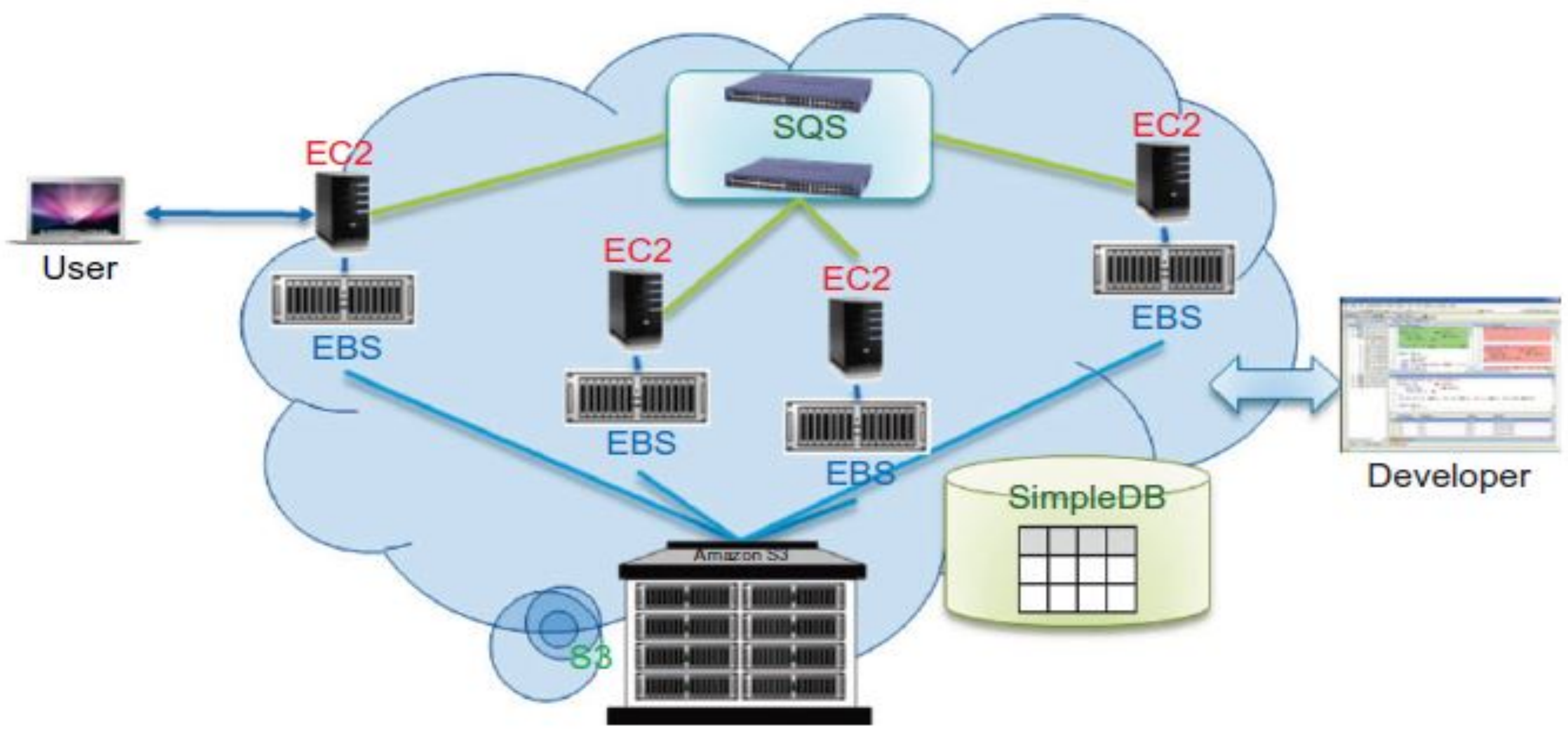
# Computing Platforms and Technologies:

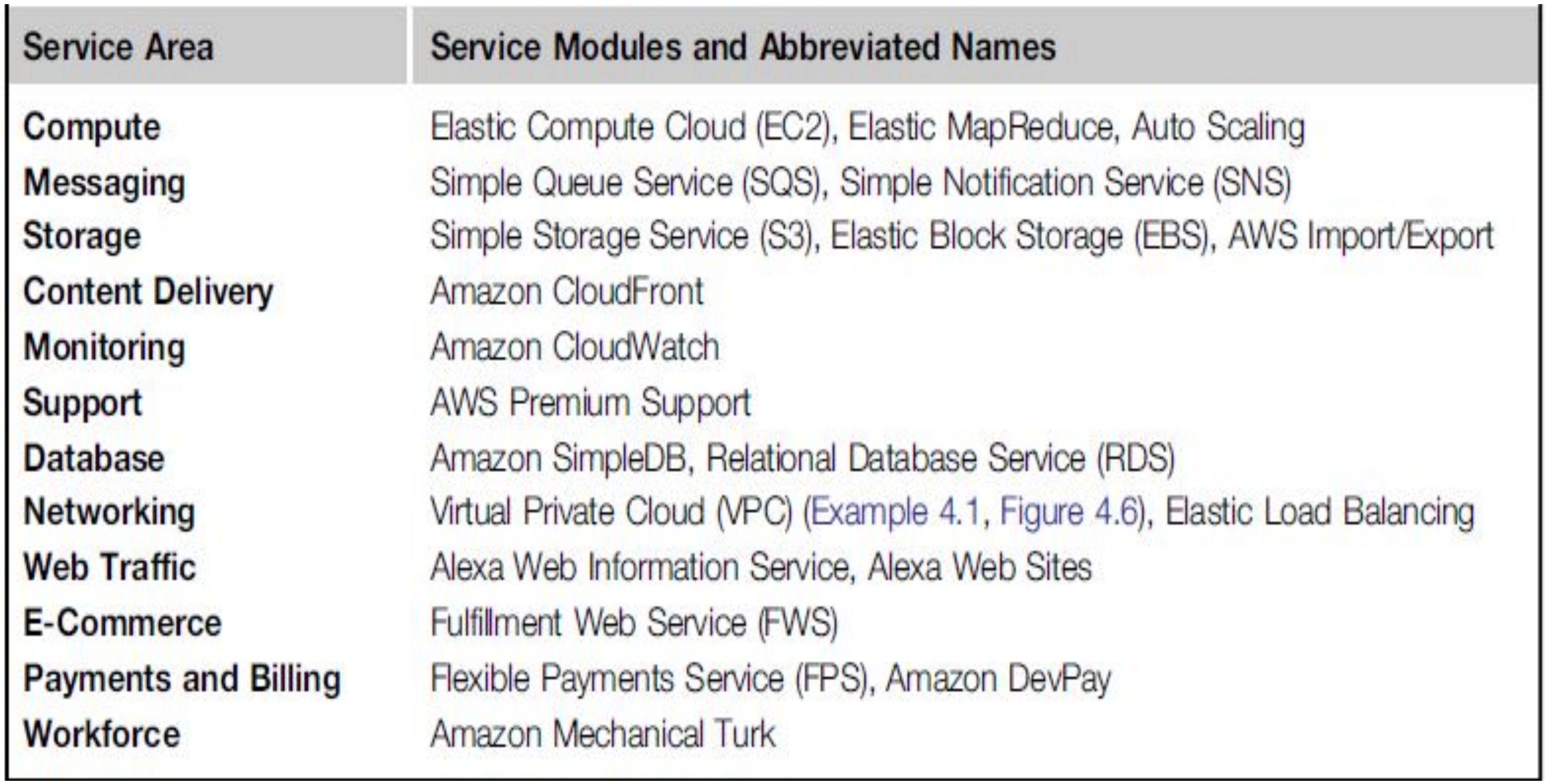
## Amazon web services (AWS):



Amazon Cloud Computing Infrastructure

# Computing Platforms and Technologies:

## Amazon web services (AWS):



| Service Area | Service Modules and Abbreviated Names |
|---|---|
| Compute | Elastic Compute Cloud (EC2), Elastic MapReduce, Auto Scaling |
| Messaging | Simple Queue Service (SQS), Simple Notification Service (SNS) |
| Storage | Simple Storage Service (S3), Elastic Block Storage (EBS), AWS Import/Export |
| Content Delivery | Amazon CloudFront |
| Monitoring | Amazon CloudWatch |
| Support | AWS Premium Support |
| Database | Amazon SimpleDB, Relational Database Service (RDS) |
| Networking | Virtual Private Cloud (VPC) (Example 4.1, Figure 4.6), Elastic Load Balancing |
| Web Traffic | Alexa Web Information Service, Alexa Web Sites |
| E-Commerce | Fulfillment Web Service (FWS) |
| Payments and Billing | Flexible Payments Service (FPS), Amazon DevPay |
| Workforce | Amazon Mechanical Turk |

**AWS Offerings**

## Computing Platforms and Technologies:

### Amazon web services (AWS):

- Amazon has been a leader in providing public cloud services.
- Amazon applies the IaaS model in providing its services.
- AWS offers comprehensive cloud IaaS services ranging from virtual compute, storage, and networking to complete computing stacks.
- AWS is mostly known for its compute and Storage-on-demand services, namely Elastic Compute Cloud (EC2) and Simple Storage Service (S3).
- EC2 provides users with customizable virtual hardware that can be used as the base infrastructure for deploying computing systems on the cloud.
- EC2 provides the virtualized platforms to the host VMs where the cloud application can run.
- It is possible to choose from a large variety of virtual hardware configurations, including GPU and cluster instances.

# Computing Platforms and Technologies:

## Amazon web services (AWS):

- EC2 instances are deployed either by using the AWS console, which is a comprehensive Web portal for accessing AWS services, or by using the Web services API available for several programming languages.
- EC2 also provides the capability to save a specific running instance as an image, thus allowing users to create their own templates for deploying systems.
- S3 (Simple Storage Service) provides the object-oriented storage service for users.
- S3 delivers persistent storage on demand.
- S3 is organized into buckets.
- Buckets are containers of objects that are stored in binary form and can be enriched with attributes.
- Users can access their objects through SOAP with either browsers or other client programs which support the SOAP standard.
- Users can store objects of any size, from simple files to entire disk images, and have them accessible from everywhere.

## Computing Platforms and Technologies:

### Amazon web services (AWS):

- Besides EC2 and S3, a wide range of services can be leveraged to build virtual computing systems including networking support, caching systems, DNS, database (relational and not) support, and others.
- EBS (Elastic Block Storage) provides the block storage interface which can be used to support traditional applications.
- SQS stands for Simple Queue Service, and its job is to ensure a reliable message service between two processes.
- The message can be kept reliably even when the receiver processes are not running.
- Elastic Load Balancer (ELB) automatically distributes incoming application traffic across multiple Amazon EC2 instances and allows user to avoid non-operating nodes and to equalize load on functioning images.
- Both auto-scaling and ELB are enabled by CloudWatch which monitors running instances.
- CloudWatch is a web service that provides monitoring for AWS cloud resources, starting with Amazon EC2.

## Computing Platforms and Technologies:

### Amazon web services (AWS):

- CloudWatch provides customers with visibility into resource utilization, operational performance, and overall demand patterns, including metrics such as CPU utilization, disk reads and writes, and network traffic.
- AWS Import/Export allows one to ship large volumes of data to and from EC2 by shipping physical disks.
- Amazon DevPay is a simple-to-use online billing and account management service that makes it easy for businesses to sell applications that are built into or run on top of AWS.
- Flexible Payments Service (FPS) provides developers of commercial systems on AWS with a convenient way to charge Amazon’s customers that use such services built on AWS.

## Computing Platforms and Technologies:

### Google App Engine (GAE):

- Google has the world's largest search engine facilities.
- Google has hundreds of data centers and has installed more than 460,000 servers worldwide.
- Data items are stored in text, images, and video and are replicated to tolerate faults or failures.
- In 2008, Google announced the Google App Engine (GAE) web application platform which is becoming a common platform for many small cloud service providers.
- Google App Engine (GAE) offers a PaaS supporting various cloud applications.
- GAE enables users to run their applications on a large number of data centers associated with Google's search engine operations.
- GAE is a scalable runtime environment mostly devoted to executing Web applications.

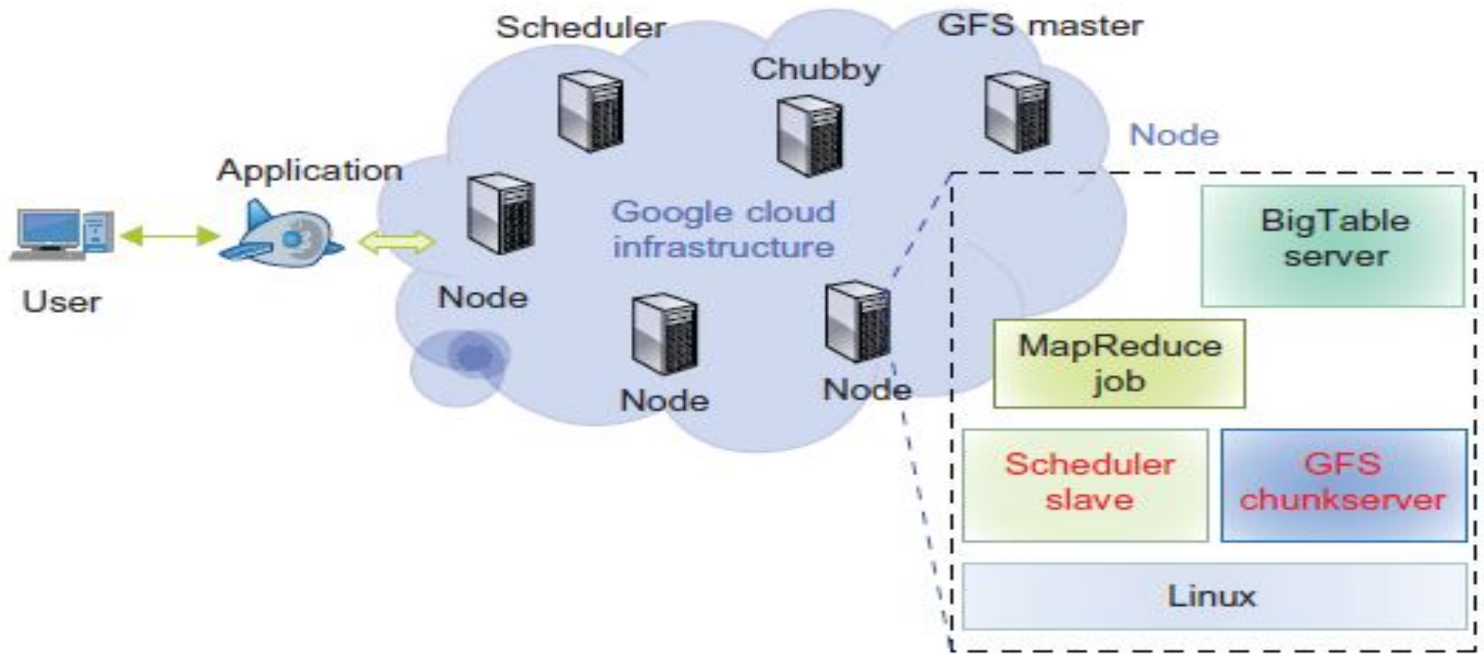
## Computing Platforms and Technologies:

### Google App Engine (GAE):

- Google App Engine take advantage of the large computing infrastructure of Google to dynamically scale as the demand varies over time.
- GAE provides both a secure execution environment and a collection of services that simplify the development of scalable and high-performance Web applications.
- GAE services also include in-memory caching, scalable data store and job queues.
- An in-memory cache is a data storage layer that sits between applications and databases to deliver responses with high speeds by storing data from earlier requests or copied directly from databases.
- Developers of GAE can build and test applications on their own machines using the AppEngine software development kit (SDK).
- Once development is complete, developers can easily migrate their application to online GAE platform and make the application available to the world.

# Computing Platforms and Technologies:

## Google App Engine (GAE):



Architecture of Google Cloud Infrastructure

- Google File System (GFS) is used for storing large amounts of data.
- MapReduce is used for application program development.
- Chubby is used for distributed application lock services.
- BigTable offers a storage service for accessing structured data.

## Computing Platforms and Technologies:

### Google App Engine (GAE):

- MapReduce is an application programming model developed by Google, which provides two fundamental operations for data processing: ***map*** and ***reduce.***
- The **map** operation transforms and synthesizes the input data provided by the user.
- The **reduce** operation aggregates the output obtained by the map operations.
- Chubby is a highly available and persistent distributed lock service and file system created by Google.
- Chubby manages locks for resources and stores configuration information for various distributed services throughout the Google cluster environment.
- The purpose of the lock service provided by chubby is to allow its clients to synchronize their activities.

# Computing Platforms and Technologies:

## Google App Engine (GAE):

- Third-party application providers can use GAE to build cloud applications for providing services.
- The GAE is not an infrastructure platform, but rather an application development platform for users.
- The GAE platform does not provide any IaaS services, unlike Amazon, which offers Iaas and PaaS.
- The GAE allows the user to deploy user-built applications on top of the cloud infrastructure that are built using the programming languages and software tools supported by the provider (e.g., Java, Python).

## Computing Platforms and Technologies:

### Google App Engine (GAE):

- **The GAE platform comprises the following five major components:**

1) The ***datastore*** offers object-oriented, distributed, structured data storage services based on BigTable techniques. The datastore secures data management operations.
2) The ***application runtime environment*** offers a platform for scalable web programming and execution. It supports two development languages: Python and Java.
3) The ***software development kit (SDK)*** is used for local application development. The SDK allows users to execute test runs of local applications and upload application code.
4) The ***administration console*** is used for easy management of user application development cycles, instead of for physical resource management.
5) The ***GAE web service infrastructure*** provides special interfaces to guarantee flexible use and management of storage and network resources by GAE.

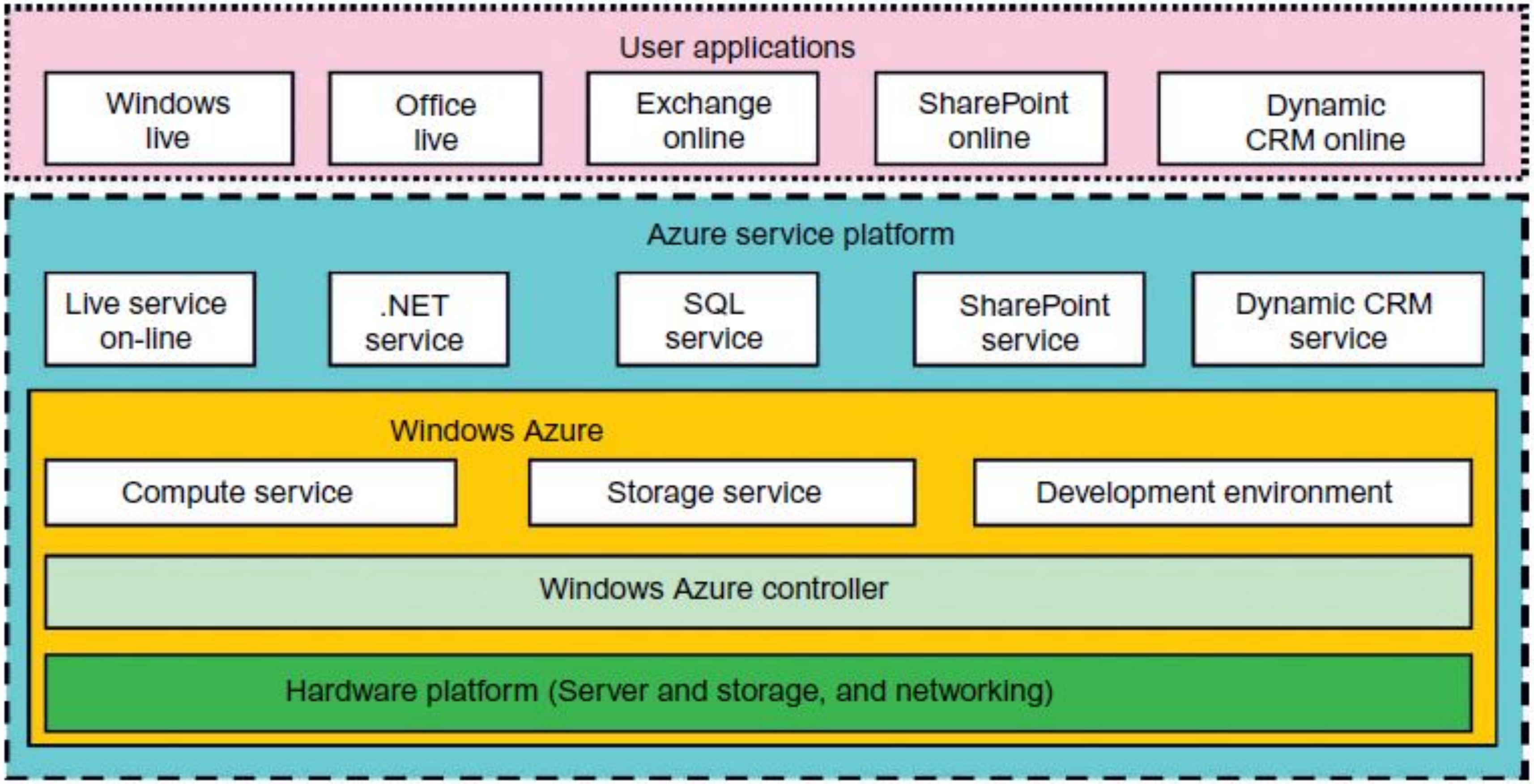
## Computing Platforms and Technologies:

### Google App Engine (GAE):

- Well-known GAE applications include the Google Search Engine, Google Docs, Google Earth, and Gmail.
- Third-party application providers can use GAE to build cloud applications for providing services.
- GAE provides storage service to store application-specific data in the Google infrastructure. The data can be persistently stored in the backend storage server while still providing the facility for queries, sorting, and even transactions similar to traditional database systems.

## Computing Platforms and Technologies:

### Microsoft Windows Azure:



Microsoft Windows Azure Platform for Cloud Computing

## Computing Platforms and Technologies:

### Microsoft Windows Azure:

- In 2008, Microsoft launched a Windows Azure platform to meet the challenges in cloud computing.
- Microsoft Azure is a cloud operating system and a platform for developing applications in the cloud.
- This platform is built over Microsoft data centers.
- It provides a scalable runtime environment for development of distributed applications.
- Windows Azure offers a cloud platform built on Windows OS and based on Microsoft virtualization technology.
- Azure manages all servers, storage, and network resources of the data center.
- Cloud-level services provided by the Azure platform:

1) **Live service:** Users can visit Microsoft Live applications and access the data involved across multiple machines concurrently.
2) **.NET service:** This package supports application development on local hosts and execution on cloud machines.

## Computing Platforms and Technologies:

### Microsoft Windows Azure:

3) **SQL Azure:** This function makes it easier for users to visit and use the relational database associated with the SQL server in the cloud.
4) **SharePoint service:** This provides a scalable and manageable platform for users to develop their special business applications in upgraded web services.
5) **Dynamic CRM service:** This provides software developers a business platform in managing CRM applications in financing, marketing, and sales and promotions.

- All these cloud services in Azure can interact with traditional Microsoft software applications, such as Windows Live, Office Live, Exchange online, SharePoint online, and dynamic CRM online.

## Computing Platforms and Technologies:

### Microsoft Windows Azure:

- Applications in Azure are organized around the concept of roles, which identify a distribution unit for applications.
- Currently, there are three types of role:

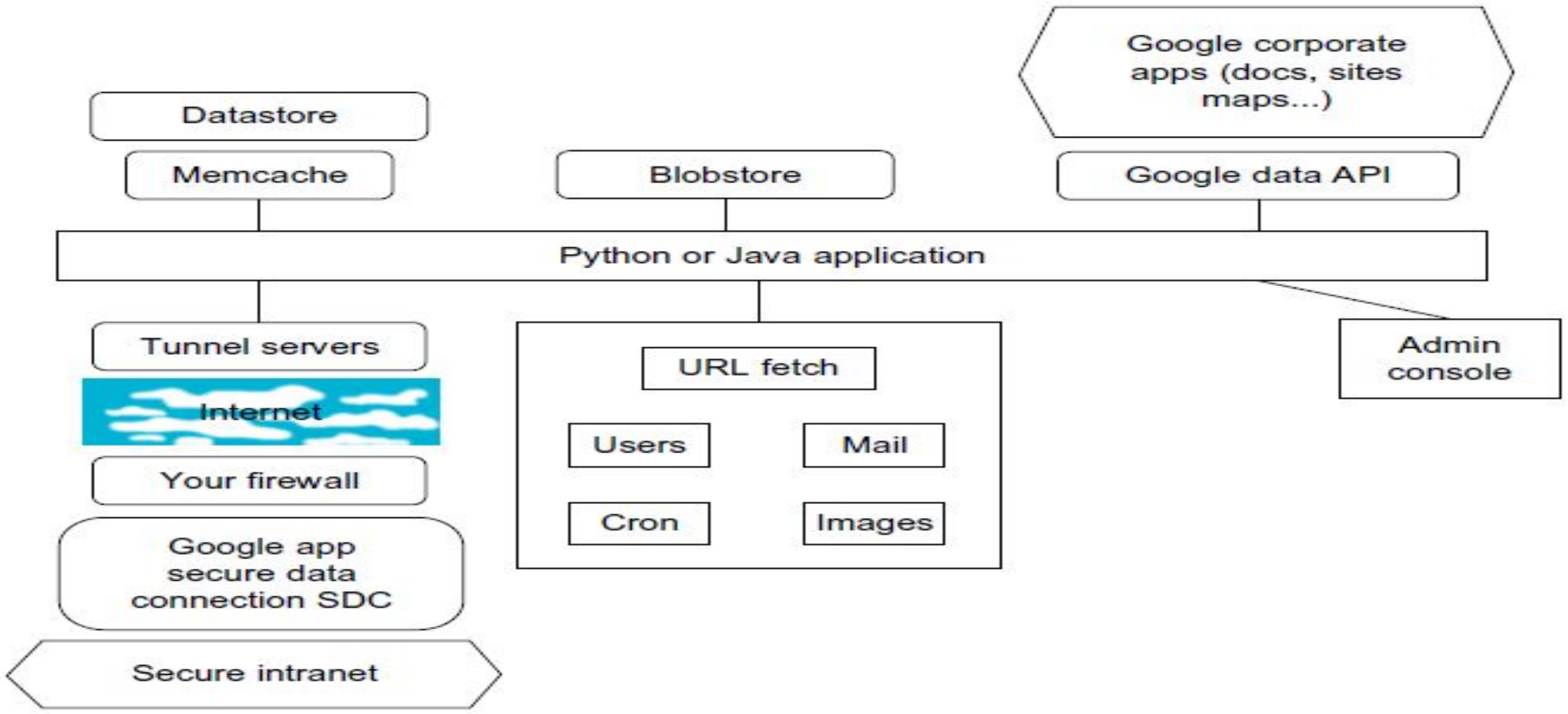
  **Web role**

  **Worker role**

  **Virtual machine role.**
- The Web role is designed to host a Web application.
- The worker role is a more generic container of applications and can be used to perform workload processing.
- The virtual machine role provides a virtual environment in which the computing stack can be fully customized, including the operating systems.
- Besides roles, Azure provides a set of additional services that complement application execution, such as support for storage, networking, caching and content delivery.

## Programming Support of Google App Engine:

### Programming the Google App Engine:



**Programming Environment for Google AppEngine**

- Above figure summarizes some key features of GAE programming model for two supported languages: Java and Python.
- A client environment that includes an Eclipse plug-in for Java, allows you to debug your GAE on your local machine.

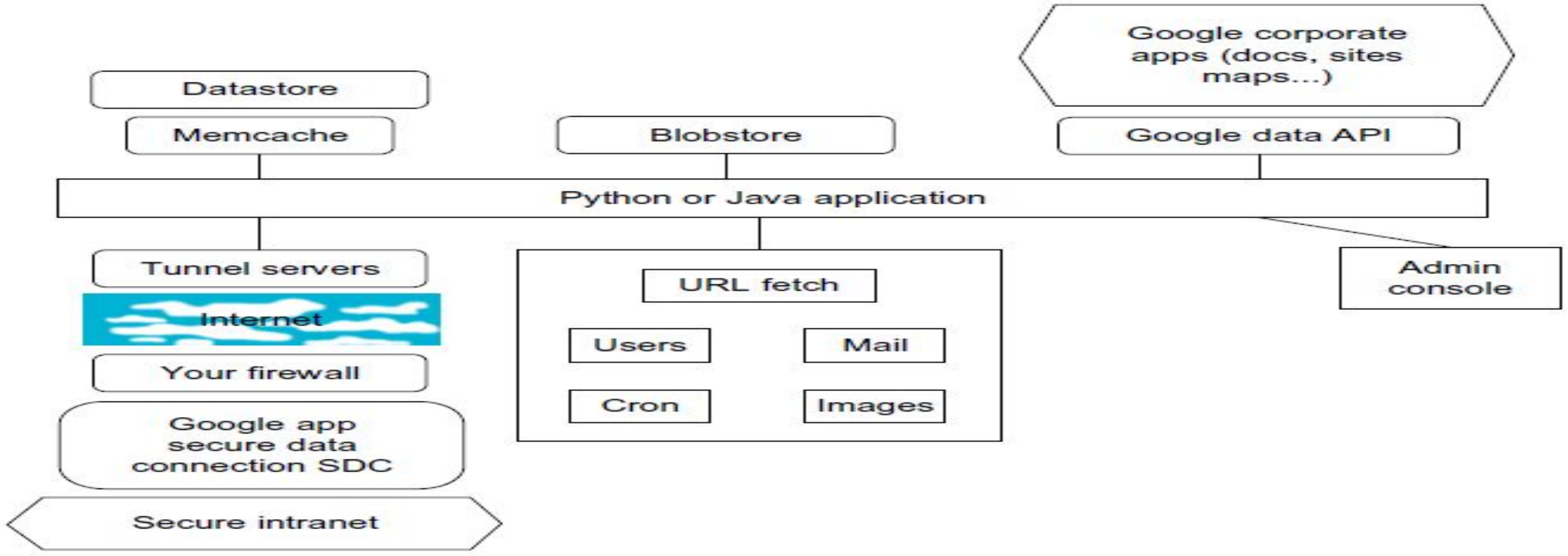
## Programming Support of Google App Engine:

### Programming the Google App Engine:

- Google Web Toolkit (GWT) is available for Java web application developers.
- Developers can use GWT, or any other language using a JVM-based interpreter or compiler, such as JavaScript or Ruby for web application development.
- Python is often used with frameworks such as Django and CherryPy, but Google also supplies a built in *webapp* Python environment.
- Java offers Java Data Object (JDO) and Java Persistence API (JPA) interfaces implemented by the open source Data Nucleus Access platform, while Python has a SQL-like query language called GQL.
- The ***datastore*** is a NOSQL data management system for entities that can be, at most, 1 MB in size and are labeled by a set of schema-less properties.

## Programming Support of Google App Engine:

### Programming the Google App Engine:

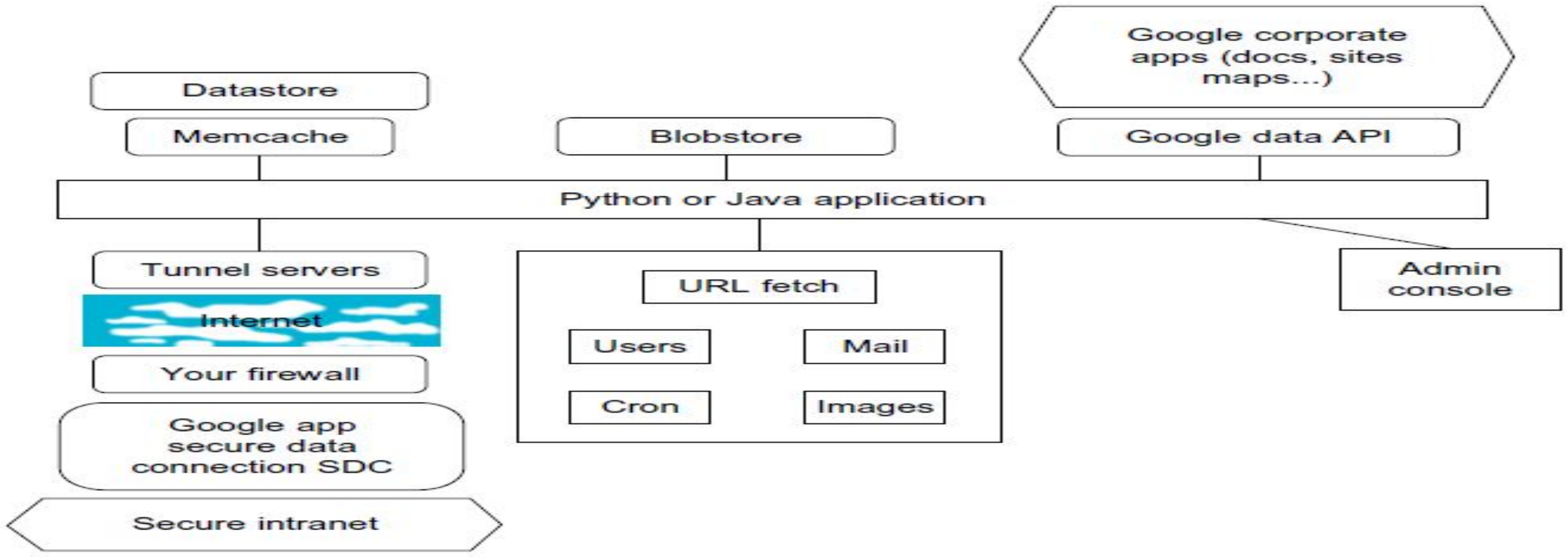


**Programming Environment for Google AppEngine**

- The ***datastore*** is strongly consistent and uses optimistic concurrency control.
- Your application can execute multiple data store operations in a single transaction which either all succeed or all fail together.
- The datastore implements transactions across its distributed network using "entity groups."
- A transaction manipulates entities within a single group.
- Entities of the same group are stored together for efficient execution of transactions.

## Programming Support of Google App Engine:

### Programming the Google App Engine:

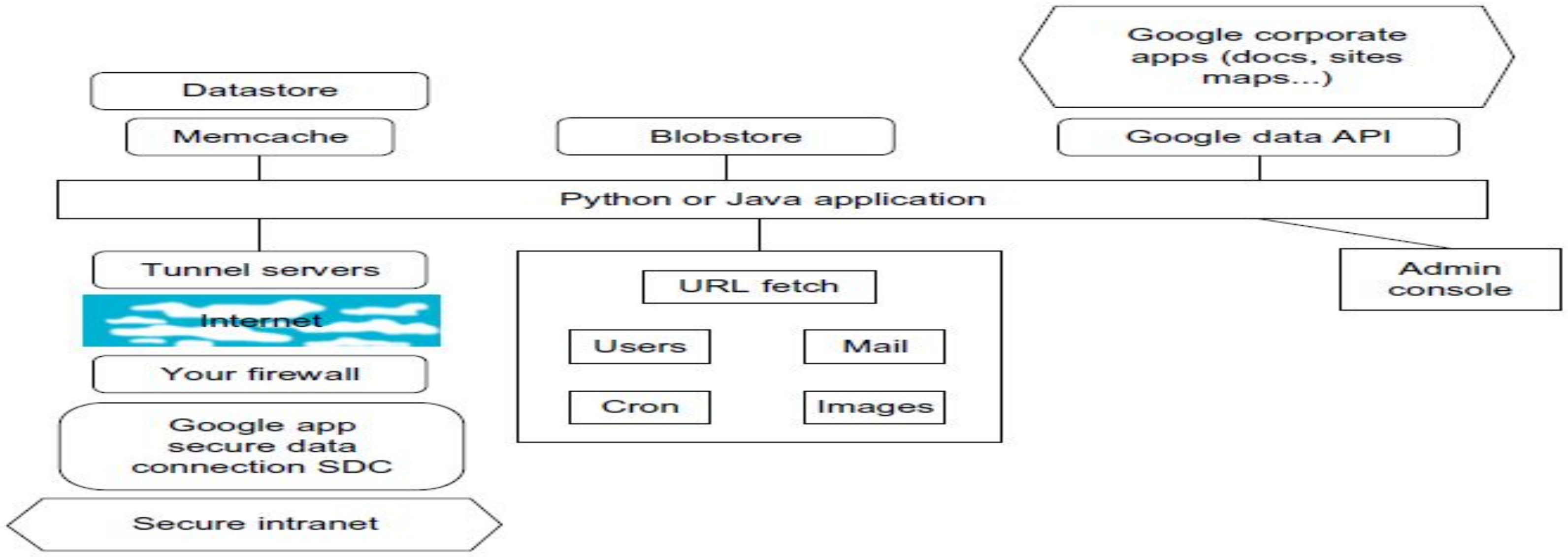


**Programming Environment for Google AppEngine**

- Your GAE application can assign entities to groups when the entities are created.
- The performance of the data store can be enhanced by in-memory caching using the ***memcache***, which can also be used independently of the data store.
- Recently, Google added the ***blobstore*** which is suitable for large files as its size limit is 2 GB.

## Programming Support of Google App Engine:

### Programming the Google App Engine:

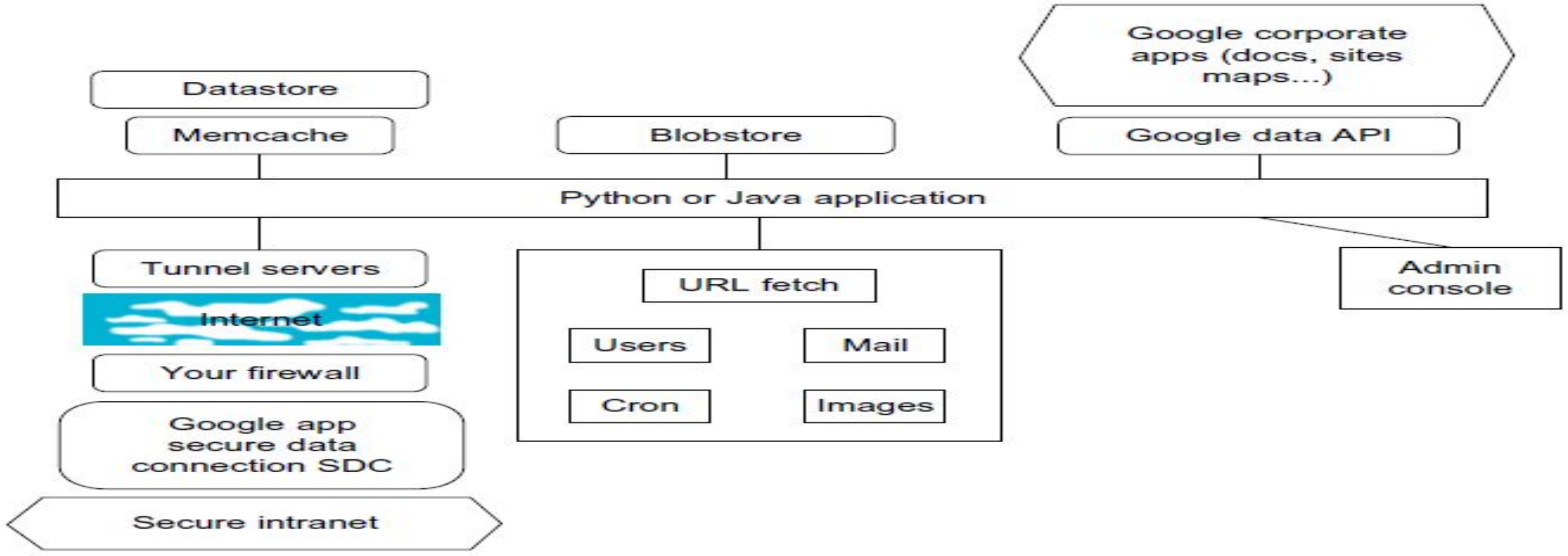


**Programming Environment for Google AppEngine**

- The ***Google Secure Data Connection (SDC)*** can tunnel through the Internet and link your intranet to an external GAE application.
- The ***URL Fetch*** operation provides the ability for applications to fetch resources and communicate with other hosts over the Internet using HTTP and HTTPS requests.
- Applications can access resources on the Internet, such as web services or other data, using GAE's URL fetch service.

# Programming Support of Google App Engine:

## Programming the Google App Engine:

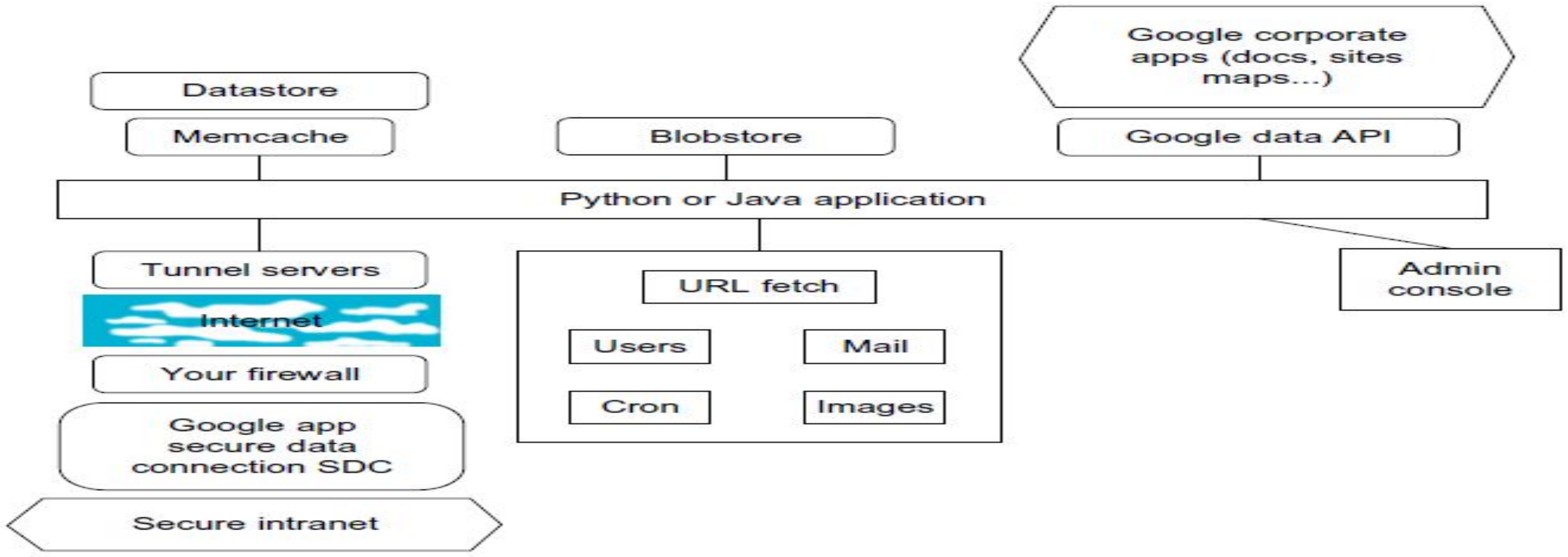


**Programming Environment for Google AppEngine**

- The URL fetch service retrieves web resources using the same high speed Google infrastructure that retrieves web pages for many other Google products.
- There are dozens of Google “corporate” facilities including maps, sites, groups, calendar, docs, and YouTube, among others.
- These support the ***Google Data API*** which can be used inside GAE.

## Programming Support of Google App Engine:

### Programming the Google App Engine:

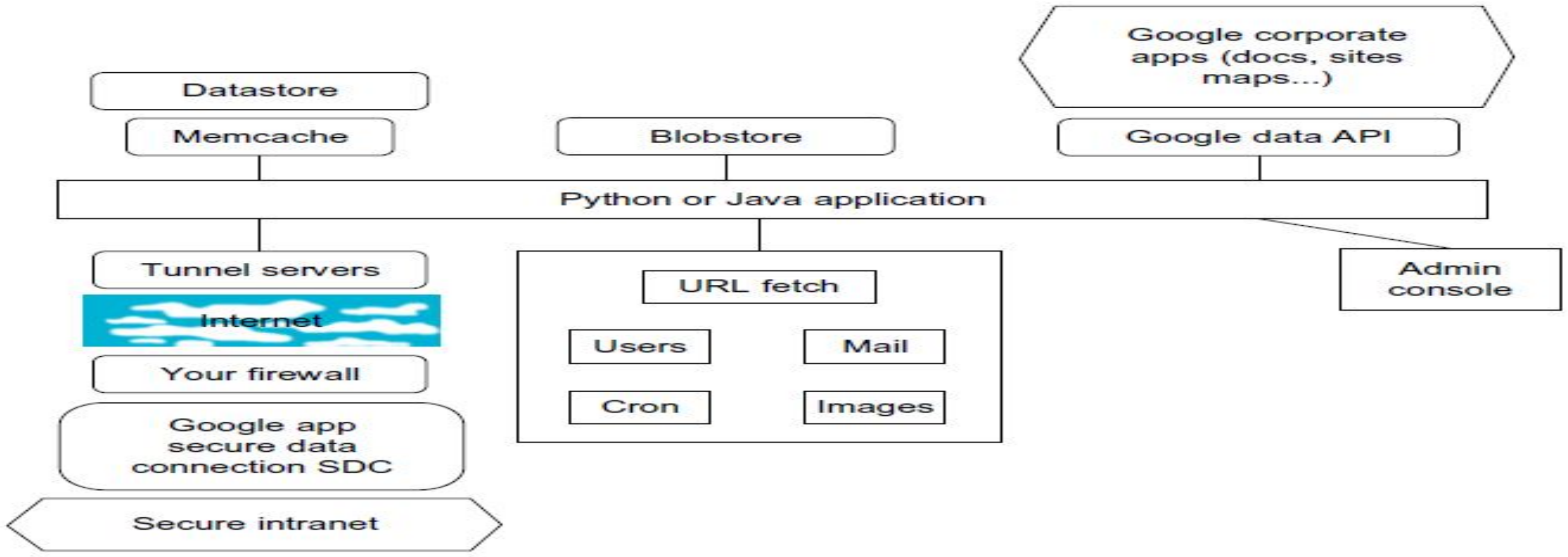


Programming Environment for Google AppEngine

- An application can use Google Accounts for ***user*** authentication.
- Google Accounts handles user account creation and sign-in, and a user that already has a Google account (such as a Gmail account) can use that account with your app.
- GAE provides the ability to manipulate image data using a dedicated ***Images*** service which can resize, rotate, flip, crop, and enhance images.

## Programming Support of Google App Engine:

### Programming the Google App Engine:



Programming Environment for Google AppEngine

- An application can perform tasks outside of responding to web requests. Your application can perform these tasks on a schedule that you configure, such as on a daily or hourly basis using "cron jobs," handled by the ***Cron*** service.
- The application can perform tasks added to a queue by the application itself, such as a background task created while handling a request.

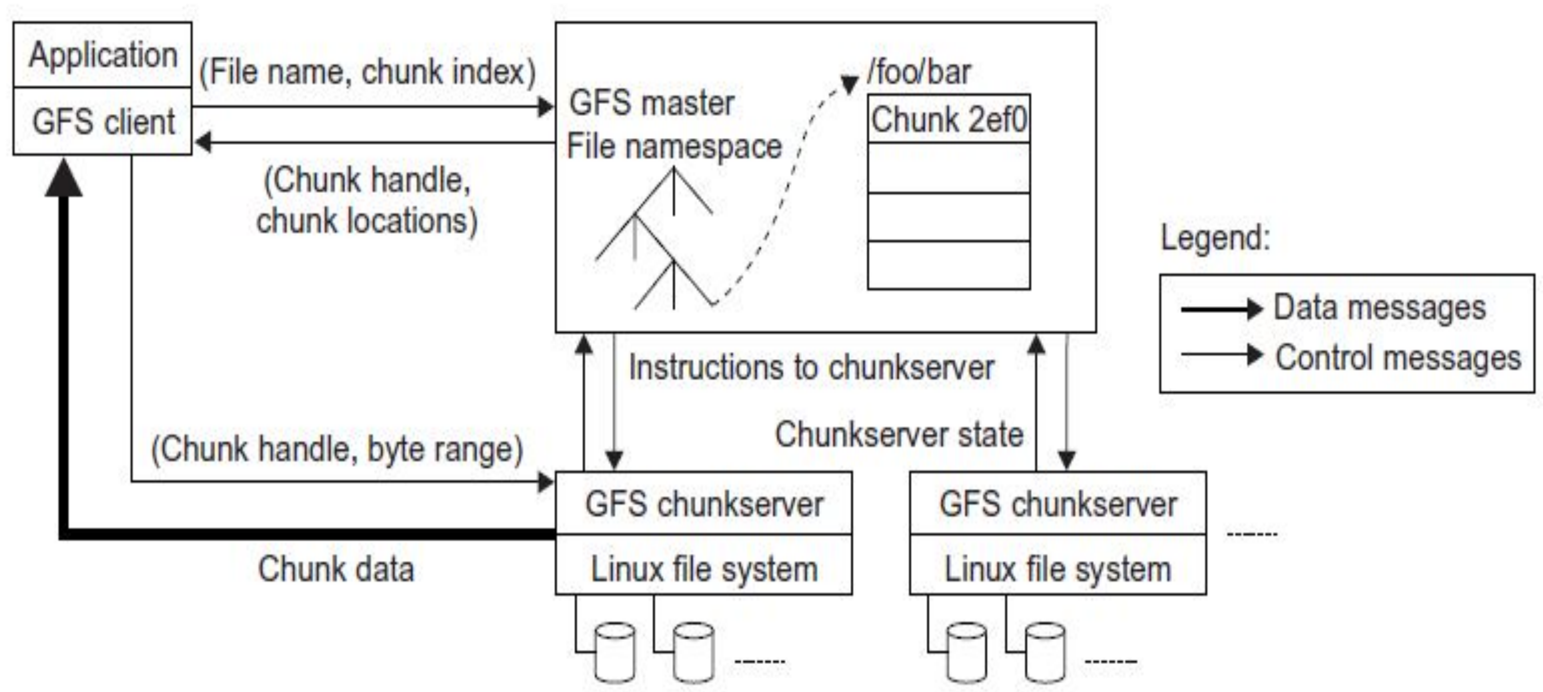
## Programming Support of Google App Engine:

### Google File System (GFS):

- GFS was built primarily as the fundamental storage service for Google's search engine.
- As the size of the web data that was crawled and saved was quite substantial, Google needed a distributed file system to redundantly store massive amounts of data on cheap and unreliable computers.
- GFS typically holds a large number of huge files, each 100MB or larger, with files that are multiple GB in size quite common.
- Google has chosen its file data block size to be 64MB instead of the 4 KB in typical traditional file systems.
- Reliability is achieved by using replications (i.e., each chunk or data block of a file is replicated across more than three chunk servers).
- A single master coordinates access as well as keeps the metadata. This decision simplified the design and management of the whole cluster.
- There is no data cache in GFS as large streaming reads and writes represent neither time nor space locality.

# Programming Support of Google App Engine:

## Google File System (GFS):

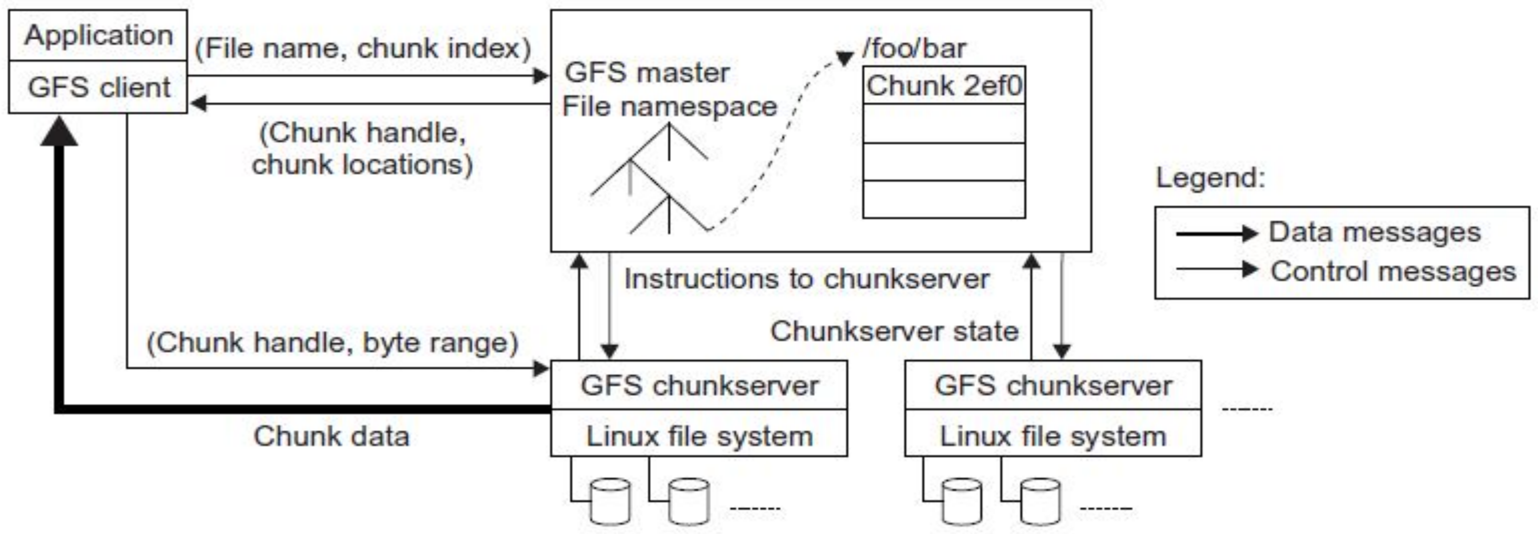


Architecture of Google File System

- There is a single master in the whole cluster.
- Other nodes act as the chunk servers for storing data, while the single master stores the metadata.
- The file system namespace and locking facilities are managed by the master.

## Programming Support of Google App Engine:

### Google File System (GFS):

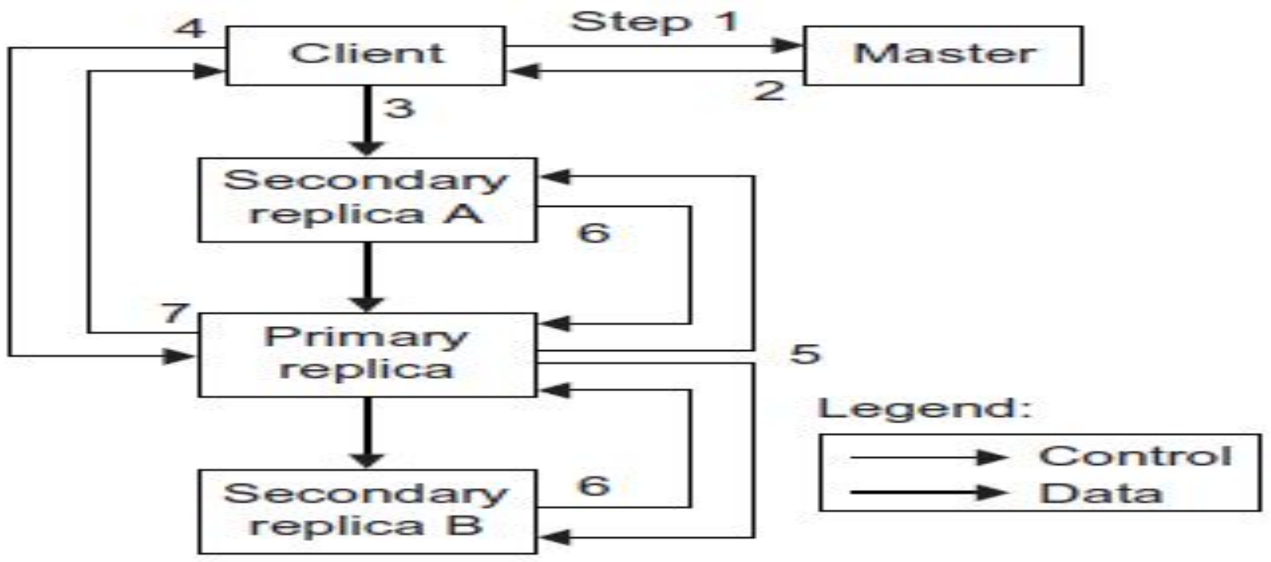


**Architecture of Google File System**

- The master periodically communicates with the chunk servers to collect management information as well as give instructions to the chunk servers to do work such as load balancing or fail recovery.
- The single GFS master could be the performance bottleneck and the single point of failure. To mitigate this, Google uses a shadow master to replicate all the data on the master.
- The shadow master handles the failure of the GFS master.

## Programming Support of Google App Engine:

### Google File System (GFS):



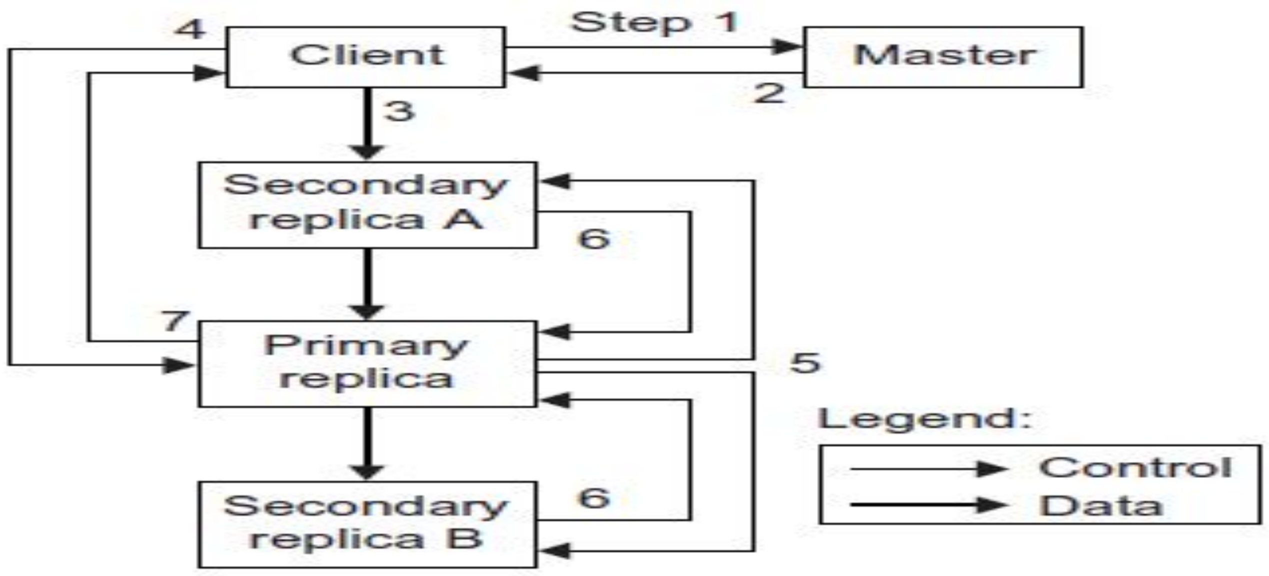
**Data mutation sequence in GFS**

Above figure shows the data mutation (write, append operations) in GFS.
The mutation takes the following steps:

1) The client asks the master which chunk server holds the current lease for the chunk and the locations of the other replicas.
2) The master replies with the identity of the primary and the locations of the other (secondary) replicas. The client caches this data for future mutations.

## Programming Support of Google App Engine:

### Google File System (GFS):



**Data mutation sequence in GFS**

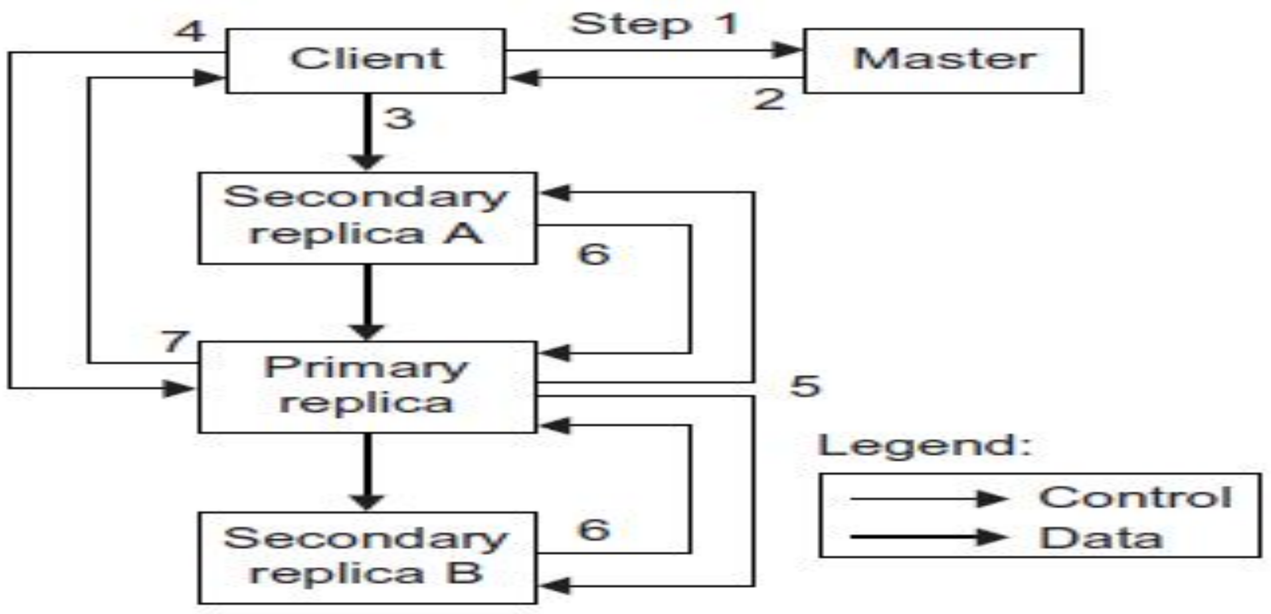
3) The client pushes the data to all the replicas. Each chunk server will store the data in an internal LRU buffer cache until the data is used or aged out.

4) Once all the replicas have acknowledged receiving the data, the client sends a write request to the primary. The request identifies the data pushed earlier to all the replicas.

5) The primary forwards the write request to all secondary replicas. Each secondary replica applies mutations in the same serial number order assigned by the primary.

# Programming Support of Google App Engine:

## Google File System (GFS):



**Data mutation sequence in GFS**

6) The secondaries all reply to the primary indicating that they have completed the operation.

7) The primary replies to the client. Any errors encountered at any replicas are reported to the client.

## Programming Support of Google App Engine:

### BigTable, Google’s NOSQL System:

- BigTable is designed to provide a service for storing and retrieving structured and semi-structured data.
- BigTable applications include storage of web pages, per-user data, and geographic locations.
- Web pages represent the URLs and their associated data, such as contents, links and page rank values.
- Per-user data has information for a specific user and includes such data as user preference settings, recent queries/search results, and the user’s e-mails.
- Geographic locations are used in Google’s well-known Google Earth software. Geographic locations include physical entities (shops, restaurants, etc.), roads, satellite image data, and user annotations.
- The scale of such data is incredibly large. It is not possible to solve such a large scale of structured or semi-structured data using a commercial database system.

## Programming Support of Google App Engine:

### BigTable, Google's NOSQL System:

- The design and implementation of the BigTable system has the following goals:

1) The applications want processes to be continuously updating different pieces of data and want access to the most current data at all times.
2) The database needs to support very high read/write rates and the scale might be millions of operations per second.
3) The application may need to examine data changes over time (e.g., contents of a web page over multiple crawls).

- BigTable provides a fault-tolerant and persistent database as in a storage service.
- The BigTable system is scalable, which means the system has thousands of servers, terabytes of in-memory data, petabytes of disk-based data, millions of reads/writes per second, and efficient scans.
- BigTable is used in many projects, including Google Search, Orkut, and Google Maps/Google Earth, among others.

## Programming on Amazon AWS:

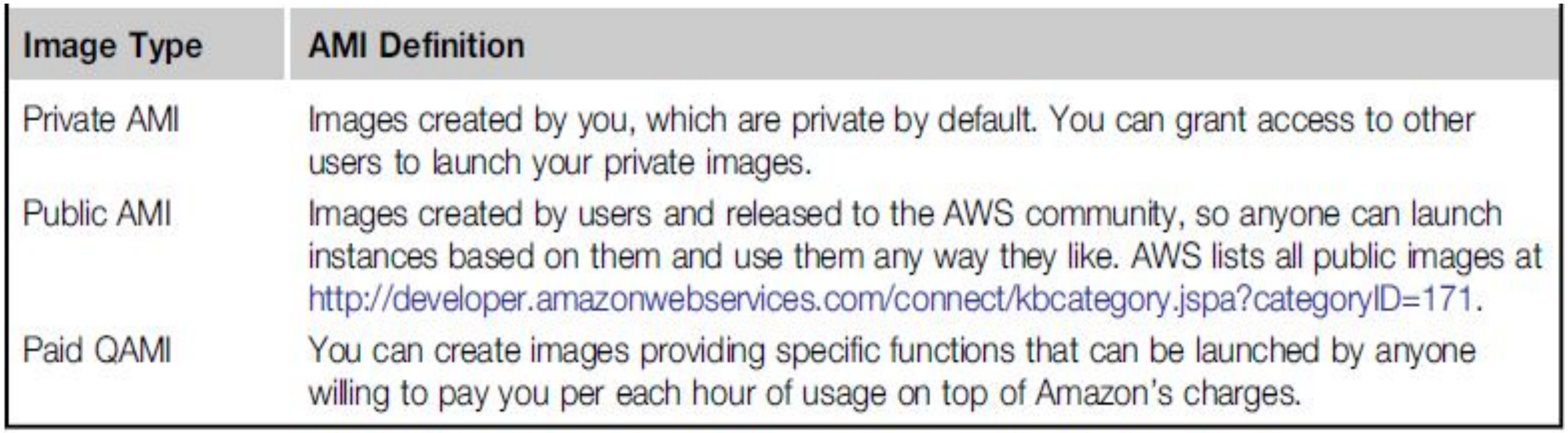
### Programming on Amazon EC2:

- Amazon was the first company to introduce VMs in application hosting.
- Customers can rent VMs instead of physical machines to run their own applications.
- By using VMs, customers can load any software of their choice.
- The elastic feature of such a service is that a customer can create, launch, and terminate server instances as needed, paying by the hour for active servers.
- Instances are often called Amazon Machine Images (AMIs) which are preconfigured with operating systems based on Linux or Windows, and additional software.
- AMIs are the templates for instances, which are running VMs.

# Programming on Amazon AWS:

## Programming on Amazon EC2:

- There are 3 types of AMIs as shown in below table.

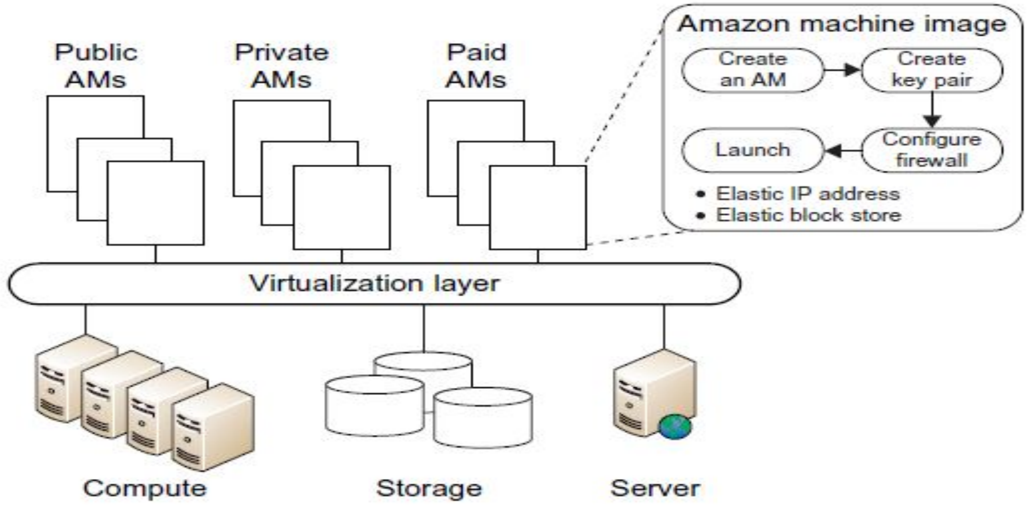


| Image Type | AMI Definition |
|---|---|
| Private AMI | Images created by you, which are private by default. You can grant access to other users to launch your private images. |
| Public AMI | Images created by users and released to the AWS community, so anyone can launch instances based on them and use them any way they like. AWS lists all public images at http://developer.amazonwebservices.com/connect/kbcategory.jspa?categoryID=171. |
| Paid QAMI | You can create images providing specific functions that can be launched by anyone willing to pay you per each hour of usage on top of Amazon's charges. |

Three Types of AMI

## Programming on Amazon AWS:

### Programming on Amazon EC2:



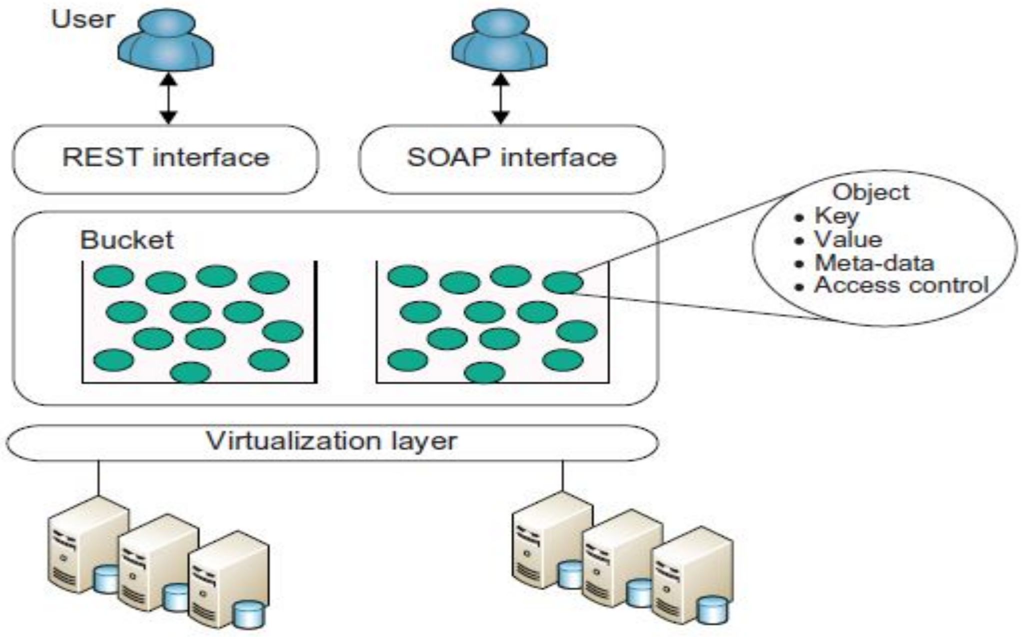
**Amazon EC2 Execution Environment**

- The AMIs are formed from the virtualized compute, storage, and server resources.

# Programming on Amazon AWS:

## Amazon Simple Storage Service (S3):

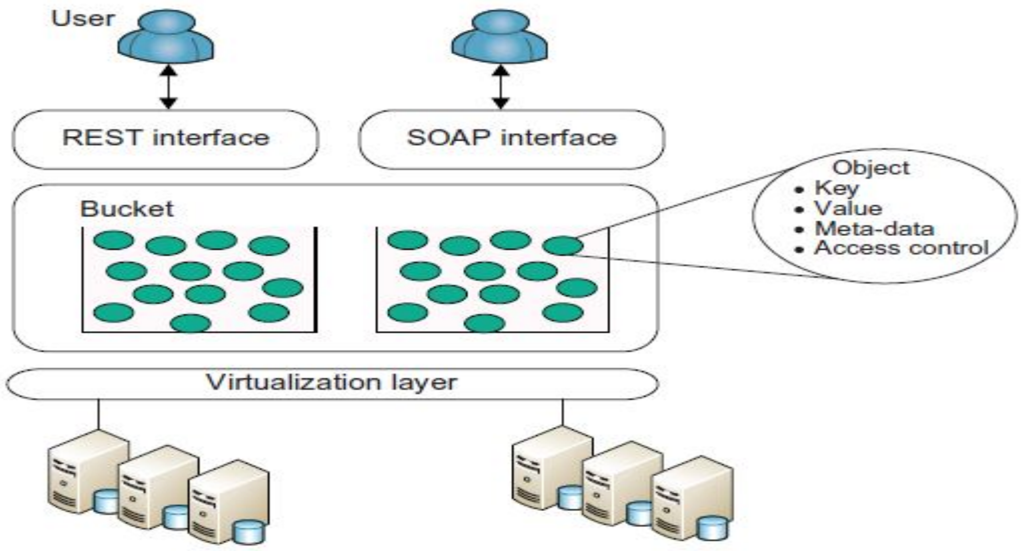
- Amazon S3 provides a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web.
- S3 provides the object-oriented storage service for users.
- Users can access their objects through *Simple Object Access Protocol (SOAP)* with either browsers or other client programs which support SOAP.



Amazon S3 Execution Environment

# Programming on Amazon AWS:

## Amazon Simple Storage Service (S3):

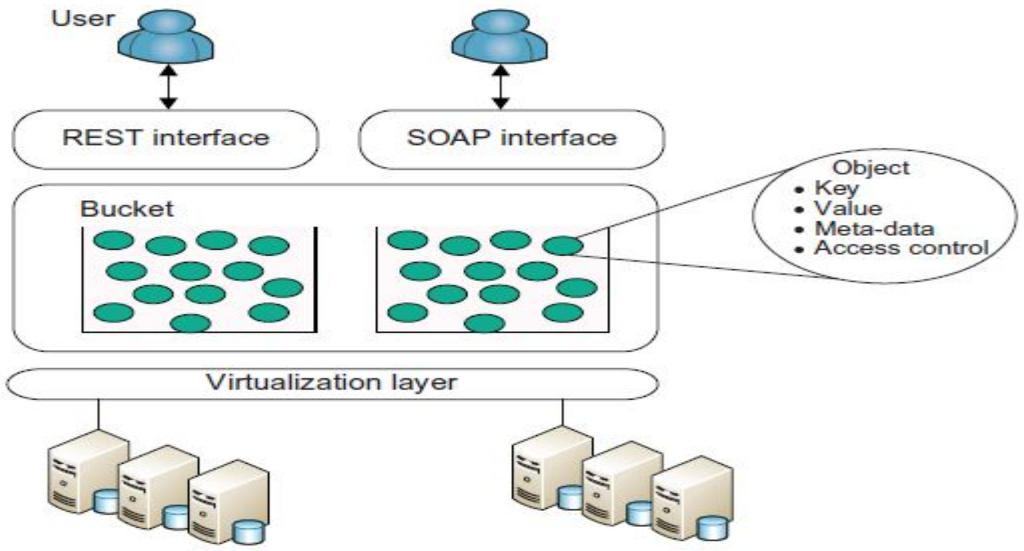


Amazon S3 Execution Environment

- The fundamental operation unit of S3 is called an object.
- Each object is stored in a bucket and retrieved via a unique, developer-assigned key.
- In other words, the bucket is the container of the object.
- Besides unique key attributes, the object has other attributes such as values, metadata, and access control information.
- From the programmer's perspective, the storage provided by S3 can be viewed as a key-value pair.

## Programming on Amazon AWS:

### Amazon Simple Storage Service (S3):



Amazon S3 Execution Environment

- Through the key-value programming interface, users can write, read, and delete objects containing from 1 byte to 5 gigabytes of data each.
- There are two types of web service interface for the user to access the data stored in Amazon clouds. One is a REST (web 2.0) interface, and the other is a SOAP interface.

## Programming on Amazon AWS:

### Amazon Simple Storage Service (S3):

**Key features of S3:**

- Redundant through geographic dispersion.
- Designed to provide 99.99 percent durability and 99.99 percent availability of objects over a given year with cheaper reduced redundancy storage (RRS).
- Authentication mechanisms to ensure that data is kept secure from unauthorized access. Objects can be made private or public, and rights can be granted to specific users.
- Per-object URLs and ACLs (access control lists).
- First 1 GB per month input or output free and then $0.08 to $0.15 per GB for transfers outside an S3 region.
- There is no data transfer charge for data transferred between Amazon EC2 and Amazon S3 within the same region.

## Programming on Amazon AWS:

### Amazon Elastic Block Store (EBS):

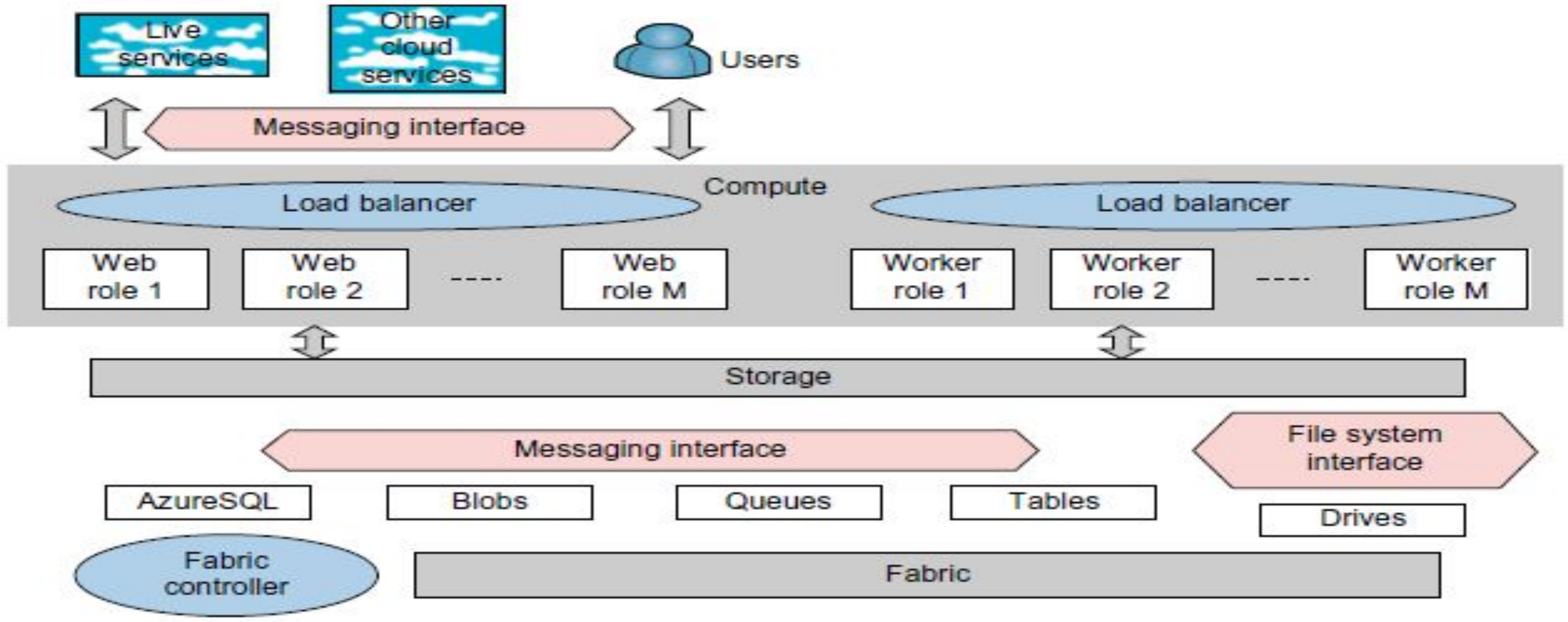
- The Elastic Block Store (EBS) provides the volume block interface for saving and restoring the virtual images of EC2 instances.
- The status of EC2 instances saved in the EBS system.
- Users can use EBS to save persistent data and mount to the running instances of EC2.
- EBS is analogous to a distributed file system accessed by traditional OS disk access mechanisms.
- EBS allows you to create storage volumes from 1 GB to 1 TB that can be mounted as EC2 instances.
- Multiple volumes can be mounted to the same instance.
- You can create a file system on top of Amazon EBS volumes, or use them in any other way you would use a block device (like a hard drive).
- For EBS, Amazon provides a similar pay-per-use schema as EC2 and S3.

## Programming on Amazon AWS:

### Amazon SimpleDB Service:

- SimpleDB provides a simplified data model based on the relational database data model.
- Structured data from users organized into domains, where each domain can be considered a table.
- In SimpleDB, it is possible to assign multiple values to a single cell in the table, which is not permitted in a traditional relational database.

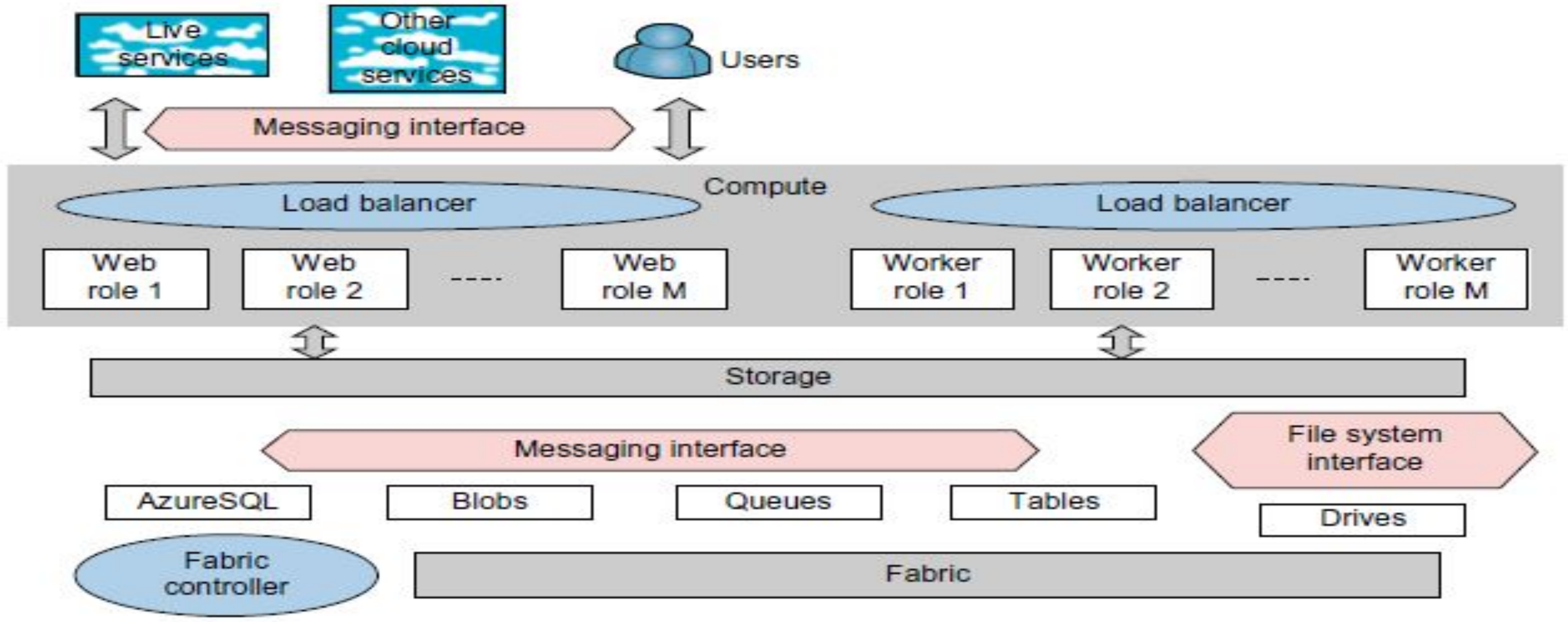
# Microsoft Azure Programming Support:



Features of the Azure cloud platform

- **Azure fabric** consisting of virtualized hardware together with a sophisticated control environment implementing dynamic assignment of resources and fault tolerance.
- **Azure storage** is used to store event logs, debug data, IIS web server logs.
- The Azure application is linked to the Internet through a customized compute VM called a ***web role*** supporting basic Microsoft web hosting.

## Microsoft Azure Programming Support:



**Features of the Azure cloud platform**

- The other important compute class is the ***worker role*** reflecting the importance in cloud computing of a pool of compute resources that are scheduled as needed.
- The AzureSQL service offers SQL Server as a service.

## Computing Platforms and Technologies:

### Manjrasoft Aneka:

- Manjrasoft Aneka is a cloud application platform for rapid creation of scalable applications and their deployment on various types of clouds in a seamless and elastic manner.
- Aneka is Manjrasoft's solution for developing, deploying, and managing cloud applications.
- Aneka consists of a scalable cloud middleware that can be deployed on top of heterogeneous computing resources (clusters, networked desktop computers, and cloud resources).
- It offers an extensible collection of services coordinating the execution of applications, helping administrators monitor the status of the cloud, and providing integration with existing cloud technologies.
- One of Aneka's key advantages is its extensible set of application programming interfaces (APIs) associated with different types of programming models—such as Task, Thread, and MapReduce—used for developing distributed applications, integrating new capabilities into the cloud, and supporting different types of cloud deployment models: public, private, and hybrid.

## Computing Platforms and Technologies:

### Manjrasoft Aneka:

- Aneka acts as a workload distribution and management platform for accelerating applications in both Linux and Microsoft .NET framework environments.

- **Additional advantages of Aneka:**

- Support of multiple programming and application environments.
- Rapid deployment tools and framework
- Ability to harness multiple virtual and/or physical machines for accelerating application provisioning based on users' Quality of Service/service-level agreement (QoS/SLA) requirements.
- Built on top of the Microsoft .NET framework, with support for Linux environments through Mono.

## Computing Platforms and Technologies:

### Manjrasoft Aneka:

- Aneka offers three types of capabilities which are essential for building, accelerating and managing clouds and their applications:

1) **Build:**
   - Aneka includes a new SDK which combines APIs and tools to enable users to rapidly develop applications.
   - Aneka also allows users to build different runtime environments such as enterprise/private cloud by harnessing compute resources in network or enterprise data centers, Amazon EC2.

2) **Accelerate:**
   - Aneka supports rapid development and deployment of applications in multiple runtime environments running different operating systems such as Windows or Linux/UNIX.
   - Aneka uses physical machines as much as possible to achieve maximum utilization in local environments.

## Computing Platforms and Technologies:

### Manjrasoft Aneka:

**2) Accelerate:**

- Whenever users set QoS parameters such as deadlines, and if the enterprise resources are insufficient to meet the deadline, Aneka supports dynamic leasing of extra capabilities from public clouds such as EC2 to complete the task within the deadline.

**3) Manage:**

- Management tools and capabilities supported by Aneka include a GUI and APIs to set up, monitor, manage, and maintain remote and global Aneka compute clouds.

## Computing Platforms and Technologies:

### Manjrasoft Aneka:

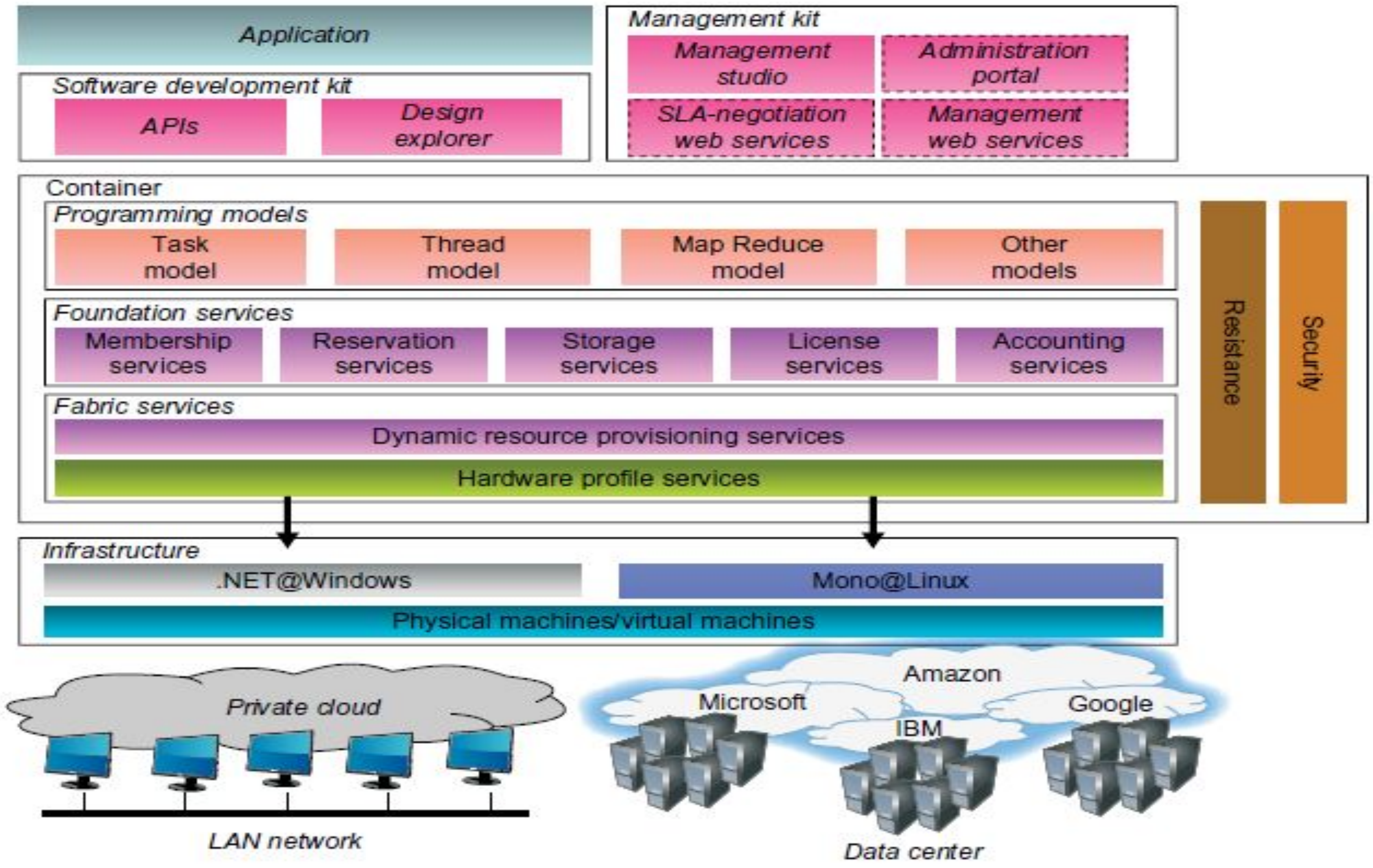
- Currently Aneka supports following three programming models:

1) **Task Programming Model:** In task model, application is modeled as a collection of tasks that are independent from each other.
2) **Thread Programming Model:** This model provides an extension to the classical multi-threaded programming to a distributed infrastructure and uses the abstraction of Thread to wrap a method that is executed remotely.
3) **MapReduce Programming Model:** This is an implementation of MapReduce as proposed by Google on top of Aneka.

# Computing Platforms and Technologies:

## Manjrasoft Aneka:

## Aneka Architecture:



Architecture and components of Aneka

## Computing Platforms and Technologies:

**Manjrasoft Aneka:**

**Aneka Architecture:**

- Aneka as a cloud application development platform features a homogeneous distributed runtime environment for applications.
- This environment is built by aggregating together physical and virtual nodes hosting the Aneka container.
- The container is a lightweight layer that interfaces with the hosting environment and manages the services deployed on a node.

The available services can be aggregated into three major categories:

**1) Fabric Services:**

- Fabric services implement the fundamental operations of the infrastructure of the cloud.
- These services include failover for improved reliability, resource provisioning, performance monitoring.

## Computing Platforms and Technologies:

**Manjrasoft Aneka:**

**Aneka Architecture:**

**2) Foundation Services:**

- Foundation services provide a basic set of capabilities that enhance application development in the cloud.
- These services provide added value to the infrastructure and are of use to system administrators and developers.
- Within this category we can list membership services, storage management services, resource reservation services, accounting services and licensing services.

**3) Application Services:**

- Application services deal directly with the execution of applications and are in charge of providing the appropriate runtime environment for each application model.
- Application services leverage foundation and fabric services for several tasks of application execution such as elastic scalability, data transfer, and performance monitoring.

## Emerging Cloud Software Environments:

### Open Source Eucalyptus:

- Eucalyptus is a product from Eucalyptus Systems that was developed out of a research project at the University of California.
- Eucalyptus is an acronym for **Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems**.
- Eucalyptus is an open source software platform for implementing Infrastructure-as-a-Service (IaaS) in a private or hybrid cloud computing environment.
- The Eucalyptus cloud platform pools together existing virtualized infrastructure to create cloud resources for infrastructure as a service, network as a service and storage as a service.
- Eucalyptus Systems announced a formal agreement with Amazon Web Services (AWS) in March 2012, allowing administrators to move instances between a Eucalyptus private cloud and the Amazon Elastic Compute Cloud (EC2) to create a hybrid cloud.

## Emerging Cloud Software Environments:

### Open Source Eucalyptus:

- Eucalyptus was initially aimed at bringing the cloud computing facility to academic supercomputers and clusters.
- Eucalyptus provides an AWS-compliant EC2-based web service interface for interacting with the cloud service.
- Additionally, Eucalyptus provides services, such as the AWS-compliant ***Walrus***, and a user interface for managing users and images.
- Eucalyptus enables pooling compute, storage, and network resources that can be dynamically scaled up or down as application workloads change.

## Emerging Cloud Software Environments:

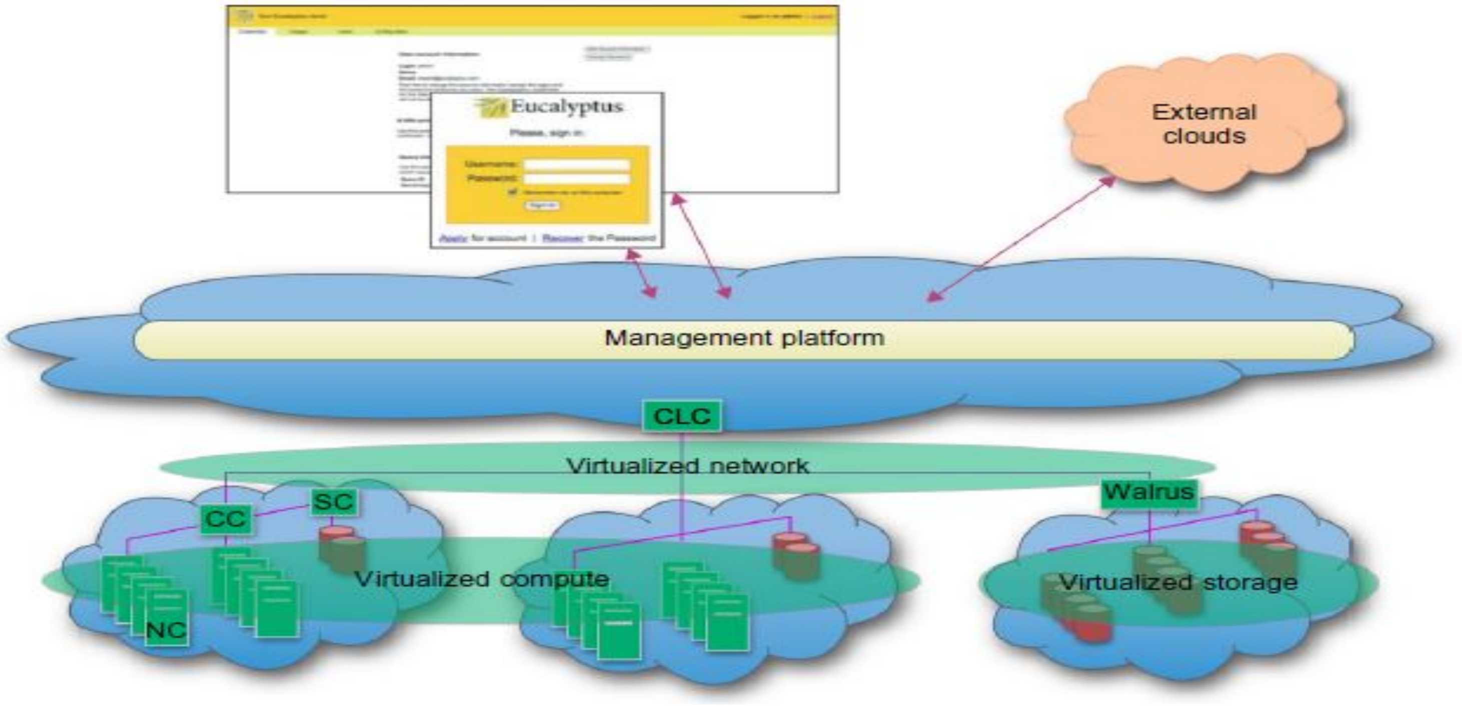
### Open Source Eucalyptus:

### Eucalyptus features include:

- Supports both Linux and Windows virtual machines (VMs).
- Application program interface- (API) compatible with Amazon EC2 platform.
- Compatible with Amazon Web Services (AWS) and Simple Storage Service (S3).
- Works with multiple hypervisors including VMware, Xen and KVM.
- Can be installed and deployed from source code or RPM packages.
- Internal processes communications are secured through SOAP and WS-Security.
- Multiple clusters can be virtualized as a single cloud.
- Administrative features such as user and group management and reports.

## Emerging Cloud Software Environments:

### Open Source Eucalyptus:

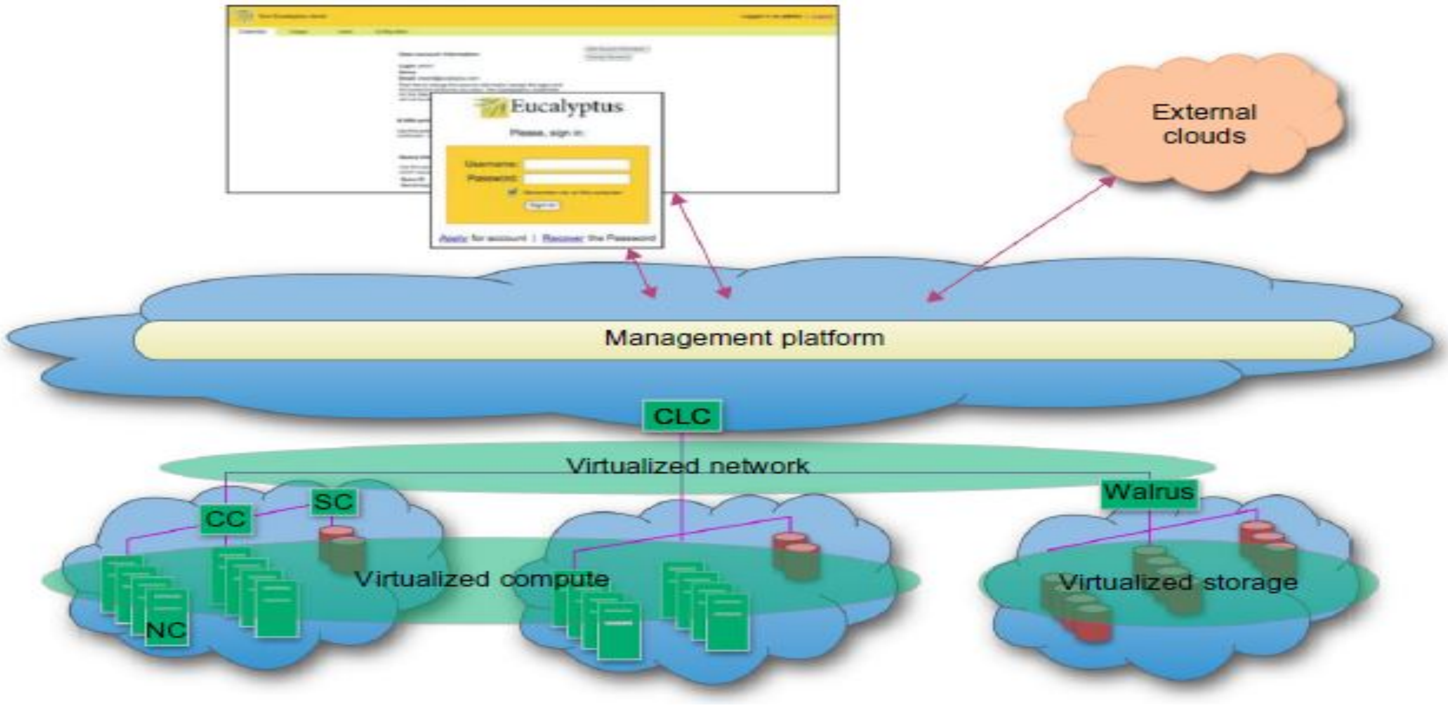


The Eucalyptus architecture for VM image management

- Essentially, the system has been extended to support the development of both the compute cloud and storage cloud.
- Eucalyptus takes many design queues from Amazon's EC2, and its image management system is no different.
- Eucalyptus stores images in Walrus, the block storage system that is analogous to the Amazon S3 service.

# Emerging Cloud Software Environments:

## Open Source Eucalyptus:



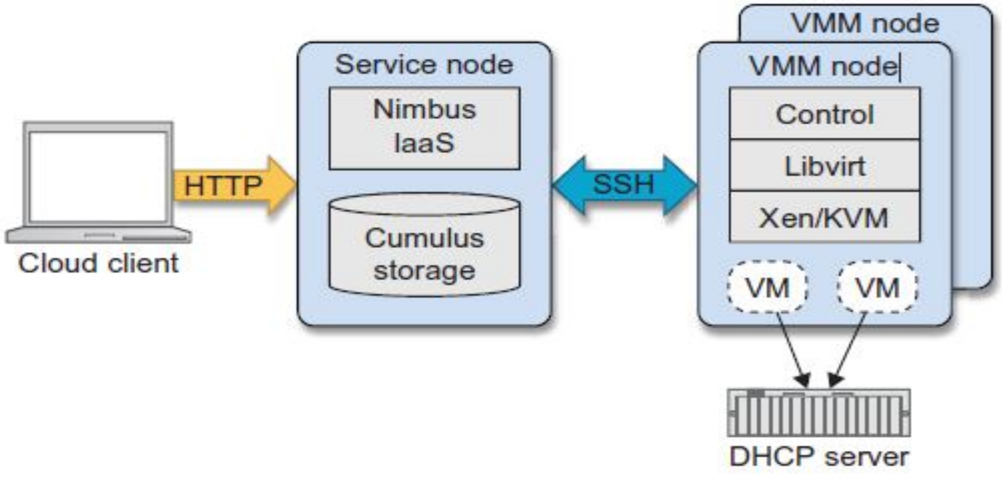
**The Eucalyptus architecture for VM image management**

- The VM image is uploaded into a user-defined bucket within Walrus, and can be retrieved anytime from any availability zone.

## Emerging Cloud Software Environments:

### Nimbus:

- Nimbus is a set of open source tools that together provide an IaaS cloud computing solution.
- Nimbus allows a client to lease remote resources by deploying VMs on those resources and configuring them to represent the environment desired by the user.

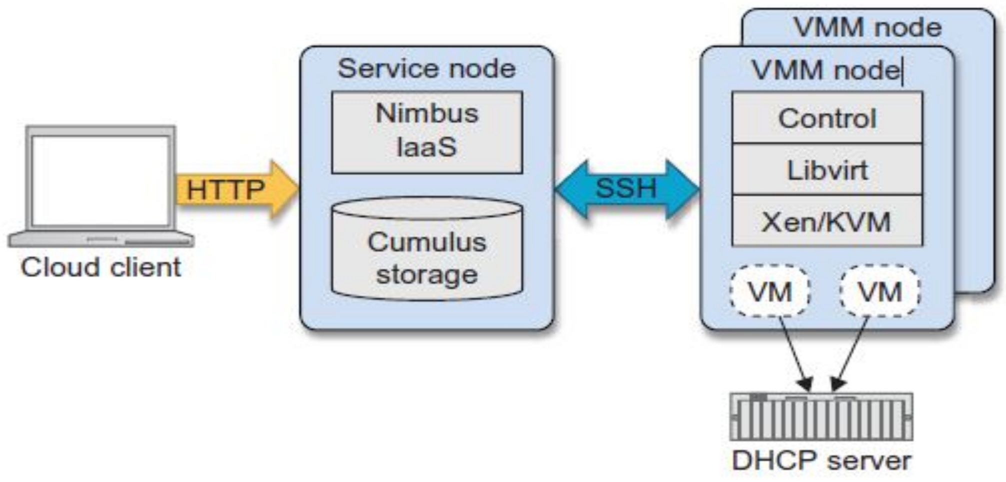


Nimbus cloud infrastructure

- A storage cloud implementation called ***Cumulus*** has been tightly integrated with the other central services, although it can also be used stand-alone.
- Cumulus is compatible with the Amazon S3.

# Emerging Cloud Software Environments:

## Nimbus:



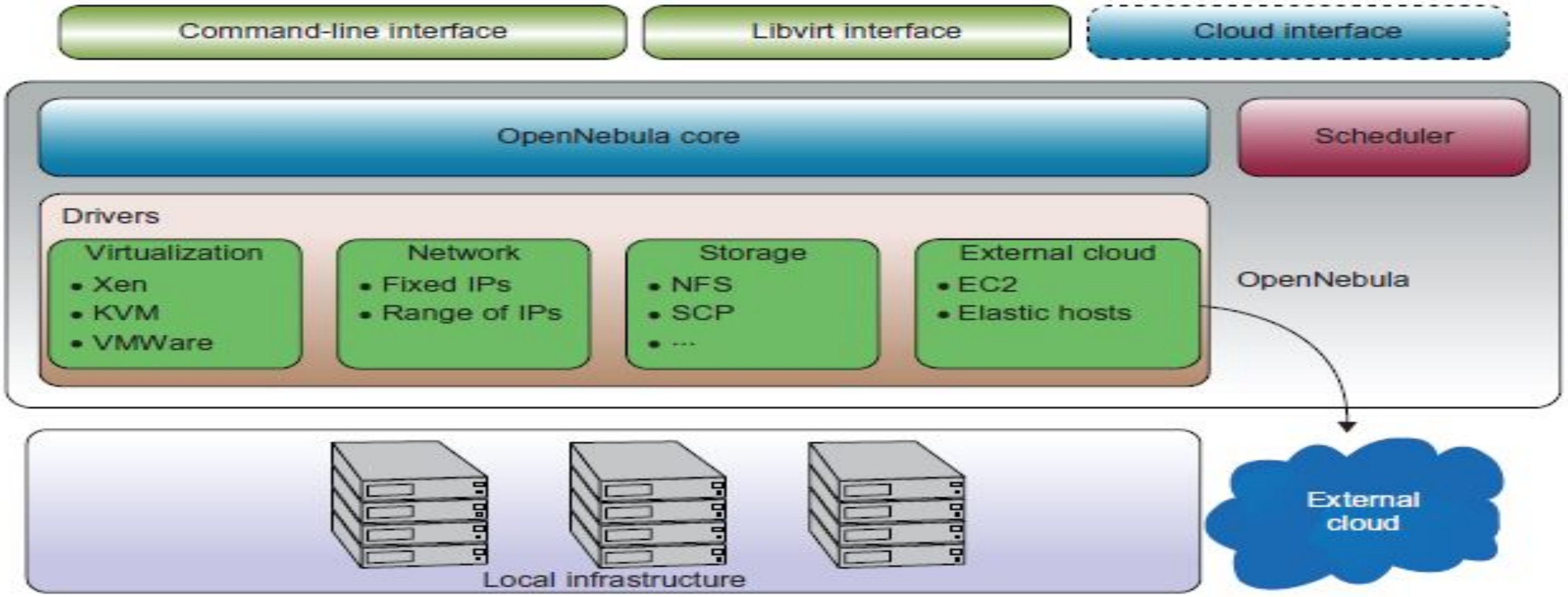
Nimbus cloud infrastructure

- Nimbus supports two resource management strategies.
- The first is the default **"resource pool"** mode. In this mode, the service has direct control of a pool of VM manager nodes and it assumes it can start VMs.
- The other supported mode is called **"pilot."** Here, the service makes requests to a cluster's Local Resource Management System (LRMS) to get a VM manager available to deploy VMs.

# Emerging Cloud Software Environments:

## OpenNebula:

- OpenNebula is an open source toolkit which allows users to transform existing infrastructure into an IaaS cloud with cloud-like interfaces.

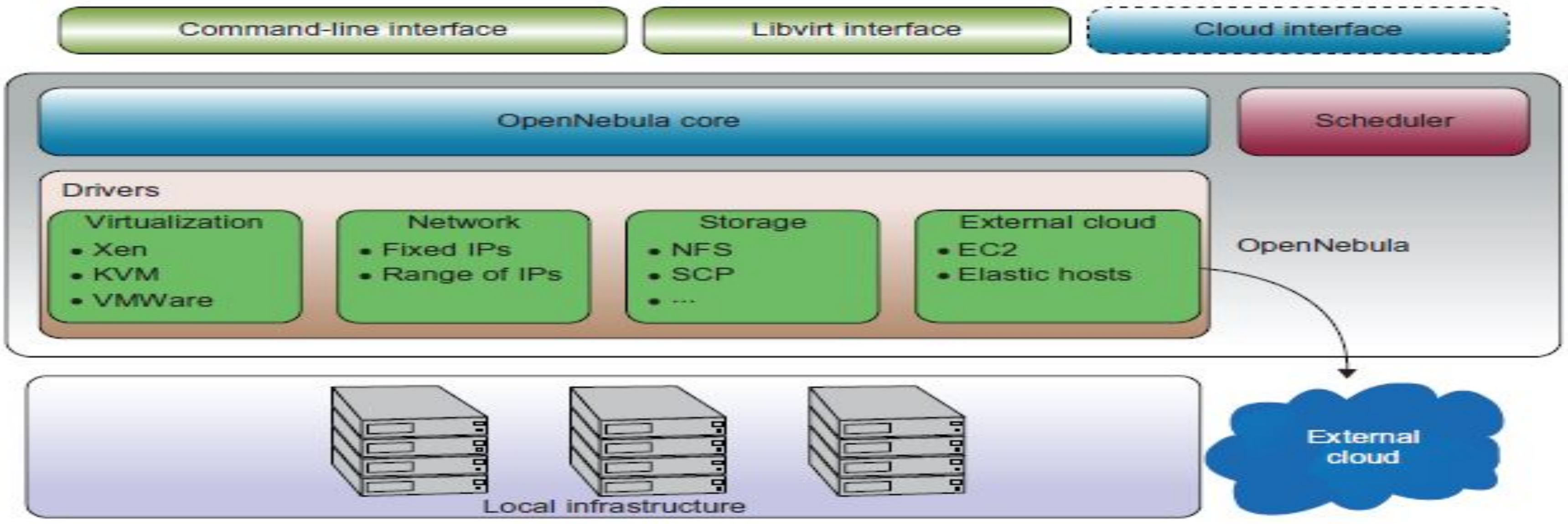


OpenNebula Architecture and Its Main Components

- The architecture of OpenNebula has been designed to be flexible and modular to allow integration with different storage and network infrastructure configurations, and hypervisor technologies.

# Emerging Cloud Software Environments:

## OpenNebula:

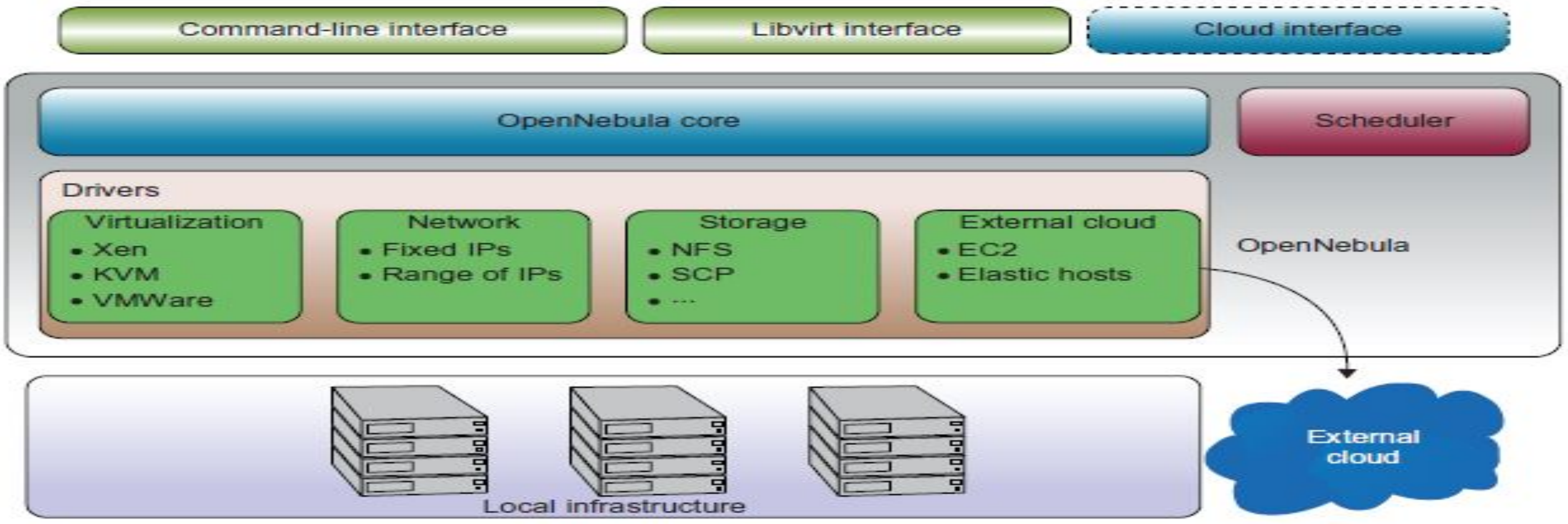


OpenNebula Architecture and Its Main Components

- The ***OpenNebula Core*** is a centralized component that manages the VM full life cycle, including setting up networks dynamically for groups of VMs and managing their storage requirements.
- ***Scheduler*** governs the functionality provided by the core.
- ***Access Drivers*** provide an abstraction of the underlying infrastructure to expose the basic functionality of the monitoring, storage, and virtualization services available in the cluster.

## Emerging Cloud Software Environments:

### OpenNebula:

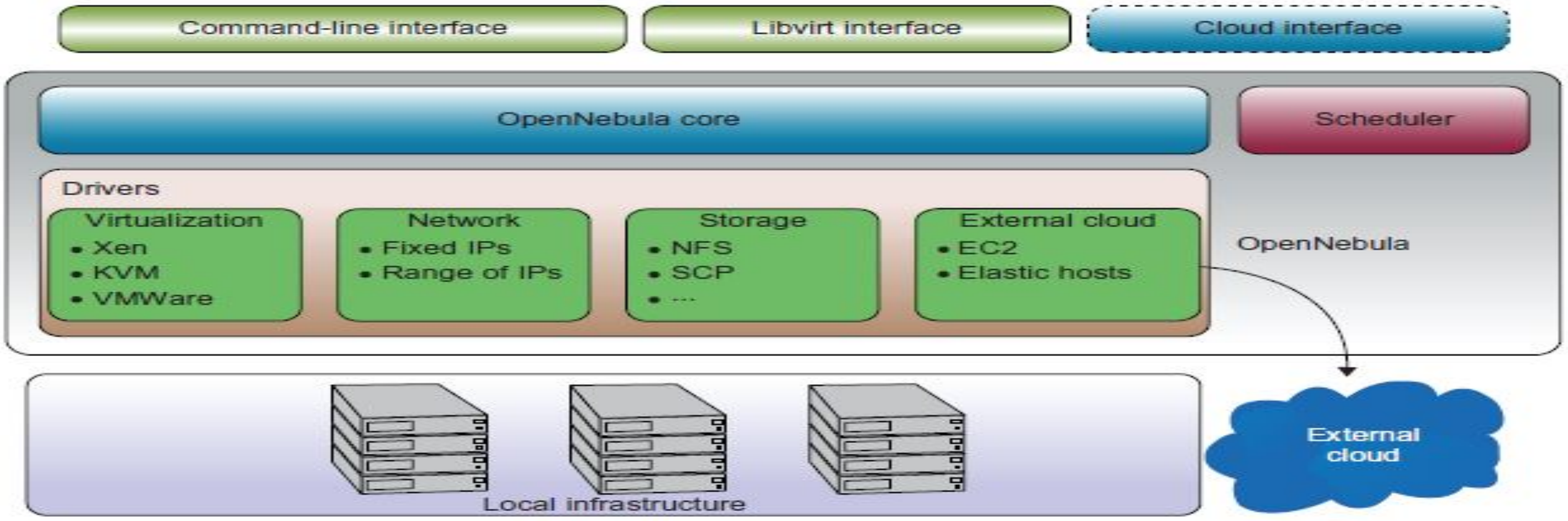


OpenNebula Architecture and Its Main Components

- OpenNebula offers management ***interfaces*** to integrate the core's functionality within other data-center management tools, such as accounting or monitoring frameworks.
- Whenever local resources are insufficient, OpenNebula can support a hybrid cloud model by using cloud drivers to interface with external clouds.
- This lets organizations supplement their local infrastructure with computing capacity from a public cloud to meet peak demands.

# Emerging Cloud Software Environments:

## OpenNebula:



OpenNebula Architecture and Its Main Components

- OpenNebula currently includes an EC2 driver, which can submit requests to Amazon EC2.

Thank you
for your attention