THE PRIME THEOREM

These stories are distributed objects on a local network, the perimeter of which is bound by the covers of this book.

Each story/object represents a fundamental concept underlying Grid Management Theory, formulated over two decades of my profession in information technology (IT) and based on a core set of central principles. Rather than presenting each concept in the traditional mode of other management books, or in the accepted academic format of research papers with statistics and an annotated bibliography, I have chosen a narrative format for two important reasons:

1. The central premise of Grid Management Theory is that the people who design, build, and manage our technology ecosystems are an essential component of these systems. Subsequently, concentration on their life stories can provide a more inclusive portrayal of these central principles.

   In this case, fiction offers a more honest picture.

2. The greatest handicap observed in any technical organization, large or small, corporate or private, is the nearly universal inability of technologists to explain themselves adequately—to their executives, their customers, and their spouses.

   These stories provide a bridge.

Each concept is embedded in a story with believable characters struggling with real IT issues, an accessible format that will hopefully engender more fruitful discussions within and beyond our organizations than those normally provoked by academic treatises or business guru-speak. One

thoughtful extension of this strategy might be an IT Director sharing a copy of Chapter 3 on Virtualization with her sales/marketing counterparts, to help them understand the impact of this issue on the Director's organization, their morale, and their place in history.

The Internet, viewed from this perspective, is a latticed system of relationships—among libraries, information objects, servers, and, as you will see in the course of these stories, the users of this system. The best example is the lesson I learned more than a decade ago: hypertext is nonhierarchical, and therefore, to be successfully implemented, it requires nonhierarchical (matrix-managed) teams. To manage Internet-based projects properly (software development, eCommerce, publication), one must manage the series of relationships among the people who build and support those projects.

The central proposition of this book is a theme I've observed at every level of the corporation and at every level of our IT architecture: *information systems mirror the people who build them*. Each story in this collection is based on this central theorem and a set of corollaries, derived from the broader discipline of systems theory as it applies to information systems.

The Prime Theorem is this:

We mirror ourselves in the systems that we build. Therefore:

Corollary 1. The systems will not "talk to each other" if the people are not "talking to each other."

Corollary 2. The relationships between systems reflect the relationships between the people who build and support them.

Corollary 3. To correct problems in our information systems, we must first address the problems among the people that build them.

Corollary 4. We must transform ourselves to the same degree that we want to transform our systems.

To introduce this collection of short stories,[1] it is important to emphasize that I have observed examples of this theorem *everywhere*. When I was an IT manager, every company and every project reflected this theme.

During my career as a management consultant, I am frequently asked to perform quiet, background audits of distressed projects for the executive team. The expectation, among most of my clients, is that a *technical issue* is the root of the problem. Invariably, I have discovered that the central issue is nested among the *people*, not the technology.

Most recently, I was asked to review the Services-Oriented Architecture project successfully implemented by a major worldwide firm. They had written a brilliant exposition of the integrated framework of data, applications, and infrastructure transformed to maintain competitive advantage. Their understanding of this transformative architecture and their execution of its initial stages was impressive, a model for others to follow. However, when I reviewed their documentation, there was not a single reference to the *organizational* aspects of the project. I asked whether they had considered the possibility that the IT organization would be transformed to the same degree that they had transformed their environment. One of the directors turned to me and said, "If someone had posed that question two years ago, it would have saved us many months of organizational confusion."[2]

Like an optical illusion in a child's gaze, patterns are subtle and easily overlooked until a parent suggests that the child look for the long tail of a squirrel. The child calls out with delight, "I see it!" and after that moment of recognition, she can never again look at the diagram without first seeing the once-hidden shapes, now in the foreground of her attention, as if the original image of intersections and arcs has become transparent.

Our mirrored image is similarly hidden in the complex constructions of technology we have created during the past two decades. Unlike the optical illusion, there has been less intentional obfuscation, yet the technology is nonetheless a diagram of intersections, arcs, wires, boxes, and closets filled with more intersections, boxes, and wires. We see only what we have been taught to see, disregarding our own reflection until the author suggests that we look for the human element.

Almost immediately, like children at that moment of clarity, changing forever our view of the design, someone says "I see it!" and their comprehension of technology is altered. The original impression, the purely physical realm of circuit boards, coaxial cables, wireless relay junctions, and dumb terminals becomes transparent.

It has been ten years since I first postulated a relationship between information systems and the "people systems" that build and maintain them, in an editorial and subsequent conference paper for the Association of Computing Machinery.[3] Back in 1995, it was only a *theory*, founded on my understanding of the writings of Gregory Bateson and, later, the mentorship of Professor Sandra Braman. In the past ten years, in every company and in every role, I have witnessed its proof. I no longer consider it merely an interesting theory to be talked about briefly in hallways. It is a fundamental concept that underlies everything we do (and cannot do) in IT.

As we move toward Grid Computing and the many related technologies composing the "new IT," this principle becomes more than theoretically intriguing. It must be an integral part of your strategic roadmap. *For success on the Grid, we must transform ourselves and our organizations to the same degree that we seek to transform our architectures.*

As corporations begin to connect their systems, and as their company's networks are connected to other networks, adding to an immense and complex architecture that becomes difficult for executives to understand and impossible for their staff to explain to them in a language the executives can comprehend, an entirely new and daunting challenge presents itself. By analogy, water does not flow easily between the new pipes and the old pipes. In this case, the water is information, and as our companies increasingly become dependent on a transfer of information among customers, partners, vendors, and consultants, the myriad layers of software, servers, routers, repositories, databases, access points, devices, and the ever-increasing volume of information itself, is now a barrier.

In many cases, you simply can't get there from here.

The IT industry finds itself at yet another evolutionary cycle, with new technologies emerging with features and functions that were impossible only five years before. Such trends—web services, open source, blade technology, commodity search, distributed computing, and the Grid—offer substantial benefits. However, we find it difficult to describe those benefits to our executives, who are inclined to say, "Just make it happen," without an appreciation of cost, complication, or risk.

We need a new way of communicating with our executive teams and with the many other significant people who can influence our lives as IT practitioners. I have elected to tell stories,[4] stories that reflect the essential principles we must incorporate into our world of IT in the coming decade. I wanted to utilize a preexisting Application Programming Interface (API): the narrative format, which has already been used successfully to convey important business issues by many others, from Eliyahu Goldratt's *The Goal* to Stephen Denning's efforts at The World Bank and Debra Stouffer's strategic use of storytelling at DigitalNet.

We need a new politics in our industry, with a new vocabulary, one that provides a bridge between companies, between individuals, and between executives and their technologists.

Of course, I recognize the double-edged challenge inherent in this task: I might oversimplify complex themes and alienate the technology professionals to whom the book is dedicated, or I might make the tinsel of computers too significant and thus bore those who delight in narrative, and well-crafted sentences.

Between these two polar challenges (the Scylla and Charybdis of this book) I envision a middle ground, the place where technology is embedded in our existence, where everything is connected to everything[5] and the place where work involving binary logic is like any other work that we do each day.

We are what we build: it is a unified theory that acts prismatically, in which the elegance of heightened prose and the artfulness of "if-then" statements cast similar colors through our stained glass windows as we come home each day.

Short stories *are* distributed objects, and they are situated in this book like services on a local area network, bound by the perimeter of its covers.

The central theories underlying well-constructed narratives mirror the practices we must put into place to implement Grid Computing successfully and to benefit from it. That mirror also offers us an opportunity to learn something about ourselves.

We can no longer focus solely on the relationships between elements in a database or servers in a data center; rather, we must also consider and manage the relationships between the people that build and support them. We, who live in the IT world, are also nodes on the network, each

adding value as every phone added value to the growth of the telecommunications industry, as every web site added value to the growth of the Internet, and we will harvest the true value of this networked world only when we begin to manage the people and the technology in a systemic (unified) way.

This book presents several ways to do just that.

NOTES

1. My career spans many years of publication, both nonfiction and fiction, and includes a Master of Fine Arts in Creative Writing from Warren Wilson College in addition to more than 20 years in Information Technology.
2. My work led to a white paper entitled "Grid Management Theory," which includes a case study of this institution's significant achievements. The paper was sponsored by Cassatt Corporation, www.cassatt.com. It is accessible at www.srobbinsconsulting.com/docs/.
3. Stuart Robbins, "Turbulence and Information Systems: The System is a Mirror, (Volume 38, Issue 5; May 1995 " *Communications of the ACM* May 1995). Note: my original essay was printed beside John Perry Barlow's infamous column about the Network and Teilhard de Chardin's notions of an integrated human consciousness. Barlow's essay comes into play in Chapters 6 and 9.
4. John Allen Paulos, *Once upon a Number: The Hidden Mathematical Logic of Stories* (New York: Basic Books, 1998).
5. Kevin Kelly, *Out of Control: The New Biology of Machines, Social Science, and the Economic World* (New York: Basic Books, 2004).