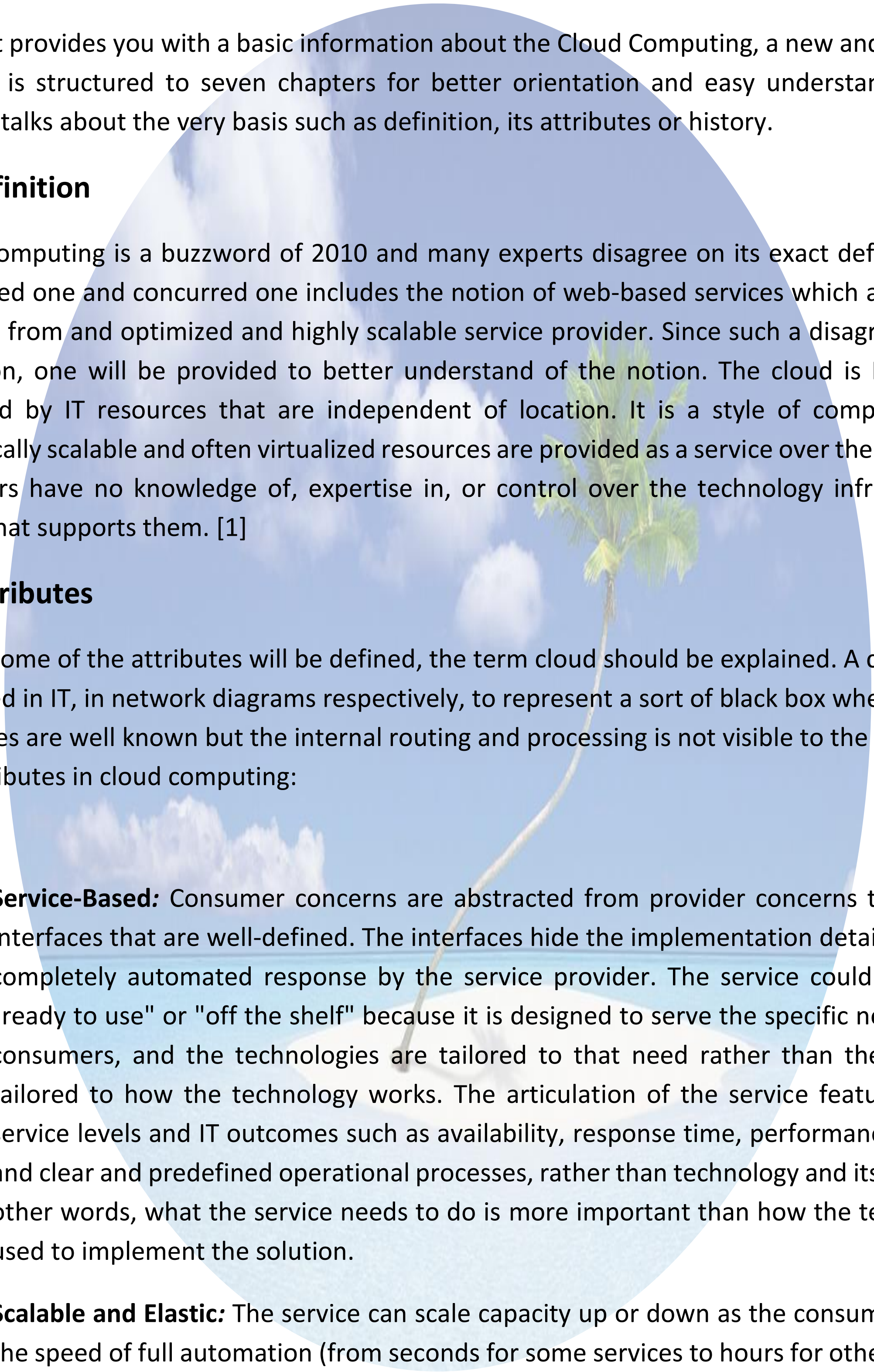
1 Introduction



This text provides you with a basic information about the Cloud Computing, a new and fastly growing term. It is structured to seven chapters for better orientation and easy understanding. The first chapter talks about the very basis such as definition, its attributes or history.

1.1 Definition

Cloud Computing is a buzzword of 2010 and many experts disagree on its exact definition. But the most used one and concurred one includes the notion of web-based services which are available on demand from and optimized and highly scalable service provider. Since such a disagreement on the definition, one will be provided to better understand of the notion. The cloud is IT as a service, delivered by IT resources that are independent of location. It is a style of computing in which dynamically scalable and often virtualized resources are provided as a service over the Internet where end-users have no knowledge of, expertise in, or control over the technology infrastructure (the cloud) that supports them. [1]

1.2 Attributes

Before some of the attributes will be defined, the term cloud should be explained. A cloud has been long used in IT, in network diagrams respectively, to represent a sort of black box where the interfaces are well known but the internal routing and processing is not visible to the network users. Key attributes in cloud computing:

- **Service-Based*:*** Consumer concerns are abstracted from provider concerns through service interfaces that are well-defined. The interfaces hide the implementation details and enable a completely automated response by the service provider. The service could be considered "ready to use" or "off the shelf" because it is designed to serve the specific needs of a set of consumers, and the technologies are tailored to that need rather than the service being tailored to how the technology works. The articulation of the service feature is based on service levels and IT outcomes such as availability, response time, performance versus price, and clear and predefined operational processes, rather than technology and its capabilities. In other words, what the service needs to do is more important than how the technologies are used to implement the solution.

- **Scalable and Elastic*:*** The service can scale capacity up or down as the consumer demands at the speed of full automation (from seconds for some services to hours for others). Elasticity is a trait of shared pools of resources. Scalability is a feature of the underlying infrastructure and software platforms. Elasticity is associated with not only scale but also an economic model that enables scaling in both directions in an automated fashion. This means that services scale on demand to add or remove resources as needed.

- **Shared:** Services share a pool of resources to build economies of scale and IT resources are used with maximum efficiency. The underlying infrastructure, software or platforms are shared among the consumers of the service (usually unknown to the consumers). This enables unused resources to serve multiple needs for multiple consumers, all working at the same time.

- **Metered by Use**: Services are tracked with usage metrics to enable multiple payment models. The service provider has a usage accounting model for measuring the use of the services, which could then be used to create different pricing plans and models. These may include pay-as-you go plans, subscriptions, fixed plans and even free plans. The implied payment plans will be based on usage, not on the cost of the equipment. These plans are based on the amount of the service used by the consumers, which may be in terms of hours, data transfers or other use-based attributes delivered.

- **Uses Internet Technologies**: The service is delivered using Internet identifiers, formats and protocols, such as URLs, HTTP, IP and representational state transfer Web-oriented architecture. Many examples of Web technology exist as the foundation for Internet-based services. Google's Gmail, Amazon.com's book buying, eBay's auctions sharing all exhibit the use of Internet and Web technologies and protocols. More details about examples are in the chapter four – Intergration [2]

1.3 History

History of Cloud Computing surprisingly began almost 50 years ago. The father of this idea is considered to be John McCarthy, a professor at MIT University in US, who first in 1961 presented the idea of sharing the same computer technology as being the same as for example sharing electricity. Electrical power needs many households/firms that possess a variety of electrical appliances but do not possess power plant. One power plant serves many customers and using the electricity example, power plant=service provider, distribution network=internet and the households/firms=computers. [3]

Since that time, Cloud computing has evolved through a number of phases which include grid and utility computing, application service provision (ASP), and Software as a Service (SaaS). One of the first milestones was the arrival of Salesforce.com in 1999, which pioneered the concept of delivering enterprise applications via a simple website. The next development was Amazon Web Services in 2002, which provided a suite of cloud-based services including storage, computation and even human intelligence. Another big milestone came in 2009 as Google and others started to offer browser-based enterprise applications, though services such as Google Apps. [4]

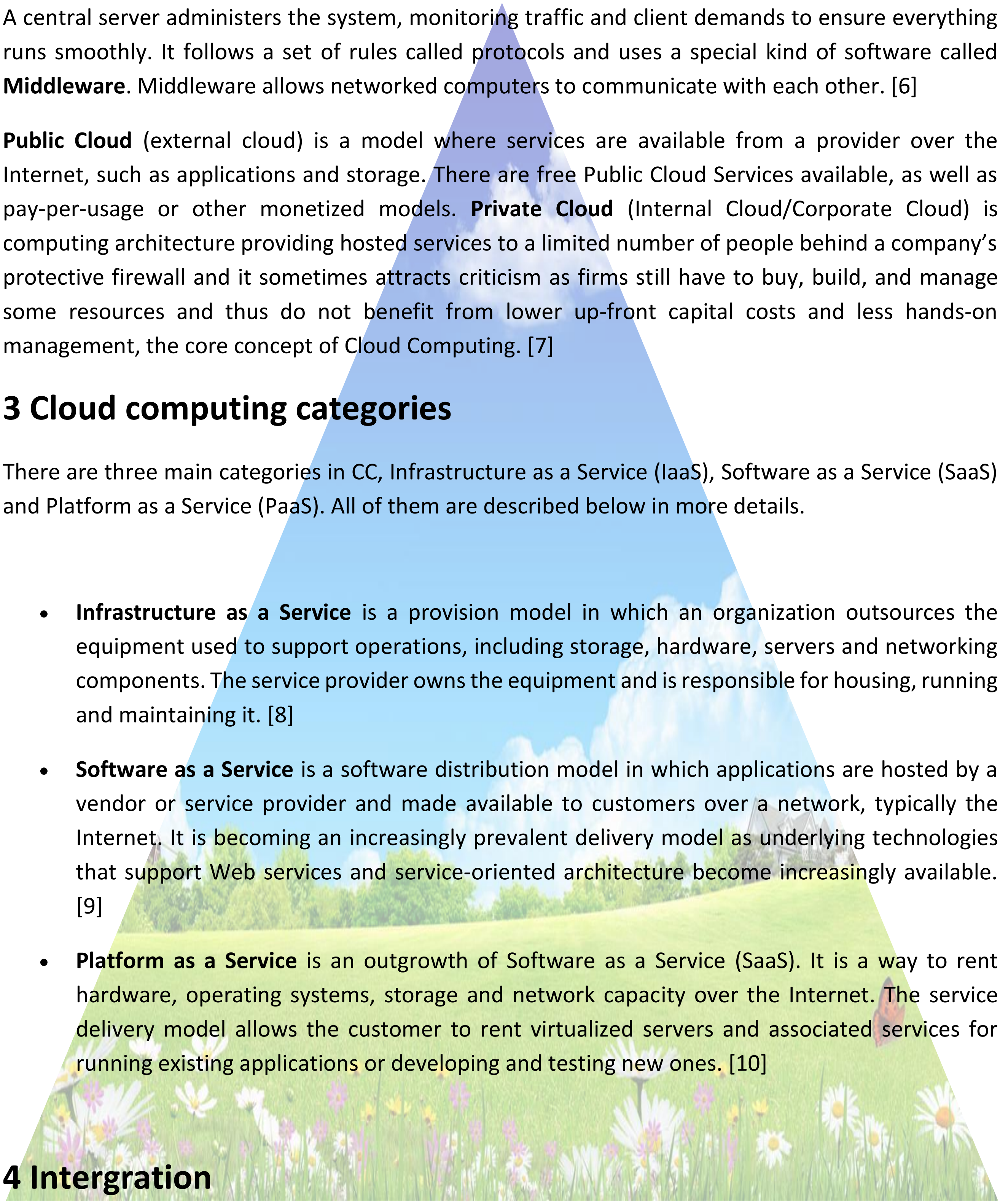
2 Architecture

A basis infromation about the architecture is provided in this chapter, together with the explanations of relevant terms such as virtualization, Frond/Back end or Middleware.

- Virtualization is best described as essentially designating one computer to do the job of multiple computers by sharing the resources of that single computer across multiple environments. Virtual servers and virtual desktops allow you to host multiple operating systems and multiple applications locally and in remote locations, freeing your business from physical and geographical limitations. [5]

The Cloud Computing architecture can be divided into two sections, the front end and the back end, connected together through a network, usually Internet. The **Front End** includes the client's computer and the application required to access the cloud computing system. Not all cloud computing systems have the same user interface. Services like Web-based e-mail programs leverage existing Web browsers like Internet Explorer or Firefox. Other systems have unique applications that provide network access to clients.

The **Back End** of the system is represented by various computers, servers and data storage systems that create the "cloud" of computing services. Practically, Cloud Computing system could include any program, from data processing to video games and each application will have its own server.



A central server administers the system, monitoring traffic and client demands to ensure everything runs smoothly. It follows a set of rules called protocols and uses a special kind of software called **Middleware**. Middleware allows networked computers to communicate with each other. [6]

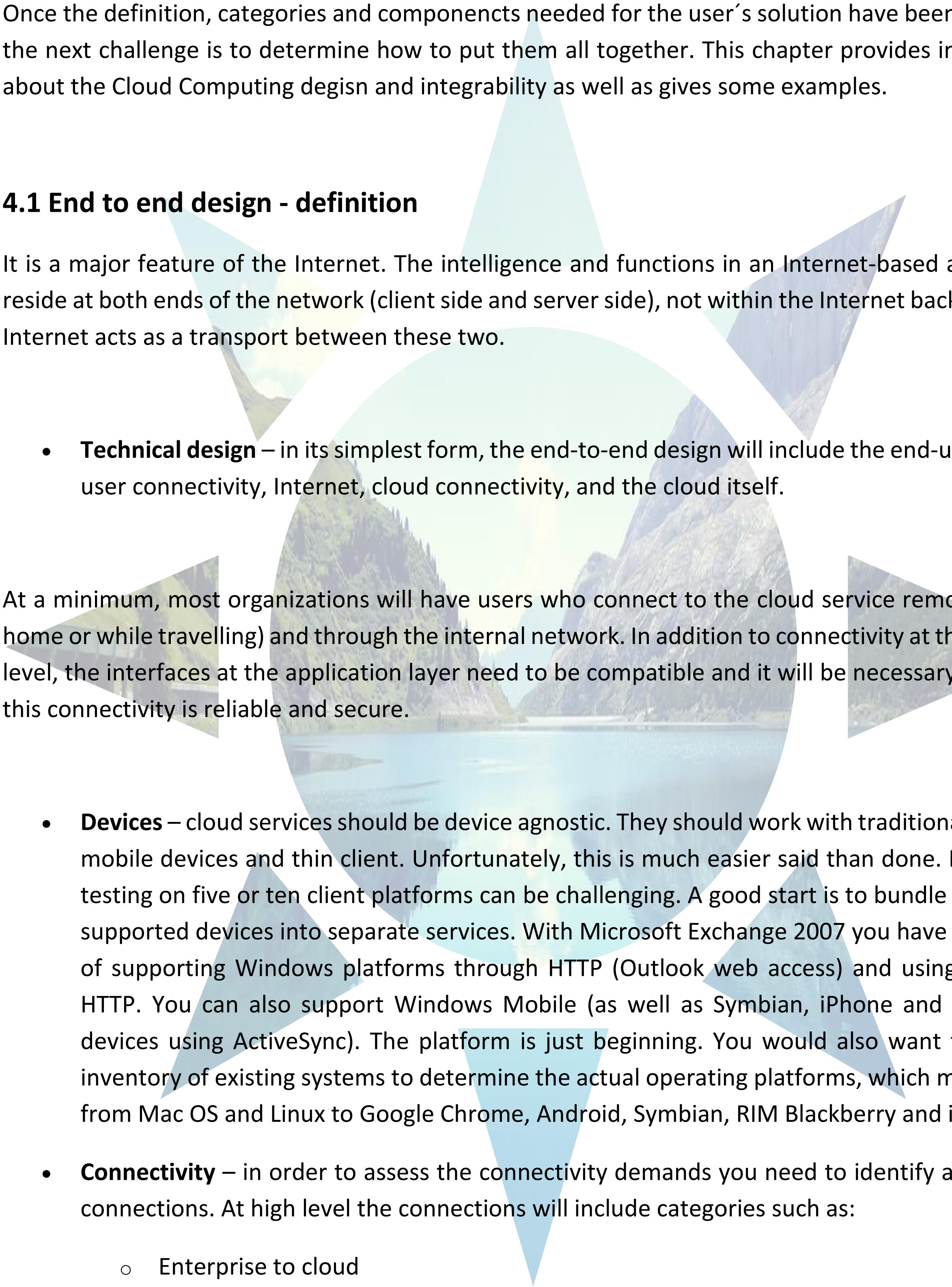
Public Cloud (external cloud) is a model where services are available from a provider over the Internet, such as applications and storage. There are free Public Cloud Services available, as well as pay-per-usage or other monetized models. **Private Cloud** (Internal Cloud/Corporate Cloud) is computing architecture providing hosted services to a limited number of people behind a company's protective firewall and it sometimes attracts criticism as firms still have to buy, build, and manage some resources and thus do not benefit from lower up-front capital costs and less hands-on management, the core concept of Cloud Computing. [7]

3 Cloud computing categories

There are three main categories in CC, Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS). All of them are described below in more details.

- **Infrastructure as a Service** is a provision model in which an organization outsources the equipment used to support operations, including storage, hardware, servers and networking components. The service provider owns the equipment and is responsible for housing, running and maintaining it. [8]
- **Software as a Service** is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. It is becoming an increasingly prevalent delivery model as underlying technologies that support Web services and service-oriented architecture become increasingly available. [9]
- **Platform as a Service** is an outgrowth of Software as a Service (SaaS). It is a way to rent hardware, operating systems, storage and network capacity over the Internet. The service delivery model allows the customer to rent virtualized servers and associated services for running existing applications or developing and testing new ones. [10]

4 Intergration



Once the definition, categories and componencts needed for the user´s solution have been identified the next challenge is to determine how to put them all together. This chapter provides information about the Cloud Computing degisn and integrability as well as gives some examples.

4.1 End to end design - definition

It is a major feature of the Internet. The intelligence and functions in an Internet-based application reside at both ends of the network (client side and server side), not within the Internet backbone. The Internet acts as a transport between these two.

- **Technical design** – in its simplest form, the end-to-end design will include the end-user device, user connectivity, Internet, cloud connectivity, and the cloud itself.

At a minimum, most organizations will have users who connect to the cloud service remotely (from home or while travelling) and through the internal network. In addition to connectivity at the network level, the interfaces at the application layer need to be compatible and it will be necessary to ensure this connectivity is reliable and secure.

- **Devices** – cloud services should be device agnostic. They should work with traditional desktop, mobile devices and thin client. Unfortunately, this is much easier said than done. Regression testing on five or ten client platforms can be challenging. A good start is to bundle the sets of supported devices into separate services. With Microsoft Exchange 2007 you have the option of supporting Windows platforms through HTTP (Outlook web access) and using RPC over HTTP. You can also support Windows Mobile (as well as Symbian, iPhone and Blackberry devices using ActiveSync). The platform is just beginning. You would also want to take an inventory of existing systems to determine the actual operating platforms, which might range from Mac OS and Linux to Google Chrome, Android, Symbian, RIM Blackberry and iPhones.
- **Connectivity** – in order to assess the connectivity demands you need to identify all required connections. At high level the connections will include categories such as:
  - Enterprise to cloud