In the Map phase, data is read from a distributed file system, partitioned among a set of computing nodes in the cluster, and sent to the nodes as a set of key-value pairs. The Map tasks process the input records independently of each other and produce intermediate results as key-value pairs. The intermediate results are stored on the local disk of the node running the Map task. When all the Map tasks are completed, the Reduce phase begins in which the intermediate data with the same key is aggregated. An optional Combine task can be used to perform data aggregation on the intermediate data of the same key for the output of the mapper before transferring the output to the Reduce task. Figure 2.19 shows the flow of data for a MapReduce job. MapReduce programs take a set of input key-value pairs and produce a set of output key-value pairs. MapReduce programs take advantage of locality of data and the data processing takes place on the nodes where the data resides. In traditional approaches for data analysis, data is moved to the compute nodes which results in significant of data transmission between the nodes in a cluster. MapReduce programming model moves the computation to where the data resides thus decreasing the transmission of data and improving efficiency. MapReduce programming model is well suited for parallel processing of massive scale data in which the data analysis tasks can be accomplished by independent map and reduce operations.

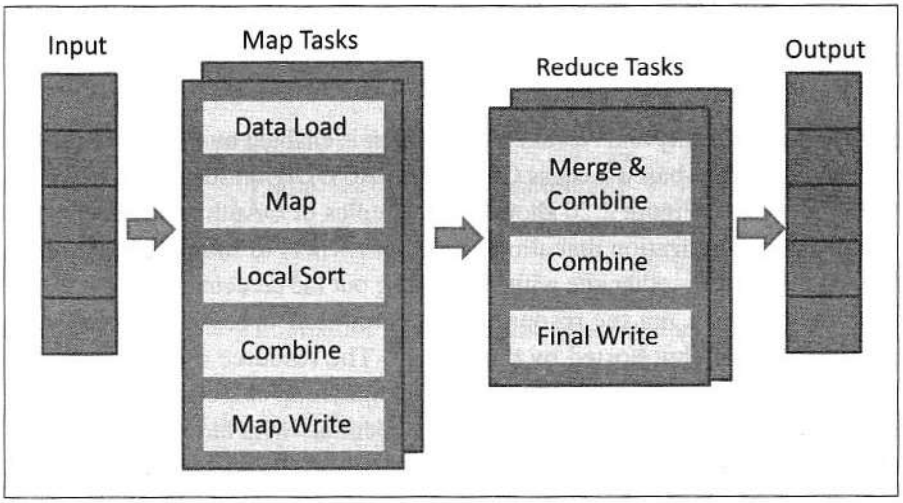


Figure 2.18: MapReduce workflow

2.10 Identity and Access Management

Identity and Access Management (IDAM) for cloud describes the authentication and authorization of users to provide secure access to cloud resources. Organizations with multiple users can use IDAM services provided by the cloud service provider for management of user identifiers and user permissions. IDAM services allow organizations to centrally manage users, access permissions, security credentials and access keys. Organizations can enable role-based access control to cloud resources and applications using the IDAM services. IDAM services allow creation of user groups where all the users in a group have the same

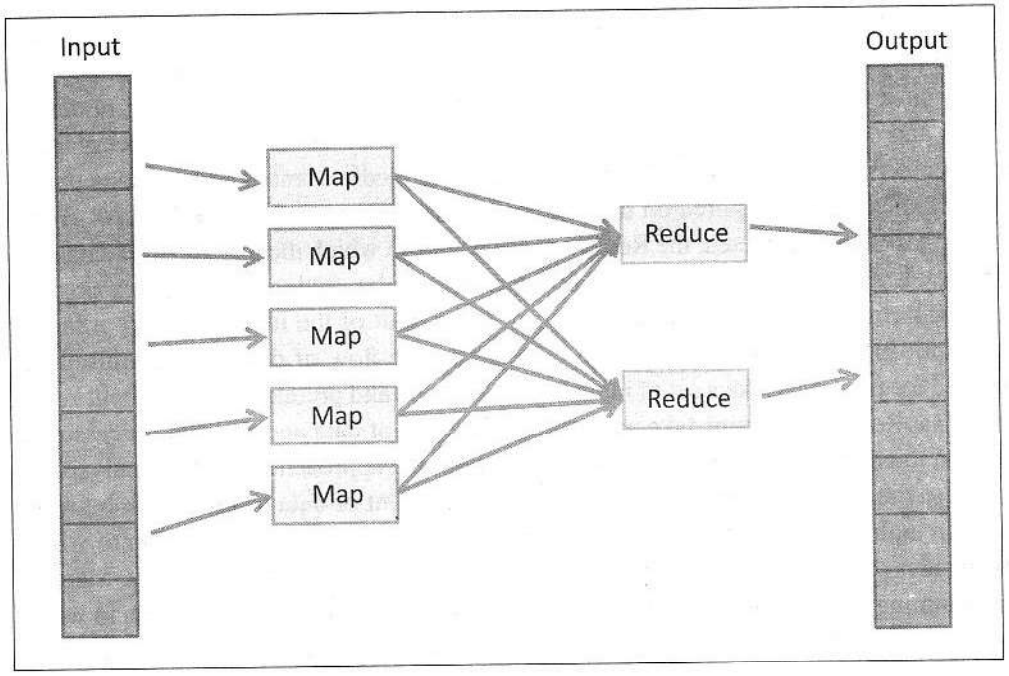


Figure 2.19: Data flow in MapReduce

access permissions. Identity and Access Management is enabled by a number of technologies such as OpenAuth, Role-based Access Control (RBAC), Digital Identities, Security Tokens, Identity Providers, etc. Figure 2.20 shows the examples of OAuth and RBAC. OAuth is an open standard for authorization that allows resource owners to share their private resources stored on one site with an- other site without handing out the credentials. In the OAuth model, an application (which is not the resource owner) requests access to resources controlled by the resource owner (but hosted by the server). The resource owner grants permission to access the resources in the form of a token and matching shared-secret. Tokens make it unnecessary for the resource owner to share its credentials with the application. Tokens can be issued with a restricted scope and limited lifetime, and revoked independently. RBAC is an approach for restricting access to authorized users. Figure 2.21 shows an example of a typical RBAC framework. A user who wants to access cloud resources is required to send his/her data to the system administrator who assigns permissions and access control policies which are stored in the User Roles and Data Access Policies databases respectively.

2.11 Service Level Agreements

A Service Level Agreement (SLA) for cloud specifies the level of service that is formally defined as a part of the service contract with the cloud service provider. SLAs provide a level of service for each service which is specified in the form of minimum level of service guaranteed and a target level. SLAs contain a number of performance metrics and the corresponding service level objectives. Table 2.5 lists the common criteria cloud SLAs.

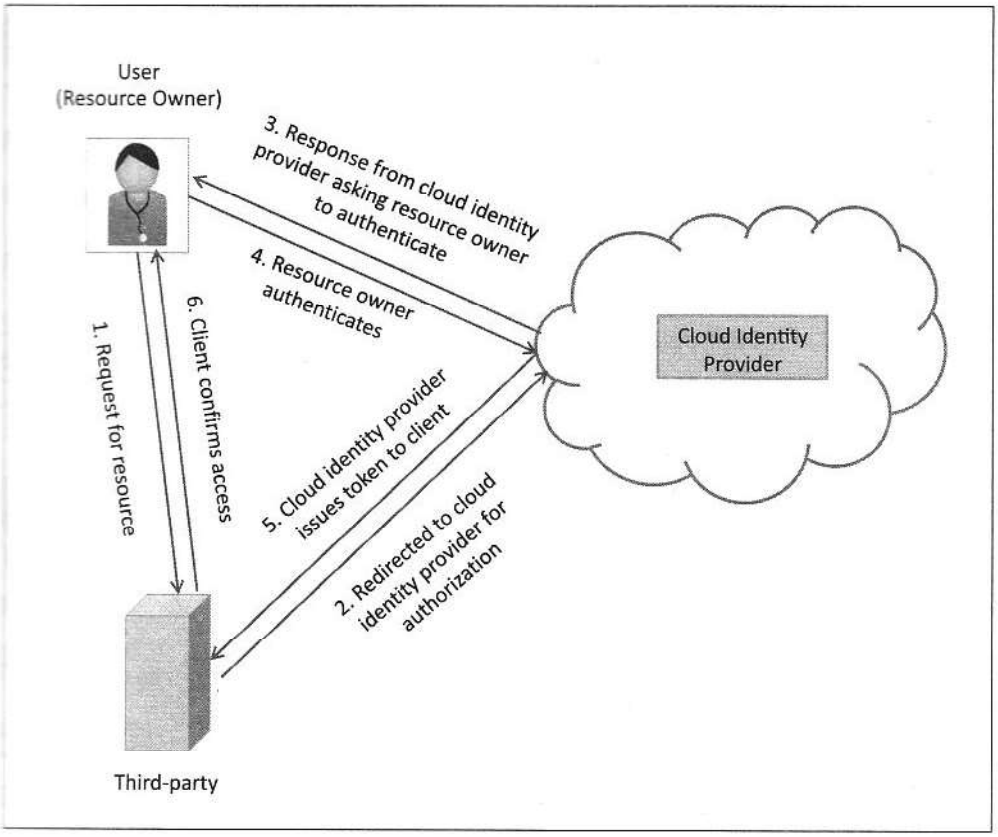


Figure 2.20: OAuth example

2.12 Billing

Cloud service providers offer a number of billing models described as follows:

Elastic Pricing

In elastic pricing or pay-as-you-use pricing model, the customers are charged based on the usage of cloud resources. Cloud computing provides the benefit of provision resources on-demand. On-demand provisioning and elastic pricing models bring cost savings for customers. Elastic pricing model is suited for customers who consume cloud resources for short durations and who cannot predict the usage beforehand.

Fixed Pricing

In fixed pricing models, customers are charged a fixed amount per month for the cloud resources. For example, fixed amount can be charged per month for running a virtual machine instance, irrespective of the actual usage. Fixed pricing model is suited for customers who want to use cloud resources for longer durations and want more control over the cloud expenses.

Spot Pricing

Spot pricing models offer variable pricing for cloud resources which is driven by market demand. When the demand for cloud resources is high, the prices increase and when the

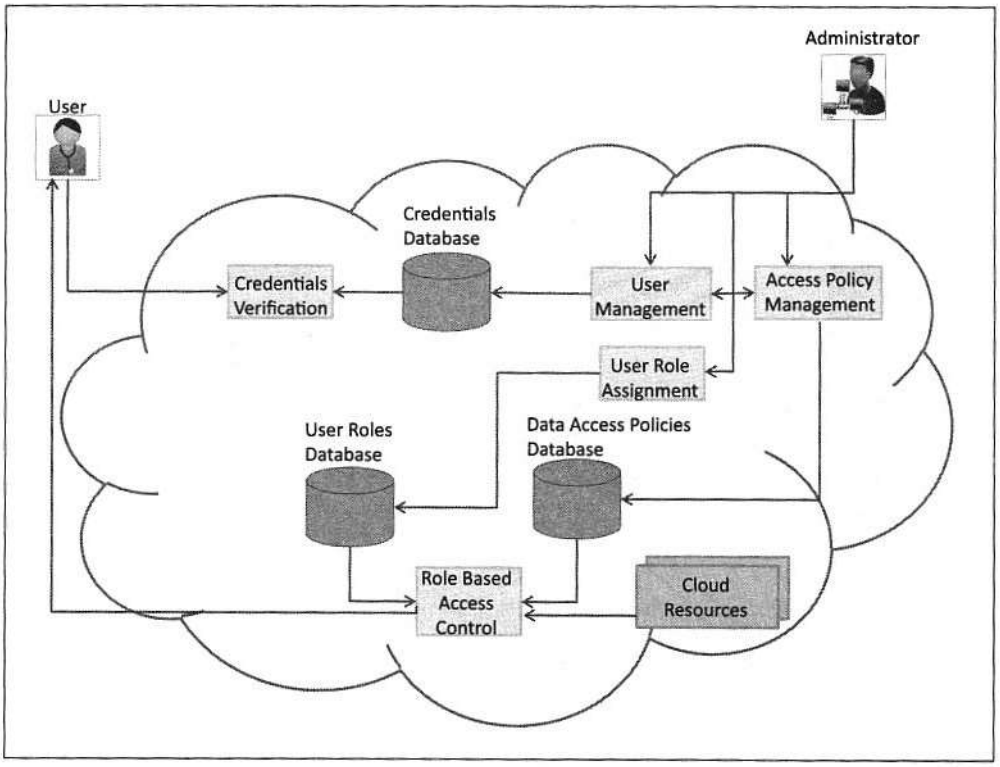


Figure 2.21: Role-based Access Control example

demand is lower, the prices decrease.

Table 2.6 lists the billable resources for cloud including virtual machines, network, storage, data services, security services, support, application services, deployment and management services.

Summary

In this chapter you learned cloud computing concepts and enabling technologies such as virtualization, load balancing, scalability & elasticity, deployment, replication, MapReduce, identity & access management, service level agreements and billing. Virtualization partitions the resources of a physical system (such as computing, storage, network and memory) into multiple virtual resources and enables resource pooling and multi-tenancy.

Review Questions

1. What are the various layers in a virtualization architecture?
2. What is the difference between full and para-virtualization?
3. What are the benefits of load balancing?
4. What are sticky sessions?
5. What are the differences between traditional and on-demand scaling approaches?
6. What are the various stages in the deployment lifecycle?
7. What is the difference between array-based and host-based replication?

Criteria	Details
Availability	Percentage of time the service is guaranteed to be available
Performance	Response time, Throughput
Disaster Recovery	Mean time to recover
Problem resolution	Process to identify problems, support options, resolution expectations
Security and privacy of data	Mechanisms for security of data in storage and transmission

Table 2.5: List of criteria for cloud SLAs

Resource	Details
Virtual machines	CPU, memory, storage, disk I/O, network I/O
Network	Network I/O, load balancers, DNS, firewall, VPN
Storage	Cloud storage, storage volumes, storage gateway
Data services	Data import/export services, data encryption, data compression, data backup, data redundancy, content delivery
Security services	Identity and access management, isolation, compliance
Support	Level of support, SLA, fault tolerance
Application services	Queuing service, notification service, workflow service, payment service
Deployment and management services	Monitoring service, deployment service

Table 2.6: List of billable resources for cloud

8. In MapReduce, what are the functions of map, reduce and combine tasks?
9. Describe three applications that can benefit from the MapReduce programming model?
10. What are the various criteria for service level agreements?

3 — Cloud Services & Platforms

This Chapter covers

- Compute Services
- Storage Services
- Database Services
- Application Services
- Content Delivery Services
- Analytics Services
- Deployment & Management Services
- Identity & Access Management Services

In this chapter you will learn about various types of cloud computing services including compute, storage, database, application, content delivery, analytics, deployment & management and identity & access management. For each category of cloud services, examples of services provided by various cloud service providers including Amazon, Google and Microsoft are described.

Figure 3.1 (a) shows the cloud computing reference model along with the various cloud service models (IaaS, PaaS and SaaS). Infrastructure-as-a-Service (IaaS) provides virtualized dynamically scalable resources using a virtualized infrastructure. Platform-as-a-Service (PaaS) simplifies application development by providing development tools, application programming interfaces (APIs), software libraries that can be used for wide range of applications. Software-as-a-Service (SaaS) provides multi-tenant applications hosted in the cloud.

The bottommost layer in the cloud reference model is the infrastructure and facilities layer that includes the physical infrastructure such as datacenter facilities, electrical and mechanical equipment, etc. On top of the infrastructure layer is the hardware layer that includes physical compute, network and storage hardware. On top of the hardware layer the virtualization layer partitions the physical hardware resources into multiple virtual resources that enabling pooling of resources. Chapter 2 described various types of virtualization approaches such as full virtualization, para-virtualization and hardware virtualization. The computing services are delivered in the form of Virtual Machines (VMs) along with the storage and network resources.

The platform and middleware layer builds upon the IaaS layers below and provides standardized stacks of services such as database service, queuing service, application frameworks and run-time environments, messaging services, monitoring services, analytics services, etc. The service management layer provides APIs for requesting, managing and monitoring cloud resources. The topmost layer is the applications layer that includes SaaS applications such as Email, cloud storage application, productivity applications, management portals, customer self-service portals, etc.

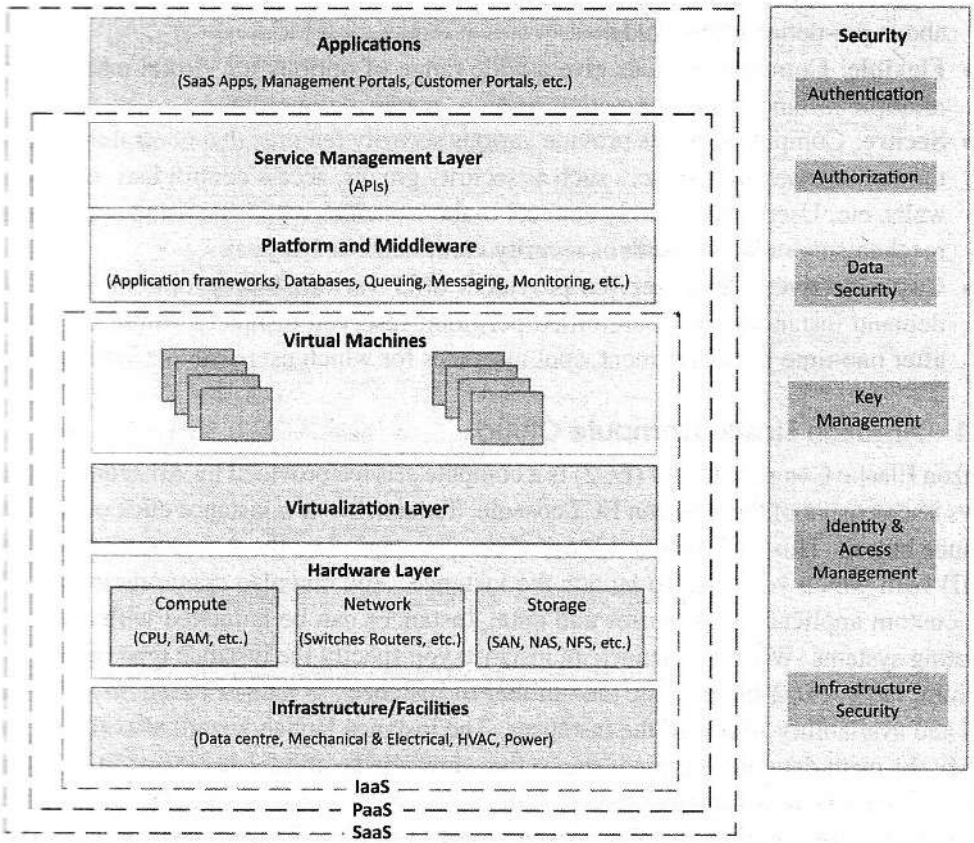
Figure 3.1 (b) shows various types of cloud services and the associated layers in the cloud reference model.

3.1 Compute Services

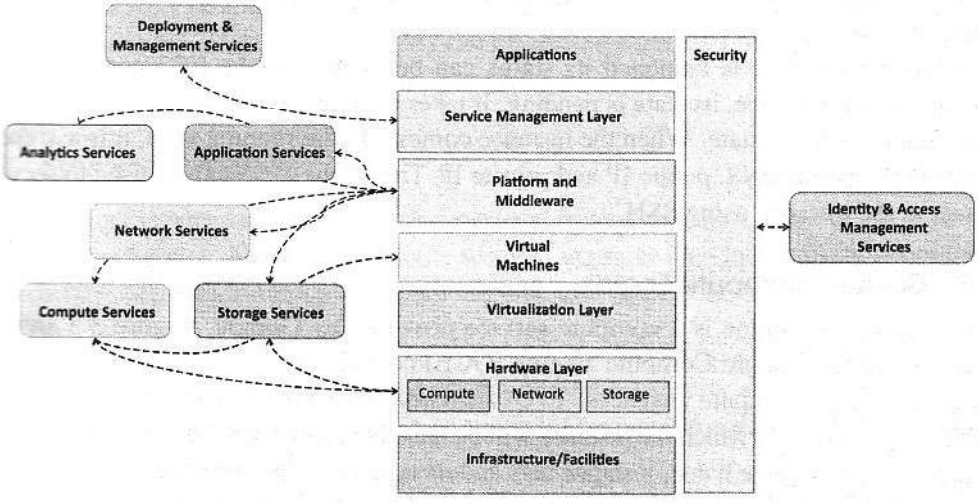
Compute services provide dynamically scalable compute capacity in the cloud. Compute resources can be provisioned on-demand in the form of virtual machines. Virtual machines can be created from standard images provided by the cloud service provider (e.g. Ubuntu image, Windows server image, etc.) or custom images created by the users. A machine image is a template that contains a software configuration (operating system, application server, and applications). Compute services can be accessed from the web consoles of these services that provide graphical user interfaces for provisioning, managing and monitoring these services. Cloud service providers also provide APIs for various programming languages (such as Java, Python, etc.) that allow developers to access and manage these services programmatically.

Features

- **Scalable**: Compute services allow rapidly provisioning as many virtual machine instances as required. The provisioned capacity can be scaled-up or down based on the



(a) Cloud reference model



(b) Cloud services

Figure 3.1: Cloud Computing reference model & services

workload levels. Auto-scaling policies can be defined for compute services that are triggered when the monitored metrics (such as CPU usage, memory usage, etc.) go above pre-defined thresholds.

- **Flexible**: Compute services give a wide range of options for virtual machines with multiple instance types, operating systems, zones/regions, etc.
- **Secure**: Compute services provide various security features that control the access to the virtual machine instances such as security groups, access control lists, network firewalls, etc. Users can securely connect to the instances with SSH using authentication mechanisms such as OAuth or security certificates and keypairs.
- **Cost effective**: Cloud service providers offer various billing options such as on-demand instances which are billed per-hour, reserved instances which are reserved after one-time initial payment, spot instances for which users can place bids, etc.

3.1.1 Amazon Elastic Compute Cloud

Amazon Elastic Compute Cloud (EC2) is a compute service provided by Amazon. Figure 3.2 shows a screenshot of the Amazon EC2 console. To launch a new instance click on the launch instance button. This will open a wizard where you can select the Amazon machine image (AMI) with which you want to launch the instance. You can also create their own AMIs with custom applications, libraries and data. Instances can be launched with a variety of operating systems. When you launch an instance you specify the instance type (micro, small, medium, large, extra-large, etc.), the number of instances to launch based on the selected AMI and availability zones for the instances. The instance launch wizard also allows you to specify the meta-data tags for the instance that simplify the administration of EC2 instances. When launching a new instance, the user selects a key-pair from existing keypairs or creates a new keypair for the instance. Keypairs are used to securely connect to an instance after it launches. The security groups to be associated with the instance can be selected from the instance launch wizard. Security groups are used to open or block a specific network port for the launched instances.

When the instance is launched its status can be viewed in the EC2 console. Upon launching a new instance, its state is pending. It takes a couple of minutes for the instance to come into the running state. When the instance comes into the running state, it is assigned a public DNS, private DNS, public IP and private IP. The public DNS can be used to securely connect to the instance using SSH.

3.1.2 Google Compute Engine

Google Compute Engine is a compute service provided by Google. Figure 3.3 shows a screenshot of the Google Compute Engine (GCE) console. GCE console allows users to create and manage compute instances. To create a new instance, the user selects an instance machine type, a zone in which the instance will be launched, a machine image for the instance and provides an instance name, instance tags and meta-data. Every instance is launched with a disk resource. Depending on the instance type, the disk resource can be a scratch disk space or persistent disk space. The scratch disk space is deleted when the instance terminates. Whereas, persistent disks live beyond the life of an instance. Network option allows you to control the traffic to and from the instances. By default, traffic between instances in the same network, over any port and any protocol and incoming SSH connections from anywhere are

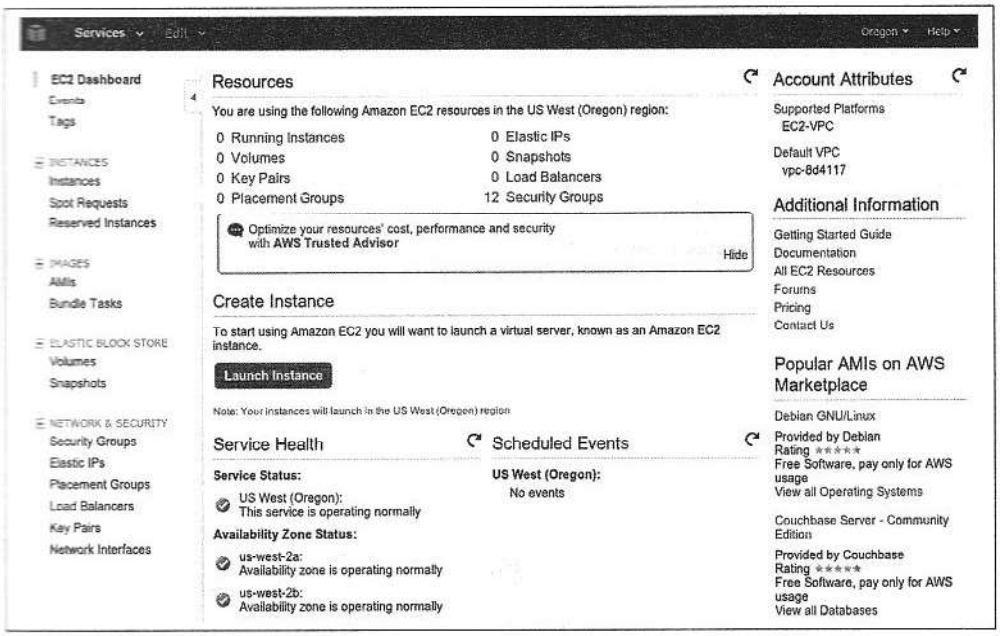


Figure 3.2: Screenshot of Amazon EC2 console

enabled. To enable other connections, additional firewall rules can be added.

3.1.3 Windows Azure Virtual Machines

Windows Azure Virtual Machines is the compute service from Microsoft. Figure 3.4 shows a screenshot of Windows Azure Virtual Machines console. To create a new instance, you select the instance type and the machine image. You can either provide a user name and password or upload a certificate file for securely connecting to the instance. Any changes made to the VM are persistently stored and new VMs can be created from the previously stored machine images.

3.2 Storage Services

Cloud storage services allow storage and retrieval of any amount of data, at any time from anywhere on the web. Most cloud storage services organize data into buckets or containers. Buckets or containers store objects which are individual pieces of data.

Features

- **Scalability**: Cloud storage services provide high capacity and scalability. Objects upto several tera-bytes in size can be uploaded and multiple buckets/containers can be created on cloud storages.
- **Replication**: When an object is uploaded it is replicated at multiple facilities and/or on multiple devices within each facility.
- **Access Policies**: Cloud storage services provide several security features such as Access Control Lists (ACLs), bucket/container level policies, etc. ACLs can be used to selectively grant access permissions on individual objects. Bucket/container level

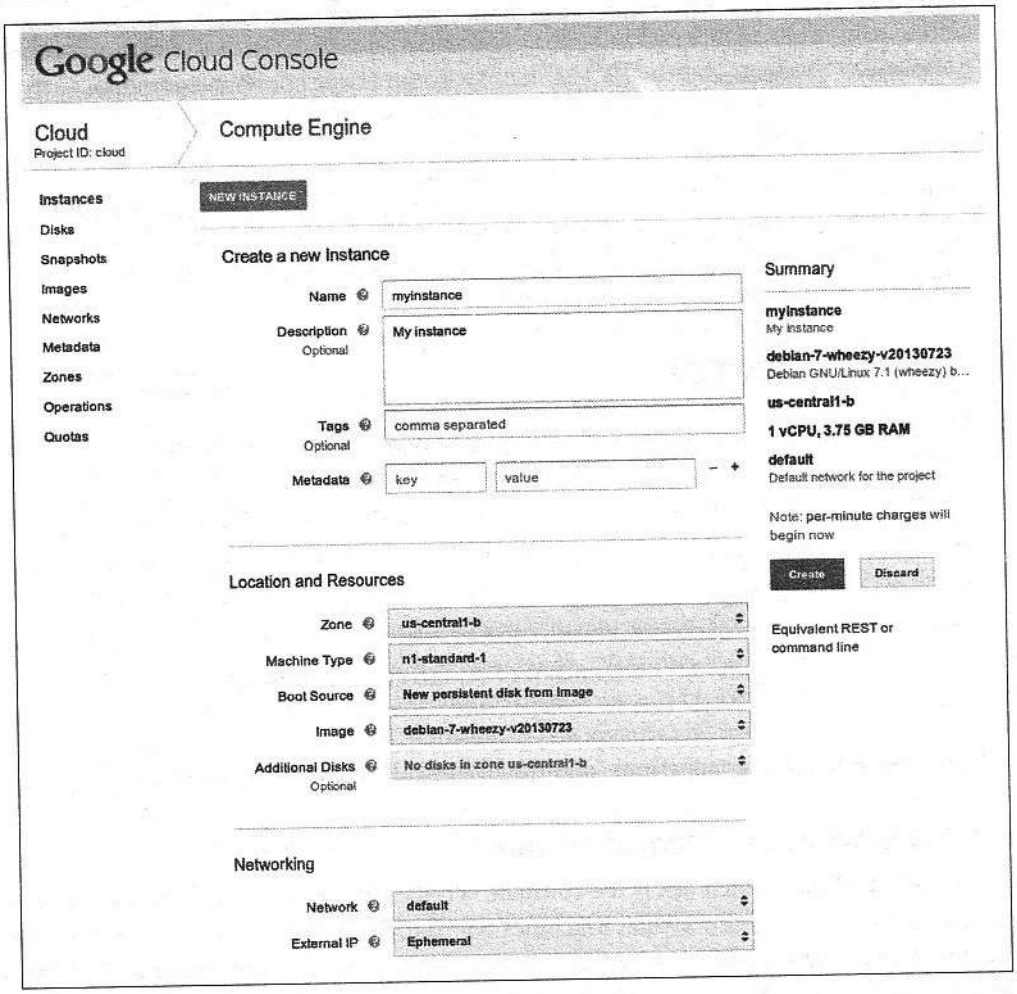


Figure 3.3: Screenshot of Google Compute Engine console

policies can also be defined to allow or deny permissions across some or all of the objects within a single bucket/container.

- **Encryption**: Cloud storage services provide Server Side Encryption (SSE) options to encrypt all data stored in the cloud storage.
- **Consistency**: Strong data consistency is provided for all upload and delete operations. Therefore, any object that is uploaded can be immediately downloaded after the upload is complete.

3.2.1 Amazon Simple Storage Service

Amazon Simple Storage Service(S3) is an online cloud-based data storage infrastructure for storing and retrieving any amount of data. S3 provides highly reliable, scalable, fast, fully redundant and affordable storage infrastructure. Figure 3.5 shows a screenshot of the Amazon S3 console. Data stored on S3 is organized in the form of buckets. You must create a bucket before you can store data on S3. S3 console provides simple wizards for creating a new bucket and uploading files. You can upload any kind of file to S3. While uploading a

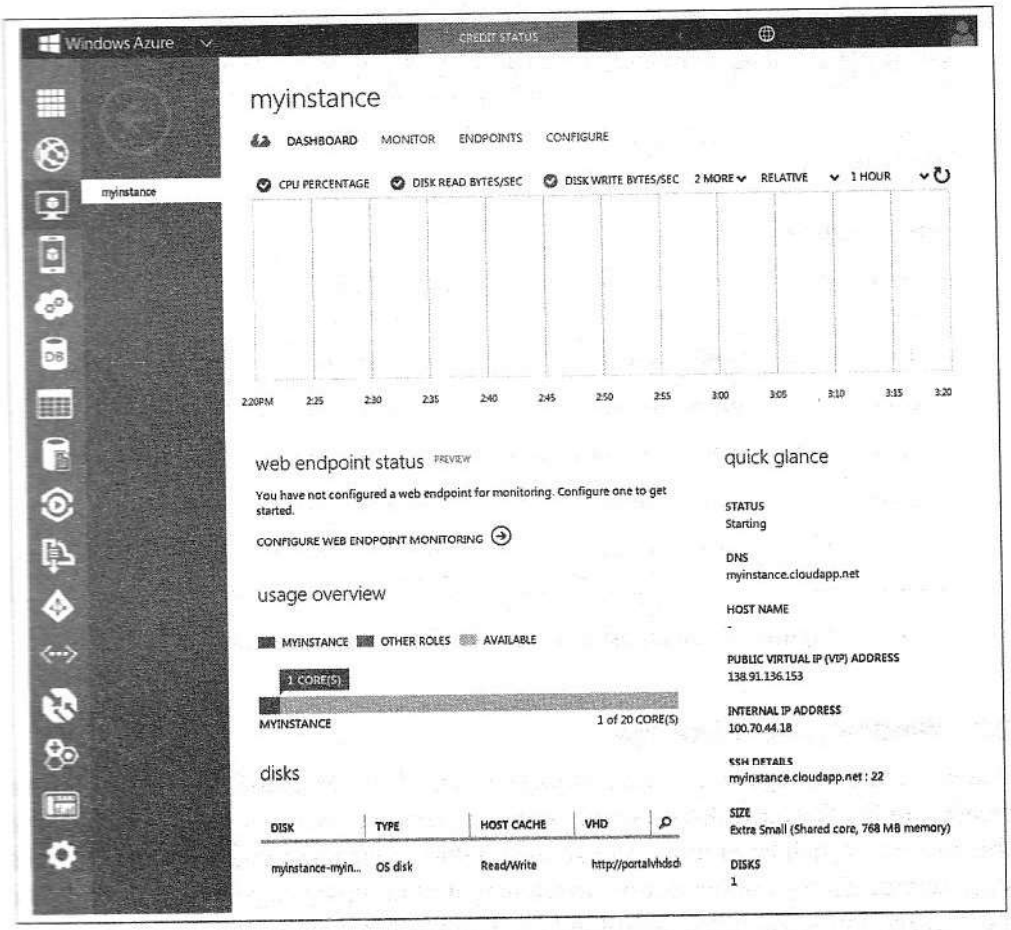


Figure 3.4: Screenshot of Windows Azure Virtual Machines console

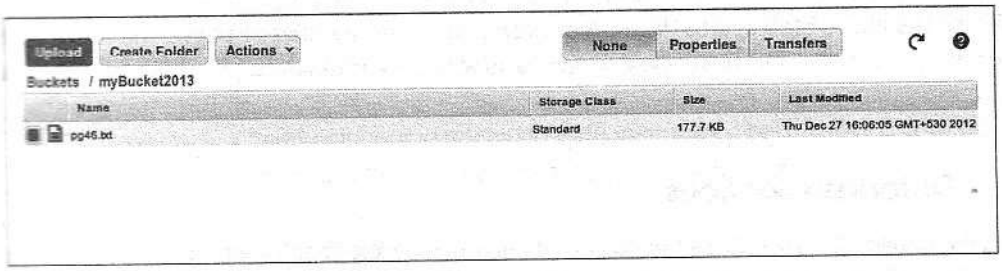


Figure 3.5: Screenshot of Amazon S3 console

file, you can specify the redundancy and encryption options and access permissions.

3.2.2 Google Cloud Storage

Figure 3.6 shows a screenshot of the Google Cloud Storage (GCS) console. Objects in GCS are organized into buckets. ACLs are used to control access to objects and buckets. ACLs can be configured to share objects and buckets with the entire world, a Google group, a Google-hosted domain, or specific Google account holders.

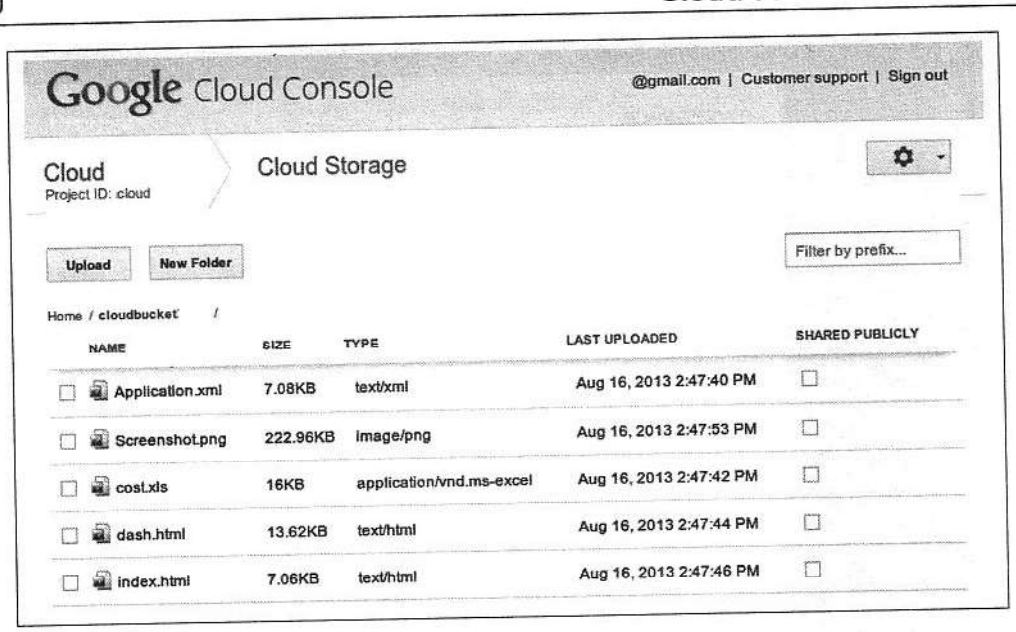


Figure 3.6: Screenshot of Google Cloud Storage console

3.2.3 Windows Azure Storage

Windows Azure Storage is the cloud storage service from Microsoft. Figure 3.7 shows a screenshot of the Windows Azure Storage console. Windows Azure Storage provides various storage services such as blob storage service, table service and queue service. The blob storage service allows storing unstructured binary data or binary large objects (blobs). Blobs are organized into containers. Two kinds of blobs can be stored - block blobs and page blobs. A block blob can be subdivided into some number of blocks. If a failure occurs while transferring a block blob, retransmission can resume with the most recent block rather than sending the entire blob again. Page blobs are divided into number of pages and are designed for random access. Applications can read and write individual pages at random in a page blob.

3.3 Database Services

Cloud database services allow you to set-up and operate relational or non-relational databases in the cloud. The benefit of using cloud database services is that it relieves the application developers from the time consuming database administration tasks. Popular relational databases provided by various cloud service providers include MySQL, Oracle, SQL Server, etc. The non-relational (No-SQL) databases provided by cloud service providers are mostly proprietary solutions. No-SQL databases are usually fully-managed and deliver seamless throughput and scalability. The characteristics of relational and non-relational databases are described in Chapter 5.

Features

- **Scalability**: Cloud database services allow provisioning as much compute and storage resources as required to meet the application workload levels. Provisioned capacity

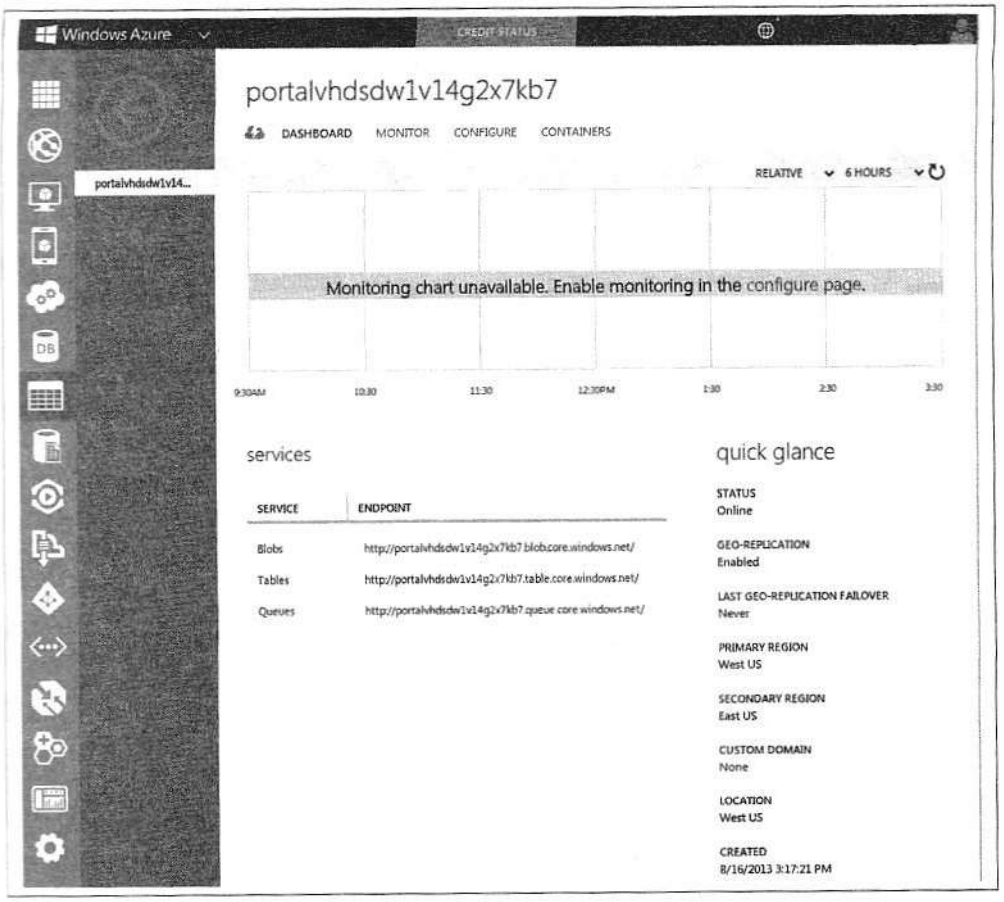


Figure 3.7: Screenshot of Windows Azure Storage console

can be scaled-up or down. For read-heavy workloads, read-replicas can be created.

- **Reliability**: Cloud database services are reliable and provide automated backup and snapshot options.
- **Performance**: Cloud database services provide guaranteed performance with options such as guaranteed input/output operations per second (IOPS) which can be provisioned upfront.
- **Security**: Cloud database services provide several security features to restrict the access to the database instances and stored data, such as network firewalls and authentication mechanisms.

3.3.1 Amazon Relational Data Store

Amazon Relational Database Service (RDS) is a web service that makes it easy to setup, operate and scale a relational database in the cloud. Figure 3.8 shows a screenshot of the Amazon RDS console. The console provides an instance launch wizard that allows you to select the type of database to create (MySQL, Oracle or SQL Server) database instance size, allocated storage, DB instance identifier, DB username and password. The status of the launched DB instances can be viewed from the console. It takes several minutes for

the instance to become available. Once the instance is available, you can note the instance end point from the instance properties tab. This end point can then be used for securely connecting to the instance.

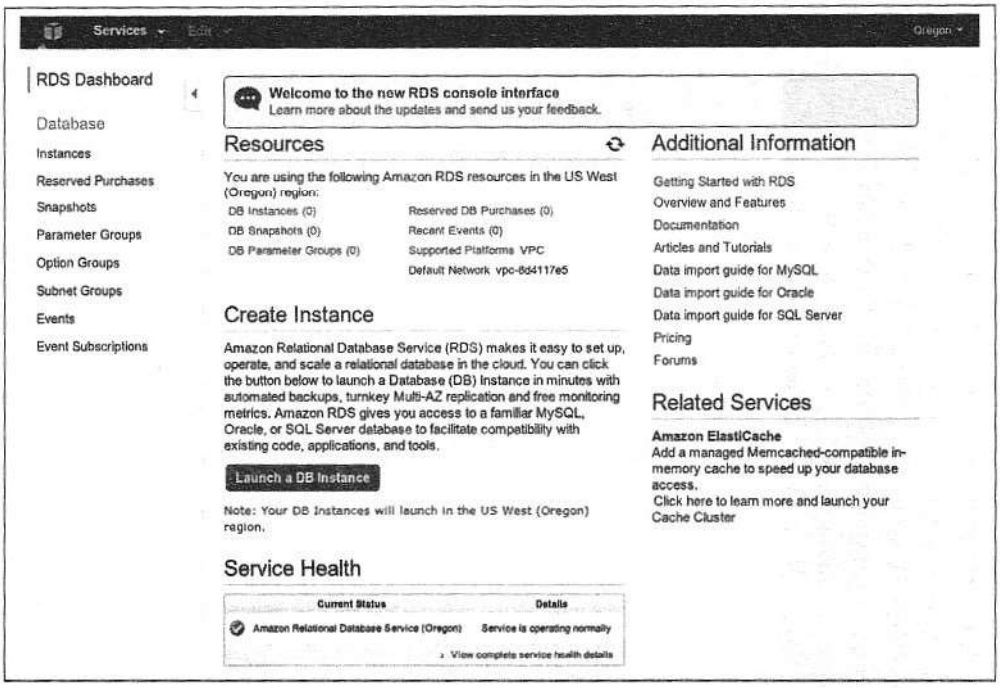


Figure 3.8: Screenshot of Amazon RDS console

3.3.2 Amazon DynamoDB

Amazon DynamoDB is the non-relational (No-SQL) database service from Amazon. Figure 3.9 shows a screenshot of the Amazon DynamoDB console. The DynamoDB data model includes include tables, items and attributes. A table is a collection of items and each item is a collection of attributes. To store data in DynamoDB you have to create a one or more tables and specify how much throughput capacity you want to provision and reserve for reads and writes. DynamoDB is a fully managed service that automatically spreads the data and traffic for the stored tables over a number of servers to meet the throughput requirements specified by the users. All stored data is automatically replicated across multiple availability zones to provide data durability.

3.3.3 Google Cloud SQL

Google SQL is the relational database service from Google. Google Cloud SQL service allows you to host MySQL databases in the Google's cloud. Cloud SQL provides both synchronous or asynchronous geographic replication and the ability to import/ export databases. Figure 3.10 shows a screenshot of the Google Cloud SQL console. You can create new database instances from the console and manage existing instances. To create a new instance you select a region, database tier, billing plan and replication mode. You can schedule daily backups for your Google Cloud SQL instances, and also restore backed-up databases.

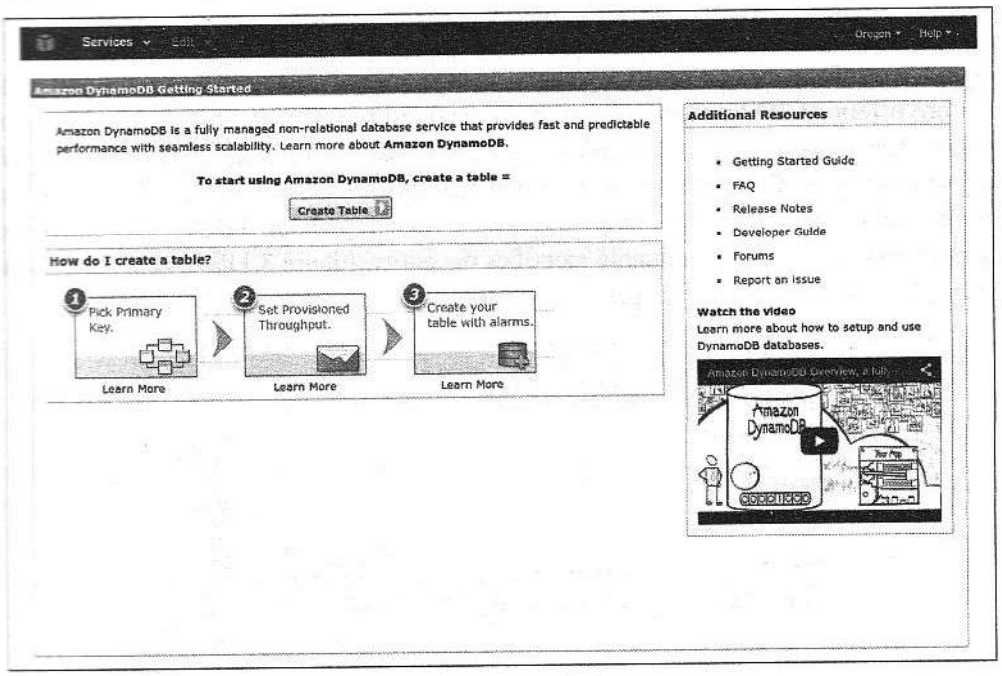


Figure 3.9: Screenshot of Amazon DynamoDB console

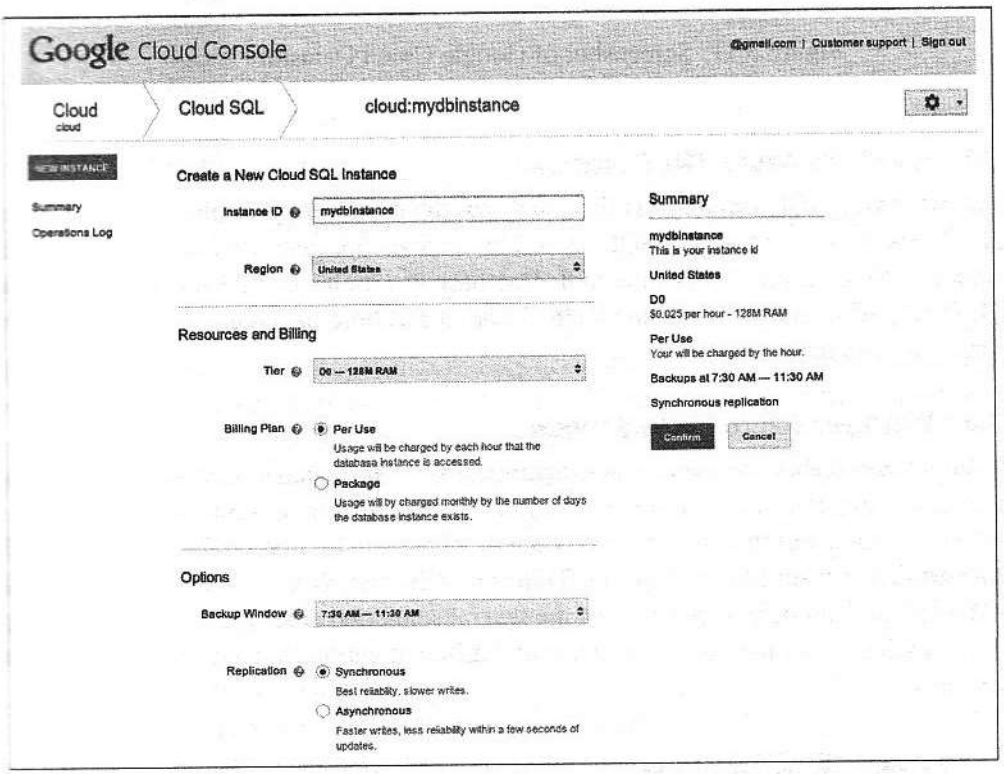


Figure 3.10: Screenshot of Google Cloud SQL console

3.3.4 Google Cloud Datastore

Google Cloud Datastore is a fully managed non-relational database from Google. Cloud Datastore offers ACID transactions and high availability of reads and writes. The Cloud Datastore data model consists of entities. Each entity has one or more properties (key-value pairs) which can be of one of several supported data types, such as strings and integers. Each entity has a kind and a key. The entity kind is used for categorizing the entity for the purpose of queries and the entity key uniquely identifies the entity. Figure 3.11 shows a screenshot of the Google Cloud Datastore console.

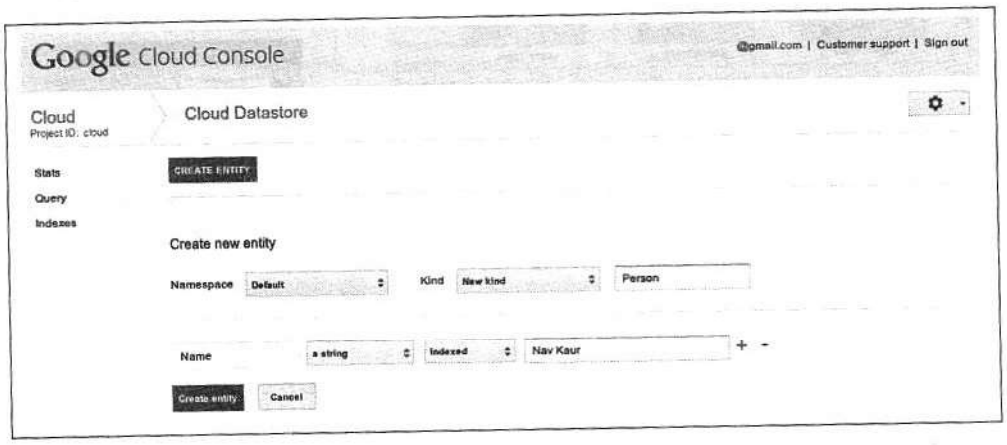


Figure 3.11: Screenshot of Google Cloud Datastore console

3.3.5 Windows Azure SQL Database

Windows Azure SQL Database is the relational database service from Microsoft. Azure SQL Database is based on the SQL server, but it does not give each customer a separate instance of SQL server. Instead the SQL Database is a multi-tenant service, with a logical SQL Database server for each customer. Figure 3.12 shows a screenshot of the Windows Azure SQL Database console.

3.3.6 Windows Azure Table Service

Windows Azure Table Service is a non-relational (No-SQL) database service from Microsoft. The Azure Table Service data model consists of tables having multiple entities. Tables are divided into some number of partitions, each of which can be stored on a separate machine. Each partition in a table holds a specified number of entities, each containing as many as 255 properties. Each property can be one of the several supported data types such as integers and strings. Tables do not have a fixed schema and different entities in a table can have different properties.

3.4 Application Services

In this section you will learn about various cloud application services such as application runtimes and frameworks, queuing services, email services, notification services and media services.

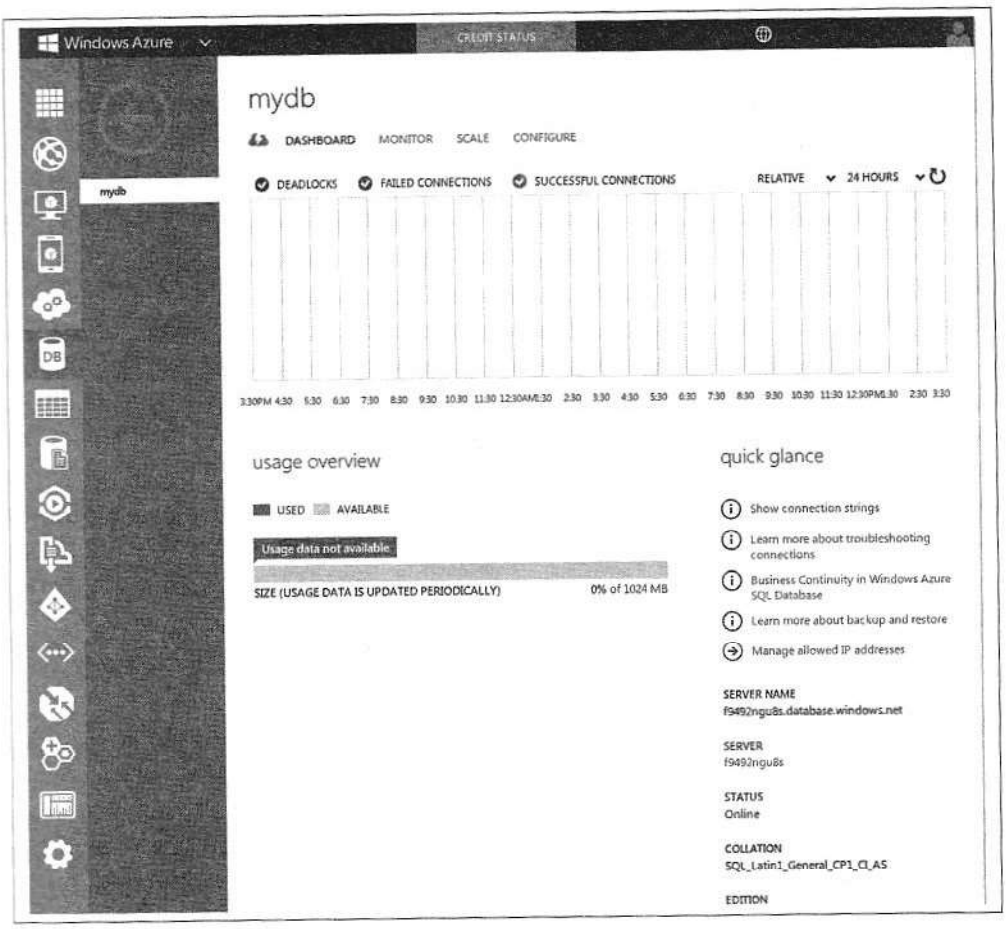


Figure 3.12: Screenshot of Windows Azure SQL Database console

3.4.1 Application Runtimes & Frameworks

Cloud-based application runtimes and frameworks allow developers to develop and host applications in the cloud. Application runtimes provide support for programming languages (e.g., Java, Python, or Ruby). Application runtimes automatically allocate resources for applications and handle the application scaling, without the need to run and maintain servers.

Google App Engine

Google App Engine is the platform-as-a-service (PaaS) from Google, which includes both an application runtime and web frameworks. Figure 3.13 shows a screenshot of the Google App Engine console.

App Engine features include:

- **Runtimes**: App Engine supports applications developed in Java, Python, PHP and Go programming languages. App Engine provides runtime environments for Java, Python, PHP and Go programming language.
- **Sandbox**: Applications run in a secure sandbox environment isolated from other applications. The sandbox environment provides a limited access to the underlying operating system. App Engine can only execute application code called from HTTP

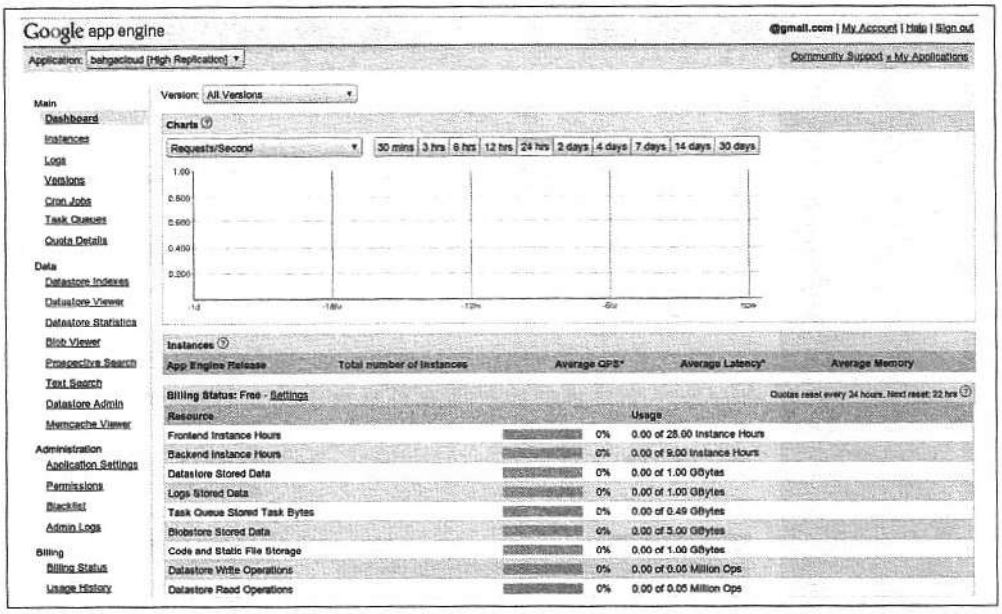


Figure 3.13: Screenshot of Google App Engine console

requests. The sandbox environment allows App Engine to distribute web requests for the application across multiple servers.

- **Web Frameworks**: App Engine provides a simple Python web application framework called webapp2. App Engine also supports any framework written in pure Python that speaks WSGI, including Django, CherryPy, Pylons, web.py, and web2py.
- **Datastore**: App Engine provides a no-SQL data storage service.
- **Authentication**: App Engine applications can be integrated with Google Accounts for user authentication.
- **URL Fetch service**: URL Fetch service allows applications to access resources on the Internet, such as web services or other data
- **Email service**: Email service allows applications to send email messages.
- **Image Manipulation service**: Image Manipulation service allows applications to resize, crop, rotate, flip and enhance images.
- **Memcache**: Memcache service is a high performance in-memory key-value cache service that applications can use for caching data items that do not need a persistent storage.
- **Task Queues**: Task queues allow applications to do work in the background by breaking up work into small, discrete units, called tasks which are enqueued in task queues.
- **Scheduled Tasks service** : App Engine provides a Cron service for scheduled tasks that trigger events at specified times and regular intervals. This service allows applications to perform tasks at defined times or regular intervals.

Windows Azure Web Sites

Windows Azure Web Sites is a Platform-as-a-Service (PaaS) from Microsoft. Azure Web Sites allows you to host web applications in the Azure cloud. Azure Web Sites provides shared

and standard options. In the shared option, Azure Web Sites run on a set of virtual machines that may contain multiple web sites created by multiple users. In the standard option, Azure Web Sites run on virtual machines (VMs) that belong to an individual user. Azure Web Sites supports applications created in ASP.NET, PHP, Node.js and Python programming languages. Multiple copies of an application can be run in different VMs, with Web Sites automatically load balancing requests across them.

3.4.2 Queuing Services

Cloud-based queuing services allow de-coupling application components. The de-coupled components communicate via messaging queues. Queues are useful for asynchronous processing. Another use of queues is to act as overflow buffers to handle temporary volume spikes or mismatches in message generation and consumption rates from application components. Queuing services from various cloud service providers allow short messages of a few kilo-bytes in size. Messages can be enqueued and read from the queues simultaneously. The enqueued messages are typically retained for a couple of days to a couple of weeks.

Amazon Simple Queue Service

Amazon Simple Queue Service (SQS) is a queuing service from Amazon. SQS is a distributed queue that supports messages of up to 256 KB in size. SQS supports multiple writers and readers and locks messages while they are being processed. To ensure high availability for delivering messages, SQS service trade-offs on the first in, first out capability and does not guarantee that messages will be delivered in FIFO order. Applications that require FIFO ordering of messages can place additional sequencing information in each message so that they can be re-ordered after retrieving from a queue. Figure 3.14 shows a screenshot of the Amazon Simple Queue Service console.

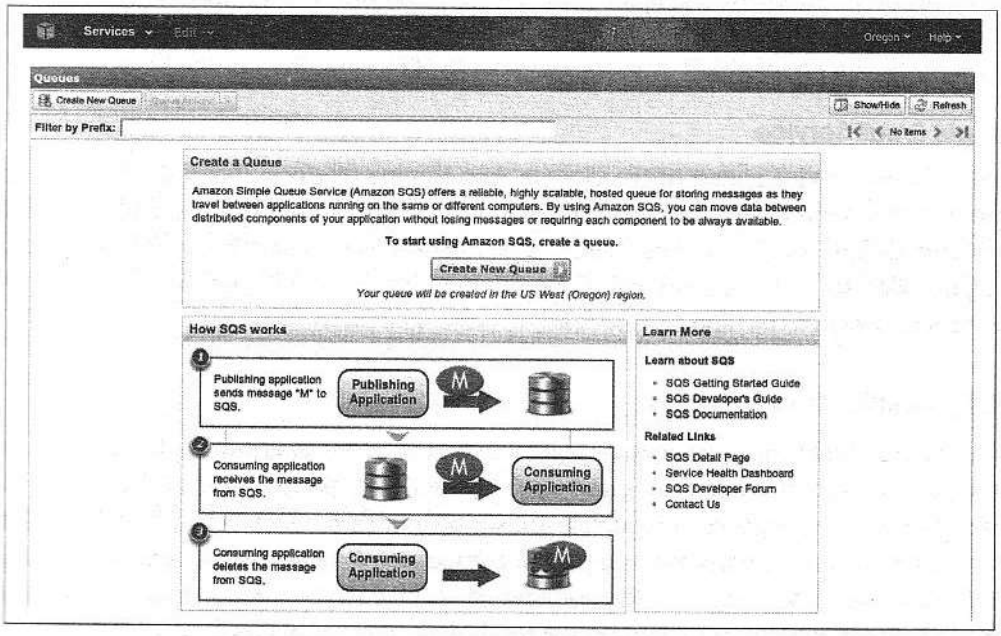


Figure 3.14: Screenshot of Amazon SQS console

Google Task Queue Service

Google Task Queues service is a queuing service from Google and is a part of the Google App Engine platform. Task queues allow applications to execute tasks in background. Task is a unit of work to be performed by an application. The task objects consist of application-specific URL with a request handler for the task, and an optional data payload that parameterizes the task. There are two different configurations for Task Queues - Push Queue and Pull Queue. Push Queue is the default queue that processes tasks based on the processing rate configured in the queue definition. Pull Queues allow task consumers to lease a specific number of tasks for a specific duration. The tasks are processed and deleted before the lease ends.

Windows Azure Queue Service

Windows Azure Queue service is a queuing service from Microsoft. Azure Queue service allows storing large numbers of messages that can be accessed from anywhere in the world via authenticated calls using HTTP or HTTPS. The size of a single message can be up to 64KB.

3.4.3 Email Services

Cloud-based email services allow applications hosted in the cloud to send emails.

Amazon Simple Email Service

Amazon Simple Email Service is bulk and transactional email-sending service from Amazon. SES is an outbound-only email-sending service that allows applications hosted in the Amazon cloud to send emails such as marketing emails, transactional emails and other types of correspondence. To ensure high email deliverability, SES uses content filtering technologies to scan the outgoing email messages to help ensure that they do not contain material that is typically flagged as questionable by ISPs. SES service can be accessed and used from the SES console, the Simple Mail Transfer Protocol (SMTP) interface, or the SES API.

Google Email Service

Google Email service is part of the Google App Engine platform that allows App Engine applications to send email messages on behalf of the app's administrators, and on behalf of users with Google Accounts. App Engine apps can also receive emails. Apps send messages using the Mail service and receive messages in the form of HTTP requests initiated by App Engine and posted to the app.

3.4.4 Notification Services

Cloud-based notification services or push messaging services allow applications to push messages to internet connected smart devices such as smartphones, tablets, etc. Push messaging services are based on publish-subscribe model in which consumers subscribe to various topics/channels provided by a publisher/producer. Whenever new content is available on one of those topics/channels, the notification service pushes that information out to the consumer. Push notifications are used for such smart devices as they help in displaying the latest information while remaining energy efficient. Consumer applications on such devices can increase their consumer engagement with the help of push notifications.

Amazon Simple Notification Service

Amazon Simple Notification Service is a push messaging service from Amazon. SNS has two types of clients - publishers and subscribers. Publishers communicate asynchronously with subscribers by producing and sending messages to topics. A topic is a logical access point and a communication channel. Subscribers are the consumers who subscribe to topics to receive notifications. SNS can deliver notifications as SMS, email, or to SQS queues, or any HTTP endpoint. Figure 3.15 shows a screenshot of the Amazon Simple Notification Service console. The SNS console has wizards for creating a new topic, publishing to a topic and subscribing to a topic.

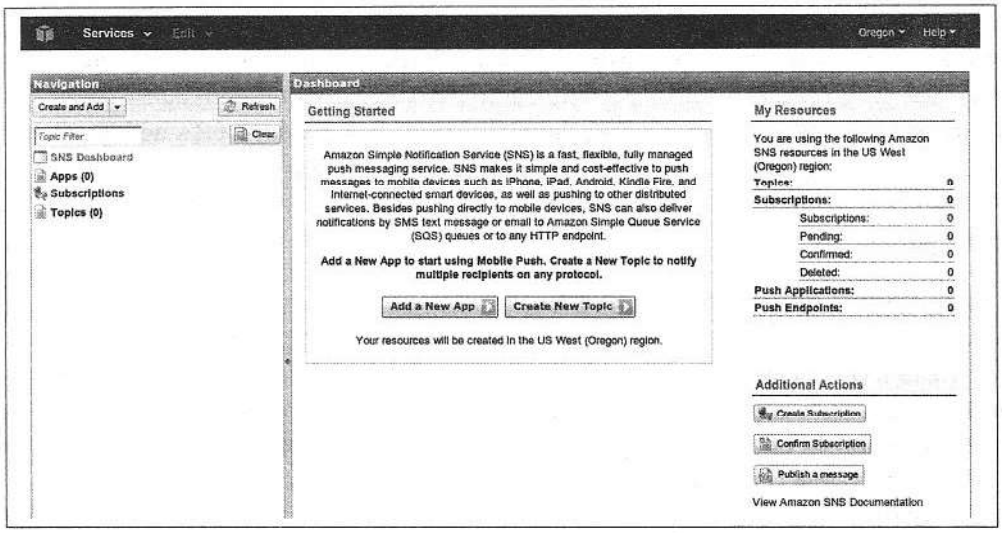


Figure 3.15: Screenshot of Amazon SNS console

Google Cloud Messaging

Google Cloud Messaging for Android provides push messaging for Android devices. GCM allows allows applications to send data from the application servers to their users' Android devices, and also to receive messages from devices on the same connection. GCM is useful for notifying applications on Android devices that there is new data to be fetched from the application servers. GCM supports messages with payload data upto 4 KB. GCM provides a 'send-to-sync' message capability that can be used to inform an application to sync data from the server.

Google Cloud Messaging for Chrome is another notification service from Google that allows messages to be delivered from the cloud to apps and extensions running in Chrome.

Windows Azure Notification Hubs

Windows Azure Notification Hubs is a push notification service from Microsoft that provides a common interface to send notifications to all major mobile platforms including Windows Store/Windows Phone 8, iOS, and Android. Platform specific infrastructures called Platform Notification Systems (PNS) are used to deliver notification messages. Devices register their PNS handles with the Notification Hub. Each notification hub contains credentials for each

supported PNS. These credentials are used to connect to the PNSs and send push notifications to the applications.

3.4.5 Media Services

Cloud service providers provide various types of media services that can be used by applications for manipulating, transforming or transcoding media such as images, videos, etc.

Amazon Elastic Transcoder

Amazon Elastic Transcoder is a cloud-based video transcoding service from Amazon. Elastic Transcoder can be used to convert video files from their source format into various other formats that can be played on devices such as desktops, mobiles, tablets, etc. Elastic Transcoder provides a number of pre-defined transcoding presets. Transcoding pipelines are used to perform multiple transcodes in parallel. Elastic Transcoder works with the Amazon S3 storage where the input and output video files are stored. Users can create transcoding jobs by specifying the input and output locations (on S3), preset to use, and optional thumbnails and job specific parameters such as frame-rate.

Google Images Manipulation Service

Google Images Manipulation service is a part of the Google App Engine platform. Image Manipulation service provides the capability to resize, crop, rotate, flip and enhance images. The Images service can accept image data directly from the App Engine apps, or from Google Blobstore or Google Cloud Storage. Image Service accepts images in various formats including JPEG, PNG, WEBP, GIF, BMP, TIFF and ICO formats and can return transformed images in JPEG, WEBP and PNG formats.

Windows Azure Media Services

Windows Azure Media Services provides the various media services such as encoding & format conversion, content protection and on-demand and live streaming capabilities. Azure Media Services provides applications the capability to build media workflows for uploading, storing, encoding, format conversion, content protection, and media delivery. To use Azure Media Services, you can create jobs that process media content in several ways such as encoding, encrypting, doing format conversions, etc. Each Media Services job has one or more tasks. Each task has preset string, an input asset and an output asset. Media assets in the Azure Media Service can be delivered either by download or by streaming.

3.5 Content Delivery Services

Cloud-based content delivery service include Content Delivery Networks (CDNs). A CDN is a distributed system of servers located across multiple geographic locations to serve content to end-users with high availability and high performance. CDNs are useful for serving static content such as text, images, scripts, etc., and streaming media. CDNs have a number of edge locations deployed in multiple locations, often over multiple backbones. Requests for static or streaming media content that is served by a CDN are directed to the nearest edge location. CDNs cache the popular content on the edge servers which helps in reducing bandwidth costs and improving response times.

3.5.1 Amazon CloudFront

Amazon CloudFront is a content delivery service from Amazon. CloudFront can be used to deliver dynamic, static and streaming content using a global network of edge locations. The content in CloudFront is organized into distributions. Each distribution specifies the original location of the content to be delivered which can be an Amazon S3 bucket, an Amazon EC2 instance, or an Elastic Load Balancer, or your own origin server. Distributions can be accessed by their domain names. Figure 3.16 shows a screenshot of the Amazon CloudFront console. CloudFront helps in improving the performance of websites in several ways: (1) by caching the static content (such as JavaScript, CSS, images, etc.) at the edge location, (2) by proxying requests for dynamic or interactive content back to the origin (such as an Amazon EC2 instance) running in the AWS cloud.

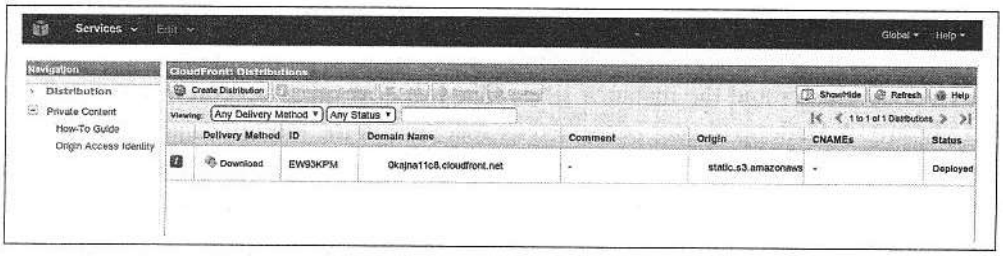


Figure 3.16: Screenshot of Amazon CloudFront console

3.5.2 Windows Azure Content Delivery Network

Windows Azure Content Delivery Network (CDN) is the content delivery service from Microsoft. Azure CDN caches Windows Azure blobs and static content at the edge locations to improve the performance of web sites. Azure CDN can be enabled on a Windows Azure storage account.

3.6 Analytics Services

Cloud-based analytics services allow analyzing massive data sets stored in the cloud either in cloud storages or in cloud databases using programming models such as MapReduce. Using cloud analytics services applications can perform data-intensive tasks such as such as data mining, log file analysis, machine learning, web indexing, etc.

3.6.1 Amazon Elastic MapReduce

Amazon Elastic MapReduce is the MapReduce service from Amazon based the Hadoop framework running on Amazon EC2 and Amazon S3. EMR supports various job types:

- Custom JAR: Custom JAR job flow runs a Java program that you have uploaded to Amazon S3.
- Hive program: Hive is a data warehouse system for Hadoop. You can use Hive to process data using the SQL-like language, called Hive-QL. You can create a Hive job flow with EMR which can either be an interactive Hive job or a Hive script.
- Streaming job: Streaming job flow runs a single Hadoop job consisting of map and reduce functions implemented in a script or binary that you have uploaded to Amazon

S3. You can write map and reduce scripts in Ruby, Perl, Python, PHP, R, Bash, or C++.

- Pig programs: Apache Pig is a platform for analyzing large data sets that consists of a high-level language (Pig Latin) for expressing data analysis programs, coupled with infrastructure for evaluating these programs. You can create a Pig job flow with EMR which can either be an interactive Pig job or a Pig script.
- HBase: HBase is a distributed, scalable, No-SQL database built on top of Hadoop. EMR allows you to launch an HBase cluster. HBase can be used for various purposes such as referencing data for Hadoop analytics, real-time log ingestion and batch log analytics, etc.

Figure 3.17 shows a screenshot of the Amazon EMR console. The EMR console provides a simple wizard for creating new MapReduce job flows. To create a MapReduce job you enter the job name, select the streaming option for the job flow, specify the locations of input, output and the mapper and reducer programs and specify the number of nodes to use in the Hadoop cluster and the instance sizes. The job flow takes several minutes to launch and configure. A Hadoop cluster is created as specified in the job flow and the MapReduce program specified in the input is executed. On completion of the MapReduce job, the results are copied to the output location specified and the Hadoop cluster is terminated.

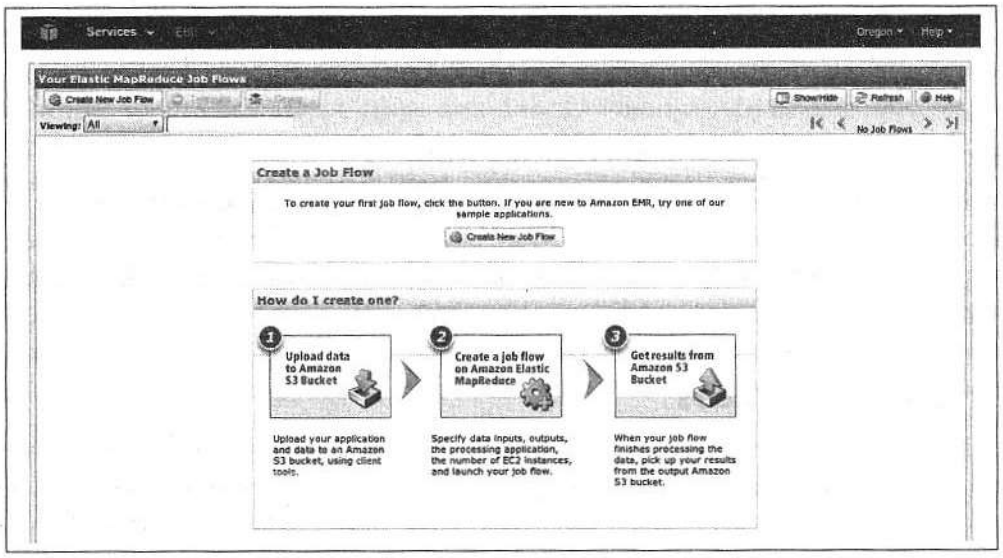


Figure 3.17: Screenshot of Amazon EMR console

3.6.2 Google MapReduce Service

Google MapReduce Service is a part of the App Engine platform. App Engine MapReduce is optimized for App Engine environment and provides capabilities such as automatic sharding for faster execution, standard data input readers for iterating over blob and datastore data, standard output writers, etc. The MapReduce service can be accessed using the Google MapReduce API. To execute a MapReduce job a MapReduce pipeline object is instantiated within the App Engine application. MapReduce pipeline specifies the mapper, reducer, data input reader, output writer.

3.6.3 Google BigQuery

Google BigQuery is a service for querying massive datasets. BigQuery allows querying datasets using SQL-like queries. The BigQuery queries are run against append-only tables and use the processing power of Google's infrastructure for speeding up queries. To query data, it is first loaded into BigQuery using the BigQuery console or BigQuery command line tool or BigQuery API. Data can be either in CSV or JSON format. The uploaded data can be queried using BigQuery's SQL dialect.

3.6.4 Windows Azure HDInsight

Windows Azure HDInsight is an analytics service from Microsoft. HDInsight deploys and provisions Hadoop clusters in the Azure cloud and makes Hadoop available as a service. HDInsight Service uses Windows Azure Blob Storage as the default file system. HDInsight provides interactive consoles for both JavaScript and Hive.

3.7 Deployment & Management Services

Cloud-based deployment & management services allow you to easily deploy and manage applications in the cloud. These services automatically handle deployment tasks such as capacity provisioning, load balancing, auto-scaling, and application health monitoring.

3.7.1 Amazon Elastic Beanstalk

Amazon provides a deployment service called Elastic Beanstalk that allows you to quickly deploy and manage applications in the AWS cloud. Elastic Beanstalk supports Java, PHP, .NET, Node.js, Python, and Ruby applications. With Elastic Beanstalk you just need to upload the application and specify configuration settings in a simple wizard and the service automatically handles instance provisioning, server configuration, load balancing and monitoring. Figure 3.18 shows a screenshot of the Amazon Elastic Beanstalk console. The launch wizard allows you to specify the environment details such as name, URL, application file, container type, instance type, etc. When the environment is launched Elastic Beanstalk automatically creates a new load balancer, launches and configures application and database servers as specified in the launch wizard, and deploys the application package on the application servers. The load balancer sits in front of the application servers which are a part of an Amazon Auto Scaling group. If the load on the application increases, Auto Scaling automatically launches new application servers to handle the increased load. If the load decreases, Auto Scaling stops additional instances and leaves at least one instance running.

3.7.2 Amazon CloudFormation

Amazon CloudFormation is a deployment management service from Amazon. With CloudFront you can create deployments from a collection of AWS resources such as Amazon Elastic Compute Cloud, Amazon Elastic Block Store, Amazon Simple Notification Service, Elastic Load Balancing and Auto Scaling. A collection of AWS resources that you want to manage together are organized into a stack. CloudFormation stacks are created from CloudFormation templates. You can create your own templates or use the predefined templates. The AWS infrastructure requirements for the stack are specified in the template. Figure 3.19 shows a screenshot of the Amazon CloudFormation console.

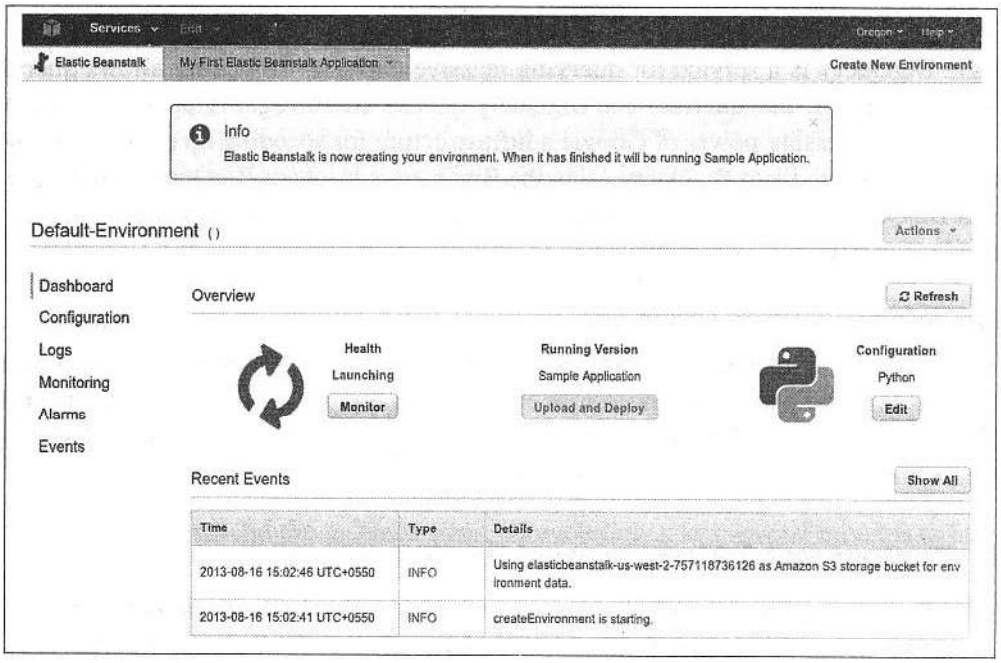


Figure 3.18: Screenshot of Amazon Elastic Beanstalk console

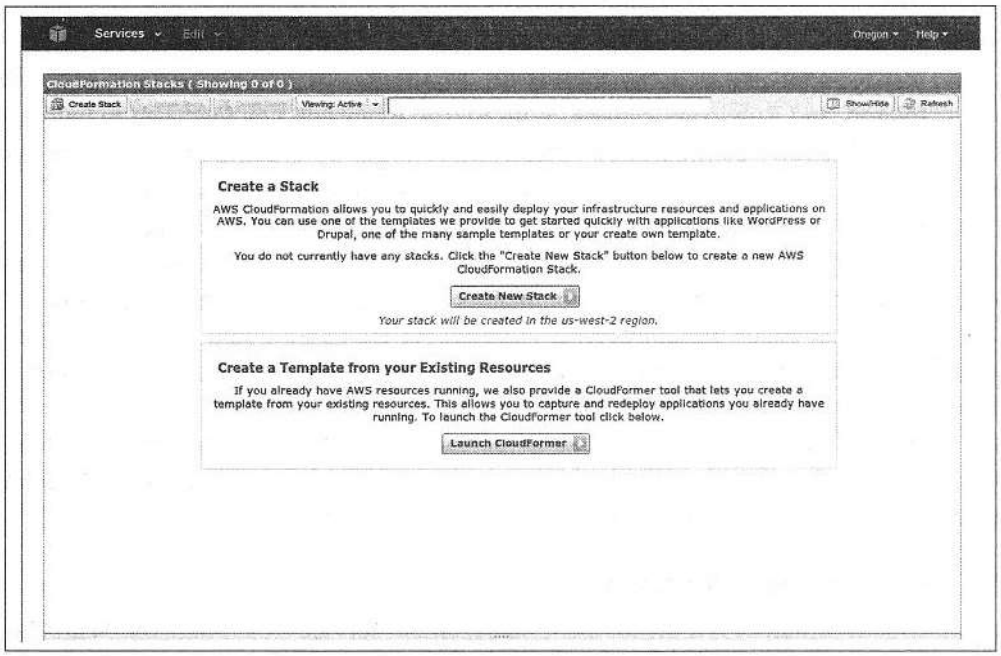


Figure 3.19: Screenshot of Amazon CloudFormation console

3.8 Identity & Access Management Services

Identity & Access Management (IDAM) services allow managing the authentication and authorization of users to provide secure access to cloud resources. IDAM services are useful

for organizations which have multiple users who access the cloud resources. Using IDAM services you can manage user identifiers, user permissions, security credentials and access keys.

3.8.1 Amazon Identity & Access Management

AWS Identity and Access Management (IAM) allows you to manage users and user permissions for an AWS account. With IAM you can manage users, security credentials such as access keys, and permissions that control which AWS resources users can access. Using IAM you can control what data users can access and what resources users can create. IAM also allows you to control creation, rotation, and revocation security credentials of users. Figure 3.20 shows a screenshot of the Amazon Identity & Access Management console.

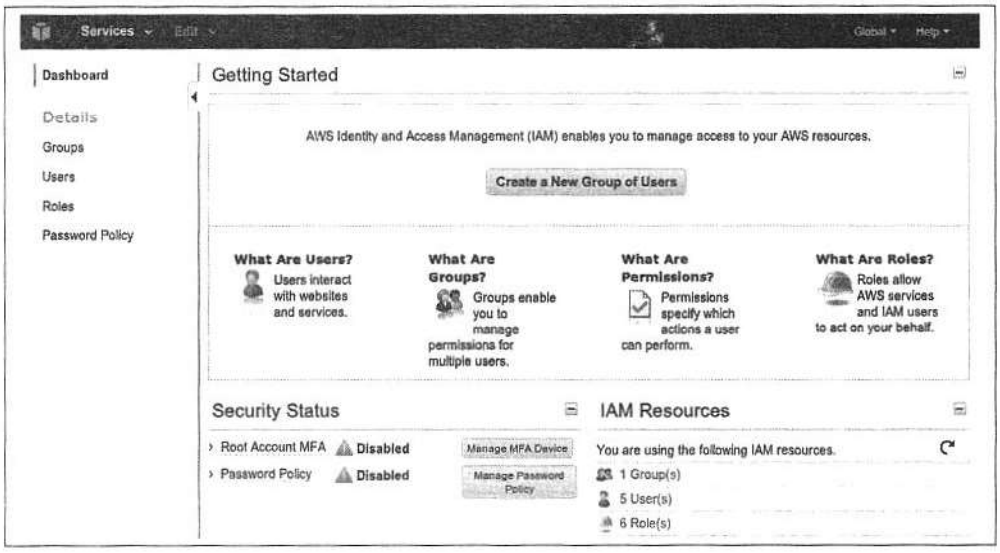


Figure 3.20: Screenshot of Amazon IAM console

3.8.2 Windows Azure Active Directory

Windows Azure Active Directory is an Identity & Access Management Service from Microsoft. Azure Active Directory provides a cloud-based identity provider that easily integrates with your on-premises active directory deployments and also provides support for third party identity providers. By integrating your on-premises active directory, you can authenticate users to Windows Azure with their existing corporate credentials. With Azure Active Directory you can control access to your applications in Windows Azure.

3.9 Open Source Private Cloud Software

In the previous sections you learned about popular public cloud platforms. This section covers open source cloud software that can be used to build private clouds.

3.9.1 CloudStack

Apache CloudStack is an open source cloud software that can be used for creating private cloud offerings [15]. CloudStack manages the network, storage, and compute nodes that make up a cloud infrastructure. A CloudStack installation consists of a Management Server and the cloud infrastructure that it manages. The cloud infrastructure can be as simple as one host running the hypervisor or a large cluster of hundreds of hosts. The Management Server allows you to configure and manage the cloud resources. Figure 3.21 shows the architecture of CloudStack which is basically the Management Server. The Management Server manages one or more zones where each zone is typically a single datacenter. Each zone has one or more pods. A pod is a rack of hardware comprising of a switch and one or more clusters. A cluster consists of one or more hosts and a primary storage. A host is a compute node that runs guest virtual machines. The primary storage of a cluster stores the disk volumes for all the virtual machines running on the hosts in that cluster. Each zone has a secondary storage that stores templates, ISO images, and disk volume snapshots.

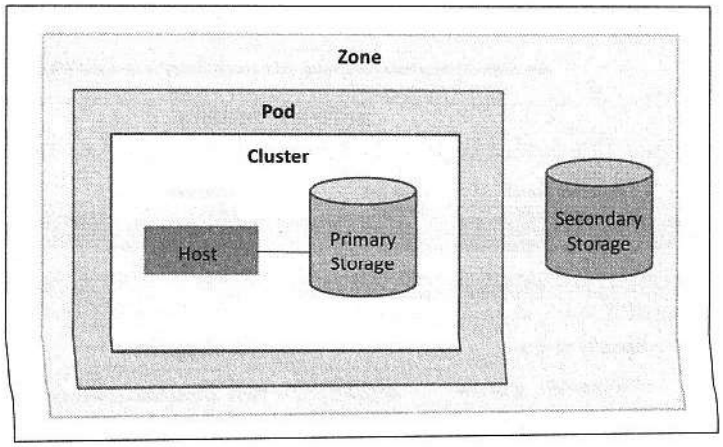


Figure 3.21: CloudStack architecture

3.9.2 Eucalyptus

Eucalyptus is an open source private cloud software for building private and hybrid clouds that are compatible with Amazon Web Services (AWS) APIs [16]. Figure 3.22 shows the architecture of Eucalyptus. The Node Controller (NC) hosts the virtual machine instances and manages the virtual network endpoints. The cluster-level (availability-zone) consists of three components - Cluster Controller (CC), Storage Controller (SC) and VMWare Broker. The CC manages the virtual machines and is the front-end for a cluster. The SC manages the Eucalyptus block volumes and snapshots to the instances within its specific cluster. SC is equivalent to AWS Elastic Block Store (EBS). The VMWare Broker is an optional component that provides an AWS-compatible interface for VMware environments. At the cloud-level there are two components - Cloud Controller (CLC) and Walrus. CLC provides an administrative interface for cloud management and performs high-level resource scheduling, system accounting, authentication and quota management.

Walrus is equivalent to Amazon S3 and serves as a persistent storage to all of the virtual machines in the Eucalyptus cloud. Walrus can be used as a simple Storage-as-a-Service

solution.

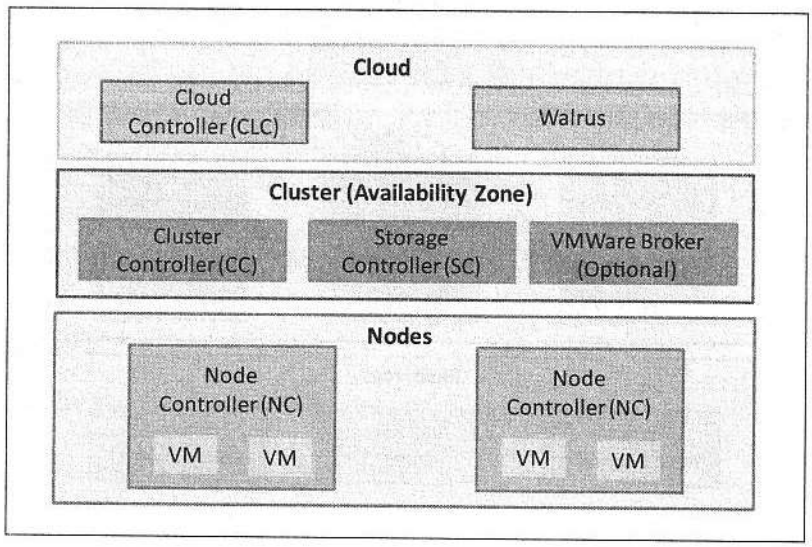


Figure 3.22: Eucalyptus architecture

3.9.3 OpenStack

OpenStack is a cloud operating system comprising of a collection of interacting services that control computing, storage, and networking resources [17]. Figure 3.23 shows the architecture of OpenStack. The OpenStack compute service (called nova-compute) manages networks of virtual machines running on nodes, providing virtual servers on demand. The network service (called nova-networking) provides connectivity between the interfaces of other OpenStack services. The volume service (Cinder) manages storage volumes for virtual machines. The object storage service (swift) allows users to store and retrieve files. The identity service (keystone) provides authentication and authorization for other services. The image registry (glance) acts as a catalog and repository for virtual machine images. The OpenStack scheduler (nova-scheduler) maps the nova-API calls to the appropriate OpenStack components. The scheduler takes the virtual machine requests from the queue and determines where they should run. The messaging service (rabbit-mq) acts as a central node for message passing between daemons. Orchestration activities such as running an instance are performed by the nova-api which accepts and responds to end user compute API calls. The OpenStack dashboard (called horizon) provides web-based interface for managing OpenStack services.

Summary

In this chapter you learned about the cloud reference model that includes the infrastructure/facilities layer, hardware layer, virtualization layer, virtual machines, platform & middleware, service management layer and applications layer and security layer. Cloud computing services can be of various types including compute service, storage services, database services application services, content delivery services, analytics services deployment & management services, identity & access management services, etc. Compute services provide dynami-