18

WEB SERVICE'S ROLE IN THE SOA/SONA EVOLUTION

DEFINITION OF WEB SERVICES

Web Services is changing how businesses and customers interact. Through the use of common protocols and standards developed as a direct result of the Internet boom of the late 1990s, businesses can package internally used siloed business applications as a service available over the network (Web Services), making them available to their customers. A common conception is that "Web Services" is an "Internet"-only-based delivery. One must keep in mind that the Internet is a network; fundamentally the same base networking technology used in the public "Internet" is also used for private and internal networks. Many of the standards that define Web Services, such as XML, are not "Internet"-exclusive technologies. They are generic methods, XML specifically, that describe data (messages and state) of a service that can be used by any other system. The same "Internet Web Services" delivery protocols apply equally to private and internal networks. The changing force potential of Web Services determines how businesses interact with all customers, internal and external alike without any distinction.

The impact of Web Service technology on how information technology (IT) organizations operate is vast. For Web Services, allowing disparate systems to interact with each other by leveling the playing fields of protocol and data sharing therefore minimizes if not totally eliminates the human interaction that is required for many traditional (non–Web Services) system integration efforts. The "business–customer interaction radial effect of Web Services" (see Figure 18.1) encompasses intra- and interorganization business service offering for consumer

Distributed Data Management for Grid Computing, by Michael Di Stefano

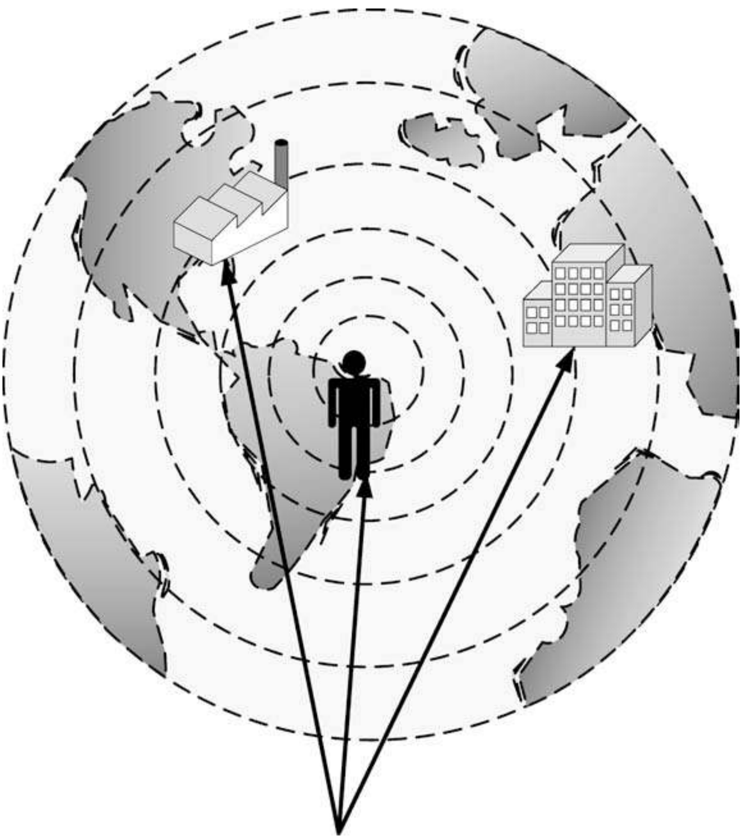

Figure 18.1. Business–customer interaction radial effect of Web Services.

service consumption. This is a "geographically independent" boundary of producer–consumer service relationship.

More and more buzzwords are seeping into our discussions, and it is time to step back and level the field of regarding we mean by producers and consumers when talking about service-oriented architectures (SOA) and Web Services. The *producers* of a service are those who manufacture the service and offer it for consumption or sale to others. It is the "who," the consumer of a service, that has a broad impact. If you are like me, the term "consumer" brings the image of someone purchasing a good such as groceries, clothing, telecommunications, or electricity. However, in the scope of Web Services, the consumer extends beyond the public market to encompass any person or group of people interacting within a business or businesses or with other businesses. This creates a new business interaction leading to new business opportunities and efficiencies not achievable otherwise.

I have provided two definitions for "Web Services" from leading organizations that are helping to shape the Web Services landscape:

1. The first is from W3C, The World Wide Web Consortium (www.w3.org). The W3C charter promotes evolution of the World Wide Web by developing common protocols (there are over 450 members of the W3C). The following excerpt is from W3C Working Draft 8, August 2003 (http://www.w3.org/TR/2003/WD-ws-arch-20030808/), *Web Services Architecture*, Section 1.5—"What is a Web Service":

> A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

2. The second definition for Web Services is from: Dr. Bob Sutor, Director of Web Services Strategy, IBM, *The Definition of Web Services*, (source: SearchWebServices.com; available online at http://searchwebservices.techtarget.com/originalContent/0,289142,sid26_gci874060,00.html).

> Web services provides standards for an electronic envelop, a language for describing how you talk to a service and what it says back, plus techniques for publishing and discovering these descriptions.

Now with our new understanding of what Web Services means to different people and how organizations and standards influence our thoughts, it will become evident that SOA consists of Web Services, the compute utility, as well as the command and control scenarios presented here.

DESCRIPTION

Real-life businesses are a connection of processes, from manufacturing, sales, delivery, and payment as examples. An event in any one part of this chain has major repercussion on the other events. Corporate business units in partnership with IT organizations are leveraging technology and its associated methodologies of distributed computing to create information systems that meet the business demand. Distributed computing is the transformation of manual, or silo-based processing of information into cohesive real-time information sharing of event-driven enterprisewide systems.

Methods to address information sharing between disparate systems start with file sharing or pipe (or sockets) communication protocols to establish connections between systems (in much the same way as we use telephones). The communication can leverage message queues, through the advent of "middleware" technology such as CORBA or messaging for the encapsulation of enterprisewide components and services. Figure 18.2 is a timeline showing the evolution of client/server communication to today's grid and Web Services.

The evolution of distributed computing continues as the encapsulation of services extends to the corporate infrastructure that delivers "business" services. For each new application a family of new hardware, software packages (i.e., databases), administrative policy and procedure, and human resources must accompany it. The next generation of distributed computing products and methodologies extract the physical process and the administrative layers of the applications into an enterprise service. This new area of distributed computing creates grid services (which consists of compute and data grids) in support of Web Services for the delivery of business services.

In other sections of this book we discuss the practical application of the data grid, for example, the feedback control loops and the compute utility service. These kinds of processes collectively constitute some of the key components that make up Web Services and the SONA architectures. For such services, flexible, fungible, and scalable infrastructures are essential.

DATA MANAGEMENT: THE KEYSTONE TO WEB SERVICES

Web Services brings together various software components, each individually built to deliver some level of service to the business, in such a way as to extend those services for a broader purpose and to a broader audience. This implies an infrasturcute that allows these components to interact and share information and provide process management, and requires a flexible underlying physical layer capable of provisioning its resources to meet the service demands. Key to most, if not all, aspects of delivering Web Services is the *data*. Some points regarding Web Services and its data are as follows:

- Web Services is close to the business, where business process implies business state; thus *state means data*.

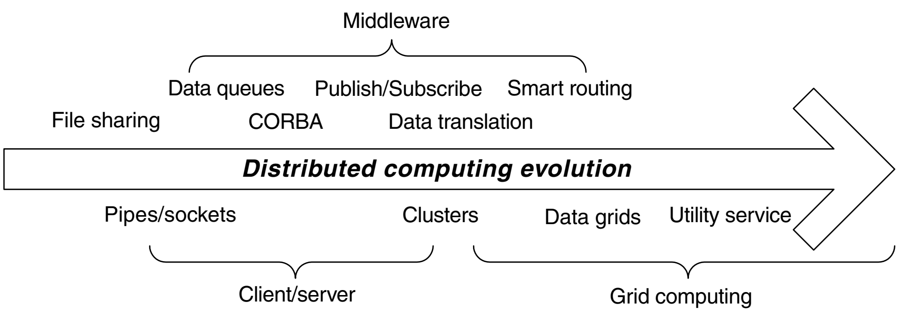

Figure 18.2. Distributed computing evolution.

- Since Web Service components interact with each other, they require *data sharing.*
- Components interacting with each other require a broader level of process management, which means the *management of state, which again implies data.*
- To deliver Web Services to broader audiences, flexible and fungible infrastructures are required. The goal of a compute utility is to become such an infrastructure through the collection of *metered* (*state*) *data* from all the nodes that make up the utility, therefore describing the *complete state of the utility*.

The common theme is that the closer one gets to the business, the more important data become, not just to the business application itself but also to the means of delivering the data to the application. Here we talk about data requirements as state management. Therefore state management is associated with two aspects: (1) the internal state of the business application being delivered as a Web service to the user community and (2) the state of the mechanisms needed to successfully deliver the Web Service. Figure 18.3 highlights the interaction of Web Services with business processes, state management, and the data management or data grid.

The compute utility service enables business compontents to be delivered anywhere on the network. With the data grid as an integral part of the utility, it also provides the means to deliver state management for the Web Services components as well as the process and infrastructure management. Figure 18.4 illustrates the relationship between the grid utility service and Web Services.

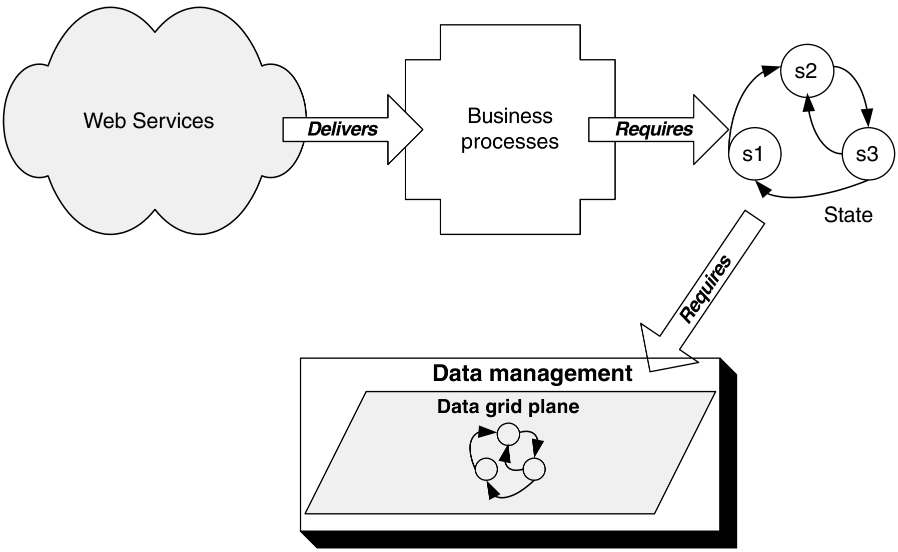

Figure 18.3. Business service, state management, and the data grid.

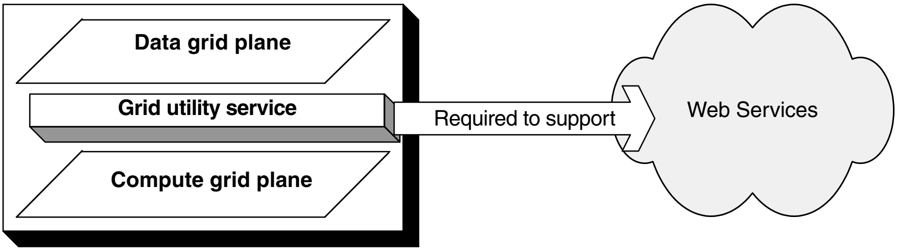

Figure 18.4. Grid utility service as a requirement for Web Services.

WEB SERVICES, GRID INFRASTRUCTURES, AND SONA

It is always beneficial to understand our own history so that we can see clearly where we need to head. Web Services, the grid, and service-oriented network architecture (SONA) are instances of importance of the past that lead us into the future. We will see that the direction of information technology closely parallels the evolution of other point-to-point services that have transformed the services into commodities such as telecommunications and electricity.

The Undiscovered Past

We all know about Moore's law and the fact that the Internet was created by Dr. Metcalfe but not how the Internet evolved from the works of the brilliant scientists and mathematicians of the twentieth century. It was not until authoring this book that I, too, came to appreciate the facts about the commoditization of information services into the consumer marketplace. All of us currently building our careers in information technology, like the seasoned veterans and students, should spend some time studying our rich history so that when we go to apply the Weiner process we can appreciate the beauty of our IT mosaic and the woven foundation on which it was built.

We will also see that the same fundamental laws and principles that has evolved the computing technology since the mid-1940s is just as relevant in moving us into the mid-2060s, including. Shannon's communication theory, the operational research and control processes of Weiner, Amdahl's law of locality of reference, Moore's law driving the cost of the CPU to zero, and Metcalfe's law of increased value of the total number of users. John Narghton's book, *A Brief History of the Future*,[6] does an excellent job in reminding us of just how we have arrived here. Below is just a brief look into the building blocks of the technological and communications evolution that we have experienced since the mid-1940s or so.

From the 1930s to today we have seen a series of progressive transition in the digitalization of data. The increasing power of the computer (i.e., Moore's law) has led to an even larger increase in the pervasiveness of information technology in our everyday lives. Access to information that was once limited is now almost transparent, creating a cycle of information creation and consumption and in some cases information overload, causing the value of the system to increase at exponential rates (Metcalfe's law). In the 1920s and 1930s computation power was limited as well as expensive; for example, telecommunication service was a manual point-to-point connection with a human switchboard establishing the physical circuit connection between two parties. During the 1950s and 1960s, as computation power increased and became more readily available, coupled with new areas of research such as *operations research* (a branch of applied math developed by scientists such as Norbert Weiner) and *agents* (an entity that has been empowered to make decisions on behalf of another entity), circuit switching emerged. After that, circuit switching, packet switching, and the Internet emerged in the 1970s. In the discussions later in this chapter it is important to understand the mechanics of the Internet. Thus the following points need to be highlighted:

- *Switching*. Switching is a signal that triggers the policy decisions that are required to commit resources to the connection. In this context, the term "agent" refers to the transmitting/receiving device—these, along with the switching infrastructure that is delegated decision making authority in terms of resource scheduling and prioritization. In computer-based communication, these transmitting/receiving devices are software-based; producing or consuming messages. These software agents may be realized as either a process or a component running on a thread in that process.
- *The Layers of the Internet*. The Internet is a layered architecture that inherited characteristics of Selfridge's "daemon" and OSI's reference model. The Internet protocol suite consists of four layers: application, transport, Internet, and data link. In the Internet architecture, these functions were purposely deferred to the application. This boundary is enforced via the "Berkeley socket API." Applications specify the configuration of port, host, transport, and so on via this API. The session layer (OSI's fifth layer) resides above the transport layer and is subsumed either by the application or the Internet's other three layers.
- *Destination Addressing*. Destination addressing is described by an IP address by either the destination's daemon process or the application service. The address also includes the use of a specific port number. Together, the IP address and the port number are analogous to a telephone number (another vast utility network).

In the 1980s and 1990s we saw an emergence of ad hoc "middleware" that started to form the initial concept of delivering a "service," but only to those producers and consumers who subscribed to the specific middleware's internal infrastructure. (*Note*: this progression of information technology also saw a progression of professional titles from applied mathematician, to system analyst, to system model architect.)

In the late 1990s the Internet craze left in its wake HTTP and XML as represented by DTDs and the likes of "Web Services." Web Services can be defined as the externalization of the service independent of its internal representation. At the same time, we saw the emergence of "grid computing," which is an evolution in distributed computing that virtualizes the physical computer and how it manages its resource and task processing. Together, grid computing, forming the "compute utility," and Web Services, externalizing the production and consumption of services, form the bridge necessary for us to cross the chasm from siloed business applications and data centers to a market-driven economy of supply and demand of IT services.

The SONA Model

The general problem of connecting consumers and suppliers of a product or service is one of logistics. The observation is that the logical connection between a consumer and supplier is point to point. In practice, a point-to-point connection does not scale, as the number of conusmers increases the complexity of managing and the associated costs to deliver the service increases. This does not allow for an efficient market for supply and demand. Considering how any commidity that is delivered over a logistical transport, (roads, telephone communication, energy, etc.), the supply of the product or service goes from multiple sources to some consumption field. This proven method of product/service delivery, connecting customer to supplier, increases the availability to better meet the demand, lowering the price and thus improving the efficiency of the market. See my discussion of supply and demand economics in Chapter 3.

In the world of computing, looking back to the 1950s and 1960s, the application was pointed toward distributed computing; the network layer was constrained primarily by the CPU. The cost of the CPU was high; therefore a limited amount of computation was available for data switching. This has been the case until relatively recently. The application of Moore's law has brought the cost of CPU mips (millions of instructions per second) to near zero in relation to the costs of the software and data movement across a network. Therefore, with the cost of computation near zero and the cost of data high, why not migrate the computation to data rather than data to computation. In effect, this would entail a shift back toward a centralized data center. The problem is that a point-to-point connection as implied by a physical centralized data center will not scale in today's Internet. Enter the data grid. The data grid, in essence, provides the virtual data center. The data grid today is about locational services and replication, how to move data effectively and efficiently, and how to find the best place to run a computation. This is similar to packet switching of the 1960s and 1970s. With packet switching, the intent is to move packets of data (scheduling of packets to an IP address) rather than today's data grid, which moves large amounts of data.

The key to any switching layer is source and destination, creating any number of paths and optimizing on the shortest path. Creating a virtual circuit of source and destination allows for choices and flexibility as to where data are serviced, thus requiring a service to locate data as well as a service to replicate data so as to put data proactively out near the consumption. The same is true for the data grid in creating a virtual data center; the data grid allows for data location independence and replication of data making them readily accessible when needed, with centralized management and control, yet enabling people to interact independently on ad hoc communications.

This bears repeating. The same fundamental laws and principles that have driven computing technology since the mid-1940s are just as relevant in moving us into the future to the mid-2060s.

Connecting the Dots of the Past into the Continuum of the Present. The history of technical communications is closely linked with the evolution of computing. The history of computing can be divided into two branches, one originating in the mainframe labs of IBM and the other in the network labs of AT&T. To understand the accelerated innovation of the present, one needs to look to the past. Specifically, the points in history in which we are interested which originated from the application of stochastics to networking and culminated with the present-day Internet. Each step of this evolution laid the foundation for a paradigm shift in our thinking. Claude Shannon originated the notion of stochastics applied to networking. Weaver, Bush, Weiner, and Dijkstra, building on Shannon's work; therefore the application of computing progressed to the command and control of technical communications. Selfridge and Licklieder (the founders of ARPANET, the original Internet backbone) followed with the publication of the application of daemons and agents and dynamic programming to technical communications. The influence of Licklieder and his predecessors can be seen in the architecture and implementation of the Internet as we know it today.

The Internet is based on a layered architecture that inherited characteristics of Selfridge's "daemon" and OSI's "reference model." Although OSI's layered model provides a well-defined mechanism for communications, it leaves one layer, the session layer, undefined. The session layer enables communication and session semantics between applications. It is this application-specific layer, called *middleware*, that received significant attention in the early to late 1990s and has led to a proliferation of nonstandard solutions (the progression of queues, CORBA, and the various flavors of MOM are illustrated in Figure 18.2). It is this layer that will be the focus of the paradigm shift of the 2000 decade. For it has already begun, the implementation of SONA is a nexus of grid computing and Web Services and can be seen in Figure 18.5, as well as at its early stages in PlanetLab. PlanetLab[31] is a consortium of universities and research facilities establishing a worldwide network of hundreds of computers (connected via the Internet) forming a grid that can be "sliced" to run applications. (For more information on the PlanetLab, refer to http://www.planet-lab.org/php/overview.php.)

Figure 18.5 shows the parallels between SONA and the OSI model. Included in the diagram are the policy-based control loops of SONA. The principles of the command-and-control loops were discussed earlier in this book and are a key part of SONA. There are three control loops: the uppermost loop, in the Presentation/Communication Protocol layer, is for "macroscheduling" of the grid. The lower two control loops in the data link/plant scheduling and transport/service delivery layers are for microscheduling of the grid facility. In brief, macroscheduling is

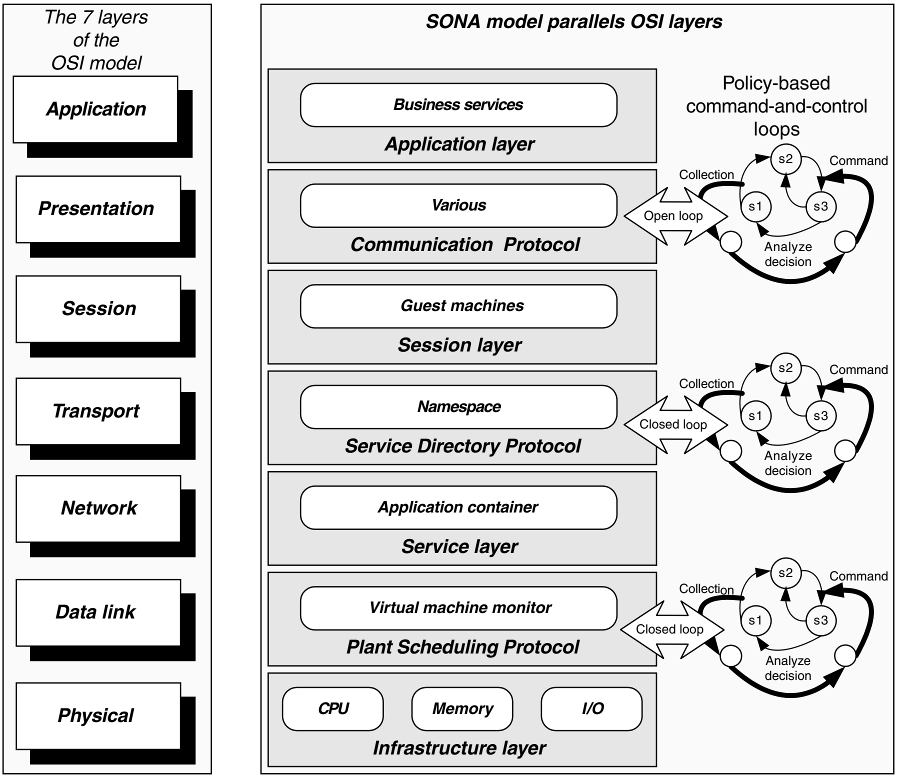

Figure 18.5. SONA layers parallel the OSI seven-layer model.

the long-term planning of plant resources and capacity looking forward for one or more years. Microscheduling involves shorter-term resource allocation, looking forward in periods of quarters, months, and weeks. Macro and microscheduling are discussed in Chapter 19. In fact, we will see that the SONA model is the foundation for the compute utility, forming the lower layers of the physical infrastructure, plant scheduling, service, and service directory.

Service-Oriented Network Architecture (SONA). SONA is an attempt to learn from our past and build on it. The basic principles that have built the infrastructures that we depend on so much today can and should be leveraged to evolve compute services into a consumer-based economic model supported by a utility infrastructure delivering quality product at an affordable cost. We will look at the drivers and the building blocks behind the SONA evolution. Please refer to earlier chapters for a description of SONA layered architecture.

The Drivers. The history and present-day state of technical communications are driven largely by three kinds of forces: macroeconomic, sociogovernmental, and technological. Furthermore, each of these forces is itself influenced by the emergence of new mediums and/or modes of communications. Figure 18.6 illustrates the interactions between such forces.

Economics has always played a major role in technical evolution. Markets drive innovation and are themselves affected by new technologies. Market economies, however, are unpredictable in nature and as such do not apply the same unidirectional force to innovations from decade to decade. At the time of this writing, in the early–mid-2000s, the macroeconomic picture is shaped largely by recession. In particular, uncertainty, dwindling profit margins, rapidly restructured business models, and increased reliance on (and thereby cost of) automation are all independent forces driving the economy.

The current economics impacts technology in several distinct ways. For instance, they place new demands on information technology (IT), including the requirement to satisfy variable demands in service levels, inventory, and people. They also require a greater transparency from IT for resource utilization and cost. The continuing economic uncertainty has generated pressures to rationalize technology investment. Elasticity of IT planning and fungibility of IT resources are two manifestations of economic pressures on information technology.

Societal and governmental pressures have played a major role in driving changes in technology. Computing devices have pervaded the very fabric of society, and as a

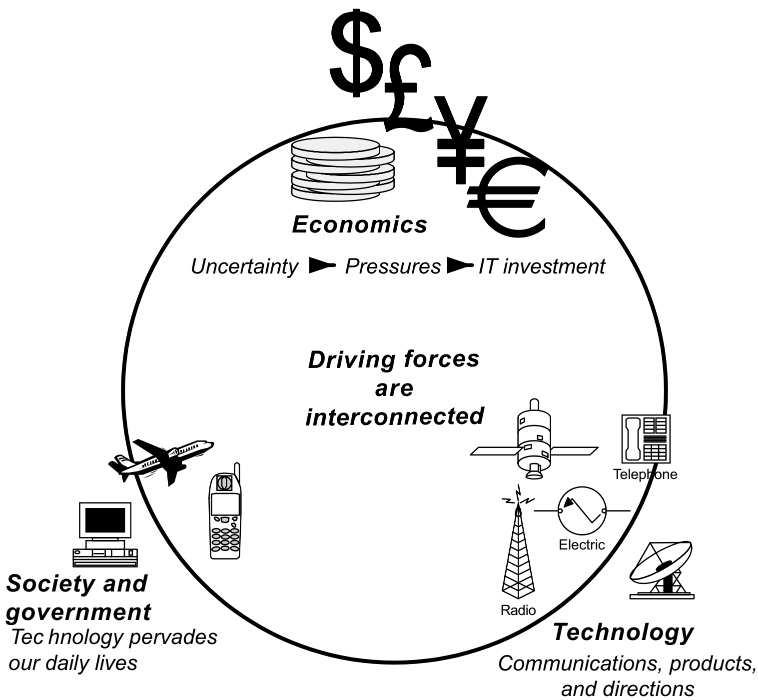

Figure 18.6. Interconnection of driving forces.

result their interconnections have increasingly driven society forward. What would happen if electricity were removed from our lives, for example? We have had very brief glimpses into this possibility with each power blackout. In the early 2000s the entire eastern U.S. seaboard went dark for hours and in some places even as long as days. Without electricity, we are without light, communication (computers, telephones, radio, television, cable, satellite, etc.), and transportation (cars, trains, motorcycles, trucks, airplanes, shipping, traffic signals, etc.), the foodchain is broken (refrigeration is gone, meat and produce cannot get from the farm to the stores, etc.), modern medicine, is disrupted, and the list goes on and on. All of these things are fundamentally dependent, or have become dependent on electricity. We humans have lived only since the late 1890s dependent on electricity, so in its absence we would continue on; however, we would be thrown into a minidark ages as most of the knowledge of simply how to survive without electricity has been lost to the masses.

Defense spending has significantly increased since September 11, 2001 and will as a result produce technology that would simply not be available because of market economics. Society as a whole plays a pivotal role in the evolution of technology. In addition to social forces that drive innovation, government involvement, particularly in research, contributes significantly to the evolution of technology. In the twentieth century, society focused and continues to focus on communications, particularly through the rapid adoption of pervasive interconnected devices. The U.S. government has focused significantly on research to enhance the nation's infrastructure and to more tightly integrate existing institutions. The resulting impact has led to an increased pace of innovation in the academic community. It has also led to an increased reliance on communications as a basic element of the social fabric.

Technological evolution has itself played a role in driving convergence and benefiting from it. Most notably, developers have taken advantage of the predictive power posited by Moore's, Metcalfe's, and Amdahl's laws. This has enabled the construction of more complex applications as well as the interconnections and communication paradigms between them. Communications has not experienced major fundamental shifts since the 1960s. All current systems have simply evolved as strata above the IP substrate. This stratum (middleware) has become heavily bloated with a variety of products, approaches, and directions without any clear winners. This in turn has led to an exponential increase in information, middleware bloat, and overprovisioning. Some of the main technological driving forces are described below.

Network Computing Power Explosion. Technical communications have been gated by the ability of the switch or router to process and route packets. Since the mid-1990s these routers and switches have become increasingly complex and have embedded more features. This network computing power explosion is enabling a higher degree of intelligence in the network layer. This will lead to a more complex routing/networking model, particularly as the previous models, which were built to take advantage of a significantly less powerful network switch fabric, reach the end of life.

Consequences of Moore's and Metcalfe's Laws. *Moore's law* states that the number of transistors on a chip (which implying an increase in compute power) doubles every 18 months and has brought us to the point of compute cycles since the most precious resource of the computer is now more of a commodity. *Metcalfe's law* observes an exponential increase in value of the network as the number of users on the network increase (specifically, the usefulness is the square of the number of users). This has led to the current computing resources that have been producing information above and beyond the ability of humans to process it, a trend that is not about to change. Thus more system power needs to be in place to address the overflow of this information.

Isomorphism to Evolution of Previous Systems. The isomorphism evolution is similar to the evolution of the telephone switch, as traced from Shannon, Weiner, and Dijkstra.

Grid and Web Services as Manifestation of State Transition. The grid addresses some of the key points dealing with this manifestation. Through the network power explosion that enables more intelligent resource management at the switch level, including not only network but other resources as well, the grid is the next solution. Combined with Moore and Metcalfe's laws, this will continue for some time; thus grid capabilities and network computing capabilities will be closely correlated. Web Services provides the ideal organizational mechanism for network services such as discovery, connection, and management.

Conclusion. The convergence of macroeconomic, Sociogovernmental, and technological forces is the fundamental driver behind the paradigm shift in communications. Macroeconomic forces are driving a demand for elasticity, fungibility, and granularity of service levels, resources, and IT planning. Societal forces are driving the emergence of novel communication mechanisms as well as the rapid adoption of such mechanisms within the basic social fabric. Coupled with the steady increase in CPU clock rates, these forces together have pushed modern communication to a breaking point, leading to a proliferation of point solutions, custom middleware, and overprovisioning.

Communication is a fundamental element of our social fabric. Technical communications is slowly beginning to attain the same status and as such is evolving through a set of paradigms. These paradigms are driven by the needs of society as well as the capabilities of technology and will eventually settle on a set of mechanisms that will become the lingua franca of our time. SONA and Web Services have the potential to become this lingua franca.