

3

Cloud Computing Architecture and Management

Learning Objectives

The objectives of this chapter are to

- Provide an overview of the cloud architecture
- Give an insight on the anatomy of the cloud
- Describe the role of network connectivity in the cloud
- Give a description about applications in the cloud
- Give a detailed description about managing the cloud
- Provide an overview about application migration to the cloud

Preamble

Cloud computing is an emerging technology that has become one of the most popular computing technologies. Each and every technology has certain concepts that form the basis for its working. Similarly, there are several aspects of a technology that needs to be looked upon before delving deeper. Thus, there are some basic issues in cloud computing that need to be discussed before going into a detailed discussion about the cloud. This chapter firstly describes the cloud architecture. Cloud architecture consists of a hierarchical set of components that collectively describe the way the cloud works. The next section explains about the cloud anatomy, followed by network connectivity in the cloud and then the fine details about managing a cloud application. Finally, an overview on migrating applications to the cloud is discussed. Some of the topics that are discussed in this chapter are elaborated in upcoming chapters.

3.1 Introduction

Cloud computing is similar to other technologies in a way that it also has several basic concepts that one should learn before knowing its core concepts. There are several processes and components of cloud computing that need to be discussed. One of the topics of such prime importance is architecture. Architecture is the hierarchical view of describing a technology. This usually includes the components over which the existing technology is built and the components that are dependent on the technology. Another topic that is related to architecture is anatomy. Anatomy describes the core structure of the cloud. Once the structure of the cloud is clear, the network connections in the cloud and the details about the cloud application need to be known. This is important as the cloud is a completely Internet-dependent technology. Similarly, cloud management discusses the important management issues and ways in which the current cloud scenario is managed. It describes the way an application and infrastructure in the cloud are managed. Management is important because of the quality of service (QoS) factors that are involved in the cloud. These QoS factors form the basis for cloud computing. All the services are given based on these QoS factors. Similarly, application migration to the cloud also plays a very important role. Not all applications can be directly deployed to the cloud. An application needs to be properly migrated to the cloud to be considered a proper cloud application that will have all the properties of the cloud.

3.2 Cloud Architecture

Any technological model consists of an architecture based on which the model functions, which is a hierarchical view of describing the technology. The cloud also has an architecture that describes its working mechanism. It includes the dependencies on which it works and the components that work over it. The cloud is a recent technology that is completely dependent on the Internet for its functioning. Figure 3.1 depicts the architecture. The cloud architecture can be divided into four layers based on the access of the cloud by the user. They are as follows.

3.2.1 Layer 1 (User/Client Layer)

This layer is the lowest layer in the cloud architecture. All the users or client belong to this layer. This is the place where the client/user initiates the

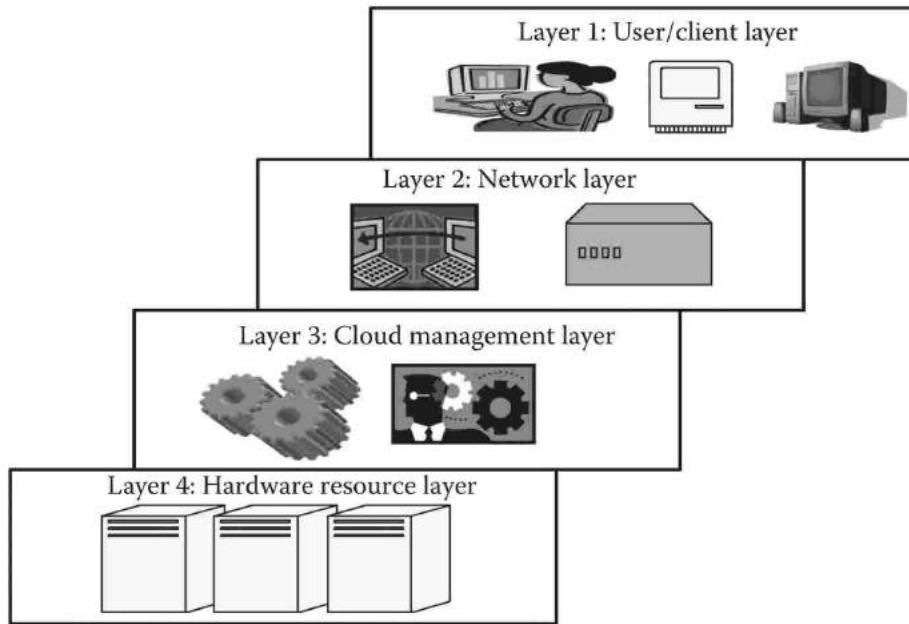


FIGURE 3.1
Cloud architecture.

connection to the cloud. The client can be any device such as a thin client, thick client, or mobile or any handheld device that would support basic functionalities to access a web application. The thin client here refers to a device that is completely dependent on some other system for its complete functionality. In simple terms, they have very low processing capability. Similarly, thick clients are general computers that have adequate processing capability. They have sufficient capability for independent work. Usually, a cloud application can be accessed in the same way as a web application. But internally, the properties of cloud applications are significantly different. Thus, this layer consists of client devices.

3.2.2 Layer 2 (Network Layer)

This layer allows the users to connect to the cloud. The whole cloud infrastructure is dependent on this connection where the services are offered to the customers. This is primarily the Internet in the case of a public cloud. The public cloud usually exists in a specific location and the user would not know the location as it is abstract. And, the public cloud can be accessed all over the world. In the case of a private cloud, the connectivity may be provided by a local area network (LAN). Even in this case, the cloud completely depends on the network that is used. Usually, when accessing the public or private cloud, the users require minimum bandwidth, which is sometimes defined by the cloud providers. This layer does not come under the purview of service-level agreements (SLAs), that is, SLAs do not take into account the Internet connection between the user and cloud for quality of service (QoS).

3.2.3 Layer 3 (Cloud Management Layer)

This layer consists of softwares that are used in managing the cloud. The softwares can be a cloud operating system (OS), a software that acts as an interface between the data center (actual resources) and the user, or a management software that allows managing resources. These softwares usually allow resource management (scheduling, provisioning, etc.), optimization (server consolidation, storage workload consolidation), and internal cloud governance. This layer comes under the purview of SLAs, that is, the operations taking place in this layer would affect the SLAs that are being decided upon between the users and the service providers. Any delay in processing or any discrepancy in service provisioning may lead to an SLA violation. As per rules, any SLA violation would result in a penalty to be given by the service provider. These SLAs are for both private and public clouds Popular service providers are Amazon Web Services (AWS) and Microsoft Azure for public cloud. Similarly, OpenStack and Eucalyptus allow private cloud creation, deployment, and management.

3.2.4 Layer 4 (Hardware Resource Layer)

Layer 4 consists of provisions for actual hardware resources. Usually, in the case of a public cloud, a data center is used in the back end. Similarly, in a private cloud, it can be a data center, which is a huge collection of hardware resources interconnected to each other that is present in a specific location or a high configuration system. This layer comes under the purview of SLAs. This is the most important layer that governs the SLAs. This layer affects the SLAs most in the case of data centers. Whenever a user accesses the cloud, it should be available to the users as quickly as possible and should be within the time that is defined by the SLAs. As mentioned, if there is any discrepancy in provisioning the resources or application, the service provider has to pay the penalty. Hence, the data center consists of a high-speed network connection and a highly efficient algorithm to transfer the data from the data center to the manager. There can be a number of data centers for a cloud, and similarly, a number of clouds can share a data center.

Thus, this is the architecture of a cloud. The layering is strict, and for any cloud application, this is followed. There can be a little loose isolation between layer 3 and layer 4 depending on the way the cloud is deployed.

3.3 Anatomy of the Cloud

Cloud anatomy can be simply defined as the structure of the cloud. Cloud anatomy cannot be considered the same as cloud architecture. It may not include any dependency on which or over which the technology works,

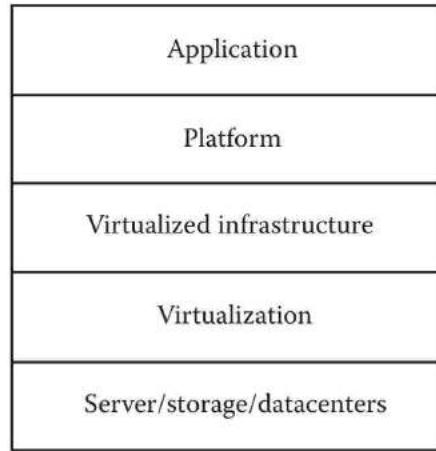


FIGURE 3.2
Cloud structure.

whereas architecture wholly defines and describes the technology over which it is working. Architecture is a hierarchical structural view that defines the technology as well as the technology over which it is dependent or/and the technology that are dependent on it. Thus, anatomy can be considered as a part of architecture. The basic structure of the cloud is described in Figure 3.2, which can be elaborated, and minute structural details can be given. Figure 3.2 depicts the most standard anatomy that is the base for the cloud. It depends on the person to choose the depth of description of the cloud. A different view of anatomy is given by Refs. [1,2].

There are basically five components of the cloud:

1. *Application*: The upper layer is the application layer. In this layer, any applications are executed.
2. *Platform*: This component consists of platforms that are responsible for the execution of the application. This platform is between the infrastructure and the application.
3. *Infrastructure*: The infrastructure consists of resources over which the other components work. This provides computational capability to the user.
4. *Virtualization*: Virtualization is the process of making logical components of resources over the existing physical resources. The logical components are isolated and independent, which form the infrastructure.
5. *Physical hardware*: The physical hardware is provided by server and storage units.

These components are the basis and are described in detail in further chapters.

3.4 Network Connectivity in Cloud Computing

Cloud computing is a technique of resource sharing where servers, storage, and other computing infrastructure in multiple locations are connected by networks. In the cloud, when an application is submitted for its execution, needy and suitable resources are allocated from this collection of resources; as these resources are connected via the Internet, the users get their required results. For many cloud computing applications, network performance will be the key issue to cloud computing performance. Since cloud computing has various deployment options, we now consider the important aspects related to the cloud deployment models and their accessibility from the viewpoint of network connectivity.

3.4.1 Public Cloud Access Networking

In this option, the connectivity is often through the Internet, though some cloud providers may be able to support virtual private networks (VPNs) for customers. Accessing public cloud services will always create issues related to security, which in turn is related to performance. One of the possible approaches toward the support of security is to promote connectivity through encrypted tunnels, so that the information may be sent via secure pipes on the Internet. This procedure will be an overhead in the connectivity, and using it will certainly increase delay and may impact performance.

If we want to reduce the delay without compromising security, then we have to select a suitable routing method such as the one reducing the delay by minimizing transit *hops* in the end-to-end connectivity between the cloud provider and cloud consumer. Since the end-to-end connectivity support is via the Internet, which is a complex federation of interconnected providers (known as Internet service providers [ISPs]), one has to look at the options of selecting the path.

3.4.2 Private Cloud Access Networking

In the private cloud deployment model, since the cloud is part of an organizational network, the technology and approaches are local to the in-house network structure. This may include an Internet VPN or VPN service from a network operator. If the application access was properly done with an organizational network—connectivity in a *precloud* configuration—transition to private cloud computing will not affect the access performance.

3.4.3 Intracloud Networking for Public Cloud Services

Another network connectivity consideration in cloud computing is intra-cloud networking for public cloud services. Here, the resources of the

cloud provider and thus the cloud service to the customer are based on the resources that are geographically apart from each other but still connected via the Internet. Public cloud computing networks are internal to the service provider and thus not visible to the user/customer; however, the security aspects of connectivity and the access mechanisms of the resources are important. Another issue to look for is the QoS in the connected resources worldwide. Most of the performance issues and violations from these are addressed in the SLAs commercially.

3.4.4 Private Intracloud Networking

The most complicated issue for networking and connectivity in cloud computing is private intracloud networking. What makes this particular issue so complex is that it depends on how much intracloud connectivity is associated with the applications being executed in this environment. Private intracloud networking is usually supported over connectivity between the major data center sites owned by the company. At a minimum, all cloud computing implementations will rely on intracloud networking to link users with the resource to which their application was assigned. Once the resource linkage is made, the extent to which intracloud networking is used depends on whether the application is componentized based on *service-oriented architecture (SOA)* or not, among multiple systems. If the principle of SOA is followed, then traffic may move between components of the application, as well as between the application and the user. The performance of those connections will then impact cloud computing performance overall. Here too, the impact of cloud computing performance is the differences that exist between the current application and the network relationships with the application.

There are reasons to consider the networks and connectivity in cloud computing with newer approaches as globalization and changing network requirements, especially those related to increased Internet usage, are demanding more flexibility in the network architectures of today's enterprises. How are these related to us? The answers are discussed later.

3.4.5 New Facets in Private Networks

Conventional private networks have been architected for on-premise applications and maximum Internet security. Typically, applications such as e-mail, file sharing, and *enterprise resource planning (ERP)* systems are delivered to on-premise-based servers at each corporate data center. Increasingly today, software vendors are offering Software as a Service (SaaS) as an alternative for their software support to the corporate offices, which brings more challenges in the access and usage mechanisms of software from data center servers and in the connectivity of network architectures. The traditional network architecture for these global enterprises was not designed to optimize performance for cloud applications, now that many applications including

mission-critical applications are transitioning (moving) from on-premise based to cloud based, wherein the network availability becomes as mission critical as electricity: the business cannot function if it cannot access applications such as ERP and e-mail.

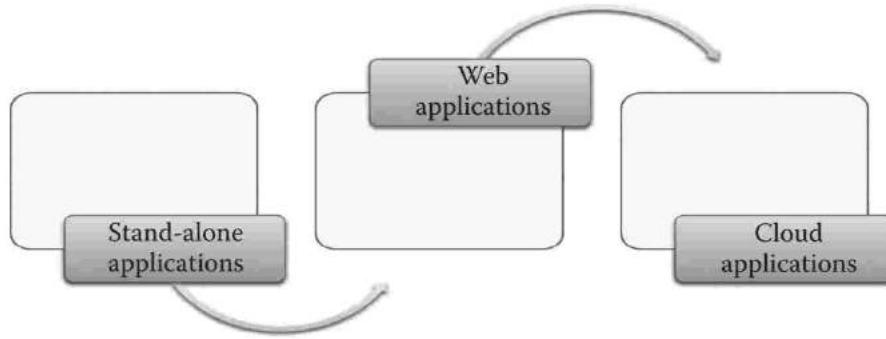
3.4.6 Path for Internet Traffic

The traditional Internet traffic through a limited set of Internet gateways poses performance and availability issues for end users who are using cloud-based applications. It can be improved if a more widely distributed Internet gateway infrastructure and connectivity are being supported for accessing applications, as they will provide lower-latency access to their cloud applications. As the volume of traffic to cloud applications grows, the percentage of the legacy network's capacity in terms of traffic to regional gateways increases. Applications such as video conferencing would hog more bandwidth while mission-critical applications such as ERP will consume less bandwidth, and hence, one has to plan a correct connectivity and path between providers and consumers.

3.5 Applications on the Cloud

The power of a computer is realized through the applications. There are several types of applications. The first type of applications that was developed and used was a stand-alone application. A stand-alone application is developed to be run on a single system that does not use network for its functioning. These stand-alone systems use only the machine in which they are installed. The functioning of these kinds of systems is totally dependent on the resources or features available within the system. These systems do not need the data or processing power of other systems; they are self-sustaining. But as the time passed, the requirements of the users changed and certain applications were required, which could be accessed by other users away from the systems. This led to the inception of web application.

The web applications were different from the stand-alone applications in many aspects. The main difference was the client server architecture that was followed by the web application. Unlike stand-alone applications, these systems were totally dependent on the network for its working. Here, there are basically two components, called as the client and the server. The server is a high-end machine that consists of the web application installed. This web application is accessed from other client systems. The client can reside anywhere in the network. It can access the web application through the Internet. This type of application was very useful, and this is extensively used from its inception and now has become an

**FIGURE 3.3**

Computer application evolution.

important part of day-to-day life. Though this application is much used, there are shortcomings as discussed in the following:

- The web application is not elastic and cannot handle very heavy loads, that is, it cannot serve highly varying loads.
- The web application is not multitenant.
- The web application does not provide a quantitative measurement of the services that are given to the users, though they can monitor the user.
- The web applications are usually in one particular platform.
- The web applications are not provided on a pay-as-you-go basis; thus, a particular service is given to the user for permanent or trial use and usually the timings of user access cannot be monitored.
- Due to its nonelastic nature, peak load transactions cannot be handled.

Primarily to solve the previously mentioned problem, the cloud applications were developed. Figure 3.3 depicts the improvements in the applications.

The cloud as mentioned can be classified into three broad access or service models, Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Cloud application in general refers to a SaaS application.

A cloud application is different from other applications; they have unique features. A cloud application usually can be accessed as a web application but its properties differ. According to NIST [3], the features that make cloud applications unique are described in the following (Figure 3.4 depicts the features of a cloud application):

1. **Multitenancy:** Multitenancy is one of the important properties of cloud that make it different from other types of application in which the software can be shared by different users with full independence. Here, independence refers to logical independence.

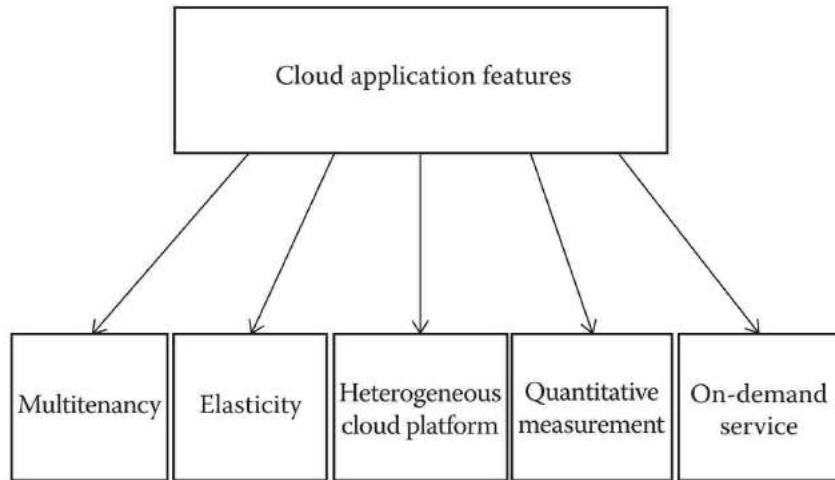


FIGURE 3.4
Features of cloud.

Each user will have a separate application instance and the changes in one application would not affect the other. Physically, the software is shared and is not independent. The degree of physical isolation is very less. The logical independence is what is guaranteed. There are no restrictions in the number of applications being shared. The difficulty in providing logical isolation depends on the physical isolation to a certain extent. If an application is physically too close, then it becomes difficult to provide multitenancy. Web application and cloud application are similar as the users use the same way to access both. Figure 3.5 depicts a multitenant application where several users share the same application.

2. *Elasticity*: Elasticity is also a unique property that enables the cloud to serve better. According to Herbst et al. [4], elasticity can be defined as the degree to which a system is able to adapt to workload changes by provisioning and deprovisioning resources in an autonomic manner such that at each point in time, the available resources match the current demand as closely as possible. Elasticity allows the cloud providers to efficiently handle the number of users, from one to

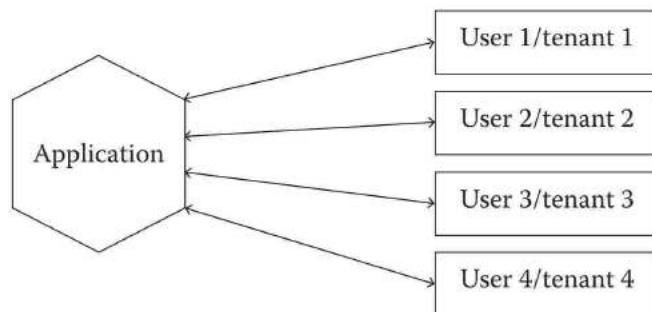


FIGURE 3.5
Multitenancy.

several hundreds of users at a time. In addition to this, it supports the rapid fluctuation of loads, that is, the increase or decrease in the number of users and their usage can rapidly change.

3. *Heterogeneous cloud platform*: The cloud platform supports heterogeneity, wherein any type of application can be deployed in the cloud. Because of this property, the cloud is flexible for the developers, which facilitates deployment. The applications that are usually deployed can be accessed by the users using a web browser.
4. *Quantitative measurement*: The services provided can be quantitatively measured. The user is usually offered services based on certain charges. Here, the application or resources are given as a utility on a pay-per-use basis. Thus, the use can be monitored and measured. Not only the services are measureable, but also the link usage and several other parameters that support cloud applications can be measured. This property of measuring the usage is usually not available in a web application and is a unique feature for cloud-based applications.
5. *On-demand service*: The cloud applications offer service to the user, on demand, that is, whenever the user requires it. The cloud service would allow the users to access web applications usually without any restrictions on time, duration, and type of device used.

The previously mentioned properties are some of the features that make cloud a unique application platform. These properties mentioned are specific to the cloud hence making it as one of the few technologies that allows application developers to suffice the user's needs seamlessly without any disruption.

3.6 Managing the Cloud

Cloud management is aimed at efficiently managing the cloud so as to maintain the QoS. It is one of the prime jobs to be considered. The whole cloud is dependent on the way it is managed. Cloud management can be divided into two parts:

1. Managing the infrastructure of the cloud
2. Managing the cloud application

3.6.1 Managing the Cloud Infrastructure

The infrastructure of the cloud is considered to be the backbone of the cloud. This component is mainly responsible for the QoS factor. If the infrastructure is not properly managed, then the whole cloud can fail and QoS would

be adversely affected. The core of cloud management is resource management. Resource management involves several internal tasks such as resource scheduling, provisioning, and load balancing. These tasks are mainly managed by the cloud service provider's core software capabilities such as the cloud OS that is responsible for providing services to the cloud and that internally controls the cloud. A cloud infrastructure is a very complex system that consists of a lot of resources. These resources are usually shared by several users.

Poor resource management may lead to several inefficiencies in terms of performance, functionality, and cost. If a resource is not efficiently managed, the performance of the whole system is affected. Performance is the most important aspect of the cloud, because everything in the cloud is dependent on the SLAs and the SLAs can be satisfied only if performance is good. Similarly, the basic functionality of the cloud should always be provided and considered at any cost. Even if there is a small discrepancy in providing the functionality, the whole purpose of maintaining the cloud is futile. A partially functional cloud would not satisfy the SLAs.

Lastly, the reason for which the cloud was developed was cost. The cost is a very important criterion as far as the business prospects of the cloud are concerned. On the part of the service providers, if they incur less cost for managing the cloud, then they would try to reduce the cost so as to get a strong user base. Hence, a lot of users would use the services, improving their profit margin. Similarly, if the cost of resource management is high, then definitely the cost of accessing the resources would be high and there is never a lossy business from any organization and so the service provider would not bear the cost and hence the users have to pay more. Similarly, this would prove costly for service providers as they have a high chance of losing a wide user base, leading to only a marginal growth in the industry. And, competing with its industry rivals would become a big issue. Hence, efficient management with less cost is required.

At a higher level, other than these three issues, there are few more issues that depend on resource management. These are power consumption and optimization of multiple objectives to further reduce the cost. To accomplish these tasks, there are several approaches followed, namely, consolidation of server and storage workloads. Consolidation would reduce the energy consumption and in some cases would increase the performance of the cloud. According to Margaret Rouse [5], server consolidation by definition is an approach to the efficient usage of computer server resources in order to reduce the total number of servers or server locations that an organization requires.

The previously discussed prospects are mostly suitable for IaaS. Similarly, there are different management methods that are followed for different types of service delivery models. Each of the type has its own way of management. All the management methodologies are based on

load fluctuation. Load fluctuation is the point where the workload of the system changes continuously. This is one of the important criteria and issues that should be considered for cloud applications. Load fluctuation can be divided into two types: predictable and unpredictable. Predictable load fluctuations are easy to handle. The cloud can be preconfigured for handling such kind of fluctuations. Whereas unpredictable load fluctuations are difficult to handle, ironically this is one of the reasons why cloud is preferred by several users.

This is as far as cloud management is concerned. Cloud governance is another topic that is closely related to cloud management. Cloud governance is different from cloud management. Governance in general is a term in the corporate world that generally involves the process of creating value to an organization by creating strategic objectives that will lead to the growth of the company and would maintain a certain level of control over the company. Similar to that, here cloud organization is involved.

There are several aspects of cloud governance out of which SLAs are one of the important aspects. SLAs are the set of rules that are defined between the user and cloud service provider that decide upon the QoS factor. If SLAs are not followed, then the defaulter has to pay the penalty. The whole cloud is governed by keeping these SLAs in mind. Cloud governance is discussed in detail in further chapters.

3.6.2 Managing the Cloud Application

Business companies are increasingly looking to move or build their corporate applications on cloud platforms to improve agility or to meet dynamic requirements that exist in the globalization of businesses and responsiveness to market demands. But, this shift or moving the applications to the cloud environment brings new complexities. Applications become more composite and complex, which requires leveraging not only capabilities like storage and database offered by the cloud providers but also third-party SaaS capabilities like e-mail and messaging. So, understanding the availability of an application requires inspecting the infrastructure, the services it consumes, and the upkeep of the application. The composite nature of cloud applications requires visibility into all the services to determine the overall availability and uptime.

Cloud application management is to address these issues and propose solutions to make it possible to have insight into the application that runs in the cloud, as well as implement or enforce enterprise policies like governance and auditing and environment management while the application is deployed in the cloud. These cloud-based monitoring and management services can collect a multitude of events, analyze them, and identify critical information that requires additional remedial actions like adjusting capacity or provisioning new services. Additionally,

application management has to be supported with tools and processes required for managing other environments that might coexist, enabling efficient operations.

3.7 Migrating Application to Cloud

Cloud migration encompasses moving one or more enterprise applications and their IT environments from the traditional hosting type to the cloud environment, either public, private, or hybrid. Cloud migration presents an opportunity to significantly reduce costs incurred on applications. This activity comprises, of different phases like evaluation, migration strategy, prototyping, provisioning, and testing.

3.7.1 Phases of Cloud Migration

1. *Evaluation:* Evaluation is carried out for all the components like current infrastructure and application architecture, environment in terms of compute, storage, monitoring, and management, SLAs, operational processes, financial considerations, risk, security, compliance, and licensing needs are identified to build a business case for moving to the cloud.
2. *Migration strategy:* Based on the evaluation, a migration strategy is drawn—a hotplug strategy is used where the applications and their data and interface dependencies are isolated and these applications can be operationalized all at once. A fusion strategy is used where the applications can be partially migrated; but for a portion of it, there are dependencies based on existing licenses, specialized server requirements like mainframes, or extensive interconnections with other applications.
3. *Prototyping:* Migration activity is preceded by a prototyping activity to validate and ensure that a small portion of the applications are tested on the cloud environment with test data setup.
4. *Provisioning:* Premigration optimizations identified are implemented. Cloud servers are provisioned for all the identified environments, necessary platform softwares and applications are deployed, configurations are tuned to match the new environment sizing, and databases and files are replicated. All internal and external integration points are properly configured. Web services, batch jobs, and operation and management software are set up in the new environments.
5. *Testing:* Postmigration tests are conducted to ensure that migration has been successful. Performance and load testing, failure and recovery testing, and scale-out testing are conducted against the expected traffic load and resource utilization levels.

3.7.2 Approaches for Cloud Migration

The following are the four broad approaches for cloud migration that have been adopted effectively by vendors:

1. *Migrate existing applications*: Rebuild or rearchitect some or all the applications, taking advantage of some of the virtualization technologies around to accelerate the work. But, it requires top engineers to develop new functionality. This can be achieved over the course of several releases with the timing determined by customer demand.
2. *Start from scratch*: Rather than cannibalize sales, confuse customers with choice, and tie up engineers trying to rebuild existing application, it may be easier to start again. Many of the R&D decisions will be different now, and with some of the more sophisticated development environments, one can achieve more even with a small focused working team.
3. *Separate company*: One may want to create a whole new company with separate brand, management, R&D, and sales. The investment and internet protocol (IP) may come from the existing company, but many of the conflicts disappear once a new *born in the cloud* company is established. The separate company may even be a subsidiary of the existing company. What is important is that the new company can act, operate, and behave like a cloud-based start-up.
4. *Buy an existing cloud vendor*: For a large established vendor, buying a cloud-based competitor achieves two things. Firstly, it removes a competitor, and secondly, it enables the vendor to hit the ground running in the cloud space. The risk of course is that the innovation, drive, and operational approach of the cloud-based company are destroyed as it is merged into the larger acquirer.

3.8 Summary

Cloud computing has several concepts that must be understood before starting off with the details about the cloud, which include one of the important concepts of cloud architecture. It consists of a basic hierarchical structure with dependencies of components specified. Similarly, anatomy is also important as it describes the basic structure about the cloud, though it does not consider any dependency as in architecture. Further, the cloud network connectivity that forms the core of the cloud model is important. The network is the base using which the cloud works. Similarly, cloud management is one of the important concepts that describe the way in which the cloud is managed, and it has two components: infrastructure management and application

management. Both are important as both affect the QoS. Finally, an application should be successfully migrated to a cloud. An application will radiate its complete properties as a cloud only when it has perfectly migrated.

Review Points

- *Cloud architecture*: Cloud architecture consists of a hierarchical set of components that collectively describe the way the cloud works. It is a view of a system (see Section 3.4).
 - *Cloud anatomy*: Cloud anatomy is the basic structure of the cloud (see Section 3.5).
 - *SLA*: SLAs are a set of agreements that are signed between the user and service providers (see Section 3.8.1).
 - *Elasticity*: Elasticity can be defined as the degree to which a system is able to adapt to the workload changes by provisioning and deprovisioning resources in an autonomic manner, such that at each point in time, the available resources match the current demand as closely as possible (see Section 3.7).
 - *Multitenancy*: Multitenancy is a property of the cloud by which the software can be shared by different users with full independence (see Section 3.7).
 - *Stand-alone application*: A stand-alone application is developed to be run on a single system that does not use a network for its functioning (see Section 3.7).
 - *Server consolidation*: Server consolidation by definition is an approach to the efficient usage of computer server resources in order to reduce the total number of servers or server locations that an organization requires (see Section 3.8.1).
-

Review Questions

1. What is server consolidation?
2. How is cloud anatomy different from cloud architecture?
3. What are the unique properties of cloud applications?
4. What are the two different management classifications?
5. Why are SLAs important?

6. Describe several approaches of cloud migration.
 7. What is public cloud access networking?
 8. List the phases of cloud migration.
 9. What are the drawbacks of a web application?
 10. What is elasticity?
 11. Explain the pay-as-you-go paradigm.
-

References

1. The anatomy of cloud computing. Available [Online]: <http://www.niallkennedy.com/blog/2009/03/cloud-computing-stack.html>. Accessed May 1, 2014.
 2. Anatomy of cloud platform. Available [Online]: <https://linux.sys-con.com/node/1120648>. Accessed May 2, 2014.
 3. Mell, P. and T. Grance. The NIST definition of cloud computing (draft). NIST Special Publication 800.145: 7, 2011.
 4. Herbst, N.R., S. Kounev, and R. Reussner. Elasticity in cloud computing: What it is, and what it is not. *Proceedings of the 10th International Conference on Autonomic Computing (ICAC 2013)*, San Jose, CA, 2013.
 5. Server consolidation. Available [Online]: <http://searchdatacenter.techtarget.com/definition/server-consolidation>. Accessed April 28, 2014.
-

Further Reading

Network considerations in cloud computing. Available [Online]: <http://searchcloudcomputing.techtarget.com/tip/Network-considerations-in-cloud-computing>. Accessed May 3, 2014.

4

Cloud Deployment Models

Learning Objectives

The objectives of this book chapter are to

- Introduce the readers to cloud deployment models
 - Describe the cloud deployments in detail
 - Analyze the advantages and disadvantages of each deployed models
 - Discuss the problems related to each deployment model
 - Elaborate the deployments models based on properties like SLA and security
-

Preamble

This chapter broadly discusses the deployment models available in the cloud which are one of the most important concepts related to cloud computing. The deployment models are the different ways in which the cloud computing environment can be set up, that is, the several ways in which the cloud can be deployed. It is important to have an idea about the deployment models because setting up a cloud is the most basic requirement prior to starting any further study about cloud computing. Cloud computing is business oriented, and the popularity of the cloud is credited to its market-oriented nature. In the business perspective, making the correct decision regarding the deployment model is very important. A model should be selected based on the needs, requirements, budget, and security. A wrong decision in the deployment model may affect the organization heavily. Hence, it is very important to know about deployment models. There are many users of the cloud, and each user has different needs. One deployment model will not suite all the cloud users. Based on the cloud setup, the properties of the cloud change. There are four types of

deployment models available in the cloud, namely, private, public, community, and hybrid. Each and every type has its own advantages and disadvantages as discussed in the succeeding sections.

4.1 Introduction

Deployment models can be defined as the different ways in which the cloud can be deployed. These models are fully user centric, that is, these depend on users' requirement and convenience. A user selects a model based on his or her requirement and needs. Basically, there are four types of deployment models in the cloud:

1. Private cloud
2. Public cloud
3. Community cloud
4. Hybrid cloud

The classification of the cloud is based on several parameters such as the size of the cloud (number of resources), type of service provider, location, type of users, security, and other issues. The smallest in size is the private cloud (Figure 4.1).

The private cloud is the most basic deployment model that can be deployed by a single organization for its personal use. It is not shared by other organizations, and it is not allowed for public use. The private cloud is to serve the people of an organization. It is usually on premise but can be outsourced also. The next one is the community cloud, which is an extension of the private cloud. Here, the cloud is the same as the private cloud but is shared by several organizations. The community cloud is established for a common cause.

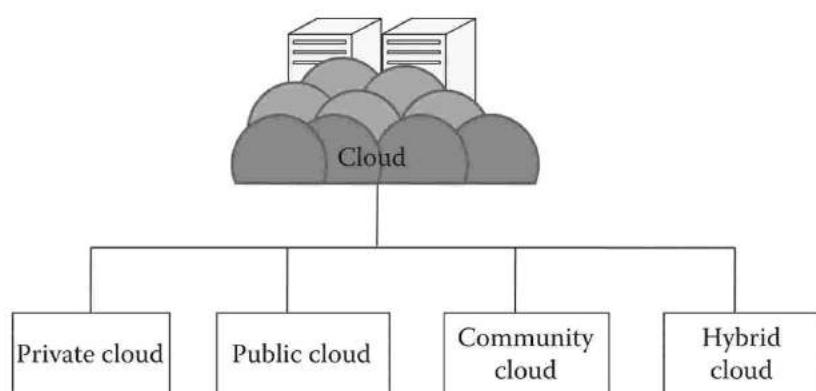


FIGURE 4.1
Cloud deployment models.

The cause can be anything, but usually it leads to mutual benefits among the participating organizations. The next is the public cloud, which is the opposite of the private cloud. This cloud allows access from any place in the world and is open to the public. This cloud is biggest in size among all other deployment models. The public cloud model is one of the most popular deployment models. The public cloud service provider charges the users on an hourly basis and serve the users according to the service-level agreements (SLAs), which are discussed in the succeeding sections. The next one is the hybrid cloud, which is a combination of other deployments. Usually, it consists of the private and public clouds combined. Several properties of the private cloud are used with the properties of the public cloud. This cloud is one of the upcoming cloud models growing in the industry.

All four types of cloud deployments are discussed in detail in subsequent sections.

4.2 Private Cloud

In this section, the private cloud deployment model is discussed. According to the National Institute of Standards and Technology (NIST), private cloud can be defined as the cloud infrastructure that is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises [1].

The private cloud in simple terms is the cloud environment created for a single organization. It is usually private to the organization but can be managed by the organization or any other third party. Private cloud can be deployed using Opensource tools such as Openstack [2], Eucalyptus [3].

The private cloud is small in size as compared to other cloud models. Here, the cloud is deployed and maintained by the organizations itself.

4.2.1 Characteristics

Certain characteristics of the private cloud are as follows:

1. *Secure*: The private cloud is secure. This is because usually the private cloud is deployed and managed by the organization itself, and hence there is least chance of data being leaked out of the cloud. In the case of outsourced cloud, the service provider may view the cloud (though governed by SLAs), but there is no other risk from anybody else as all the users belong to the same organization.
2. *Central control*: The organization mostly has full control over the cloud as usually the private cloud is managed by the organization

itself. Thus, when managed by the organization itself, there is no need for the organization to rely on anybody.

3. *Weak SLAs:* Formal SLAs may or may not exist in a private cloud. But if they exist they are weak as it is between the organization and the users of the same organization. Thus, high availability and good service may or may not be available. This depends on the organization that is controlling the cloud.

4.2.2 Suitability

Suitability refers to the instances where this cloud model can be used. It also signifies the most suitable conditions and environment where this cloud model can be used, such as the following:

- The organizations or enterprises that require a separate cloud for their personal or official use.
- The organizations or enterprises that have a sufficient amount of funds as managing and maintaining a cloud is a costly affair.
- The organizations or enterprises that consider data security to be important.
- The organizations that want autonomy and complete control over the cloud.
- The organizations that have a less number of users.
- The organizations that have prebuilt infrastructure for deploying the cloud and are ready for timely maintenance of the cloud for efficient functioning.
- Special care needs to be taken and resources should be available for troubleshooting.

The private cloud platform is not suitable for the following:

- The organizations that have high user base
- The organizations that have financial constraints
- The organizations that do not have prebuilt infrastructure
- The organizations that do not have sufficient manpower to maintain and manage the cloud

According to NIST [4], the private cloud can be classified into several types based on their location and management:

- On-premise private cloud
- Outsourced private cloud

4.2.3 On-Premise Private Cloud

On-premise private cloud is a typical private cloud that is managed by a single organization. Here, the cloud is deployed in organizational premises and is connected to the organizational network. Figure 4.2 describes a private cloud (on premise).

4.2.3.1 Issues

There are several issues associated with private clouds as discussed in the following:

1. *SLA*: SLA plays a very important role in any cloud service deployment model. For any cloud to operate, there must be certain agreements between the user and the service provider. The service provider will agree upon certain terms and conditions regarding the service delivery. These terms and conditions need to be strictly followed; if not, there will be a penalty on the part of the defaulting party. If the service provider fails to provide services as per the SLA, then he has to pay a penalty to the user; this penalty can be in any form, which is termed according to the SLA. These SLAs have different effects on different cloud delivery models. Here in the private cloud, the SLAs are defined between an organization and its users, that is, mostly employees. Usually, these users have broader access rights than the general public cloud users. Similarly in the service provider's side, the service providers are able to efficiently provide the service because of the small user base and mostly efficient network.
2. *Network*: The cloud is totally dependent on the network that is laid out. The network usually consists of a high bandwidth and has a low latency.

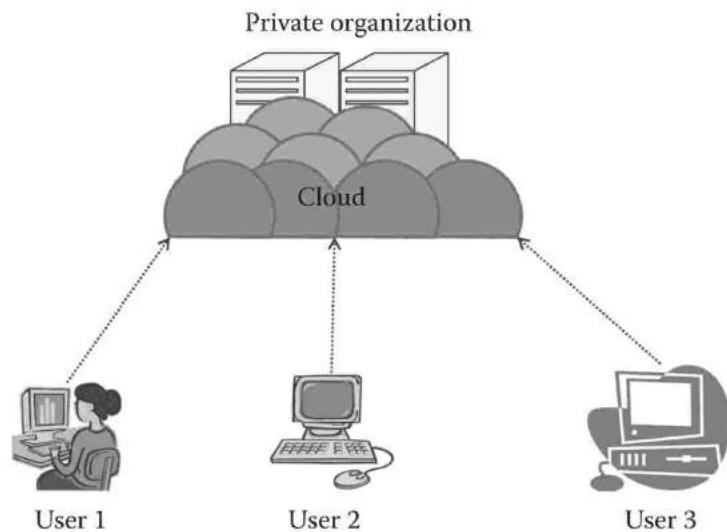


FIGURE 4.2
On-premise private cloud.

This is because the connection is only inside the organization. Network management is easier in this case, and resolving a network issue is easier.

3. *Performance:* The performance of a cloud delivery model primarily depends on the network and resources. Since here the networks are managed internally, the performance can be controlled by the network management team, and mostly this would have good performance as the number of resources is low.
4. *Security and data privacy:* Security and data privacy, though a problem with every type of service model, affect the private cloud the least. As the data of the users are solely managed by the company and most of the data would be related to the organization or company, here there is a lesser chance that the data will be leaked to people outside as there are no users outside the organization. Hence, comparatively, the private cloud is more resistant to attacks than any other cloud type purely because of the type of users and local area network. But, security breaches are possible if an internal user misuses the privileges.
5. *Location:* The private cloud does not have any problems related to the location of data being stored. In a private cloud, the data are internal and are usually stored in the same geographical location where the cloud users, that is, organization, are present (on-premise cloud). If a company has several physical locations, then the cloud is distributed over several places. In this case, there is a possibility that cloud resources have to be accessed using the Internet (by establishing a virtual private network [VPN] or without a VPN).
6. *Cloud management:* Cloud management is a broad area where the entire cloud-related tasks are managed in order to provide seamless services to the customers. This involves several tasks such as resource scheduling, resource provisioning, and resource management. The number of users, the network size, and the amount of resources are some of the important parameters that affect the management of the cloud. Here, the network is small, and the numbers of users and the amount of resources are less.
7. *Multitenancy:* The cloud basically has a multitenant architecture. As multitenant architecture supports multiple tenants with the same physical or software resource, there is a chance of unwanted access of data, and it will have less effect in the private cloud as all the issues will be intraorganizational.
8. *Maintenance:* The cloud is maintained by the organization where the cloud is deployed. The defective resources (drives and processors) are replaced with the good resources. The number of resources is less in the private cloud, so maintenance is comparatively easier.

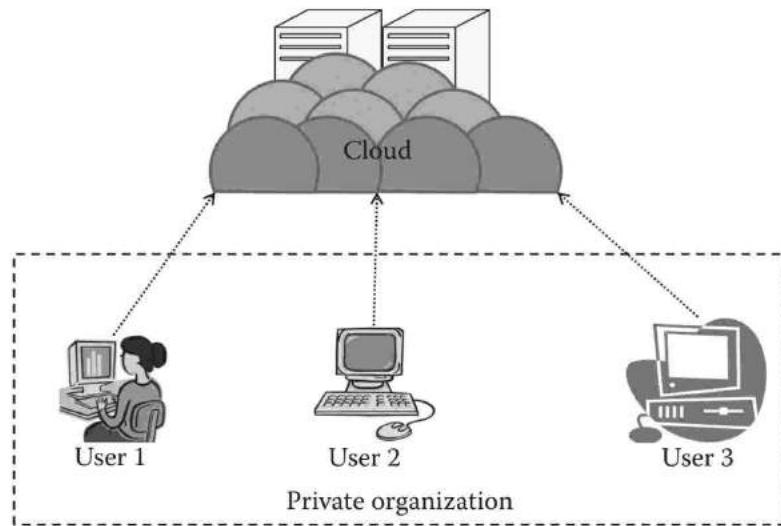


FIGURE 4.3
Outsourced private cloud.

4.2.4 Outsourced Private Cloud

The outsourced private cloud has a cloud outsourced to a third party. A third party manages the whole cloud. Everything is same as usual private cloud except that here the cloud is outsourced. There are several advantages and disadvantages of outsourcing the cloud. The following are the properties that have a significant change due to the outsourced nature of the cloud. All the other aspects are same as on-site private cloud. Figure 4.3 depicts an outsourced private cloud.

4.2.4.1 Issues

The issues that are specific to outsourced private cloud are discussed in the following:

1. *SLA*: The SLA is between the third party and the outsourcing organization. Here, the whole cloud is managed by the third party that will be usually not available on premise. The SLAs are usually followed strictly as it is a third-party organization.
2. *Network*: The cloud is fully deployed at the third-party site. The cloud's internal network is managed by a third party, and the organizations connect to the third party by means of either a dedicated connection or through the Internet. The internal network of the organization is managed by the organization, and it does not come under the purview of the SLA.
3. *Security and privacy*: Security and privacy need to be considered when the cloud is outsourced. Here, the cloud is less secure than the on-site private cloud. The privacy and security of the data mainly depend on the hosting third party as they have the control of the

cloud. But, basically the security threat is from the third party and the internal employee.

4. *Laws and conflicts:* If this cloud is deployed outside the country, then the security laws pertaining to that will apply upon the data and the data are still not fully safe. Usually, private clouds are not deployed outside, but if the off-site location is outside the country's boundary, then several problems may arise.
5. *Location:* The private cloud is usually located off site here. When there is a change of location, the data need to be transmitted through long distances. In few cases, it might be out of the country, which will lead to certain issues regarding the data and its transfer.
6. *Performance:* The performance of the cloud depends on the third party that is outsourcing the cloud.
7. *Maintenance:* The cloud is maintained by a third-party organization where the cloud is deployed. As mentioned, the defective resources (drives and processors) are replaced with the good resources. Here, again the process is less complex compared to the public cloud. The cost of maintenance is a big issue. If an organization owns a cloud, then the cost related to the cloud needs to be borne by the organization and this is usually high.

The deployment of the private cloud into a medium-sized (configuration) machine has now become an easier task. To experience a real cloud, the private cloud can be used. The minimum configuration varies for each type of platforms, but in general, a machine with an 8 GB RAM, 250 GB hard disk, and at least an i7 processor will allow the user to install a private cloud in it. Further, this private (Infrastructure-as-a-Service [IaaS]) cloud can be used to create a virtual machine, and then a user can test these virtual machines. Based on the configuration, the efficiency of the cloud varies. This deployment may not offer a full-fledged private cloud for several users but can be very useful to understand the working of a private cloud.

There are several advantages and disadvantages of a private cloud.

4.2.5 Advantages

- The cloud is small in size and is easy to maintain.
- It provides a high level of security and privacy to the user.
- It is controlled by the organization.

4.2.6 Disadvantages

- For the private cloud, budget is a constraint.
- The private clouds have loose SLAs.

4.3 Public Cloud

According to NIST, the public cloud is the cloud infrastructure that is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them [1]. It exists on the premises of the cloud provider.

The typical public cloud is depicted in Figure 4.4. Public cloud consists of users from all over the world. A user can simply purchase resources on an hourly basis and work with the resources. There is no need of any prebuilt infrastructure for using the public cloud. These resources are available in the cloud provider's premises. Usually, cloud providers accept all the requests, and hence, the resources in the service providers' end are considered *infinite* in one aspect. Some of the well-known examples of the public cloud are Amazon AWS [5], Microsoft Azure [6], etc.

4.3.1 Characteristics

1. *Highly scalable*: The public cloud is highly scalable. The resources in the public cloud are large in number and the service providers make sure that all the requests are granted. Hence, the public cloud is considered to be scalable.
2. *Affordable*: The public cloud is offered to the public on a pay-as-you-go basis; hence, the user has to pay only for what he or she is using (usually on a per-hour basis). And, this does not involve any cost related to the deployment.

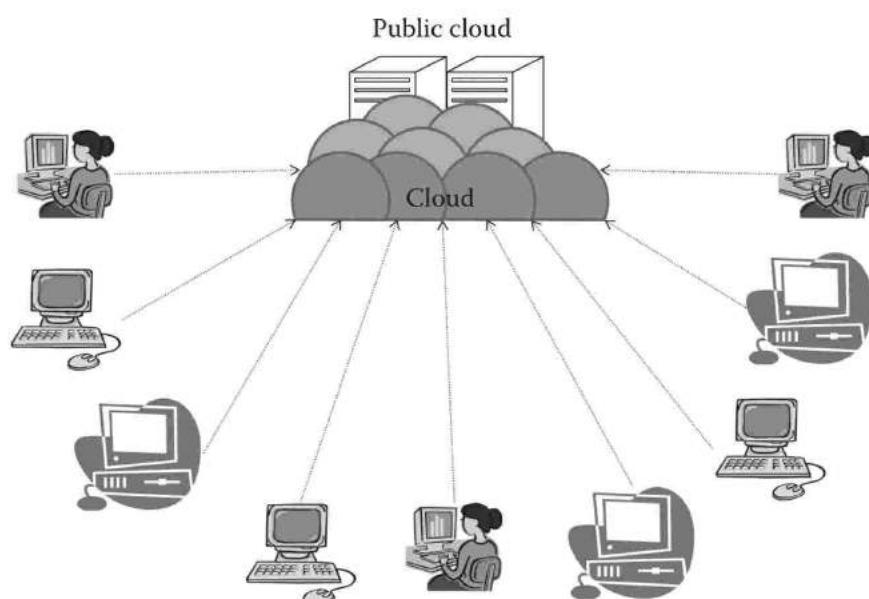


FIGURE 4.4
Public cloud.

3. *Less secure*: The public cloud is less secure out of all the four deployment models. This is because the public cloud is offered by a third party and they have full control over the cloud. Though the SLAs ensure privacy, still there is a high risk of data being leaked.
4. *Highly available*: The public cloud is highly available because anybody from any part of the world can access the public cloud with proper permission, and this is not possible in other models as geographical or other access restrictions might be there.
5. *Stringent SLAs*: SLA is very stringent in the case of the public cloud. As the service provider's business reputation and customer strength are totally dependent on the cloud services, they follow the SLA strictly and violations are avoided. These SLAs are very competitive.

4.3.2 Suitability

There are several occasions and environments where the public cloud is suitable. Thus, the suitability of the public cloud is described. The public cloud can be used whenever the following applies:

- The requirement for resources is large, that is, there is large user base.
- The requirement for resources is varying.
- There is no physical infrastructure available.
- An organization has financial constraints.

The public cloud is not suitable, where the following applies:

- Security is very important.
- Organization expects autonomy.
- Third-party reliability is not preferred.

4.3.3 Issues

Several issues pertaining to the public cloud are as follows:

1. *SLA*: Unlike the private cloud, here the number of users is more and so are the numbers of service agreements. The service provider is answerable to all the users. The users here are diverse. The SLA will cover all the users from all parts of the world. The service provider has to guarantee all the users a fair share without any priority. Having the same SLA for all users is what is usually expected, but it depends on the service provider to have the same SLA for all the users irrespective of the place they are.
2. *Network*: The network plays a major role in the public cloud. Each and every user getting the services of the cloud gets it through

the Internet. The services are accessed through the Internet by all the users, and hence, the service delivery wholly depends on the network. Unlike the private cloud where the organization takes responsibility for the network, here the service provider is not responsible for the network. The service provider is responsible for providing proper service to the customer, and once the services are given from the service provider, it goes on in transit to the user. The user will be charged for even if he or she has problem due to the network. The network usually consists of a high bandwidth and has a low latency. This is because the connection is only inside the organization. Network management is easier in this case.

3. *Performance*: As mentioned, the performance of a cloud delivery model primarily depends on the network and the resources. The service provider has to adequately manage the resources and the network. As the number of users increases, it is a challenging task for the service providers to give good performance.
4. *Multitenancy*: The resources are shared, that is, multiple users share the resources, hence the term multitenant. Due to this property, there is a high risk of data being leaked or a possible unprivileged access.
5. *Location*: The location of the public cloud is an issue. As the public cloud is fragmented and is located in different regions, the access to these clouds involves a lot of data transfers through the Internet. There are several issues related to the location. For example, a user from India might be using the public cloud and he might have to access his personal resources from other countries. This is not good as the data are being stored in some other country.
6. *Security and data privacy*: Security and data privacy are the biggest challenges in the public cloud. As data are stored in different places around the globe, data security is a very big issue. A user storing the data outside his or her country has a risk of the data being viewed by other people as that does not come under the jurisdiction of the user's country. Though this might not always be true, but it may happen.
7. *Laws and conflicts*: The data are stored in different places of the world in different countries. Hence, data centers are bound to laws of the country in which they are located. This creates many conflicts and problems for the service providers and the users.
8. *Cloud management*: Here, the number of users is more, and so the management is difficult. The jobs here are time critical, and as the number of users increases, it becomes more difficult. Inefficient management of resources will lead to resource shortage, and user service might be affected. It has a direct impact on SLA and may cause SLA violation.

9. *Maintenance*: Maintaining the whole cloud is another task. This involves continuous check of the resources, network, and other such parameters for long-lasting efficient delivery of the service. The resource provider has to continuously change the resource components from time to time. The task of maintenance is very crucial in the public cloud. The good the cloud is maintained, the better is the quality of service. Here, the cloud data center is where the maintenance happens; continuously, the disks are replaced from time to time.

The issues discussed earlier will help to understand the public cloud. Before using the public cloud, one has to choose a cloud service provider. One can choose the public cloud based on certain parameters like SLA violations, security, and cost of resources. Thus, a cloud's quality is determined by the SLA violation it does. The less the SLA violation it does, the better the cloud is. This is one way of selecting the public cloud; another way is by cost. If the job for which the resources are used is not time sensitive, then the service provider who offers the least cost is selected.

There following are several advantages and disadvantages of public clouds.

4.3.4 Advantages

- There is no need of establishing infrastructure for setting up a cloud.
- There is no need for maintaining the cloud.
- They are comparatively less costly than other cloud models.
- Strict SLAs are followed.
- There is no limit for the number of users.
- The public cloud is highly scalable.

4.3.5 Disadvantages

- Security is an issue.
- Privacy and organizational autonomy are not possible.

4.4 Community Cloud

According to NIST, the community cloud is the cloud infrastructure that is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises [1]. It is a further extension of the private cloud. Here, a private cloud is shared between

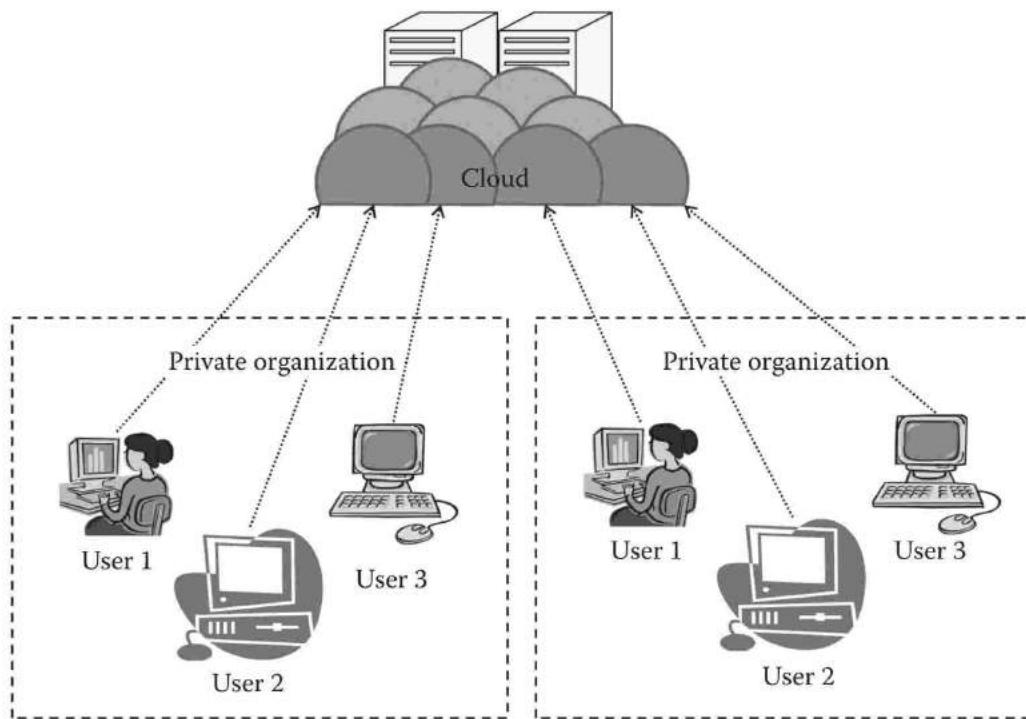


FIGURE 4.5
Community cloud.

several organizations. Either the organizations or a single organization may collectively maintain the cloud.

The main advantage of the public cloud is that the organizations are able to share the resources among themselves based on specific concerns. Thus, here the organizations are able to extract the power of the cloud, which is much bigger than the private cloud, and at the same time, they are able to use it at a usually less cost. The community is formed based on any common cause, but eventually, all the members of the community are benefitted.

This model is very suitable for organizations that cannot afford a private cloud and cannot rely on the public cloud either. Figure 4.5 describes the community cloud.

4.4.1 Characteristics

1. *Collaborative and distributive maintenance:* The community cloud is wholly collaborative, and usually no single party has full control over the whole cloud (in some cases, it may be controlled by one party). This is usually distributive, and hence, better cooperation gives better results. Even though it may be outsourced, collaboration based on purpose always proves to be beneficial.
2. *Partially secure:* Partially secure refers to the property of the community cloud where few organizations share the cloud, so there is a possibility that the data can be leaked from one organization to another, though it is safe from the outside world.

3. *Cost effective*: The community cloud is cost effective as the whole cloud is being shared by several organizations or a community. Usually, not only cost but every other sharable responsibilities are also shared or divided among the groups.

4.4.2 Suitability

This kind of cloud is suitable for organizations that

- Want to establish a private cloud but have financial constraint
- Do not want to complete maintenance responsibility of the cloud
- Want to establish the cloud in order to collaborate with other clouds
- Want to have a collaborative cloud with more security features than the public cloud

This cloud is not suitable for organizations that

- Prefer autonomy and control over the cloud
- Does not want to collaborate with other organizations

There are two types of community cloud deployments:

1. On-premise community cloud
2. Outsourced community cloud

4.4.3 On-Premise Community Cloud

On-premise community cloud consists of the cloud deployed within the premises and is maintained by the organizations themselves.

4.4.3.1 Issues

The issues related to on-site community cloud are as follows:

1. *SLA*: Here, SLA is a little more stringent than the private cloud but is less stringent than the public cloud. As more than one organization is involved, SLA has to be there to have a fair play among the users of the cloud and among the organizations themselves.
2. *Network*: The private cloud can be there in any location as this cloud is being shared by more than one organization. Here, each organization will have a separate network, and they will connect to the cloud. It is the responsibility of each organization to take care of their own network. The service provider is not responsible

for the network issues in the organization. The network is not big and complex as in the public cloud.

3. *Performance*: In this type of deployment, more than one organization coordinate together and provide the cloud service. Thus, it is on the maintenance and management team that the performance depends.
4. *Multitenancy*: There is a moderate risk due to multitenancy. As this cloud is meant for several organizations, the unprivileged access into interorganizational data may lead to several problems.
5. *Location*: The location of the cloud is very important in this case. Usually, the cloud is deployed at any one of the organizations or is maintained off site by any third party. In either case, the organizations have to access the cloud from another location.
6. *Security and privacy*: Security and privacy are issues in the community cloud since several organizations are involved in it. The privacy between the organizations needs to be maintained. As the data are collectively stored, the situation is more like that of a public cloud with less users. The organizations should have complete trust on the service provider, and as all other cloud models, this becomes the bottleneck.
7. *Laws and conflicts*: This applies if organizations are located in different countries. If the organizations are located in the same country, then there is no issue, but if these organizations are located elsewhere, that is, in different countries, then they have to abide by the rules of the country in which the cloud infrastructure is present, thus making the process a bit more complex.
8. *Cloud management*: Cloud management is done by the service provider, here in this case by the organizations collectively. The organizations will have a management team specifically for this cloud and that is responsible for all the cloud management-related operations.
9. *Cloud maintenance*: Cloud maintenance is done by the organizations collectively. The maintenance team collectively maintains all the resources. It is responsible for continuous replacement of resources. In the community cloud, the number of resources is less than the public cloud but usually more than the private cloud.

4.4.4 Outsourced Community Cloud

In the outsourced community cloud, the cloud is outsourced to a third party. The third party is responsible for maintenance and management of the cloud.

4.4.4.1 Issues

The following are some aspects in the community cloud that changed because of the outsourced nature of the community cloud:

1. *SLA*: The SLA is between the group of organizations and the service provider. The SLA here is stringent as it involves a third party. The SLA here is aimed at a fair share of resources among the organizations. The service provider is not responsible for the technical problems within the organization.
2. *Network*: The issues related to the network are same as the on-site community cloud, but here the service provider is outsourced and hence organizations are responsible for their own network and the service provider is responsible for the cloud network.
3. *Performance*: The performance totally depends on the outsourced service provider. The service provider is responsible for efficient services, except for the network issue in the client side.
4. *Security and privacy*: As discussed earlier, there are security and privacy issues as several organizations are involved in it, but in addition to that, the involvement of a third party as a service provider will create much more issues as the organizations have to completely rely on the third party.
5. *Laws and conflicts*: In addition to the issues related to laws due to organizations' location, there is a major issue associated with the location of the cloud service provider. If the service provider is outside the country, then there is conflict related to data laws in that country.
6. *Cloud management and maintenance*: Cloud management and maintenance are done by the service provider. The complexity of managing and maintenance increases with the number of organizations in the community. But, this is less complex than the public cloud.
7. The community cloud as said is an extension of the private cloud. The issues discussed earlier would be more or less the same as the issues related to the private cloud with a very few differences. The community cloud would prove to be successful if a group of organizations work cooperatively.

The following describes the several advantages and disadvantages of the community cloud.

4.4.5 Advantages

- It allows establishing a low-cost private cloud.
- It allows collaborative work on the cloud.

- It allows sharing of responsibilities among the organization.
- It has better security than the public cloud.

4.4.6 Disadvantages

- Autonomy of an organization is lost.
- Security features are not as good as the private cloud.
- It is not suitable if there is no collaboration.

4.5 Hybrid Cloud

According to NIST, the hybrid cloud can be defined as the cloud infrastructure that is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability [1].

The hybrid cloud usually is a combination of both public and private clouds. This is aimed at combining the advantages of private and public clouds. The usual method of using the hybrid cloud is to have a private cloud initially, and then for additional resources, the public cloud is used. There are several advantages of the hybrid cloud. The hybrid cloud can be regarded as a private cloud extended to the public cloud. This aims at utilizing the power of the public cloud by retaining the properties of the private cloud. One of the popular examples for the hybrid cloud is Eucalyptus [7]. Eucalyptus was initially designed for the private cloud and is basically a private cloud, but now it also supports hybrid cloud. Figure 4.6 shows the hybrid cloud. The hybrid cloud can be further extended into a vast area of federated clouds that is discussed in subsequent chapters.

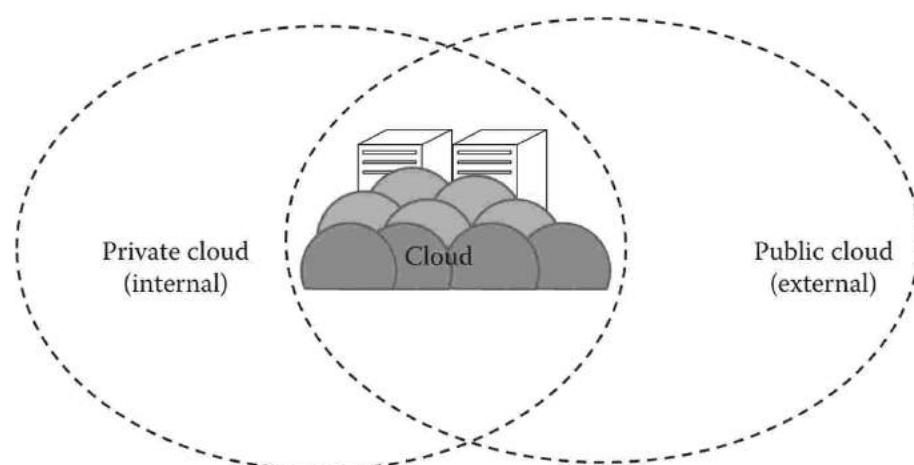


FIGURE 4.6
Hybrid cloud.

4.5.1 Characteristics

1. *Scalable*: The hybrid cloud is a combination of one or more deployment models. Usually, the private with public cloud gives hybrid cloud. The main reason of having a hybrid cloud is to use the property of a public cloud with a private cloud environment. The public cloud is used whenever needed; hence, as the public cloud is scalable, the hybrid cloud with the help of its public counterpart is also scalable.
2. *Partially secure*: The hybrid cloud usually is a combination of public and private. The private cloud is considered to be secured, but as the hybrid cloud also uses the public cloud, there is high risk of security breach. Thus, it cannot be fully termed as secure but as partially secure.
3. *Stringent SLAs*: As the hybrid cloud involved a public cloud intervention, the SLAs are stringent and might as per the public cloud service provider. But overall, the SLAs are more stringent than the private cloud.
4. *Complex cloud management*: Cloud management is complex and is a difficult task in the hybrid cloud as it involves more than one type of deployment models and also the numbers of users are high.

4.5.2 Suitability

The hybrid cloud environment is suitable for

- Organizations that want the private cloud environment with the scalability of the public cloud
- Organizations that require more security than the public cloud

The hybrid cloud is not suitable for

- Organizations that consider security as a prime objective
- Organizations that will not be able to handle hybrid cloud management

4.5.3 Issues

The cloud can be analyzed in the following aspects:

1. *SLA*: SLA is one of the important aspects of the hybrid cloud as both private and public are involved. There is a right combination of SLAs between the clouds. The private cloud does not have stringent agreements, whereas the public cloud has certain strict rules to be covered. The SLAs to be covered under each purview are clearly defined, and it wholly depends on the service provider (private cloud) to provide efficient services to the customers.

2. *Network*: The network is usually a private network, and whenever there is a necessity, the public cloud is used through the Internet. Unlike the public cloud, here there is a private network also. Thus, a considerable amount of effort is required to maintain the network. The organization takes the responsibility from the network.
3. *Performance*: The hybrid cloud is a special type of cloud in which the private environment is maintained with access to the public cloud whenever required. Thus, here again a feel of an infinite resource is restored. The cloud provider (private cloud) is responsible for providing the cloud.
4. *Multitenancy*: Multitenancy is an issue in the hybrid cloud as it involves the public cloud in addition to the private cloud. Thus, this property can be misused and the breaches will have adverse affects as some parts of the cloud go public.
5. *Location*: Like a private cloud, the location of these clouds can be on premise or off premise and they can be outsourced. They will have all the issues related to the private cloud; in addition to that, issues related to the public cloud will also come into picture whenever there is intermittent access to the public cloud.
6. *Security and privacy*: Whenever the user is provided services using the public cloud, security and privacy become more stringent. As it is the public cloud, the threat of data being lost is high.
7. *Laws and conflicts*: Several laws of other countries come under the purview as the public cloud is involved, and usually these public clouds are situated outside the country's boundaries.
8. *Cloud management*: Here, everything is managed by the private cloud service provider.
9. *Cloud maintenance*: Cloud maintenance is of the same complexity as the private cloud; here, only the resources under the purview of the private cloud need to be maintained. It involves a high cost of maintenance.

The hybrid cloud is one of the fastest growing deployment models, which is now being discussed because of its characteristics as discussed earlier. The issues discussed provide an overview about the difference between the other cloud models and the hybrid cloud model. There is another part of the cloud called as federated cloud that is described in the subsequent chapter.

There are several advantages and disadvantages of the hybrid cloud.

4.5.4 Advantages

- It gives the power of both the private and public clouds.
- It is highly scalable.
- It provides better security than the public cloud.

4.5.5 Disadvantages

- The security features are not as good as the public cloud.
 - Managing a hybrid cloud is complex.
 - It has stringent SLAs.
-

4.6 Summary

Cloud computing forms the base for many things in today's world. To start with, the deployment models form the base and need to be known before starting with other aspects of the cloud. These deployment models are based on several properties such as size, location, and complexity. There are four types of deployment models discussed in this chapter. The description of each deployment model with its characteristic and its suitability to different kinds of needs is provided. Each type of deployment model has its own significance. Each deployment model is used in one or other aspects. These deployment models are very important and usually have a great impact on the businesses that are dependent on the cloud. A smart choice of deployment model always proves to be beneficial, avoiding heavy losses. Hence, high importance is given to deployment models.

Review Points

- *Deployment models*: Deployment models can be defined as the different ways in which the cloud can be deployed (see Section 4.1).
- *Private cloud*: Private cloud is the cloud environment created for a single organization (see Section 4.2).
- *Public cloud*: Public cloud is the cloud infrastructure that is provisioned for open use by the general public (see Section 4.3).
- *Hybrid cloud*: Hybrid cloud can be defined as the cloud infrastructure that is a composition of two or more distinct cloud infrastructures (see Section 4.5).
- *Community cloud*: Community cloud is the cloud infrastructure that is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (see Section 4.4).
- *SLA*: SLAs are terms and conditions that are negotiated between the service provider and the user (See Section 4.2.3.1).
- *Multitenancy*: Multitenancy is a property of cloud in which multiple users share the same software resource as tenants (see Section 4.2.3.1).

Review Questions

1. Compare and contrast public and private clouds.
 2. What is SLA? Are SLAs different for each type of cloud deployment?
 3. Analyze the cloud deployment models based on security.
 4. How do laws of different countries affect the public cloud model?
 5. Differentiate community cloud and hybrid cloud based on their properties.
 6. Public cloud is less secure. Justify.
 7. What is outsourced community cloud?
 8. What are the characteristics of hybrid cloud?
 9. What are the advantages of using the community cloud?
-

References

1. Mell, P. and T. Grance The NIST definition of cloud computing (draft). NIST Special Publication 800.145: 7, 2011.
2. Openstack. Available [Online]: <http://www.openstack.org>. Accessed April 7, 2014.
3. Eucalyptus: An open source private cloud. Available [Online]: <https://www.eucalyptus.com/eucalyptus-cloud/iaas>. Accessed April 5, 2014.
4. Badger, L. et al. Cloud computing synopsis and recommendations. NIST Special Publication 800: 146, 2012.
5. Amazon EC2. Available [Online]: <http://aws.amazon.com/ec2/>. Accessed April 16, 2014.
6. Microsoft Azure. Available [Online]: <http://www.azure.microsoft.com/en-us/>. Accessed March 20, 2014.
7. Hybrid cloud simplified. Available [Online]: <https://www.eucalyptus.com/>. Accessed March 12, 2014.

5

Cloud Service Models

Learning Objectives

The main objective of this chapter is to introduce the different service delivery models of cloud computing. After reading this chapter, you will

- Understand the basics of cloud computing stack and cloud service models
- Understand how the Infrastructure as a Service (IaaS) changes computing
- Understand how the Platform as a Service (PaaS) changes the application developer
- Understand how the Software as a Service (SaaS) changes the application delivery
- Understand the characteristics, suitability, and pros and cons of IaaS, PaaS, and SaaS
- Understand the other cloud service models such as Network as a Service (NaaS) and Storage as a Service (STaaS)

Preamble

Cloud computing provides computing resources, development platforms, and applications as a service to the end users. The information technology (IT) industries started subscribing the cloud services instead of buying the products. This chapter gives an insight into the three basic service models of cloud computing, namely, IaaS, PaaS, and SaaS. According to the services provided and subscribed, the responsibilities of the end user and service may vary. This chapter also discusses the responsibilities of the end user and service providers of IaaS, PaaS, and SaaS. Characteristics, suitability, and pros and cons of different cloud service models are also discussed in this

chapter along with the summary of popular service providers. At the end, this chapter gives a brief idea about other cloud service models such as NaaS, StaaS, Database as a Service (DBaaS), Security as a Service (SECaS), and Identity as a Service (IDaaS).

5.1 Introduction

Cloud computing is a model that enables the end users to access the shared pool of resources such as compute, network, storage, database, and application as an on-demand service without the need to buy or own it. The services are provided and managed by the service provider, reducing the management effort from the end user side. The essential characteristics of the cloud include on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. The National Institute of Standards and Technology (NIST) defines three basic service models, namely, IaaS, PaaS, and SaaS, as shown in Figure 5.1.

The NIST definition of the three basic service models is given as follows:

1. *IaaS*: The ability given to the infrastructure architects to deploy or run any software on the computing resources provided by the service provider. Here, the underlying infrastructures such as compute, network, and storage are managed by the service provider. Thus, the infrastructure architects are exempted from maintaining the data center or underlying infrastructure. The end users are responsible for managing applications that are running on top of the service

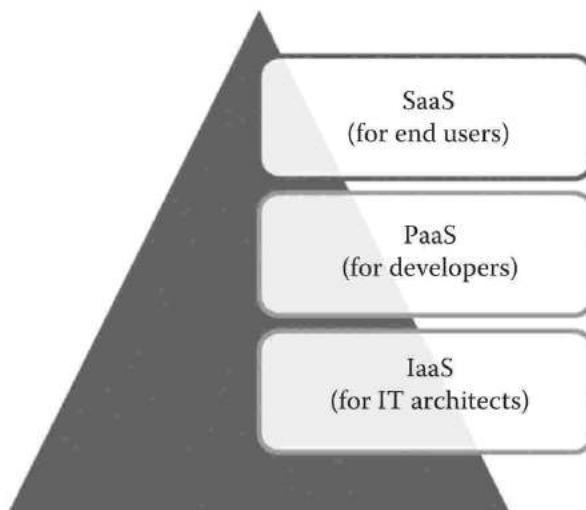


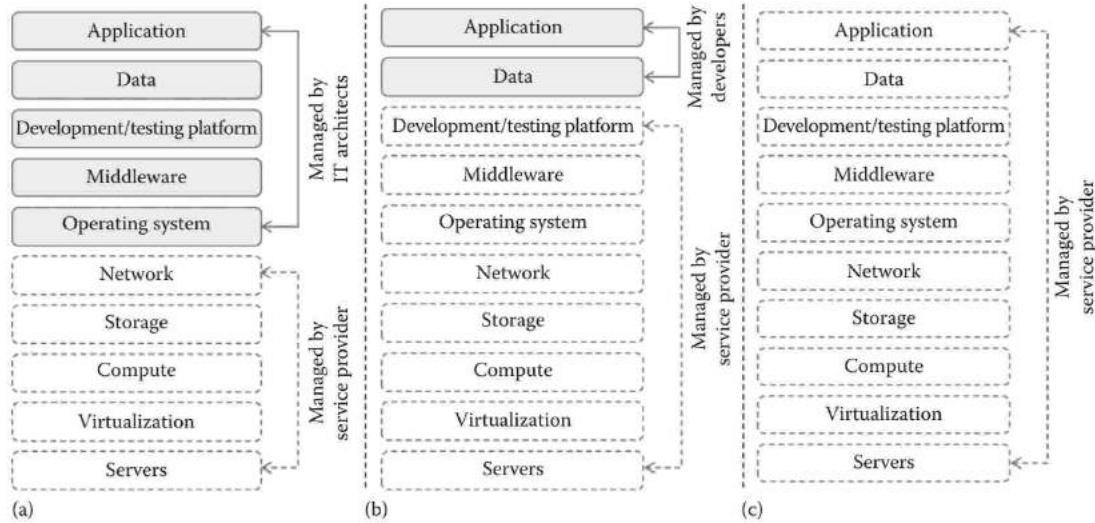
FIGURE 5.1
Basic cloud service models.

provider cloud infrastructure. Generally, the IaaS services are provided from the service provider cloud data center. The end users can access the services from their devices through web command line interface (CLI) or application programming interfaces (APIs) provided by the service providers. Some of the popular IaaS providers include Amazon Web Services (AWS), Google Compute Engine, OpenStack, and Eucalyptus.

2. *PaaS*: The ability given to developers to develop and deploy an application on the development platform provided by the service provider. Thus, the developers are exempted from managing the development platform and underlying infrastructure. Here, the developers are responsible for managing the deployed application and configuring the development environment. Generally, PaaS services are provided by the service provider on an on-premise or dedicated or hosted cloud infrastructure. The developers can access the development platform over the Internet through web CLI, web user interface (UI), and integrated development environments (IDEs). Some of the popular PaaS providers include Google App Engine, Force.com, Red Hat OpenShift, Heroku, and Engine Yard.
3. *SaaS*: The ability given to the end users to access an application over the Internet that is hosted and managed by the service provider. Thus, the end users are exempted from managing or controlling an application, the development platform, and the underlying infrastructure. Generally, SaaS services are hosted in service provider-managed or service provider-hosted cloud infrastructure. The end users can access the services from any thin clients or web browsers. Some of the popular SaaS providers include Salesforce.com, Google Apps, and Microsoft office 365.

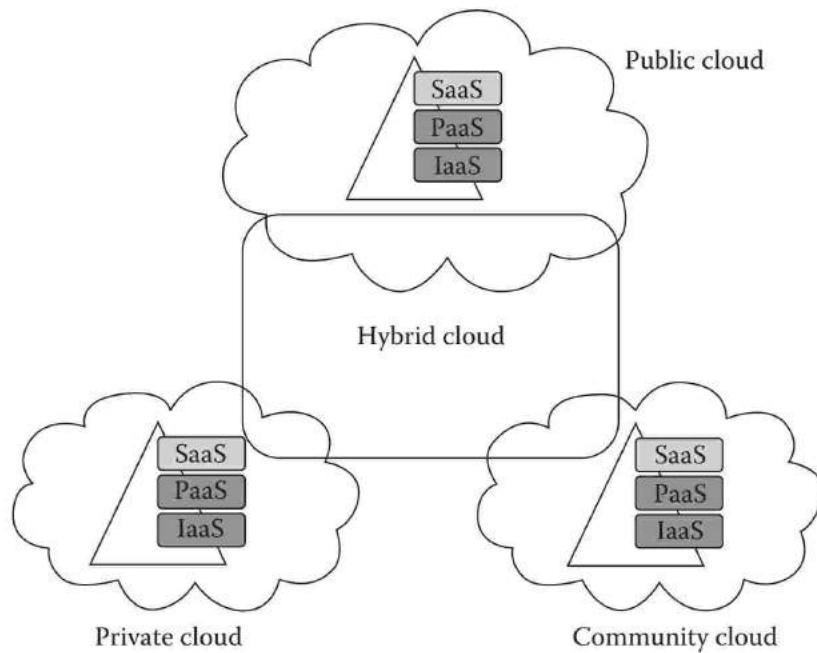
The different cloud service models target different audiences. For example, the IaaS model targets the information technology (IT) architects, PaaS targets the developers, and SaaS targets the end users. Based on the services subscribed, the responsibility of the targeted audience may vary as shown in Figure 5.2.

In IaaS, the end users are responsible for maintaining the development platform and the application running on top of the underlying infrastructure. The IaaS providers are responsible for maintaining the underlying hardware as shown in Figure 5.2a. In PaaS, the end users are responsible for managing the application that they have developed. The underlying infrastructure will be maintained by the infrastructure provider as shown in Figure 5.2b. In SaaS, the end user is free from maintaining the infrastructure, development platform, and application that they are using. All the maintenance will be carried out by the SaaS providers as shown Figure 5.2c.

**FIGURE 5.2**

User and service provider responsibilities of cloud service models: (a) IaaS, (b) PaaS, and (c) SaaS.

The different service models of cloud computing can be deployed and delivered through any one of the cloud deployment models. The NIST defines four different types of cloud deployment models, namely, public cloud, private cloud, community cloud, and hybrid cloud. The public cloud is provided for the general public. The private cloud is used by an organization for its multiple business units. The community cloud is for some group of organization with the same goals. The hybrid cloud is any combination of the public, private, and community clouds. The service delivery of cloud services through different deployment models is shown in Figure 5.3.

**FIGURE 5.3**

Deployment and delivery of different cloud service delivery models.

This chapter discusses about the characteristics, suitability, and pros and cons of different cloud service models. Additionally, this chapter gives the summary of popular IaaS, PaaS, and SaaS providers.

5.2 Infrastructure as a Service

IaaS changes the way that the compute, storage, and networking resources are consumed. In traditional data centers, the computing power is consumed by having physical access to the infrastructure. IaaS changes the computing from a physical infrastructure to a virtual infrastructure. IaaS provides virtual computing, storage, and network resources by abstracting the physical resources. Technology *virtualization* is used to provide the virtual resources. All the virtual resources are given to the virtual machines (VMs) that are configured by the service provider. The end users or IT architects will use the infrastructure resources in the form of VMs as shown in Figure 5.4.

The targeted audience of IaaS is the IT architect. The IT architect can design virtual infrastructure, network, load balancers, etc., based on their needs. The IT architects need not maintain the physical servers as it is

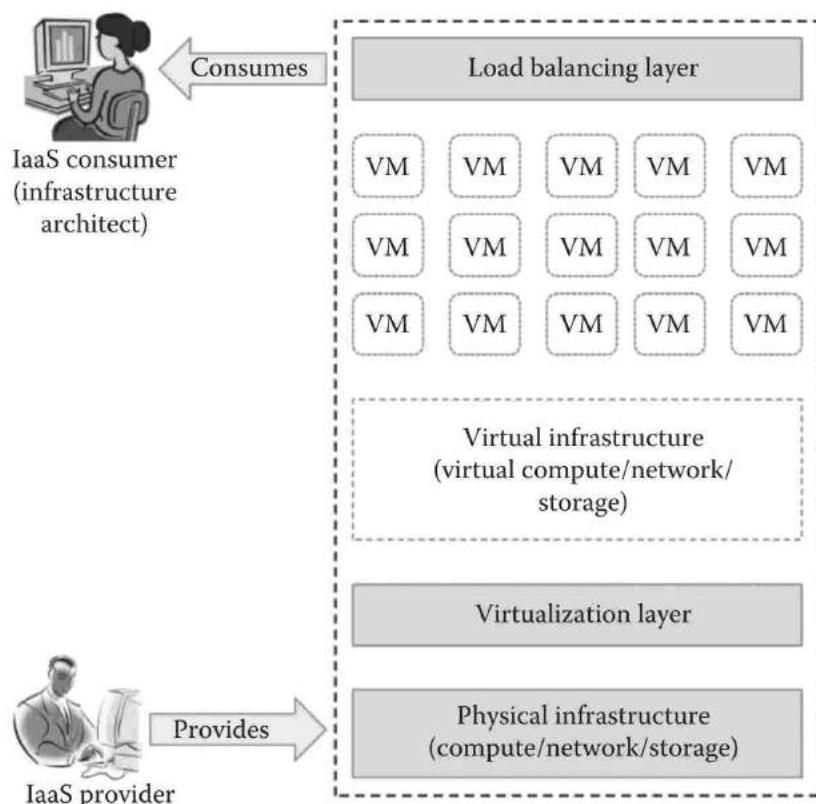
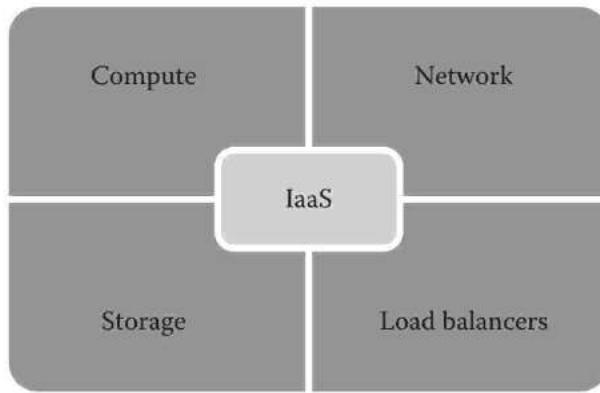


FIGURE 5.4
Overview of IaaS.

**FIGURE 5.5**

Services provided by IaaS providers.

maintained by the service providers. The physical infrastructure can be maintained by the service providers themselves. Thus, it eliminates or hides the complexity of maintaining the physical infrastructure from the IT architects. A typical IaaS provider may provide the flowing services as shown in Figure 5.5:

1. *Compute*: Computing as a Service includes virtual central processing units (CPUs) and virtual main memory for the VMs that are provisioned to the end users.
2. *Storage*: STaaS provides back-end storage for the VM images. Some of the IaaS providers also provide the back end for storing files.
3. *Network*: Network as a Service (NaaS) provides virtual networking components such as virtual router, switch, and bridge for the VMs.
4. *Load balancers*: Load Balancing as a Service may provide load balancing capability at the infrastructure layer.

5.2.1 Characteristics of IaaS

IaaS providers offer virtual computing resources to the consumers on a pay-as-you-go basis. IaaS contains the characteristics of cloud computing such as on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. Apart from all these, IaaS has its own unique characteristics as follows:

1. *Web access to the resources*: The IaaS model enables the IT users to access infrastructure resources over the Internet. When accessing a huge computing power, the IT user need not get physical access to the servers. Through any web browsers or management console, the users can access the required infrastructure.

2. *Centralized management*: Even though the physical resources are distributed, the management will be from a single place. The resources distributed across different parts can be controlled from any management console. This ensures effective resource management and effective resource utilization.
3. *Elasticity and dynamic scaling*: IaaS provides elastic services where the usage of resources can be increased or decreased according to the requirements. The infrastructure need depends on the load on the application. According to the load, IaaS services can provide the resources. The load on any application is dynamic and IaaS services are capable of proving the required services dynamically.
4. *Shared infrastructure*: IaaS follows a one-to-many delivery model and allows multiple IT users to share the same physical infrastructure. The different IT users will be given different VMs. IaaS ensures high resource utilization.
5. *Preconfigured VMs*: IaaS providers offer preconfigured VMs with operating systems (OSs), network configuration, etc. The IT users can select any kind of VMs of their choice. The IT users are free to configure VMs from scratch. The users can directly start using the VMs as soon as they subscribed to the services.
6. *Metered services*: IaaS allows the IT users to rent the computing resources instead of buying it. The services consumed by the IT user will be measured, and the users will be charged by the IaaS providers based on the amount of usage.

5.2.2 Suitability of IaaS

IaaS reduces the total cost of ownership (TCO) and increases the return on investment (ROI) for start-up companies that cannot invest more in buying infrastructure.

IaaS can be used in the following situations:

1. *Unpredictable spikes in usage*: When there is a significant spike in usage of computing resources, IaaS is the best option for IT industries. When demand is very volatile, we cannot predict the spikes and troughs in terms of demand of the infrastructure. In this situation, we cannot add or remove infrastructure immediately according to the demand in a traditional infrastructure. If there is an unpredictable demand of infrastructure, then it is recommended to use IaaS services.
2. *Limited capital investment*: New start-up companies cannot invest more on buying infrastructure for their business needs. And so by using IaaS, start-up companies can reduce the capital investment on

hardware. IaaS is the suitable option for start-up companies with less capital investment on hardware.

3. *Infrastructure on demand:* Some organizations may require large infrastructure for a short period of time. For this purpose, an organization cannot afford to buy more on-premise resources. Instead, they can rent the required infrastructure for a specific period of time. IaaS best suits the organizations that look for infrastructure on demand or for a short time period.

IaaS helps start-up companies limit its capital expenditure. While it is widely used by start-up companies, there are some situations where IaaS may not be the best option. In following situations, IT users should avoid using the IaaS:

1. *When regulatory compliance does not allow off-premise hosting:* For some companies, its regulation may not allow the application and data to be hosted on third-party off-premise infrastructure.
2. *When usage is minimal:* When the usage is minimal and the available on-premise infrastructure itself is capable of satisfying their needs.
3. *When better performance is required:* Since the IaaS services are accessed through the Internet, sometimes the performance might be not as expected due to network latency.
4. *When there is a need for more control on physical infrastructure:* Some organizations might require physical control over the underlying infrastructure. As the IaaS services are abstracted as virtual resources, it is not possible to have more control on underlying physical infrastructure.

5.2.3 Pros and Cons of IaaS

Being one of the important service models of cloud computing, IaaS provides lot of benefits to the IT users. The following are the benefits provided by IaaS:

1. *Pay-as-you-use model:* The IaaS services are provided to the customers on a pay-per-use basis. This ensures that the customers are required to pay for what they have used. This model eliminates the unnecessary spending on buying hardware.
2. *Reduced TCO:* Since IaaS providers allow the IT users to rent the computing resources, they need not buy physical hardware for running their business. The IT users can rent the IT infrastructure rather than buy it by spending large amount. IaaS reduces the need for buying hardware resources and thus reduces the TCO.

3. *Elastic resources:* IaaS provides resources based on the current needs. IT users can scale up or scale down the resources whenever they want. This dynamic scaling is done automatically using some load balancers. This load balancer transfers the additional resource request to the new server and improves application efficiency.
4. *Better resource utilization:* Resource utilization is the most important criteria to succeed in the IT business. The purchased infrastructure should be utilized properly to increase the ROI. IaaS ensures better resource utilization and provides high ROI for IaaS providers.
5. *Supports Green IT:* In traditional IT infrastructure, dedicated servers are used for different business needs. Since many servers are used, the power consumption will be high. This does not result in Green IT. In IaaS, the need of buying dedicated servers is eliminated as single infrastructure is shared between multiple customers, thus reducing the number of servers to be purchased and hence the power consumption that results in Green IT.

Even though IaaS provides cost-related benefits to small-scale industries, it lacks in providing security to the data. The following are the drawbacks of IaaS:

1. *Security issues:* Since IaaS uses virtualization as the enabling technology, hypervisors play an important role. There are many attacks that target the hypervisors to compromise it. If hypervisors get compromised, then any VMs can be attacked easily. Most of the IaaS providers are not able to provide 100% security to the VMs and the data stored on the VMs.
2. *Interoperability issues:* There are no common standards followed among the different IaaS providers. It is very difficult to migrate any VM from one IaaS provider to the other. Sometimes, the customers might face the vendor lock-in problem.
3. *Performance issues:* IaaS is nothing but the consolidation of available resources from the distributed cloud servers. Here, all the distributed servers are connected over the network. Latency of the network plays an important role in deciding the performance. Because of latency issues, sometimes the VM contains issues with its performance.

5.2.4 Summary of IaaS Providers

There are many public and private IaaS providers in the market who provides infrastructure services to the end users. Table 5.1 provides the summary of popular infrastructure providers.

In the table, the popular IaaS providers are classified based on the license, deployment model, and supported host OS, guest OS, and hypervisors. The end user may choose any IaaS provider that matches their needs. Generally,

TABLE 5.1
Summary of Popular IaaS Providers

Provider	License	Deployment Model	Host OS	Guest OS	Supported Hypervisor(s)
Amazon Web Services	Proprietary	Public	Not available	Red Hat Linux, Windows Server, SuSE Linux, Ubuntu, Fedora, Debian, CentOS, Gentoo Linux, Oracle Linux, and FreeBSD	Xen
Google Compute Engine	Proprietary	Public	Not available	Debian 7 Wheezy, CentOS 6, Red Hat Enterprise Linux, SUSE, Windows Server, CoreOS, FreeBSD, and SELinux	KVM
Microsoft Windows Azure	Proprietary	Public	Not available	Windows Server, CentOS, FreeBSD, openSUSE Linux, and Oracle Enterprise Linux	Windows Azure hypervisor
Eucalyptus	GPLv3	Private and hybrid	Linux	Linux and Windows	Xen, KVM, VMware
Apache CloudStack	Apache 2	Private	Linux	Windows, Linux, and various versions of BSD	KVM, vSphere, XenServer / XCP
OpenNebula	Apache 2	Private, public, and hybrid	CentOS, Debian, and openSUSE	Microsoft Windows and Linux	Xen, KVM, VMware
OpenStack	Apache 2	Private and public	CentOS, Debian, Fedora, RHEL, openSUSE, and Ubuntu	CentOS, Ubuntu, Microsoft Windows, and FreeBSD	libvirt, Hyper-V, VMware, XenServer 6.2, baremetal, docker, Xen, LXC via libvirt

public IaaS consumers need not consider the host OS as it is maintained by the service provider. In managing the private cloud, the users should see the supported host OS. However, most of the private IaaS supports popular guest OS, fully depending on the hypervisor that the IaaS providers are supporting.

5.3 Platform as a Service

PaaS changes the way that the software is developed and deployed. In traditional application development, the application will be developed locally and will be hosted in the central location. In stand-alone application development, the applications will be developed and delivered as executables. Most of the applications developed by traditional development platforms result in a licensing-based software, whereas PaaS changes the application development from local machine to online. PaaS providers provide the development PaaS from the data center. The developers can consume the services over the Internet as shown in Figure 5.6.

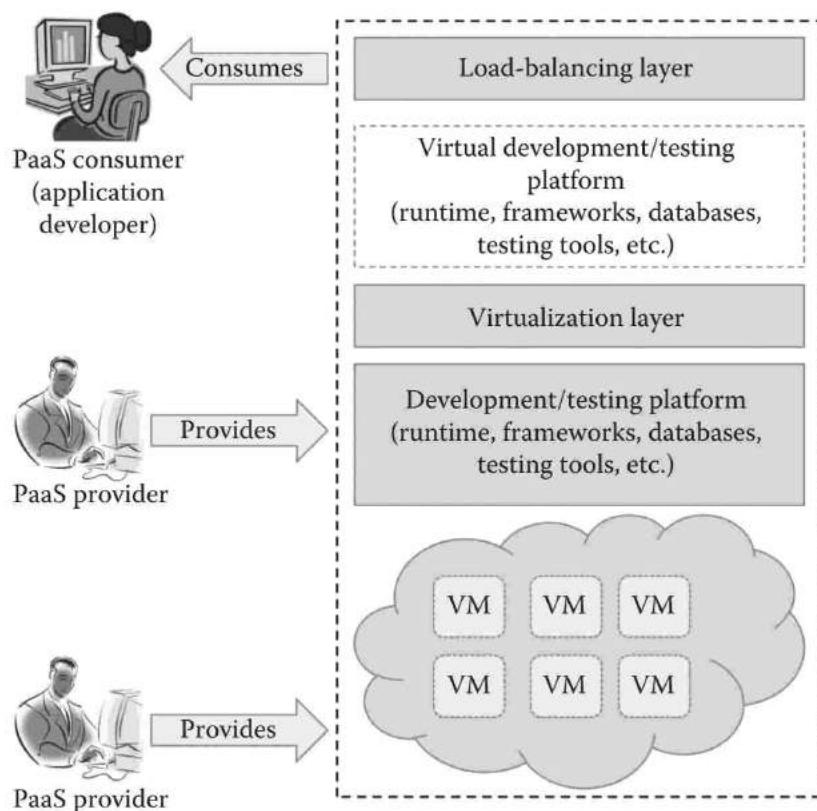


FIGURE 5.6
Overview of PaaS.

PaaS allows the developers to develop their application online and also allows them to deploy immediately on the same platform. PaaS consumers or developers can consume language runtimes, application frameworks, databases, message queues, testing tools, and deployment tools as a service over the Internet. Thus, it reduces the complexity of buying and maintaining different tools for developing an application. Typical PaaS providers may provide programming languages, application frameworks, databases, and testing tools as shown in Figure 5.7. Some of the PaaS providers also provide build tools, deployment tools, and software load balancers as a service:

1. *Programming languages*: PaaS providers provide a wide variety of programming languages for the developers to develop applications. Some of the popular programming languages provided by PaaS vendors are Java, Perl, PHP, Python, Ruby, Scala, Clojure, and Go.
2. *Application frameworks*: PaaS vendors provide application frameworks that simplify the application development. Some of the popular application development frameworks provided by a PaaS provider include Node.js, Rails, Drupal, Joomla, WordPress, Django, EE6, Spring, Play, Sinatra, Rack, and Zend.
3. *Database*: Since every application needs to communicate with the databases, it becomes a must-have tool for every application. PaaS providers are providing databases also with their PaaS platforms. The popular databases provided by the popular PaaS vendors are ClearDB, PostgreSQL, Cloudant, Membase, MongoDB, and Redis.
4. *Other tools*: PaaS providers provide all the tools that are required to develop, test, and deploy an application.

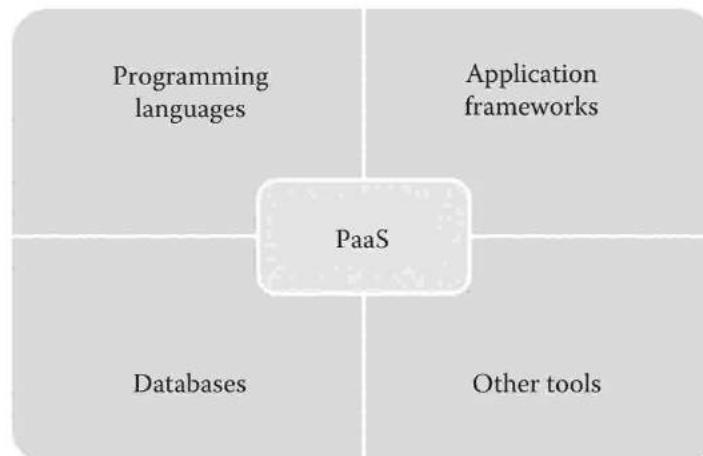


FIGURE 5.7

Services provided by PaaS providers.

5.3.1 Characteristics of PaaS

PaaS development platforms are different from the traditional application development platforms. The following are the essential characteristics that make PaaS unique from traditional development platforms:

1. *All in one*: Most of the PaaS providers offer services to develop, test, deploy, host, and maintain applications in the same IDE. Additionally, many service providers provide all the programming languages, frameworks, databases, and other development-related services that make developers choose from a wide variety of development platforms.
2. *Web access to the development platform*: A typical development platform uses any IDEs for developing applications. Typically, the IDE will be installed in the developer's machines. But, PaaS provides web access to the development platform. Using web UI, any developer can get access to the development platform. The web-based UI helps the developers create, modify, test, and deploy different applications on the same platform.
3. *Offline access*: A developer may not be able to connect to the Internet for a whole day to access the PaaS services. When there is no Internet connectivity, the developers should be allowed to work offline. To enable offline development, some of the PaaS providers allow the developer to synchronize their local IDE with the PaaS services. The developers can develop an application locally and deploy it online whenever they are connected to the Internet.
4. *Built-in scalability*: Scalability is an important requirement for the new-generation web or SaaS applications. It is very difficult to enable the dynamic scalability for any application developed using traditional development platforms. But, PaaS services provide built-in scalability to an application that is developed using any particular PaaS. This ensures that the application is capable of handling varying loads efficiently.
5. *Collaborative platform*: Nowadays, the development team consists of developers who are working from different places. There is a need for a common platform where the developers can collaboratively work together on the same project. Most of the PaaS services provide support for collaborative development. To enable collaboration among developers, most of the PaaS providers provide tools for project planning and communication.
6. *Diverse client tools*: To make the development easier, PaaS providers provide a wide variety of client tools to help the developer. The client tools include CLI, web CLI, web UI, REST API, and IDE. The developers can choose any tools of their choice. These client tools are also capable of handling billing and subscription management.

5.3.2 Suitability of PaaS

Most of the start-up SaaS development companies and independent software vendors (ISVs) widely use PaaS in developing an application. PaaS technology is getting attention from other traditional software development companies also. PaaS is a suitable option for the following situations:

1. *Collaborative development*: To increase the time to market and development efficiency, there is a need for a common place where the development team and other stakeholders of the application can collaborate with each other. Since PaaS services provide a collaborative development environment, it is a suitable option for applications that need collaboration among developers and other third parties to carry out the development process.
2. *Automated testing and deployment*: Automated testing and building of an application are very useful while developing applications at a very short time frame. The automated testing tools reduce the time spent in manual testing tools. Most of the PaaS services offer automated testing and deployment capabilities. The development team needs to concentrate more on development rather than testing and deployment. Thus, PaaS services are the best option where there is a need for automated testing and deployment of the applications.
3. *Time to market*: The PaaS services follow the iterative and incremental development methodologies that ensure that the application is in the market as per the time frame given. For example, the PaaS services are the best option for application development that uses agile development methodologies. If the software vendor wants their application to be in the market as soon as possible, then the PaaS services are the best option for the development.

PaaS is used widely to accelerate the application development process to ensure the time to market. Most of the start-up companies and ISVs started migrating to the PaaS services. Even though it is used widely, there are some situations where PaaS may not be the best option:

1. *Frequent application migration*: The major problem with PaaS services are vendor lock-in. Since there are no common standards followed among PaaS providers, it is very difficult to migrate the application from one PaaS provider to the other.
2. *Customization at the infrastructure level*: PaaS is an abstracted service, and the PaaS users do not have full control over the underlying infrastructure. There are some application development platforms that need some configuration or customization of underlying infrastructure. In these

situations, it is not possible to customize the underlying infrastructure with PaaS. If the application development platform needs any configuration at the hardware level, it is not recommended to go for PaaS.

3. *Flexibility at the platform level:* PaaS provides template-based applications where all the different programming languages, databases, and message queues are predefined. It is an advantage if the application is a generic application.
4. *Integration with on-premise application:* A company might have used PaaS services for some set of applications. For some set of applications, they might have used on-premise platforms. Since many PaaS services use their own proprietary technologies to define the application stack, it may not match with the on-premise application stack. This makes the integration of application hosted in on-premise platform and PaaS platform a difficult job.

5.3.3 Pros and Cons of PaaS

The main advantage of using PaaS is that it hides the complexity of maintaining the platform and underlying infrastructure. This allows the developers to work more on implementing the important functionalities of the application. Apart from this, the PaaS has the following benefits:

1. *Quick development and deployment:* PaaS provides all the required development and testing tools to develop, test, and deploy the software in one place. Most of the PaaS services automate the testing and deployment process as soon as the developer completes the development. This speeds up application development and deployment than traditional development platforms.
2. *Reduces TCO:* The developers need not buy licensed development and testing tools if PaaS services are selected. Most of the traditional development platforms requires high-end infrastructure for its working, which increases the TCO of the application development company. But, PaaS allows the developers to rent the software, development platforms, and testing tools to develop, build, and deploy the application. PaaS does not require high-end infrastructure also to develop the application, thus reducing the TCO of the development company.
3. *Supports agile software development:* Nowadays, most of the new-generation applications are developed using agile methodologies. Many ISVs and SaaS development companies started adopting agile methodologies for application development. PaaS services support agile methodologies that the ISVs and other development companies are looking for.
4. *Different teams can work together:* The traditional development platform does not have extensive support for collaborative development.

PaaS services support developers from different places to work together on the same project. This is possible because of the online common development platform provided by PaaS providers.

5. *Ease of use:* The traditional development platform uses any one of CLI- or IDE-based interfaces for development. Some developers may not be familiar with the interfaces provided by the application development platform. This makes the development job a little bit difficult. But, PaaS provides a wide variety of client tools such as CLI, web CLI, web UI, APIs, and IDEs. The developers are free to choose any client tools of their choice. Especially, the web UI-based PaaS services increase the usability of the development platform for all types of developers.
6. *Less maintenance overhead:* In on-premise applications, the development company or software vendor is responsible for maintaining the underlying hardware. They need to recruit skilled administrators to maintain the servers. This overhead is eliminated by the PaaS services as the underlying infrastructure is maintained by the infrastructure providers. This gives freedom to developers to work on the application development.
7. *Produces scalable applications:* Most of the applications developed using PaaS services are web application or SaaS application. These applications require better scalability on the extra load. For handling extra load, the software vendors need to maintain an additional server. It is very difficult for a new start-up company to provide extra servers based on the additional load. But, PaaS services are providing built-in scalability to the application that is developed using the PaaS platform.

PaaS provides a lot of benefits to developers when compared to the traditional development environment. On the other hand, it contains drawbacks, which are described in the following:

1. *Vendor lock-in:* The major drawback with PaaS providers are vendor lock-in. The main reason for vendor lock-in is lack of standards. There are no common standards followed among the different PaaS providers. The other reason for vendor lock-in is proprietary technologies used by PaaS providers. Most of the PaaS vendors use the proprietary technologies that are not compatible with the other PaaS providers. The vendor lock-in problem of PaaS services does not allow the applications to be migrated from one PaaS provider to the other.
2. *Security issues:* Like in the other cloud services, security is one of the major issues in PaaS services. Since data are stored in off-premise third-party servers, many developers are afraid to go for PaaS

services. Of course, many PaaS providers provide mechanisms to protect the user data, and it is not sufficient to feel the safety of on-premise deployment. When selecting the PaaS provider, the developer should review the regulatory, compliance, and security policies of the PaaS provider with their own security requirements. If not properly reviewed, the developers or users are at the risk of data security breach.

3. *Less flexibility:* PaaS providers do not give much freedom for the developers to define their own application stack. Most of the PaaS providers provide many programming languages, databases, and other development tools. But, it is not extensive and does not satisfy all developer needs. Only some of the PaaS providers allow developers to extend the PaaS tools with the custom or new programming languages. Still most of the PaaS providers do not provide flexibility to the developers.
4. *Depends on Internet connection:* Since the PaaS services are delivered over the Internet, the developers should depend on Internet connectivity for developing the application. Even though some of the providers allow offline access, most of the PaaS providers do not allow offline access. With slow Internet connection, the usability and efficiency of the PaaS platform do not satisfy the developer requirements.

5.3.4 Summary of PaaS Providers

PaaS providers are more in the IT market for public as well as the private clouds. Table 5.2 gives a summary of popular private and public PaaS providers.

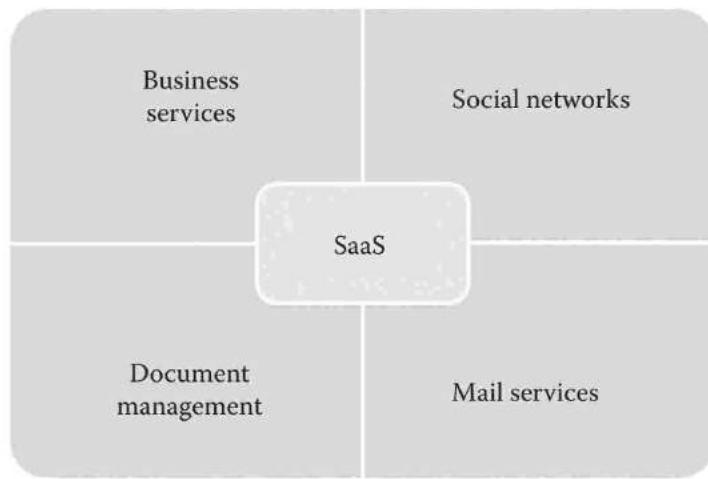
5.4 Software as a Service

SaaS changes the way the software is delivered to the customers. In the traditional software model, the software is delivered as a license-based product that needs to be installed in the end user device. Since SaaS is delivered as an on-demand service over the Internet, there is no need to install the software to the end user's devices. SaaS services can be accessed or disconnected at any time based on the end user's needs. SaaS services can be accessed from any lightweight web browsers on any devices such as laptops, tablets, and smartphones. Some of the SaaS services can be accessed from a thin client that does not contain much storage space and cannot run much software like the traditional desktop PCs. The important benefits of using thin clients for accessing the SaaS application are as follows: it is less vulnerable to attack, has a longer life cycle, consumes less power, and is less expensive. A typical

TABLE 5.2
Summary of Popular PaaS Providers

Provider	License	Deployment Model	Supported Languages	Supported Frameworks	Supported Databases	Client Tools
Cloud Foundry	Open source and proprietary	Public	Python, PHP, Java, Groovy, Scala, and Ruby	Spring, Grails, Play, Node.js, Lift, Rails, Sinatra, and Rack	MySQL, PostgreSQL, MongoDB, and Redis	cf. CLI, IDEs, and build tools
Google App Engine	Proprietary	Public	Python, Java, Groovy, JRuby, Scala, Clojure, Go, and PHP	Django, CherryPy, Pyramid, Flask, web2py, and webapp2.	Google Cloud SQL, Datastore, BigTable, and Blobstore	APIs
Heroku	Proprietary	Public	Ruby, Java, Scala, Clojure and Python, PHP, and Perl	Rails, Play, Django, and Node.js.	ClearDB, PostgreSQL, Cloudant, Membase, MongoDB, and Redis	CLI and RESTful API
Microsoft Windows Azure	Proprietary	Public	.Net, PHP, Python, Ruby, and Java	Django, Rails, Drupal, Joomla, WordPress, DotNetNuke, and Node.js.	SQL Azure, MySQL, MongoDB, and CouchDB	RESTful API and IDEs

Red Hat OpenShift Online	Proprietary	Public	Java, Ruby, Python, PHP, and Perl	Node.js, Rails, Drupal, Joomla, WordPress, Django, EE6, Spring, Play, Sinatra, Rack, and Zend.	MySQL, PostgreSQL, and MongoDB	Web UI, APIs, CLI, and IDEs
ActiveState Stackato	Proprietary	Private	Java, Perl, PHP, Python, Ruby, Scala, Clojure, and Go	Spring, Node.js, Drupal, Joomla, WordPress, Django, Rails, and Sinatra.	MySQL, PostgreSQL, MongoDB, and Redis	CLI and IDE
Apprenda	Proprietary	Private	.Net and Java	Most of the frameworks from .Net.	SQL Server	REST APIs
CloudBees	Proprietary	Private	Java, Groovy, and Scala	Spring, JRails, JRuby, and Grails.	MySQL, PostgreSQL, MongoDB, and CouchDB	API, SDK, and IDEs
Cumulogic	Proprietary	Private	Java, PHP, and Python	Spring and Grails.	MySQL, MongoDB, and Couchbase	RESTful API
Gigaspaces Cloudify	Open source	Private	Any programming language specified by recipe	Rails, Play, and others.	MySQL, MongoDB, Couchbase, Cassandra, and others	CLI, web UI, and REST API

**FIGURE 5.8**

Services provided by SaaS Providers.

SaaS provider may provide business services, social networks, document management, and mail services as shown in Figure 5.8:

1. *Business services*: Most of the SaaS providers started providing a variety of business services that attract start-up companies. The business SaaS services include ERP, CRM, billing, sales, and human resources.
2. *Social networks*: Since social networking sites are extensively used by the general public, many social networking service providers adopted SaaS for their sustainability. Since the number of users of the social networking sites is increasing exponentially, cloud computing is the perfect match for handling the variable load.
3. *Document management*: Since most of the enterprises extensively use electronic documents, most of the SaaS providers started providing services that are used to create, manage, and track electronic documents.
4. *Mail services*: E-mail services are currently used by many people. The future growth in e-mail usage is unpredictable. To handle the unpredictable number of users and the load on e-mail services, most of the e-mail providers started offering their services as SaaS services.

5.4.1 Characteristics of SaaS

SaaS services are different and give more benefits to end users than the traditional software. The following are the essential characteristics of SaaS services that make it unique from traditional software:

1. *One to many*: SaaS services are delivered as a one-to-many model where a single instance of the application can be shared by multiple tenants or customers.

2. *Web access:* SaaS services provide web access to the software. It allows the end user to access the application from any location if the device is connected to the Internet.
3. *Centralized management:* Since SaaS services are hosted and managed from the central location, management of the SaaS application becomes easier. Normally, the SaaS providers will perform the automatic updates that ensure that each tenant is accessing the most recent version of the application without any user-side updates.
4. *Multidevice support:* SaaS services can be accessed from any end user devices such as desktops, laptops, tablets, smartphones, and thin clients.
5. *Better scalability:* Since most of the SaaS services leverage PaaS and IaaS for its development and deployment, it ensures a better scalability than the traditional software. The dynamic scaling of underlying cloud resources makes SaaS applications work efficiently even with varying loads.
6. *High availability:* SaaS services ensure the 99.99% availability of user data as proper backup and recovery mechanisms are implemented at the back end.
7. *API integration:* SaaS services have the capability of integrating with other software or service through standard APIs.

5.4.2 Suitability of SaaS

SaaS is popular among individuals and start-up companies because of the benefits it provides. Most of the traditional software users are looking for SaaS versions of the software as SaaS has several advantages over traditional applications. SaaS applications are the best option for the following:

1. *On-demand software:* The licensing-based software model requires buying full packaged software and increases the spending on buying software. Some of the occasionally used software does not give any ROI. Because of this, many end users are looking for a software that they can use as and when they needed. If the end users are looking for on-demand software rather than the licensing-based full-term software, then the SaaS model is the best option.
2. *Software for start-up companies:* When using any traditional software, the end user should buy devices with minimum requirements specified by the software vendor. This increases the investment on buying hardware for start-up companies. Since SaaS services do not require high-end infrastructure for accessing, it is a suitable option for start-up companies that can reduce the initial expenditure on buying high-end hardware.

3. *Software compatible with multiple devices:* Some of the applications like word processors or mail services need better accessibility from different devices. The SaaS applications are adaptable with almost all the devices.
4. *Software with varying loads:* We cannot predict the load on popular applications such as social networking sites. The user may connect or disconnect from applications anytime. It is very difficult to handle varying loads with the traditional infrastructure. With the dynamic scaling capabilities, SaaS applications can handle varying loads efficiently without disrupting the normal behavior of the application.

Most of the traditional software vendors moved to SaaS business as it is an emerging software delivery model that attracts end users. But still many traditional applications do not have its SaaS versions. This implies that SaaS applications may not be the best option for all types of software. The SaaS delivery model is not the best option for the applications mentioned in the following:

1. *Real-time applications:* Since SaaS applications depend on Internet connectivity, it may not work better with low Internet speed. If data are stored far away from the end user, the latency issues may delay the data retrieval timings. Real-time applications require fast processing of data that may not be possible with the SaaS applications because of the dependency on high-speed Internet connectivity and latency issues.
2. *Applications with confidential data:* Data security, data governance, and data compliance are always issues with SaaS applications. Since data are stored with third-party service providers, there is no surety that our data will be safe. If the stored confidential data get lost, it will make a serious loss to the organization. It is not recommended to go for SaaS for applications that handle confidential data.
3. *Better on-premise application:* Some of the on-premise applications might fulfill all the requirements of the organization. In such situations, migrating to the SaaS model may not be the best option.

5.4.3 Pros and Cons of SaaS

SaaS applications are used by a wide range of individuals and start-up industries for its cost-related benefits. Apart from the cost-related benefits, SaaS services provide the following benefits:

1. *No client-side installation:* SaaS services do not require client-side installation of the software. The end users can access the services directly from the service provider data center without any installation. There is no need of high-end hardware to consume SaaS

services. It can be accessed from thin clients or any handheld devices, thus reducing the initial expenditure on buying high-end hardware.

2. *Cost savings:* Since SaaS services follow the utility-based billing or pay-as-you-go billing, it demands the end users to pay for what they have used. Most of the SaaS providers offer different subscription plans to benefit different customers. Sometimes, the generic SaaS services such as word processors are given for free to the end users.
3. *Less maintenance:* SaaS services eliminate the additional overhead of maintaining the software from the client side. For example, in the traditional software, the end user is responsible for performing bulk updates. But in SaaS, the service provider itself maintains the automatic updates, monitoring, and other maintenance activities of the applications.
4. *Ease of access:* SaaS services can be accessed from any devices if it is connected to the Internet. Accessibility of SaaS services is not restricted to any particular devices. It is adaptable to all the devices as it uses the responsive web UI.
5. *Dynamic scaling:* SaaS services are popularly known for elastic dynamic scaling. It is very difficult for on-premise software to provide dynamic scaling capability as it requires additional hardware. Since the SaaS services leverage elastic resources provided by cloud computing, it can handle any type of varying loads without disrupting the normal behavior of the application.
6. *Disaster recovery:* With proper backup and recovery mechanisms, replicas are maintained for every SaaS services. The replicas are distributed across many servers. If any server fails, the end user can access the SaaS from other servers. It eliminates the problem of single point of failure. It also ensures the high availability of the application.
7. *Multitenancy:* Multitenancy is the ability given to the end users to share a single instance of the application. Multitenancy increases resource utilization from the service provider side.

Even though SaaS services are used by many individuals and start-up industries, the adoption from the large industries is very low. The major problem with SaaS services is security to the data. All companies are worried about the security of their data that are hosted in the service provider data center. The following are the major problems with SaaS services:

1. *Security:* Security is the major concern in migrating to SaaS application. Since the SaaS application is shared between many end users, there is a possibility of data leakage. Here, the data are stored in the service provider data center. We cannot simply trust some third-party service provider to store our company-sensitive and

TABLE 5.3

Summary of Popular SaaS Providers

Provider	Services Provided
Salseforce.com	On-demand CRM solutions
Google Apps	Gmail, Google Calendar, Talk, Docs, and Sites
Microsoft Office 356	Online office suite, software, plus services
NetSuite	ERP, accounting, order management, inventory, CRM, professional services automation (PSA), and e-commerce applications
Concur	Integrated travel and expense management solutions
GoToMeeting	Online meeting, desktop sharing, and video-conferencing software
Constant Contact	E-mail marketing, social-media marketing, online survey, event marketing, digital storefronts, and local deals tools
Workday, Inc.	Human capital management, payroll, and financial management
Oracle CRM	CRM applications
Intacct	Financial management and accounting software solutions

confidential data. The end user should be careful while selecting the SaaS provider to avoid unnecessary data loss.

2. *Connectivity requirements:* SaaS applications require Internet connectivity for accessing it. Sometimes, the end user's Internet connectivity might be very slow. In such situations, the user cannot access the services with ease. The dependency on high-speed Internet connection is a major problem in SaaS applications.
3. *Loss of control:* Since the data are stored in a third-party and off-premise location, the end user does not have any control over the data. The degree of control over the SaaS application and data is lesser than the on-premise application.

5.4.4 Summary of SaaS Providers

There are many SaaS providers who provide SaaS services such as ERP, CRM, billing, document management, and mail services. Table 5.3 gives a summary of popular SaaS vendors in the market.

5.5 Other Cloud Service Models

The basic cloud services such as IaaS, PaaS, and SaaS are widely used by many individual and start-up companies. Now, cloud computing becomes the dominant technology that drives the IT world. Because of the extensive use of basic cloud services, the end users realize the importance and benefits

of specific services such as network, storage, and database. The basic cloud service models are the unified models that contain multiple services in it. Now, the end users' expectation changed, and they are expecting the individual services to be offered by service providers. This makes most of the service providers to think about the separate services that meet end user requirements. Many service providers already started offering separate services such as network, desktop, database, and storage on demand as given in the following:

1. *NaaS* is an ability given to the end users to access virtual network services that are provided by the service provider. Like other cloud service models, NaaS is also a business model for delivering virtual network services over the Internet on a pay-per-use basis. In on-premise data center, the IT industries spent a lot of money to buy network hardware to manage in-house networks. But, cloud computing changes networking services into a utility-based service. NaaS allows network architects to create virtual networks, virtual network interface cards (NICs), virtual routers, virtual switches, and other networking components. Additionally, it allows the network architect to deploy custom routing protocols and enables the design of efficient in-network services, such as data aggregation, stream processing, and caching. Some of the popular services provided by NaaS include virtual private network (VPN), bandwidth on demand (BoD), and mobile network virtualization.
2. *Desktop as a Service (DEaaS)* is an ability given to the end users to use desktop virtualization without buying and managing their own infrastructure. DEaaS is a pay-per-use cloud service delivery model in which the service provider manages the back-end responsibilities of data storage, backup, security, and upgrades. The end users are responsible for managing their own desktop images, applications, and security. Accessing the virtual desktop provided by the DEaaS provider is device, location, and network independent. DEaaS services are simple to deploy, are highly secure, and produce better experience on almost all devices.
3. *STaaS* is an ability given to the end users to store the data on the storage services provided by the service provider. STaaS allows the end users to access the files at any time from any place. The STaaS provider provides the virtual storage that is abstracted from the physical storage of any cloud data center. STaaS is also a cloud business model that is delivered as a utility. Here, the customers can rent the storage from the STaaS provider. STaaS is commonly used as a backup storage for efficient disaster recovery.
4. *DBaaS* is an ability given to the end users to access the database service without the need to install and maintain it. The service provider

is responsible for installing and maintaining the databases. The end users can directly access the services and can pay according to their usage. DBaaS automates the database administration process. The end users can access the database services through any API or web UIs provided by the service provider. The DBaaS eases the database administration process. Popular examples of DBaaS include SimpleDB, DynamoDB, MongoDB as a Service, GAE datastore, and ScaleDB.

5. *Data as a Service (DaaS)* is an ability given to the end users to access the data that are provided by the service provider over the Internet. DaaS provides data on demand. The data may include text, images, sounds, and videos. DaaS is closely related to other cloud service models such as SaaS and STaaS. DaaS can be easily integrated with SaaS or STaaS for providing the composite service. DaaS is highly used in geography data services and financial data services. The advantages of DaaS include agility, cost effectiveness, and data quality.
6. *SECaaS* is an ability given to the end user to access the security service provided by the service provider on a pay-per-use basis. In SECaaS, the service provider integrates their security services to benefit the end users. Generally, the SECaaS includes authentication, antivirus, antimalware/spyware, intrusion detection, and security event management. The security services provided by the SECaaS providers are typically used for securing the on-premise or in-house infrastructure and applications. Some of the SECaaS providers include Cisco, McAfee, Panda Software, Symantec, Trend Micro, and VeriSign.
7. *IDaaS* is an ability given to the end users to access the authentication infrastructure that is managed and provided by the third-party service provider. The end user of IDaaS is typically an organization or enterprise. Using IDaaS services, any organization can easily manage their employees' identity without any additional overhead. Generally, IDaaS includes directory services, federated services, registration, authentication services, risk and event monitoring, single sign-on services, and identity and profile management.

The different new service models discussed in this section emerged after the introduction of cloud computing. This field still evolves and introduces new service models based on the end user's needs. Many researchers from industry and academia already started introducing their innovative idea to take cloud computing to the next level. Apart from the service models discussed in this chapter, cloud computing researchers are thinking to add more service models. Now, cloud computing moves to the scenario where everything can be given as a service. This can be termed as Everything as a Service (XaaS). In the future, we expect many new service models to achieve the goal

of XaaS. XaaS may include Backup as a Service (BaaS), Communication as a Service (CaaS), Hadoop as a Service (HaaS), Disaster Recovery as a Service (DRaaS), Testing as a Service (TaaS), Firewall as a Service (FWaaS), Virtual Private Network as a Service (VPNaas), Load Balancers as a Service (LBaaS), Message Queue as a Service (MQaaS), and Monitoring as a Service (MaaS).

5.6 Summary

Cloud computing composes of three basic service models and four deployment models. The service models include IaaS, PaaS, and SaaS, and the deployment models include private, public, community, and hybrid clouds. The service models decide the type of services provided by each service models. The deployment models decide the way of delivering the services. Cloud computing eliminates a lot of management overhead at each level. IaaS hides the complexity of maintaining the underlying hardware. PaaS hides the complexity of maintaining the development platform and the hardware. In the same way, SaaS hides the complexity of maintaining the application, the development platform, and the hardware. All the basic cloud service models have an essential characteristic of cloud computing: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. Apart from these characteristics, each service model has its own unique characteristics. The characteristics of IaaS include web access to the resources, centralized management, elasticity and dynamic scaling, shared infrastructure, preconfigured VMs, and metered services. The characteristics of PaaS include all in one, web access to the development platform, offline access, built-in scalability, collaborative platform, and diverse client tools. The characteristics of SaaS include one to many, web access, centralized management, multidevice support, better scalability, high availability, and API integration. Cost-based benefits of cloud services make individuals and start-up industries use cloud computing for their computing needs. Even though cloud services are used by many individuals and start-up industries, adaptability from large enterprises is very low. We cannot use cloud services in all places. Generally, the cloud service can be used in start-up companies where the initial capital investment is very low. Cloud services cannot be used when the application uses more sensitive and confidential data. The general benefits of cloud services are cost savings, elastic and dynamic scaling, and centralized management. The general drawbacks include security issues, interoperability issues, and performance issues. Apart from the basic service models, there are other specific cloud services also provided by some vendors, including NaaS, STaaS, DBaaS, SECaaS, and IDaaS.

Review Points

- *SaaS*: The ability given to the end users to access on-demand software services provided by the SaaS provider over the Internet (see Section 5.1).
 - *PaaS*: The ability given to the end users to access on-demand development and deployment platforms provided by the PaaS provider over the Internet (See Section 5.1).
 - *IaaS*: The ability given to the end users to access on-demand computing resources provided by the IaaS provider over the Internet (see Section 5.1).
 - *NaaS*: The ability given to the end users to access on-demand virtual networking components provided by the NaaS provider over the Internet (see Section 5.5).
 - *DEaaS*: The ability given to the end users to access on-demand virtual desktop services provided by the DEaaS provider *over the Internet* (see Section 5.5).
 - *STaaS*: The ability given to the end users to access on-demand cloud storage provided by the STaaS provider over the Internet (see Section 5.5).
 - *DBaaS*: The ability given to the end users to access the on-demand database services provided by the DBaaS provider over the Internet (see Section 5.5).
 - *DaaS*: The ability given to the end users to access on-demand data provided by the DaaS provider over the Internet (see Section 5.5).
 - *SECaas*: The ability given to the end users to access security services provided by the SECaas provider over the Internet (see Section 5.5).
 - *IDaaS*: The ability given to the end users to access the identity management services provided by the IDaaS provider over the Internet (see Section 5.5).
-

Review Questions

1. Define Infrastructure as a Service (IaaS).
2. Define Platform as a Service (PaaS).
3. Define Software as a Service (SaaS).
4. Write short notes on end user and service provider responsibilities of cloud service models with a suitable diagram.
5. Write short notes on the deployment and delivery of cloud service models with a neat diagram.

6. Explain in detail about the overview of IaaS, PaaS, and SaaS with suitable diagrams.
 7. Write short notes on the characteristics of IaaS, PaaS, and SaaS.
 8. Explain the suitability of different cloud service models.
 9. Write short notes on pros and cons of IaaS, PaaS, and SaaS.
 10. Write short notes on cloud service models that emerged after the introduction of cloud computing.
-

Further Reading

- Costa, P., M. Migliavacca, P. Pietzuch, and A. L. Wolf. NaaS: Network-as-a-service in the cloud. Hot-ICE, 2012.
- Kepes, B. Understanding the cloud computing stack: SaaS, PaaS, IaaS, 2013. Available [Online]: http://www.rackspace.com/knowledge_center/whitepaper/understanding-the-cloud-computing-stack-saas-paas-iaas. Accessed February 18, 2013.
- Liu, F., J. Tong, J. Mao, R. B. Bohn, J. V. Messina, M. L. Badger, and D. M. Leaf. NIST cloud computing reference architecture. NIST Special Publication 500-292, 2011. Available [Online]: http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505. Accessed September 3, 2013.
- Orlando, D. Convert cloud computing service models, Part 1: Infrastructure as a service. Technical article. IBM developerWorks, 2011a.
- Orlando, D. Convert cloud computing service models, Part 2: Platform as a service. Technical article. IBM developerWorks, 2011b.
- Orlando, D. Convert cloud computing service models, Part 3: Software as a service. Technical article. IBM developerWorks, 2011c.
- Mell, P. and T. Grance. The NIST definition of cloud computing. NIST Special Publication 800-145, 2011. Available [Online]: <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>. Accessed September 3, 2013.
- Cloud Computing Identity as a Service (IDaaS). Available [Online]: http://www.tutorialspoint.com/cloud_computing/cloud_computing_identity_as_a_service.htm. Accessed September 3, 2013.
- Cloud Computing Infrastructure as a Service (IaaS). Available [Online]: http://www.tutorialspoint.com/cloud_computing/cloud_computing_infrastructure_as_a_service.htm. Accessed April 1, 2014.
- Cloud Computing Network as a Service (NaaS). Available [Online]: http://www.tutorialspoint.com/cloud_computing/cloud_computing_network_as_a_service.htm. Accessed March 4, 2014.
- Cloud Computing Platform as a Service (PaaS). Available [Online]: http://www.tutorialspoint.com/cloud_computing/cloud_computing_platform_as_a_service.htm. Accessed February 27, 2014.
- Cloud Computing Software as a Service (SaaS). Available [Online]: http://www.tutorialspoint.com/cloud_computing/cloud_computing_software_as_a_service.htm. Accessed March 20, 2014.

