CHAPTER 3

CLOUD AS A SERVICE

3.1 INTRODUCTION

In today's economy, many businesses are faced with the challenge of 'taking cost out' of their infrastructure while continuing to deliver new, innovative business services – basically they need to 'do more with less'. These days, many businesses face the necessities of a fast change to their IT infrastructure to manage requests during peak times. Organizations are dealing with IT resource optimization and lowering cost, and are looking for a way to manage these resources to meet such requirement. Also, they are trying to add rental-style capability to IT resource usage.

Cloud computing defines a new way to manage IT resources enabling self-service provisioning of IT resources, metering-style accounting based on use/time, automation of IT management in a standard process environment.

Cloud computing is a user experience and a business model. It is an emerging style of computing in which applications, data, and IT resources are provided as services to users over the network. Cloud computing is also an infrastructure management methodology. It is a way of managing large numbers of highly virtualized resources such that, from a management perspective, they resemble a single large resource, which can then be used to deliver services.

This chapter visualizes the different cloud models with respect to services. It also takes into account what service is all about and the different type of infrastructure services that can be offered as cloud as a service.

Common attributes of a cloud infrastructure are defined as:

- **Flexible pricing**, which means utility pricing, variable payments, pay-by-consumption; subscription models make pricing of IT services more flexible.
- **Elastic scaling**, which means resources scale up and down by large factors as the demand changes.
- **Rapid provisioning**, which means IT and network capacity and capabilities are – ideally automatically – rapidly provisioned using Internet standards without transferring ownership of resources.
- **Standardized offerings**, which mean uniform offerings readily available from a services catalogue on a metered basis.

There are two primary levers to achieve cost optimization – operating expense (Op-ex) and capital expense (Cap-ex) – and for many businesses, it is not just a question of lowering costs. It is also important to strike the right balance between Op-ex and Cap-Ex.

When you look at the different cloud types, the common terminology that comes up is 'as a service', with infrastructure as a service being the most basic type of service. It includes compute power, storage, and file systems as a service. At the next level, you have platform as a service, where a compute platform or middleware is provided.

At the next level, this is software as a service, and this is where you take software capability that would typically be in a package – like customer relationship management or e-commerce – and you deliver that as a service. At the highest level is business process as a service. This is where a business can take a function that it considers to be a commodity, and not a differentiator, and just completely outsource it and buy it as a service. So cloud computing has really taken a hold because of the fact that you take virtualization, standardization, and automation, and drive this automated delivery of services at a reduced cost.

Cloud computing introduces the concept of 'IT-as-a-Service'. To support this service, the cloud infrastructure must deliver:

- **Abstraction:** Alleviate IT consumers from the operations of applications, allowing end-users to focus instead on the execution of high-value activities.
- **Virtualization:** Access to business services on-demand independent of location and resource constraints.
- **Dynamic allocation:** Dynamically provisions, configures, reconfigures, and de-provisions IT capability as and when needed, transparently and seamlessly.
- **Data management:** Fast, secure, reliable data access, and mobility, with integrated data protection and recovery management.

Because all data reside on the same shared storage systems, effective and efficient data and storage management become critical in a cloud deployment.

complete range

3.2 GAMUT OF CLOUD SOLUTIONS

Even within the cloud computing space, there is a spectrum of offering types. There are five commonly used categories.

- **Platform-as-a-Service (PaaS):** This is the provisioning of hardware and OS, frameworks and database, for which developers write custom applications. There will be restrictions on the type of software they can write, offset by built-in application scalability.
- **Software-as-a-Service (SaaS):** This is the provisioning of hardware, OS, and special-purpose software made available through the Internet.
- **Infrastructure-as-a-Service (IaaS):** This is the provisioning of hardware or virtual computers where the organization has control over the OS, thereby allowing the execution of arbitrary software.
- **Storage-as-a-Service (SaaS):** This is the provisioning of database-like services, billed on a utility computing basis, for example, per gigabyte per month.
- **Desktop-as-a-Service (DaaS):** This is the provisioning of the desktop environment either within a browser or as a terminal server.

The distinction between the five categories of cloud offering is not necessarily clear-cut. In particular, the transition from Infrastructure-as-a-Service to Platform-as-a-Service is a very gradual one.

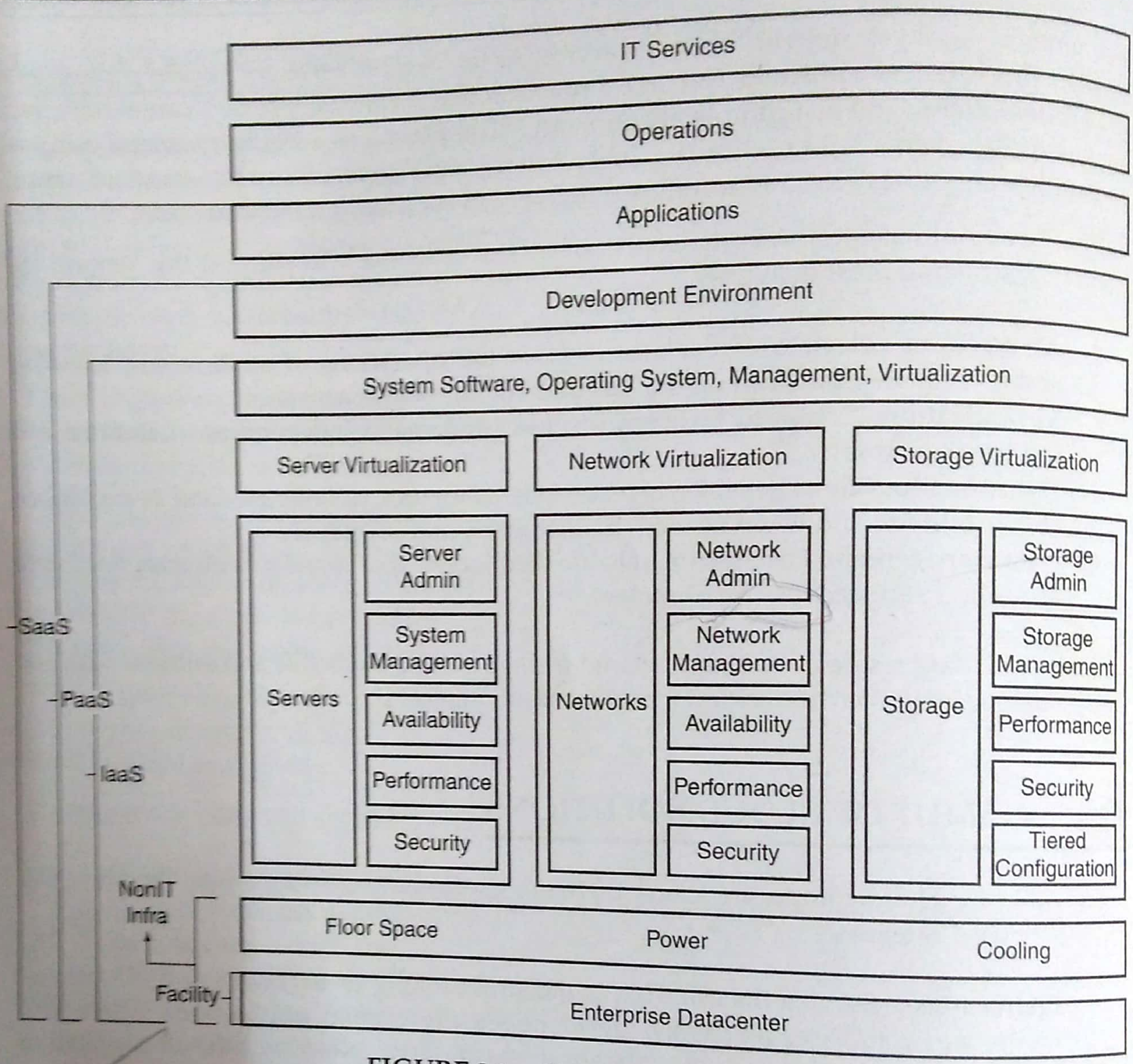

FIGURE 3.1 Cloud@datacenter.

3.2.1 *Platform-as-a-Service*

Instead of just offering applications over the Web in the form of Software-as-a-Service (SaaS), PaaS public cloud players are actually offering an entire Platform-as-a-Service (PaaS). They provide the foundation to build highly scalable and robust Web-based applications in the same way that the traditional operating systems like Windows and Linux have done in the past for software developers. What is very different about this model is that no longer is the platform itself 'sold' to the customer who is then responsible for running and maintaining it. In this model, it is this very operational capability of the platform hosting that is of primary value here (and that is how such platforms are typically billed). This has far-reaching implications to both the business models of PaaS vendors as well as their customers. One can use private clouds to speed application deployments for fast deployment in minutes. It will help tracking usage for the chargeback and will give the option of cost effective and secure appliance.

In order to optimize deployments, many organizations are looking to extend SOA to cloud services.

Cloud capabilities can improve the productivity of your development and test teams to roll out new applications and SOA services faster and reduce application backlog. It provides a catalogue of virtual images, and patterns all ready for immediate use. Patterns define a cluster of servers working together.

PaaS saves costs by reducing upfront software licensing and infrastructure costs, and by reducing ongoing operational costs for development, testing, and hosting environments.

PaaS significantly improves development productivity by removing the challenges of integration with services such as database, middleware, web frameworks, security, and virtualization. Software development and delivery times are shortened since software development and testing are performed on a single PaaS platform. There is no need to maintain separate development and test environments.

PaaS fosters collaboration among developers and also simplifies software project management. This is especially beneficial to enterprises that have outsourced their software development.

There is a challenge for tight binding of the applications with the platform which makes portability across vendors extremely difficult. PaaS in general is still maturing, and the full benefits of componentization and collaboration between services is still to be demonstrated. PaaS offerings lack the functionality needed for converting legacy applications into full fledged cloud services.

SaaS, PaaS, and IaaS suit different target audiences. SaaS is intended to simplify the provision of specific business services. PaaS provides a software development environment that enables rapid deployment of new applications. IaaS provides a managed environment into which existing applications and services can be migrated to reduce operational costs.

3.2.2 *Software-as-a-Service*

SaaS saves costs by removing the effort of development, maintenance, and delivery of software; eliminating up-front software licensing and infrastructure costs; and reducing ongoing operational costs for support, maintenance, and administration.

The time to build and deploy a new service is much shorter than for traditional software development. By transferring the management and software support to a vendor, internal IT staff can focus more on higher-value activities.

Applications that require extensive customization are not good candidates for SaaS. Typically, this includes most complex core business applications that will not be the best suit for SaaS.

There are also issues involved in moving to SaaS. Moving applications to the Internet cloud might require upgrades to the local network infrastructure to handle an increase in network bandwidth usage. Normally only one version of the software platform will be provided. Therefore, businesses are obliged to upgrade to the latest software versions on the vendor's schedule. This could introduce compatibility problems between different vendor offerings.

3.2.3 *Infrastructure-as-a-Service*

IaaS saves costs by eliminating the need to over-provision computing resources to be able to handle peaks in demand. Resources dynamically scale up and down as required, reducing capital expenditure on infrastructure and ongoing operational costs for support, maintenance, and administration. Organizations can massively increase their datacenter resources without significantly increasing the number of people needed to support it (Figure 3.2).

The time required to provision new infrastructure resources is reduced from typically months to just minutes – the time required to add the requirements to an online shopping cart, submit it, and have it approved. IaaS platforms are generally open platforms, supporting a wide range of operating systems and frameworks. This minimizes the risk of vendor lock-in.

Infrastructure resources are leased on a pay-as-you-go basis, according to the hours of usage. Applications that need to run 24×7 may not be cost-effective. To benefit from the dynamic scaling capabilities, applications have to be designed to scale and execute on the vendor's infrastructure. There can be integration challenges with third-party software packages. This should improve over time, however, as and when independent software vendors (ISVs) adopt cloud licensing models and offer standardized APIs to their products.

SaaS is considered to be considerably more mature as a cloud offering than PaaS or IaaS. Even then, it is mainly Small & Medium Businesses that have adopted cloud services. Adoption by the larger enterprises is still extremely low. A perception of cloud services as a high-risk technology option has led the large organizations to restrict the use of cloud services

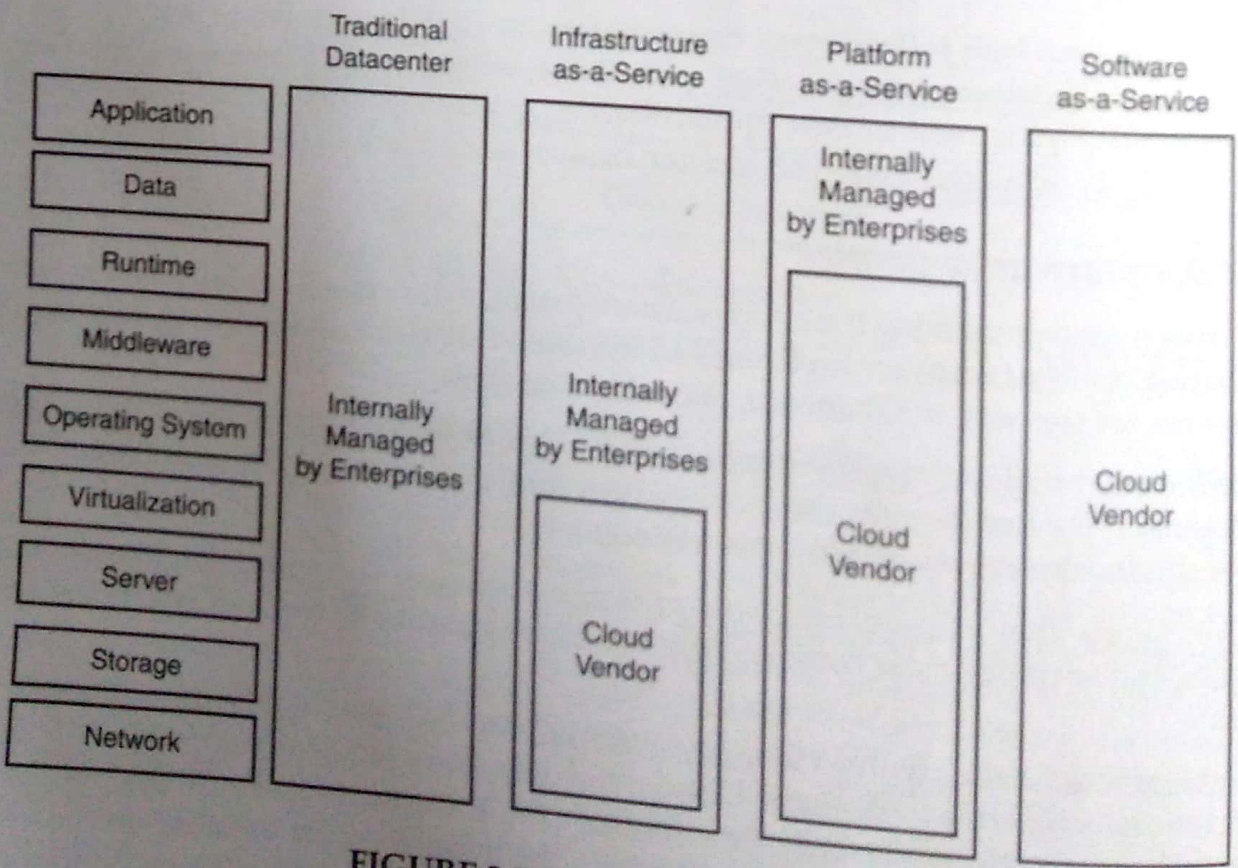

FIGURE 3.2 Cloud taxonomy.

to a limited number of projects. PaaS is a more sophisticated service platform, and is still an emerging product. It will need to stabilize and mature before developers can use it for extensive building of new SaaS applications. For IaaS, the entry of large vendors such as Amazon and other cloud vendors is driving up the maturity of the offering rather quickly, with Amazon leading efforts to define the IaaS market.

There are now hundreds of vendors offering some flavour of cloud computing. Other vendors have attached themselves to the 'Cloud bandwagon' by providing ancillary services to available offerings.

3.3 PRINCIPAL TECHNOLOGIES

The key to being able to provide the dynamic cloud infrastructure is the virtualization layer that sits between the cloud instances and the physical hardware it runs on. The platform virtualization software – the hypervisor – allows multiple operating system instances to run as guests on the same server.

The main drivers for cloud computing are *cost*, *agility*, and *time to market*. By building cloud infrastructures using any cloud orchestrator and provisioning engine one can realize cost savings and improve time to market. This sits on top of the virtualization layer working on network, server, and storage. It is a layer of software that (a) interacts with multiple servers, (b) enables IT departments to pool resources together across servers, and (c) defines standardized tiers of services called *virtual compute centres*. This helps break down infrastructure silos and drives sharing of infrastructure.

Using cloud orchestrator and provisioning engine, IT departments can define organizations and on-board users that can share the underlying cloud infrastructure in a secure multi-tenant fashion. IT can then create standardized collections of VMs and set policies on how users can use these VMs. Users can login into orchestrator and self-provision respective workloads which IT has setup already. This effectively removes IT involvement each time users require infrastructure-enabling agility and faster time to market for applications.

These orchestrators also transport with the API which allow cloud administrator and users to interrelate with the cloud infrastructure in a systematic way. In addition, cloud orchestrator and provisioning engine allows writing workflows to automate creation of cloud infrastructure.

The increased pooling and sharing of resources, self-provisioning, and increased automation deliver greater cost savings in IT infrastructure, agility, and faster time to market for applications. One can deliver cloud benefits for today's applications and for applications developed in the future as cloud orchestrator and provisioning engine builds the top of the hypervisor layer.

Virtualization is the foundation for cloud. It consists of physical hardware with hypervisor layered on top of it. Cloud orchestrator and provisioning engine consists of one or more cells that communicate with a single database and offer a web portal. Using the web portal, cloud administrators create cloud infrastructure resources and users' self-provision cloud infrastructure resources in a secure multi-tenant fashion, thus enabling Infrastructure-as-a-Service (IaaS).

Scanned with CamScanner

Chargeback and metering is a key piece of the on-premise cloud solution. This server talks to own database, server database, and cloud orchestrator and provisioning engine databases and allows to associate costs with the cloud and generate usage and billing reports. This should also integrate with workflow systems, LDAPs, approval process, etc. to provide the lifecycle management of the cloud environment.

3.4 CLOUD STRATEGY

This section provides a high-level guidance to define the cloud strategy and the artefacts that capture the architecture of a cloud-enabled application. These architectural artefacts are meant for the implementation planning phase of cloud, enabling an application. Only the high-level architecture of the system that is to be cloud-enabled is captured in these artefacts.

The implementation planning phase of cloud enables an application lying between the business strategy definition for the adoption of cloud and the design, development, and implementation phases of the application that is being implemented to be offered on the cloud platform. It takes care of linking the business strategy that is defined for a business to adopt a cloud-based strategy and the IT requirements for the applications on the cloud that are needed to support this strategy. So, this critical piece of the implementation planning translates the business intent to a set of IT requirements for the cloud-based application, deriving the high-level structures of the cloud-based application and defining a roadmap for the implementation of the application.

So, the primary input for the cloud implementation planning phase comes from the cloud strategy for the business that is driving the cloud-based implementation of one or more applications on the cloud.

The key steps in cloud implementation planning are as follows:

- Understand cloud strategy.
- Define cloud application requirements.
- Assess cloud readiness.
- Define high-level cloud architecture.
- Identifying change management requirements.
- Develop roadmap and implementation plan.

Of these phases, the artefacts defined here relate to the phase of 'Defining the high level cloud architecture'. In this phase, the high-level structure of the cloud-based application is defined from the inputs of the previous step, the 'Define cloud application requirements'.

The first step in the architecture development is the usage of existing asset analysis to understand which of the components, including components of the existing application, can be used for building the application and how they can be leveraged in the cloud environment. The next step is to derive the high-level structure of the solution from the IT requirements for cloud in the form of the following artefacts. The non-functional characteristics that the application must support and deliver are captured in the artefact 'Non-Functional Requirements'.

Infrastructure strategy and planning for cloud computing helps you develop a cloud strategy, plan, and roadmap:

- Business and IT executive workshop to identify where and how cloud computing can drive business value.
- Develop the value proposition for cloud computing in the enterprise.
- Identify priority of workloads to migrate to cloud.
- Assess the current environment to determine strengths, gaps, and readiness.
- Strategy, plan and roadmap to successfully implement the selected cloud.
- Analyse cloud computing opportunity.
- Analyse IT environment and capability gap.
- Assess cloud readiness.
- Develop high-level cloud roadmap and value proposition.

This helps to deploy the cloud deployment with following benefits:

- **Reduced risk and faster deployment:** It leverages cloud vendor assets, skills, and experience to reduce risk. It accelerates development and implementation by identifying the gaps, activities, and risks and defines mitigation strategies within an implementation roadmap.
- **Improve service:** It identifies the optimal delivery model mix and prioritizes the workloads to migrate to cloud to achieve your business and IT objectives.
- **Lower cost:** It identifies opportunities to reduce capital and operating expense across the infrastructure.

3.5 CLOUD DESIGN AND IMPLEMENTATION USING SOA

Service-Oriented Architecture (SOA) is a very useful architectural style for implementing applications in the cloud. Adoption of SOA would provide the best way to leverage and consume the application services provided by the cloud. A cloud-based application consists of many granular coarse-grained services offered on the cloud. These cloud offerings may in turn integrate and leverage services and systems from different environments. The coarse-grained services that the cloud offerings leverage can come from traditional IT environment in the form of standard services already provided and are internal to the cloud.

Standardization across these different environments is not possible, and hence, they mostly consist of heterogeneous environments. So a cloud service offering should be based on open standards that can be consumed and leveraged by this environment. Language- and platform-independent services can be provided using standards-based, platform-agnostic SOA architecture, with support for appropriate industry and technology standards.

The cloud-based application services consist of a coherent set of business processes that are aligned to the business boundaries, provide a coherent, integrated set of operations that adheres to the business intent and provides value to the users. They also comprise the consumable interfaces whether they are user interfaces on different devices, coarse-grained Web service interfaces, feeds, or widgets.

Map the cloud application services to the business processes that they consist of. These business processes could be at different levels. Drill down the processes to the levels that make business sense.

These processes, along with business strategy, the modular business alignment model, and the process scenarios to be supported form the input to Service-Oriented Modeling and Architecture (SOMA) methodology. SOMA is applied with a meet-in-the-middle approach. The business processes and business strategy along with business goals stated are used to arrive at the service portfolio using SOMA. While deriving the services, a bottoms-up approach is also taken, taking into consideration the existing assets, which consist of the existing application components and services available on the public clouds and internal to the cloud as well as industry cloud component maps.

3.5.1 *Architecture Overview*

The purpose of the architecture overview artefact in a cloud-based implementation is to communicate to the sponsor and external stakeholders a conceptual understanding of the architectural goals of the cloud implementation. It offers a layered conceptual model of the application services to be cloud-enabled and provides a high-level vision of the cloud architecture and its scope to developers. It is easy to explore and evaluate alternative architectural options for the cloud implementation. This stage enables early recognition and validation of the implications of the cloud-based architectural approach and facilitates effective communication between different communities of stakeholders and developers.

This artefact provides an overview of the main conceptual elements of the cloud implementation and relationships within and outside the cloud infrastructure, which may include other cloud and non-cloud environments, candidate offerings, cloud components, nodes, connections, data stores, users, external systems, and technical components to support requirements. As such, it represents the governing ideas and candidate building blocks of the cloud implementation.

This artefact can also provide key stakeholders with the first high-level view of the cloud architecture landscape of their transformed cloud-based implementation.

Typically, the artefact is produced over multiple iterations and as the project moves through the solution definition and design phases, the conceptual models get clearer and this document is kept up-to-date and governed to always have the conceptual elements current.

An architecture summary depiction represents the governing ideas and candidate building blocks of a cloud-based offering and enterprise architecture. It also provides an overview of the main conceptual elements and relationships in the architecture. The main purpose of such a depiction is to communicate a simple, brief, clear, and understandable overview of the target IT system.

At the enterprise-level, an architecture summary depiction is often produced as part of an overall IT strategy to move to a cloud-based offering model. In this instance, it is used to describe the vision of the business and IT capabilities required by an organization that needs the offering to be hosted on the cloud. It provides an overview of the main offerings and the relationships with other offerings, external systems, components, nodes, connections, data stores, users, external systems, and a definition of the key characteristics and requirements

At a system- and component-level, the architecture summary depiction is developed very early in the project (possibly pre-proposal), and influences the initial cloud hosting vision with the component model and operational model. It is intended that design commitments be based on this conceptual overview as the (more detailed) component model and operational model are developed and validated. Subsequently, the component model and the operational model are the primary models used for implementation activities, while their compliance with the architecture summary depiction is maintained continually. Changes to the architecture summary depiction are made using a governance process.

For most infrastructure cloud engagements, the project scope is something less than the client's enterprise architecture. Therefore, the depiction will represent the governing ideas and candidate building blocks of the offerings that are to be hosted on the cloud and represents the focus of the engagement. There can be, then, multiple views of the cloud-based IT environment represented by architecture summary depiction: the current and future view, views by waves of transformation or views by types of clouds, for each solution alternative.

3.6 CONCEPTUAL CLOUD MODEL

So far, we have discussed the different service paradigms for cloud deployment. Now let us discuss the conceptual cloud model based on cloud services.

The conceptual cloud model describes the structure of the cloud-based services as a system in terms of its software components with their responsibilities, interfaces, relationships, and the way they collaborate to deliver the required functionality. The cloud component model for implementation planning is specified at the conceptual level.

The highest level of the conceptual model is the set of cloud-based offerings that make up the cloud-based business solution. These highest-level conceptual components are derived based on the business intent and business functionality. These conceptual components, which form the offerings, align tightly to the business intent that initiated the creation of the cloud-based offerings.

The conceptual offerings that provide the cloud-based implementation of the solution can be further broken down and depicted in a layered composition. The next level of conceptual component model broken down below the offering consists of the following elements:

- High-level service components that form the services provided by the offerings.
- The resources that support the cloud services.
- The technical components that provide the technical underpinnings of the cloud service and support the non-functional needs of the cloud application.
- External and internal services that are leveraged by the cloud application services.

The high-level conceptual component model is the main artefact that provides an abstract view of the design of a cloud application to business stakeholders. This abstract view describes how the business needs will be met by the cloud application components without delving into their technical details. Components identified can be decomposed into further layered conceptual component structures to convey further details of the respective components (Figure 3.3).

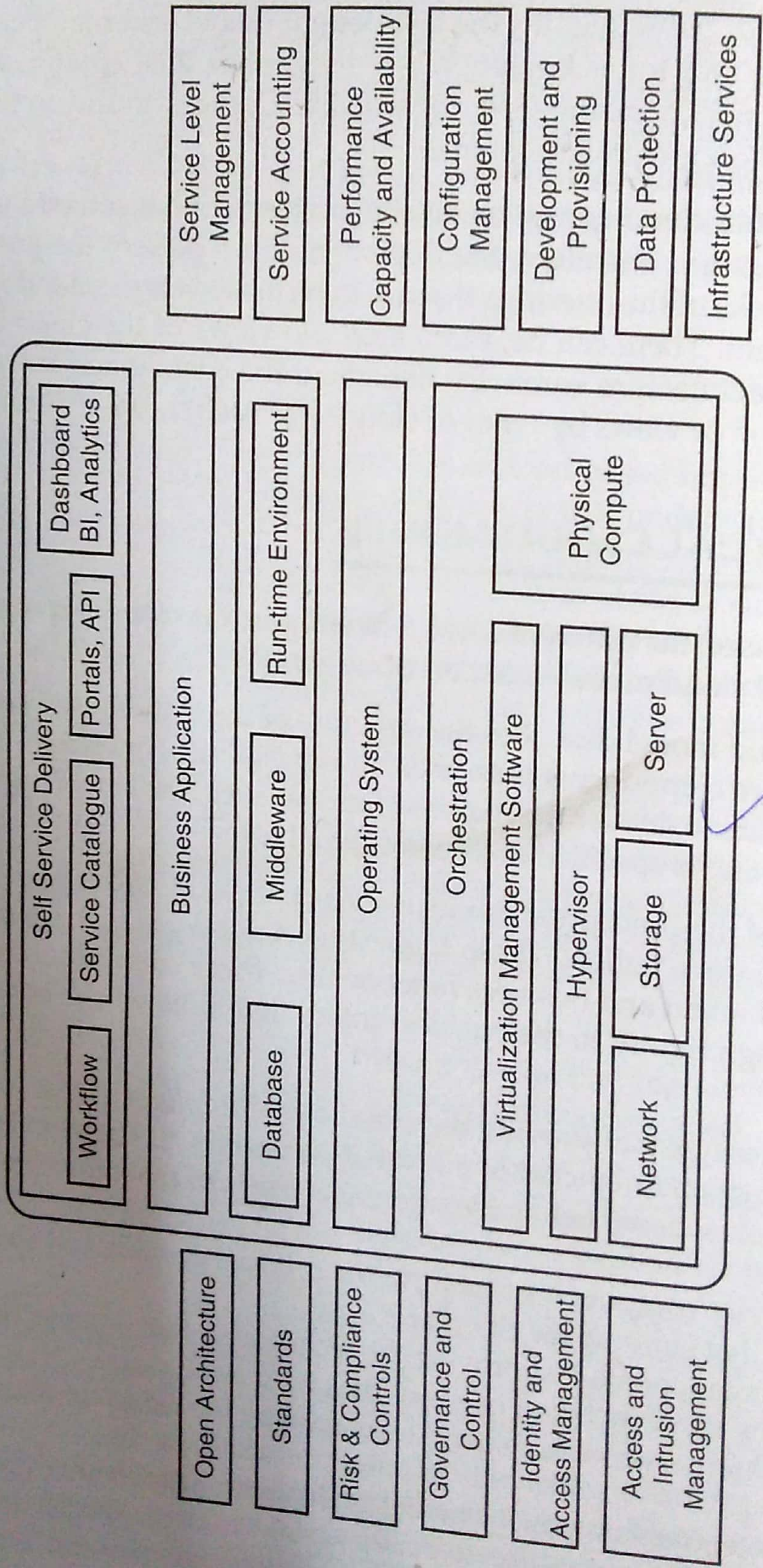

FIGURE 3.3 Cloud model.

The cloud conceptual component model should contain the following elements:

- The conceptual structure of the cloud application.
- The dynamic interactions and dependencies between various conceptual components.
- The components that comprise the cloud services provided by the application, each of which may be made up of sub-components.

At this level, the conceptual components are not converted to physical components but retained at the conceptual level only.

3.6.1 *Cloud Application Security and Privacy Principles*

The cloud application security principles contain the high-level guidance to cover the security and privacy characteristics identified in the Information Asset Profile. This is a summary document of the information asset required for the architecture. Cloud application security principles deliverable contains a number of security principles that occur in a typical application. Typically, quite a number of the characteristics identified in the information asset profile effort will relate to existing enterprise security elements in the client.

In addition to the elements from the existing enterprise security program, consider the following areas that tend to change as part of moving to the cloud environment.

3.6.2 *Governance*

How will decisions like different SLAs be made between cloud vendors and cloud customers for the cloud environment given the new stakeholders? These decisions are typically made by either another infrastructure group within the client IT organization or an external vendor who provides cloud infrastructure management for either a managed private cloud or public cloud environment.

Authentication and Access Control

As an application moves to the cloud, the methods used to authenticate and authorize users require examination. Existing methods may require additional support to work in the cloud environment, such as providing connectivity to an existing server, and may require significant extension to accommodate the requirements of an external vendor providing infrastructure support.

Data Protection

With the move to a cloud environment, the existing methods of data protection – protecting the data from disclosure or modification both on the network and on a storage medium – are likely to require change with the move to virtualized storage and with the introduction of additional infrastructure administrators, often from a vendor organization.

Logging and Alerting

Logging is the ability to tie actions to an individual. Alerting is the method of recognizing activities that may indicate a malicious act and bringing those to the attention of the security staff. With the move to a cloud infrastructure, with different network and additional administrators, often from an external vendor, tying actions to specific individuals requires additional attention.

The focus in this work product is to provide the broad guidance as to the security and privacy elements that must be present in and around the cloud application.

3.7 CLOUD SERVICE DEFINED

This section highlights the aspects of service, its scope, and cloud-based different platform integration and deployment services.

3.7.1 *Service Definitions*

Let us begin with a cursory glance on what 'Service' is.

- **Service:** A specific IT deliverable that provides customer value. It is measurable in customer terms and provides the basis for doing business with the customer. It is delivered through a series of processes and/or activities.
- **Services Portfolio:** The collection of services provided by IT that in their aggregation represents all the 'value add' activities performed by IT.
- **Service Component:** A logically grouped set of activities that represent part of a service that touches the customer. Service components are grouped together to create deliverables that form the basis for doing business with customers.
- **Service Owner:** The individual accountable for ensuring the customer receives the identified value of the service. They take the customer's end-to-end view of IT activities by working with process owners to ensure all the required delivery components fit together smoothly.
- **Process:** A collection of related activities that take inputs, transforms them, and produces outputs that support an enterprise goal.
- **Enablers:** The decomposed components of a service (process, organization, technology) that are combined to create *Service Deliverables*. The collection of activities (and their supporting roles and technology) become the workflow needed for *Service Delivery*.
- **Service Level Agreements:** A grouping of Services or Service Components that have had specific delivery commitments and roles identified with the customer. SLAs can be grouped together in different ways to represent various products. Examples would include things like e-mail and service delivery.
- **Service Level Management:** Service Level Management governs the planning, coordinating, drafting, agreeing, monitoring, and reporting on Service Level Agreements (SLAs), and the on-going review of service achievements to ensure that the required and cost-justifiable service quality is agreed to, maintained, or where necessary improved. SLAs provide the basis for managing the relationship between the provider and the customer. Service Level Management is essential in any organization so that the level of IT Service needed to support the business can be determined and monitoring can be

initiated to identify whether the required service levels are being achieved – and if not, why not.

- **Service Level Management Objectives:** SLA objective is to maintain and improve IT Service quality, through a constant cycle of agreeing, monitoring, and reporting upon IT service achievements and instigation of actions to eradicate poor service – in line with business or cost justification. It develops a better relationship between IT and its customers.

SLAs should be established for all IT services being provided. Underpinning Contracts (UCs) and Operational Level Agreements (OLAs) should also be in place with those suppliers (external and internal) upon whom the delivery of service is dependent.

SLAs are likely to be a service . . . if the 'what' is separated from the 'how', and we can change the underlying assets – processes, technologies, data – as well as suppliers, but still provide the promised business outcomes and value. It is a service if its description and design highlight separate roles for customers, users, providers, and suppliers, and it is offered with a pre-defined value proposition stating specific price points and performance metrics tied to business (not just IT operational) outcomes. It is something that makes sense as a customer/user-selectable item, with corresponding service requests available in a menu or service catalogue.

The facts here are startling – inefficiency is prolific – clearly, progress is needed.

3.7.2 *Services Scope Overview*

Platform Integration and Deployment Services

These provide a set of project services for the planning, design, procurement, assembly and integration, site installation, and project management of the deployment mainstream and special-purpose end-user devices, and also includes a number of asset lifecycle services.

It integrates and customizes multiple devices – including PCs, wireless and mobile devices, kiosks, ATMs, point-of-sale (POS) devices, and printers – to end-user specifications. These services utilize a factory approach for the off-customer-site build and integration services, using build and integration centres around the globe.

Software Platform Management Services

It provides a set of project and annuity services to manage end-user software platforms, including image development and management, application software packaging and distribution, and services to manage the availability of the end-user platform proactively. This includes services to design and migrate end-user platforms, for example, Microsoft XP to Vista, or Linux. It utilizes a factory approach for the off-customer-site platform management services, using management and configuration centres.

3.7.3 *Platform Integration and Deployment Component Services*

This section discusses some of the very important platform integration and deployment component services.

Order Management

This service handles the procurement of hardware and/or software on behalf of the customer (regardless of asset ownership), and the fulfilment (delivery) of that hardware and software to the central build centre prior to Platform Build and Pre-Load.

Warehousing and Stock Management

This gives provision of central warehousing facilities to store hardware and other agreed components before and after Build and PDP and before shipment to site. It provides services for receiving and warehousing, and ensures that inventory is sufficiently maintained and protected while in storage.

Platform Build and Test

This service provides services to build, integrate, customize, prepare, and test the hardware and software platform before shipment to the customer site or end-user location.

Base Backup

This service provides a base backup to be taken during platform pre-build (assumes the software platform supports this feature).

Data and Personality Migration

It migrates data and personality settings (e.g., desktop wallpaper, Internet Favourites, Desktop layout) from original to replacement platform.

Asset Tagging and Custom Labelling

This service provides for custom asset tagging of hardware components during Build and pre-delivery preparation at the central build facility.

Asset Inventory Update and Report

This service includes adding any new asset to the customers or managed asset database.

Logistics and Delivery

This service includes the packaging and shipment to site of the user hardware platform following Platform Build and Test services.

Installation

This service provides the deployment of the platform into the customer's operational environment. This could be either at the end-users' desk, or agreed deployment centre location for machines that are to be collected or deployed by the customer.

Extended Project Management

This is an extension to the base project management services that cloud uses to manage internal functions and activities. This service extends project management scope to cover overall management of the platform deployment program, including customer and customer third-party resources and activities.

Platform Removal and Return

This includes decommission of platform from customer location, and return to cloud vendors for refurbishment or disposal (but not including for refurbishment or disposal activities).

Asset Refurbishment

This service checks for suitability of the hardware platform, refurbishment, and upgrade as appropriate, and integration into the deployment process.

Asset and Data Disposal

This service provides removal of sensitive data from the hardware platform to varying levels of security (e.g., to department of defence standards), as well as safe environmental disposal to international standards and/or asset value recovery.

Emergency Replacement

This service provides for the emergency replacement of like for like hardware platforms (pre-built to the customers standards) to agreed service levels.

Software Platform Management Services

This section explains the software platform management services in detail.

Software Platform Design Consulting

These services help the client understand the business and technology needs for a new software platform, and provide the design specification.

Software Platform Creation and Customisation

This service is to create and test a software platform to support the needs of the business.

Software Platform Support and Maintenance

It is for ongoing support and maintenance of the client's software platform, including platform updates and management, and ongoing problem support.

Application Scripting

This is the creation of application software unattended installation scripts to accommodate the customer environment. This service can include new software applications and updates to existing packages.

Application Discovery

This service is to discover what applications are in use across the organization.

Application Portfolio Management

This service is to help the clients manage their application portfolio.

Software Delivery

It services the schedule and transport application packages to target end points. This service includes the capability to PUSH software out to user PCs from a central distribution and/or the ability for the users to PULL Software to their PCs using a Web interface.

Antivirus Management

These are the services to manage the delivery of antivirus signature files. It provides real-time protection from malicious virus attacks that can cause potentially disastrous system.

Patch Management

It services to ensure end-user devices have the latest patch releases installed. It provides real-time deployment of critical Operating System patches to help protect against flaws and vulnerabilities.

Health Check Services

These services ensure that end-user devices are in a good state of health of the system, like infrastructure and PC checks. It performs remote monitoring of supported workstations for critical hardware alerts and initiates scripted responses as alerts are received. Examples are identification of low memory, detection of spyware, identify low disk space.

Compliance Services

These services are to ensure that end-user devices are compliant with client standards. They perform remote monitoring of supported workstations to perform activities such as detecting peer-to-peer software detection of games, etc.

Until there is an agreement to industry standard definition of what an IT service is, you must agree with the client on a definition of 'service'. So, we will consider different type of the services visibility with respect to customer.

External Services

These services are visible and seen by the customer, and include business services (business intelligence, logistics, receiving orders, marketing services, invoicing, accounting, etc.) and user services (desktop support, maintenance, education, etc.).

Internal Services

These services are invisible or less visible to the customer, but essential to the delivery of IT services. They include infrastructure services (hosting services, storage, availability, data retention, or recovery) and network services (network, remote access, mobile or wireless services, etc.). It takes care of application services (integration, testing, design, maintenance, optimization, etc.).

User-Initiated Service Request

It includes service request handling in incident management, for example, the service request progresses through its lifecycle exactly as an incident. Most companies separate user service requests and incidents. For example, the service requests follow a different process and use a different tool to track it throughout its lifecycle. Significant IT workload is responding to user-initiated requests for some work to be done.

Users request services for which their businesses have already contracted with a service provider, and to which they are already entitled. Some people have referred to the list of services from which a user can order services as a service catalogue. Perhaps it is a 'User Services Catalogue'. These service requests and this type of listing of services are appropriately offered through the single point of contact for users of IT services, the service desk.

Customer-Initiated Service Request – A Service Catalogue-Based Request

It includes the concept of a service catalogue as a list of services that the customer can order. The customer is the one who pays for services. When customers order a service, their users are entitled to receive services under that agreement. Each individual request made by a user is a user initiated service request. The user is entitled to services that their business has ordered through a service catalogue request. This type of request is from the customer to some account rep or 'business relationship manager' who responds to this request and initiates the service provisioning for that customer.

3.8 SUMMARY

This chapter visualizes the different cloud models with respect to services. It also takes into account what service is all about and the different types of infrastructure services that can be offered as cloud as a service.

CHAPTER 4

CLOUD SOLUTIONS

4.1 INTRODUCTION

Cloud environment presents an opportunity to enhance the user experience by providing a broader communication path for reaching out to the user or for providing a series of business services to the user via the application features.

Deploying the application to the cloud is somewhat different since the deployment process will not be done locally within the enterprise and the existence of the provisioned image and a series of deployment steps is needed to deploy the application and validate the deployment.

Development and testing environments are readily available within the cloud environment. The advantages of these environments, especially from a costing perspective, are numerous as there is no need to purchase and deploy servers within the normal enterprise environment. If a POC was being developed and project was cancelled, no software, hardware, or even development tools would have to purchased, only to be thrown away later as the cloud supports development and testing of applications.

4.1.1 *Cloud Application Planning*

The design and development of cloud applications requires many unique considerations:

- Business functions.
- Application architecture.
- Security for cloud computing.
- Cloud delivery model.
- User experience.
- Development, testing, and run-time environments.

Application architecture is selected through some sort of criteria evaluation.

The key thing to talk about from a security aspect is the enhancements to the existing security model where data protection and isolation of the data from other areas of the cloud environment. Encryption is one possibility to further enhance the security model whereas the enterprise would not necessary invoke that option. Other aspect of security would be to further authenticate and authorize users of the application and the services the users have been entitled to use.

4.1.2 *Cloud Business and Operational Support Services (BSS and OSS)*

Business Support Services (BSS) are the components that cloud operators use to run their business operations. The term BSS applies to service providers in all sectors such as utility providers. Typical types of activities that count as part of BSS are taking customer orders, managing customer data, managing order data, billing, rating, and offering services.

Operational Support Services (OSS) are computer systems used by cloud service providers. The term OSS most frequently describes 'network systems' dealing with the network itself, supporting processes such as maintaining network inventory, provisioning services, configuring network components, and managing faults.

BSS and OSS components need to be externalized so that the supporting services of the application being transformed to the cloud environment can capitalize on the various functions the OSS and BSS provides. For example, provisioning can be adapted to support the applications provisioning requirements instead of creating self-provisioning from scratch. The ability to tap into the monitoring, metering events and keep track of the activity within the cloud environment can assist the application to continue to provide the service levels and quality of service. Each of the OSS and BSS offered by the cloud environment will continue to support the application and the consumers of the application in maintaining the key characteristics of cloud computing.

The cloud application architecture brings together the business services, security, infrastructure, and integration required for an optimal solution. Cloud services represent any type of IT capability that is provided by the cloud service provider to cloud service consumers. Typical categories of cloud services are infrastructure, platform, software, or business process services. In contrast to traditional IT services, cloud services have attributes associated with cloud computing, such as a pay-per-use model, self-service usage, flexible scaling, and shared of underlying IT resources.

The cloud vendor is responsible for delivering instances of cloud services of any category to cloud service consumers, the ongoing management of all cloud service instances from a provider perspective, and allowing cloud service consumers to manage their cloud service instances in a self-service fashion. The technical aspects of a cloud service are captured in a service template, which is also the artefact that describes how the OSS capabilities of the cloud vendor are exploited within the context of the respective cloud service.

For most cloud services, specific software are required for implementing cloud service specifics: For IaaS, these are typically hypervisors installed on the managed infrastructure; for PaaS, this would a multi-tenancy enabled middleware platform; for SaaS, a multi-tenancy enabled end-user application; and for BPaaS, multi-tenancy enabled business process engine. Depending on the nature of the respective cloud service, the notion of a cloud service instance represents different entities.

Cloud services can be built on top of each other, for example, a software service could consume a platform or infrastructure service as its basis, and a platform service could consume an infrastructure service as its foundation. However, this is not required, that is, a software service could also directly be built on top of 'traditional' infrastructure, clearly inheriting all constraints associated with such an infrastructure. In general, basic cloud architectural postulates to share as much as possible across cloud services with respect to management platform and underlying infrastructure. However, it does not require to only having one single, fully homogeneous infrastructure – of course, this would be the ideal goal, but given different infrastructure requirements, this is not possible. For example, if a particular cloud service has very specific infrastructure needs, it is clearly allowed to run this cloud service on a dedicated infrastructure (e.g., the Google search engine or HPC cloud services would always run on a purpose-built physical infrastructure for performance and efficiency reasons; they wouldn't run virtualized compute cloud service).

In the context of building cloud services on top of each other, it is important to distinguish the sharing of a common OSS/BSS structure across multiple cloud services and the usage of the actual cloud service capability by another one.

In any case, each cloud service offered by a cloud service provider is 'known' to the BSS and OSS of the cloud vendors. Consequently, a cloud service provider offers cloud services as a result of very conscious business decisions, since taking a cloud service to market must be supported by a corresponding solid business model and investments for the development and operations of the cloud service.

4.2 CLOUD ECOSYSTEM

It is very important to understand the relationship between a cloud service and the artefacts that can be developed based on and within the boundaries of an ecosystem-focused IaaS or PaaS cloud service. Bringing any cloud service to market requires corresponding pre-investment, along with respective metering and charging models in support of the corresponding business model. Therefore, making the characteristics flexible to artefact developers is not possible as it would be very hard to make the corresponding costs flexible and by that very hard to predict.

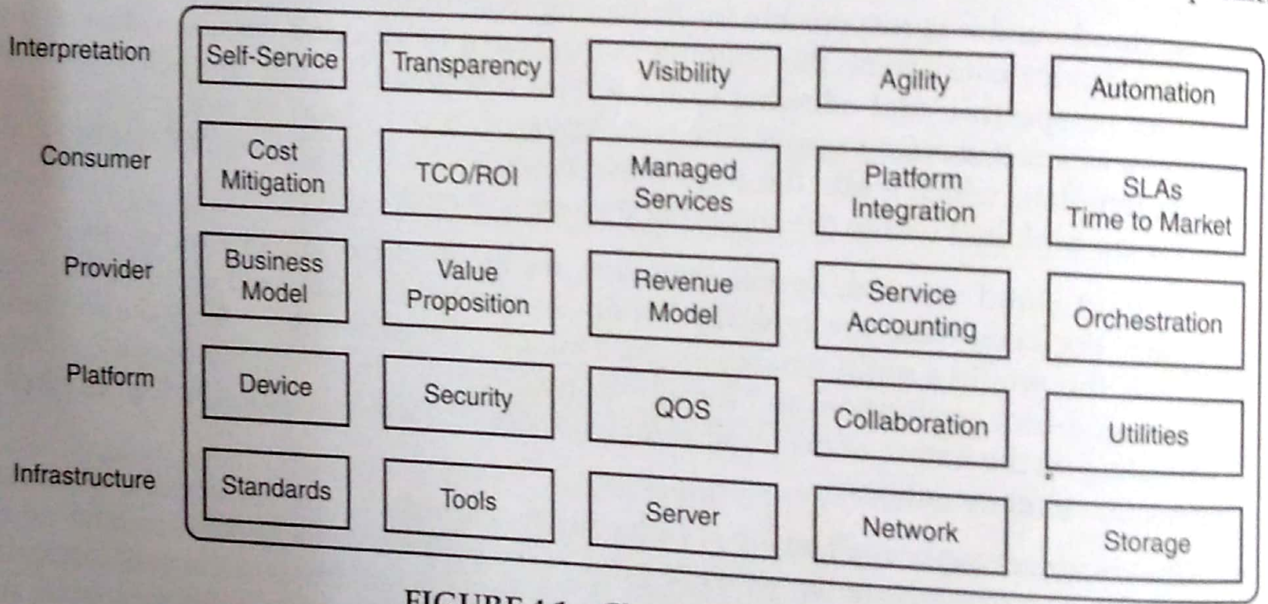

FIGURE 4.1 Cloud ecosystem.

This illustrates that defining and delivering a cloud service requires nailing down all corresponding functional and non-functional requirements. The artefacts developed on top of an ecosystem-focused cloud service have then only very minimal room to change how these functional and non-functional requirements are addressed (Figure 4.1). Note that this is not to be viewed as something negative, but rather as something very positive from an ecosystem perspective – it is a core value proposition of ecosystem-focused cloud services to provide pretty strict guidelines with respect to how they can be exploited as this is the main factor driving a reduction in cost of artefact development. The easier it is to develop artefacts for such a cloud service, the more likely the cloud service is successful.

As a summary, it is important to note that there is a difference between developing cloud services as a very conscious technical and business decision and developing artefacts on top of ecosystem focused cloud services prescribing the boundaries for how these artefacts can run.

Note that sometimes the concept of a 'Cloud Service' is also referred to as a 'Cloud Service Product'. The cloud's capability to have multiple environments to deploy the application is a major advantage because it can be a mix and match condition that best fits the business function and the application. This means that the organization is not tied to one solution for cloud computing but rather multiple solutions. So, depending on the business needs, the application, and what the application has to offer, cloud requirements will help to select the proper cloud environments.

Cloud-based environments come handy especially when used to develop, test, and run your application for the following reasons:

- Available in your own private cloud environment or on the public cloud.
- Rapid access to a configurable development and test environment to speed time to market.
- Self-service Web portal for enterprise account management and provisioning in minutes.
- Pay-as-you-go pricing, with the choice of preferred pricing through reserved capacity packages.
- Security-rich environment designed to protect your systems and data.
- Access to a rich catalogue of software images for improved flexibility and rapid provisioning.
- Rapid provisioning and faster time to value.

4.3 CLOUD BUSINESS PROCESS MANAGEMENT

Business Process Management (BPM) governs an organization's cross-functional, customer-focussed, end-to-end *core business processes*. It achieves strategic business objectives by directing the deployment of resources from across the organization into efficient processes that create customer value. Its focus is on driving overall bottom line success by *integrating verticals and optimizing core work* (for example, order-to-cash, integrated product development, integrated supply chain). This is what differentiates BPM from traditional functional management disciplines.

In addition, intrinsic to BPM is the principle of 'continuous improvement', perpetually increasing value-generation and sustaining market competitiveness (or dominance) of the organization. It clearly defines and aligns operations, organizations, and information technology. The cloud environment can help in the following ways:

- **Integration of core business:**
  - Holistic.
  - Crosses organizational functions and boundaries (height and breadth).
  - Includes business and technology.
- **Value-focused efficiency:**
  - Customer-centric perspective.
  - Bottom-line success.
  - Speed at which ROI is delivered.
  - Performance measurement.
- **Continuous:**
  - This is based on longer period of intervals pertaining to cloud business.
  - Continual improvement.

- **Cultural:**
  - Cultural considerations of the organization and geographical area kept in mind at the time of due diligence of the requirement.

4.3.1 *Identifying BPM Opportunities*

This section discusses the opportunities required for successful cloud business process management and the characteristics of cloud deployment offerings. The answers to the following questions can help you identify cloud opportunities better:

- Are the strategic value proposition and capabilities defined for your organization?
- How does your overall strategy drive the design and execution of your business processes? Is there a traceability of execution to goals?
- How do you manage your core business processes?
- What are your current process initiatives?
- What are your current process governance facilities?
- Are your existing organizational structures aligned to enable efficient process operations?
- How do your customers measure and assess the performance of your processes?
- How does your process performance compare to your competitors?
- How effectively does your current technology (information, systems, tools, machines, etc.) enable the enterprise's core business processes?
- What risks and challenges does your current technology present for current and future process capabilities?
- What products does your organization have? What type of products?
- What are the notable pieces of your IT portfolio?
- Has your organization adopted SOA?
- How are processes currently modelled in your organization? What's included in the model?
- What design/development tools are currently used in the organization?
- What testing tools are currently used by the organization? What are the strengths/weaknesses of the tools?
- Please describe the different business processes that are automated in the organization?

Cloud application development offerings provide:

- Cloud application reference architecture.
- Unmatched experience developing high-performing, secure applications across a wide range of technologies of the cloud vendor.
- Unmatched application security expertise.
- Leadership in cloud related technologies – multi-tenancy, virtualization, pervasive computing.
- Significant expertise with cloud business models.
- Ability to integrate a portfolio of related cloud services.

4.3.2 *Cloud Technical Strategy*

This section gives the technical strategy on how the cloud customers can enable the cloud deployment.

Cloud services enable our cloud users to build middleware clouds in their datacenter and utilize public clouds, where it make sense by providing the following cloud-enabled middleware services:

- Infrastructure Services.
- Platform Services.
- Application Services.

Cloud strategy enables our organizations to do the following:

- Build middleware clouds in their datacenter.
- Utilize public clouds where it makes sense.

It does so by providing support in the following areas:

- **Cloud-enabled middleware services:**
  - Infrastructure Services.
  - Platform Services.
  - Application Services.
  - Serving the on premise and public clouds.

So, what does it mean to develop an application service for the cloud? For one, it means product features development similar to on-premise software. This means:

- **Enabling the software for cloud essentially implies:**
  - Support for collaborative multi-tenancy.
  - Self-service registration.
  - Managing customers and their entitlements.
  - Single sign on.
  - Additional security concerns.
- **Integration with the datacenter:**
  - Firewalls, reverse proxy configurations.
  - Fully qualified domain names, certificates.
  - Management of services, patch procedures.
  - Isolation, recovery, backup issues.

4.3.3 *Cloud Use Cases*

Infrastructure as a Service (IaaS) or Test/Development

Problem: Development teams require unpredictable amounts of infrastructure to get their job done. In majority of the cases, getting all of these resources in place before they are required in the development cycle can be quite a challenge. Purchasing the hardware consumes project budget and procurement of it is often quite slow. Static development and testing resources require manual re-provisioning in order to re-purpose resources for use, or new resources need to be purchased to meet demand. In cases where project timelines are short, project managers often choose not to set up much of an environment because it depletes their budget or jeopardizes the project's delivery schedule. Actual usage of the system(s) can be quite short in terms of absolute time. Different types of projects need different kinds of development components (like SQL server, SharePoint, BizTalk, etc.) depending on the architecture of the solution. Besides the testing environment, the team usually needs a system that looks quite similar to the production environment in order to perform simulation, stress and load tests, for example, or to delivery end-user training, etc.

Solution: Companies can create standardized service catalogue items for common infrastructure requirements and enable development and test teams to access infrastructure in a self-service model (IaaS). Overall control over the process is maintained with business policies around quotas, reservations, reclamation, and standardized offerings.

Standardized Development Platforms/Middleware (PaaSEnable)

Problem: Developers are often not concerned about the impact of their code on Operations. They deliver their code without involving Operations into architectural decisions or code reviews. Enterprise architects recognize the large costs associated with non-standard development platforms. To simplify the ongoing maintenance and streamline development operations, many companies are creating corporate standards around development stacks that include middleware/applications. However, most companies are hesitant to an external PaaS offering due to the constraints of having to rewrite their internal applications to fit the external PaaS API sets.

Solution: Companies can create standardized development platform definitions for use by development teams to standardize and streamline their efforts. This improves corporate IT productivity by helping them build a 'private cloud' that provides a common foundation for building custom applications which run securely behind their own firewall.

Application Cloud

Problem: Companies want to move beyond self-service for infrastructure and provide application owners the ability to define, instantiate, and manage complex multi-tier applications. This includes configuring the application for production usage and monitoring performance within the application for SLA optimization.

Solution: End users can access complete application definitions and manage them according to their quotas and preferences defined by cloud administrators. Applications in production can be monitored across multiple factors and automatically scaled up and down according to business policies.

Software-as-a-Service (SaaS) to End Customers

Problem: Many companies or ISVs want to deliver their applications to end users as a service. However, creating a multi-tenant SaaS offering requires substantial development to support security, performance, and scalability needs. Due to these high costs, companies cannot offer new services based upon existing applications.

Solution: Companies can provision a unique application instance per customer with private cloud automation capabilities. Environments are provisioned according to business policies and unique SLAs can be delivered per customer according to the business arrangements.

The cloud engine should provide the automation to provision on-demand complex application and configuration environments required along with the dynamic application scaling. It should also deliver unique business policy selections including custom placement and high availability across multiple datacenters.

4.4 CLOUD SERVICE MANAGEMENT

A service management system provides the visibility, control, and automation needed for efficient cloud delivery in both public and private implementations:

- **Simplify user interaction with IT:**
  - User-friendly self-service interface accelerates time to value.
  - Service catalogue enables standards which drive consistent service delivery.
- **Enable policies to lower cost with provisioning:**
  - Automated provisioning and de-provisioning speeds service delivery.
  - Provisioning policies allow release and reuse of assets.
- **Increase system administrator productivity:**
  - Move from management silos to a service management system.

The emergence of cloud deployments is prompting enterprises to either assemble in-house teams to manage specialized cloud service providers or look to third-party cloud brokers chiefly due to the following reasons:

- Every service-oriented approach needs a mechanism to enable discovery and end-point resolution.
- Registry/repository technology provides this where service delivery is inside the firewall.
- Cloud services delivered across firewalls need something similar – a third party that serves as a 'service broker'.

Leveraging service brokers will probably become a critical success factor in cloud computing as cloud services multiply and expand faster than the ability of cloud consumers to manage or govern them. The growth of service brokerage businesses will increase the ability of cloud consumers to use services in a trustworthy manner. Cloud service providers are expected to begin to partner with cloud brokerages to ensure that they can deliver the services they promote. These cloud intermediaries will help companies choose the right platform, deploy apps across multiple clouds and perhaps even provide cloud arbitrage services that allow end-users to shift between platforms to capture the best pricing.

There can be three categories of opportunities for cloud brokers:

- **Cloud service intermediation:** Building services atop an existing cloud platform, such as additional security or management capabilities.
- **Cloud aggregation:** Deploying customer services over multiple cloud platforms.
- **Cloud service arbitrage:** Supplying flexibility and 'opportunistic choices' – and fostering competition between clouds.

It will be similar to cloud services under one umbrella except that the services being aggregated won't be fixed. This flexibility will be important while doing chores such as providing multiple e-mail services through one service provider or providing a credit-scoring service that checks multiple scoring agencies and then selects the best score.

The ability to federate an application across multiple clouds will become important – if one service goes down, another can be started – and the service broker will just simplify it. To help federate the clouds, a 'storefront' (Apps.gov) site can be created with services that are pre-screened to meet government procurement guidelines. The new site can be expected to cut red tape and make it easier for government agencies to quickly deploy the latest technology.

4.4.1 *Key Cloud Solution Characteristics*

The essential cloud orchestrator and engine key characteristic capabilities are:

- **Scalability:** Cloud orchestrator should maintain an index of the resources that are acquired from the hypervisor, giving the master a low overhead and enabling it to scale across tens of thousands of machines across multiple geographies.
- **High Availability:** Cloud orchestrator should play for the master node to support 'Active–Passive' as well as 'Active–Active' scenarios for availability and Disaster Recovery (DR). Cloud orchestrator should also monitor individual physical server for availability and in case of a physical resource server failure, should restart the VM on another running server to meet the requirements.
- **Application Lifecycle:** Cloud orchestrator should offer complete application lifecycle support from the creation of infrastructure to installation, configuration, and launching an application to deletion or expiration. This allows applications to be instantiated, removed, or flexed very quickly to respond to real-time demand for those applications.
- **Multi-tenancy/Role-based Administration:**
  - Cloud orchestrator should support multi-tenant capability with specific user permissions. Cloud orchestrator should have multiple personas which are like Cloud Admin, Account Owner, and User. Application definitions are only 'published' to specific users. The application owner or administrator logs in with his/her credentials and can view (User) the application VMs that have been allocated to him/her and do admin operations (Account Owner) on those VMs if needed.
  - Role-based administration allows fine-grained control of what each person can or cannot do in terms of cloud orchestrator features.
- **Policies:** Cloud orchestrator should provide rich set of policies that can be enabled. These policies can be modified or new ones can be created to take effect at the global level on applications, VMs, hosts, etc. These policies can also be embedded in the service definitions to take effect automatically depending on metric threshold. For example, a policy can be created to allow an application to flex up to 10 VMs during high load or demand and to be reduced to only 2 running VMs during low load or demand. This frees up resources that can be used by other applications that are experiencing high load.
- **Alarms:** Cloud orchestrator should provide pre-defined alarms that can be set at the global level for applications, VMs, hosts, etc. These alarms can be used to notify individual users or application owners regarding the application thresholds being reached.

For example, an alert can be sent if the response time of an application is below an SLA but there are no more resources for the application to flex-up.

- **Application Awareness and Policy-based Allocation:** Cloud orchestrator should be aware of application requirements and optimizes the placement of the application accordingly, e.g. placing the VMs running the application close to each other to reduce latency. Cloud orchestrator should support the major application servers.
- **Resource Awareness and Policy-based Allocation:** Cloud orchestrator should optimize the usage of the cloud infrastructure through intelligent resource allocation policies and allows load balancing of the VMs
- **Elasticity Based on Performance (Flex-up/Flex-down):** Cloud orchestrator should provide out-of-box functionality to flex-up or flex-down an application instance or resource based on performance metrics.
- **Reporting and Accounting:** Cloud orchestrator should provide metering and billing reports on resource allocation and actual usage. Additionally, this data can be used to create reports on inventory capacity and consumption. This allows the different business owners to create reports on how much or how little the application is used, and administrators can then adjust the resources allocated to each application accordingly.
- **Self-Service Portal:** Cloud orchestrator should enable a self-service portal for application owners. Application owners can request machines or entire multi-machine application environment, monitor, and control them through this portal. It should drive the workflows necessary to create the environment, and provide run-time environment management in order to support application elasticity. For example, the owner of the auditing application may request more resources for his application during a busy period.

4.5 ON-PREMISE CLOUD ORCHESTRATION AND PROVISIONING ENGINE

On-Premise Cloud Orchestration and Provisioning Engine can be a bundled offering that includes hardware, software, and the services one needs to get started with cloud computing. It should include all the elements in a services ecosystem. It must have a self-service portal, it should include the automation, and it should track and control all of resources.

The other objective is to completely integrate and include a service and then, on top of that, users can add additional services to do integration or other types of cloud work. It can be a pre-packaged private cloud offering that can bring together the hardware, software, and services needed to establish a private cloud to accelerate selling efforts and effectiveness.

Cloud Orchestration and Provisioning Engine should be designed from client cloud implementation experiences and integrated with the service management software system with servers, storage, and services to enable aprivate cloud in IT environment. This will help to remove the guess work of establishing a private cloud by pre-installing and configuring the necessary software on the hardware and leveraging services for customization to your environment. All that is required is to install your applications and start leveraging the benefits of cloud computing, such as virtualization, flexibility, scalability, and a self-service portal for provisioning new services.

Cloud Orchestration and Provisioning Engine should provide an alternative to traditional IT infrastructure for IT executives seeking to enhance delivery of services and to transform the datacenter into a cost-effective Dynamic Infrastructure. Cloud Orchestration and Provisioning Engine ecosystem should be 'Built for performance' and should be based on architectures and configurations required by specific workloads. It should enable the datacenter to accelerate the creation of services for a variety of workloads with a high degree of flexibility, reliability, and resource optimization.

4.5.1 *Benefits/Value Proposition*

Powers faster time to innovation, lowers cost per unit of innovation

- **Innovation:** It should dramatically improve business value and IT's effect on time-to-market by enabling the business workloads to rapidly and accurately be deployed when and where they are needed.
- **Decrease Operational Expenses:** It must gain productivity increases in IT labour costs through automation. Maximize capital usage and reduce added capital expense.
- **Reduce Complexity and Risk:** With automation and standardization, the human error-factor should be minimized.

4.5.2 *Cloud Orchestration and Provisioning Requirement Analysis*

In order to understand the Cloud Orchestration and Provisioning Engine requirements, we need to understand the test and development requirements of the cloud deployment. Therefore, we should be sure about the automation of the testing and development cycles to reduce the deployment time. In order to do so, we can initiate the process of discussion and try to talk to cloud customer about the opportunity for the deployment of the Cloud Orchestration and Provisioning Engine. After this, we can set the boundaries of the environment. About 30–50 percent of any given IT environment is devoted to test/development purpose. It can take developers days, weeks, or even months to procure and configure appropriate hardware, networking, management software, storage, with which they can test. With the Orchestration and Provisioning Engine, a developer can log into a self-service portal, select resources required and timeframe, select an image to provision from the service catalogue, and be ready to go in hours as opposed to months. Customer datacenters support hundreds or thousands of distinct composite applications representing business workloads. They leverage multiple types of servers, storage, networks, middleware, and operating systems. About 70 percent of the IT datacenter expense is spent assembling and re-assembling existing infrastructure, leaving fewer resources for innovation.

Cloud solutions are based on its service management solutions. The first solution in a series of Cloud Orchestration and Provisioning Engine solutions that we should consider should be workload-specific deployments. It will be a great entry point for users who want to get into cloud computing as you can roll a Cloud Orchestration and Provisioning Engine into their environment without effecting anything else in the environment, and use it as your initial pilot project to start running cloud services.

Cloud Orchestration and Provisioning Engine in any IT environment represents an alternative entry point into cloud computing. So, in some cases, the users want to turn their existing environment into a cloud. In that case, we go into the datacenter, install a cloud management platform, and assign the existing resources that are there to the cloud. One scenario for this is that the organization already has plenty of equipment that they are just not using efficiently, and they see a lot of benefit from turning their existing investment into a cloud.

Cloud Orchestration and Provisioning Engine can be a great door opener and a great way to jump-start your cloud services. It can be bundled with the hardware, software, and services that you need to quickly getup and running; you can use this as a seed-and-grow model.

Entry points:

- Turn existing environment into a cloud:
  - Install cloud management platform and assign existing resources to the cloud.
  - Scenario – you already have enough equipment or Cloud Orchestration and Provisioning Engine offering is not the right platform.
- Jump start your cloud
  - Hardware + Software + Services required for a quick start up.
  - Can use a 'seed and grow' model – start with a Cloud Orchestration and Provisioning Engine and then addmore.

So what solution should you use? Let's look at some scenarios. So when can you opt for Cloud Orchestration and Provisioning Engine? This should be when you wish to get started with cloud computing and see the advantage of having a cloud management platform bundled along with the resources. This is the most rapid way to get a cloud platform up and running. In a scenario where you want to transition your existing resources into a cloud, you use this service. Also, a user who uses Cloud Orchestration and Provisioning Engine will benefit when cloud vendor can help them plan, design, and implement services, whether they want to implement infrastructure as a service, platform as a service, or turn it into a production cloud.

So, in some cases as you can see, it's appropriate to opt for Cloud Orchestration and Provisioning Engine and the business development and test services, because even if you just start with provisioning engine, you may eventually want to build out your service catalogue, integrate it into your directory, integrate it into cloud vendor or third-party service management products, and extend the capability of the platform.

4.5.3 *Cloud Infrastructure Security*

The security aspect of the cloud infrastructure goes side by side with Service Oriented Architecture (SOA) security. We can introduce it as a layered approach. At the top we can see the service layer with the run-time secure virtualized environment. This is available as the distributed service environment. These services include administrative and security aspects across different clouds as well as within a single cloud. This should gel with Web services stack and it is important that we should bind the internal resources with the other cloud services to offer better hybrid cloud services for successful cloud deployments. We will discuss more about this in Chapter 9.

One of the key aspects of SOA is the ability to easily integrate different services from different providers. Cloud computing is pushing this model one step further than most enterprise SOA environments, since a cloud sometimes supports a very large number of tenants, services and standards. This support is provided in a highly dynamic and agile fashion, and under very complex trust relationships. In particular, a cloud SOA sometimes supports a large and open user population, and it cannot assume a pre-established relationship between cloud provider and subscriber.

The Secure Virtualized Runtime layer on the bottom is a virtualized system that runs the processes that provide access to data on the data stores. This run-time differs from classic run-time systems in that it operates on virtual machine images rather than on individual applications. It provides security services such as antivirus, introspection, and externalized security services around virtual images. While the foundations of Secure Virtualized Runtime predate SOA security and are built on decades of experience with mainframe architectures, the development of Secure Virtualized Runtime is still very much in flux. Cloud vendors continuously invest in research and development of stronger isolation at all levels of the network, server, hypervisor, process, and storage infrastructure to support massive multi-tenancy.

Cloud Orchestration and Provisioning Engine Integrated service management is offered with network, servers, storage, services, and financing as an integrated offering for client test platforms.

- **Improved time to value:** Quickly deliver a cloud using a preloaded and integrated system.
- **Improved innovation:** Dramatically improve business value and IT's effect on time-to-market by delivering services faster.
- **Decrease capital expenses:** Maximize capital usage and reduce added capital expense.
- **Reduce complexity and risk:** With automation and standardization the human error factor is minimized.
- **'Fit for purpose':** Based on architectures required by specific workloads.
- **Self-contained:** Service management, software, hardware, storage, networking.
- **Modular:** Automatically expandable and scalable.
- **Virtualized:** End-to-end across server, network, and storage.
- **Self-service:** Ease of consumption.
- **'Lights-out':** Zero touch automated operations.

Cloud Orchestration and Provisioning Engine is offered as a services engagement which can build a solution to a client's needs, including creation of custom virtual images for dispensing. Summary points are given below

- **Drastically reduce set-up and configuration time:**
  - New environments in minutes!
- **Reduce risk by codifying infrastructure:**
  - Freeze-dry best practices for repeated, consistent deployments.
- **Security throughout the entire lifecycle.**
- **Simplify maintenance and management:**
  - Flexibly manage and update the components of your patterns.
  - Ensure consistency in versions across development, test, production.
- **Spend less time administering, more time developing new solutions.**

Scanned with CamScanner

4.6 COMPUTING ON DEMAND (CoD)

On-demand computing is the need of the hour. It is very essential even in a supercomputing environment. On-demand computing can be implemented using various virtualization techniques. Cloud gives you an option to leverage the computing infrastructure without actually buying the hardware. This helps you to transfer the workload if your resources are not able to support it, and at other times, lets others utilize your resources that are lying idle. In this way this makes it possible to use the resources in most efficient ways. It may be possible that there can be many spikes that can come for the utilization but cloud experts can make it smooth.

The uniquely rich set of features that on-demand computing can offer enables service seeker to deploy a true utility. The platform allows users to:

- **Align cost with utilization** so that users can scale costs down as well as up. This allows a workload to start with minimal upfront costs and scale as the demand grows without paying a penalty to increase capacity. Additionally users can benefit from not incurring the disruption to move to a larger machine.
- **Increase end-users availability significantly.** As workload can be moved dynamically, it is possible to move workload from one server to another without interruption so remedial work can be carried out if server down time is required.
- **Balance workload dynamically across multiple servers** without taking applications offline. Using the workload mobility features customer can align their costs by ensuring that workload is deployed in such a way as to optimise systems resource.
- **React to short-term resource requirements** almost instantly. If a workload has to be deployed at short notice, a virtual machine (VM) can be created on the server and resources allocated instantaneously using the dynamic capacity model.
- **Reduce the physical foot print** in the datacenter. Consolidation of workload on to a smaller number of servers will improve space, power, and cooling metrics.
- **Confidently increase system utilization** to over 75 percent without fear of degrading performance for end-users.
- **Develop a simple charging model that reflects usage** for end users as the service delivery culture continues to mature.
- **Double the workload delivered** in the power and cooling envelope.

4.6.1 *Pre-Provisioning*

For the on-demand computing requirement pre-provisioning is the viable option as it helps organization meet the requirement of the dynamic datacenter requirements. Organizations would like to reduce the time they take to commission servers when a new workload is to be deployed.

This approach is ideal when:

- The sizing and capacity planning is fully understood.
- The workload is fairly constant, ensuring good utilization levels are achieved.
- There are business reasons that require the physical separation of workload.
- Workload can be scaled horizontally.

ideal = best possible
idle = empty vacant

4.6.2 *On-Demand CPU/Memory/VM Resources*

In the dynamic environment it is important to track the requirement of CPU, Memory, and VMs. It is based on the common pool concept where resources are allocated and de-allocated as the requirement is over. This approach is ideal when:

- Workloads are trending upwards so investment can be aligned with utilization.
- Peeks in workload are longer term.
- Workload scales vertically.
- It is more economically advantageous to 'buy out' dynamic capacity.

4.6.3 *Dynamic Capacity*

Utility CoD is used to automatically provide additional processor capacity on a temporary basis within the shared processor pool. Usage is measured in processor-minute increments, and is reported via a Web interface or collection of report by cloud vendor engineer. Billing is based on the reported usage.

This approach is ideal when:

- The workload is very variable and multiple workloads can be hosted on a single machine so that the utilization can be levelled out.
- The workload has short periods where system utilization increases massively, but for the majority of the time it is not resource-intensive.
- Workloads can share a physical platform.
- The workload is designed to scale vertically only.
- Users want to dynamically balance workloads across servers.
- Users want to continue to run very small workloads without incurring the overheads associated with running a physical server to support it.

Benefits

- Partition mobility, significantly reduced power/cooling footprint, donation of unused processor cycles of VMs with dedicated processors to uncapped partitions and at the same time guaranteed performance of these VMs.
- Very short deployment time (time-to-market optimized).
- Lowest possible cost for deployment of small workloads.
- Less management effort, for example, when using VMs.
- Most granular charging scheme, pay for the CPU and memory cycles actually used.
- Complete decommissioning of partitions; resources are available for other purposes.
- Flexible workload management, workloads can compensate for each other, thus reducing overall utilization.
- Ideal for environments with identical systems management, utilities for development and testing.

Limitations

- Short peaks, must not exceed certain limits, and needs to be monitored (via Web interface) to ensure best value is obtained.
- Utility CoD provides processor resources only to the uncapped partitions.

However, one of the most important advantages of the Dynamic Capacity model is to be flexible, that is, to allow the switching of CPU capacity on and off as needed, which can reduce costs significantly. This not reflected in this calculation as it requires input from application owners.

4.6.4 Cloud Platform Characteristics Based on CoD

This section discusses cloud platform characteristics on the basis of low-end, on-demand, and dynamic-capacity–based servers.

Low-End Servers

- Physical segregation of servers.
- High administration cost due to management of more physical servers.
- Limited and complex scalability, process – maximum 8 processors per server – slower turn-around time for server deployment.
- Longer lead time for server deployment from ordering of servers to setting up of infrastructure.
- Not ideal for short product lifecycle application due to fixed cost expenditure for hardware.
- Wastage of hardware resources for applications that reacts to volatile market.
- Unable to share resources between applications.
- Wastage of un-used processing cycles if the application does not fully utilized the resources.
- No hardware/application interdependency forcing down time on application owners.
- No capacity on demand capability. Downtime is required for adding new hardware.
- Low price per CPU cycle purchased but higher cost per CPU cycle actually used.

On-Demand Platform

- Physical or logical segregation of servers or partitions implementation.
- Lower administration cost due to less physical servers' management.
- Can cater for quick turn-around time for new application deployment or increase capacity due to business requirements.
- Enhanced time to market for new product launch with immediate availability of CPU/memory capacity.
- Not ideal for short product life cycle application due to fixed cost expenditure for hardware.
- Wastage of hardware resources for applications which reacts to volatile market.
- Able to share I/O, CPU, and memory resources between applications.
- Able to take advantage of un-used CPU/memory if dynamic VM reallocation or share pool methodology is implemented.
- To provide an environment in which there are no hardware/application interdependencies forcing down time on application owners care full capacity planning and management is required.
- Capacity on demand capability – no downtime is required if COD CPU/memory is sufficient.
- Higher price per CPU cycle, lower cost per CPU cycle actually used.

Dynamic Capacity Platform

- Choose virtual machine or workload virtual machine implementation for application consolidation.
- Lower administration cost due to less physical and logical servers management.
- Can cater for quick turn-around time for new application deployment.
- Enhanced time to market for new product launch with immediate availability of infrastructure and setup.
- Ability to scale up and down which will be ideal for application with short product life cycle.
- Able to cater for applications which react to volatile market, i.e., scaling up and down capacity.
- Able to share I/O, CPU, and memory resources between applications.
- Able to take advantage of un-used processing cycle of other applications.
- No hardware/application interdependency forcing down time on application owners as workload can be dynamically moved to facilitate maintenance, etc.
- Capacity on-demand capability. No downtime is required as the machine is fully configured.
- Higher price per CPU cycle, lower cost per CPU cycle actually used. Average price due to the ability to optimize utilization and rapidly deploy workload.

4.7 CLOUDSOURCING

Today we are living in the era of optimizing the hardware resources and moving towards the large enterprise day by day so cloud computing is becoming the ingredient part of infrastructure deployments. Now you may not need only cloud computing, you may need the entire consulting, implement and management solutions.

The new wave that is igniting the cloud deployments in the service industry as a new trend is Cloudsourcing – outsourcing the end to end solution using cloud methodology using public cloud, infrastructure and platforms. This will be more planned approach as it will comprise the whole service cycle of outsourcing business with cloud principles with the help of strategized connected cloud platforms that will match the overall enterprise requirements.

This includes the whole cloud implementation, IT business consulting, integration, and configuration of the business. This will give the option through which we can enjoy the benefits of service industry with the benefits of the clouds that gives the innovative approach of paying the resources over subscription.

Real deployment of the Cloudsourcing will requires the business model with the impact of cloud customer and cloud vendor requirements.

With respect to cloud customers, it is important to note there is no control on the infrastructure layout of the cloud deployments. Even there is no control over the place from where the data services are offered from the cloud vendor. It is also to known that cloud customer don't have think about the operational staff for the deployments.

Thus, Cloudsourcing will be playing the vital role in the next generation of cloud implementation. With the availability of new open source tools it is like icing on the cake, integrated with partner cloud solutions, platforms and infrastructure. Also the new charging models like the services on both a project and subscription basis will give new wave to deploy and adopt the cloud sourcing models.

This will help to customize application on cloud infrastructure. This will be primarily being offered as a public cloud and all these offering will be available as managed services. These services will be prototype based that is developed internally on the product and working applications. Therefore, it will give a good chance to use the intellectual property for developing different business vertical solution easily.

4.8 SUMMARY

In this chapter, we pointed out the main features of Cloud Orchestration and Provisioning Engine, BPM clouds, cloud sourcing, and requirements of service management. Next chapter will discuss about different types of cloud offerings.