CHAPTER 1

AUTONOMIC ATTRIBUTES AND THE GRAND CHALLENGE

INTRODUCTION

In March 2001, IBM Senior Vice President and Director of Research Dr. Paul Horn spoke about the importance and direction of autonomic computing before the National Academy of Engineering conference at Harvard University. He had a very direct message:

> "The information technology industry loves to prove the impossible possible. We obliterate barriers and set records with astonishing regularity. But now we face a problem springing from the very core of our success—and too few of us are focused on solving it. More that any other IT problem, this one—if it remains unsolved—will actually prevent us from moving to the next era of computing. The obstacle is complexity . . . . Dealing with the single most important challenge facing the IT industry."[1]

This was the first time the world was told of IBM's autonomic computing program. Shortly after, Mr. Irving Wladawsky Berger, the IBM Vice President of Strategy and Technology for the IBM Server Group, introduced the Server's Group project (known then by the internal IBM codename eLiza). He stated goal was to provide "self-managing systems." This was expanded to many other divisions and business units within IBM. It was and remains a company-wide project. Project eLiza would eventually become known as the autonomic computing project. Thus began the autonomic computing journey within IBM. Dr. Paul Horn's presentation was released as a manifesto, and as many as 75,000 copies were reportedly distributed to customers, press, media, and researchers worldwide. In the manifesto, Paul Horn invites customers, competitors, and colleagues alike to accept the "Grand Challenge of building computing systems that regulate themselves."

The term Autonomic Computing derives from the human autonomic nervous system (ANS). The same way we take for granted the human body's management of breathing, digestion, and fending off germs and viruses, shown in Figure 1.1, we will take for granted the computer's ability to manage, repair, and protect itself. That process has begun with autonomic computing.

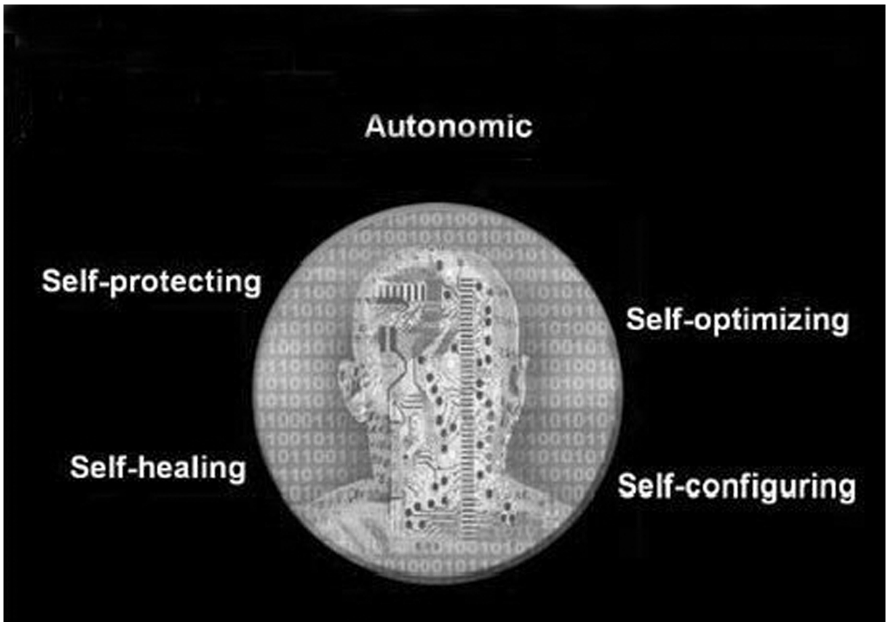

Figure 1.1 Defining autonomic—self-configuring, self-optimizing, self-healing, self-protecting.

We can learn much from how the human body manages itself and apply those same techniques to software to create system management functions for commercial corporations. This is the fundamental purpose of autonomic computing.

The grand challenge of autonomic computing is not just about one company. It is bigger than IBM. To be successful, it must be a joint effort of the entire technology industry, involving software and hardware vendors of all sizes. In addition, it will require the involvement of the academic and university community, as well as customers, who will be the ultimate users of this technology.

The Information Technology (IT) industry today faces the biggest threat to its continuing success. That threat is *complexity.* The systems that we have developed and installed are becoming so complex that they verge on becoming unmanageable. Complexity is everywhere—in architectures, networks, programming languages, and applications and software packages. The IT industry is a victim of its own success. The complexity now facing most CIOs is a direct result of internal and external pressures—management, economic, market—to make everything *cheaper, faster, and smaller.* Now IT staff are succumbing to the effects of this complexity. Data center staff, network administrators, and IT support staff are wrestling daily with major problems of incompatibility and software failures, and they must deal with these problems manually. The situation is further convoluted by computers from multiple vendors—each with their own proprietary software, data protocols, transmission standards, and so on. This presents challenges and makes it difficult for IT personnel to manage an environment containing diverse and heterogeneous infrastructures. Often, IT personnel can't integrate different systems. Even when they can, IT personnel still face difficulties adding new systems to existing environments and then configuring and managing them. Thus, most organizations now spend about three-fourths of their application deployment time and costs on the integration of different systems. They must also deliver services across geographical and business boundaries. This means organizations, in addition to managing heterogeneous vendor and technical environments, also have to put extensive efforts into customizing technologies to meet the requirements of different IT policies, while delivering unique services to customers. This complexity keeps the cost of managing IT infrastructure high. At times, the complexity causes overruns in IT cost and delays in implementation. This, in turn, translates into losses in productivity and missed business opportunities.

The second complexity factor is the increased size of IT infrastructures. The accelerating pace by which myriad devices are being added to almost all IT networks (especially the Internet) further complicates the already sophisticated technological environment. Rapid advances in technology have led to significant improvements in price/performance ratios, thus making technology accessible to many. Today, corporations are no longer dealing with one person accessing one application on a local PC or on a network server. Instead, organizations are seeing thousands, and eventually even millions, of users accessing the same service hosted on one or more servers, potentially at the same time. Relying on human intervention to manage this complexity bears a steadily increasing risk of as the scale and level of complexity extend beyond the comprehension of even highly skilled IT personnel.

The third complexity factor is the escalating costs of systems. The increasing complexity of integrated systems makes the job of maintaining and fixing systems more challenging than ever. In today's competitive world, where customers expect uninterrupted services, even a short breakdown can cost organizations millions of dollars in lost business. In fact, it has been reported that one-third to one-half of typical IT budgets are spent on preventing or recovering from crashes.

The fourth complexity factor is the shortage of skilled labor. Workers who have the knowledge to manage complex IT infrastructures are expensive and remain in short supply, even in today's depressed economy. According to a study by researchers at the University of Berkeley, depending on the type of system, labor costs could surpass infrastructure costs by a factor of 3 to 18. Therefore, the strategy of relying on human intervention to manage IT infrastructure might not be a favorable strategy in the long run, as there might come a point where existing skilled labor and manpower will not be enough to supply e-business on demand. Complexity is discussed in more detail in Chapter 2.

The challenge is to simplify IT. The sheer size and complexity of current computing environments has hindered efforts to integrate systems, databases, applications, and business processes, and has substantially decreased management and operational efficiency. Making IT infrastructures flexible enough to respond quickly and effectively to dynamic customer requirements, marketplace shifts, and competitive demands remains a challenge. Autonomic computing can meet that challenge.

DEFINITIONS

The strict and formal definitions of autonomic and autonomous are as follows:[2]

AUTONOMIC
SYLLABICATION: au·to·nom·ic ADJECTIVE: 1. Physiology a. Of, relating to, or controlled by the autonomic nervous system. b. Occurring involuntarily; automatic: an autonomic reflex. 2. Resulting from internal stimuli; spontaneous. OTHER FORMS: auto·nomi·cal·ly —ADVERB

ATONOMOUS
SYLLABICATION: au·ton·o·mous ADJECTIVE: 1. Not controlled by others or by outside forces; independent: an autonomous judiciary; an autonomous division of a corporate conglomerate. 2. Independent in mind or judgment; self-directed. 3a. Independent of the laws of another state or government; self-governing. b. Of or relating to a self-governing entity: an autonomous legislature. c. Self-governing with respect to local or internal affairs: an autonomous region of a country. 4. Autonomic. ETYMOLOGY: From Greek autonomos: auto-, auto- + nomos, law; OTHER FORMS: au·tono·mous·ly —ADVERB

A QUICK GUIDE TO THE HUMAN AUTONOMIC NERVOUS SYSTEM

IBM chose well when naming their new initiative of self-healing, self-configuring, and self-protecting systems. We would not be able to live healthy normal lives if it were not for our internal autonomic nervous system (ANS). This silent guardian is working constantly to ensure that the body is stable and performing at optimum levels. The human nervous system is divided into the voluntary and involuntary systems. You control the voluntary system. For example, when you feel something uncomfortable, such as pain, heat or bright light, you can choose to move away from it or react to it in some fashion, normally defensive. The involuntary system—which is the autonomic nervous system—handles actions over which you have no control, such as heartbeat, digestion, circulation, and glandular function.

Although it is located in your physical body, the ANS is affected by everyday emotions. For example, when you feel fear, that emotion will be translated into a physical response—the release of the hormone adrenaline—that will increase your heart rate, blood pressure, and digestive processes. This process is managed and controlled by your ANS.

The ANS is very complex. In keeping with the goals of this book, however, we are going to present it as simply as possible. The ANS runs throughout the entire body. It originates from the spinal column and is connected to every gland and organ. See Figure 1.2.

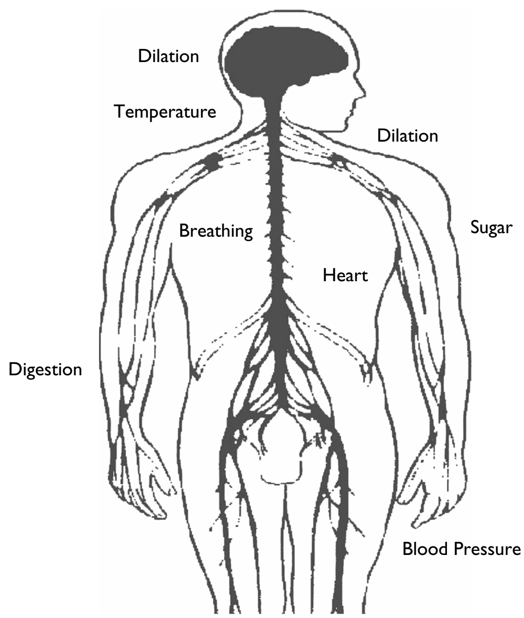

Figure 1.2 An illustration of the human ANS and some of the features automatically regulated by this system. Note that everything originates at the spine.

The ANS is divided into further subsystems: the sympathetic nervous system (SNS) and the parasympathetic nervous system (PNS). A third division, the metasympathetic system (MNS) is related to the central nervous system in the brain. Put very simply, the SNS tends to speed up responses in our muscles and organs to help us adapt to stress, while the PNS slows down those responses. When these systems are in balance, after a high stress response, the PNS will help to calm you down. It is known in modern terms as the “flight or fight” response.

Under conditions of balance, the SNS turns on organ responses to high levels of environmental stress. When the stressful conditions are removed, the PNS turns on to restore balance within your organ systems. Under conditions of imbalance, the SNS may be turned on for long periods of time, and the PNS may be turned on as well. This is like having your foot on the accelerator pedal and the brake at the same time.

The ANS system is a marvel of bodily management. If it is running well, you will feel well. The autonomic nervous system is always working, never resting or going offline. Even when you are sleeping, it continues to manage bodily functions. It is NOT only active during “fight or flight” or “rest and digest” situations. Rather, the ANS acts to maintain normal internal functions and works with the rest of the nervous system. Most important of all, the ANS does its work without any conscious recognition. This allows you to think and act the way you want. It simply says:

> “Don’t worry about it. I’ve got it covered.”

This is precisely the approach we need to build future business computing systems.

E-BUSINESS ON DEMAND

IBM has announced their vision for the next major strategy of business adoption. They call it *e-business on demand*™. It is a statement of IBM’s belief of how businesses will need to transform themselves to be successful. Businesses will have to adapt to cope with ever-increasing pressures from competition and other factors associated with the global economy. The modern business will require full integration across people, processes, and information, including suppliers, distributors, customers, and employees.

Autonomic computing is a fundamental aspect of IBM’s e-business on demand strategy. E-business on demand is designed to address the growing dynamic between IT solutions and business strategies. Corporations and their management regard technology as a means to power business evolution to the next competitive level. To compete and win in today’s markets, companies must be agile and responsive to customer demands, marketplace shifts, and competitive pressure.

The ability of e-business on demand solutions to leverage existing IT infrastructures with additional integrated services and capacities has implications for corporations of every size. Small- to medium-sized businesses can use e-business on demand solutions to gain the economic scale of large companies. Large corporations can use e-business on demand to become as nimble as small businesses. At its heart, IBM’s e-business on demand offers corporate IT environments where every single employee, customer, partner, application, and process has seamless access to any and all necessary business assets, information, and resources.

IBM defines an e-business on demand business as a corporation whose business processes, integrated end-to-end across the company and with key partners, suppliers, and customers, can respond with speed to any customer demand, market opportunity, or external threat.

There are four key attributes of an e-business on demand business:

- **Response—**The ability to sense and respond to dynamic, unpredictable changes in demand, supply, pricing, labor, competition, capital markets, and the needs of its customers, partners suppliers, and employees.
- **Variability—**The ability to adapt processes and cost structures to reduce risk while maintaining high productivity and financial predictability.
- **Focus—**The ability to concentrate on its core competencies and differentiating capabilities.
- **Resiliency—**The ability to manage changes and external threats while consistently meeting the needs of all of its constituents.

These attributes define the business itself. For a business to successfully attain and maintain these attributes, it must build an IT infrastructure that is designed to specifically support the goals of the business. An entirely new kind of IT infrastructure must be put in place to support an e-business on demand business. IBM calls this infrastructure the e-business on demand operating environment.

To achieve this vision, IBM believes that information resources must be:

- **Virtualized—**allowing business assets, including data, applications and resources, to flow freely wherever and whenever they are needed;
- **Integrated—**enabling the efficient movement of information between systems, applications, databases, and business processes;
- **Open—**assuring the free integration and flow of business assets in an IT world that is both practically and philosophically heterogeneous; and
- **Autonomic—**empowering the stability and reliability of solutions in endemically complex corporate IT environments.

There are many technologies, both new and evolving, that can make this environment a reality. An e-business on demand operating environment is about a broad set of standards working together to provide a consistent and comprehensive set of services and deliverables, as illustrated by Figure 1.3.

An e-business on demand business is responsive, variable, focused, and resilient. To attain these business attributes, an information technology infrastructure must be put in place that supports the business and provides true business value. An e-business on demand operating environment supports the e-business on demand business by providing the capability to integrate, virtualize, and automate systems and processes.

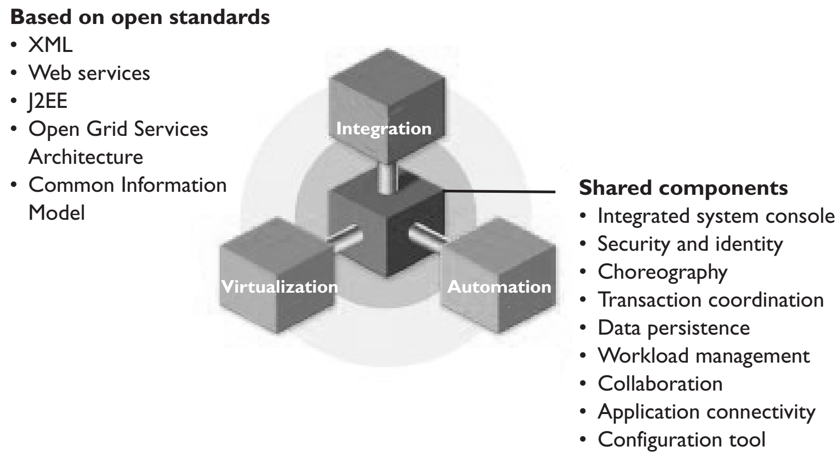

Figure 1.3 The e-business on demand environment based on open standards and shared components.

AUTONOMIC COMPUTING ELEMENTS

To be truly autonomic, a computing system needs to "know and understand itself"—and comprise components that also possess a system identity. Since a "system" can exist at many levels, an autonomic system will need detailed knowledge of its components, current status, operating environment, and ultimate capacity, and of all connections with other systems. It will need to know the extent of its "owned" resources, those it can borrow, buy, or lend, and which ones can be shared or should be isolated. As shown in Figure 1.4, autonomic computing has four basic elements: self-configuring, self-healing, self-optimizing, and self-protecting.

SELF-CONFIGURING

An autonomous computing system must be able to install and set up software automatically. To do so, it will utilize dynamic software configuration techniques, which means applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configurable item. Also to control changes to those characteristics, to record and report change processing and implementation status, and to verify compliance with specified service levels.

Also, downloading new versions of software and installing regular service packs are required. When working with other autonomous components, an autonomous system will update new signatures for virus protection and security levels. Self-configuration will use adaptive algorithms to determine the optimum configurations.

Figure 1.4 The four aspects of autonomic computing: self-configuring, self-healing, self-optimizing, and self-protecting.

Examples:

1. Updating Web pages dynamically with software changes, testing those changes, analyzing the results, releasing the system back into production, and reporting back to self-management whether the procedure was successful.
2. Installation, testing, and release of regular vendor service packs.
3. Installation of vendor patches, corrections, and modifications together with the necessary testing and release.
4. Installation of new software releases—automatically and seamlessly.

SELF-OPTIMIZING

An autonomous system will never settle for the status quo. It will be constantly monitoring predefined system goals or performance levels to ensure that all systems are running at optimum levels. With the business constantly changing and demands from customers and suppliers changing equally fast, self-adapting requirements will be needed.

Self-optimization will be the key to allocating e-utility-type resources—detemining when an increase in processing cycles is needed, how much in needed, where they are needed, and for how long. To be effective, autonomous self-optimization will need advanced data and feedback. The metrics need to be in a form where rapid analysis can take place. Many new and innovative techniques are needed for optimization to be successful—for example, control theory is needed in new autonomous infrastructures. New algorithms to process control decisions will be needed.

Examples:

1. Calling for additional processing power from the e-utility when needed. Releasing those additional cycles when peaks are over.
2. Working with outside vendor software.
3. Interfacing with other autonomic modules to exchange data and files.
4. Optimum sub-second response times for all types of access devices, such as personal computers, handheld devices, and media phones.

SELF-HEALING

Present computer systems are very brittle. They fail at the slightest problem. If a period, a comma, or a bracket is not correct, the software will fail. We still have much to do in designing forgiving systems. Autonomous computing systems will have the ability to discover and repair potential problems to ensure that the systems run smoothly.

With today's complex IT architectures, it can be hours before a problem is identified at the root cause level. System staff members need to pore over listings of error logs and memory dumps, tracing step-by-step back to the point of failure. The cost of downtime to the business is prohibitive. For example, in large-scale banking networks, the cost can be as much as $2,600,000 per hour. Self-healing systems will be able to take immediate action to resolve the issue, even if further analysis is required. Rules for self-healing will need to be defined and applied. As autonomous systems become more sophisticated, embedded intelligence will be applied to discover new rules and objectives. For example, IBM will be building SMART (Self-Managing and Resource Tuning) databases into upcoming versions of their DB2 database product. This database is designed to run with less need for human intervention. For example, the user can opt not to be involved, and the database will automatically detect failures when they occur (and correct them) and configure itself by installing operating systems and data automatically to cope with the changing demands of e-business and the Internet.

Examples:

1. Self-correcting Job Control Language (JCL): when a job fails, the errors or problems are identified and jobs rerun without human intervention.
2. An application error forces the entire system to halt. After root cause analysis, the error is corrected, recompiled, tested, and moved back into production.
3. A database index fails. The files are automatically re-indexed, tested, and loaded back into production.
4. Automatically extend file space and database storage, according to previous data on growth and expansion.

SELF-PROTECTING

In an increasingly hostile corporate world, autonomous systems must identify, detect, and protect valuable corporate assets from numerous threats. They must maintain integrity and accuracy and be responsible for overall system security. For years before the Internet, each corporation was an isolated island where threats usually came from within.

Now, outside threats come daily, and security and protection are paramount. Threats must be identified quickly and protective action taken.

Autonomic system solutions must address all aspects of system security at the platform, operating system, network, application, Internet, and infrastructure levels. This involves developing new cryptographic techniques and algorithms, their secure implementation, and designing secure networking protocols, operating environments, and mechanisms to monitor and maintain overall system integrity. Such security solutions need to be standardized to provide/preserve interoperability and to ensure that these techniques are used in a correct way.

To achieve this will require continuous sensors feeding data to a protection center. A log of events will be written and accessed when appropriate for audit purposes. To manage the threat levels, we might expect a tiered level. Threats can be escalated through the tiers for increasing action and priority.

Examples:

1. Confirm the ability of backup and recovery resources that may be needed.
2. Implement tiered security levels.
3. Focus resources on network monitoring and immediately disconnect computer systems with suspicious network traffic.
4. Verify that network configurations inventories are correct and, if not, take action.
5. Contact system administrators outside of autonomous system and other offices that may be affected by the increasing threat levels.
6. Have the system verify that all computer systems are at the appropriate version levels, including “patches.” Update automatically as needed.
7. Resolve any open security concerns.
8. Implement any special software for additional security protection according to the threat level.
9. Contact offsite vendors to determine if any preventive measures (patches, etc.) to be applied to both hardware and software.

OPEN STANDARDS

We have already identified that the IT industry is going through major changes. New technology—such as autonomic computing, Web services, and grid computing—is creating tremendous opportunities to massively increase business profitability. The potential of these technologies to transform business is amazing, and open standards will play a critical role in this new e-business on demand world. Just as open standards were critical to the emergence of the Internet and the first generation of e-business, they will play a critical role in the e-business on demand generation.

We can define open standards as interfaces or formats that are openly documented and have been accepted in the industry through either formal or *de facto* processes, and are freely available for adoption by the industry. Examples include HTTP, HTML, WAP, TCP/IP, VoiceXML, XML, and SQL. They are typically built by software engineers and programmers working for software companies that collaborate under known industry standards-based organizations such as the W3C, OASIS, OMA, and IETF.

Figure 1.5 illustrates the variety of different standards organizations.

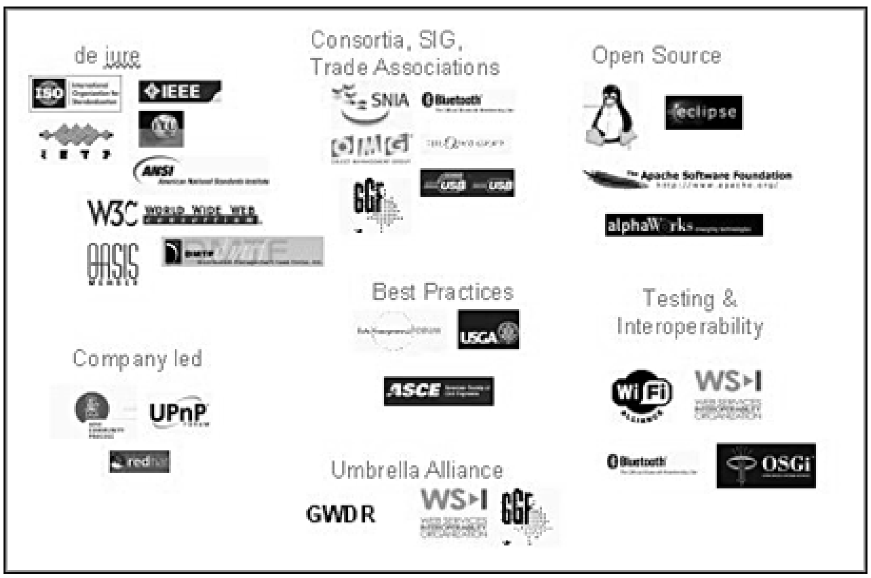

Figure 1.5 The landscape of standards organizations.

The open standards community counts major vendors such as IBM, HP, Sun, Microsoft, Cisco, and Oracle among its active contributors. Other members include Red Hat, Apple, and Intel. These examples are, of course, only a drop among the thousands of other companies involved in these initiatives. Their members also include contributors from the academic and university world, such as Stanford, Berkeley, and MIT.

Why are major IT vendors and software companies increasingly interested in open standards? Primarily because investing in these communities makes good business sense. The global computer industry must cooperate in developing the necessary open standards and interfaces to make future technology work and to establish standards that will support an autonomic computing environment.

Therefore autonomic computing needs open standards to be successful. IBM strongly agrees that open standards are *imperative* for the whole e-business on demand strategy. We define open standards as:

> The specifications that enable the construction of the universal building blocks for the development of software and solutions.

The adoption of open standards for autonomic computing may appear a daunting task, as each component of autonomic computing will need to "describe" itself to other software, their resources, and most importantly, their requirements. For example, the self-protection component will need to contact a virus signature vendor and state, "Send me the latest signature file to interface with adaptive algorithms."

A final comment on open standards:

> The days of developing and selling proprietary systems are over and we must go forward together with strategies that will benefit our future industry. Our customers deserve better.

AUTONOMIC COMPUTING—WHY NOW?

There are four answers to the question of how we know the time of autonomic computing has come:

1. Computing infrastructure and system complexity demands it.
2. The current business climate demands it.
3. The technology has evolved enough to deliver it.
4. We must act to prevent the situation becoming any worse.

First, computing systems continue to evolve to meet changing business needs. But process-based computing systems are more complex than any of the preceding paradigms. Dynamic process-based computing systems give corporations the ability to automate business processes and the flexibility to optimize and adapt processes according to their business needs. These systems interconnect all aspects of doing business and integrate multiple business applications. Integration results in computing infrastructures that are both interconnected, because multiple technologies are involved in delivering business services and processes, and dynamic, because new application modules and functions can be added at a faster rate. Technology interconnection and dynamism breed maintenance difficulties for three reasons:

1. It becomes impossible for manual-user administration, technology maintenance, and management processes to keep pace with the rate of infrastructure change.
2. Management of individual technologies and software modules does not guarantee the availability and performance of the end-to-end service or process.
3. Inter-relationships between individual technologies and software modules increase the complexity of configuration, performance, and security problems that need to be resolved.

Second, almost every Total Cost of Ownership (TCO) study shows that the more moving parts a system has, the more expensive it is to maintain in good working order. Indeed, the current rule of thumb is that for every dollar spent on computing infrastructure, another ten are spent for ongoing management. This ratio increases as system complexity grows. Eventually the growing cost structure becomes too much for the corporation to bear, even when the economy as a whole is performing well. The current weak spending environment places enormous pressure on IT departments to lower costs and increase the returns delivered by existing infrastructure. There is only one way to do that in an IT operations center. That is to automate tasks that are normally handled by people. The autonomic computing technology provides executives a blueprint for increasing the cost-effectiveness of their IT departments.

Finally, IT management technology has been evolving to deliver true “lights out operations,” a concept for computing infrastructures having distributed processing power, modular software architectures, accessibility over public networks, and frequent configuration changes. Until now, these trends have been developing separately, related only by the basic concept of increasing the intelligence of management tools. The time to act is NOW.

IS AUTONOMIC COMPUTING NEW?

The phrase autonomic computing is new to many IT staff and is used to define the new technology of self-managing systems. However, the practice of designing and implementing self-managing systems has long been a goal and objective of IBM. Numerous examples exist within the IBM portfolio of products in hardware and software, some of which go back over twenty years or more.

A few examples:

1. The IBM 4300 series of mainframe computers were introduced in 1982. This line was a forerunner of the newer range of mainframes in use today, such as the OS/390 series. A feature was introduced with the 4300 that included internal self-diagnostics. Moreover, it contained a feature to transmit findings, logs, memory dumps, and return codes to remote IBM system engineers, who could then provide customers with quicker solutions.

2. More recently, personal computers now have some self-configuration abilities and can receive updates from the Internet, such as critical updates to operating systems. They also have the power to allocate memory.
3. The IBM Tivoli suite of software products has many autonomic capabilities. The IBM Tivoli Storage Manager has self-configuration features, such as automatic domain configuration and file identification; self-correction features, such as storage pool, restoration, and correction; self-protection features, such as automated backup to protect data and files when needed; and policy-based systems that optimize disk and tape storage systems.
4. RAID (Redundant Array of Independent Disks) systems are another example. These systems connect multiple disk arrays into one logical unit. If errors or problems arise in read/write functions, data is switched automatically to another part of the array.

So, the processes of self-managing systems have existed for some time. While these technologies move in the right direction, they are a precursor to fully autonomic computing. Autonomic computing enhances and builds on the base of existing IBM portfolio products and brings better levels of automation through the five level processes.

WHAT HAPPENS IF IT DOES NOT CHANGE?

What if the unthinkable happens, and we do not adopt autonomic computing or similar technology? This nightmarish scenario hardly bears consideration.

Mixtures of the following events will happen with different severity.

1. Complexity will continue to increase, reaching unmanageable proportions.
2. Further pressures will increase on IT staff to fix unfixable problems.
3. Reliability of systems and performance will deteriorate. Businesses will suffer.
4. Corporations will lose their competitive edge and lose substantial markets and profits.
5. Corporations will be required to increase their IT budgets to astronomical proportions.
6. Senior management will reject these astronomical budget requests—the cycle of problems will continue.
7. More and more skilled IT staff will be needed at substantial costs.
8. The health of many IT staff will suffer.
9. Chaos.

The IT industry has lingered too long in the religion of overspecialization, in which integration was just another specialty. The IT industry has made spectacular progress in almost every aspect of computing. But it has not made enough in the one area that now counts most: dealing with the complexity generated by all the systems we have installed so far. In this heady rush, there is a danger of losing sight of the very people who use IT and who have come to depend on us for increased productivity and improvement in many aspects of their daily lives. We've made it unnecessarily difficult for them to tap the potential we've promised them. *It's time for this to change. It must change.* Chaos is not an option.

This next era of computing will enable progress and abilities we can barely envision today. But the best measure of our success will be when our customers do not think about the functioning of computing systems.

CREATING THE AUTONOMIC CULTURE

Managing an increasingly diverse IT population requires cross-cultural competencies. A global, agile, virtual, matrixed, e-business on demand organization should be able to build effectiveness across national, organizational, team, and interpersonal barriers. Successful cross-cultural IT managers typically are flexible and possess a broad behavioral repertoire.

Culture is the complex pattern of ideas, emotions and observable behaviors that tend to be expected, reinforced, and rewarded by and within a particular group. You can most easily understand and observe culture at four levels:

- National
- Individual
- Team
- Organizational

Culture is not an inherited characteristic. It is shaped by what we learn in the context of our social group or category. Culture goes beyond narrow nationalistic definitions.

WHY IS A CULTURE IMPORTANT?

As an IT person, culture can be important to you. There are thousands of examples where cultural misunderstandings have killed deals, harmed working relationships, inhibited sales, or increased costs. You must be able to succeed everywhere. Understanding culture is part of success. When you become aware of potential cultural differences and realize their implications, you dramatically increase your ability to work with people across the globe. You will also enhance your competitive advantage by building stronger, more sustainable relationships. If you remain unaware of cultural differences, you can jeopardize business relationships and entire projects. Regrettably, IT culture is a subject that has been avoided.

IBM uses the COM (Cultural Orientation Model) developed by TMC of Princeton, NJ. This model is based on common tendencies of people on a number of dimensions, or values, and has identified the dimensions as:

1. **Environment:** How individuals view and relate to the people, objects, and issues in their sphere of influence.
2. **Time:** How individuals perceive the nature of time and its use.

3. **Action:** How individuals conceptualize actions and interactions.
4. **Communication:** How individuals express themselves.
5. **Space:** How individuals demarcate their physical and psychological space.
6. **Power**: How individuals view differential power relationships.
7. **Individualism:** How individuals define their identity.
8. **Competitiveness:** How individuals are motivated.
9. **Structure:** How individuals approach change, risk, ambiguity, and uncertainty.
10. **Thinking:** How individuals conceptualize.

Figure 1.6 illustrates this model.

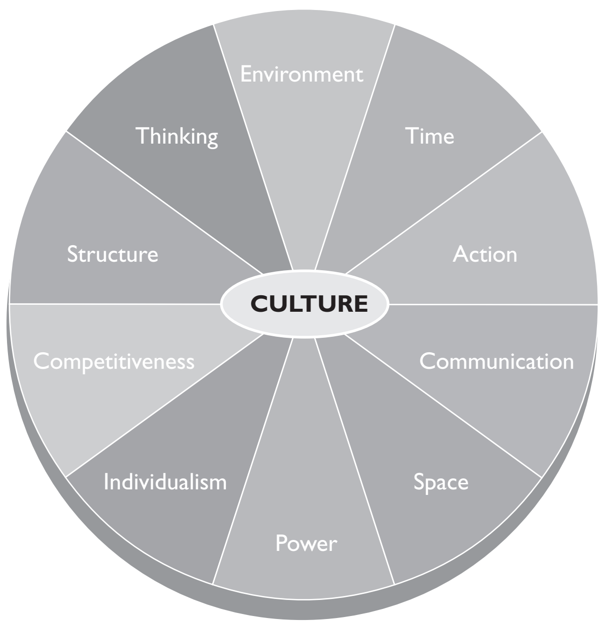

Figure 1.6 The Cultural Orientation Model.

This approach will be valuable for corporations and IT departments as one of the first steps towards autonomic computing transition. What is needed is an *autonomic cultural transition plan* based on the elements described above.

IS AUTONOMIC COMPUTING WORKING TODAY?

Autonomic computing is real. IBM customers are using it TODAY, and they are beginning to reap the rewards and benefits of this technology.

A few brief examples follow.

Efes Pilsen

This European brewer Efes Pilsen taps into increased productivity and sales with a high-availability data warehouse solution offering self-managing autonomic functions.

Santix

This systems integrator reduces the workload on administrators with self-optimizing, predictive software using IBM's Tivoli management suite to help prevent breaches in its service level agreements with its clients.

Bankdata

This bank uses autonomic hardware and software to help transform numbers into meaningful knowledge, allowing its management to understand customer habits, preferences, and profitability potential.

Below are two quotes and observations from IBM customers of autonomic technology that sum up the feeling of success that has been achieved.

> "People make mistakes when they are under pressure. Autonomic computing holds the promise in helping administrators reduce human error and improve data integrity. Our users can take advantage of the autonomic, self-managing features that IBM delivers."
>
> —Andrew Hall, President, ETI-NET[3]

> "We have derived great benefit from IBM's autonomic features—including self-configuration features, such as automatic hardware detection, as well as self-healing 'call home' functionality that alerts support staff in the event of a failure."
>
> —Frank Buthler, System Programming Manager, Bankdata[4]

SAME SOUP—DIFFERENT FLAVOR

True to its technological roots, the new self-managing industry, press and vendors have come up with a plethora of different names for autonomic computing and related technologies. This is likely to cause confusion to management and end users alike. There is a case for standardization here before the industry becomes mature. Below are a few examples.

- **Adaptive Enterprise**—Cap Gemini Ernst & Young
- **Dynamic Systems Initiative**—Microsoft

- **e-Business on Demand**—IBM
- **Autonomic Computing**—IBM
- **N1**—Sun Microsystems
- **On-Demand Computing**—EDS
- **Organic IT**—Forrester Research
- **Policy-Based Computing**—Gartner
- **Real-Time Enterprise**—Gartner
- **Utility Data Center (UDC)**—Hewlett-Packard

SUMMARY AND CONCLUSIONS

Today's competitive business landscape is bristling with innovation and change. Tomorrow will be different—it will be heightened and even more forceful. Therefore only the most durable and robust business models will survive. In this brutal environment, the IT strategy must be vigorous enough to stand out, yet flexible enough to evolve in an e-business on demand world. CIOs and IT management must be attuned to this shifting reality and aim to implement the technologies that directly address this actuality.

New technologies are rapidly emerging, and new versions of existing technologies will continue to be released. To keep pace, businesses need to quickly adapt their existing applications to new technologies and business requirements without losing their investments in current systems.

The ultimate scope of autonomic computing will go beyond just the corporate world. The technology of autonomic computing is based on sensors and effectors. (See Chapter 8 for more explanation.) When sensors are embedded in other devices, exciting things can happen. For example, Chrysler has long been perfecting Antilock Braking Systems (ABS) to ensure driver and passenger safety. The ABS is comprised of electronic sensors and solenoid valves in the wheel hubs. These sensors and valves use the concept of autonomic systems to prevent the wheels from locking when cars go into a skid. Goodyear and Michelin have created "Run-Flat" tires that allow drivers to drive safely for a few more miles after a tire puncture. Besides having a reinforced sidewall in the tire that acts to maintain the chassis level when a tire is deflated, Run-Flat tires also have sensors that relay information about the air pressure to the dashboard of the car so that drivers can monitor the pressure levels and act accordingly.

Sensors and embedded computers already exist in most industrial environments, but they are typically reserved for use on small numbers of expensive pieces of equipment. What happens when these devices make their way from the few mission-critical components to the multitude of equipment all around you? Just one benefit alone, such as real-time awareness of potential component failure, can save a utility from costly downtime, which in some cases can cost up to $1 million each hour.

Prototypes of smart homes are being built and tested. Smart homes are equipped with a network of sensors that track and monitor everything from cooking habits to purchasing activities to level and quality of physical movement. One potential outcome of this research is the early detection of the onset of dementia by monitoring changes in activity patterns and levels. A broader goal is to use the sensor data to build technologies that support and enhance people's ability to conduct normal daily activities, by providing an appropriate level of support without making people prematurely dependent on technology-based assistance.

Once sensors are placed in everything, computing will be truly pervasive. Already we have sensors and chips in cell phones to refrigerators and toasters. In the future, sensors in shopping malls will pick up your preferences as you enter the mall. This trend will run in parallel with the adoption of autonomic computing in the corporate world.

Autonomic computing in the corporate world is needed. There is not much of an alternative.

NOTES

1. Dr. Paul Horn, *Autonomic Computing, IBM's Perspective on the State of Information Technology, IBM Corporation*. October 15, 2001.
2. *The American Heritage® Dictionary of the English Language*: Fourth Edition. 2000.
3. See *www.306.ibm.com/autonomic/success_stories.shtml.*
4. See *www.ibm.com/software/data/solutions/pdf2/bankdata.pdf*\.