UNIT – III

Energy efficiency in Clouds:

- Modern datacenters that operate under the cloud computing model are hosting a variety of applications ranging from those that……
  - Run for a few seconds (e.g., serving requests of Web applications such as ecommerce and social network portals) to those that run for longer periods of time (e.g., simulations or large dataset processing) on shared hardware platforms.
- The need to manage multiple applications in a datacenter creates the challenge of on-demand resource provisioning and allocation in response to time-varying workloads.
  - Normally, datacenter resources are statically allocated to applications based on peak load characteristics in order to maintain isolation and provide performance guarantees.
- Until recently, high performance has been the sole concern in datacenter deployments, and this demand has been fulfilled without paying much attention to energy consumption.
- According to the McKinsey report on "Revolutionizing Data Center Energy Efficiency", a typical datacenter consumes as much energy as 25,000 households.
- Energy costs of powering a typical data center doubles every five years.
- Because energy costs are increasing while availability decreases, there is a need to shift focus from optimizing datacenter resource management for pure performance alone to optimizing for energy efficiency while maintaining high service-level performance (see Figure 11.1).

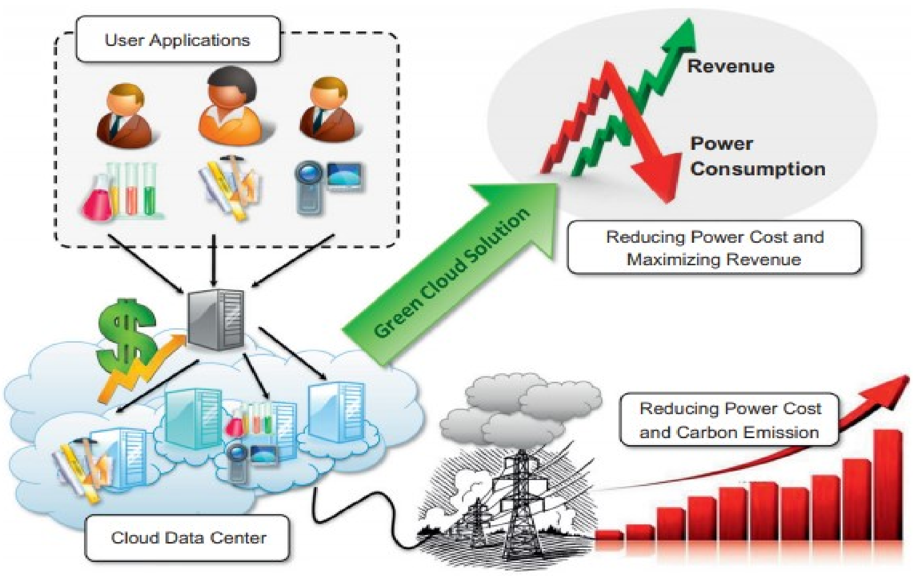

FIGURE 11.1 A "green" cloud computing scenario.

- Datacenters are not only expensive to maintain, they are also unfriendly to the environment.
    - Carbon emissions due to datacenters worldwide are now more than the emissions of both Argentina and the Netherlands.
- High energy costs and huge carbon footprints are incurred due to the massive amount of electricity needed to power and cool the numerous servers hosted in these datacenters.
- Cloud service providers need to adopt measures to ensure that their profit margins are not dramatically reduced due to high energy costs.
    - According to Amazon's estimate, the energy related costs of its datacenters amount to 42% of the total budget, which includes both direct power consumption and the cooling infrastructure.
- Lowering the energy usage of datacenters is a challenging and complex issue because computing applications and data are growing so quickly that larger servers and disks are needed to process them fast enough within the required time period.
- Green cloud computing is intended to achieve not only efficient processing and utilization of computing infrastructure but also minimize energy consumption.
- Cloud computing, with increasingly universal front-end client devices such as iPhones interacting with back-end datacenters, will cause an enormous growth in energy usage.
- To address this problem, datacenter resources need to be managed in an energy-efficient manner to drive green cloud computing.
- In particular, cloud resources need to be allocated not only to satisfy QoS requirements specified by users via service-level agreements (SLAs) but also to reduce energy usage.
- This can be achieved by applying market-based utility models to accept user requests that can be fulfilled to enhance revenue along with energy-efficient utilization of cloud infrastructure.

Energy-efficient and green cloud computing architecture:

- A high-level architecture for supporting energy-efficient resource allocation in a green cloud computing infrastructure is shown in Figure 11.2.
- It consists of **four main components**:
- **Consumers/brokers:**
    - Cloud consumers or their brokers submit service requests from anywhere in the world to the cloud.
    - It is important to note that there can be a difference between cloud consumers and users of deployed services.

- For instance, <u>a consumer can be a company deploying a Web application, which presents varying workloads according to the number of “users” accessing it</u>.

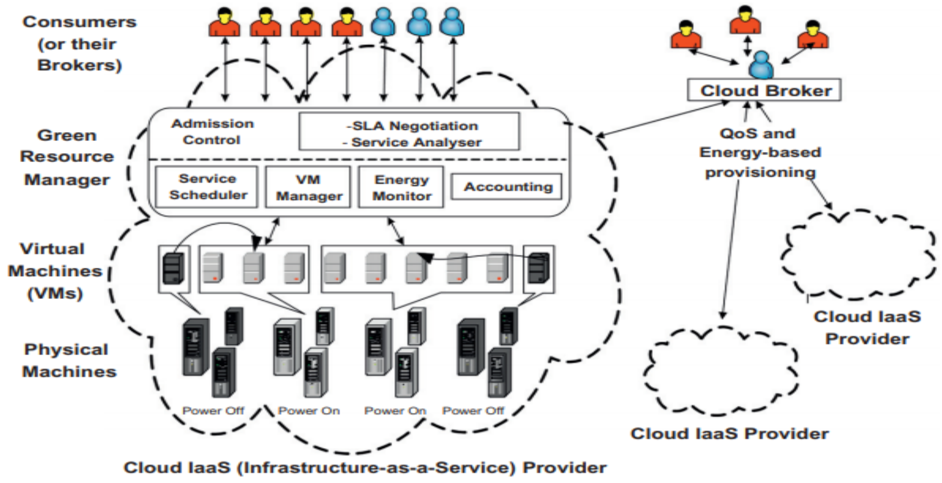

FIGURE 11.2
High-level system architectural framework for green cloud computing.

- **Green Resource Allocator:**
    - <u>Acts as the interface between the cloud infrastructure and consumers</u>.
    - It requires the interaction of the following components to support energy-efficient resource management.
        - **Green Negotiator.** <u>Negotiates with the consumers/brokers to finalize the SLAs</u> with specified prices and penalties between the cloud provider and the consumer, depending on the consumer’s QoS requirements and energy-saving schemes.
        - **Service Analyzer.** <u>Interprets and analyzes the service requirements of a submitted request</u> before deciding whether to accept or reject it. Hence, it needs the latest load and energy information from VM Manager and Energy Monitor, respectively.
        - **Consumer Profiler.** <u>Gathers specific characteristics of consumers</u> so that important consumers can be granted special privileges and prioritized over other consumers.
        - **Pricing.** <u>Decides how service requests are charged</u> to manage the supply and demand of computing resources and facilitate prioritizing service allocations effectively.
        - **Energy Monitor.** Observes and determines <u>which physical machines to power on or off</u>.
        - **Service Scheduler.** <u>Assigns requests to VMs and determines resource entitlements for allocated VMs</u>. It also decides when VMs are to be added or removed to meet demand.
        - **VM Manager.** <u>Keeps track of the availability of VMs and their resource entitlements</u>. It is also in charge of migrating VMs across physical machines.

- **Accounting.** Maintains the actual usage of resources by requests to compute usage costs. Historical usage information can also be used to improve service allocation decisions.
- **VMs.** Multiple VMs can be dynamically started and stopped on a single physical machine to meet accepted requests. Multiple VMs can also run concurrently applications and dynamically migrating VMs across physical machines.
- **Physical machines.** The underlying physical computing servers provide hardware infrastructure for creating virtualized resources to meet service demands.

Market-based management of clouds:

- Cloud computing is still in its beginning, and its prominent use is twofold:
  - Complete replacement of in-house IT infrastructure and services with the same capabilities rented by service providers; and
  - Elastic scaling of existing computing systems in order to address peak workloads.
- The efforts in research and industry have been mostly oriented to design and implement systems that actually enable business vendors and enterprises to achieve these goals.
- The real potential of cloud computing resides in the fact that it actually facilitates the establishment of a market for trading IT utilities.
- This opportunity until now has been mildly explored and falls in the domain of what it is called market-oriented cloud computing.

Market-oriented cloud computing (MOOC):

- Cloud computing already embodies the concept of providing IT assets as utilities.
- Then, what makes cloud computing different from market-oriented cloud computing?
  - First, it is important to understand what we intend by the term market. The Oxford English Dictionary (OED) defines a market as a "place where a trade is conducted".
  - More precisely, market refers to a meeting or a gathering together of people for the purchase and sale of goods.
  - A broader definition is as the action of buying and selling, a commercial transaction, a purchase.
- Therefore, essentially the word market is the act of trading mostly performed in an environment—either physical or virtual—that is specifically dedicated to such activity.

- If we consider the way, IT assets and services are consumed as utilities, it is evident that there is a trade-off between the service provider and the consumer; this enables the use of the service by the user under a given SLA.
- Therefore, cloud computing already expresses the concept of trade, even though the interaction between consumer and provider is not as sophisticated as happens in real markets:
  - Users generally select one cloud computing vendor from among a group of competing providers and leverage its services as long as they need them.
- Moreover, at present, most service providers have inflexible pricing, generally limited to flat rates or tariffs based on usage thresholds.
- In addition, many providers have exclusive interfaces to their services, thus restricting the ability of consumers to quickly move—and with minimal conversion costs—from one vendor to another.
- This inflexibility, known as vendor lock-in, undermines the potential of cloud computing to be an open market where services are freely traded.
- Therefore, to remove such restrictions, it is required that vendors expose services through standard interfaces.
- What differentiates market-oriented cloud computing (MOCC) from cloud computing is the presence of a virtual marketplace where IT services are traded and brokered dynamically.
- More precisely, what is missing is the availability of a market where desired services are published and then automatically bid on by matching the requirements of customers and providers.
- Moreover, the presence of a demand-based marketplace represents an opportunity for enterprises to shape their infrastructure for dynamically reacting to workload spikes and for cutting maintenance costs.
- It also allows the possibility to temporarily lease some in-house capacity during low usage periods, thus giving a better return on investment.
- These developments will lead to the complete realization of market-oriented cloud computing.

A reference model for MOCC:

- Market-oriented cloud computing originated from the coordination of several components:
  - Service consumers, service providers, and other entities that make trading between these two groups possible.
- Market orientation not only influences the organization on the global scale of the cloud computing market.

- It also shapes the internal architecture of cloud computing providers that need to support a more flexible allocation of their resources, which is driven by additional parameters such as those defining the quality of service.
- A reference scenario that realizes MOCC at a global scale is given in Figure 11.3.

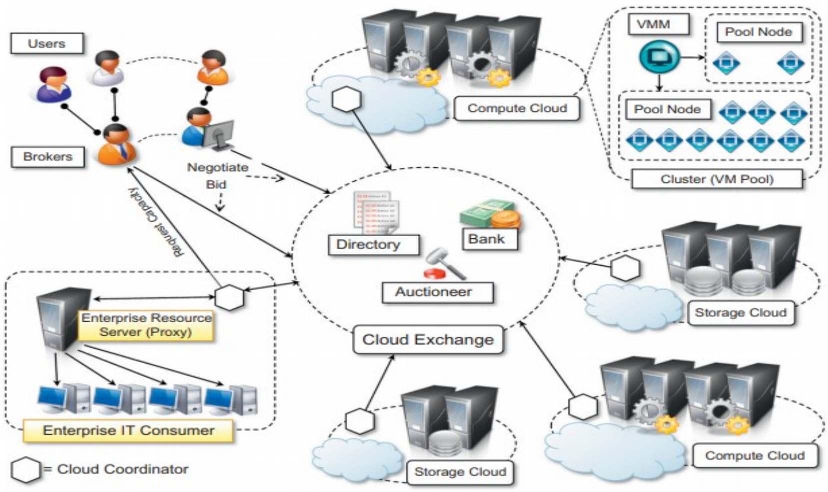

FIGURE 11.3
Market-oriented cloud computing scenario.

- It provides guidance on how MOCC can be implemented in practice, several components and entities contribute to the definition of a global market-oriented architecture.
- The fundamental component is the virtual marketplace—represented by the Cloud Exchange (CEx)—which acts as a market maker, bringing service producers and consumers together.
- The principal players in the virtual marketplace are the cloud coordinators and the cloud brokers.
  - The cloud coordinators represent the cloud vendors and publish the services that vendors offer.
  - The cloud brokers operate on behalf of the consumers and identify the subset of services that match customers requirements in terms of service profiles and quality of service.
- Brokers perform the same function as they would in the real world:
  - They mediate between coordinators and consumers by acquiring services from the first and subleasing them to the latter.
  - Brokers can accept requests from many users. At the same time, users can leverage different brokers.
- A similar relationship can be considered between coordinators and cloud computing services vendors.
  - Coordinators take responsibility for publishing and advertising services on behalf of vendors and can gain benefits from reselling services to brokers.

- Negotiations and trades are carried out in a secure and dependable environment and are mostly driven by SLAs, which each party has to fulfill.
- There might be different models for negotiation among entities, even though the auction model seems to be the more appropriate in the current scenario.
- The same consideration can be made for the pricing models:
- Prices can be fixed, but it is expected that they will most likely change according to market conditions.
- Several components contribute to the realization of the Cloud Exchange and implement its features. In the reference model depicted in Figure 11.3, it is possible to identify three major components……

1. **Directory.**
   - The market directory contains a listing of all the published services that are available in the cloud marketplace.
   - The directory not only contains a simple mapping between service names and the corresponding vendor (or cloud coordinators) offering them.
   - It also provides additional metadata that can help the brokers or the end users in filtering from among the services of interest those that can really meet the expected quality of service.
   - This component is modified in its content by service providers and queried by service consumers.
2. **Auctioneer.**
   - The auctioneer is in charge of keeping track of the running auctions in the marketplace and of verifying that the auctions for services are properly conducted.
   - And that malicious market players are prevented from performing illegal activities.
3. **Bank.**
   - The bank is the component that takes care of the financial aspect of all the operations happening in the virtual marketplace.
   - It also ensures that all the financial transactions are carried out in a secure and dependable environment.
   - Consumers and providers may register with the bank and have one or multiple accounts that can be used to perform the transactions in the virtual marketplace.

Technologies and initiatives supporting MOCC:

- Existing cloud computing solutions have very limited support for market-oriented strategies to deliver services to customers.
- Most current solutions mainly focused on enabling cloud computing concern the delivery of infrastructure, distributed runtime environments, and services.
- Since cloud computing has been recently adopted, the merging of the technology constitutes the first step toward the full realization of its promise.
- Until now, a good deal of interest has been directed toward IaaS, with several different players and competitive vendors new PaaS solutions are gaining momentum like google, Microsoft etc.,

Framework for trading computing utilities:

- A considerable amount of research has been carried out in defining models that enable the trading of computing utilities, with a specific focus on the design of market-oriented schedulers for grid computing systems.
- As discussed, computing grids are often leased for long-term use by means of agreements among these organizations.
- Within this context, market-oriented schedulers, which are aware of the price of a given computing resource and schedule user's applications according to their budgets, have been investigated and implemented.
- The research in relevance to MOCC, have provided a complete taxonomy and analysis of such schedulers, which is reported in Figure 11.5.

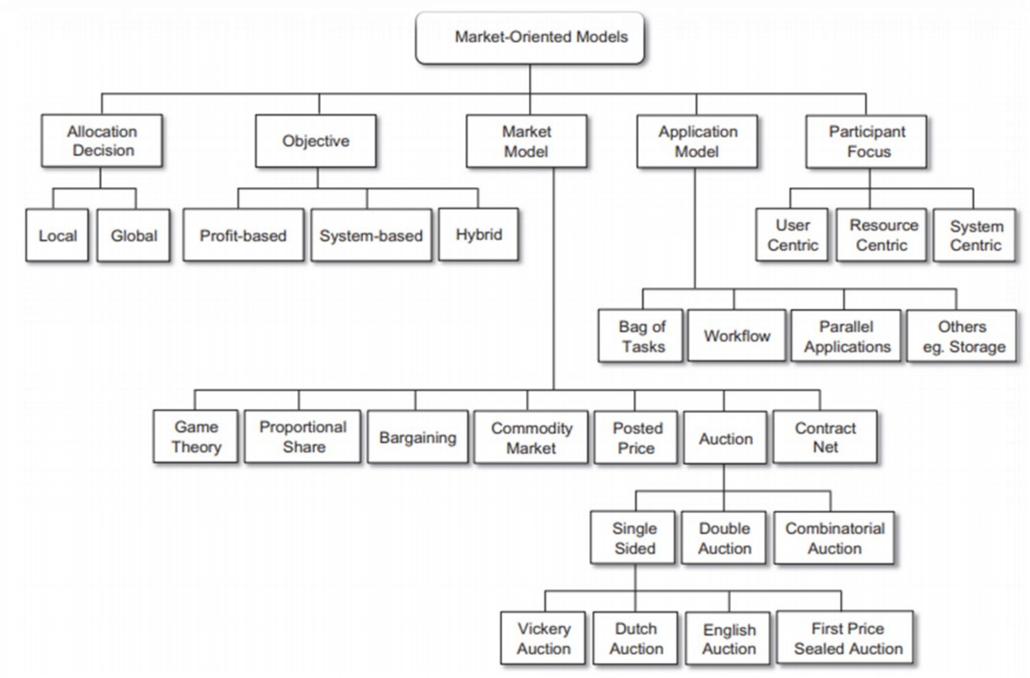

FIGURE 11.5 Market-oriented scheduler taxonomy.

- A major classification categorizes these schedulers according to allocation decision, objective, market model, application model, and participant focus.
- Of particular interest is the classification according to the market model, which is the mechanism used for trading between users and providers.
- Along this dimension, it is possible to classify the schedulers into the following categories……
- **Game theory.** In market models that are based on game theory, participants interact in the form of an allocation game, with different payoffs as a result of specific actions that employ various strategies.
- **Proportional share.** This market model originates from proportional share scheduling, which aims to allocate jobs fairly over a set of resources.
- **Commodity market.** In this model the resource provider specifies the price of resources and charges users according to the number of resources they consume. Moreover, prices might be subject to vary over time.
- **Posted price.** This model is similar to the commodity market, but the provider may make special offers and discounts to new clients. Furthermore, with respect to the commodity market, prices are fixed over time.
- **Contract-Net.** In market models based on the Contract-Net protocol, users advertise their demand and invite resource owners to submit bids.
    - Resource owners check these advertisements with respect to their requirements. If the advertisement is favorable to them, the providers will respond with a bid.
    - The user will then consolidate all the bids and compare them to select those most favorable to him.
- **Bargaining.** In market models based on bargaining, the negotiation among resource consumers and providers is carried out until a mutual agreement is reached or it is stopped when either of the parties is no longer interested.
- **Auction.** In market models based on auctions, the price of resources is unknown, and competitive bids regulated by a third party—the auctioneer—contribute to determining the final price of a resource.
    - The bid that ultimately sets the price of a resource is the winning bid, and the corresponding user gains access to the resource.
- The most popular and interesting market models for trading computing utilities are the commodity market, posted price, and auction model.

Federated clouds/InterCloud:

- In the previous section, we discussed how to define models and systems for trading computing resources as utilities and how cloud computing systems can interact with each other to make this trading happen.
- In this section, we address the same problem from the administrative and organizational points of view and introduce the concept of cloud federation and the InterCloud.
- These are enablers for MOCC since they provide means for interoperation among different cloud providers.
- Cloud computing strongly implies the presence of financial agreements between parties, since services are available on demand on a pay-per-use basis.

Characterization and definition:

- The terms cloud federation and InterCloud, convey the general meaning of an aggregation of cloud computing providers that have separate administrative domains.
- It is important to clarify what these two terms mean and how they apply to cloud computing.
    - The term federation implies the creation of an organization that overtakes the decisional and administrative power of the single entities and that acts as a whole.
    - Within a cloud computing context, the word federation implies that there are agreements between the various cloud providers, allowing them to leverage each other's services in a privileged manner.
- A definition of the term cloud federation was given by……
    - Cloud federation manages consistency and access controls when two or more independent geographically distinct Clouds share either authentication, files, computing resources, command and control or access to storage resources.
- InterCloud is a term that is often used interchangeably to express the concept of Cloud federation.
    - It was introduced by Cisco for expressing a composition of clouds that are interconnected by means of open standards to provide a universal environment that leverages cloud computing services.
- By mimicking the Internet term, often referred as the "network of networks," InterCloud represents a "Cloud of Clouds" and therefore expresses the same concept of federating together clouds that belong to different administrative organizations.
- Therefore, the term InterCloud refers mostly to a global vision in which interoperability among different cloud providers is governed by standards, thus creating an open platform.

- On the other hand, the concept of a cloud federation is more general and includes ad-hoc aggregations between cloud providers on the basis of private agreements and proprietary interfaces.

Cloud federation stack:

- Creating a cloud federation involves research and development at different levels: conceptual, logical and operational, and infrastructural.
- Figure 11.7 provides a comprehensive view of the challenges faced in designing and implementing an organizational structure.

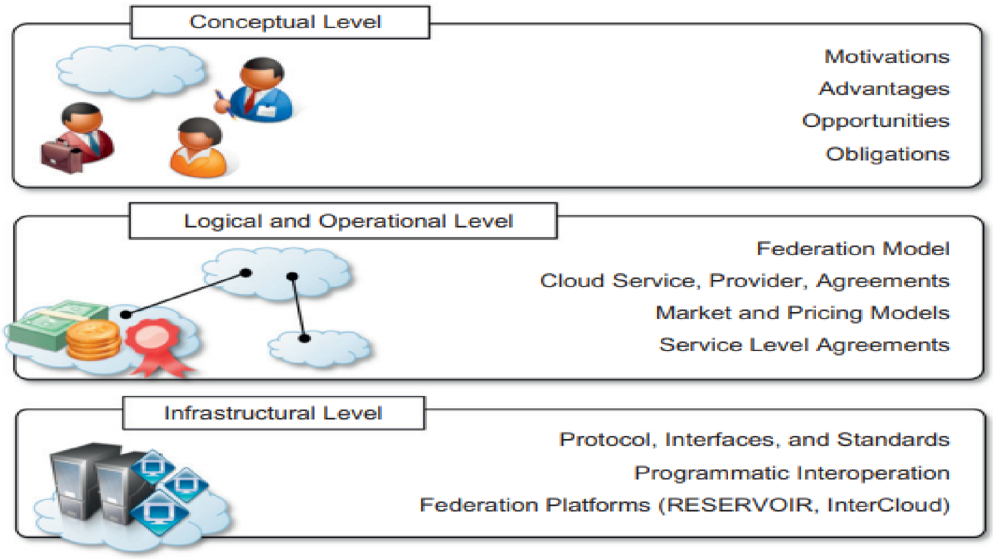

FIGURE 11.7 Cloud federation reference stack.

- This structure coordinates together cloud services that belong to different administrative domains and makes them operate within a context of a single unified service middleware.
  - Each cloud federation level presents different challenges and operates at a different layer of the IT stack.
  - It then requires the use of different approaches and technologies.
- Taken together, the solutions to the challenges faced at each of these levels constitute a reference model for a cloud federation.
- **Conceptual level:**
  - The conceptual level addresses the challenges as a favorable solution with respect to the use of services leased by single cloud providers.
  - In this level it is important to clearly identify the advantages for either service providers or service consumers in joining a federation and to define the new opportunities that a federated environment creates with respect to the single-provider solution.
  - Elements of concern at this level are:
    - Motivations for cloud providers to join a federation.

- Motivations for service consumers to leverage a federation.
- Advantages for providers in leasing their services to other providers.
- Obligations of providers once they have joined the federation.
- Trust agreements between providers
- Transparency versus consumers.

- **Logical and operational level:**
    - This level identifies and addresses the challenges in devising a framework that enables the aggregation of providers that belong to different administrative domains within a context of a single overlay infrastructure, which is the cloud federation.
    - At this level, policies and rules for interoperation are defined.
    - Moreover, this is the layer at which decisions are made as to how and when to lease a service to or to leverage a service from – another provider.
    - The logical component defines a context in which agreements among providers are settled and services are negotiated, whereas the operational component characterizes and shapes the dynamic behavior of the federation as a result of the single providers' choices.
    - This is the level where MOCC is implemented and realized. It is important at this level to address the following challenges:
        - How should a federation be represented? How should we model and represent a cloud service, a cloud provider, or an agreement?
        - How should we define the rules and policies that allow providers to join a federation?
        - What are the mechanisms in place for settling agreements among providers?
        - What are providers responsibilities with respect to each other?
        - When should providers and consumers take advantage of the federation? Which kinds of services are more likely to be leased or bought?
        - How should we price resources that are leased, and which fraction of resources should we lease?
- **Infrastructural level:**
    - The infrastructural level addresses the technical challenges involved in enabling heterogeneous cloud computing systems to interoperate seamlessly.

- It deals with the technology barriers that keep separate cloud computing systems belonging to different administrative domains. By having standardized protocols and interfaces, these barriers can be overcome.
- In other words, this level for the federation is what the TCP/IP stack is for the Internet: a model and a reference implementation of the technologies enabling the interoperation of systems.
- The infrastructural level lays its foundations in the IaaS and PaaS layers of the Cloud Computing Reference Model.
- Services for interoperation and interface may also find implementation at the SaaS level, especially for the realization of negotiations and of federated clouds.
- At this level it is important to address the following issues……
  - What kind of standards should be used?
  - How should design interfaces and protocols be designed for interoperation?
  - Which are the technologies to use for interoperation?
  - How can we realize a software system, design platform components, and services enabling interoperability?
- Interoperation and composition among different cloud computing vendors is possible only by means of open standards and interfaces.
- Moreover, interfaces and protocols change considerably at each layer of the Cloud Computing Reference Model.

Aspects of Interest:

- Several aspects contribute to the successful realization of a cloud federation. Besides motivation and technical enablers, other elements should be considered.
- In particular, standards for interoperability, security, and legal issues have to been taken into consideration while defining a platform for interoperability among cloud vendors.

Standards:

- Standard's main role is to organize a platform for interoperation that goes beyond ad-hoc aggregations and private settlements between providers.
- <u>Standardized interfaces and protocols facilitate the realization of an open organization where providers can easily join.</u>
- The advantages are primarily technical; <u>standards facilitate the development of software and services that interconnect systems</u>.
- Interoperation between vendors has always been an element of concern for enterprises and one of the reasons that initially prevented them from fully embracing the cloud computing model.

- More specifically, the absence of common standards for developing applications and systems in a portable way initially developed the fear of vendor lock-in.
  - For example, IaaS solutions still leverage exclusive formats for virtual machine instances and templates.
- The technical barriers are even more considerable in the case of PaaS solutions, where even the development technology might differ.
  - Standards, for instance, is a solution that provides customers with a development platform that can be transparently deployed over different IaaS vendors.
  - For example, on the PaaS segment Aneka provides middleware that can connect heterogeneous hardware and support cross-platform deployment across IaaS vendors.
- These initiatives constitute technical improvements that help system designers and developers develop systems that are less subject to the vendor lock-in problem.
- The support required for building a federation, which can only be achieved by means of standards, protocols, and formats designed for interoperation.

Open cloud manifesto:

- The Open Cloud Manifesto constitutes the first step toward the realization of a cloud interoperability platform.
- The manifesto was drafted in 2009 as a result of the coordinated activity of different cloud vendors and currently lists more than 400 cloud computing services providers that support the vision it embodies.
- More than proposing standards, the manifesto is a declaration of intent, endorsed by commercial players in the field of cloud computing, to realize an interoperable and open cloud computing platform.
- The document is intended to be of guidance to the CIOs, governments, IT users, and business leaders to use cloud and to establish a set of core principles.
- The document enumerates the advantages of cloud computing and discusses the challenges and barriers to its adoption.
- It introduces the goals of an open cloud platform, which can be summarized as follows……
- **Choice:**
  - With the use of open technology, it will be possible for IT consumers to select the best provider, architecture, or usage model as the business environment changes.

- Furthermore, the use of open technologies simplifies the integration of cloud computing solutions provided by different vendors or with the existing infrastructure.

- **Flexibility:**
  - The change between one provider and another becomes easier if the different vendors do not use a closed proprietary technology.
- **Speed and Agility:**
  - The open technologies facilitate the integration of cloud computing solutions, thus realizing the promise of scaling demand with speed and agility.
- **Skills:**
  - Open technologies simplify the learning process and contribute to developing a common knowledge that can be used to design, develop, and deploy systems across multiple providers.
  - This simplifies the chances for organizations to find someone with the appropriate skills for their needs.
  - The manifesto ends with a set of recommendations for cloud computing vendors to pursue an open collaboration and an appropriate use of standards.

Technologies for cloud federations:

- Even though the concept of cloud federation or the InterCloud is still undeveloped, there are some supporting technologies that enable the deployment of interoperable clouds, at least from an operational point of view.
- These initiatives mostly originated within academia and principally focus on supporting interoperability among different IaaS implementations.

Reservoir:

- Resources and Services Virtualization Without Barriers, or RESERVOIR, is a European research project focused on developing an architecture that supports providers of cloud infrastructures to dynamically partner with each other to extend their capabilities while preserving their administrative autonomy.
- RESERVOIR defines a software stack enabling interoperation at the IaaS layer and providing support for SLA-based execution of applications on top of the infrastructure overlay those results from the federation of infrastructure providers.
- RESERVOIR is based on the concept of dynamic federation:
  - Each infrastructure provider is an autonomous business with its own business goals and that might decide to partner with other businesses when needed.

- The federation is obtained by means of the RESERVOIR middleware that needs to be deployed at each site.
- The IT management at a specific site is fully autonomous and dictated to by policies that depend on the site's business goals.

- When needed, internal IT resources are leased to other providers within the context of a negotiated SLA.
- The role of RESERVOIR is to arrange this process and to minimize the barriers obstructing interoperation among different administrative domains.
- Figure 11.10 provides a general overview of a RESERVOIR cloud.

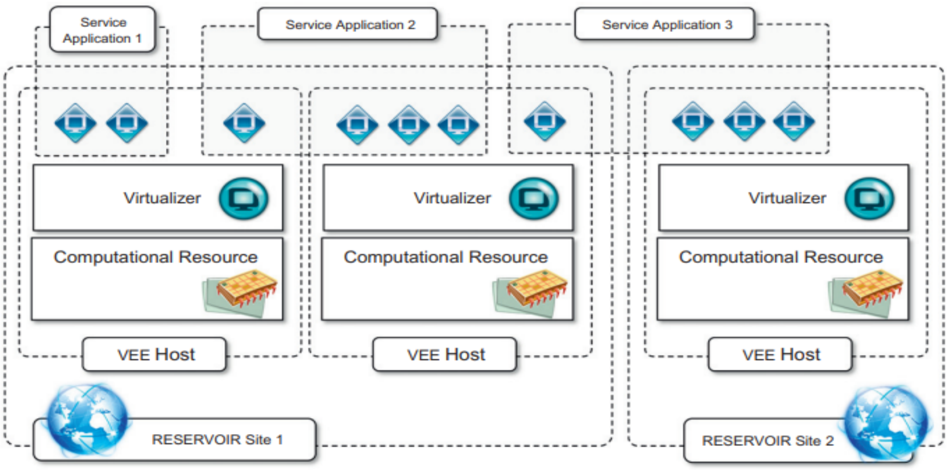

FIGURE 11.10
RESERVOIR cloud deployment.

- The framework defines an infrastructure overlay that spans multiple administrative domains and different geographic locations.
- Each site runs the RESERVOIR software stack and provides an on-demand execution environment in which components of a service application can be deployed and executed.
- The model introduces a clear separation between service providers and infrastructure providers.

- Service providers are the entities that understand the needs of a particular business and offer service applications to address those needs; they do not own infrastructure but rather lease it from infrastructure providers.
- Infrastructure providers operate RESERVOIR sites and offer a virtually infinite pool of computational, network, and storage resources.
- Service providers define service applications that are modeled as a collection of components that can be deployed over a distributed virtual infrastructure, which can be either explicitly or implicitly provisioned.

- In the first case, the service provider conducts sizing and capacity-planning studies to identify the appropriate number of components to be required for a given workload condition.
- The specification is obtained by means of minimal service configuration and a set of elasticity rules that are used by RESERVOIR to dynamically provision resources under varying workload conditions.
- In the second case, the service provider provides neither a minimal service configuration nor elasticity rules. The sizing is automatically made by the RESERVOIR middleware, which tries to minimize overprovisioning.
- Service providers are billed on a pay-as-you-go model and can ask for usage reports to verify the billing.

- Service application components are represented as virtual packages, which also specify execution environment requirements.
- Infrastructure providers can leverage any virtualization technology of choice and are in charge of providing the required environment.
- An important element for determining the virtual environment for service applications is the service manifest.
- This is one of the key elements of the RESERVOIR model and specifies the structure of the service applications in terms of component types that are to be deployed as virtual execution environments(VEEs).
  - The service manifest contains a reference to a master image, which is a self-contained software stack (OS, middleware, applications, data, and configuration) that fully captures the functionality of the component type.
  - Additional information and rules specify how many instances to deploy and how to dynamically grow and shrink their number.
  - Moreover, the manifest also specifies the grouping of components into virtual networks and/or tiers that form the service applications.
  - The manifest is expressed by extending the Open Virtualization Format to simplify interoperability and leverage an already existing popular standard.
- Figure 11.11 describes the internal architecture of a RESERVOIR site that enables the federated and SLA-based execution of applications.
- The RESERVOIR stack consists of three major components:

1. **Service Manager:**
   - The Service Manager is the highest level of abstraction and constitutes the front-end used by service providers to submit

service manifests, negotiate pricing, and monitor applications.

- This component deploys and provisions VEEs according to the service manifest and monitors and enforces SLA compliance by controlling the capacity of a service application.

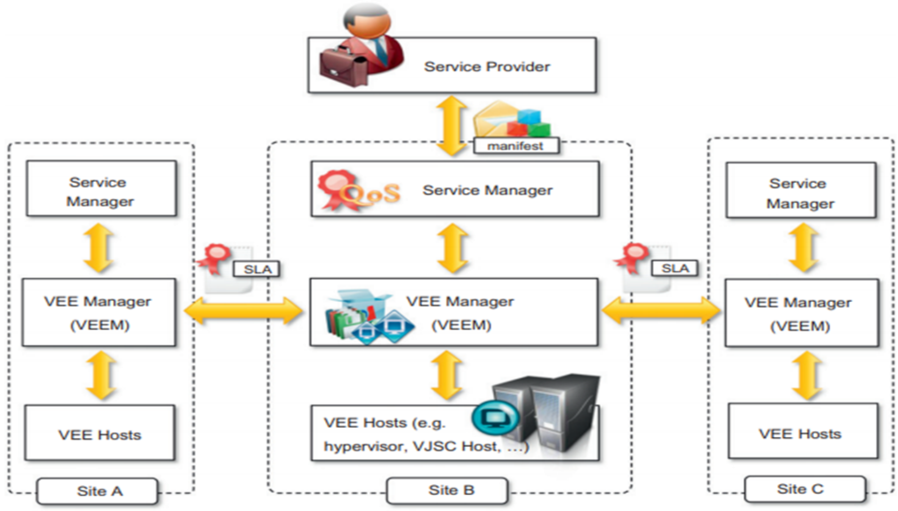

FIGURE 11.11
RESERVOIR architecture.

2. Virtual Execution Environment (VEE) Manager.

- This component is the core of the RESERVOIR middleware and is responsible for the optimal placement of VEEs into VEE hosts according to the constraints expressed by the Service Manager.
- Moreover, the VEE Manager also interacts with VEE Managers in other sites to provision additional instances for the execution of service applications or move VEEs to other sites in case of overload. This component realizes the cloud federation.

3. VEE Host (VEEH).

- This is the lowest level of abstraction and interacts with the VEE Manager to put into practice the IT management decisions regarding heterogeneous sets of virtualization platforms.
- This level is also in charge of ensuring appropriate and isolated networking among VEEs that belong to the same application.
- The VEEH encapsulates all platform-specific management that is required to expose the used virtualization technology through a standardized interface to the VEE Manager.

- Each of these components adopts a standardized interface and protocol to interoperate with the adjacent layer.

InterCloud:

- InterCloud is a service-oriented architectural framework for cloud federation that supports utility driven interconnection of clouds.

- It is composed of a set of decoupled elements that interact via a market-oriented system to enable trading of cloud assets such as computing power, storage, and execution of applications.
- As depicted in Figure 11.12, the InterCloud model comprises two main elements: CloudExchange and CloudCoordinator:

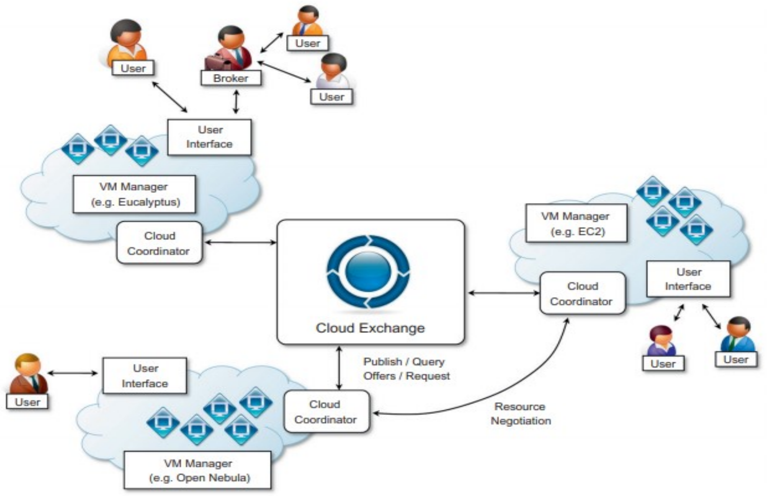

FIGURE 11.12
InterCloud architecture.

- **CloudExchange.**
  - This is the market-making component of the architecture. It offers services that allow providers to find each other in order to directly trade cloud assets, as well as allowing parties to register and run auctions.
  - In the former case, CloudExchange acts as a directory service for the federation. In the latter case, it runs the auction.
  - For offering such services to the federation, CloudExchange implements a Web service-based interface that allows many services such as
    - Datacenters to join and leave the federation;
    - To publish resources, they want to sell;
    - To register their resource requirements so that parties interested in selling providers are able to locate potential buyers for their resources. etc.,
- **CloudCoordinator:**
  - This component manages domain-specific issues related to the federation. This component is present on each party that wants join the federation.
  - CloudCoordinator has front-end components (i.e., elements that interact with the federation) as well as back-end components (i.e., components that interact with the associated datacenter).
  - Front-end components interact with the CloudExchange and with other coordinators.

- The former allows datacenters to announce their offers and requirements, whereas the latter allows the Coordinator to learn about the current state of the datacenter to decide whether actions from the federation are required or not.

- Therefore, wherever the Coordinator detects that extra resources are required by the datacenter, it triggers the process of discovery of potential providers (by interacting with the cloud federation).
- Once potential providers are discovered and the preferred one is selected, the Coordinator contacts the remote Coordinator and negotiates.
- Similarly, when the Coordinator detects that local resources are underutilized, they can publish an offer for resources in the CloudExchange or they can look for matches among requirements registered in the Exchange service.

- Negotiation between parties follows the Alternate Offers protocol.
- The value that each Coordinator is willing to pay for a resource, as well as the value for which each Coordinator wants to sell resources, is not defined by the federation;
- Instead, each Coordinator is free to value resources according to utilization and profit estimation, and they are also free to reject offers that are not attractive for them.
- Moreover, <u>InterCloud acts primarily in the operational layer, means that issues related to security, VM images, and networking are not handled by the framework</u>.

Third-party cloud services:

- One of the key elements of cloud computing is the possibility of <u>composing services that belong to different vendors or integrating them into existing software systems</u>.
- The <u>service-oriented model</u>, facilitates such an approach and provides the opportunity for developing a new class of services that can be called third-party cloud services.
- These are the <u>result of adding value to preexisting cloud computing services</u>, thus providing customers with a different and more sophisticated service.
- Added value can be either created by smartly coordinating existing services or implementing additional features on top of an existing service.

MetaCDN:

- <u>MetaCDN provides users with a Content Delivery Network (CDN) service by leveraging and connecting together heterogeneous storage clouds.</u>

- It implements a software overlay that coordinates the service offerings of different cloud storage vendors and uses them as distributed elastic storage on which the user content is stored.
- MetaCDN provides users with the high-level services of a CDN for content distribution and interacts with the low-level interfaces of storage clouds to optimally place the user content in accordance with the expected geography of its demand.
- By leveraging the cloud as a storage back-end it makes a complex—and generally expensive—content delivery service available to small enterprises.
- The architecture of MetaCDN is shown in Figure 11.13.

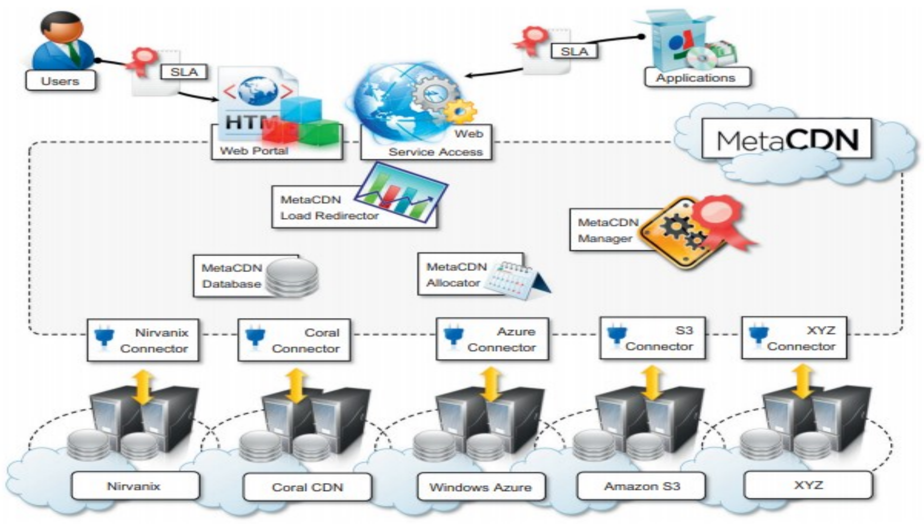

FIGURE 11.13
MetaCDN architecture.

- The MetaCDN interface exposes its services to users and applications through the Web; users interact with a portal, while applications take advantage of the programmatic access provided by means of Web services.
- The main operations of MetaCDN are the creation of deployments over storage clouds and their management.
- The portal constitutes a more spontaneous interface for users with basic requirements, while the Web service provides access to the full capabilities.
- In particular, four different deployment options can be selected……
  - **Coverage and performance-optimized deployment.** In this case MetaCDN will deploy as many replicas as possible to all available locations.
  - **Direct deployment:** In this case MetaCDN allows the selection of the deployment regions for the content and will match the selected regions with the supported providers serving those areas.
  - **Cost-optimized deployment:** In this case MetaCDN deploys as many replicas in the locations identified by the deployment

request. The available storage transfer allowance and budget will be used to deploy the replicas and keep them active for as long as possible.

  - **QoS optimized deployment:** In this case MetaCDN selects the providers that can better match the QoS requirements attached to the deployment, such as average response time and throughput from a particular location.

- A collection of components coordinates their activities in order to offer the services, which constitute the additional value that MetaCDN brings on top of the direct use of storage clouds by the users.
- Of particular important are three components: the MetaCDN Manager, the MetaCDN QoS Monitor, and the Load Redirector.
  - The Manager is responsible for ensuring that all the content deployments are meeting the expected QoS. It is supported in this activity by the Monitor,
  - The Moniotor constantly probes storage providers and monitors data transfers to assess the performance of each provider.
  - Content serving is controlled by the Load Redirector, which is in charge of redirecting user content requests to the most suitable replica given the condition of the systems and the options specified during the deployment.
- Interaction with storage clouds is managed by means of connectors, which abstract away the different interfaces exposed by the providers and present a uniform interface within the MetaCDN system.

Cloud Applications:

- Cloud computing has gained huge popularity in industry due to its ability to host applications for which the services can be delivered to consumers rapidly at minimal cost.
- Here we discuss some application case studies, detailing their architecture and how they leveraged various cloud technologies.

Scientific applications:

- Scientific applications are a sector that is increasingly using cloud computing systems and technologies.
- The immediate benefit seen by researchers and academics is the potentially infinite availability of computing resources and storage.
- Cloud computing systems meet the needs of different types of applications in the scientific domain:
  - High-performance computing (HPC) applications, high-throughput computing (HTC) applications, and data-intensive applications.

- The opportunity to use cloud resources is even more appealing because minimal changes need to be made to existing applications in order to leverage cloud resources.
- The most relevant option is IaaS solutions, which offer the optimal environment for running bag-of-tasks applications and workflows.
- Virtual machine instances are suitably customized to host the required software stack for running such applications.
- PaaS solutions allow scientists to explore new programming models for tackling computationally challenging problems.
- Applications have been redesigned and implemented on top of cloud programming application models and platforms to leverage their unique capabilities.
  - For instance, the MapReduce programming model provides scientists with a very simple and effective model for building applications that need to process large datasets.
  - Problems that require a higher degree of flexibility in terms of structuring of their computation model can leverage platforms such as Aneka, which supports MapReduce and other programming models.

Healthcare: ECG analysis in the cloud:

- Healthcare is a domain in which computer technology has found several and diverse applications: from supporting the business functions to assisting scientists in developing solutions to cure diseases.
- An important application is the use of cloud technologies to support doctors in providing more effective diagnostic processes.
- In particular, here we discuss electrocardiogram (ECG) data analysis on the cloud.
- The capillary development of Internet connectivity and its accessibility from any device at any time has made cloud technologies an attractive option for developing health-monitoring systems.
- ECG data analysis and monitoring constitute a case that naturally fits into this scenario.
  - ECG is the electrical manifestation of the contractile activity of the heart’s myocardium.
  - This activity produces a specific waveform that is repeated over time and that represents the heartbeat.
  - The analysis of the shape of the ECG waveform is used to identify arrhythmias and is the most common way to detect heart disease.

- Cloud computing technologies allow the remote monitoring of a patient's heartbeat data, data analysis in minimal time, and the notification of first-aid personnel and doctors should these data reveal potentially dangerous conditions.
- This way a patient at risk can be constantly monitored without going to a hospital for ECG analysis.

- An illustration of the infrastructure and model for supporting remote ECG monitoring is as in Figure 10.1.

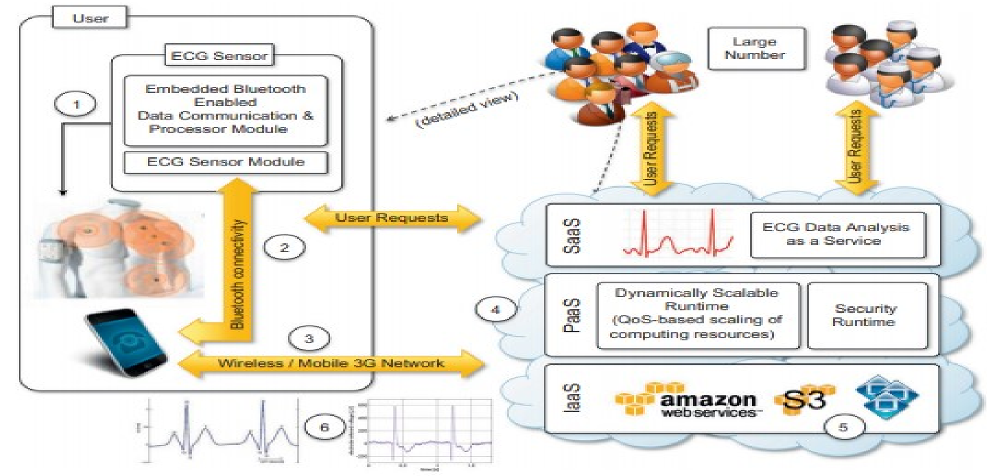

FIGURE 10.1
An online health monitoring system hosted in the cloud.

- Wearable computing devices equipped with ECG sensors constantly monitor the patient's heartbeat.
- Such information is transmitted to the patient's mobile device, which will eventually forward it to the cloud-hosted Web service for analysis.
- The Web service forms the front-end of a platform that is entirely hosted in the cloud and that leverages the three layers of the cloud computing stack: SaaS, PaaS, and IaaS.
- The Web service constitute the SaaS application that will store ECG data in the Amazon S3 service and issue a processing request to the scalable cloud platform.
- The runtime platform is composed of a dynamically sizable number of instances running the workflow engine and Aneka.
- The number of workflow engine instances is controlled according to the number of requests in the queue of each instance.
- The Aneka controls the number of EC2 instances used to execute the single tasks defined by the workflow engine for a single ECG processing job.
- Each of these jobs consists of a set of operations involving the extraction of the waveform from the heartbeat data and the comparison of the waveform with a reference waveform to detect anomalies.

- If anomalies are found, doctors and first-aid personnel can be notified to act on a specific patient.

- Even though remote ECG monitoring does not necessarily require cloud technologies, cloud computing introduces opportunities that would be otherwise hardly achievable.
- The first advantage is the elasticity of the cloud infrastructure that can grow and shrink according to the requests served.
  - As a result, doctors and hospitals do not have to invest in large computing infrastructures designed after capacity planning, thus making more effective use of budgets.
- The second advantage is ubiquity.
  - Cloud computing technologies have now become easily accessible and promise to deliver systems with minimum or no downtime.
  - Computing systems hosted in the cloud are accessible from any Internet device through simple interfaces.
- Finally, cost savings constitute another reason for the use of cloud technology in healthcare.
  - Cloud services are priced on a pay-per-use basis and with volume prices for large numbers of service requests.
- The charging costs based on effective use rather than capital costs.

Business and consumer applications:

- The business and consumer sector is the one that probably benefits the most from cloud computing technologies.
  - On one hand, the opportunity to transform capital costs into operational costs makes clouds an attractive option for all enterprises that are IT-centric.
  - On the other hand, the sense of ubiquity that the cloud offers for accessing data and services makes it interesting for end users as well.
- Moreover, the elastic nature of cloud technologies does not require huge up-front investments, thus allowing new ideas to be quickly translated into products and services.
- The combination of all these elements has made cloud computing the preferred technology for a wide range of applications, from CRM and ERP systems to productivity and social-networking applications.

1. **CRM and ERP:**

   - Customer relationship management (CRM) and enterprise resource planning (ERP) applications are market segments that are flourishing in the cloud.

- Cloud CRM applications constitute a great opportunity for small enterprises and start-ups to have fully functional CRM software without large up-front costs and by paying subscriptions.
- Moreover, CRM is not an activity that requires specific needs, and it can be easily moved to the cloud.
- Such a characteristic, together with the possibility of having access to your business and customer data from everywhere and from any device, has fostered the spread of cloud CRM applications.
- ERP solutions on the cloud are less mature and have to compete with well-established in-house solutions.
    - ERP systems integrate several aspects of an enterprise: finance and accounting, human resources, manufacturing, supply chain management, project management, and CRM
- Their goal is to provide a uniform view and access to all operations that need to be performed to sustain a complex organization.

2. Salesforce.com:

- Salesforce.com is probably the most popular and developed CRM solution available today. As of today more than 100,000 customers have chosen Safesforce.com to implement their CRM solutions.
- The application provides customizable CRM solutions that can be integrated with additional features developed by third parties.
- Salesforce.com is based on the Force.com cloud development platform, which represents scalable and high-performance middleware executing all the operations of all Salesforce.com applications.
- The architecture of the Force.com platform is shown in Figure 10.5.

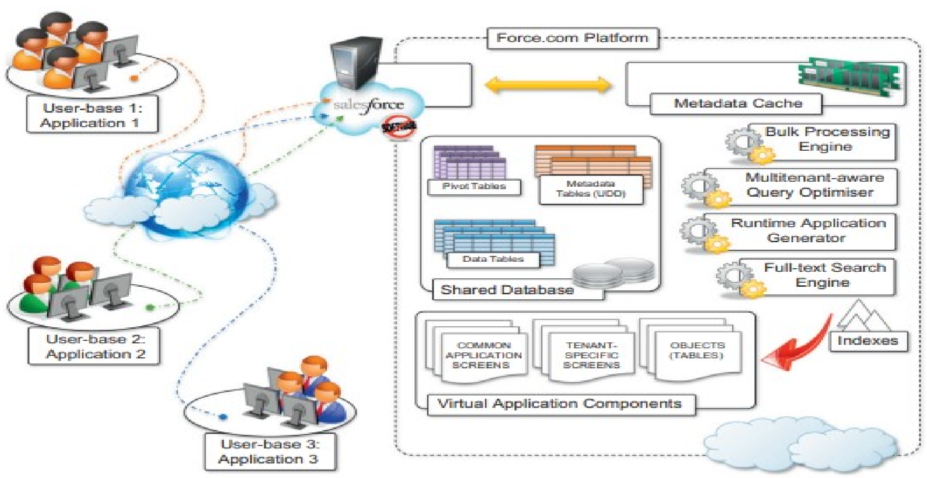

FIGURE 10.5
Salesforce.com and Force.com architecture.

- Initially designed to support scalable CRM applications, the platform has evolved to support the entire life cycle of a wider range of cloud applications by implementing a flexible and scalable infrastructure.
- At the core of the platform resides its metadata architecture, which provides the system with flexibility and scalability.
- Rather than being built on top of specific components and tables, application core logic and business rules are saved as metadata into the Force.com store.
- Both application structure and application data are stored in the store.
    - A runtime engine executes application logic by retrieving its metadata & then performing the operations on the data.
- Although running in isolated containers, different applications logically share the same database structure, and the runtime engine executes all of them uniformly.
- A full-text search engine supports the runtime engine, which allows application users to have an effective user experience despite the large amounts of data that need to be crawled.
- The search engine maintains its indexing data in a separate store and is constantly updated by background processes triggered by user interaction.
- Users can customize their application by leveraging the "native" Force.com application framework or by using programmatic APIs in the most popular programming languages.
- The application framework allows users to visually define either the data or the core structure of a Force.com application.
- Customization of application processes and logic can also be implemented by developing scripts in APEX.
    - APEX is a Java-like language that provides object-oriented and procedural capabilities for defining either scripts executed on demand or triggers.
- APEX also offers the capability of expressing searches and queries to have complete access to the data managed by the Force.com platform.

References:

- Cloud Computing - Concepts, Technology, Security, and Architecture - Second Edition - Thomas Erl, Eric Barcelo - Pearson
- Mastering Cloud Computing: Foundations and Applications Programming – Rajkumar Buyya – Tata Mcgraw Hill publishing.

Sample Questions:

1. Explain the architecture for supporting energy-efficient and green cloud computing infrastructure.
2. Explain the need and role of Market Oriented Cloud Computing (MOOC) with its reference model.
3. Explain the categories of schedulers based on classification according to the market model.
4. Explain the levels constitute a reference model for a cloud federation.
5. Explain the RESERVOIR technology used for Cloud Federation with its relevant diagram.
6. Explain the internal architecture of a RESERVOIR site with its main components.
7. InterCloud is a service-oriented architectural framework for cloud federation, justify the same with its elements.
8. Explain the role of Meta CDN with its architecture.
9. Explain the benefit of Cloud computing in context of remote ECG monitoring.
10. Discuss how Cloud computing is beneficial in context of CRM and ERP as applications.

****