UNIT-2 CLOUD COMPUTING ARCHITECTURE

Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects

Introduction

- Utility-oriented data centers are the first outcome of cloud computing, and they serve as the infrastructure through which the services are implemented and delivered.
- Any cloud service, whether virtual hardware, development platform, or application software, relies on a distributed infrastructure owned by the provider or rented from a third party.
- As noted in the previous definition, the characterization of a cloud is quite general: It can be implemented using a datacenter, a collection of clusters, or a heterogeneous distributed system composed of desktop PCs, workstations, and servers. Commonly, clouds are built by relying on one or more datacenters.
- In most cases hardware resources are virtualized to provide isolation of workloads and to best exploit the infrastructure. According to the specific service delivered to the end user, different layers can be stacked on top of the virtual infrastructure: a virtual machine manager, a development platform, or a specific application middleware.

As noted in earlier chapters, the cloud computing paradigm emerged as a result of the convergence of various existing models, technologies, and concepts that changed the way we deliver and use IT services.

A broad definition of the phenomenon could be as follows: Cloud computing is a utility-oriented and Internet-centric way of delivering IT services on demand. These services cover the entire computing stack: from the hardware infrastructure packaged as a set of virtual machines to software services such as development platforms and distributed applications.

This definition captures the most important and fundamental aspects of cloud computing. We now discuss a reference model that aids in categorization of cloud technologies, applications, and services

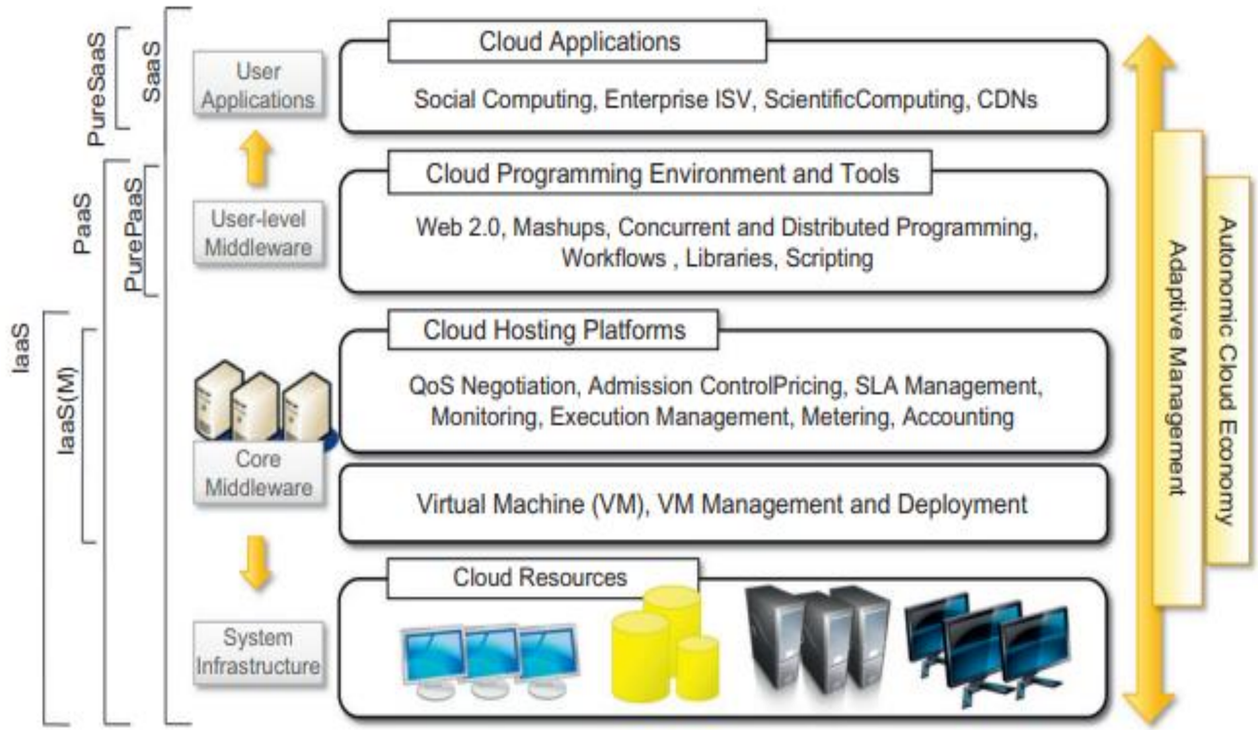


FIGURE 4.1

The cloud computing architecture.

Infrastructure- and hardware-as-a-service

- Infrastructure- and Hardware-as-a-Service (IaaS/HaaS) solutions are the most popular and developed market segment of cloud computing.
- They deliver customizable infrastructure on demand. The available options within the IaaS offering umbrella range from single servers to entire infrastructures, including network devices, load balancers, and database and Web servers.
- The main technology used to deliver and implement these solutions is hardware virtualization: one or more virtual machines opportunely configured and interconnected define the distributed system on top of which applications are installed and deployed.
- Virtual machines also constitute the atomic components that are deployed and priced according to the specific features of the virtual hardware: memory, number of processors, and disk storage. IaaS/HaaS solutions bring all the benefits of hardware virtualization: workload partitioning, application isolation, sandboxing, and hardware tuning.
- From the perspective of the service provider, IaaS/HaaS allows better exploiting the IT infrastructure and provides a more secure environment where executing third party applications. From the perspective of the customer it reduces the administration and maintenance cost as well as the capital costs allocated to purchase hardware.
- At the same time, users can take advantage of the full customization offered by virtualization to deploy their infrastructure in the cloud; in most cases virtual machines come with only the selected operating system installed and the system can be

- configured with all the required packages and applications. Other solutions provide prepackaged system images that already contain the software stack required for the most common uses: Web servers, database servers, or LAMP1 stacks.
- Besides the basic virtual machine management capabilities, additional services can be provided, generally including the following: SLA resource-based allocation, workload management, support for infrastructure design through advanced Web interfaces, and the ability to integrate third-party IaaS solutions.

Table 4.1 Cloud Computing Services Classification

Category	Characteristics	Product Type	Vendors and Products
SaaS	Customers are provided with applications that are accessible anytime and from anywhere.	Web applications and services (Web 2.0)	SalesForce.com (CRM) Clarizen.com (project management) Google Apps
PaaS	Customers are provided with a platform for developing applications hosted in the cloud.	Programming APIs and frameworks Deployment systems	Google AppEngine Microsoft Azure Manjrasoft Aneka Data Synapse
IaaS/HaaS	Customers are provided with virtualized hardware and storage on top of which they can build their infrastructure.	Virtual machine management infrastructure Storage management Network management	Amazon EC2 and S3 GoGrid Nirvanix

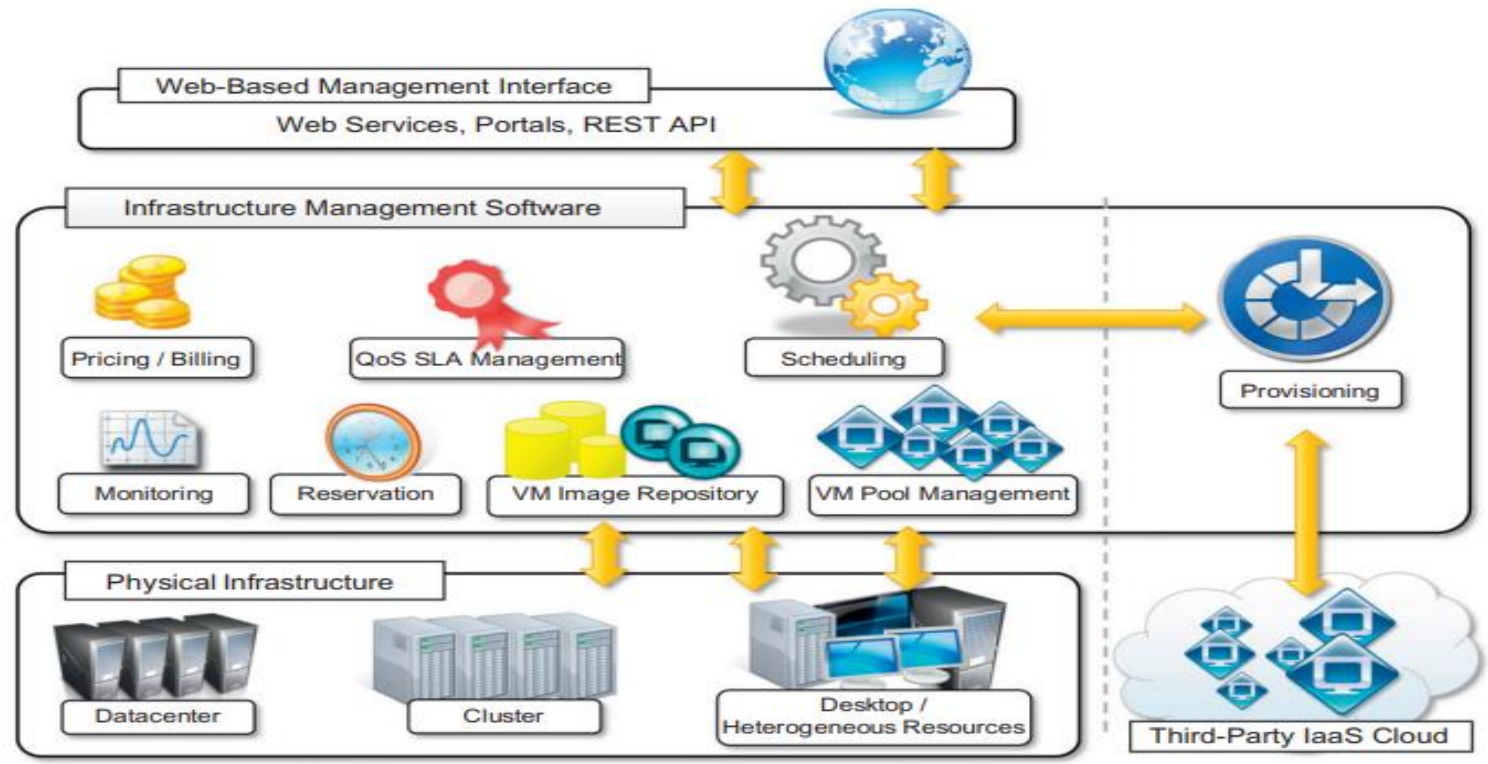


FIGURE 4.2

Infrastructure-as-a-Service reference implementation.

UNIT-2 CLOUD COMPUTING ARCHITECTURE

Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects

Platform as a service Platform-as-a-Service

- (PaaS) solutions provide a development and deployment platform for running applications in the cloud. They constitute the middleware on top of which applications are built. A general overview of the features characterizing the PaaS approach is given in Figure 4.3

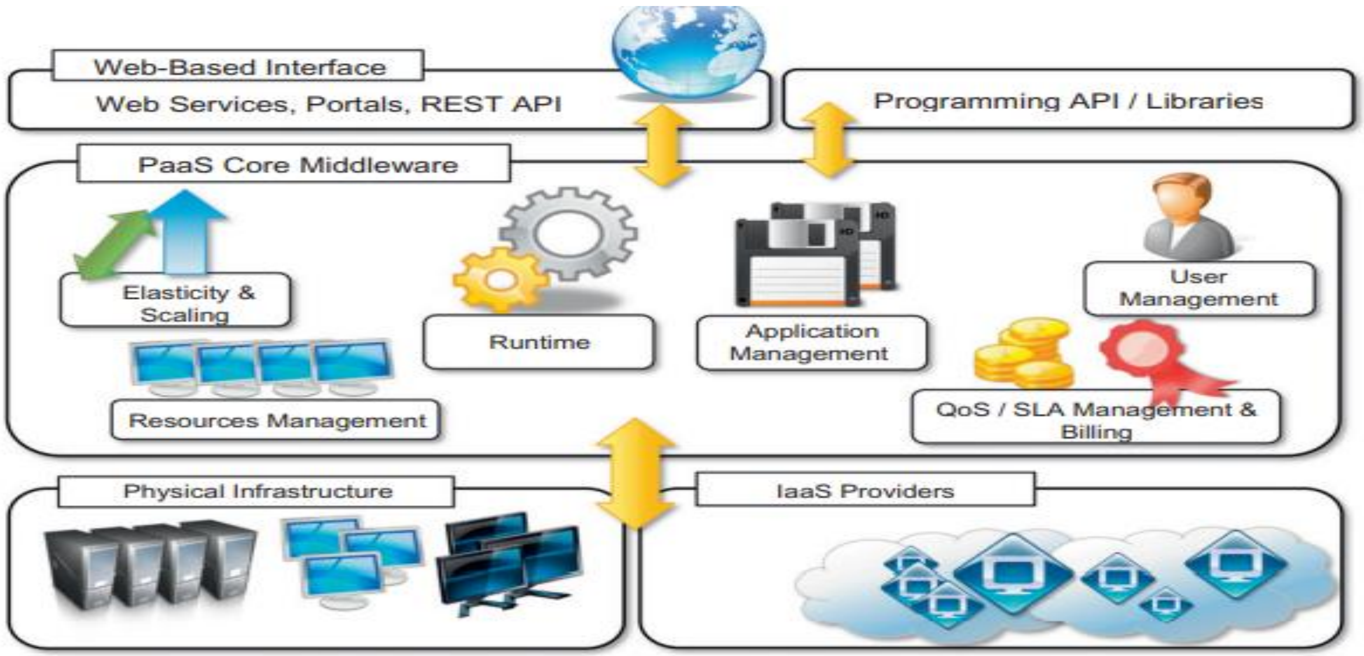


FIGURE 4.3

The Platform-as-a-Service reference model.

Table 4.2 Platform-as-a-Service Offering Classification

Category	Description	Product Type	Vendors and Products
PaaS-I	Runtime environment with Web-hosted application development platform. Rapid application prototyping.	Middleware + Infrastructure Middleware + Infrastructure	Force.com Longjump
PaaS-II	Runtime environment for scaling Web applications. The runtime could be enhanced by additional components that provide scaling capabilities.	Middleware + Infrastructure Middleware Middleware + Infrastructure Middleware + Infrastructure Middleware + Infrastructure Middleware	Google AppEngine AppScale Heroku Engine Yard Joyent Smart Platform GigaSpaces XAP
PaaS-III	Middleware and programming model for developing distributed applications in the cloud.	Middleware + Infrastructure Middleware Middleware Middleware Middleware Middleware	Microsoft Azure DataSynapse Cloud IQ Manjrasof Aneka Apprenda SaaSGrid GigaSpaces DataGrid

there are some essential characteristics that identify a PaaS solution:

- *Runtime framework*. This framework represents the "software stack" of the PaaS model and the most intuitive aspect that comes to people's minds when they refer to PaaS solutions. The runtime framework executes end-user code according to the policies set by the user and the provider.
- *Abstraction*. PaaS solutions are distinguished by the higher level of abstraction that they provide. Whereas in the case of IaaS solutions the focus is on delivering "raw" access to virtual or physical infrastructure, in the case of PaaS the focus is on the applications the cloud must support. This means that PaaS solutions offer a way to deploy and manage applications on the cloud rather than a bunch of virtual machines on top of which the IT infrastructure is built and configured.
- *Automation*. PaaS environments automate the process of deploying applications to the infrastructure, scaling them by provisioning additional resources when needed. This process is performed automatically and according to the SLA made between the customers and the provider. This feature is normally not native in IaaS solutions, which only provide ways to provision more resources.
- *Cloud services*. PaaS offerings provide developers and architects with services and APIs, helping them to simplify the creation and delivery of elastic and highly available cloud applications. These services are the key differentiators among competing PaaS solutions and generally include specific components for developing applications, advanced services for application monitoring, management, and reporting.

- Another essential component for a PaaS-based approach is the ability to integrate third-party cloud services offered from other vendors by leveraging service-oriented architecture. Such integration should happen through standard interfaces and protocols.
- This opportunity makes the development of applications more agile and able to evolve according to the needs of customers and users. Many of the PaaS offerings provide this facility, which is naturally built into the framework they leverage to provide a cloud computing solution.
- One of the major concerns of leveraging PaaS solutions for implementing applications is vendor lock-in. Differently from IaaS solutions, which deliver bare virtual servers that can be fully customized in terms of the software stack installed, PaaS environments deliver a platform for developing applications, which exposes a well-defined set of APIs and, in most cases, binds the application to the specific runtime of the PaaS provider.
- Even though a platform-based approach strongly simplifies the development and deployment cycle of applications, it poses the risk of making these applications completely dependent on the provider. Such dependency can become a significant obstacle in retargeting the application to another environment and runtime if the commitments made with the provider cease.
- The impact of the vendor lock-in on applications obviously varies according to the various solutions. Some of them, such as Force.com, rely on a proprietary runtime framework, which makes the retargeting process very difficult.

Software as a Service

- Software-as-a-Service (SaaS) is a software delivery model that provides access to applications through the Internet as a Web-based service. It provides a means to free users from complex hardware and software management by offloading such tasks to third parties, which build applications accessible to multiple users through a Web browser.
- In this scenario, customers neither need install anything on their premises nor have to pay considerable up-front costs to purchase the software and the required licenses.
- They simply access the application website, enter their credentials and billing details, and can instantly use the application, which, in most of the cases, can be further customized for their needs. On the provider side, the specific details and features of each customer's application are maintained in the infrastructure and made available on demand.

- The SaaS model is appealing for applications serving a wide range of users and that can be adapted to specific needs with little further customization.
- This requirement characterizes SaaS as a “one-to-many” software delivery model, whereby an application is shared across multiple users. This is the case of CRM3 and ERP4 applications that constitute common needs for almost all enterprises, from small to medium-sized and large business.
- Every enterprise will have the same requirements for the basic features concerning CRM and ERP; different needs can be satisfied with further customization.
- This scenario facilitates the development of software platforms that provide a general set of features and support specialization and ease of integration of new components.

The benefits delivered at that stage were the following: • Software cost reduction and total cost of ownership (TCO) were paramount • Service-level improvements • Rapid implementation • Standalone and configurable applications • Rudimentary application and data integration • Subscription and pay-as-you-go (PAYG) pricing

In the software as a service model, the application, or service, is deployed from a centralized datacenter across a network—Internet, Intranet, LAN, or VPN—providing access and use on a recurring fee basis. Users "rent," "subscribe to," "are assigned," or "are granted access to" the applications from a central provider. Business models vary according to the level to which the software is streamlined, to lower price and increase efficiency, or value-added through customization to further improve digitized business processes.

The analysis carried out by SIIA was mainly oriented to cover application service providers (ASPs) and all their variations, which capture the concept of software applications consumed as a service in a broader sense. ASPs already had some of the core characteristics of SaaS:

- The product sold to customer is *application access*.
- The application is centrally managed.
- The service delivered is *one-to-many*.
- The service delivered is an integrated solution *delivered on the contract*, which means provided as promised.

ponents and services they want to integrate.

Software-as-a-Service applications can serve different needs. CRM, ERP, and social networking applications are definitely the most popular ones. SalesForce.com is probably the most successful and popular example of a CRM service. It provides a wide range of services for applications: customer relationship and human resource management, enterprise resource planning, and many other features. SalesForce.com builds on top of the Force.com platform, which provides a fully featured environment for building applications. It offers either a programming language or a visual environment to arrange components together for building applications. In addition to the basic features provided, the integration with third-party-made applications enriches SalesForce.com's value. In particular, through AppExchange customers can publish, search, and integrate new services and features into their existing applications. This makes SalesForce.com applications completely extensible and customizable. Similar solutions are offered by NetSuite and RightNow. NetSuite is an integrated software business suite featuring financials, CRM, inventory, and ecommerce functionalities integrated all together. RightNow is customer experience-centered SaaS application that integrates together different features, from chat to Web communities, to support the common activity of an enterprise.

Another important class of popular SaaS applications comprises social networking applications such as Facebook and professional networking sites such as LinkedIn. Other than providing the basic features of networking, they allow incorporating and extending their capabilities by integrating third-party applications. These can be developed as plug-ins for the hosting platform, as happens for Facebook, and made available to users, who can select which applications they want to add to their profile. As a result, the integrated applications get full access to the network of contacts and users' profile data. The nature of these applications can be of different types: office automation components, games, or integration with other existing services.

Office automation applications are also an important representative for SaaS applications: Google Documents and Zoho Office are examples of Web-based applications that aim to address all user needs for documents, spreadsheets, and presentation management. They offer a Web-based interface for creating, managing, and modifying documents that can be easily shared among users and made accessible from anywhere.

It is important to note the role of SaaS solution enablers, which provide an environment in which to integrate third-party services and share information with others. A quite successful example is Box.net, an SaaS application providing users with a Web space and profile that can be enriched and extended with third-party applications such as office automation, integration with CRM-based solutions, social Websites, and photo editing.

UNIT-2 CLOUD COMPUTING ARCHITECTURE

Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects

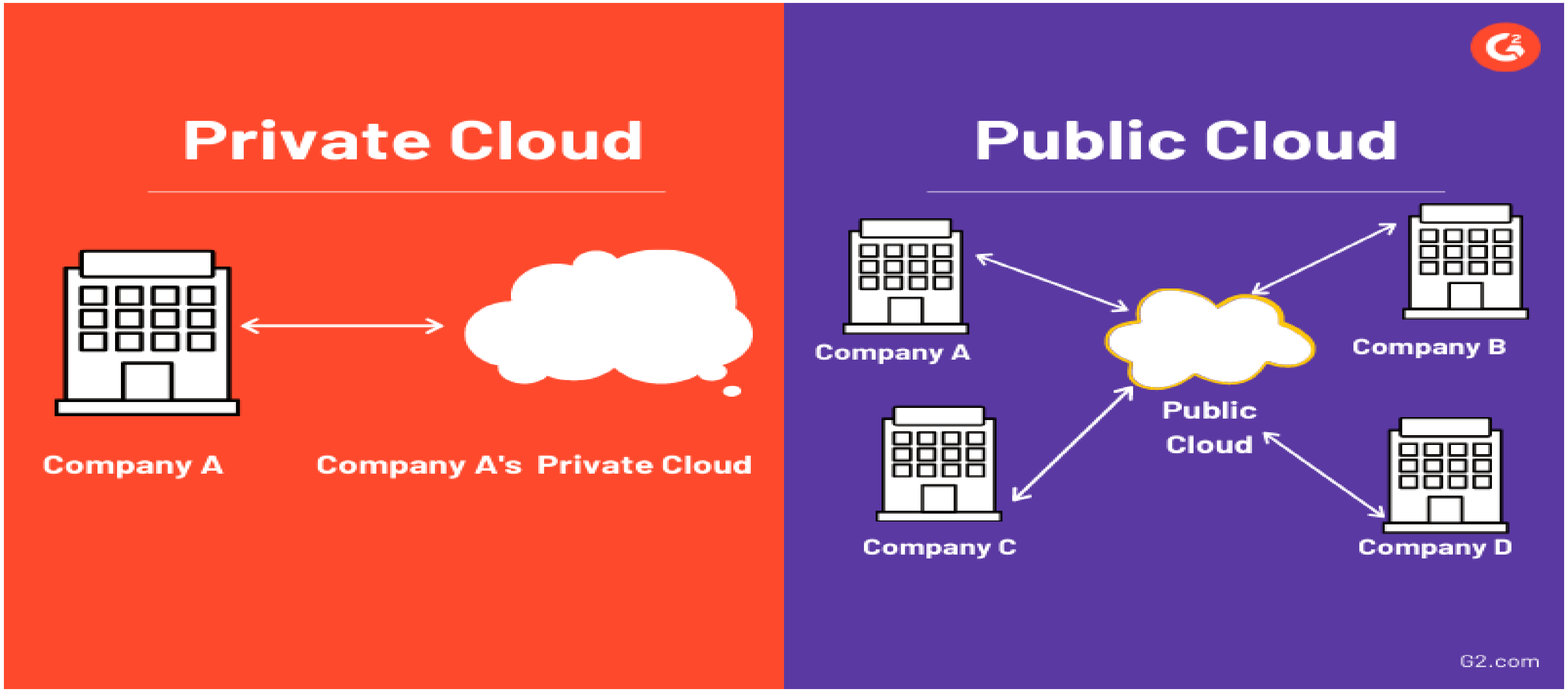
Types of Clouds

Clouds constitute the primary outcome of cloud computing. They are a type of parallel and distributed system harnessing physical and virtual computers presented as a unified computing resource. Clouds build the infrastructure on top of which services are implemented and delivered to customers. Such infrastructures can be of different types and provide useful information about the nature and the services offered by the cloud. A more useful classification is given according to the administrative domain of a cloud: It identifies the boundaries within which cloud computing services are implemented, provides hints on the underlying infrastructure adopted to support such services, and qualifies them. It is then possible to differentiate four different types of cloud:

- *Public clouds*. The cloud is open to the wider public.
- *Private clouds*. The cloud is implemented within the private premises of an institution and generally made accessible to the members of the institution or a subset of them.
- *Hybrid or heterogeneous clouds*. The cloud is a combination of the two previous solutions and most likely identifies a private cloud that has been augmented with resources or services hosted in a public cloud.
- *Community clouds*. The cloud is characterized by a multi-administrative domain involving different deployment models (public, private, and hybrid), and it is specifically designed to address the needs of a specific industry.

Almost all the implementations of clouds can be classified in this categorization. In the following sections, we provide brief characterizations of these clouds.

Public Clouds


Private Cloud
Company A
Company A's Private Cloud
Public Cloud
Company A
Company B
Public
Cloud
Company C
Company D
G2.com

Public clouds constitute the first expression of cloud computing. They are a realization of the canonical view of cloud computing in which the services offered are made available to anyone, from anywhere, and at any time through the Internet. From a structural point of view they are a distributed system, most likely composed of one or more datacenters connected together, on top of which the specific services offered by the cloud are implemented. Any customer can easily sign in with the cloud provider, enter her credential and billing details, and use the services offered.

Historically, public clouds were the first class of cloud that were implemented and offered. They offer solutions for minimizing IT infrastructure costs and serve as a viable option for handling peak loads on the local infrastructure. They have become an interesting option for small enterprises, which are able to start their businesses without large up-front investments by completely relying on public infrastructure for their IT needs. What made attractive public clouds compared to the reshaping of the private premises and the purchase of hardware and software was the ability to grow or shrink according to the needs of the related business. By renting the infrastructure or subscribing to application services, customers were able to dynamically upsize or downsize their IT according to the demands of their business. Currently, public clouds are used both to completely replace the IT infrastructure of enterprises and to extend it when it is required.

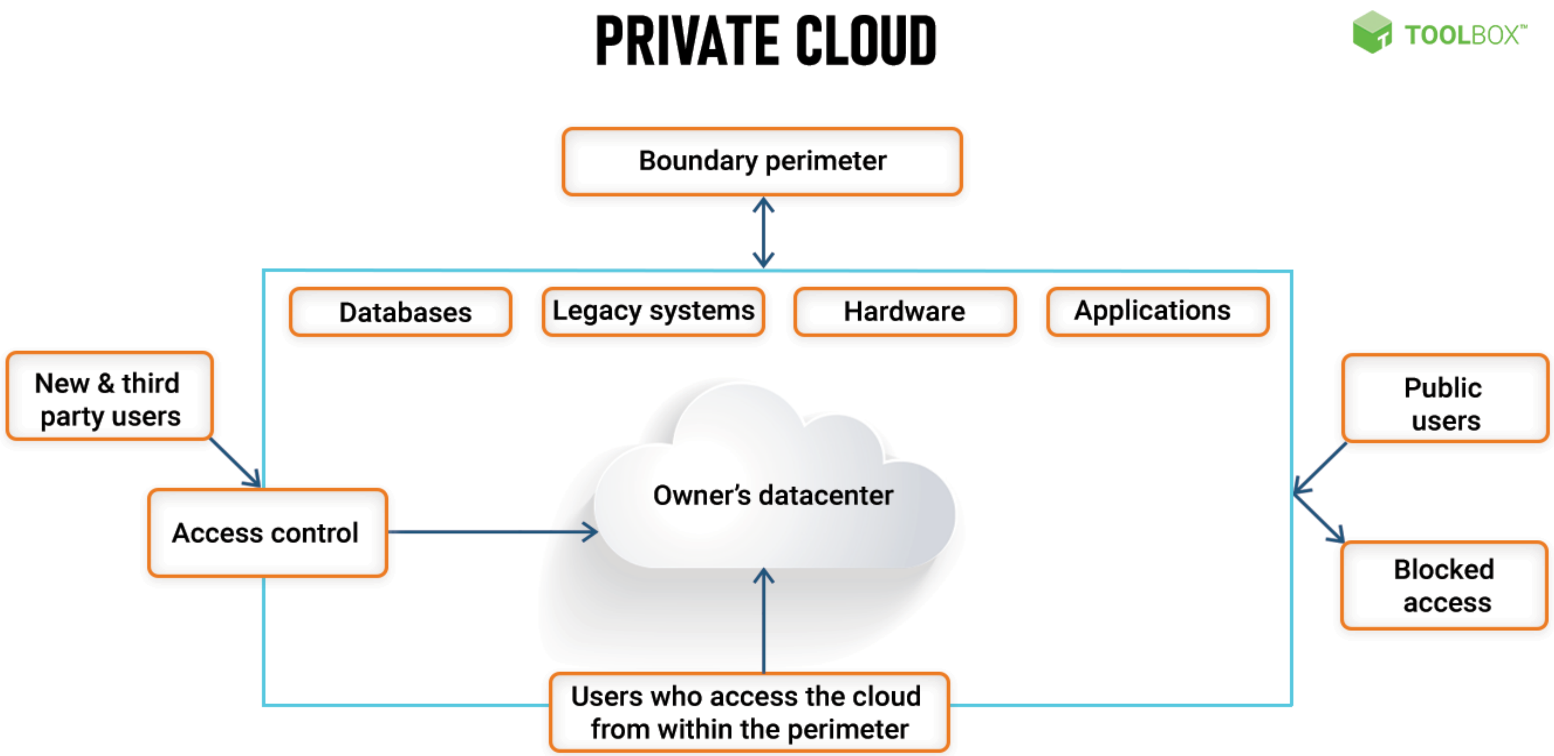
A fundamental characteristic of public clouds is multitenancy. A public cloud is meant to serve a multitude of users, not a single customer. Any customer requires a virtual computing environment that is separated, and most likely isolated, from other users. This is a fundamental requirement to provide effective monitoring of user activities and guarantee the desired performance and the other QoS attributes negotiated with users. QoS management is a very important aspect of public clouds. Hence, a significant portion of the software infrastructure is devoted to monitoring the cloud resources, to bill them according to the contract made with the user, and to keep a complete history of cloud usage for each customer. These features are fundamental to public clouds because they help providers offer services to users with full accountability.

A public cloud can offer any kind of service: infrastructure, platform, or applications. For example, Amazon EC2 is a public cloud that provides infrastructure as a service; Google AppEngine is a public cloud that provides an application development platform as a service; and SalesForce.com is a public cloud that provides software as a service. What makes public clouds peculiar is the way they are consumed: They are available to everyone and are generally architected to support a large quantity of users. What characterizes them is their natural ability to scale on demand and sustain peak loads.

A public cloud can offer any kind of service: infrastructure, platform, or applications. For example, Amazon EC2 is a public cloud that provides infrastructure as a service; Google AppEngine is a public cloud that provides an application development platform as a service; and SalesForce.com is a public cloud that provides software as a service. What makes public clouds peculiar is the way they are consumed: They are available to everyone and are generally architected to support a large quantity of users. What characterizes them is their natural ability to scale on demand and sustain peak loads.

From an architectural point of view there is no restriction concerning the type of distributed system implemented to support public clouds. Most likely, one or more datacenters constitute the physical infrastructure on top of which the services are implemented and delivered. Public clouds can be composed of geographically dispersed datacenters to share the load of users and better serve them according to their locations. For example, Amazon Web Services has datacenters installed in the United States, Europe, Singapore, and Australia; they allow their customers to choose between three different regions: *us-west-1*, *us-east-1*, or *eu-west-1*. Such regions are priced differently and are further divided into availability zones, which map to specific datacenters. According to the specific class of services delivered by the cloud, a different software stack is installed to manage the infrastructure: virtual machine managers, distributed middleware, or distributed applications.

Private Clouds


PRIVATE CLOUD
TOOLBOX
Boundary perimeter
Databases
Legacy systems
Hardware
Applications
New & third party users
Access control
Owner's datacenter
Public users
Blocked access
Users who access the cloud from within the perimeter

organization.

Private clouds are virtual distributed systems that rely on a private infrastructure and provide internal users with dynamic provisioning of computing resources. Instead of a pay-as-you-go model as in public clouds, there could be other schemes in place, taking into account the usage of the cloud and proportionally billing the different departments or sections of an enterprise. Private clouds have the advantage of keeping the core business operations in-house by relying on the existing IT infrastructure and reducing the burden of maintaining it once the cloud has been set up. In this scenario, security concerns are less critical, since sensitive information does not flow out of the private infrastructure. Moreover, existing IT resources can be better utilized because the private cloud can provide services to a different range of users. Another interesting opportunity that comes with private clouds is the possibility of testing applications and systems at a comparatively lower price rather than public clouds before deploying them on the public virtual infrastructure. A Forrester report [34] on the benefits of delivering in-house cloud computing solutions for enterprises highlighted some of the key advantages of using a private cloud computing infrastructure:

- *Customer information protection*. Despite assurances by the public cloud leaders about security, few provide satisfactory disclosure or have long enough histories with their cloud offerings to provide warranties about the specific level of security put in place on their systems. In-house security is easier to maintain and rely on.
- *Infrastructure ensuring SLAs*. Quality of service implies specific operations such as appropriate clustering and failover, data replication, system monitoring and maintenance, and disaster recovery, and other uptime services can be commensurate to the application needs. Although public cloud vendors provide some of these features, not all of them are available as needed.
- *Compliance with standard procedures and operations*. If organizations are subject to third-party compliance standards, specific procedures have to be put in place when deploying and executing applications. This could be not possible in the case of the virtual public infrastructure.

All these aspects make the use of cloud-based infrastructures in private premises an interesting option.

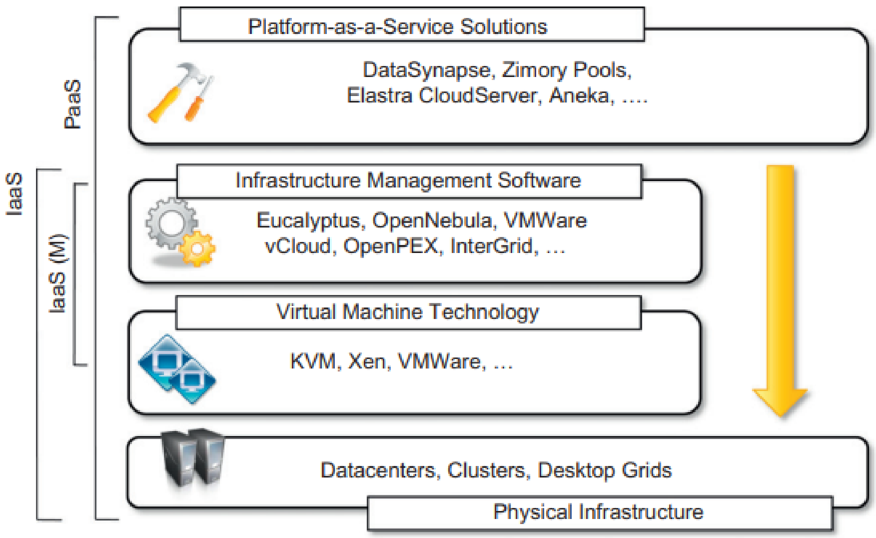
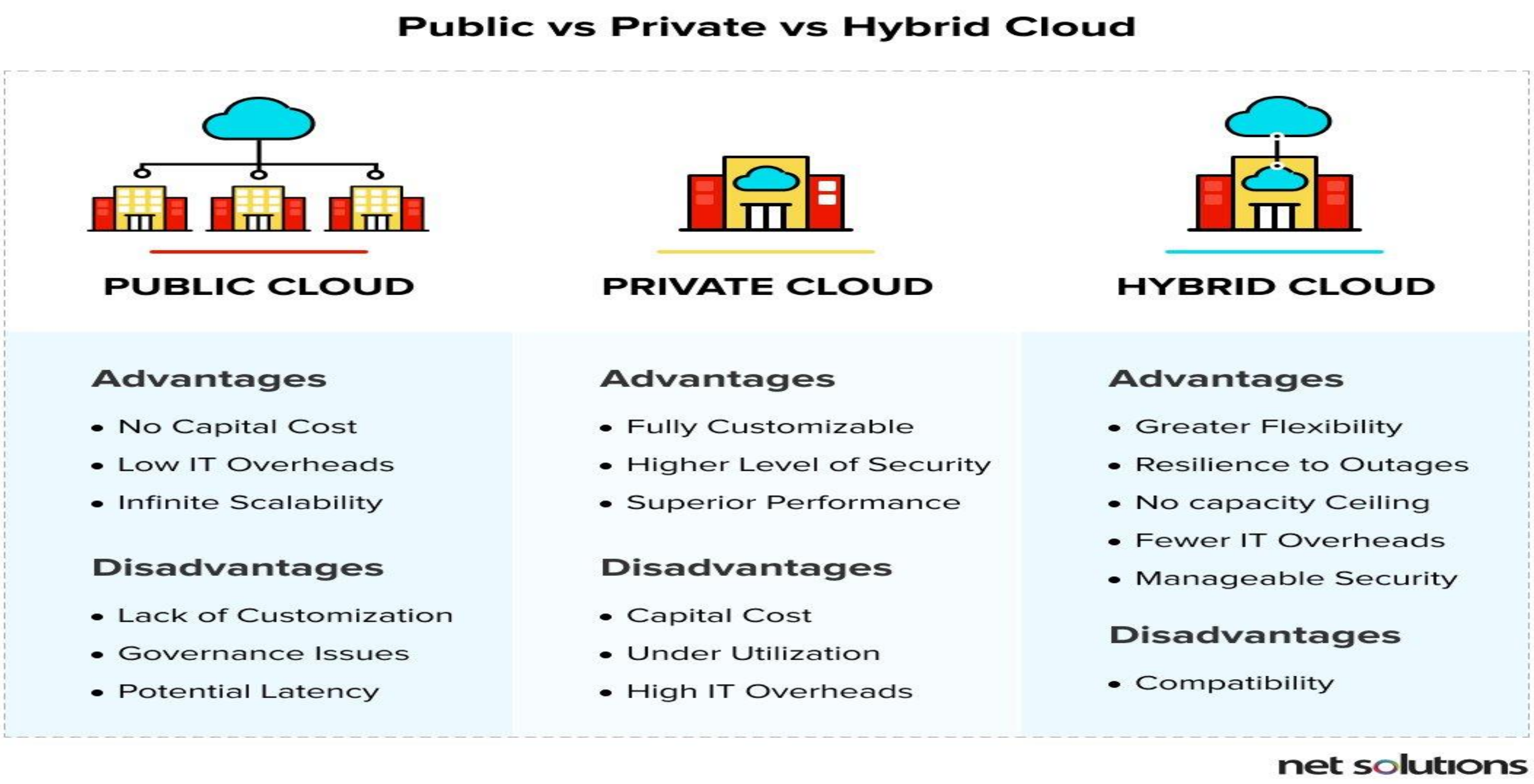


FIGURE 4.4

Private clouds hardware and software stack.


Public vs Private vs Hybrid Cloud
PUBLIC CLOUD
Advantages
No Capital Cost
Low IT Overheads
Infinite Scalability
Disadvantages
Lack of Customization
Governance Issues
Potential Latency
PRIVATE CLOUD
Advantages
Fully Customizable
Higher Level of Security
Superior Performance
Disadvantages
Capital Cost
Under Utilization
High IT Overheads
HYBRID CLOUD
Advantages
Greater Flexibility
Resilience to Outages
No capacity Ceiling
Fewer IT Overheads
Manageable Security
Disadvantages
Compatibility
net solutions

UNIT-2 CLOUD COMPUTING ARCHITECTURE

Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects

Hybrid Clouds

- Public clouds are large software and hardware infrastructures that have a capability that is huge enough to serve the needs of multiple users, but they suffer from security threats and administrative pitfalls.
- Although the option of completely relying on a public virtual infrastructure is appealing for companies that did not incur IT capital costs and have just started considering their IT needs (i.e., start-ups), in most cases the private cloud option prevails because of the existing IT infrastructure

Private clouds are the perfect solution when it is necessary to keep the processing of information within an enterprise's premises or it is necessary to use the existing hardware and software infrastructure. One of the major drawbacks of private deployments is the inability to scale on demand and to efficiently address peak loads. In this case, it is important to leverage capabilities of public clouds as needed. Hence, a hybrid solution could be an interesting opportunity for taking advantage of the best of the private and public worlds. This led to the development and diffusion of hybrid clouds.

Hybrid clouds allow enterprises to exploit existing IT infrastructures, maintain sensitive information within the premises, and naturally grow and shrink by provisioning external resources and releasing them when they're no longer needed. Security concerns are then only limited to the public portion of the cloud that can be used to perform operations with less stringent constraints but that are still part of the system workload. Figure 4.5 provides a general overview of a hybrid cloud: It is a heterogeneous distributed system resulting from a private cloud that integrates additional services or resources from one or more public clouds. For this reason they are also called *heterogeneous clouds*. As depicted in the diagram, dynamic provisioning is a fundamental component in this scenario. Hybrid clouds address scalability issues by leveraging external resources for exceeding

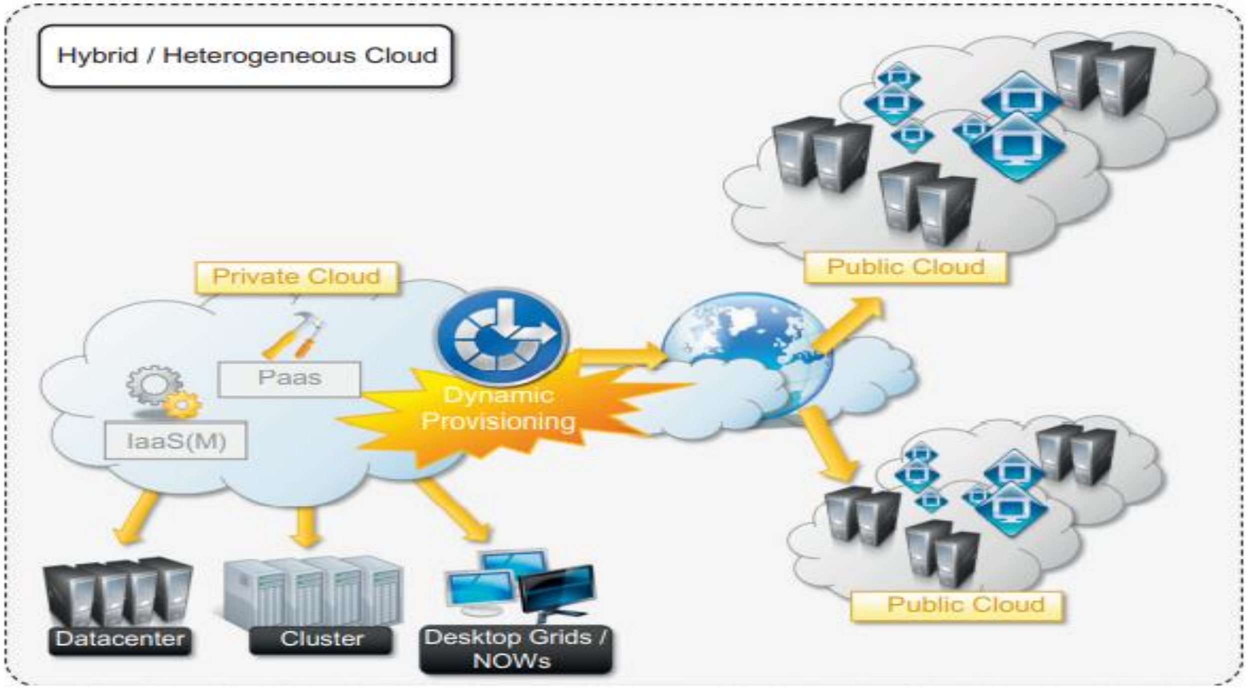


FIGURE 4.5

Hybrid/heterogeneous cloud overview.

capacity demand. These resources or services are temporarily leased for the time required and then released. This practice is also known as *cloudbursting*.[6]

Whereas the concept of hybrid cloud is general, it mostly applies to IT infrastructure rather than software services. Service-oriented computing already introduces the concept of integration of paid software services with existing application deployed in the private premises. In an IaaS scenario, *dynamic provisioning* refers to the ability to acquire on demand virtual machines in order to increase the capability of the resulting distributed system and then release them. Infrastructure management software and PaaS solutions are the building blocks for deploying and managing hybrid clouds. In particular, with respect to private clouds, dynamic provisioning introduces a more complex scheduling algorithm and policies, the goal of which is also to optimize the budget spent to rent public resources.

Infrastructure management software such as OpenNebula already exposes the capability of integrating resources from public clouds such as Amazon EC2. In this case the virtual machine obtained from the public infrastructure is managed as all the other virtual machine instances maintained locally. What is missing is then an advanced scheduling engine that's able to differentiate these resources and provide smart allocations by taking into account the budget available to extend the existing infrastructure. In the case of OpenNebula, advanced schedulers such as Haizea can be integrated to provide cost-based scheduling. A different approach is taken by InterGrid. This is essentially a distributed scheduling engine that manages the allocation of virtual machines in a collection of peer networks. Such networks can be represented by a local cluster, a gateway to a public cloud, or a combination of the two. Once a request is submitted to one of the InterGrid gateways, it is served by possibly allocating virtual instances in all the peered networks, and the allocation of requests is performed by taking into account the user budget and the peering arrangements between networks.

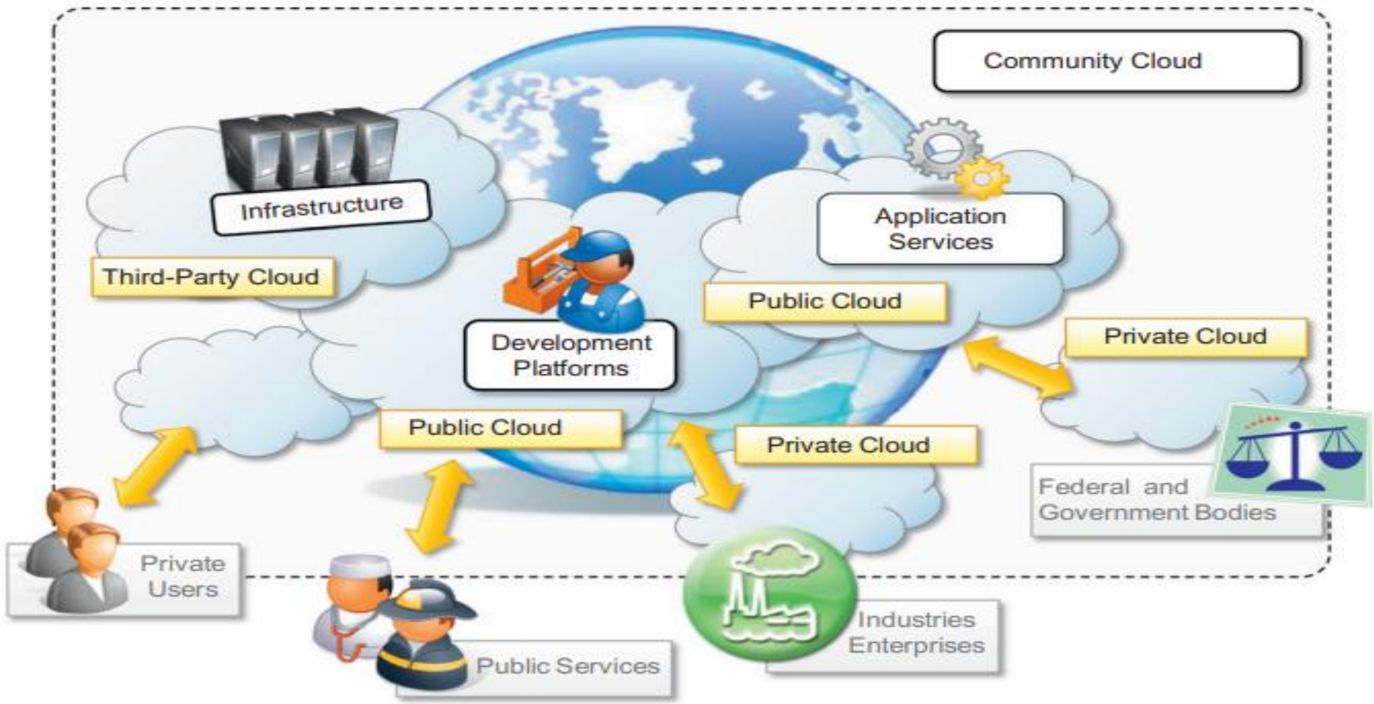


FIGURE 4.6

A community cloud.

Community Clouds,

4.3.4 Community clouds

Community clouds are distributed systems created by integrating the services of different clouds to address the specific needs of an industry, a community, or a business sector. The National Institute of Standards and Technologies (NIST) [43] characterizes community clouds as follows:

> *The infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.*

Figure 4.6 provides a general view of the usage scenario of community clouds, together with reference architecture. The users of a specific community cloud fall into a well-identified community, sharing the same concerns or needs; they can be government bodies, industries, or even simple users, but all of them focus on the same issues for their interaction with the cloud. This is a different scenario than public clouds, which serve a multitude of users with different needs. Community clouds are also different from private clouds, where the services are generally delivered within the institution that owns the cloud.

From an architectural point of view, a community cloud is most likely implemented over multiple administrative domains. This means that different organizations such as government bodies, private enterprises, research organizations, and even public virtual infrastructure providers contribute with their resources to build the cloud infrastructure.

Candidate sectors for community clouds are as follows:

- *Media industry*. In the media industry, companies are looking for low-cost, agile, and simple solutions to improve the efficiency of content production. Most media productions involve an extended ecosystem of partners. In particular, the creation of digital content is the outcome of a collaborative process that includes movement of large data, massive compute-intensive rendering tasks, and complex workflow executions. Community clouds can provide a shared environment where services can facilitate business-to-business collaboration and offer the horsepower in terms of aggregate bandwidth, CPU, and storage required to efficiently support media production.
- *Healthcare industry*. In the healthcare industry, there are different scenarios in which community clouds could be of use. In particular, community clouds can provide a global platform on which to share information and knowledge without revealing sensitive data maintained within the private infrastructure. The naturally hybrid deployment model of community clouds can easily support the storing of patient-related data in a private cloud while using the shared infrastructure for noncritical services and automating processes within hospitals.
- *Energy and other core industries*. In these sectors, community clouds can bundle the comprehensive set of solutions that together vertically address management, deployment, and orchestration of services and operations. Since these industries involve different providers, vendors, and organizations, a community cloud can provide the right type of infrastructure to create an open and fair market.

- *Public sector*. Legal and political restrictions in the public sector can limit the adoption of public cloud offerings. Moreover, governmental processes involve several institutions and agencies and are aimed at providing strategic solutions at local, national, and international administrative levels. They involve business-to-administration, citizen-to-administration, and possibly business-to-business processes. Some examples include invoice approval, infrastructure planning, and public hearings. A community cloud can constitute the optimal venue to provide a distributed environment in which to create a communication platform for performing such operations.
- *Scientific research*. Science clouds are an interesting example of community clouds. In this case, the common interest driving different organizations sharing a large distributed infrastructure is scientific computing.

which each can be at the same time a consumer, a producer, or a coordinator of the services offered by the cloud. The benefits of these community clouds are the following:

- *Openness*. By removing the dependency on cloud vendors, community clouds are open systems in which fair competition between different solutions can happen.
- *Community*. Being based on a collective that provides resources and services, the infrastructure turns out to be more scalable because the system can grow simply by expanding its user base.
- *Graceful failures*. Since there is no single provider or vendor in control of the infrastructure, there is no single point of failure.
- *Convenience and control*. Within a community cloud there is no conflict between convenience and control because the cloud is shared and owned by the community, which makes all the decisions through a collective democratic process.
- *Environmental sustainability*. The community cloud is supposed to have a smaller carbon footprint because it harnesses underutilized resources. Moreover, these clouds tend to be more organic by growing and shrinking in a symbiotic relationship to support the demand of the community, which in turn sustains it.

UNIT-2 CLOUD COMPUTING ARCHITECTURE

Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects

Economics of the Cloud

4.4 Economics of the cloud

The main drivers of cloud computing are economy of scale and simplicity of software delivery and its operation. In fact, the biggest benefit of this phenomenon is financial: the *pay-as-you-go* model offered by cloud providers. In particular, cloud computing allows:

- Reducing the capital costs associated to the IT infrastructure
- Eliminating the depreciation or lifetime costs associated with IT capital assets
- Replacing software licensing with subscriptions
- Cutting the maintenance and administrative costs of IT resources

A *capital cost* is the cost occurred in purchasing an asset that is useful in the production of goods or the rendering of services. Capital costs are one-time expenses that are generally paid up front and that will contribute over the long term to generate profit. The IT infrastructure and the software are capital assets because enterprises require them to conduct their business. At present it does not matter whether the principal business of an enterprise is related to IT, because the business will definitely have an IT department that is used to automate many of the activities that are performed within the enterprise: payroll, customer relationship management, enterprise resource planning, tracking and inventory of products, and others. Hence, IT resources constitute a capital cost for any kind of enterprise. It is good practice to try to keep capital costs low because they introduce expenses that will generate profit over time; more than that, since they are associated with material things they are subject to *depreciation* over time, which in the end reduces the profit of the enterprise because such costs are directly subtracted from the enterprise revenues. In the case of IT capital costs, the depreciation costs are represented by the loss of value of the hardware over time and the aging of software products that need to be replaced because new features are required.

Before cloud computing diffused within the enterprise, the budget spent on IT infrastructure and software constituted a significant expense for medium-sized and large enterprises. Many enterprises own a small or medium-sized datacenter that introduces several operational costs in terms of maintenance, electricity, and cooling. Additional operational costs are occurred in maintaining an IT department and an IT support center. Moreover, other costs are triggered by the purchase of potentially expensive software. With cloud computing these costs are significantly reduced or simply disappear according to its penetration. One of the advantages introduced by the cloud computing model is that it shifts the capital costs previously allocated to the purchase of hardware and software into operational costs inducted by renting the infrastructure and paying subscriptions for the use of software. These costs can be better controlled according to the business needs and prosperity of the enterprise. Cloud computing also introduces reductions in administrative and maintenance costs. That is, there is no or limited need for having administrative staff take care of the management of the cloud infrastructure. At the same time, the cost of IT support staff is also reduced. When it comes to depreciation costs, they simply disappear for the enterprise, since in a scenario where all the IT needs are served by the cloud there are no IT capital assets that depreciate over time.

The amount of cost savings that cloud computing can introduce within an enterprise is related to the specific scenario in which cloud services are used and how they contribute to generate a profit for the enterprise. In the case of a small startup, it is possible to completely leverage the cloud for many aspects, such as:

- IT infrastructure
- Software development
- CRM and ERP

In this case it is possible to completely eliminate capital costs because there are no initial IT assets. The situation is completely different in the case of enterprises that already have a considerable amount of IT assets. In this case, cloud computing, especially IaaS-based solutions, can help manage unplanned capital costs that are generated by the needs of the enterprise in the short term. In this case, by leveraging cloud computing, these costs can be turned into operational costs that last as long as there is a need for them. For example, IT infrastructure leasing helps more efficiently manage peak loads without inducing capital expenses. As soon as the increased load does not justify the use of additional resources, these can be released and the costs associated with them disappear. This is the most adopted model of cloud computing because many enterprises already have IT facilities. Another option is to make a slow transition toward cloud-based solutions while the capital IT assets get depreciated and need to be replaced. Between these two cases there is a wide variety of scenarios in which cloud computing could be of help in generating profits for enterprises.

Another important aspect is the elimination of some indirect costs that are generated by IT assets, such as software licensing and support and carbon footprint emissions. With cloud computing, an enterprise uses software applications on a subscription basis, and there is no need for any licensing fee because the software providing the service remains the property of the provider. Leveraging IaaS solutions allows room for datacenter consolidation that in the end could result in a smaller carbon footprint. In some countries such as Australia, the carbon footprint emissions are taxable, so by reducing or completely eliminating such emissions, enterprises can pay less tax.

In terms of the pricing models introduced by cloud computing, we can distinguish three different strategies that are adopted by the providers:

- *Tiered pricing*. In this model, cloud services are offered in several tiers, each of which offers a fixed computing specification and SLA at a specific price per unit of time. This model is used by Amazon for pricing the EC2 service, which makes available different server configurations in terms of computing capacity (CPU type and speed, memory) that have different costs per hour.
- *Per-unit pricing*. This model is more suitable to cases where the principal source of revenue for the cloud provider is determined in terms of units of specific services, such as data transfer and memory allocation. In this scenario customers can configure their systems more efficiently according to the application needs. This model is used, for example, by GoGrid, which makes customers pay according to RAM/hour units for the servers deployed in the GoGrid cloud.
- *Subscription-based pricing*. This is the model used mostly by SaaS providers in which users pay a periodic subscription fee for use of the software or the specific component services that are integrated in their applications.

Cloud Definition

- As discussed earlier, there have been several attempts made to define cloud computing and to provide a classification of all the services and technologies identified as such.
- One of the most comprehensive formalizations is noted in the NIST working definition of cloud computing It characterizes cloud computing as on-demand self-service, broad network access, resource-pooling, rapid elasticity, and measured service; classifies services as SaaS, PaaS, and IaaS; and categorizes deployment models as public, private, community, and hybrid clouds.
- The view is in line with our discussion and shared by many IT practitioners and academics
- These characterizations and taxonomies reflect what is meant by cloud computing at the present time, but being in its infancy the phenomenon is constantly evolving, and the same will happen to the attempts to capture the real nature of cloud computing.

UNIT-2 CLOUD COMPUTING ARCHITECTURE

Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards, Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects

CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

Cloud Interoperability and Standards

Cloud computing is a service-based model for delivering IT infrastructure and applications like utilities such as power, water, and electricity. To fully realize this goal, introducing standards and allowing interoperability between solutions offered by different vendors are objectives of fundamental importance. Vendor lock-in constitutes one of the major strategic barriers against the seamless adoption of cloud computing at all stages. In particular there is major fear on the part of enterprises in which IT constitutes the significant part of their revenues. Vendor lock-in can prevent a customer from switching to another competitor's solution, or when this is possible, it happens at considerable conversion cost and requires significant amounts of time. This can occur either because the customer wants to find a more suitable solution for customer needs or because the vendor is no longer able to provide the required service. The presence of standards that are actually implemented and adopted in the cloud computing community could give room for interoperability and then lessen the risks resulting from vendor lock-in.

The current state of standards and interoperability in cloud computing resembles the early Internet era, when there was no common agreement on the protocols and technologies used and each organization had its own network. Yet the first steps toward a standardization process have been made, and a few organizations, such as the Cloud Computing Interoperability Forum (CCIF),[9] The standardization efforts are mostly concerned with the lower level of the cloud computing architecture, which is the most popular and developed. In particular, in the IaaS market, the use of a proprietary virtual machine format constitutes the major reasons for the vendor lock-in, and efforts to provide virtual machine image compatibility between IaaS vendors can possibly improve the level of interoperability among them. The Open Virtualization Format (OVF) [51] is an attempt to provide a common format for storing the information and metadata describing a virtual machine image. Even though the OVF provides a full specification for packaging and distributing virtual machine images in completely platform-independent fashion, it is supported by few vendors that use it to import static virtual machine images. The challenge is providing standards for supporting the migration of running instances, thus allowing the real ability of switching from one infrastructure vendor to another in a completely transparent manner.

Another direction in which standards try to move is devising a general reference architecture for cloud computing systems and providing a standard interface through which one can interact with them. At the moment the compatibility between different solutions is quite restricted, and the lack of a common set of APIs make the interaction with cloud-based solutions vendor specific. In the IaaS market, Amazon Web Services plays a leading role, and other IaaS solutions, mostly open source, provide AWS-compatible APIs, thus constituting themselves as valid alternatives. Even in this case, there is no consistent trend in devising some common APIs for interfacing with IaaS (and, in general, XaaS), and this constitutes one of the areas in which a considerable improvement can be made in the future.

Scalability and fault tolerance

Scalability and fault tolerance

The ability to scale on demand constitutes one of the most attractive features of cloud computing. Clouds allow scaling beyond the limits of the existing in-house IT resources, whether they are infrastructure (compute and storage) or applications services. To implement such a capability, the cloud middleware has to be designed with the principle of scalability along different dimensions in mind—for example, performance, size, and load. The cloud middleware manages a huge number of resource and users, which rely on the cloud to obtain the horsepower that they cannot obtain within the premises without bearing considerable administrative and maintenance costs. These costs are a reality for whomever develops, manages, and maintains the cloud middleware and offers the service to customers. In this scenario, the ability to tolerate failure becomes fundamental, sometimes even more important than providing an extremely efficient and optimized system. Hence, the challenge in this case is designing highly scalable and fault-tolerant systems that are easy to manage and at the same time provide competitive performance.

Security, Trust, and Privacy

Security, trust, and privacy issues are major obstacles for massive adoption of cloud computing. The traditional cryptographic technologies are used to prevent data tampering and access to sensitive information. The massive use of virtualization technologies exposes the existing system to new threats, which previously were not considered applicable. For example, it might be possible that applications hosted in the cloud can process sensitive information; such information can be stored within a cloud storage facility using the most advanced technology in cryptography to protect data and then be considered safe from any attempt to access it without the required permissions. Although these data are processed in memory, they must necessarily be decrypted by the legitimate application, but since the application is hosted in a managed virtual environment it becomes accessible to the virtual machine manager that by program is designed to access the memory pages of such an application. In this case, what is experienced is a lack of control over the environment in which the application is executed, which is made possible by leveraging the cloud. It then happens that a new way of using existing technologies creates new opportunities for additional threats to the security of applications. The lack of control over their own data and processes also poses severe problems for the trust we give to the cloud service provider and the level of privacy we want to have for our data.

On one side we need to decide whether to trust the provider itself; on the other side, specific regulations can simply prevail over the agreement the provider is willing to establish with us concerning the privacy of the information managed on our behalf. Moreover, cloud services delivered to the end user can be the result of a complex stack of services that are obtained by third parties via the primary cloud service provider. In this case there is a chain of responsibilities in terms of service delivery that can introduce more vulnerability for the secure management of data, the enforcement of privacy rules, and the trust given to the service provider. In particular, when a violation of privacy or illegal access to sensitive information is detected, it could become difficult to identify who is liable for such violations. The challenges in this area are, then, mostly concerned with devising secure and trustable systems from different perspectives: technical, social, and legal.

Organizational aspects

Cloud computing introduces a significant change in the way IT services are consumed and managed. More precisely, storage, compute power, network infrastructure, and applications are delivered as metered services over the Internet. This introduces a billing model that is new within typical enterprise IT departments, which requires a certain level of cultural and organizational process maturity. In particular, a wide acceptance of cloud computing will require a significant change to business processes and organizational boundaries. Some interesting questions arise in considering the role of the IT department in this new scenario. In particular, the following questions have to be considered:

- What is the new role of the IT department in an enterprise that completely or significantly relies on the cloud?
- How will the compliance department perform its activity when there is a considerable lack of control over application workflows?

- What are the implications (political, legal, etc.) for organizations that lose control over some aspects of their services?
- What will be the perception of the end users of such services?

From an organizational point of view, the lack of control over the management of data and processes poses not only security threats but also new problems that previously did not exist. Traditionally, when there was a problem with computer systems, organizations developed strategies and solutions to cope with them, often by relying on local expertise and knowledge. One of the major advantages of moving IT infrastructure and services to the cloud is to reduce or completely remove the costs related to maintenance and support. As a result, users of such infrastructure and services lose a reference to deal with for IT troubleshooting. At the same time, the existing IT staff is required to have a different kind of competency and, in general, fewer skills, thus reducing their value. These are the challenges from an organizational point of view that must be faced and that will significantly change the relationships within the enterprise itself among the various groups of people working together.

Thank You