UCS18E08 – CLOUD COMPUTING

UNIT 1 - NOTES

Computing on Cloud

Cloud computing is the next stage in evolution of the Internet. The cloud in cloud computing provides the means through which everything — from computing power to computing infrastructure, applications, business processes to personal collaboration — can be delivered to you as a service wherever and whenever you need.

Cloud computing is offered in different forms:

- Public clouds
- Private clouds
- Hybrid clouds, which combine both public and private

In general the cloud — similar to its namesake of the cumulus type — is fluid and can easily expand and contract. This elasticity means that users can request additional resources on demand and just as easily release those resources when they're no longer needed. This elasticity is one of the main reasons individual, business, and IT users are moving to the cloud.

In the traditional data center it has always been possible to add and release resources. However, this process couldn't be done in an automated or self-service manner.

This evolution to cloud computing — already underway — can completely change the way companies use technology to service customers, partners, and suppliers. Some businesses already have IT resources almost entirely in the cloud. They feel that the cloud model provides a more efficient, cost effective IT service delivery.

Defining Cloud

The cloud itself is a set of hardware, networks, storage, services, and interfaces that enable the delivery of computing as a service. Cloud services include the delivery of software, infrastructure, and storage over the Internet (either as separate components or a complete platform) based on user demand.

The world of the cloud has lots of participants:

1. The end user doesn't really have to know anything about the underlying technology. In small businesses, for example, the cloud provider becomes the de facto data center. In larger organizations, the IT organization oversees the inner workings of both internal resources and external cloud resources.
2. Business management needs to take responsibility for overall governance of data or services living in a cloud. Cloud service providers must provide a predictable and guaranteed service level and security to all their constituents.
3. The cloud service provider is responsible for IT assets and maintenance.

Characteristics of Cloud Computing

Cloud services like social networks (such as Facebook or LinkedIn) and collaboration tools (like video conferencing, document management, and webinars) are changing the way people in businesses access, deliver, and understand information. Cloud computing infrastructures make it easier for companies to treat their computing systems as a pool of resources rather than a set of independent environments that each has to be managed.

Overall, the cloud embodies the following four basic characteristics:

1. Elasticity and the ability to scale up and down
2. Self-service provisioning and automatic deprovisioning
3. Application programming interfaces (APIs)
4. Billing and metering of service usage in a pay-as-you-go model

1. Elasticity and scalability

The service provider can't anticipate how customers will use the service. One customer might use the service three times a year during peak selling seasons, whereas another might use it as a primary development platform for all of its applications.

Therefore, the service needs to be available all the time (7 days a week, 24 hours a day) and it has to be designed to scale upward for high periods of demand and downward for lighter ones. Scalability also means that an application can scale when additional users are added and when the application requirements change.

This ability to scale is achieved by providing elasticity. Think about the rubber band and its properties. If you're holding together a dozen pens with a rubber band, you probably have to fold

it in half. However, if you're trying to keep 100 pens together, you will have to stretch that rubber band. Why can a single rubber band accomplish both tasks? Simply, it is elastic and so is the cloud.

2. Self-service provisioning

Customers can easily get cloud services without going through a lengthy process. The customer simply requests an amount of computing, storage, software, process, or other resources from the service provider.

When a department is about to implement a new application, it has to submit a request to the data center for additional computing hardware, software, services, or process resources. The data center gets similar requests from departments across the company and must sort through all requests and evaluate the availability of existing resources versus the need to purchase new hardware. After new hardware is purchased, the data center staff has to configure the data center for the new application. These internal procurement processes can take a long time, depending on company policies.

While the on-demand provisioning capabilities of cloud services eliminate many time delays, an organization still needs to do its homework. These services aren't free; needs and requirements must be determined before capability is automatically provisioned.

3. Application programming interfaces (APIs)

Cloud services need to have standardized APIs. These interfaces provide the instructions on how two application or data sources can communicate with each other.

A standardized interface lets the customer more easily link a cloud service, such as a customer relationship management system with a financial accounts management system, without having to resort to custom programming.

4. Billing and metering of services

A cloud environment needs a built-in service that bills customers. And, of course, to calculate that bill, usage has to be metered (tracked). Even free cloud services (such as Google's Gmail or Zoho's Internet-based office applications) are metered.

In addition to these characteristics, cloud computing must have two overarching requirements to be effective:

- A comprehensive approach to service management
- A well-defined process for security management

Performance monitoring and measuring

A cloud service provider must include a service management environment. A service management environment is an integrated approach for managing your physical environments and IT systems. This environment must be able to maintain the required service level for that organization.

In other words, service management has to monitor and optimize the service or sets of services. Service management has to consider key issues, such as performance of the overall system, including security and performance. For example, an organization using an internal or external email cloud service would require 99.999 percent uptime with maximum security. The organization would expect the cloud provider to prove that it has met its obligations.

Many cloud service providers give customers a dashboard — a visualization of key service metrics — so they can monitor the level of service they're getting from their provider. Also, many customers use their own monitoring tools to determine whether their service level requirements are being met.

Security

Many customers must take a leap of faith to trust that the cloud service is safe. Turning over critical data or application infrastructure to a cloud-based service provider requires making sure that the information can't be accidentally accessed by another company (or maliciously accessed by a hacker).

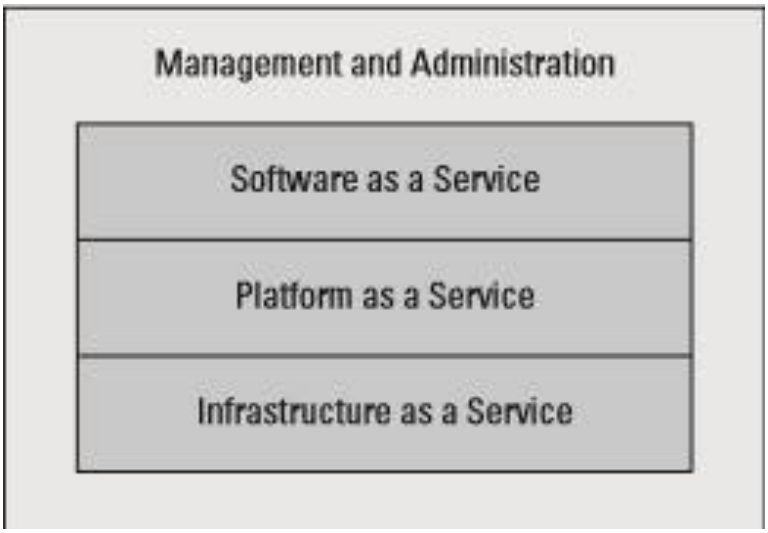
Many companies have compliance requirements for securing both internal and external information. Without the right level of security, you might not be able to use a provider's offerings

Cloud Services

The reality of cloud computing is that there is a blending between the types of service delivery models that are available from cloud vendors. For example, a Software as a Service vendor might decide to offer separate infrastructure services to customers. The purpose of grouping these services into three models is to aid in understanding what lies beneath a cloud service. All these service delivery models require management and administration (including security)

The three cloud service delivery models are Infrastructure as a Service, Platform as a Service, and Software as a Service, and the purpose of each model is as follows:

- The **Infrastructure as a Service layer** offers storage and compute resources that developers and IT organizations use to deliver custom business solutions.
- The **Platform as a Service layer** offers development environments that IT organizations can use to create cloud-ready business applications.
- The **Software as a Service layer** offers purpose-built business applications



1. Infrastructure as a Service

Infrastructure as a Service (IaaS) is the delivery of computer hardware (servers, networking technology, storage, and data center space) as a service. It may also include the delivery of operating systems and virtualization technology to manage the resources.

The IaaS customer rents computing resources instead of buying and installing them in their own data center. The service is typically paid for on a usage basis. The service may include dynamic scaling so that if the customer winds up needing more resources than expected, he can get them immediately (probably up to a given limit).

Dynamic scaling as applied to infrastructure means that the infrastructure can be automatically scaled up or down, based on the requirements of the application.

Additionally, the arrangement involves an agreed-upon service level. The service level states what the provider has agreed to deliver in terms of availability and response to demand. It might, for example, specify that the resources will be available 99.999 percent of the time and that more resources will be provided dynamically if greater than 80 percent of any given resource is being used.

Currently, the most high-profile IaaS operation is Amazon's Elastic Compute Cloud (Amazon EC2). It provides a Web interface that allows customers to access virtual machines. EC2 offers scalability under the user's control with the user paying for resources by the hour

2. **Platform as a Service**

With Platform as a Service (PaaS), the provider delivers more than infrastructure. It delivers what you might call a solution stack — an integrated set of software that provides everything a developer needs to build an application — for both software development and runtime.

PaaS can be viewed as an evolution of Web hosting. In recent years, Webhosting companies have provided fairly complete software stacks for developing Web sites. PaaS takes this idea a step farther by providing lifecycle management — capabilities to manage all software development stages from planning and design, to building and deployment, to testing and maintenance. The primary benefit of PaaS is having software development and deployment capability based entirely in the cloud — hence, no management or maintenance efforts are required for the infrastructure. Every aspect of software development, from the design stage onward (including source-code management, testing, and deployment) lives in the cloud.

PaaS is inherently multi-tenant and naturally supports the whole set of Web services standards and is usually delivered with dynamic scaling. In reference to Platform as a Service, dynamic scaling means that the software can be automatically scaled up or down. Platform as a Service

typically addresses the need to scale as well as the need to separate concerns of access and data security for its customers.

Some examples of Platform as a Service include the Google App Engine, AppJet, Etelos, Qrimp, and Force.com, which is the official development environment for Salesforce.com

3. Software as a Service

One of the first implementations of cloud services was Software as a Service (SaaS) — business applications that are hosted by the provider and delivered as a service.

SaaS has its roots in an early kind of hosting operation carried out by Application Service Providers (ASPs). The ASP business grew up soon after the Internet began to mushroom, with some companies offering to securely, privately host applications. Hosting of supply chain applications and customer relationship management (CRM) applications was particularly prominent, although some ASPs simply specialized in running email. Prior to the advent of this type of service, companies often spent huge amounts of money implementing and customizing these applications to satisfy internal business requirements. Many of these products weren't only difficult to implement but hard to learn and use. However, the most successful vendors were those who recognized that an application delivered as a service with a monthly fee based on the number of users had to be easy to use and easy to stay with.

CRM is one of the most common categories of Software as a Service; the most prominent vendor in this category is Salesforce.com, described in this chapter's sidebar. For a more extensive look at some of the other examples of Software as a Service.

Administering & Monitoring cloud services

Cloud services impact your organization in subtle ways. The cloud impacts the whole company, not just the IT department:

- How do cloud services fit into your overall corporate and IT strategy?
- How will you manage cloud service providers along with your internal services?
- How will you make sure that your customers are well supported by services that are moving to a cloud?
- Does the cloud support your corporate and IT governance requirements?
- What are the important issues of emerging corporate and governmental standards, business process management, and the overall issues of managing costs?

1. Deciding on a strategy

Like any other technology strategy, a cloud strategy is considered in relationship to the following:

- Your IT organization's overall strategy
- Your company's overall strategy

You must make a complex evaluation of costs, benefits, business cultural issues, risks, and corporate and government standards before developing a comprehensive cloud strategy. Although very few organizations have tested cloud services in these heavy usage situations, a well-planned cloud service strategy has the potential to significantly reduce costs.

Over time, however, as more well-tested commercial cloud services become available, companies will increasingly be able to rely on these services not just for IT cost savings, but also for delivering new value to the organization. The trend toward well-managed cloud services is especially important because of the increased automation across the organization. This may include the software embedded in everything from manufacturing systems to radio frequency identification tags that track inventory.

2. **Coping with governance issues**

Each approach presents different governance challenges:

- Infrastructure as a Service
- Platform as a Service
- Software as a Service
- Business Process as a Service

Governing internally provided services and the externally provided cloudbased services introduces new challenges for a company's strategy:

1. How do you manage the overall lifecycle of your IT resources, including software licensing, cost allocation, and charge backs?
2. How to you protect the integrity of your information resources?
3. How do you ensure that you're complying with data privacy rules and regulations?
4. How do you make sure that all your service providers can prove and document that they're meeting governmental and corporate requirements?

3. **Monitoring business processes**

Most cloud services impact the way business processes are implemented within an organization. For example, your organization may be using a cloud based service to check credit worthiness for potential customers. Therefore, you have to make sure that these services are linked back to your internal systems so things don't fall through the cracks.

Your business should standardize a way to monitor business processes that live entirely or partially in a cloud environment. An organization's important computer-dependent business processes need to be constantly monitored by software. Linking internal and external processes together in a seamless way is the best way to ensure customer satisfaction.

4. **Managing IT costs**

All IT departments monitor costs, but few monitor them in terms of asset performance — the requirement to optimize the return on investments for both hardware and software. This is likely to change with the onset of cloud services. Unlike traditional licensing models, cloud propositions are based on rental arrangements.

You must compare two cost models:

- Operating expenses (paying per month, per user for each service)
- Capital investments (paying a purchase fee plus yearly maintenance for software that resides within your organization)

5. **Service level agreements and monitoring**

Every company that buys any service from a cloud service provider must either accept a standard service level agreement (SLA) from the provider or negotiate such an agreement. A service level agreement is a contract that stipulates the type of service you need from providers and what type of penalties would result from an unexpected business interruption.

No organization should commit mission-critical systems to the cloud without negotiating an SLA that includes significant penalties for not delivering the promised service level. Management needs to know what service level is appropriate under changing business conditions. Management can't assume that the service provider will provide all the monitoring. Rather, the administrators must have their own ability to monitor service to satisfy the company's goals for performance.

6. **Support**

Support problems don't disappear when applications or infrastructures move to the cloud. You have to make sure that support targets are agreed on in advance with a cloud services provider. Therefore, your company must align its internal support team that deals with internal customers with the cloud provider.

What processes are in place to resolve problems when they arise? Just consider the situation where some important application has a performance problem. Especially in a hybrid

environment, it's not always easy to tell if a problem resides within the cloud or outside of it. Such situations need to be prevented or at least dealt with very efficiently.

7. **Billing and accounting**

One cloud benefit is that, as a customer you can acquire just as much capability as needed. For this to work, billing and account management must be automated. Customers, therefore, need to be able to monitor what they're using and how much it costs.

Potential problems arise if service level penalties aren't clear and if the provider adds too many incidental charges. Customers can run up unexpected bills if they can't accurately track usage.

BENEFITS AND LIMITATIONS OF CLOUD COMPUTING

BENEFITS

Your organization is going to have different needs from the company next door. However, cloud computing can help you with your IT needs. Let's take a closer look at what cloud computing has to offer your organization.

1. **Scalability**

If you are expecting a huge demand in computing need or a sudden demand, cloud computing can help you manage. *Without having to buy, install, and configure new equipment, you can buy additional CPU cycles or storage from a third party*.

Since your costs are based on consumption, you likely wouldn't have to pay out as much as if you had to buy the equipment.

Once you have fulfilled your need for additional equipment, you just stop using the cloud provider's services, and you don't have to deal with unneeded equipment. You simply add or subtract based on your organization's need.

2. **Simplicity**

Again, not having to buy and configure new equipment allows you and your staff to get in to your business. The cloud solution makes it possible to get your application started immediately,

and it costs a fraction of what it would cost to implement an on-site solution.

3. **Knowledgeable Vendors**

When new technology becomes popular, there are plenty of vendors who pop up to offer their version of that technology. Companies like **Amazon, Google, Microsoft, IBM, and Yahoo**! have been good vendors because they have offered reliable service, plenty of capacity, and you get some brand familiarity with these well-known names.

4. **More Internal Resources**

By shifting your critical data needs to a third party, your IT department is freed up to work on important, business-related tasks. You also don't have to add more manpower and training that stem from having to deal with these low-level tasks.

When you're looking at service providers, make sure you find someone who offers 24-hour help and support and can respond to emergency situations.

5. **Security**

There are plenty of security risks when using a cloud vendor, but reputable companies make every effort to keep you safe and secure. Vendors have strict privacy policies and employ stringent security measures, like proven cryptographic methods to authenticate users.

Further, you can always encrypt your data before storing it on a provider's cloud. In some cases, between your encryption and the vendor's security measures, your data may be more secure than if it were stored in-house.

LIMITATIONS

1. **Your Sensitive Information**

We've talked about the concern of storing sensitive information on the cloud, but it can't be simple. Once data leaves your hands and lands in the lap of a service provider, you've lost a layer of control

What's the Worry?

Let's say a financial planner is using Google Spreadsheets to maintain a list of employee social security numbers. Now the financial planning company isn't the only one who should protect the data from hackers and internal data breaches. In a technical sense, it also becomes Google's problem. But, Google may forgive itself of responsibility in its agreement with you.

Also, less scrupulous service providers might even share that data with a marketing firm.

What's important is that you realize what the provider's policies are governing the management and maintenance of your data.

And in the media we regularly hear about retailers and others losing credit card numbers. In 2007, the British government even misplaced 25 million taxpayer records. ***The point is, if you have sensitive or proprietary data, the cloud might not be the safest place for it.***

Protect Your Data

If you want to maintain your data on a cloud; you just need to be safe. The best way is to encrypt your data before you send it to a third party. Programs like PGP (www.pgp.com) or open-source TrueCrypt can encrypt the file so that only those with a password can access it. Encrypting your data before sending it out protects it. If someone does get your data, they need the proper identification or all they get is rubbish.

2. Application Not Ready

In some cases the applications themselves are not ready to be used on the cloud. They may have little quirks that prevent them from being used to their fullest abilities, or they may not work whatsoever.

First, the application might require a lot of bandwidth to communicate with users. Since cloud computing is paid based on how much you use, it might turn out to be less expensive in the long run to simply house the application locally until it can be rewritten or otherwise modified to operate more efficiently.

The application might also take a lot of effort to integrate with your other applications. If you try to relocate it to a cloud, you may find that the savings are erased by the additional effort required to

maintain the integration. In this case it may end up being more cost effective to continue to host it locally.

If the application has to talk with a database that you have onsite, it may be better to also have the application hosted locally until you can move the entire infrastructure to the cloud. Again, this helps you avoid the service cost of having to transfer to and from the cloud. It's also more efficient, because the application can talk to the database without having to reach out across the network to do so.

3. Developing Your Own Applications

Often, the applications you want are already out there. However, it may be the case that you need a very specific application. And in that case, you'll have to commission its development yourself.

Rolling Up Your Sleeves

Developing your own applications can certainly be a problem if you don't know how to program, or if you don't have programmers on staff. In such a case, you'll have to hire a software company (or developer) or be left to use whatever applications the provider offers.

And it isn't just applications that you might need some programming confidence to deploy. If you have a database on the cloud, you'll need some sort of customized interface and some knowledge of Structured Query Language (SQL) to access and manage that data.

Deploying Application over Cloud

As a developer, you probably hear a lot about new technologies that promise to increase the speed at which you can develop software, as well as ones that can increase the resiliency of your applications once you have deployed them. Your challenge is to wade through these emerging technologies and determine which ones actually hold promise for the projects that you are currently working on.

No doubt, you are aware that cloud computing offers great promise for developers. However, you might not know about the areas where this technology can provide value to you and your projects. You also might not know good practices to employ when implementing a project in the

cloud. This article explores the types of cloud computing systems available, and provides guidelines that can help you with real-world application deployments on top of a cloud infrastructure.

Requirements for deploying an application over cloud

1. **Licensing**

Application is made up of many components which are associated with some license agreements Analysis should be made about the effects of those license agreements on the deployment of application on cloud. Applications which are designed licensed for CPU, when we deploy it on the cloud increases the load by exciding the CPU license limit.

2. **Processing requirements**

Application should be designed to work on the parallel architectures, because of the dynamic scalability of cloud. Multi-threaded code which allows process to split in to small chunks suits for the cloud environment. A single threaded application cannot take the real advantage of clouds distributed nature

3. **Bandwidth requirements**

Because a public cloud is accessed via the Internet, bandwidth is significantly limited when compared to a private cloud. Given the public cloud's bandwidth limitation, applications that have moderate client bandwidth requirements should only be considered.

4. **Communication protocol**

The cloud is based on the Internet Protocol (IP), so for an application to be considered, it must use IP as its communication mechanism. While there are many protocols that can be run over IP, the use of Transport Control Protocol (TCP) is preferred.

5. **Data security**

The application will need to provide security at the data storage, processing and transmission stages. Three critical components of this are

- Data in transit needs to be protected either at the application or the transmission level.
- Data at rest must be protected by the application. The application must provide a mechanism to protect the data stored in the cloud. Encrypting data at rest is the best option at this time, and a future technical tip will delve into the specifics of this area
- Servers to server communications are typically forgotten because they currently exist within the data center

The following steps comprise the deployment of the application

- A load balancer, Web server, and database server appliances should be selected from a library of preconfigured virtual machine images.
- Configuring each component to make a custom image should be made. Load balancer is configured accordingly; web server should be populated with the static contents by uploading them to the storage cloud whereas the database servers are populated with the dynamic content of the site.
- The developer then feeds the custom code in to the new architecture making components meet their specific requirements.
- The developer chooses a pattern that takes the images for each layer and deploys them, handling networking, security, and scalability issues.

The secure, high-availability Web application is up and running. When the application needs to be updated, the virtual machine images can be updated, copied across the development chain, and the entire infrastructure can be redeployed. In this example, a standard set of components can be used to quickly deploy an application. With this model, enterprise business needs can be met quickly, without the need for the time-consuming, manual purchase, installation, cabling, and configuration of servers, storage, and network infrastructure.

Comparison of Saas, PaaS and IaaS

Features	SaaS	PaaS	IaaS
What is offered	Users get the infrastructure such as virtual machines, load balancers, IP addresses and firewalls for them to create a platform, which it can use to test applications.	Users get a work environment on-demand. A platform made of software, hardware, and operating systems. It is a platform where new codes can be added for the development of the end product on a use	The user gets a ready to use package. The user just needs to install it on their systems and start using it. The pre-configured package is as per user requirement, and the user may or may not have to pay to use the

		& pay basis.	services provided.
Importance	Basic layer of cloud computing useful for administrators.	The middle layer of cloud computing that enables development of applications.	The final product, ready to use package.
Technicalities Involved	Deep technical knowledge required. IaaS is the basic layer and if not built strongly, it will not be able to support the further development of the service.	Medium technical know-how necessary for further development of the service takes place in this layer. Proper knowledge of coding and application development is essential to eliminate any possible bugs.	No technical knowledge required. It is the end product. The end-user just needs to use the product that has been created. The SaaS provider handles all the technical aspects of the product.
Deals with	Servers, Load Balancers, Network arrays, virtual machines, storage disks.	Java Runtimes, databases like Oracle and Web Servers.	Applications like Gmail, Yahoo mail. Dropbox and Google Drive services.
Popularity Graph	Used mostly by highly experienced and skilled developers. Custom configuration according to their field of research.	Medium-skilled developers use the platform and the favorable work environment to develop their own applications. Developers don't need to worry about traffic	Most popular amongst users of emails and entertainment stream services. Users don't need to worry about technicalities. Users simply enjoy the end product or service.

		loads or server management.	
Examples	Cisco Metapod, Amazon Web Services, Microsoft Azure	Apprenda, Google App engine, Heroku	Google Apps, Cisco WebEx, Workday

Cloud computing platforms

INFRASTRUCTURE AS A SERVICE: AMAZON EC2

The Amazon cloud provides infrastructure as a service (IaaS), whereby computing infrastructure such as for servers, storage or network end points of a desired capacity are virtually provisioned in minutes through an automated web-based management console. This core IaaS service, called Elastic Compute Cloud, or EC2, is but one of a set of services that constitute the Amazon cloud platform, but the term EC2 is also often used to describe the entire cloud offering.

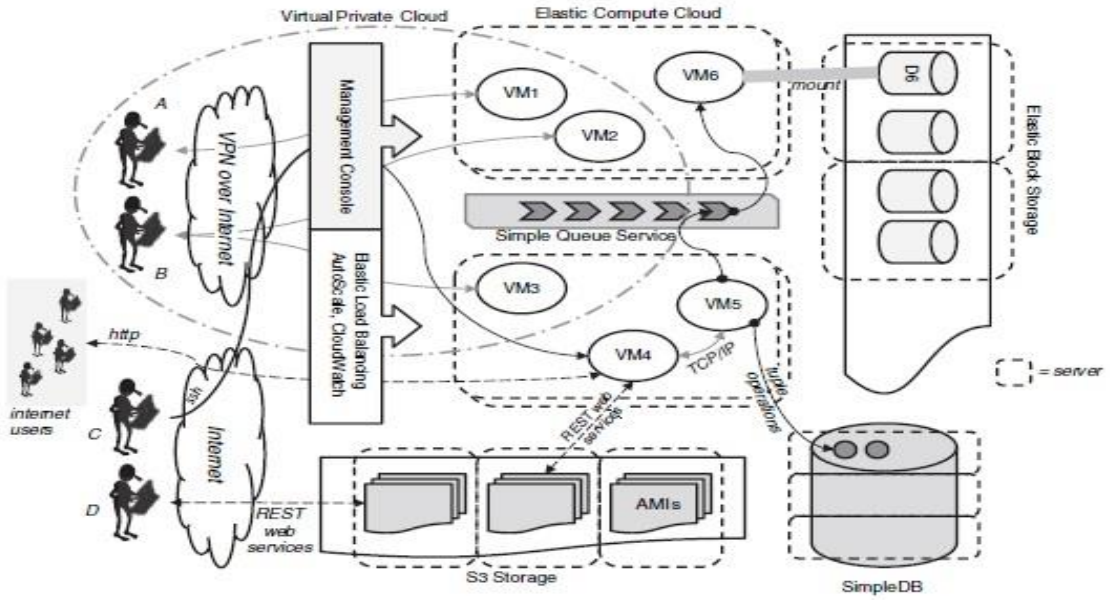


FIGURE 5.1. Amazon infrastructure cloud

The above figure illustrates the services provided by the Amazon infrastructure cloud from a user perspective. These services are implemented on a very large network of servers, shown as dashed

boxes in the figure. The Elastic Compute Cloud service provides users access to dedicated virtual machines of a desired capacity that are provisioned on these physical servers, with details of the actual physical server, such as its location, capacity, etc. being transparent to the end-user. Through the management console users generate PKI1 key-pairs using which they can securely login to these virtual servers over the internet.

The user's account is charged on an hourly basis based on actual consumption, i.e. time that the server is up. Charges vary depending on the AMI used and capacity of server chosen while provisioning. For example, a 'small' Linux server costs a few cents per cpu-hour, whereas a larger server preloaded with licensed software, such as Windows, as well as other database or middleware products, could end up costing close to a dollar per hour

An important goal of any cloud service is insulating users from variable demand by automatically managing scaling up and down of the resources allocated to a cloud application. In an infrastructure cloud, such as Amazon EC2, the user needs to explicitly define an architecture that enables scalability using tools provided by Amazon to manage elasticity: Runtime performance parameters, such as CPU and I/O utilization, of a user's virtual servers can be monitored in real-time by Amazon Cloud Watch; this data can be used by Amazon Auto Scale to add (or remove) virtual servers from an application cluster and automatically provision them with predefined machine images. Finally, Elastic Load Balancing allows a group of servers to be configured into a set across which incoming requests (e.g. HTTP connections) are load balanced. The performance statistics of the load-balanced requests can also be monitored by Cloud Watch and used by Auto Scale to add or remove servers from the load balanced cluster. Using these tools users can configure a scalable architecture that can also elastically adjust its resource consumption. Note however that for a complex architecture, such as multi-tier transaction processing system, there may need to be many layers of clustering, e.g. at the web server, application server, database server etc. It remains the user's responsibility to configure a scalable cluster for each of these layers, define what performance parameters need to be monitored in Cloud Watch and set the Auto Scale parameters for each cluster

PLATFORM AS A SERVICE: GOOGLE APP ENGINE

The Google cloud, called Google App Engine, is a 'platform as a service' (PaaS) offering. In contrast with the Amazon infrastructure as a service cloud, where users explicitly provision virtual machines and control them fully, including installing, compiling and running software on them, a PaaS offering hides the actual execution environment from users. Instead, a software platform is provided along with an SDK, using which users develop applications and deploy them on the cloud. The PaaS platform is responsible for executing the applications, including servicing external service requests, as well as running scheduled jobs included in the application. By making the actual execution servers transparent to the user, a PaaS platform is able to share application servers across users who need lower capacities, as well as automatically scale resources allocated to applications that experience heavy loads

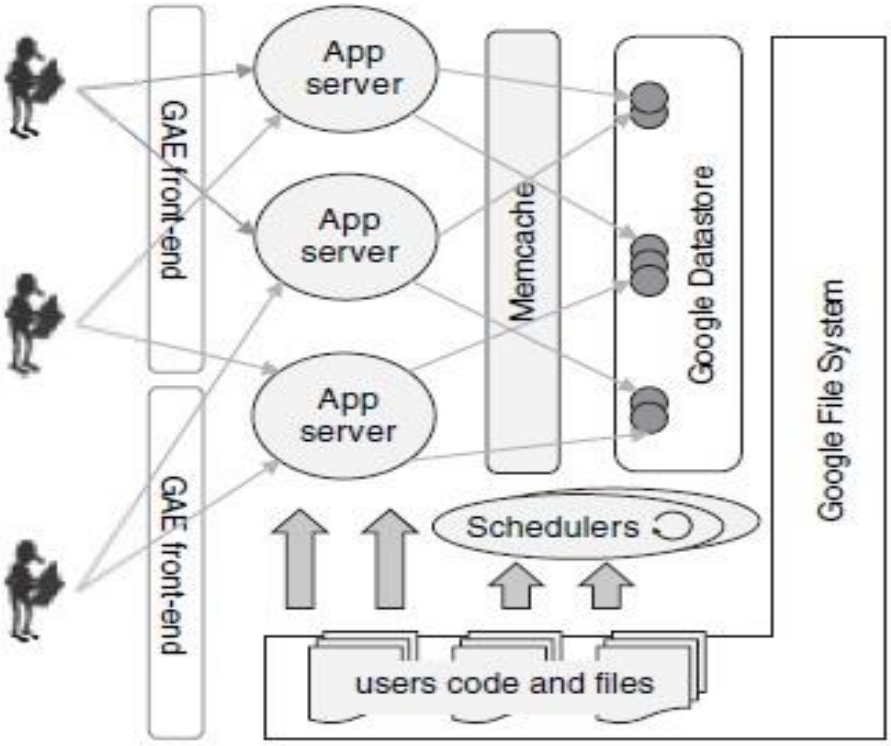


FIGURE 5.2. Google App Engine

Resource usage for an application is metered in terms of web requests served and CPU-hours actually spent executing requests or batch jobs. Note that this is very different from the IaaS model: A PaaS application can be deployed and made globally available 24×7, but charged only when accessed (or if batch jobs run); in contrast, in an IaaS model merely making an application continuously available incurs the full cost of keeping at least some of the servers running all the

time. Further, deploying applications in Google App Engine is free, within usage limits; thus applications can be developed and tried out free and begin to incur cost only when actually accessed by a sufficient volume of requests. The PaaS model enables Google to provide such a free service because applications do not run in dedicated virtual machines; a deployed application that is not accessed merely consumes storage for its code and data and expends no CPU cycles.

GAE applications are served by a large number of web servers in Google's data centers that execute requests from end-users across the globe. The web servers load code from the GFS into memory and serve these requests. Each request to a particular application is served by any one of GAE's web servers; there is no guarantee that the same server will serve requests to any two requests, even from the same HTTP session. Applications can also specify some functions to be executed as batch jobs which are run by a scheduler.

While this architecture is able to ensure that applications scale naturally as load increases, it also means that application code cannot easily rely on in-memory data. A distributed in-memory cache called Memcache is made available to partially address this issue: In particular HTTP sessions are implemented using Memcache so that even if requests from the same session go to different servers they can retrieve their session data, most of the time (since Memcache is not guaranteed to always retain cached data).

MICROSOFT AZURE

Microsoft's cloud offering, called Azure, has been commercially released to the public only recently, though a community preview beta has been publicly available for longer. Thus, it is important to note that some elements of the Azure platform are likely to change in future commercial editions.

Azure is a PaaS offering. Developers create applications using Microsoft development tools (i.e. Visual Studio along with its supported languages, C#, Visual Basic, ASPs, etc.); an Azure extension to the standard toolset allows such applications to be developed and deployed on Microsoft's cloud, in much the same manner as developing and deploying using Google App Engine's SDK. There are also similarities with aspects of Amazon's IaaS offering, such as the use of virtualization, user control over the number of virtual servers allocated to an application and user control on elasticity. However, unlike the non-relational Google Datastore and Amazon

SimpleDB, the recently released commercial edition of Azure provides relational storage services, albeit with certain limitations as we cover below. Azure also allows storage of arbitrary files and objects like Amazon S3 as well as a queuing service similar to Amazon SQS.

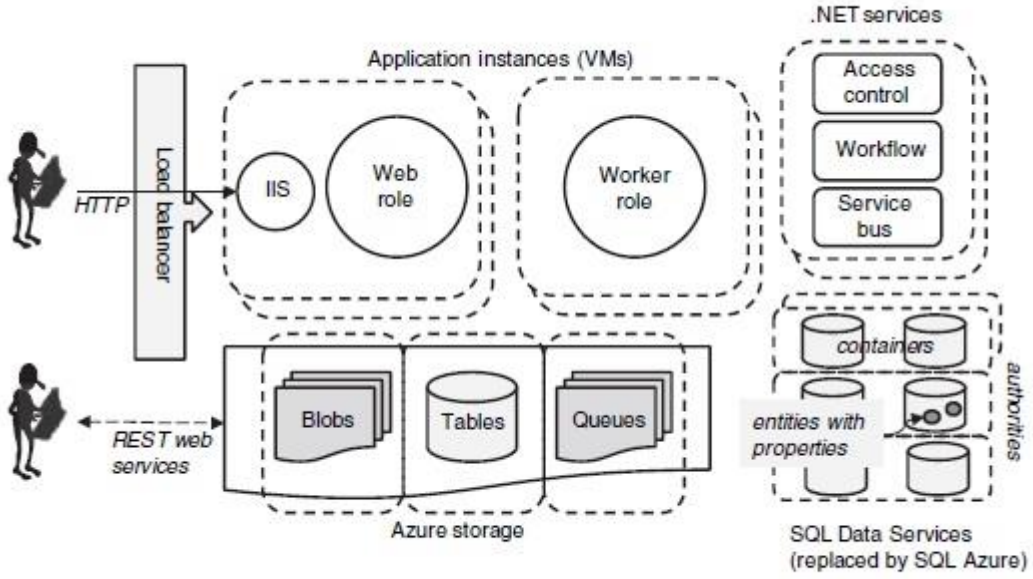


FIGURE 5.4. Microsoft Azure

Application code can be deployed on Azure as a web role or a worker role. Code in web-role instances is run through a web server (Microsoft IIS) that is included in the instance. Web roles respond to HTTP requests that are automatically load balanced across application instances. Worker role code can run as a batch process that can communicate with a web role through Azure data storage, such as queues or tables. Worker role applications cannot be accessed from an external network, but can make external HTTP requests, either to worker roles or over the internet.

Applications access SQL Data Services at the level of containers, through a global address scheme. Each container contains entities which have properties, and like SimpleDB or Google Datastore, properties for each entity can differ in type and number. Microsoft SQL Data Services was rebranded as 'SQL Azure' in the commercial edition of Azure. The architecture of authorities, containers and tables has been replaced by a traditional relational model supported by Microsoft SQL Server. In particular this includes support for joins between tables, transactions and other features of relational database. However, as of this writing, each SQL Azure database

is limited to under 10 GB in size. In case larger volumes of data need to be stored, multiple virtual database instances need to be provisioned.

In addition to compute and storage services, Microsoft Azure also provides what are called .NET services. These include access control services that provide globally configurable and accessible security tokens, a service bus that enables globally published service end points and a configurable web-service based workflow-orchestration service. Like SQL Data services and SQL Azure, these are all based on Microsoft's enterprise middleware products for identity management and web services, deployed in the cloud on a distributed and globally accessible platform.

UTILITY COMPUTING

Utility computing is the process of providing computing service through an on-demand, pay-per-use billing method. Utility computing is a computing business model in which the provider owns, operates and manages the computing infrastructure and resources, and the subscribers accesses it as and when required on a rental or metered basis.

Utility computing is one of the most popular IT service models, primarily because of the flexibility and economy it provides. This model is based on that used by conventional utilities such as telephone services, electricity and gas. The principle behind utility computing is simple. The consumer has access to a virtually unlimited supply of computing solutions over the Internet or a virtual private network, which can be sourced and used whenever it's required. The back-end infrastructure and computing resources management and delivery is governed by the provider.

ELASTIC COMPUTING

Elastic computing is a concept in cloud computing in which computing resources can be scaled up and down easily by the cloud service provider. Elastic computing is the ability of a cloud service provider to provision flexible computing power when and wherever required. The elasticity of these resources can be in terms of processing power, storage, bandwidth, etc.

Cloud computing is about provisioning on-demand computing resources with the simplicity of a mouse click. The amount of resources which can be sourced through cloud computing incorporates almost all the facets of computing from raw processing power to massive storage space.

Besides providing these services on demand basis, the resources are elastic in nature, i.e. they can be easily scaled depending upon the underlying resource requirements on run time without even disrupting the operations and this ability is known as elastic computing. On a small scale this is done manually, but for larger installations, the scaling is automatic. For example, a larger provider of online video could setup a system so that the number of webservers online scaled during peak viewing hours.

.