Business Intelligence and the Cloud

STRATEGIC IMPLEMENTATION GUIDE

Michael S. Gendron

WILEY

Contents

Foreword

In 1991 ADNET Technologies was cofounded with a vision of bridging the gap between information technology (IT) and business—essentially, making IT real for businesses. Michael Gendron and I first met several years ago when he was working on his book *Business Intelligence Applied: Implementing an Effective Information and Communications Technology Infrastructure*. He was (and still is) a professor of information systems at Central Connecticut State University. It was immediately apparent that we shared a common vision to bridge the gap between IT and business.

In the past 30 years we have seen the rise and fall of an array of technologies. Are we now in the process of witnessing the fall of the personal computer as tablets and smartphones take over? I will leave that for history to tell, but what is certain is that technology changes at a rapid pace. As the speed of technology changes, so does the speed of business and competition. As technology has become part of the DNA of business, the speed of technological change has naturally affected the rate at which businesses change. The effects have been widespread, including leveling the playing field between large and small companies. If technology is changing at such a rapid pace that even large companies have difficulty staying abreast of the change, how can small companies possibly compete? What chance do they have in this ever-changing world of technology? The answer is the cloud. The book you are reading makes that point.

While ever more powerful devices with easier-to-use operating systems and applications have made their way from the corporate workplace into our homes, the real game changer has been the rapid adoption of the Internet and the widespread availability of high-speed broadband service. What was not realistically possible (i.e., affordable) back in 2000 is suddenly common. The planets aligned, so to speak, and the "cloud" was born.

Having maintained responsibilities in both the technical and the business worlds, I have seen firsthand the disconnection between the two. My career has been dedicated to bridging the gap between IT (what is possible) and business (the reason IT exists). The author and I share the same drive and vision, as is evident in this book.

Over time, IT has shifted from merely automating manual processes and being viewed as an "expense to be managed" to a business driver. The chief information officer has morphed from a technical guru with minimal business knowledge to a technically savvy business strategist with more of a direct line to becoming the chief executive officer. In my experience consulting with organizations on technology strategy, it is generally the case that the gap between IT and business needs has not been fulfilled. As a result, we spend a lot of time attempting to educate our clients on the very topics covered in this book.

Michael Gendron has done an excellent job of providing the reader with a solid foundation of what the cloud is and how it came about. He provides the reader with the background of the cloud as well as an overview of some of the more important underlying technologies that drive the cloud, such as virtualization. This discussion provides the reader with the necessary foundation to begin to fully engage in discussions on cloud computing.

Once the foundation has been set, the three dimensions of cloud computing are explored, starting with the essential characteristics. What is the cloud? What are the service models? What are the deployment models? Why do I care? These questions are answered in depth and with an eye toward the business reader to begin to bridge the gap between IT and business. Michael continually builds on the foundation set earlier, weaving through the multitude of terms and concepts that face the reader in the real world—cutting through the hype and myths to simplify what may at first seem complicated.

The old adage "you can't manage what you don't measure" still applies and has certainly not been forgotten here. As the reader will discover, the decision to move to cloud computing is not always clear-cut; it depends on a multitude of factors. Measurement of the financial effects of such a move, both short-term and long-term, must be considered. How do we measure the financial impact of the cloud? How do we balance the financial consequences with performance (service level agreements)? How does our staff fit into the overall equation? Total cost of ownership and the measurement of results are given due importance in making decisions. Examples are used to help put the various concepts into context so that the reader can gain a better understanding of how the pieces of the puzzle fit together.

The puzzle is completed in this book with a solid discussion of current trends and how businesses can benefit from those trends. The Big Data, mobile computing, and business intelligence discussions not only educate the reader on each of these trends but also highlight the intersection of these trends with cloud computing.

IT must be an integral part of an organization's strategy and strategic planning process. This book meets the challenge by presenting the material from a strategic implementation perspective. Michael Gendron has a unique talent for maintaining the technological strength of a topic while making it clearly understandable to the business reader—successfully bridging the gap between IT and business. I highly recommend this book for any manager looking to gain a competitive advantage through the strategic application of technology.

EDWARD D. LAPRADE
President and CEO
ADNET Technologies
Farmington, CT
www.thinkADNET.com

CHAPTER 1

A History of How We Got to Cloud Computing

Cloud computing is not just a technical orientation, a set of computer architectures, or computing standards developed by the technocrats of the twenty-first century. It is a new way to deliver information systems (IS) that require new business policies; it is a strategic orientation driven by changes in technology that enable rapid business innovation. Cloud computing is enabled by many technologies, but more important than the technologies, cloud computing is a new way to deliver IS and information technology (IT). Many of the old ways (if you can call five years ago "old") are still around and are needed. However, the organization that wants to be competitive has to closely examine cloud computing as a strategic initiative.

The decision to build and fund a cloud computing strategy is one that must be driven by the highest levels of your organization. Making the decision to move from a traditional in-house technology strategy (possibly a "bare metal server approach") to a cloud computing strategy represents substantial exposure for the organization in the form of risk and expense, but the benefits of a successful cloud computing strategy will far outweigh that risk and expense. Those benefits include the potential for greater innovation, faster time to market, greater market share, and greater customer loyalty.

In this chapter, a brief history is given to introduce how the evolution of technology has brought us to cloud computing. Special attention is given to service-oriented architecture and why it is important, and to virtualization and why the reader should care about it. This brief history will set the stage for Chapter 2, which will define cloud computing and its delivery models and describe how those models fit into an organization. Cases are given for many of the difficult concepts so the models of delivery for cloud computing services can be understood. These topics provide the foundation for understanding how cloud computing can help an organization attain its strategic orientation.

THE RISE OF CLOUD COMPUTING

As already said, cloud computing is a business model and a new way to deliver IS and IT. That new delivery is through a "pay as you go" model. To understand how we got to cloud computing, it is helpful to understand the historical evolution of these foundational technologies. Understanding the history informs us about the pitfalls made by others and gives us a foundation on which to base our forward-looking strategies. Reviewing this history will help us understand how cloud computing creates business strategy. (This history will be an overview of some of the major technology antecedents to cloud computing, not a detailed technical review.)

Computing Hardware

Changes in computing hardware have been a large driving force in how IS and IT are delivered to the user. In early computing, the mainframe was largely the "computer in the backroom" approach, in which the users did not have any direct access. Today, the users have direct access and have more processing power on their desks than what was in those early mainframes. In this section, we will examine some of the trends in the last 60 years.

Early Computing and the Mainframe

Through the 1950s and 1960s, large corporations adopted mainframe computers to process their data. The integrated circuit (IC) was

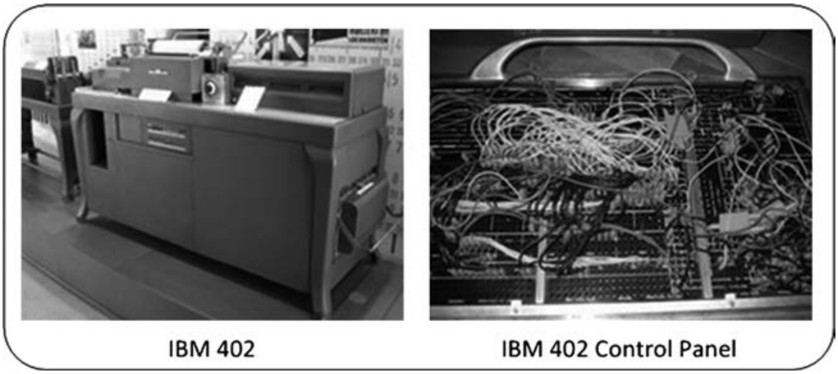

Figure 1.1 IBM 402 *Source*: IBM 402 photo courtesy of Stahlkocher. Control panel photo taken by Chris Shrigley in May 2003.

developed in 1958 by John Kirby and Robert Noyce.[1] In the ensuing years, the IC was reduced in size and cost, making it the technology that would eventually enable the creation of the microcomputer and minicomputer. Before the IC, discrete transistors were the building blocks for mainframes. Terms such as *business intelligence* (BI) and *Big Data* had not been conceived yet, nor was there the need for technology to support them. The mainframe was created to support large organizations in their need to process their data.

By today's standards, these early systems were difficult to use. They had no direct human interface (i.e., no terminal or keyboard) but were given data and programs through hard wiring, punch cards, or paper tape. The IBM 402 (see Figure 1.1) was such a machine, but many other examples also existed.

In these early days of modern computing, "IBM and the seven dwarfs" dominated the computer industry; the "seven dwarfs" were Burroughs, Control Data Corporation, General Electric, Honeywell, National Cash Register (NCR), RCA, and UNIVAC. In the early 1970s, General Electric sold its computer business to Honeywell, and RCA sold to Sperry. The remaining companies became known as "IBM and the BUNCH"; BUNCH was an acronym for Burroughs, UNIVAC NCR, Control Data Corporation, and Honeywell. These companies formed the foundation of modern mainframe computing.

Mainframes remain in use today; they provide *big iron* processing power in organizations that process large amounts of data. These

Table 1.1 Mainframes and Supercomputer Attributes

Mainframe	Supercomputer
■ Handles large amounts of data quickly ■ Good at processing a large number of business transactions quickly	■ Handles large amounts of data quickly ■ Good at processing data and performing scientific number crunching

machines are powerful and able to handle large amounts of data quickly. This makes them especially suitable for things like back-end processing to support a government need to handle its tax returns quickly or an organization's need to process a large inventory in an enterprise resource planning system. Mainframes are good at moving large amounts of data and processing them quickly so other users and other systems (perhaps personal computer–based systems) can further process the information. The mainframe exists as the right tool to do the right job. It's a matter of engineering and architecture: A personal computer (PC) using a high-end graphics card is better at displaying images and streaming video than a mainframe, but a mainframe is better at processing large amounts of transaction-oriented data.

It is important not to confuse mainframes with supercomputers. Both are good at handling a large amount of data quickly, but they have different purposes. Supercomputers are fast at processing data and performing scientific number crunching. Mainframes are good at moving large amounts of data and processing large numbers of business transactions quickly. This is summarized in Table 1.1. With the emergence of cloud computing, multiple servers can be used in unison to approach or even exceed the processing power of a mainframe, but it is a matter of architecture, cost, and business benefit.

Minicomputers

Through the mid-1960s, a class of computers emerged called *minicomputers*. In 1970, the *New York Times* proposed the definition of a minicomputer as a machine costing less than $25,000. Those machines had to have both input and output devices, at least 4,000 words of memory, and be capable of running programs in a computer language such as Fortran or Basic. The minicomputer was built differently from the mainframe and brought computing power to smaller organizations that could not make the investment for a big iron mainframe, but that still had the need to process more data than a human could do in a reasonable amount of time. These machines did not normally have input and output devices for each staff member in the organization and were mainly used for batch processing. As the minicomputer and its software evolved, and as ICs made devices more accessible, some small and medium-size businesses were able to deploy the minicomputer with individual terminals for staff members, but this was expensive and often not cost-effective.

Microprocessor-Based Computers and the PC

As the use of mainframes and minicomputers became common, users began to develop the need to process their own information—especially considering the large amount of information they needed to handle to do their jobs effectively. Mainframes were too unwieldy and not accessible to the normal staff member in an organization, and minicomputers were too expensive for each staff member to have one.

The development and pricing of the IC, along with other technology trends, made PCs accessible to more people and made small scalable servers a reality. The first PCs were introduced in the 1970s by manufacturers like Coleco, Radio Shack, Commodore, and Atari. For several years, PCs from these and other manufacturers were largely seen as "hobbyist" computers. There was also a doubling of the number of transistors on a processor chip approximately every two years that caused evolutionary developments in processing power and capacity; this phenomenon was known as Moore's Law.[2]

By the late 1970s, the need for all staff members to have access to computing power was better understood. In 1981, IBM announced the PC, and in the same year Microsoft introduced MS-DOS under contract with IBM. There are varied accounts of how MS-DOS was created, ranging from it being a spin-off of the earlier Quick and Dirty Operating System (QDOS) to it being copied from Gary Kildall's CP/M operating system. Microsoft founder Bill Gates convinced IBM to let Microsoft retain the rights to MS-DOS and to market it separately from the IBM PC. Subsequently, in 1985, Microsoft introduced the first version of Windows. Coincident with these events, Steve Jobs introduced the Apple Macintosh (Mac) computer in 1984. The Mac, too, evolved through multiple hardware and software versions. This series of events began the revolution toward desktop computing.

Networking and HTML

While mainframes, minicomputers, and desktop PCs were coming into vogue, another phenomenon was occurring: People were developing ways to connect computers so they could transfer messages to one another. In 1969, the development of ARPANET (which some say is the grandparent of the Internet) began. ARPANET is the network of the Defense Advanced Research Projects Agency (DARPA). DARPA is a branch of the federal government that specializes in "the technological superiority of the U.S. military and preventing technological surprise from harming our national security by sponsoring revolutionary, high-payoff research bridging the gap between fundamental discoveries and their military use."[3] ARPANET was built primarily to give researchers access to mainframes that were too far away from their work locations to make frequent access convenient. ARPANET was the first large-scale packet-switched network, and it eventually gave way to the TCP/IP protocol and the Internet.

ARPANET was built to connect four locations: (1) the University of California, Los Angeles (UCLA), (2) the Stanford Research Institute, (3) the University of California, Santa Barbara, and (4) the University of Utah. ARPANET carried its first message on October 29, 1969, from UCLA to the Stanford Research Institute. ARPANET grew, and in 1970 it reached the East Coast by including the technology company Bolt, Beranek and Newman. By 1981, ARPANET had grown to 213 locations, and it began to acquire approximately one new location every 20 days.

In the meantime (around 1975), a transatlantic satellite link was added to ARPANET, and the Norway Seismic Array was brought online. In 1983 ARPANET was split and the military maintained its own network, MILNET, for unclassified communications. The combination of networks for unclassified military and civilian traffic was called the Defense Data Network.

In 1985, the National Science Foundation (NSF) began a program to coordinate its projects over the National Science Foundation

Network (NSFNET). This name was also given to a group of nationwide networks that were constructed from 1985 to 1995. NSFNET was initially created to link the nation's NSF-funded supercomputers, but through collaboration between the NSF and industry partners, the NSFNET became a major part of the Internet. This backbone was connected to a number of regional networks that enabled smaller regional and campus networks to connect to one another through the NSFNET.

The federal law that appropriated the funds for the NSFNET was interpreted to mean that commercial traffic was not allowed. In order to make the appropriate use of the NSFNET known to those connecting to it, the NSF developed an acceptable use policy (AUP). The AUP restricted commercial use even though several commercial Internet service providers (ISPs) were connected to the NSFNET. A number of the early NSFNET industry partners (notably, Merit, IBM, and MCI) created the Commercial Internet Exchange to connect many of the private ISPs, relieving the restrictions imposed by the NSF AUP. The NSFNET became a transitional network between ARPANET and the Internet as we know it today.

Simultaneous to the ARPANET, NSFNET, and Internet transition, standards for local area networks (LANs) were developed. LANs were built to allow computers within organizations to share data and devices such as printers. In the 1970s, there were a number of competing standards—Ethernet, Token Ring, and ARCNET, to name a few of the major ones—and all of these technologies had proponents and a place in the market. With the proliferation of desktop PCs, easy-to-use (compared to the mainframe and minicomputer) operating systems like CP/M and MS-DOS bringing computing power to the desktop, and the user's need to share devices and data, the implementation of LANs grew.

Parc is a Xerox company that describes itself as a technology platform pioneer.[4] Robert Metcalfe and others at Xerox Parc developed Ethernet in the early 1970s, and it was patented in 1975. Because of its ability to adapt to the PC market and its low cost, Ethernet has become the de facto standard for desktop PC and server interconnectivity. Ethernet has been standardized through the Institute of Electrical and Electronics Engineers 802 working group and has evolved from the slow (by today's standards) form of connectivity it initially allowed to the advanced networking we use today.

Another development was the creation of hypertext transfer protocol (HTTP) and hypertext markup language (HTML) by Tim Berners-Lee. Berners-Lee saw the ability to use the newly formed Internet and its supporting protocols and technologies to communicate marked-up text. The proposal and development of HTTP and HTML occurred in the late 1980s and early 1990s, with the first web site ever deployed being that of the European Organization for Nuclear Research in 1991. Berners-Lee went on to create the World Wide Web Foundation (later to be called the World Wide Web Consortium [W3C]) to launch transformative programs that build local capacity to utilize the Web for positive change.[5] The evolution of HTTP and HTML directly contributed to the ability to deliver cloud computing services.

All of these technologies—and many others—have converged to provide the Internet communications infrastructure that is in place today. Companies like Level3.com provide Internet connectivity; AT&T, Verizon, Clear.com, and many others provide mobile connectivity; hardware manufacturers like Cisco provide WiFi and other communication devices; the plain old telephone system has evolved into a digital communication medium; and numerous other companies and providers deliver technologies that converge today to create a global infrastructure that enables us to work (and play) virtually anywhere in the developed world.

Bandwidth

Bandwidth is the rate at which a computer network can transfer information between devices. In this case we are specifically discussing bandwidth sold to businesses that allows them to make connections in their information and communication technology (ICT) infrastructures. Bandwidth is often measured in bits per second or some denomination thereof (e.g., megabits per second, gigabits per second).

As computing platforms developed and the Internet emerged, the cost of bandwidth declined dramatically. The bandwidth marketplace has been inundated with ambitious (some say overambitious) network construction, bankruptcies, and dramatic price declines through 2004. By 2005, the wholesale bandwidth market had stabilized, with prices largely holding at the 2004 level.[6] The prices on the more popular transmission routes experienced the greatest decline and subsequent stabilization. Today there is still constant economic and marketplace pressure on bandwidth price structure.

The decreased cost of communication has enabled cheap telephone calls and inexpensive data transmission. This has fueled the cloud computing revolution and is a large reason that cloud computing has become ubiquitous.

Computing Platforms

With this brief review of the evolution of computers, minicomputer operating systems, and networks completed, we can move on to the effect they had on computing platforms and information systems. It is probably fair to say that these technologies created a "perfect storm" so that current computing platforms and information systems could exist.[7] Three computing platforms have significantly contributed to moving us toward cloud computing: (1) clustering and grid computing, (2) service-oriented architecture, and (3) virtualization.

Clustering and Grid Computing

For a long time, humanity has understood the advantage of clustering. An example of this is the way we process language and how that is used in natural language processing (NLP) by computers to make interactions between humans and machines more humanlike. NLP—a field of computer science, artificial intelligence, and linguistics—refers to software that allows a computer to interact with another computer using spoken human language. NLP creates clusters of sound in human language that can be easily understood, analyzed, and processed by computers.

Clustering can also be seen in the way an Internet search engine delivers its content, grouped by category or cluster. Clustering is a way to group things to make classification easier as well as a way to group computer resources together to maximize their functioning. Clustering is therefore useful in many venues.

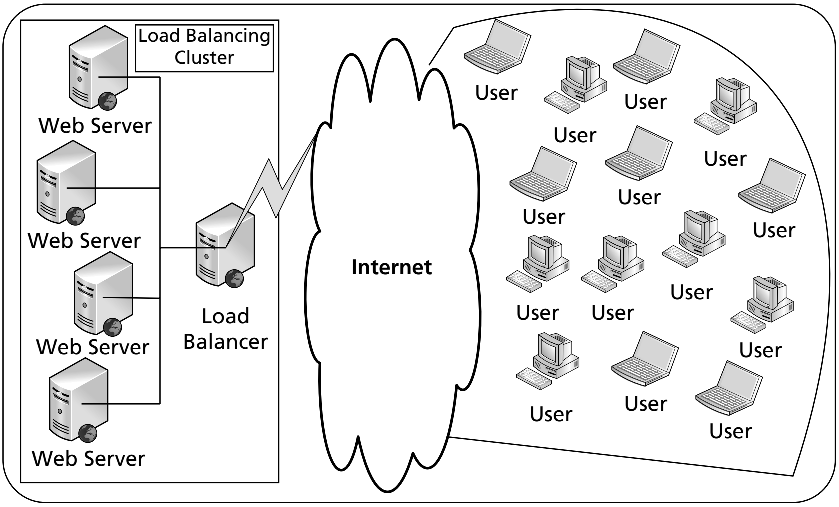

Figure 1.2 Load-Balancing Cluster

Computer clusters are architecture rather than a type of computer. Many types of computers can be clustered together to maximize their use and productivity—think of the adage "the whole is greater than the sum of its parts" applied to computer technology. These clusters can be as simple as two PCs loosely connected so they can share a task or provide backup for each other should one go down. Alternatively, it can be as elegant as a supercomputer composed of a linked set of smaller computers. Clusters are also used to group computer memory and disk drives, which is often referred to as *pooling*.

All computer clusters have one thing in common: They all appear to the user to be one device rather than a number of interconnected devices. Clusters are created with software or hardware that orchestrates the functioning of the cluster to make it appear as one system. A common use of clustering (still very much in use today) is the load-balancing cluster used to deliver web services to a group of users (see Figure 1.2).

The cluster consists of a load balancer, which acts as the orchestrator between the user and the servers of the web site. In this example, behind the load balancer are four servers that each offer up the same web site when requested by a user. The load balancer monitors and distributes the workload of each server. In this way, the load balancer

Table 1.2 Grid and Cluster Computing Characteristics

Cluster Computing Platforms	Grid Computing Platform
Interconnected, and look like one system to the user	Loosely interconnected, and resources do not appear to be one system
Administered using middleware to appear as one system	Administered using middleware that distributes the workload across systems connected to the grid
May be proprietary or have open standards	Often has open standards
Normally has a local or proprietary connection	Can exist on a local or proprietary connection or can interconnect using the Internet

acts as *middleware*, or the orchestrator that manages the servers so their functioning can be maximized. To the user, it appears that he or she is interacting with one web site when in fact there may be a number of web servers that stand ready to deliver the requested web site. It's up to the load balancer to determine which available web server actually does the task.

An architecture that seems similar is grid computing. *Grid computing* platforms have a number of distinguishing characteristics and are compared to cluster computing platforms in Table 1.2. What largely distinguishes a grid from a cluster is that a cluster appears as one system to the user, whereas a grid is true distributed processing. Both clusters and grids use middleware to orchestrate a common task. In our example of cluster computing, the common task was seamlessly delivering a web site to the user. A grid may be used to work on parts of the same problem. A typical grid is shown in Figure 1.3, with one computer that distributes a workload to a number of other computers attached to the Internet.

Service-Oriented Architecture

When service-oriented architecture (SOA) entered the scene, we had already seen all the network standards, grid and cluster computing architectures, and other changes to business and technology that we have discussed thus far. SOA is a software engineering technique

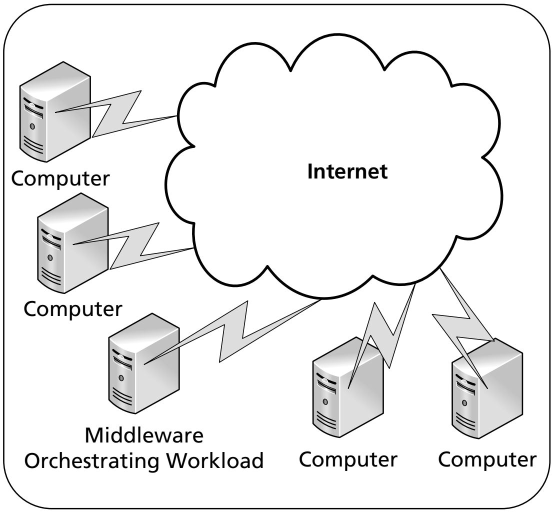

Figure 1.3 A Typical Internet-Based Grid

directed toward designing and building computer software services that interoperate. It was a new way to build computer software that allowed IS developers to build software from components of a business process rather than needing to "reinvent the wheel" each time software was created. This let each organization that provided a service do what it did best, and most cost-effectively.

SOA is a complex but extremely useful and efficient architecture for software development, and many people believe that it was the direct predecessor of cloud computing. Although it is a somewhat technical discussion, we will review some of the concepts that underlie SOA:

- SOA is based on a set of loosely coupled services. An example of this is a web site that uses a service from one vendor to provide shopping cart functionality and uses a service from another vendor (e.g., a bank) to provide credit card validation.
- Each service in an SOA application communicates and exchanges data in well-defined formats or protocols, often XML. These well-defined formats and protocols allow for standardization and interoperability.
- SOA can be seen as orchestration of services from different organizations or vendors. In the example of a web site with

a shopping cart and credit card validation, there would be an orchestrating layer (perhaps the user interface of the web site) that calls and interacts with the services (i.e., the shopping cart and credit card validation).

- Two standards, both based on XML, were initially used to describe the services. Web services description language (WSDL) was frequently used to describe the service itself, and simple object access protocol (SOAP) was frequently used to exchange information between the orchestration layer and the services it calls. Other standards exist (i.e., REST) today.
- A hallmark of SOA is the ability to reuse services from one application in another. In our example, any web site that wanted to use the shopping cart functionality could do so, with proper authorization from the service provider.
- A common implementation of SOA is the web services approach. In that approach there is a service provider that exposes a service so the service consumer can access it.
- Service consumers consume the offerings of a service provider through a software wrapper, which can be a web site, an application, or other software that orchestrates the consumption of the services.
- Services are often provided on a pay-as-you-go basis (e.g., each credit card validation transaction is charged a fee).

IBM provides a convenient conceptual model that describes the SOA architecture as the interaction of three primary parties:

1. The service provider, who publishes a service description and provides implementation for it.
2. The service consumer, who consumes (uses) the service.
3. The service broker, who, if present, maintains a service registry describing the service. Today, the broker is often referred to as the *managed service provider*.

SOA provides key elements that we see in today's cloud computing environments, including software reuse, a pay-as-you-go (or metered use) model, and orchestration. These are very similar to the cloud computing software-as-a-service (SaaS) model, which we will discuss in Chapter 2.

Virtualization

In the earlier stages of computing discussed in this brief historical review, hardware was a single resource—one mainframe meant one mainframe, one server meant one server, and one storage device meant one storage device. There was time slicing (i.e., multiple concurrent operations) on mainframes and minicomputers, but that did not carry forward to the PC. Eventually, we understood that treating each resource individually did not use it effectively; each device had excess capacity that could not be used.

The solution to that problem was virtualization. In computing, *virtualization* means creating copies of something that all share the same physical resource. For example, suppose you have a server (a single resource) and know that that server is being used to only 20 percent of its capacity. It is possible, through virtualization software like VMware or Microsoft Hyper-V, to take the single resource and make it appear like many copies of the same or similar resource.[8] Each copy will execute on the single server, but to the user each copy will appear to be its own dedicated resource.

Three types of virtualization are common: (1) hardware virtualization, (2) desktop virtualization, and (3) application virtualization. These share the following attributes:

- Each virtual instance shares the same hardware. For example, one physical server can be made to look like many physical servers (e.g., a VMware solution); one physical server can be made to provide computing power for many desktops or application (e.g., Microsoft Hyper-V, Citrix, XenDesktop, and XenApp).
- Virtualization consolidates servers to improve utilization and cost-effectiveness; by maximizing the use of the existing resources, the existing hardware can be used more cost-effectively. This also enables an organization to decrease its carbon footprint. The environment and the organization both benefit from lower energy consumption.

- A major advantage of virtualization is that the infrastructure becomes more scalable. The existing hardware can have more virtualized instances added and dropped as needed. This is especially helpful when multiple users are sharing the same physical resource; this is discussed more in the next point.
- Resource sharing often drives virtualization and makes it effective. The fact that multiple users can access the same physical resource and have it appear that each has his or her own resource makes virtualization attractive. Add to this the ability to charge a user by the amount of each resource consumed, and you have an effective business model.
- Virtualization is normally autonomous and based on demand. A virtualized infrastructure in which you purchase the computing resource (e.g., disk or compute power) will sense your usage and scale those resources as needed. As those resources are added or removed, the amount charged is adjusted. This represents a utility model, not unlike the local electric or telephone utility that charges you for what you consume.
- There are unique security issues in a virtualized environment that do not exist in a nonvirtualized infrastructure. For example, in a nonvirtualized environment, each resource may map to one IP address; moving this same environment to a virtualized infrastructure means that each physical resource may have many virtualized resources that it supports, each with its own IP address.

Today we see a mix of nonvirtualized environments, virtualized hardware, and virtualized desktop or application infrastructures implemented. To summarize: hardware virtualization mainly involves making one device appear like many copies of that same (or a similar) device. Desktop virtualization involves making one device (i.e., a server) deliver the entire desktop to multiple client devices simultaneously; application virtualization involves delivering a specific application or set of applications (over a web browser or through another client) to users. An example of each is shown in Figure 1.4. Desktop hardware in a virtualized environment can be less expensive, and centralizing

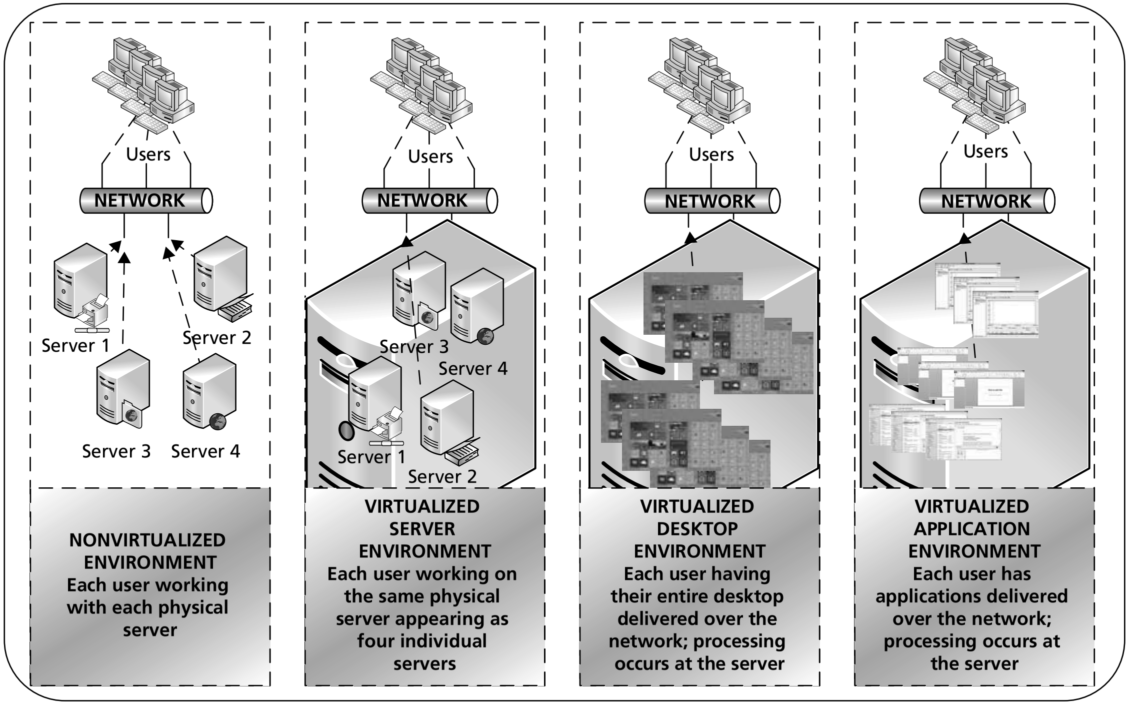

Figure 1.4 Virtualized Server, Desktops, Application

desktop and application delivery can make management of these resources easier and cheaper.

Cloud Computing

Cloud computing evolved from, and has been built on, the technologies discussed in this chapter. At the start of this chapter, a case was made that cloud computing is a new way to deliver IT and IS, and to do business; it is not just a technical orientation. Moving applications to a cloud rather than having them deployed on bare metal servers takes a decided approach that is fundamentally different and can require a change in an organization's strategic orientation. This is especially true when an organization decides to move from delivering all information systems from its in-house technologies and staff to an external cloud vendor. The recent history of technological innovation has prepared us to deliver IT via the cloud through infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS). These are discussed in Chapter 2.

CONCLUSION

Looking back over the changes in computing technologies during the last 60 or so years, it is easy to see how we arrived at cloud computing. Functionalities that we think of today as commonplace—like OneClick shopping at Amazon.com, MyLowes at Lowes.com, and online credit authorization—were unheard of before the Internet, HTTP, the World Wide Web, and now cloud computing.

There are many versions of history, and the one presented here is based on the author's own experience in the computing industry as well as some reference materials to assist with dates and other specifics when the timeline might be blurred in his memory. This history is an accurate representation of how we got to cloud computing and what it took to get us there. Hardware, networking, data communications, and software architecture have evolved, and they have come together to bring us to where we are today: the delivery of IT and IS using a cloud computing approach. We are standing at a precipice:

cloud computing has emerged as the dominate way to deliver IT and IS when resource utilization structures must be managed.

Gone are the "field of dreams" days when we built ICT with the hope that the users would find a way for it to add value to the organization and its customers. Today we must ask what the objectives of an ICT project are, how the project can add value, and what resources the organization has to apply to that project.

Before we leave the discussion of how we evolved to cloud computing, it is appropriate to talk about how our current global economic crisis has affected that evolution. The recent fiscal crisis is a major socioeconomic force on technical developments that have pushed us to a greater source of elasticized value with new IT and IS delivery models like cloud computing. It has created a realization that businesses must be more aware of their own budgets and the effect of technology on them—in short, businesses need to find more economical ways to deliver IT and IS. Cloud computing is the next step in computing evolution and will be here to stay. As organizations move from traditional data centers to consuming cloud services, this new way to deliver IT will only become more prevalent.

The historical overview given in this chapter sets the stage for understanding how we got to where we are today, with cloud computing as the inevitable outgrowth of many factors. However, Chapter 1 has only set the stage. That stage is necessary for understanding how and why cloud computing makes sense. The rest of this book will define cloud computing, deal with specific issues affected by cloud computing, and discuss the strategic implementation of cloud-based solutions. As pressure increases to deliver IT and IS in a way that supports business agility, it is more important than ever that IT and IS staff, managers, and the C-suite have the tools to understand and build what is expected. In addition, the growth of social and mobile technology adoption is adding information overload and creating a need for more IT and IS resources to run enterprises and build business intelligence. This book introduces the strategy-oriented concepts and tools required to create cloud-based infrastructure that fulfills the ICT value proposition and builds business intelligence.

NOTES

1. M. Wright, "The Planar IC: Revolution Underestimated," *Edn* 51, no. 9 (2006): 38; and M. Wright, "Miniaturization Enables Innovation—Past, Present, and Future," *Edn* 51, no. 20 (2006): 172–86, http://0-search.proquest.com.www.consuls.org/docview/222434808?accountid=9970.
2. "Moore's Law Inspires Intel Innovation," Intel, www.intel.com/content/www/us/en/silicon-innovations/moores-law-technology.html.
3. "About," DARPA, www.darpa.mil/About.aspx.
4. "Parc Today," Parc, www.parc.com/about.
5. "Frequently Asked Questions," World Wide Web Foundation, www.webfoundation.org/faq.
6. E. Schoonover, "Wholesale Bandwidth Prices Stabilize," *Business Communications Review* 36, no. 7 (2006): 14–17.
7. The *perfect storm*, in colloquial terms, is a set of circumstances that when brought together create the environment for a specific effect. The term is often used to describe weather patterns that come together to create the environment for a particularly strong storm. A business or financial example is the use of the term for the recent global financial crisis.
8. VMware, www.vmware.com/; and "Microsoft Hyper-V Server 2012," Microsoft, www.microsoft.com/en-us/server-cloud/hyper-v-server/default.aspx.