ELSEVIER DIGITAL PRESS

Drew Wigodsky

RAPID Value Management for the Business Cost of Ownership

Readiness, Architecture, Process, Integration, Deployment

ENTERPRISE COMPUTING

publishing partner

hp invent

RAPID
Value Management for the Business Cost of Ownership: Readiness, Architecture, Process, Integration, Deployment

This Page Intentionally Left Blank

RAPID
Value Management for the Business Cost of Ownership: Readiness, Architecture, Process, Integration, Deployment

Andrew S. Wigodsky

ELSEVIER
DIGITAL
PRESS
Amsterdam • Boston • Heidelberg • London • New York • Oxford
Paris • San Diego • San Francisco • Singapore • Sydney • Tokyo

Digital Press™ is an imprint of Elsevier Science.

Recognizing the importance of preserving what has been written,
Elsevier Science prints its books on acid-free paper whenever possible.

ISBN: 1-55558-289-3 (pbk.: alk: paper)

British Library Cataloging-in-Publication Data

A catalogue record for this book is available from the British Library.

Library of Congress Cataloging-in-Publication Data
Wigodsky, Andrew S.
RAPID : taking the razor to information age management / Andrew S. Wigodsky.
p. cm.
On t.p. the acronym RAPID is spelled-out.
Includes index.
ISBN 1-55558-289-3
1. Technological innovations—Management. 2. High technology industries—Management. 3. New products—Management. I. Title: Taking the razor to information age management. II. Title: Readiness, Architecture, Process, Integration, Deployment. III. Title.
HD45.W443 2002
658.5′14—dc22
2003055571

The publisher offers special discounts on bulk orders of this book. For information, please contact:

Manager of Special Sales
Elsevier Science
200 Wheeler Road
Burlington, MA 01803
Tel: 781-904-2500
Fax: 781-904-2620

For information on all Digital Press publications available, contact our World Wide Web home page at: http://www.bh.com/digitalpress

10 9 8 7 6 5 4 3 2 1

Printed in the United States of America

Dedication

Cinderella: on this road we walk together—you and our boys are forever and always my teachers, my inspiration, my wishing stars, and my greatest hope.

Thank you, princess.

This Page Intentionally Left Blank

Table of Contents

This Page Intentionally Left Blank

Wading Through Acronyms

We live in an acronym-dominated world and it is easy to get confused with the barrage of new terms. Information technology (IT), or rather the people who inhabit the weird and wonderful world of IT, is perhaps the guiltiest of all industries. It is good to have a book that helps you through some of the terms consultants love to throw out, such as TCO (total cost of ownership) and BCO (business cost of ownership). Best of all, it is good to have a book that is practical in nature while forcing you to think a little.

I know Andrew has worked hard to create this book. It is a labor of love, but also maybe a millstone he has now cast from around his neck. As an author, I know the adrenaline rush that comes when you commit the last word to the page, save the last draft, and get the book submitted to the publisher. The unfortunate thing is that new thoughts inevitably pop up immediately after the book appears, so you can only ever treat a book as a starting point, something to help you along the way. I think this book succeeds in that regard. It is not perfect, but no book ever is. It does not have all the answers, but no one has. Instead, it contains a lot of useful advice about practical IT. Do not expect instant insight into why people make investments in technology, or how you are suddenly going to become the world's best innovator through your strategic use of technology. If I knew a book that could teach me those things, I would be doing something very different now, like relaxing on a beach in Bermuda watching my investments gently grow.

IT evolves all the time, so we must evolve with the industry. The only way I have found to do that is through a conscious effort to learn, almost on a daily basis. I have enjoyed seeing this book develop, and I think I have learned something from it. I hope you have the same experience.

Tony Redmond

Vice President and Chief Technology Officer

Hewlett-Packard Consulting and Integration

This Page Intentionally Left Blank

Preface

July 11, 2003
Dear Reader,

It is very difficult to make an IT strategy that–no matter the scope of each particular decision–disregards a span of time. A decade in our world of technology brought to our vocabulary many complex ideas; words, acronyms, lists, and even new languages. Ten years brought browser wars, bursting dotcom bubbles, information attacks, and computer virus outbreaks. In fact, we live in a world of devices that force us to make technology decisions every day: consider the consumer purchasing the latest software or electronically enhanced children's toy.

IT seems to change every moment. In this new world are names and acronyms (some trademarked, and all a reality for every business manager today) like HP, IBM, Compaq, Digital, Microsoft, Sun, Oracle, UNIX, Windows, "C", Linux, Java, Netscape, and eBay. These names represent some of the best and worst investments in technology, but also the promise of many new investments yet to come. But, we have also gone so far as to consider the possibility of destruction at the hands of new technology ala Y2K, and many of us even receive emails that we refer to with the brand name of a canned ham.

I see the effects of these technology-induced changes everyday on the smallest and largest of scales. This first book on RAPID presents an overview of ideas that I gathered along the way, turned sideways and intertwined with those of my company, peers, and most especially my family and friends. This book contains two decades of research and work performed with both large and small stakeholders: organizations including schools, hospitals, governments, and many types of corporations.

It includes the points of view I perceived in my past and present employers: Microsoft, Compaq, and Hewlett-Packard (HP). It also includes the thoughts of countless customers: Intel, Procter and Gamble, Principal Financial, Farm Bureau Insurance, Mutual of Omaha, and other organizations due equal parts of respect and admiration. This book, however, does not directly express any of their opinions except when cited as good examples.

I set out to tackle something that has bothered my customers for about ten years: the relationship between information technology (IT) choices and the total cost of ownership (TCO) for any particular investment. Used often in the world of the IT professional, TCO implies that you must track and control the entirety of each investment. In practice, many companies have tried to control and even reduce TCO by measuring and comparing individual vendors' solutions.

Is it indeed possible to make these decisions in a world of technology–in our world–that will span the length of a technology strategy? This book argues you can do exactly that: boldly make predictions about the future of many different IT investments. However, I should point out that I did not consider, at first, the possibility that there is no single way of making a strategy for technology investment work: in fact, the most important lesson was that only by combining ideas could you find a strategy that works.

How can you measure the complex world of technology?

When things get too complicated, it is natural to want to see them according to a simple and well-balanced model: right and wrong. The reality is that when we see information technology (IT) as a whole, our views are at the same time diametrically opposed and confusingly congruent: black, white, and a lot of gray. Some believe the solution is in the details, others believe the answers lie in the strategies of "getting there." One person believes it worth massive investment to ready the architecture and improve the real problems in technology. The other person focuses on the process of integrating technology at a better cost for business. In my opinion, both are right and both are wrong. My answer (as all good consultants will often tell you) is "It depends." In that light, I introduce this book as exactly what it is: my gripes and some very general observations about the intersection of business and IT, structured as a textbook of sorts for the entire IT department. This book is purely an experiment—and sometimes even confusing—but this is intentional.

I have some of my best thoughts while driving; the humid autumn breeze in the U.S. Midwest and a route driven hundreds of times before, are a catalyst for all sorts of dreams and other semilucid thoughts. I never remember that others may share my two hours' commute without sharing the same 120 miles of serenity and self-focus; from home to my customer—this time a state patrol department in the next state. It was almost surprising, as our team at Rainier (a U.S. regional IT consultancy, purchased in October 2000 by Compaq) had never really focused on government accounts. Deb had managed to help us win a project to help the patrol change their solution for tying their office computers to the state's mainframe. She was always willing to look outside the box to find customers, and we were barely staying afloat as the tech services world crashed around us.

I have worked (for more than half my years) in the world of technology to help business and government friends find new ways to adapt and innovate using technology. Not everyone likes what I have to say, but more often than not, even those who disagree listen anyway. On the morning of September 10, 2001—like so many mornings of my recent life—I was driving along the highway cursing the constant travel required in an IT consulting job. The hoops that my customer, the state patrol, had to jump through to keep technology up-to-date were insane. It had been years since I first worked with corporate customers to move from "green screen scraping" to PC databases. The process the department should have used to approve this very small project would have cost the state possibly more than the project itself.

The red tape, the lengthy request-for-proposals (RFP) process, the competing vendors that overpromise and underdeliver—there were so many different obstacles and so little time. They were lucky: We took credit cards, and the project was small enough to keep under the radar. The manager had long been fighting takeovers by the state IT department. The government red tape was likely the only reason the agency avoids the consolidation of the average IT department. This IT department maintained its local identity, and the customers fought to keep it local. It is not enough for the state patrol to use IT services—the agency needs the ability to innovate, to use technology as a tool to help it fight crime and protect the citizens. Their job—our security—depends on the information provided by the federal, state, and local governments. And technology is the means of keeping that critical information up to date.

Why is it important to find new ways to measure technology and information value? Consider the advice my proposal reviewer, Roxanne Burkey, gave me when I started this adventure:

> *I would suggest a section included for the impact on the enterprise user community. The costs for user acceptance/application of change in IT services are also a factor. IT departments over the last 20 years have nickeled and dimed organizations to death in an effort to stabilize network environments and permit access to users from many points. This part also should be addressed in helping again to increase credibility.*

Each of the industry analysts and other technology stalwarts has its own formula for the "business value" of information technology. The variety in these formulas is painful; just compare Gartner with IBM or Microsoft. Even when these vendors work together, their answers often appear skewed to a particular point-of-view or market strategy. Until the Internet brought the world together, nobody fully understood the scope of IT or the value of integrating its components. Put simply, before the electronic network and the Internet any manager could buy a PC and put it on a local desk without the corporate IT department ever knowing about it. As a Microsoft consultant in the late 1990s, I worked with customers and partners to understand various approaches to managing technology, especially when training customers to use the Microsoft Solution Framework (MSF). Rather than trying to add up the business value of solutions using economics, as do the total cost of ownership (TCO) models of Gartner and Forrester, MSF was a great start at defining the actual steps of how to build valuable technology solutions using a businesslike strategy: envisioning, planning, developing, stabilizing, and deploying. As microcomputer users started to request access to the corporate mainframe, the big central IT department started getting nervous.

MSF looked like something out of H. James Harrington's *Business Process Improvement* (1991) or Michael Hammer and James Champy's *Reengineering the Corporation* (1993). It was a polite attempt to tell our IT customers that they had some very bad habits. In fact, we even attended a one-day "Champy Training" that was very helpful in trying to control our often-arrogant Microsoft personas. For Microsoft Consulting Services (MCS), it was almost a mantra that we would "never do a project without a Vision/Scope." What we meant was that we would push our customers to define their world in *business terms*—in short, we made them talk like *their* customers. You cannot believe how unbridled an environment Microsoft really was. It was like trying to confront the "aromatically challenged"; you were telling people their solutions stunk. Not until after leaving Microsoft to pursue a family (my wife and I finally started shortly after our year-delayed honeymoon) did I really learn why MSF was so challenging to explain. It applied well to development, but it did not begin to address the challenges of integration, the careful balance of business and technology, architecture, and process.

The team I work with now helps customers focus on the important business parts—the architecture and infrastructure—of many different technology solutions. I use an acronym to describe this balanced relationship between our consultancy and our customer's technology. Originally, "RAPID" was simply a better term from our marketing department than "Lite" to describe a series of preconfigured technology solutions that I imagined might propel us to technology greatness in the middle of a growing recession. The first RAPID was (as Jerry Seinfeld might say) a "show about nothing": Who would buy a complicated-to-manage technology solution when neither of us understood their basic IT needs in the first place?

Once again, I found a little truth during a drive—this time through western Minnesota. As a game to avoid the desire to strangle the marketing staff, I was trying to put words to the letters of the central word they had given me, *RAPID:*

R: Readiness

A: Architecture

P: Process

I: Implementation (nope, Integration)

D: Deployment

What is RAPID really about?

What I realized at that moment, and again as I later changed the "I" from an individually focused and concrete "implementation" to a balanced concept of "integration," was that the entire concept of having a consistent two-week method for completely rebuilding major IT infrastructure was nuts. There are too many dependencies on the nature of each individual organization: the tolerance to risk and the ability to learn and innovate. However, I also found that some models can indeed significantly improve IT decision making and enable the massive potential improvements in speed, quality, and innovation.

My intention was to write an honest book to explain why I think two simple ideas are so important:

- Technology is inherently measurable, because it consists of discrete components and processes that are measurable.
- Management of technology is as much a function of your individual stakeholders as it is of groups and the organizations.

My key assertion with BCO and RAPID is that most IT management and investment valuation models (based on TCO) are incomplete:

Readiness (Chapter 4): Initiatives to measure the business costs of technology investment start at the top, using supposition and averages of averages to validate investments, often until the variables and the source data become meaningless. Any statistician will tell you this model does not work if the units are not identical. The reality is that this happens when we hide people's annual salaries behind technology and TCO. We do this instead of agreeing on formulas to aggregate the measurable costs of individual technology investments: in terms of people's time, cost of capital, and the processor cycles of individual applications deployed to individual personal business computers. People are not the machine; we built it.

Architecture (Chapter 5): We make major efforts to build life solutions using architecture (many IT consultants call this the "building a house" scenario) to enforce a predictable structure. We describe standards on how things are different, to help us focus on the unique capabilities of each part. We build structures to live in, to communicate, and to travel. We certainly build many structures in IT.

Process (Chapter 6): Attempts at caution through careful business project planning often try to circumvent the unknown; as a result, most projects are too long and often miss the moving target anyway. The true innovations are the ones we can bring to market before our competitors; providing a conflicting message when explaining the two-year timeline typical for many large IT projects.

Integration (Chapter 7): Decisions to buy or build are often the result of polarization rather than integration. IT folks talk about "enterprise application integration" without asking whether customers need or want so many confusing and hard-to-use applications in the first place. "Integration for people resources" is often another word for "downsizing." This is where we often fail, in the infrastructure, where we expect the pieces to fit simply together. In reality, the infrastructure is where we find it often takes only moments to notice an issue but days or months to determine the root cause of even the most serious technology problems.

Deployment (Chapter 8): Organizations try to implement strategy from the bottom using technology standards, usually starting with the technology (where the bigger concern is more often fighting fires, floods, and other sources of downtime) instead of the business. Even as businesses struggle to cascade the more balanced "scorecard" approach to management, we too often forget that only the individuals can best describe their potential value. We can find ways to integrate the goals of the highest level (executives and shareholders) and the lowest levels (individual contributors and customers).

Who should read this book?

People—each of us—make enormous investments in the various technologies we use to live our lives and communicate information with each other.

We depend on technology for much of our lives. This is not news; ask the average investor or business manager about what technology has recently done for (or, rather, to) them. We should now clean up this mess by asking each other some very challenging questions about how and why technology is valuable.

One of my peers in this effort, Steve, is both a former customer and my constant personal nemesis in our local team. His role as my work-protagonist (my wife is my "home-protagonist") has continued since the day he hired me as his consultant while he rebuilt our city's digital networks so many years ago. He has been driving me insane (which I sincerely appreciate!) for years, constantly reminding me about how complicated IT integration really is. He is a persistent advocate of readiness, attack on market opportunities, and goal-oriented training. As a former U.S. Air Force imagery analyst, no wonder Steve's eye trains on the many nuances and structures of such a large picture as that of the IT department. Steve represents one extreme in the spectrum of IT beliefs this book addresses. A CIO of a large U.S. insurance company (I know him from one of my previous projects at his former employer, now his competitor) explained to me the other end of the spectrum of IT beliefs.

Doug described the process of business as a stream flowing through his organization. In the stream were islands of technology—through which his organization needed to navigate—representing Microsoft, Novell, IBM, HP, and any number of thousands of technology vendors that might some day become too powerful or go out of business. His primary goal as a CIO was to maintain a constant direction and path for the organization as it flowed through the stream of technologies and vendors. He pointed out that improvement in this direction should be consistent, iterative, and careful. Focused on consistent improvement and careful integration, Doug's circles on a napkin were limited in number. In each person's lifetime, you should always find new ways to maintain some kind of focus.

I work as an individual contributor in the worldwide family of about 150,000 innovators called Hewlett-Packard. Nothing happens in just one place anymore. My life, my family, my job, and my world depend on the decisions we all make every day. In an age based on the exchange of information, this simple thought should terrify you. The more often people communicate (and globally we transact over the Internet billions of times each day) the less private or secure we really are. You cannot imagine the number of times your home address is stored on hard drives at many different companies, for any number of reasons. Some of these companies simply do not care about or understand the concept of security.

If you think like me, or Steve, or Doug—but you just need a clearer definition of what success looks like—you should read this book. Please understand, however, that some of this work is my own creative mind at work. In places I intentionally diffuse the message by incessantly categorizing things in changing ways. This habit (some people call it attention deficit disorder) lets me syncopate the rhythm of my writing—keeping you off guard but open to the beat. "In practice, it helps emphasize that my ideas are only descriptive models—that is, they require you to adapt the generic model to meet your specific needs."

I hope this book touches certain nerves: in my industries (information, communications, and technologies) in my peer groups (consultants, project managers, HP employees, IT architects, Microsoft specialists, directory integration specialists, single-income two-parent two-kid often-traveling worker families, citizens, believers) in my communities (Midwest, United States, North America, Western Hemisphere, Earth).

For this, I really want to apologize, but I cannot. Maybe I am young and arrogant, or maybe I still dream; either way, I believe we all are skewing the whole concept of "value" in the technology industry. I see value as relative in the first place. Value is relative to industries, peer groups, communities, markets, nations, and ecosystems. I am not sure I have a correct answer, but I certainly have many questions about how different things are related. For these reasons, I intentionally diffused some topics to provide a gray area. I believe somewhere in this gray area are many new answers and subsequent questions we have not yet begun to explore.

Who should **not** *read this book?*

If you want to read a book that has all the answers you need to make decisions about information, communications, and technology, please put this book back where you found it.

If you want to read a book full of deep technical processes or strategies for guaranteed IT management success, again, this book is not for you. (I intend this book to be a little confusing, to force readers to fill in the blanks according to their own unique situations.)

If you want to read a book full of specific, detailed, and focused case studies on the absolute best practices that everyone should use to manage their IT investments, again this is not the book. This book focuses on how to measure the value of IT based on your own view of reality.

If you want to read a book on a particular product, technology, or vendor (although there is a lot of HP in here, I admit), you should probably look elsewhere. This book is about customer service and efficient communications.

However, if you want some new questions and ideas to help you make decisions and investments in technology, information, and most importantly, communications—and you are committed to overcoming the fog, friction, learning curve, and resistance—this is the book to start with.

Why is this book different?

This book explores IT and our methods of communications but leaves the individual decisions to the reader. This book describes one methodology for technology management—a theory, wired-up model, or framework full of ideals and similarities across the department—instead of blueprints constrained by technoreligions and hardware inventories.

This book is foremost about two things:

- The importance of making good decisions about investments in technology, information, and communications
- The most popular opportunities to make bad decisions

This book is not about the acronym RAPID. I believe individuals and private industry have the ability to continue innovation and leadership in managing their own information, communications, and technology. If we do not use these abilities, governments will. The governments also have work to do; we can all learn to integrate, share, and generally provide much more while consuming much less. We must make the right decisions in our various uses of information, communications, and technology; our other option is to let it keep growing until it falls down.

Five years ago, at the end of my daylong "tag team" interview at Microsoft, the local executive asked me what I thought of the Microsoft-versus-Java fight. I am not sure anymore how I responded at the time, but I certainly have a response for Diane now: I do not care. Instead, I focus on helping customers make good decisions when managing the integration of information and technology. I like controversy, it makes people talk about what we think is important. I need a little chaos, because I have come to expect it. I am a U.S. citizen, first in line to doubt the actions of my government. I do invest in Microsoft—I recommend their solutions every day—but I will not for a moment forget my roots in Atari, Commodore, the Apple II, Amiga, IBM mainframes, the Macintosh, BSD Unix, AS/400, Ultrix, Solaris, FreeBSD, Linux, Netscape, and any number of past and present Microsoft and HP competitors. In fact, I sell many of their solutions today. It's called "coopetition" (*cooperation* + *competition*).

I'm also an optimist; I believe that if we can negotiate the land mines we have all set between each other through the centuries, we may find a place to shake hands in the middle and agree to put on an honest fight focused on the needs of our all-too-common yet forever uncommon customers. Focus on sharing the prize, not the fight. It is possible to create a win–win situation. We can agree to disagree. Competition is one thing, but there should be only measured room for the *Art of War* mentality in our world. We need to focus on our common desire for security, and one way to do this is to learn simpler ways to communicate.

As Alexis de Tocqueville warned us almost 200 years ago in *Democracy in America:* "A war is the easiest way in which freedom can be destroyed in a democracy, because in times of war government gains many extra powers." Mix this with a world of information, communications, and technology and you get a powder keg. We have something more important to do (as Tony said, Bermuda!), and it starts with the Razor of science.

Acknowledgements

I cannot possibly remember all of the people who helped in the creation of this book, but let me start with my family and closest friends. Without their help and many months of forgiveness that came too easily none of this would have come to pass: Tereza, Jacob, Brandon, Mom, Dad, Dan, Sarah, Glenn, Tom, Judy, Lisa, Anthony, Kathy, Paul, Brenda, Robert, Karen, David, Tiffany, Jesica, Mary, Tod, Aryeh, Shari, Heidi, and Mikey. Much applause to the rest of my family, friends, teachers, mentors, teammates, bosses, and leaders who inspire quite often by having the sense to disagree or even be wrong. Greatest admiration goes to my exemplary and patient mentors: Tom Norton, Dennis McIntyre, and Tony Redmond.

Special thanks to Steve Brown for years of keeping me in line first as a customer, then a friend, then a thorn of a coworker! He and the rest of the Rainier team stuck patiently by while I recorded them as they defined the meaning of delighted customers, even as we struggled through an unimaginable merger: Dave Eaton, Deb Best, Sabrina Biancalana, Ellen Hoeg, Terry Palmberg, Jim Fenn, Tom Peterson, Chris Fox, Troy Thierstein, Eric Smith, Steve Christian, Mike Arter, Jeff Harding, Chris Mueller, Dan Smith, Mark London, Malinda Jepsen, Tom Kern, Rich Spencer, Mike Tesmer, Kathy-Tomzak, Pat Hoyt, Dan Willis, Kim Eich, Kip Russell (to whom I owe more beer than any individual on earth), the Hofer brothers, the Rooke brothers, and at least a hundred other people that made Rainier a family. To my new teammates at Compaq and HP thanks for letting me drive you nuts in these months of searching for the best material: Pat Campbell (honorary Rainier), Lori Arriola, John Henderson, Laurie Zankel, Brad TerEick, Paul Wirtzfeld, Debbie Whitehurst, Cliff Love, Lyn Baird, Robert Uram, Dennis Self, Lyndon Wilkes, Jim Fischer, Anne Cinquemani, and Jeff Louie. Also thanks to Peter and Karen Kiss and the folks at Sentar for inspiring me with their software that actually helps you make good decisions without having to google anyone.

Thanks to Dr. Jeanne Ross at the MIT Center for Information Systems Research for understanding and explaining the information executive like no other. Theron Shreve and the folks at Digital Press took great care of me; even though I was a new author and probably asked many strange questions.

Special thanks to Bruce Siebert and the fantastic staff at Graphic World Publishing Services for helping me say what I mean, hopefully without losing and confusing anyone.

Without the following friends this would have never come together: Doug Gumm, Scott Brown, Darwin Ten Haken, Mark Clark, Joe Dehart, GregMartin, Glenn Leatherwood, John Reisinger, Ed Churchill, Ed Bowen, Andy Cook, John Messersmith, Chad Riley, Kent Tegels, Ray Gibson, Jan Rowe, John Weyenberg, Sandy Novak, Mike Mendez, Leon Frederick, Larry Grund, Steve Harris, Steve Schwartz, Steve Lewis, and the rest of the incredible customers and partners who have shared so much as they joined me on this journey.

Andrew Wigodsky

1

Taking a Razor to Information Technology (IT)

> *Pluralitas non est ponenda sine neccesitate. (Entities should not be multiplied unnecessarily.) (William of Ockham)*

There is a problem with IT; no matter who you are, from CEO to parent, the problem starts at the first moment you think about buying a new computer. If you are anything like me, it starts with an unconscious suggestion by myself, to me—usually in response to a particular challenge for a particular subject—that something out there compels enough to get me to spend money on it. This *event* (a stimulus, subsequent processing, and my eventual response) continues until I look in my pocketbook and decide to pull out a little plastic card or some cold, hard cash. Similar individual *events*, strange as it may seem, are the primary difficulty at the core of managing the modern IT department. This book explores the upcoming challenges of technology change for IT decision makers (from investor to CIO), to help each of you understand what you buy, every time you purchase a computing device. To do this, however, requires you to use a Razor to slice up IT.

Where do you start?

If you are a business decision maker, this book encourages you to consider trying a model called RAPID. Although very complicated for me to explain, in practice this methodology is rather easy. The next time you buy a computer, start with readiness. List the things you think you need, based on the event that started you looking in the first place. Look carefully at the demonstration models and Web write-ups. Notice the things that distinguish the PC. Each of these *objects*—a substantive form or useful feature—from keyboard and mouse to software to printer, modem, and Internet, does something to supplement the architecture or process of the PC itself, creating *features* (architecture or form) and *benefits* (process or function) that you find valuable. In fact, the price you are willing to pay is ultimately the value of any product or service. If you already have at least one other PC, think about the integration of this new investment. Find each of the components that make it valuable to you. Take a glance around you on the physical or virtual shelves. Look at the many similar models that are available on the shelf, and compare the features and proposed benefits of each item. Build a picture in your mind of each of the PCs in terms of your current reality, and think up a few scenarios for possible deploying the new solution.

In summary, ask yourself a few questions:

- Why did I want a new computer in the first place?
- What structure of PC features will be most important to meet your needs in step 1?
- How will this PC help you improve your own processes?
- Whom will you call for support when you are integrating the structure and processing capabilities of this PC with the nuances of your work structure and processes?
- How will you maintain your investment; what decisions should you be prepared to make to support this investment in the future?

Why is technology so challenging?

Nicely tucked inside every computing device, technology surrounds you every moment of every day. It is on every streetcorner pedestrian light and in every digital wristwatch, bearing the signature of nameless millions. Based on ideas only decades old, information technology has been powerful enough to propel our bodies to the moon and our eyes and minds to the edge of the universe. It is reshaping the world, in ways we cannot yet begin to understand. We have information technology because of the countless features and functions invented by millions of stakeholders: investors, inventors, vendors, and customers. Information (and communications) technology (IT) is both hardware and software. Only if we learn the right ways to use IT will we be able to see the fruits of our collective passion for technology:

- *Technology is powerful.* No one can avoid the potential of technology. In 50 years time the computer grew from a device only in research labs to one common in every form of business and government. In 20 years time, the PC revolutionized the technology industry and simultaneously became a major market for the global economy. In 10 years time, the Internet has changed the lives of nearly every citizen of every developed country of the world. Most dramatically, in 2 years time the popping of the dotcom bubble dented our investment plans and pocketbooks.

- *Technology affects each of us*. The technologies of the Internet boom have affected everyone—but surprisingly very few organizations have yet accepted and pursued the underlying importance Internet founders placed on building and enforcing internal standards and processes for maintaining the value of technology. Every organization has designed, purchased, built, and operated internal technology systems that—by Internet standards—are disorganized and very sensitive to the changing needs both within and outside of the organization. The uncontrolled growth rate of technology purchases in the late 20th century—likely caused by the perceived benefits of technology on business processes—has created a mêlée of rapidly depreciating technology systems and solutions.
- *Technology and business can work together*. IT executives and managers, vendors of technology products and services, governments, and others have recently begun to look very closely at the long-term costs, benefits, and most importantly, value of technology. It's not that these leaders have never before questioned technology value, it's simply that the 1990s assumption "If we build IT, they will come" has recently fallen, in concert with the market cap of the NASDAQ. People are concerned about technology—we want to be comfortable in buying new devices, and we need to feel justified in the investment. Sure—computers are cheap, but what headaches are you really purchasing?
- *Technology changes us all in real time*. We live in a real-time world. Images of war are as old as the ancient writings of ancient peoples. Images of the first world wars took weeks or months to reach only a small portion of the human race. By 1991, most of the world watched as satellite technology brought us ever so many reruns of the massive firepower that rained down on Baghdad. Twelve years later, technology brings us live global coverage from scores of individual embedded journalists—at the same time humanity "monitors" and mobile intelligent battle-damage assessors. At the same time, bloggers (authors of web logs or diaries) in Baghdad post their personal views of the situation. Moments later, video of protests

throughout the world brings us each closer to those who fight both for and against a seemingly constant part of the human condition: war.

The first authors of the world's religions did not see this future; founders of the United Nations could never have predicted the ultimate extension of our new global integration: real-time coverage of conflict, protest, battle, and war. You cannot avoid what we have created. Every time you watch TV or use the Internet—in fact, each time you drive an automobile or eat a cooked meal—you feed our common desire for technology. Once you understand these challenges, the next likely question to ask is, What should we do about them?

1.1 Occam's Razor on decomposing the myths of computer science, information technology, and business value

It is one thing to tell someone an IT solution is "going to be scalable" or "going to produce great return on investment." It is quite another thing, in my experience, to deliver on this promise. In fact, most often the greatest challenge is simply finding where to start. The great challenge of *decomposing* large objects (breaking apart the individual components to find relationships and to separate myth from reality) is not unique to IT, but our approach most certainly is. It is an *extremely* challenging task, even for the most seasoned professionals and scientists, to develop the right skills to break apart, discuss, and understand the many different structures and functions of our world.

When decomposing complex systems, especially those consisting of many different classifications of information, an element of scientific principle called Occam's Razor is a valuable tool:

> ***Skeptic's FAQ Number 1.6: What is Occam's Razor?***
>
> *Ockham's Razor ("Occam" is a Latinized variant) is the principle proposed by William of Ockham in the 15th century that "Pluralitas non est ponenda sine necessitate," which translates as "Entities should not be multiplied unnecessarily." Various other rephrasings have been incorrectly attributed to him. In more modern terms, if you have two theories that both explain the observed facts then you should use the simplest until more evidence comes along. See W. M. Thorburn,*

"The Myth of Occam's Razor," Mind *27(1918):345–353, for a detailed study of what Ockham actually wrote and what others wrote after him.*

The reason behind the Razor is that for any given set of facts there are an infinite number of theories that could explain them. For instance, if you have a graph with four points in a line then the simplest theory that explains them is a linear relationship, but you can draw an infinite number of different curves that all pass through the four points. There is no evidence that the straight line is the right one, but it is the simplest possible solution. So you might as well use it until someone comes along with a point off the straight line.

Also, if you have a few thousand points on the line and someone suggests that there is a point that is off the line, it's a pretty fair bet that they are wrong.

The following argument against Occam's Razor is sometimes proposed: "This simple hypothesis was shown to be false; the truth was more complicated. So Occam's Razor doesn't work."

This is a straw man argument. The Razor doesn't tell us anything about the truth or otherwise of a hypothesis; rather, it tells us which one to test first. The simpler the hypothesis, the easier it is to shoot down.

A related rule, which can be used to slice open conspiracy theories, is Hanlon's Razor: "Never attribute to malice that which can be adequately explained by stupidity." This definition comes from "The Jargon File" (edited by Eric Raymond), but one poster attributes it to Robert Heinlein, in a 1941 story called "Logic of Empire."

This book uses the precept of Occam's Razor to help you find new ways to look at your IT investments in terms of the most important components. You may find, however, that the point of view this book presents is very different from your current technology comfort zone. In this book, the argument is not about the products or services you need to purchase—it is not HP and Microsoft versus IBM and Oracle—rather, the focus of this book is on how to manage all kinds of technologies.

What is your "next great thing"; what do you actually hope and expect to accomplish?

Technology management should be the scientific study of business value in computers and other technology, but too often leaders and individuals make decisions based on information that, for one reason or another, shows bias for or against the hidden or unspoken needs of at least one stakeholder. In no other industry is this truer, because many of the components of greatest opportunity and risk are technological. The sin by IT decision makers of omitting details and ignoring components runs deep, leaving technology stakeholders on the precipice of either a new age or a total collapse to the global economy and social structure. Information is at the heart of this challenge.

Stakeholders at the turn of the millennium became acutely aware of the great risk of investing in products or services that we do not understand. We saw this challenge in its ugliest, basest truth in fiascos such as those that devastated Enron and WorldCom. It was not simply that the leaders shredded information or made bad business deals. For many people, the worst was that some business leaders needed so badly to cover their butts that they were willing to go to great lengths to cover their tracks—even beyond the point of fraud, to deceit and information asset annihilation. However, is it possible that no single person could see the big picture in the first place?

IT is still looking for the next great thing. Despite the economic downturn, IT leaders are unwilling, unaware, or simply unable to comprehend the component effects of their strategies. Stakeholders are still confused and looking for clear direction. In the meantime, as change slows to a grinding halt so do our investments in learning to do better what we already know how to do today. Put simply, in spite of computers the office still spits out reams of paper every day and we still demand face-to-face meetings when we need to discuss important opportunities or threats.

When you look at the issues of IT management on a macro scale, you find yourself looking at a network of, in September 2002, approximately 605 million stakeholders. These users are supported by hundreds of millions of computing devices, from handheld and watch-format PCs to telecommunications systems capable of managing the information generated by millions of daily customer events: data packets and telephone calls. Each device runs any one of millions of over-the-counter or homegrown software applications. Each IT customer has a distinct set of needs (precipitating events) that we can address. While everyone is looking for a panacea, IT's "next great thing," or even a unification theory for the physical universe, the reality is that Occam would find every one of these goals backward and unrealistic. We can set goals we can actually expect to achieve, and leave some of the harder-to-control elements to chance.

1.2 A hypothesis on the causal mechanism of technology investment

This book begins with the question "Where do I start?" I would counter that when talking about investments, perhaps a more important concern is where exactly you want to finish. If you do not know the answer to this colloquial question in your organization, you will find a never-ending set of choices that will leave you dazed and confused. All too often we ask questions and then make statements that are actually questions in disguise. There is, however, a way to bring strategy together with the infrastructure itself. The technology systems we have built have great potential, but we will only understand the value of technology after combining standard business *metrics*—measurements such as process execution, knowledge publication, organizational efficiency, and customer satisfaction—with electronic and human monitors.

In this book, you can learn how the models so far available for technology cost and value analysis are too complicated and are insufficient and unsustainable for the technology systems of tomorrow. This book, much like Occam's Razor itself, does not pretend to give one particular answer or formula for success; rather, this book explains useful models for visualizing and exploring these effects. In years of working with people who both cheer and jeer technology, one rule has helped me more than any other. It is a consultant's expression of the basis of Occam's Razor called KISS, or "Keep It Simple, Stupid." I will try to take my own advice in explaining this book now, because the material only gets deeper and more technical from here.

Two rather simple explanations describe best the basis for the hypothesis I present in this book. The first explanation comes from my "life" boss and partner, my wife Tereza: "When we stop too often to think about what we are trying to do, we start second-guessing ourselves; then we don't get anything accomplished."

The second comes from my "work" boss at the highest level, Hewlett-Packard CEO Carly Fiorina. In a December 2002 company meeting to kick off the first fiscal year of the newly integrated Hewlett-Packard, she framed the new management strategy (see Chapter 7, "How Do You Manage Operations?") by saying simply, "So, how we get things done is as important as what we get done. It is vital to delivering on the entire architecture."

What is your hypothesis, and how will you test it?

In this book, you can learn about how to move beyond TCO and other complicated financial models—to technology-enabled processes where you set a strategy, and then how to use instruments and auditing methods to measure the cost and value of technology ownership, not at a single point in time, but in real time. You can learn to see technology as a metric and to see the trends in technology management performance, much as you see the trends in customer spending or satisfaction.

This book looks deep into the systems and processes of IT and presents a formula for integrating IT value measurement into both the technology department and your organization. Most importantly, I focus on some of the best practices HP and its customers believe you need to follow to implement this type of change. However, this is not enough to help you make good decisions. Best practices can be useless in new situations, especially under the pressure from 24-hours-per-day, 7-days-per-week Internet customers. In this sense, the focus of this book is less on how to perform the practices and more on how to identify the right standards and best practices for your organization.

Too often, IT ignores scientific principles; we make assumptions and never bother to document or confirm them. We build complicated cost models but never test to see if they really work. We assume vendors have our best interests in mind. It is important in science for *you* to begin with a hypothesis, based on your own view of reality. The skeptics explain the challenge as follows:

> ***Skeptic's FAQ Number 1.2: What are the differences among a fact, a theory, and a hypothesis?***
>
> *In popular usage, a theory is just a vague and fuzzy sort of fact. But to a scientist a theory is a conceptual framework that "explains" existing facts and predicts new ones. For instance, today I saw the Sun rise. This is a fact. This fact is explained by the theory that the Earth is round and spins on its axis while orbiting the Sun. This theory also explains other facts, such as the seasons and the phases of the moon, and allows me to make predictions about what will happen tomorrow.*
>
> *This means that in some ways the words "fact" and "theory" are interchangeable. The organization of the solar system, which I used as a simple example of a theory, is normally considered to be a fact that is explained by Newton's theory of gravity. And so on.*

A hypothesis is a tentative theory that has not yet been tested. Typically, a scientist devises a hypothesis and then sees if it "holds water" by testing it against available data. If the hypothesis does hold water, the scientist declares it to be a theory.

An important characteristic of a scientific theory or hypothesis is that it be "falsifiable." This means that there must be some experiment or possible discovery that could prove the theory untrue. For example, Einstein's theory of relativity made predictions about the results of experiments. These experiments could have produced results that contradicted Einstein, so the theory was (and still is) falsifiable.

On the other hand, the theory that "there is an invisible snorg reading this over your shoulder" is not falsifiable. There is no experiment or possible evidence that could prove that invisible snorgs do not exist. So the Snorg Hypothesis is not scientific. On the other hand, the "Negative Snorg Hypothesis" (that they do not exist) is scientific. You can disprove it by catching one. Similar arguments apply to yetis, UFOs, and the Loch Ness Monster. See also question 5.2 on the age of the Universe.

It is always important to start with questions, and you may prefer to start with some of the harder ones:

- What do your business and IT customers need from technology?
- How do you justify the cost of technology improvement?
- What methods are valuable for improving both systems and processes?
- What are the challenges and benefits of the solution you have envisioned?
- How important is it to develop future-ready solutions?

Once you have identified your stakeholders' most important questions, you can then use this information to build a testable, scientific hypothesis.

1.3 Measuring IT value with science

There are many different models for determining the value, or at least the cost, of an IT solution. One of the best-known models is TCO, or *total cost of ownership*. This model, promoted extensively by IT industry analysts Gartner and Forrester in the late 1990s, assumes that organizations can achieve substantial cost savings through effectively controlling information technology costs. Microsoft, HP/Compaq, IBM, Dell, and others long accepted this model as valid for the purposes of selling technology—but most purchasers of IT have begun to ask an important question: Why, when we purchased and implemented TCO-slashing solutions, have total technology costs seemed to grow faster rather than shrink? Perhaps ownership cost is only one factor of the business decision surrounding technology purchases.

One idea under increasing scrutiny (by all involved, including the original stakeholders) is that the TCO model did not take into account the difference between cost and value. CEOs and business leaders continually demand increasing value from IT, using the TCO savings to invest in new products. TCO generally focuses on large IT "solutions," rather than explaining the organization's actual total investments in technology. As the complexity of solutions and their business relationships increased, so did the management costs—especially those of integration and deployment. Previously simple tasks, such as purchasing and installing software, became subject to immense IT scrutiny and often long delays. The delays, for non-IT business leaders, may have actually reduced the perceived value of the technology solutions. In short, it is useless to deliver a perfect solution if that solution is too late to be of value to a market. For this same reason, the analysts are slowly starting to present "value" models for IT. These models, although focused on a valid point, still have one major failing: They are often either too technical or too complicated for most business leaders to implement.

This book presents a new method of viewing ownership costs, called *business cost of ownership*. BCO is the measurable and predictable part of TCO. Business cost of ownership avoids the difficulty of predicting random events such as downtime and user "futzing," and replaces them with standard business metrics (measurements) such as risk avoidance, customer satisfaction, and process efficiency. *The business cost of ownership (BCO) is the predictable parts of a mature solution's lifetime cost, including the cost of hardware, software, vendor services, training, administration, and long-term support*. It also includes the predictable improvements in the way IT and business operate. To be included in BCO, a factor must be measurable and performed on a regular schedule or in response to common events. This book does not ignore the hard-to-predict costs; rather, it replaces them with variables you address in the form of measurable solutions and service-level agreements. In short, BCO requires you to use science to measure your solution, rather simply making educated guesses.

1.3.1 Total Cost of Ownership

Total cost of ownership (TCO) started out as a great idea to get business excited about IT. *TCO is the point-in-time average cost of a technology solution over a certain part of its lifetime: design, build, testing, implementation, operational support, and/or retirement*. Some businesses actually view TCO as what happens when you build a bad IT solution. We bought IT solutions such as mainframes, object orientation, client-server, n-Tier, desktop management, wide area networks (WANs), and finally the Internet, using TCO. Then the bubble burst. The dotcoms have been falling like leaves, often taking large corporations down with them. Business demands results, and IT came up short. Total cost of ownership has become, at best, a marketing term for a great idea gone awry. TCO was supposed to be a way for IT to sell great solutions to the business, while presenting the case in an easy-to-understand manner. Instead, it has become a sales tool, and not everyone is buying it anymore.

To appreciate the challenges of defining TCO, you should first figure out whose definition or interpretation you prefer. The IT gurus have spoken in discord, and some even contend that TCO is on the way out. One of the former vice presidents of Gartner Group, Michael Gartenberg, published in *Computerworld* of October 2000 one of the best articles to date on the challenges businesses have experienced trying to measure and manage TCO. In "Myths Behind TCO," Gartenberg asserts that customers have incorrectly applied TCO because of mistakes in analysis. He presents three common misinterpretations that he believes businesses make when using TCO:

- *You need to match someone else's numbers.*
- *First and foremost, TCO is about technology.*
- *The platform with the lowest TCO is always the best choice.* Excerpted from an opinion posted at http://www.computerworld.com/managementtopics/mangement/story10,10801,52999,00.html and accessed on 5/1/03.

Gartenberg is unfortunately correct, but the biggest challenge is that most business customers would tell you we never fully understood TCO in the first place. It did not come with a detailed instruction manual, and various companies and people offer numerous descriptions of what should be included. Some categories overlap, some only appear in a single vendor's model. Worse, TCO is a moving target that often seems to correlate with the release of some vendor's new products. Radicati (www.radicati.com), a popular source of TCO advertising for Microsoft and IBM, provides an annual report on the neck-and-neck race for email TCO between Microsoft Exchange and IBM/Lotus Notes. A quick review of Radicati's annual reports on messaging-system TCO seems to indicate the products are simply both moving closer to even. The reports do not pretend to tell you how to best integrate email with your existing systems, and they do not tell you how to maintain your TCO over a long period. At its best, TCO is simply a target for technology management efficiency, correlated to a particular solution at a range of potential costs.

1.3.2 Business Cost of Ownership

The basic intent of total cost of ownership is—and will continue to be—valid. Each of these systems will indeed cost money to both implement and support. Each system will affect both the IT staff and the user. TCO lets IT vendors remind their customers that a recurring cost is associated with supporting technology. Unfortunately, many organizations do not focus on the most important point of TCO: The majority of the costs involved with owning technology are by nature recurring, interrelated—and in many cases, even predictable (see Chapter 9). By failing to ensure a well-structured, thoroughly evaluated, and clearly documented system, you assume that the system you design will never require maintenance, upgrades, or support. In fact, you can assume the opposite: *Everything you use will continue to cost you money until it is gone*.

If you want to understand business cost of ownership (BCO), you need look no further than the actual ways that information technology affects your business. BCO consists of the TCO parts that are measurable, without guessing the ultimate responsibility for fate or chance. BCO focuses on the development of solutions that we immediately assume will need some form of long-term care and feeding until you upgrade, replace, or retire them. Businesses will still need to monitor the other parts of TCO, those without a predictable impact, outside the context of BCO. (See Figure 1.1)

Where TCO seeks to eliminate downtime because of downtime's unpredictable nature, BCO focuses on reducing the cost of recovering from failure of a downed system, or even on reducing the time to record the reason for failure in a support database. BCO is a model for gathering information, statistics, and data to put business sense into every decision made by IT. Executives, managers, architects, programmers, and users can use BCO to study the various known impacts of IT. Potential BCO solutions are easy to find; just start in the last place you would normally look: processes that happen

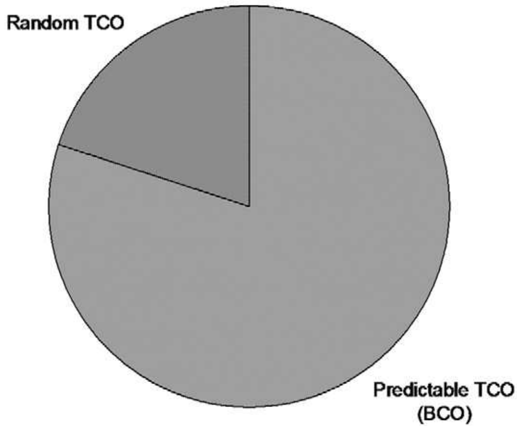


Figure 1.1 *Relationship between total cost of ownership (TCO) and business cost of ownership (BCO)*

every day. Every business has a process that is out of control, communication that takes too long, or a customer who wants more responsiveness. Each of these items is measurable.

BCO is not simply a cost model; it is a framework for the relationship between business and its technology service providers. BCO does not assume a vendor's product will solve a business problem; rather, it assumes the technology providers should use these products to design business solutions that are flexible and supportable. Some organizations may even choose to implement BCO together with RAPID (see Chapter 3) as a framework for defining vendor contracts and internal service-level agreements: between business and IT, between IT and its vendors, even between managers and project teams.

TCO addresses predictions of computing costs for a particular solution, and BCO is a framework used to tackle "ever-growing projects" created by implementing or consolidating frequently used enterprise solutions into an organization's infrastructure. Although this book frequently uses HP and Microsoft as examples, BCO is not specific to their solutions and you can use BCO and RAPID as guidelines for developing cross-platform standards for optimizing technology investments. Most organizations have already begun to address some components of BCO, but only through the broad goals of TCO. Although organizations may try to manage ownership costs through control of overall budget, we can also strategically plan and manage BCO costs at every level, to prevent recurring increases to overall TCO budgets. Focused on the predictable and tangible costs of system ownership, the BCO framework allows organizations to quantify the benefits of enterprise systems consolidation, expansion, or integration. This book presents and explains one model for continually monitoring, predicting, and improving a technology investment's BCO.

1.3.3 Implementing BCO Using RAPID

This book documents the experiences and best practices of consultants and customers that took significant career risks to invest in and to insist on both performance and manageability from their IT solutions. To help a variety of readers relate to the content of this book, I use a model called RAPID to organize experiences as described by various users, computer operators, programmers, IT consultants, system architects, executives, and even some of today's young Internet junkies. You will also learn how to use BCO to measure some of these experiences, providing you with a framework for making RAPID decisions.

The RAPID framework allows you to

- *Prepare for IT change:* Assess the readiness and operational efficiency of your infrastructure.
- *Design the critical future:* Structure solutions designed to maximize the benefits of the Internet platform.
- *Build best practices:* Test and validate your target architecture while developing effective processes.
- *Develop integrated solutions:* Integrate management of multiplatform IT services using Internet technologies.
- *Maintain satisfied stakeholders:* Deploy new solutions while ensuring the business efficiency of the infrastructure.

RAPID is not, however, a cure-all for what ails information technology. The framework cannot tell you what decisions to make, but it can help you make measurable improvements over time by keeping you informed throughout the decision-making process. RAPID is not a complete methodology for IT management; the method is simply not detailed enough for this purpose, but can be used to supplement most IT and project management methodologies. RAPID can serve simply as an auditing model, used to ensure that information technology providers understand the needs of both the organization's and IT department's business. In fact, the key deliverable of both RAPID and BCO is simply good decision making.

1.3.4 An Overview of RAPID Decision Making

You can use the decision-making framework called RAPID to explore the similarities between detailed real-world solutions and best practices submitted by my peers and coworkers, other experienced IT consultants at HP and Microsoft. The framework helps you to organize the resources and processes needed to address the challenges of your own Internet-era planning, design, testing, and deployment. (See Table 1.1)

Chapter 2 discusses how to decompose IT components to classify them for scientific study. Chapter 3 explains the relationships between architecture and process, and the essentials of how to apply RAPID decision making to common technology projects. Chapters 4–12 expand this knowledge with a detailed look into some of the challenges facing real IT managers and projects—the reasons we can move "beyond TCO." You will benefit from descriptions of real-world solutions, designed to maximize business value from investment in Internet technologies. To help set a common and strategic direction, this book presents solutions in the context of preparing your organization's infrastructure to support, use, and interact with Internet

Table 1.1 *RAPID decision making*

Readiness	Understand the vision and risks. Prepare appropriate resources for implementation and maintenance. Gather statistics about the current state of resources and stakeholders.
Architecture	Plan for the most effective and appropriate architecture. Identify and evaluate related systems and processes. Measure and prioritize architectural and economic requirements.
Process	Test architectural concepts by building clean solutions. Address feedback from users and decision makers. Measure required changes to existing systems and processes.
Integration	Integrate new systems into the existing environment. Address resource issues and stakeholder concerns. Measure the integration impacts to stakeholders and resources.
Deployment	Maintain systems and processes. Enhance and optimize the architecture. Measure the system value and performance. Make good decisions!

technologies. You can evaluate each phase of a RAPID decision in this context and consider your own responses to these common organizational challenges.

Readiness: How does one prepare for IT change?

It is not easy to get ready for IT change; like all changes, such change starts with both uncertainty and opportunity. In the world of tomorrow's data centers and 99.999% availability systems (or so they promise!), we will not continue to accept the uncertainty or "futz factor" of previous IT changes. As IT continues to compete for business investment, executives and other business leaders want reliable information systems that respond quickly to common challenges of organizations:

- You must understand the requirements and issues affecting both stakeholders and resources.
- From executives to disk drives, it is important to prepare the entire organization and its resources for upcoming changes.
- Preparation sometimes requires massive coordination and effort to develop a complete inventory of your resources, even if some of the resources do not want to cooperate.

In Chapter 4, you will learn best practices for getting your organization ready for large IT projects such as infrastructure consolidation and Internet platform implementation.

Architecture: How do you design the future?

It is difficult to understand the relationships between technology components when approaching a large solution, especially when the blueprints or schematics of the existing ones are often missing. To build an appropriate architecture, it is important to develop thorough knowledge of the new solution as well as effects to the current environment. Enterprise technology architecture often changes with each new or modified solution. Designing a cost-effective IT architecture specific to the organization is important:

- Implementation teams should identify relationships between new and existing components throughout the organization, while constantly managing process changes.
- Each organization requires a specific architecture of standards and procedures for building enterprise solutions.
- Organizations should build flexible standards that address the needs of business while optimizing long-term operational costs.

In Chapter 5, you will examine best practices for designing valuable technology solutions using both existing resources and new technologies.

Process: What are your best practices?

Any opportunity for reducing human or machine processes is a measurable event you can record and study. One outstanding method for delivering cost improvement in any technology solution is to maximize the business value of process improvement. To reduce processing and process support activities, you should deliver packaged and instrumented solutions (like thermometers and wind speed meters—but for IT). By forming web services for each solution, you can create technology solutions to help drive measurable process improvements, such as reducing the time and effort to hire and prepare for a new employee. Process affects many parts of an IT solution:

- Build and test each process of an IT solution in a clean environment, and work with stakeholders to clarify the requirements and desired end-state of the solution.
- Evaluate, understand, and prioritize architecture changes that stakeholders request.
- Communicate on a schedule with stakeholders to discuss new discoveries and the measurable effects to existing systems and processes.

In Chapter 6, you will examine best practices for packaging and monitoring new processes while systematically reengineering your information technology.

Integration: How do you develop integrated solutions?

It's not enough to "put up" a good solution; rather, you must constantly adapt the solutions to the changing needs of the business. After designing a great solution, it is important to expedite investment returns while minimizing impact to users and the organization. Developing flexible integration methods can be a valuable method of achieving project goals. Using Internet-based integration methods, organizations can address common issues while providing technology flexibility based on globally accepted standards. Integration is often the most dangerous part of any technology solution:

- New and updated systems require careful integration into an existing environment, to prevent serious impacts to users and other stakeholders.
- Stakeholders are most concerned about how the solution affects them personally, and challenges during integration may require solution changes to minimize integration impact.

- Changes during and after integration require accurate measurement and careful consideration.

In Chapter 7, you will examine best practices for integrating new solutions into the existing environment while minimizing negative effects and preparing for sustainable deployment.

Deployment: How do you keep stakeholders satisfied?

Organizations should deploy the best solutions quickly but carefully, with minimal impact to continuing operations. No solution can ignore the long-term costs of operation and ownership—solutions should adapt quickly to meet the ever-changing needs of business. Organizations should implement a standard schedule for each group of stakeholders (weekly, monthly, and quarterly) to review and manage changes after deployment. Deployment is often the longest and most costly phase of a technology solution's life cycle:

- Organizations must keep track of and maintain every component of a technology solution—both the technology and the people it affects.
- To respond to the natural changes experienced by the organization and its environment, technology requires constant monitoring, enhancement, and optimization.
- Most technology is easy to measure—the key is using measurements to discover and enhance its value.

In Chapter 8, you will examine best practices for maintaining the deployment of scalable solutions focused on driving business value.

1.4 Developing your own IT value experiments

Technology value is not a well-known measurement. In fact, the value of technology is probably a variable closely related to your needs. Put simply, technology has no value without someone to use it. To understand the challenges of measuring IT value, consider your own answers to the following questions:

- What can you do to measure, maintain, and enhance the value of technology investments?
- How do you manage changes and challenges to your technology environment?

- How should you account for technology-related impacts to your user or customer?
- In a rapidly changing economy, how important is it to keep technology assets current?
- How do you develop a statistically valid model for describing technology value?
- Who is responsible for maintaining technology value?
- Are other factors affecting technology value?

In Chapter 9, you will take a deeper look at BCO while considering some of the most important factors not addressed by TCO. By using the iterative approach of the RAPID model to enhance IT decision making, you can begin to notice opportunities for improving the lifetime performance and cost of your largest IT investments. If you develop IT solutions to include both detailed instrumentation and a flexible architecture, you will find it easier to understand and maintain the value of your IT investments.

How will you build a vision for the next generation of technology?

Every day, information technology becomes a more visible part of our lives. For those growing up in Generation X, the Internet showed up in colleges before 1990. High schools found the Internet within a year, and corporations began heavily investing soon after. The growth of Internet usage has surpassed the initial growth of radio, television, and even the personal computer itself. By the mid-1990s, it was impossible to watch television in California without hearing someone say "www.*somebody*.com." By the late 1990s, the Internet had grown into a worldwide center of education, research, commerce, communication, and even crime and terrorism.

Generation X was also the first generation to grow up with PCs. We used modems and electronic bulletin boards to connect users in every country, and what started as PC-to-host and PC-to-PC communications eventually became the Internet boom. Some of us have chosen technology-dominated lifestyles, and others have not. Almost all of us have email access, and millions of us spend hours per day on the World Wide Web. When building next-generation information technology, we must always consider not only the demands of today's solution but also the effects of tomorrow's needs on today's solution. Our children, many of them becoming computer- and Internet-savvy during grade school, will redefine the role technology plays in society.

What will the next generation demand?

Although some consumers will always avoid technology, the next generation will overwhelmingly demand it. In the home, in school, in the workplace, when voting and paying taxes, this generation will never know life without a computer. From the first days in school until their last days of life, this generation will direct and orchestrate the application of technology in ways we simply cannot imagine.

Generation Y may be the first to stop using most paper forms in everyday life, preserving trees for only the most important documents and books. Our children will learn more, faster than any generation before them. Exposing our children to the diversity of the Internet is both dangerous and powerful. Given the right opportunities, our children's generation has the potential to cure cancer, bring an end to hunger and war, and develop some kind of Internet-based *Star Trek* world that is sure to make life even more confusing than it is today.

The next generation of technology must be flexible enough to meet the demands of these users, and optimized to meet the needs of value-oriented investors, consumers, and citizens. The growth of the Microsoft platform since its introduction has been an amazing tribute to the flexibility and value of the personal computer in a business world. Windows, Office, and Internet Explorer are the most (and often only) common basic components of all PCs. The next generation of Microsoft technologies will be a fundamental change in direction, using Internet standards to provide basic services for enabling the next generation of computing. Agree with them or not, Microsoft owns the PC world and has set its sights very effectively on the internet generation.

How will you get there?

Holding together the remnants of the Internet bubble are at least three generations of information technology vendors, users, workers, managers, and executives. Each generation brings new methods, new products, and new challenges. The typical IT environment has 20 to 40 years of software and source code to support, running on hardware that changes hands every 18 to 36 months. Many PCs still use variations of Windows 95, almost eight years after its introduction. Most Intel servers still run some version of Windows NT. Dotcoms and home users quickly adopted Windows 2000 and XP, but these platforms have only begun recently to breach the very risk-averse corporate and government infrastructure. As an HP consultant, I have watched as customers have slowly decided that at some point they will likely have to deal with an Internet-centered world.

Systems do not always work together, and major IT projects are disruptive to both the information technology and the business. The PC itself has long needed a major redesign to become a "BC," or business computer. Microsoft and .NET may not be the perfect or only answers, but Microsoft-inspired concepts of automation, Web services, and loosely coupled systems described in this book are a reality that IT and business must not ignore.

How will you organize your enterprise to be ready for Internet effects?

Each organization and every industry have different views of their technology needs. Consumers, in fact, have led business in determining that—much to the chagrin of the technology industry—new technology is not always better technology. The resulting challenge to buyers and sellers of technology is a market that often defies logic and reason. How else could one explain that in the PC industry declining prices have led to increased inventory and waning demand? Not everyone wants a new PC, even as PCs continue to perform better with each passing chip speed.

The costs of preparing for the Internet can be staggering, but the cost of *not* adopting some form of Internet-centric infrastructure could be disastrous. After evaluating the explosion of applications directly supporting users during the Internet 1990s, many organizations found that localized and inconsistent purchasing, configuration, and support of common applications not only caused difficulty implementing new technologies but often also causes a significant increase in total cost of ownership (TCO) when organizations try to integrate Internet technologies with their infrastructure.

Microsoft, as an example, has created ".NET" ("dot net") to strike a balance between the needs of Internet-enabled organizations and the assets they use to create their current infrastructures. In preparing for its change of focus from individual proprietary applications to full-featured Internet standard service platforms, Microsoft has redesigned its business model to address the demands of a very meticulous customer base. Over time, popular PC solutions grew into systems that organizations must periodically consolidate to maintain a supportable and flexible architecture for business unit applications. Internet solutions such as email and web portal services are good examples of new solutions that have quickly grown to become a vital part of the next generation infrastructure.

Because of the massive scope of organizational Internet needs, many organizations are considering sizable investments in technology infrastructure. To quantify the importance of building cost-effective Internet services, it is important to specifically address the common costs that a quality infrastructure can affect. The infrastructure includes the common products and services required to satisfy the needs of the majority of users. Infrastructure includes the systems that allow applications to communicate, such as disk storage and transaction processing. Infrastructure also includes the services, such as networking and messaging, required to connect applications and users. Finally, popular applications can quickly become part of the infrastructure. For organizations making a significant investment in the Internet, the infrastructure includes the devices, tools, services, and standards used to support internal and external customers.

The task of getting ready for the next-generation Internet is not simple: It requires an organization to develop detailed and accurate inventories of every technology component used to support the organization's current internal and external information systems, processes, and other resources.

1.5 Conclusion

To help your organization get ready to service Web customers, make sure you address the following key issues introduced in this chapter:

- Identify the most important measurable costs and benefits of addressing your customers' needs.
- Communicate with your customers to determine and categorize their desires.
- Review the experiences and best practices of others in your role that have faced a similar challenge.
- Remember to match the investment to the potential rewards, not just the possible cost savings.
- Learn about new ways to increase your organization's efficiency while ensuring the agility to respond to the changing needs of customers.

We may not ever see a paperless home or office, and we will likely have to continue showing up at work for a while. Nevertheless, in the lifetimes of each generation alive today we will face a fundamental struggle to find out where technology and the Internet really belong. In this book, you will also learn why businesses, governments, the Internet, HP, Microsoft and the .NET strategy, Java, Linux, digital media, and the next great technology will always be in the fluid epicenter of that struggle. To keep track of this mess, however, requires some science.

Reference

1. Reprinted with permission from Paul Johnson, "sci.skeptic FAQ: The Frequently Questioned Answers" (1996). Available on the Web at http://www.faqs.org/faqs/skeptic-faq/ and accessed on June 26, 2003.

2

On the Classification of IT Components

Crude classification and false generalizations are the curse of organized life. (George Bernard Shaw)

The landscape of technology management is changing for every organization; the IT department has become a critical part of every mission, and business decision makers are again becoming more responsible for influencing technology decisions. At the annual Gartner Symposium/ITXpo 2002, one of the more controversial predictions, according to an article in *Computerworld*[1] was that "business units will eventually become responsible for proving the potential value of IT investments." Does this, by implication, mean that the same business units will become responsible for the long-term care and feeding of technology components?

Although some IT managers may see this trend as desirable, the dangers here are indeed great: What happens to the technology for which nobody wants to claim responsibility? In the same sense, what happens to investments in technology infrastructure that affect every part of the business organization? The answers are still somewhat unclear; however, the immediate result is an amazing new library of guidance on business cost justification aimed squarely at the CIO and his or her managers. TCO, REJ, ROI, NPV, IRR, and other similar acronyms* refer to a growing set of knowledge that indicates not all technology investments are good ones—but that some IT projects indeed have a potential value that extends beyond the realm of IT and into the core

* TCO: Total cost of ownership (Gartner, Forrester and others), REJ: Rapid Economic Justification (Microsoft), ROI: Return on Investment (Investment analysis), NPV: Netpresent value (Investment analysis)

of business. These tools let us methodically evaluate the potential value of certain IT investments to organizations and customers.

The devil, however, is in the details of how we classify technology investments. According to another *Computerworld* article, "It's nearly impossible to apply IRR to infrastructure or equipment upgrade investments, such as expanding a server farm . . . not all IT executives see a need to use sophisticated calculations to cost-justify IT investments."[2]

This is a significant challenge; from this observation, some quandaries appear:

- Why does it take such sophisticated models to cost-justify IT investments?
- Which investments do economic models such as internal rate of return (IRR) *not* address?
- Why do the models fail to address these investments?
- Is cost justification really an important component of IT?

The answer to this dilemma is not a simple one. It runs deeply to the core of not only information technology, but also communications and business in general. Technology has radically changed the world we live in, for better or for worse, throughout history. To answer the question of how valuable information and technology really is and how important it is to manage this value, we must understand the formative sources of technology value itself.

The challenge is that our current model for IT is based on long lists of products, vendors, and programming languages. Each list of technology components seemingly has its own system of classification. As the technology world changes, the models change to become iteratively more complicated. Consider the most common classifiers for the IT department:

- Hardware (generation, vendor, brand, bus architecture, memory architecture, interfaces, and so forth)
- Software (platform, language, vendor, brand, features, functions, and so forth)
- Services and support (vendors, consultants, contractors, upgrade projects, and so forth)
- Training (products, languages, management and so forth)

2.1 A recent history of the IT department

After Y2K, the pattern of technology investment changed dramatically. The *bubble era*, as one news writer described it, is gone. Also gone are the frequent departmental purchases of every form of technology—and the revenues of vendors dependent on these purchases have evaporated as well. Entire segments of the technology industry are in chaos. Many companies have closed their doors forever. Even once-powerful Netscape did not survive. Throughout the technology industry, companies feel the recent impact on their bottom lines as pricing pressures have shredded margins and profits. The top line has dropped as well—it dropped in the same double digits of the previous gains. The pain is also being felt in the governments now experiencing the turmoil of reduced tax revenues, because of post–"bubble era" spending patterns. No one remains unaffected by the bubble burst—but most people don't even know it.

Most people, that is, except for technology managers. This particular group has personal knowledge of the effects of the bubble burst. On the purchasing side, the entire IT world seems to be on sale: Products, services, contractors, and employees have all seen massive cost drops as the market changes. On the selling side, margins wore thin and revenues are down almost across the board, at least when compared to late 1999. Internal knowledge assets in technology organizations (both vendors and IT departments) are at an all-time low—rapid business change, sparse documentation, and massive turnover have left many organizations in the dark when it comes to the actual state of their technology systems.

At the same time (or maybe as a result) the executive floor has tightly locked its wallet—and has begun to demand the detailed financial analysis and business planning that have long been required of almost every other department. On the other hand, you may find it plausible that the slamming of the IT wallets is simply because the other business counterparts are better at pleading their case to the CEO in a recession. Maybe it is easier for the CFO, for example, because they have the backing of financial audits and many years of standardized historical performance.

2.1.1 A brief history of technology

To understand why technology management is so important, you must consider the history of the practice—surprisingly, one of the most important aspects of our advancement as a species. Humans have long managed technology to support our own development. Out of necessity and primal drive, we created complex processes for understanding, developing, using, maintaining, integrating, modifying, and retiring every type of technology. Technology management is not new; however, we have recently applied it to a new product: computing devices.

In the later part of the Stone Age, humankind began an aggressive reshaping of the world around us, in part based on exchanges of information and technology:

- We began to take possessions and make permanent claims to land. Some of these claims were recorded information. We use this information to make observations on both our history and our modern world.
- We learned to tend animals and plants. We created increasingly complex technology to help us in agricultural effort. We took technological items as property, used technology to develop and store resource surpluses, and trade of commodities emerged. This allowed us to divide labor, forming at the same time larger communities and greater stratification in our societies.
- We stopped moving ourselves and learned how to move goods instead. Villages, parishes, and townships united, forming cities and states. States grew into great nations and kingdoms. Our communities became more complex and moved further apart; as a result, our communications needs changed.

When many people began to move again (not until the Industrial Revolution, about 7000 years later), so did information. We learned to build tools to assist in our work: the manufacture of textiles, steam locomotion for trains and ships, and industrial processing of raw materials into finished goods. In the centuries to come, human beings learned to build machines and proliferate other technology to help us with any imaginable task. From feeding the masses to waging a war, each technological machine and its surrounding processes played a part in our development as a species:

- From the ancient Chinese abacus and Sir Samuel Morland's mechanical calculator to the newest scientific calculator of today
- From Morse code to the Internet
- From the ENIAC to the handheld PC

Our desire to process information has helped us invent technologies to take us to the moon and beyond. Technology changed the way we work, play, and even wage war. Computing processes have rapidly replaced human processes.

With each technological generation, computers have grown faster, cheaper, and smaller. According to Intel cofounder Gordon Moore, computer processors give us double the work every year at the same size and cost. Is this value?

How will technology changes affect value?

Consider that a certain part of the technology leaders' and analysts' predictions actually come true over the next decade. Assuming we find a way to continue microprocessor improvements under Moore's law, by 2010 the current power of a Pentium 4 processor should fit in a wristwatch for a fraction of the cost—a very valuable scenario, assuming we find an appropriate way to use and power a device that small. By 2020, as prices for equivalent processing and communications continue to plummet, many people will be permanently connected to their families, organizations, employers, and the Internet using computing devices that are truly both personal and mobile—beyond the current personal digital assistant (PDA)/mobile phone combo and various other portable PC shapes.

Many of these devices will recognize and process voice communication, and larger models will include portables that recognize handwriting. Some users will still use larger desktop PCs. Servers will come in both horizontal and vertical dimensions that, using blade-style* components borrowed from telecommunications, will place thousands of servers in the space of a small refrigerator. The management of these systems will be much like managing a mainframe today—partitioned into business service areas and functional applications. By this time, much of the daily business paperwork will have disappeared forever—from expense reports to vacation requests—replaced by electronic systems that allow the business to monitor its rapidly changing systems and processes in nearly real time.

Our lives in the future will change rapidly with technology: Internet communications today bring information and rich content such as music and video, available for a click of the mouse. Homes, classrooms, and offices of tomorrow will connect to a vast network of multimedia information available at the touch of a fingertip—where three-dimensional displays will enhance the experience by allowing interaction with virtual objects. Games will never be the same. Virtual actors will try to upstage Hollywood in a manner never

* Small for factor "blade" technology allows computing vendors to compact the processing elements of a server onto a surface size of a small handheld calculator (the typical size of a computer interface in PCs today). The user purchases a chassis that connects many blades together, and then purchases and inserts individual blades to meet growth in computing needs.

imagined. Movies such as *Toy Story* and the latest computer-animated Disney films are only the beginning: It is a high-tech world out there.

As the cost of technology products continues to decrease, these devices will grow and change in ways we cannot yet predict. Whereas organizations of today must get ready to compete with newer, more efficient competitors, businesses of tomorrow must understand that the learning curve is rapidly increasing. Children, raised from birth with the Internet, will make demands on society that relate to the changing nature of information and communication—especially technology—further stretching a reality we cannot yet perceive. The next generation of information and technology has already started. Are you ready?

2.1.2 Technology of highest importance

How important is your technology—not just the technology in your workplace or your home but also the technology that supports your life? The questions about technology value certainly exceed the answers:

- What technologies are the most valuable to you, your organizations, and your customers?
- Do you know all the places that technology affects you?
- How do you choose the right technologies?
- How do you measure and maintain the value of your technology?
- What happens, and who is responsible, if your technology fails?

One class of computer measures the power production and distribution requirements of every machine in the world. Another class of computers is responsible for processing each telephone call placed anywhere on the globe. One more is responsible for routing Internet traffic. There is even a class of computers responsible for maintaining the purity and availability of water you use to drink, cook, wash, clean, build, refine, mix, engineer, sail, and even fish. What happens if all these computers suddenly stop functioning? At the turn of the year 2000, we considered what would happen in that scenario—if the lights truly went out.

The fear of Y2K problems poured trillions of dollars into the global economy, simply because the technology departments of the past wanted to save on storage costs (storing the date as “99” instead of “1999”). Companies and governments spent IT dollars in a frenzy to mitigate the risk of Y2K disasters—despite the admittedly low probability that anything “critically bad” would actually happen. Many governments passed laws and resolutions requiring mitigation of the Y2K risk. Entire industries made radical changes to prevent Y2K disasters. For these reasons and others, the pocketbooks were open wide.

Clearly, we see immense value in technology—but the actual measure of that value remains an unknown. The technology industry has only recently begun to understand the factors that affect technology value, and on these observations no consensus exists. Tech companies have developed an alphabet soup of methods for measuring value. Executives, managers, and business consultants have tried desperately to determine an appropriate process for managing the many ways in which technology affects our world.

For many reasons, but most of all for the sake of generations to come—we must learn to make the right decisions about technology. We need to make technology decisions as a consensus, not just a simple majority. Technology vendors and departments must be sensitive and responsible to the needs of their users and other stakeholders. We must also find ways to help technology support and enhance the productivity of workers, not replace them with electronic decision makers. These are weighty responsibilities indeed, especially considering that you must combine this with the knowledge that not everyone desires or needs technology.

2.1.3 The next IT infrastructure

In technology, infrastructure is of the highest importance—it is the medium used to connect all the parts to form a whole. The Internet has rapidly created a massive gateway and infrastructure for information of every type and has quickly grown to become an integral part of the global economy. We have used the Internet to break down communications barriers that, a decade ago, seemed impossible to overcome. We have used the Internet to share information, to share ideas, and to bring the people of the world a step closer.

Corporations have invested billions of dollars in new technologies—primarily applications—expecting dramatic investment returns. Many of these projects, although looking great on paper, never made it into production: The users simply cannot use them. Sometimes this happens when the network is too slow or the operating system is too old. Occasionally the databases are not compatible, or perhaps the old driver does not work on a new system. Applications built decades apart often require significant modification to work with the systems of today, and these modifications are expensive. Enterprise application integration is a whole field of its own. Building systems that work together is an occasional afterthought, rather than a steadfast requirement.

Whereas the dictionary definition of *infrastructure* refers to foundations, facilities, and services, the information technology definition of infrastructure often refers to gigahertz, gigabytes, and those mysterious “dark fibers.” We often assume technology will just “plug in,” whereas in reality very few organizations build IT infrastructures that truly operate that way. For some organizations, infrastructure is a component of each application. For others, it is simply a map of the network. Very few perceive the importance of building coherent infrastructure. In January 2002, as the post-Y2K economy soured, even Harvard University closed its Information Infrastructure Project because of an “unanticipated financial shortfall” after two decades of work. This trend, however, cannot continue. Although some may argue that the Internet provides the ultimate infrastructure for connectivity of IT systems, the Internet does not address the massive costs of supporting applications and services provided inside organizations. (See Figure 2.1)

Our work, however, is far from complete. The Internet would never have grown so rapidly popular were it not for its foundation of mature but flexible standards and processes. Without both the significant past investment and subsequent architectural and procedural compromises that institutions and governments made to develop the Transmission Control Protocol/Inter-

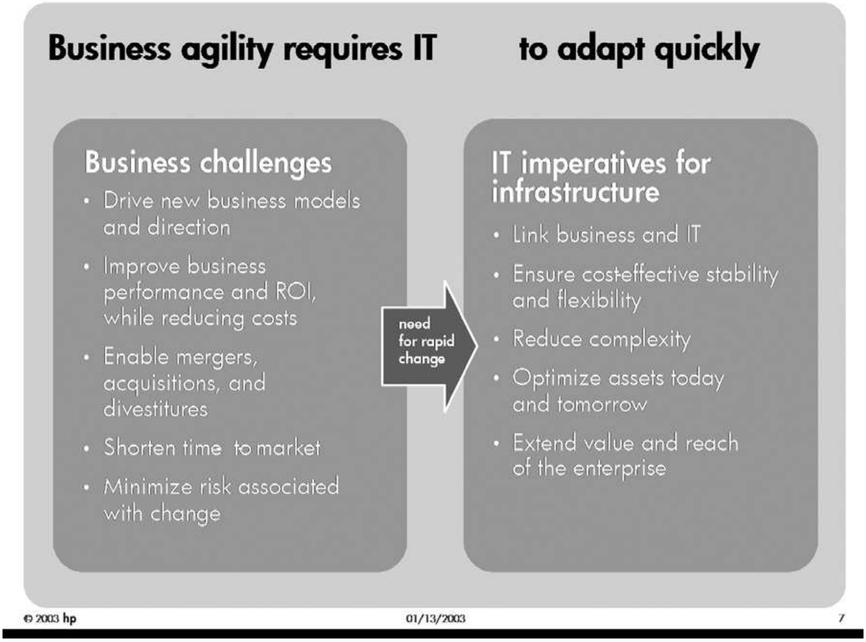


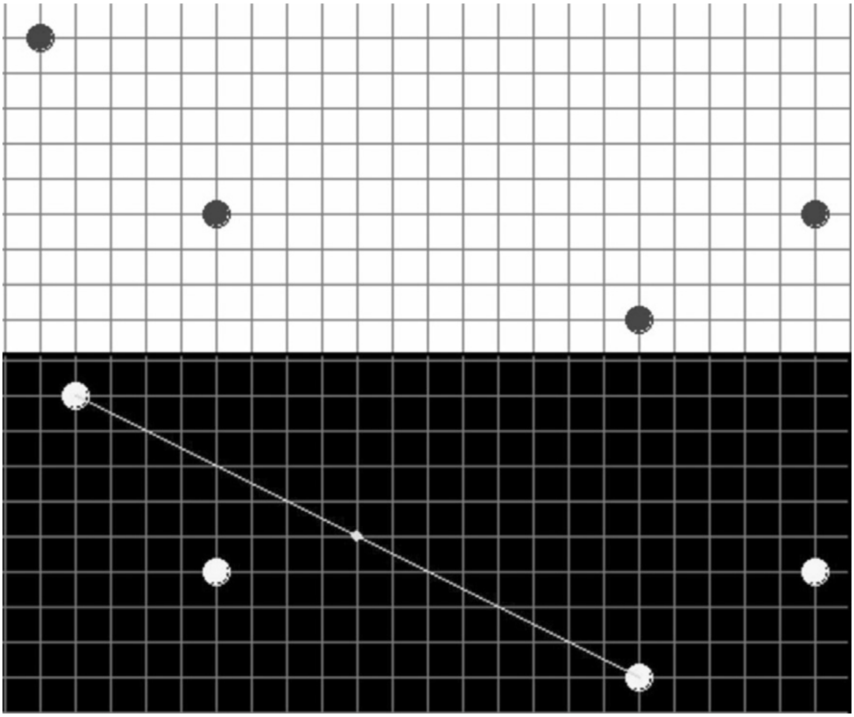
Figure 2.1
HP’s view of the next IT infrastructure

net Protocol (TCP/IP) communications language, the Internet would have never grown to become the revolutionary force it is today. In fact, without TCP/IP there would be no Internet—no Web sites, no chat rooms, no EBay, and very likely no AOL (America Online).

2.2 IT through a pigeon's hole

The pigeonhole principle is a rule of the balance that is inherent in math, science, and nature. For instance, in its most basic form it states that (assuming you have cooperative pigeons) if you put 10 pigeons in 10 boxes, and when you are done 1 box is empty, then it is a certainty that 1 box has more than 1 pigeon. Why is this so important? Alex Bogomolny gives another example of this principle on the Mathematical Association of America Web site at http://www.maa.org/editorial/knot/pigeonhole.html. In this version of the pigeonhole principle, Bogomolny explains the principle by allowing the Web user to place four or five dots on the intersections of a grid. (See Figure 2.2)

Figure 2.2 *"Four dot" pigeonhole analysis*



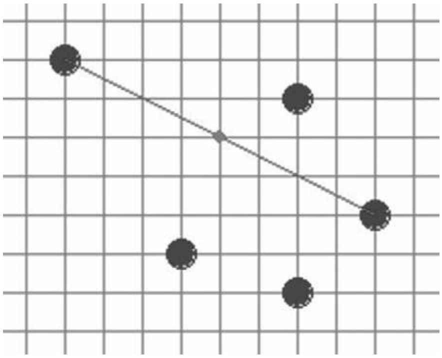


Figure 2.3
"Five dot" pigeon-hole analysis

As Figure 2.2 illustrates, a situation that has four points has two possibilities. It is possible, as in the top image, to place four points on the grid without any two defining a line segment that does not have a midpoint (a center) on a grid intersection. Another property is that to make an intersection appear or disappear, you only need to move one point. Each point is capable of throwing off the balance. (See Figure 2.3)

When there are five points, however, at least one of the potential midpoints connecting each pair of two points lies on a grid intersection. There must be a relationship between the structure (four or five points) and the function (finding a midpoint at a grid intersection). This rule is important, because it applies to most things in nature. Have you ever tried to make a challenging decision with a team of four? In negotiations among four people, a disagreement could easily lead to a tie vote—an impasse—beyond which they cannot move.

2.3 Classification of information and technology

As humans, we tend to classify our theories of the universe and expect up front that we have found steadfast rules. In truth, even the rules of science find basis only in "acceptable theories." We often forget about change; *Plus ça change, plus c'est la même chose (*"The more things change, the more they stay the same"). If you can take a sufficiently classified view of anything, you can find ways to measure it before, during, and after change. Finding the similarities is the key to measuring *deltas* (as in science, the difference between two states). For information technology, the challenge is that we often cannot see the forest for the trees. Instead, we often try to change only the individual trees (components) while leaving the forest (solutions) subject to

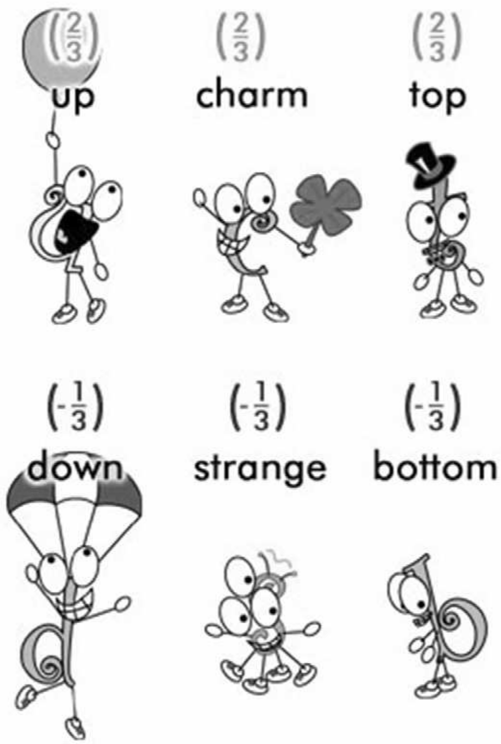


Figure 2.4 *Classification of quarks in Physics (Source: http://particleadvantage.org/particle-adventure/frameless/quarks.html. Accessed April 29, 2003.)*

traditional hierarchical management. This, alone, does not suffice; in IT, we must classify and organize the trees for specialists to maintain according to categories such as color, size, species, and location. Classification is a major aspect of scientific process. (See Figure 2.4)

Scientists love to arrange information. They classify the visible world, from physics to biology. They arranged our knowledge of dinosaurs into species that were primarily either "lizard-hipped" or "bird-hipped" (a human observation on animal body architecture). The bird-hipped dinosaurs such as Stegosaurus and Triceratops looked little like each other, but were much more similar than lizard-hipped dinosaurs such as Diplodocus and Tyrannosaurus.

Classification is not new to IT, either:

- Computer-processing architectures can be classified into *big-endian* and *little-endian*, referring to the order in which data are stored while being processed.

- Object-oriented computer software is often stored in components called *classes* that reflect the hierarchy of the processes they support.
- The IT Infrastructure Library (ITIL) classifies repeatable IT processes for managing services into disciplines such as "change management."

Classifying objects into categories helps us find similarities between them. (One group of dinosaurs has long, whiplike tails, where one group of information technologists speaks Linux and has ponytails.) Like physicists and paleontologists, information technologists will be required to dig into our solutions and classify them, to make them more palatable for customers. To classify the natural relationships that affect IT solutions correctly, we must understand them. The key to relating science and IT is that as nature's needs changed, the animal's skeleton changed based on "splits" in nature's design. A few early dinosaurs, we cannot classify in any category. The importance of identifying and prioritizing these splits is the key to measuring the effects of IT change. The splits are changes, and they affect us in sometimes-peculiar ways that may later seem obvious.

2.4 Conclusion

Classification allows you to maintain an organized view of a very large set of components, such as those in a typical IT organization. Classifying components is a good beginning, but the classification system is most useful if it allows you to observe and maintain the relationships between components.

Once you create these classifications, it is necessary to understand the differences and similarities between the components you classified and the classes you created. Creating the classifications also creates more work. To identify an maintain the relationships between components, it is important to create a consistent process for managing your architecture.

References

1. Thomas Hoffman (2002), available on the Web at www.computerworld.com/printthis/2002/0,4814,74990,00.html.
2. Thomas Hoffman (2002), available on the Web at www.computerworld.com/managementtopics/roi/story/0,10801,74447,00.html.

3

Using RAPID to Identify Relationship Between Architecture and Process

The ultimate test of a relationship is to disagree but hold hands. (Alexander Penney)

Like any other business organization, information technology must learn to provide reliable forecasts. No matter how you describe it—in the 1970s it was *data processing*, the 1980s called it *information systems*, in the 1990s it was *information technology*, and now it is ICT *(information and communications technology)*—IT is like any other business investment. Organizations demand returns on investments. When the dotcom bubble burst, enterprises looked hard at information and communications technology—and what they found was an incredibly complex web of internal and external resources, each managed according to different metrics in a different way and often without any documentation. Put simply, business caught IT with its pants down. The structure of the IT industry seems to run against the process of business, but we have only hidden the answers in the complex networks so typical of humanity.

3.1 Implications of change for the IT department

Hidden behind the Internet—behind email, instant messaging, and every Web site—are *servers*. These servers (computers that serve others' needs) use networks to communicate with printers, scanners, storage, and (at many workers' desks) a PC. The PC is most often a user's playground; it is often the user, not the IT department, who controls the functions of an individual PC. Some users even further connect their work PCs to a mobile computer or mobile phone. Computing devices are everywhere, and much of computing really is as complicated as it seems. Businesses have tried to lock down or manage PC desktops, with minimal success and at great expense. Implementing new software—say enterprise resource planning (ERP) or customer relationship management (CRM)—is an absolute nightmare for many organizations. Worse, all information technology needs frequent service and is prone to decay and aging, like any other machine.

Each of these devices runs software applications: instructions to instill in the computing devices the ability to make decisions according to circumstances (and other business rules): "IF A = 1, THEN B = 2." This software is very important: It makes everything happen. From phone calls to financial transactions to online relationships, software contains processes that people have written to perform a certain set of functions according to a known set of parameters. In a networked world, this software connects to software that connects to software. Some software parts come from vendors, and some each organization develops internally. When the software somewhere in that chain fails, an application fails. Sometimes the users get an error message, but more often they get a headache. That is when users start getting mad and wanting to kick the hardware—as they would any other machine.

Behind these often-malfunctioning applications are the same networks mentioned earlier—this time connecting both the applications and their devices according to complicated "webs" of networks and network services. Devices connect applications to a local network, which in turn often connects (because of organizational structure) to any number of subsequent local networks and devices using a wide area network or intranet. The Internet compounds the problem by connecting your networks to the world's network—which, again, is quite similar to a playground. Each network in the chain is subject to any number of failures—from hardware, software, or connections. Again, no matter what breaks, the user wants to kick the closest computer.

Hiding further behind the wall of devices, applications, and networks are people. No matter in which direction you look, you find people who

- Select, purchase, design, plan, install, manage, and change IT
- Create solutions, write software, build systems, and teach IT to perform valuable functions
- Use IT to reduce work
- Use technology to instruct both people and technology
- Define how technology changes and grows to meet people's needs

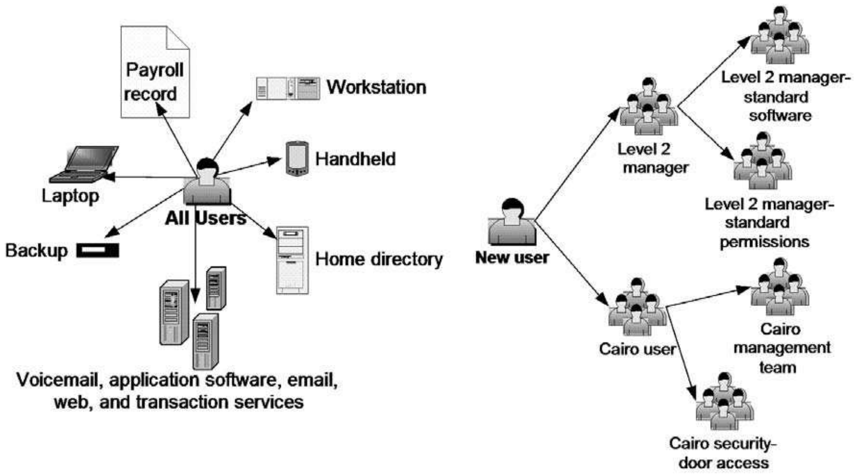
In a financial view of the business, both technology and people are resources: cost of goods sold. Both people and computers are lumped together in a bucket, according to the processes they both support. For technology investments, you cannot be concerned with the business alone. You must be concerned with the technology as well. Most of all, you must consider the people. The "people needs" are the ones that cause the most frequent changes.

3.1.1 IT + business changes = IT process flux

According to their role, people in an organization often require a certain set of technology in order to function. Some employees use a word processor, others need a database—but many roles require both. In the same sense, support staff are assigned to similar roles when supporting technology—some support word processors, some support databases, and some support both. Some roles include other roles, defined by both individual responsibilities and capabilities. Sometimes large groups of resources share the same role—and often large groups of roles share the same person. (See Figure 3.1)

This arrangement works well—until (whether by lottery, promotion, layoff, or simply taxi) somebody leaves, and leaves behind a group of roles and responsibilities. In IT, the result of an employee leaving is often a hole—in two weeks we simply cannot learn enough about what someone did. Right there lies the fundamental and eternal challenge of people, processes, and technology: The world changes. In business, some changes

Figure 3.1 *The natural variety of IT needs*



are due to the constant flow of employees and other stakeholders in and out of a business. People change or "turn over," and with this change comes changes in both processes and technology. You may need to hire someone new, or you may need to distribute that person's role among other resources. With each employee lost goes knowledge, and that knowledge is often valuable. Whether your business is governing, thinking, manufacturing, or even parenting, you are subject to a constant process flux that changing people create with technology. You cannot effectively manage the change of people or technology without accommodating the "process flux" that affects these changes.

IT process flux is a phenomenon whereby technology changes in response to process, architecture, and integration changes in a human work system. As the users grow, so grows the IT. Like flux in physics and medicine, IT process flux is an effect related to movement—specifically, the movement of people and technology. Much like particles or energy flowing through a surface, people's needs flow through IT. As the needs change, we must be able to respond to changes in flux (flow) by clearly understanding the thresholds for acceptable performance. Much like any blood vessel, we must be able to grow and shrink, to respond to cold and hot—while still maintaining the integrity of the passageway.

For IT to be effective,

- When growth is at its fastest, we must innovate and build solutions.
- When growth slows down or recession hits, we must be able to maintain the infrastructure and the solutions while simultaneously improving our performance.
- In all cases, we must always maintain the channel controlling the flow: the infrastructure.

It's not easy to manage IT process flux. Change happens both because of and despite IT itself. In *Net Ready*[2], authors Hartman, Sifonis, and Kador observe that in the electronic economy (their "E-conomy"), past performance does not equate with future success. So we must find ways to adapt quickly to the changing needs of our stakeholders. We must also learn to manage the future, rather than the inevitable past. This requires vision and communication, creativity and consensus. Remember the sage advice *Tempis fugit:* Time flies. Once the present becomes past, it is too late to measure what you never thought to measure. Don't wait—hurry up—be RAPID about it!

3.1.2 Impacts of the economy on technology Management

What do you get when you mix information technology with a recession? During the Internet boom, customers saw value everywhere—in almost every IT investment, whereas after the dwindling of the dotcoms customers have demanded value (and cut budgets) with unrelenting vigor. For a vendor or customer trying to justify an IT project, large or small, the economic environment can significantly affect the willingness or even ability of any business to fund any technology project. Some suggest the slowdown in technology is cyclic—that it is simply a "bump in the road" of the technology revolution. Others argue the most important factor was the overzealousness of the early Internet climate mixed with the frenzied buying right before Y2K. A small but growing number of past technology buyers are simply sick of broken promises and missed expectations—software and hardware that never seemed to return anything on any measure of investment.

More often than not, the most plausible answer is that we IT folks do not understand how to sell the value of information technology. The challenge, however, is bigger than it may seem. Our industry has grown more quickly than any before—and has changed the very definition of many important industries that came before it. Using technology we have redefined workplace, school, home, and country. Each generation of technology has led to a new and more provocative opportunity. Every 10 years or so, the information industry has managed to create something business has found compelling enough to invest in:

- The massive yet expensive information-processing power of mainframes
- The personal yet frustrating experience of a PC
- The up-again, down-again network
- The freedom and insecurity of the Internet

Technology is not immune from the redefinition it has imposed on other markets. As we grow, we must be able to develop and refine our abilities using information technology. For IT services, status quo is never acceptable. You must carefully orchestrate and execute progress and use the best technology for the right processes at a fair price. After building a solution, IT must be able to monitor its operation and clean house like any other department—frequently reinventing itself to address new challenges, opportunities, and customer needs. When approaching some of the most critical IT decisions, the only defense against the profit-sapping effects of process flux is communications. Stakeholders—whether customers, shareholders, managers, or employees—must all communicate in order to first understand and then set priorities on the most important aspects of major organizational change. If the goal of change is a paradigm shift—say, retooling the computing infrastructure of a brick-and-mortar company for the Internet age, then the potential impacts on people, processes, and technology are stunning.

3.1.3 The need for models in IT

Whether you are building a log cabin or a skyscraper, chances are you should work with architects and builders to develop a detailed model long before breaking ground. In many places, you may need a permit and a building inspector to verify that the project is safe and appropriate. Getting approval differs with each project: Sometimes all you need are blueprints and building instructions, but often the most challenging projects require the architect to build a scaled-down model (in cardboard, on a computer, or likely both) of the design, long before the bulldozers and cement trucks appear. Workers and managers face similar inspections in a wide variety of industries: manufacturing, health care, food production, finance, and even defense. Why would IT be different? It probably isn't, but we surely act as if it were.

Much like the early days of the industrial revolution, the entire IT department sometimes seems to be a deep void without standards. On the average, IT looks like the railroad right after someone decided to create standards for track sizes. Without these standards, trains would never have been able to move easily between cities, states, and countries. These standards, in fact, are still in place today—constantly updated for the continuing changes in the rail industry. Standards help prevent too much advantage for any one vendor while helping integrate solutions from many vendors. In IT, we have only begun to learn that creating standards is a challenging but rewarding political process: It causes both competitors and allies to team up to fight disarray, for the common good.

For most corporate and government IT departments, continuous improvement of written technology standards is the exception, not the rule. Although some standards have been recently set, most of the back office for IT still consists of systems built over decades. Some organizations, especially insurance companies, are still using PC applications built in the DOS days—the computer disk operating system that preceded Windows. These 8-bit applications, from the late 1980s, are three complete PC computing generations behind today's 64-bit applications. The worst part is that most organizations must continue to support most or all of the applications they have ever supported—from mainframe to PC and from 8 bits to 64! Each application has the ability to birth a new set of children. Massive technological change simply resulted in more processes, not all of them better. (See Figure 3.2)

The most popular model in the world for organizing IT services into best practices, ITIL (Information Technology Infrastructure Library) is only in use by a small handful of organizations. Some organizations have created their own standards for technology management, not based on any common model. Many organizations have no model for managing IT other than the IT organization chart locked in the CIO's office. We developed specific and common models for IT project management and execution, but there is little consistency between the ways in which IT services operate once in production. Put simply, IT does not have a common method for building a model of what our customers want. The goal of ITIL is to arrange these processes in a much more structured, friendly model. (See Figure 3.3)

Figure 3.2
Typical management architecture of people, process, and technology

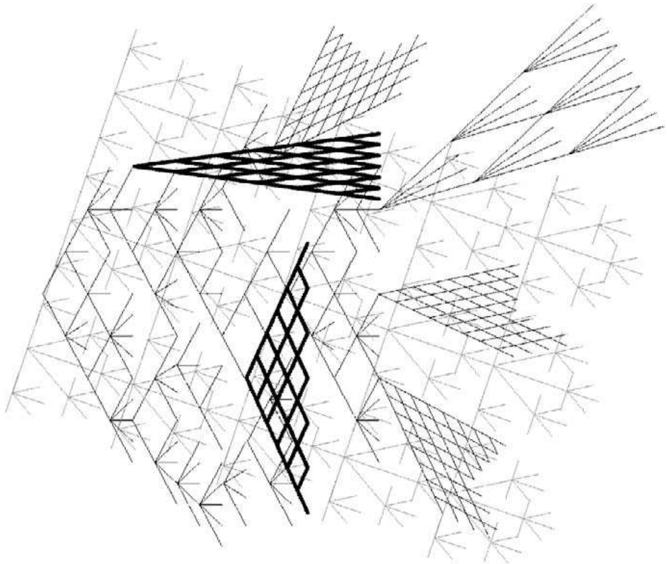
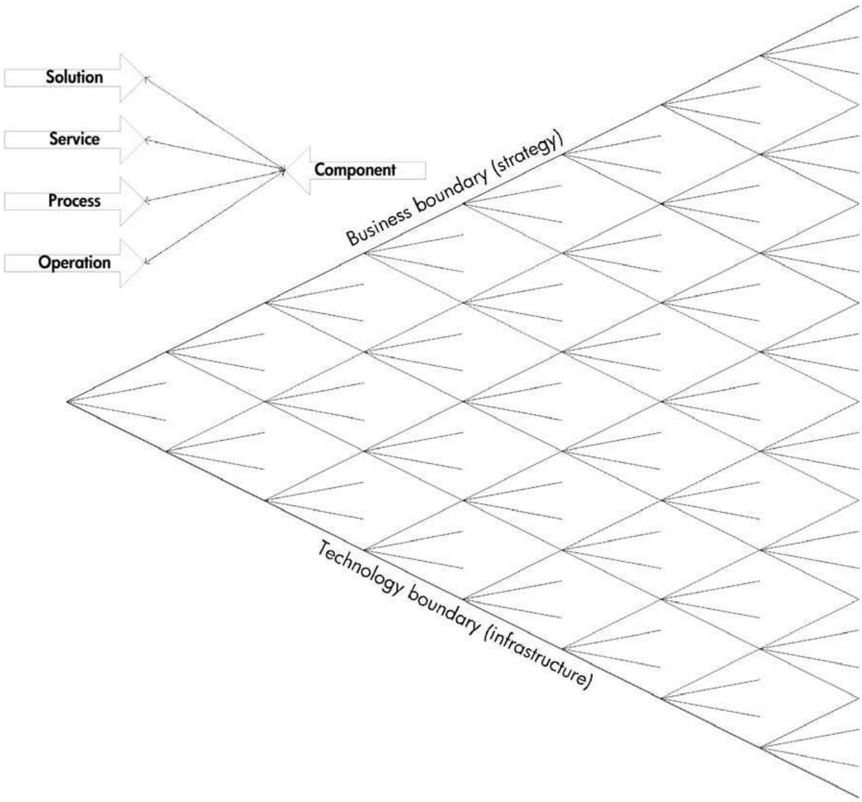


Figure 3.3
Balanced management architecture for people, processes, and technology



Why is it important to use models?

We all lose when our customers cannot articulate the business value of technology investments. For both HP and our customers, making the best possible IT investments is paramount in winning the battle for business investment. The challenge, however, is that we "IT folks" speak a language fundamentally different from that of the businesses that fund both our projects and our paychecks. CEOs and CFOs are demanding financial analysis of individual IT investments using traditional and respected economic models such as *return on investment* and *net present value*—but the technology industry has only begun to apply these models in its own business justification efforts. The risk in learning these models is that they mean little to the average tech worker.

Understanding and implementing these traditional financial models is not the only challenge facing the IT department. Once we deliver the economic projections, the business expects the technology to deliver what it promises.

IT executives and managers are subject to the same level of scrutiny (or perhaps more) as any other business unit. To earn trust and receive additional funding, IT must compete with other business units to meet long-term projections while demonstrating measurable business value. And to meet this challenge with any hope of success, IT must learn to help customers understand the important parts of the technology solution within the business solution.

3.2 RAPID is a razor

The goal of the RAPID method is not to "fix" IT problems; rather, the goal is to help you record a baseline against which you can measure. In that sense, much like Occam's Razor, it forces you to classify the relationships (the variables) in your theories. Adapted to the needs of the information age, RAPID requires you to create models for both business and technology management. Occam showed us why models are important. They help show, visually, how and why an investment of one kind or another makes sense. Models are a visual representation of an object—larger, smaller, or otherwise inefficient to reproduce—that allows you to understand its structure and function. Many of our most important achievements in science are due to the use of models: models of the globe, the atom, and even the universe itself. We use models to describe the finer points of roads and bridges, power distribution, city planning, fashion, weather, and the effects of nuclear or biological attack. We even use models to predict the future.

Skeptic's FAQ Number 0.6.4: How can I persuade the other side?

This isn't a FAQ, but it should be! Originally this question referred only to persuading skeptics, but of course the paranormalists are not the only ones who need to learn how to argue.

- *Be prepared to offer evidence. Ideally, evidence consists of an experiment I can reasonably do myself. Failing that, list articles in peer-reviewed journals.*
- *Make predictions. These predictions should be specific and surprising. For example, a prediction that "crime will cause concern" is not specific (it does not say who is going to be concerned about what aspect of crime when) and it is not surprising (someone, somewhere is going to be concerned about it). On the other hand, a prediction that "The British House of Commons will hold an emergency debate on juvenile crime next month" is both specific (it specifies an*

event that either will or will not happen) and surprising (emergency debates on this subject don't happen every month).

- *Be prepared to look at the evidence presented by the other side. On the other hand, if you claim as your evidence a paper that came out in some obscure journal in 1903, don't be too surprised if no one goes to the expense of digging it out just to debunk it for you.*
- *Don't try argument by assertion. A statement such as "The evidence for psi is overwhelming" will generate lots of queries asking where this evidence may be found. Conversely, the "extraordinary claims require extraordinary evidence" line should only be used when someone tries to shift the burden of proof.*
- *Don't try argument by authority unless the authority you are citing is generally acknowledged as an expert on the subject. I might cite C. S. Lewis in a debate on the nature of Christianity. I would not cite him on the age of the universe, because he is not an authority on that.1*

For more on how to construct a logical argument, see the talk.origins and talk.atheism FAQs, both of which have extensive sections on this subject.[3]

Information technology is not immune from the effects of science or the economy. IT is subject to the same forces that affect traditional disciplines, yet we are only starting to mature as an industry to a level where we are capable of enforcing the same standards that other services have come to expect. Power has standards for voltage, gas has them for octane, medicine considers life expectancy, and virtually every industry is subject to metrics—measurements that organizations use to both monitor and evaluate a particular industry. Do we ask the business to base IT on processor speed and storage bytes, or should we be proposing different metrics? It is important to build a method of modeling that will let us see the big picture of IT value.

Where can you find best practices?

There are many places to find best practices for IT. Books on business process improvement, IT process improvement, IT architecture, enterprise application integration (EAI), and other topics important to developing best practices—the right things to do—for IT. By implementing best practices, you will be able to communicate with all the stakeholders about how IT can reinvent itself to meet the needs of its customers with greater expertise. One of the best sources of IT process knowledge is ITIL, a library of best practices for IT service management conceived during the late 1980s by the UK government office of commerce:

> *IT Infrastructure Library (ITIL) is the only consistent and comprehensive documentation of best practice for IT Service Management. Used by many hundreds of organizations around the world, a whole ITIL philosophy has grown up around the guidance contained within the ITIL books.*[1]

ITIL is only the beginning. IT industry vendors, analysts, and publications have placed a significant focus on the importance of best practices lately. One standout, Gartner, has emphasized understanding and optimizing the total value of information technology. Gartner has focused lately on the cost of managing information technology after deployment and seems to have marked it as the best possibility for IT process improvement.

It is not sufficient to consider only the IT process improvement. You must also review business guidance such as that found in books like *Business Process Improvement: The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness* (H. James Harrington; April 1991), *Reengineering the Corporation: A Manifesto for Business Revolution* (Michael Hammer, James A. Champy; June 2001), and *Shared Services*: *Mining for Corporate Gold* (Barbara Quinn, Robert Cooke, Andrew Kris; April 2000) Process improvement is not new to business, and you can often use it to help make a compelling case for potential business investment returns. Process improvement can be an effective method of management, but as most current studies will indicate, it can be overdone. Psychologist Abraham Maslow, for instance, observed that it is easy to get a temporary uplift in performance—at the sacrifice of worker satisfaction, customer satisfaction, and long-term profits.

This book presents best practices you can customize for your own organization, specifically those described by BCO and RAPID. In this chapter, you will learn how to couple best practices with traditional concepts of business improvement: best practices, architectural guidance, careful integration, and effective management. To acknowledge the importance of IT in this endeavor, you will also learn about the importance of architecture to IT management, as well as specific ITIL best practices for ITSM, IT service management.

How do you manage IT services?

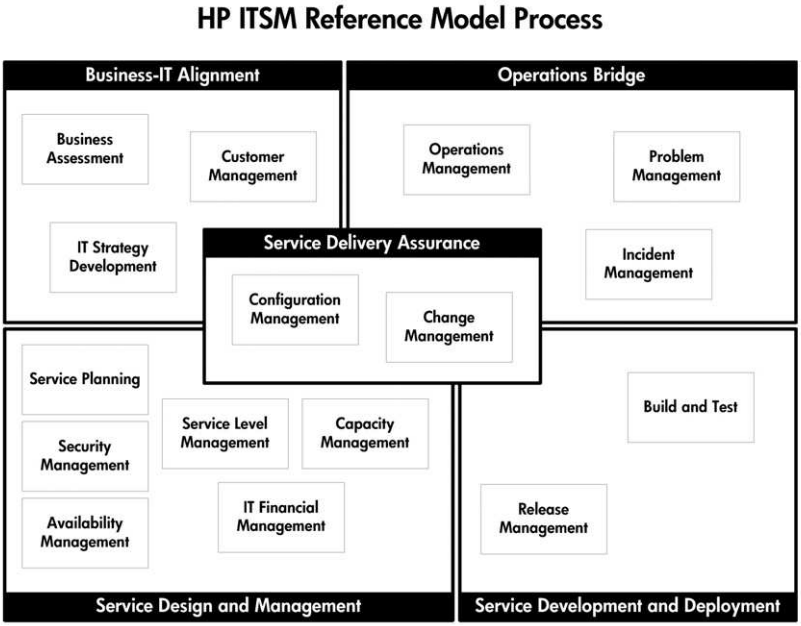
IT service management is an architecture for process reengineering, a business-driven approach to reengineering IT. ITSM focuses on delivering valuable IT solutions to business users at agreed-on service quality and cost targets. Targets are broken down in to easy-to-understand components based on the services provided by IT to the business. For Hewlett-Packard, ITSM represents the architecture for business implementation of the ITIL process guides, important documentation for managing some of the most important and challenging IT processes. HP uses ITSM to organize ITIL into the following core areas:

- Business—IT alignment
- Service design and management
- Service development and deployment
- Operations bridge
- Service delivery assurance

3.2.1 Using RAPID to understand ITIL and ITSM

Information Technology Infrastructure Library and IT service management provide one excellent avenue for IT departments to make a technology and business case for investment in information technology. By focusing on the best practices required for implementing an IT services infrastructure, these process-focused models allow users to understand

Figure 3.4
HP ITSM reference model



what services we actually provide to the business. By using an architectural model to increase our understanding of how business views our services, ITSM will prepare better for reinventing IT according to the needs of its customers. (See Figure 3.4)

To make good IT decisions, technology decision makers must understand and address each of the five common elements affecting every technology solution. The ITSM model helps us understand the most important processes in IT management, as well as how to explain them to the business. The RAPID model helps us understand the technical implications of models such as ITSM. You can use RAPID to model your path to implement ITSM or any other IT management model. The acronym is a common word for a reason: It reminds us that our customers must often make decisions quickly, to keep up with the rapid pace of Internet-era business.

3.2.2 RAPID management strategies

This chapter is about one method of managing IT process flux. RAPID is the IT side of the business value equation. Just as business categorizes an investment into different buckets, you can use RAPID to categorize IT into manageable components. RAPID is a framework for measurable IT investments, from the IT point of view. Most specifically, RAPID is a method for making good decisions and reducing risk by accurately auditing or modeling an information technology investment or strategy. RAPID is an acronym for the key components of any IT decision: readiness, architecture, process, integration, and deployment. By focusing on these components and their inherent relationships, it is possible to create a flexible model for detailed analysis of ever-changing IT value systems.

Much like any other audit or inspection, RAPID focuses on the most important components of validating a decision, according to a standard model for success. If you are building a house, you will not want to ask the building inspector if your house is attractive—only if it is a safe building plan that meets the physical and legal requirements of the community and the particular lot. A financial audit does not guarantee success: It simply confirms that an organization has followed the most widely accepted accounting principles. The most important job of an inspection is to ensure that whatever you are doing was a sound idea in the first place. RAPID is one method for building a model used to communicate the core structural requirements of an IT solution:

- To the customers
- To the contractors, architects, and builders

- To the managers and directors
- To the inspectors

RAPID is not a complete methodology for IT management, nor does it define an IT project management methodology. RAPID is about implementing, within information technology, the ability to inspect our foundations and structural integrity. It does not tell you how your solution should work—it only tells you whether you have considered the most important elements of a particular IT decision. RAPID does not guarantee success; it only seeks to make sure you define what success looks like, in terms appropriate for all the stakeholders.

RAPID is just one example of a model you can use to create your own models, to focus on the key decisions you make before implementing a big solution. By focusing on the most important components of a solution first, we can help customers move very quickly from paper to production. The RAPID solution model was not built for speed, however. We built it for accuracy and consistency in proposal generation. We had plenty of different yet successful methods of delivering technology solutions projects, but we had always built solutions in a time-and-materials ("pay as you go") fashion. With our customers closely watching the bottom line, we needed to build projects that were conservative yet flexible investments for both our customers and our employer.

To articulate the expected results of any technology solution, you must be able to explain clearly and accurately the impact of a particular opportunity on each of the following elements. You can remember these elements by using the acronyms RAPID and RACE:

- *Readiness:* What does the business need to do to prepare for the decision?
- *Architecture:* How will the solution's architecture support technology needs?
- *Process:* Where are the opportunities for process improvement?
- *Integration:* How and where will the solution fit?
- *Deployment:* How do you measure success?

And

- *Resources:* What resources does the solution require?
- *Activities:* What documents and milestones are required?
- *Constraints:* What restrictions affect the resources?
- *Expectations:* How do you measure success?

3.2.3 Implementing RAPID in your environment

Once you understand the specifics of RAPID, you will need to determine the best way to integrate it into your environment. RAPID is a model for technology decision making, and as such it assumes you will continue to use some level of methodology to ensure that its components are properly documented and considered. Alone, it will not tell you what decisions to make—but it will help you list the decisions required. RAPID does not answer all the questions—but it provides a thorough list of questions that you need to answer to develop a thoughtful decision.

RAPID is an iterative approach to both quantitative and qualitative analysis of an IT solution. Each successive activity (or phase) requires you to produce an increasing level of detail about your decisions. Each phase can be broken down into components that are similar to the whole. Within each phase, the overall phases reappear. For example, if you apply RAPID to technology implementation projects, you might create the following checklist of tasks to be completed: (See Figure 3.5)

To be clear, although this book presents information on all five phases of RAPID, it only includes the level of detail required to get ready for Internet solutions. To include the level of architectural and process details required for actual implementation would require volumes of information beyond the scope of this book—much more than the RAPID model addresses. It would also require focus on a specific solution to a specific problem, which is similarly out of scope. In fact, one of the best uses of RAPID is to determine the most appropriate scope for components of a particular solution—what is important, and what is not—to make great decisions.

How does RAPID modeling help?

This chapter introduces the core elements of RAPID, in the context of using them to introduce IT service management. By learning how you can use RAPID to describe ITSM, you will be able to communicate better

Figure 3.5
RAPID checklist for technology implementation (example)

	Readiness					Architecture					Process					Integration					Deployment				
	R	A	P	I	D	R	A	P	I	D	R	A	P	I	D	R	A	P	I	D	R	A	P	I	D
Hardware																									
Software																									
Services																									
Training																									
Administration																									
Support																									

with your technology stakeholders by helping them both understand and implement the best practices of the IT Infrastructure Library. This can also help prepare you to use RAPID as an assessment or auditing tool that works in conjunction with existing processes to measure and improve IT value.

RAPID and ITSM are related. Whereas ITSM is a map of what the IT department should look like after implementation, RAPID is a map of how to get there. It defines the logical "places to look" for the factors that affect the scope of an IT decision. It also defines the order of major activities during the IT life cycle. (See Figure 3.6)

For instance, you can use the RAPID model as a communications tool for working with customers to clarify project scope, deliverables, or even risks. Because RAPID focuses on key decision-making factors, it is valid for a wide variety of high-level IT planning tasks. When you are determining project scope, for instance, RAPID works as a "fill in the blanks" chart for you and your customers to share when discussing areas where IT can provide tangible and valuable assistance.

Readiness: How do you align IT services with business needs?

To help our customers quickly achieve readiness for technology decisions, Hewlett-Packard must be able to articulate—clearly, quickly, and accurately—the business benefits of a technology decision. The first objective of

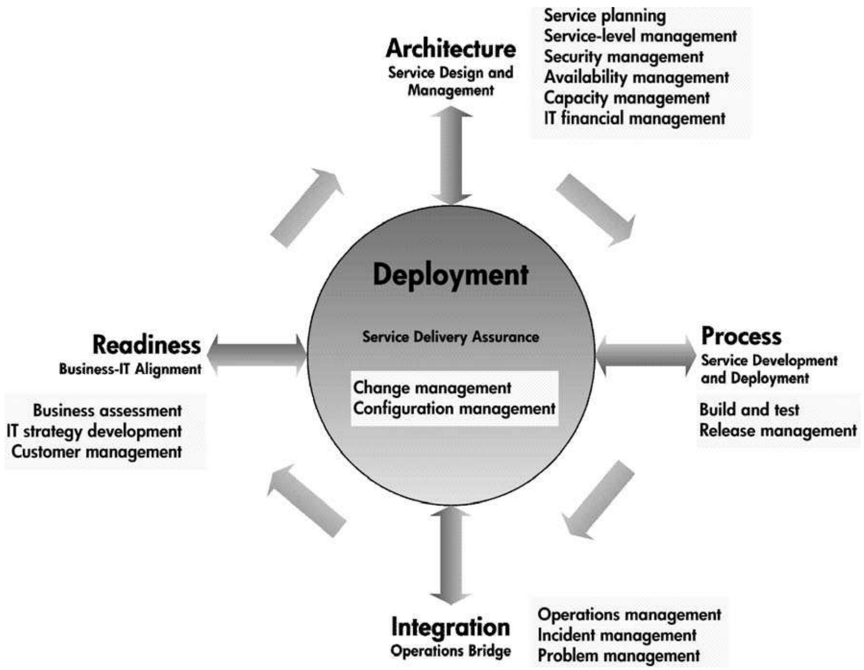
Figure 3.6
RAPID as a communications tool

	R READINESS	A ARCHITECTURE	P PROCESS	I INTEGRATION	D DEPLOYMENT
R RESOURCES	?	?	?	?	?
A ACTIVITIES	?	?	?	?	?
C CONSTRAINTS	?	?	?	?	?
E EXPECTATIONS	?	?	?	?	?

HP, customer loyalty, depends on our ability to develop "products, services, and solutions of the highest quality and the greatest possible value to our customers," to facilitate our shared long-term success. Implicit within this objective is the desire to help our customers make great decisions—especially in difficult times. (See Figure 3.7)

To get ready, you need to be able to explain clearly how a given strategy addresses the components of RAPID: readiness, architecture, process, integration, and deployment. By mapping RAPID elements onto the ITSM model, you will begin to see the focal points that RAPID defines for IT management. Figure 3.3, for example, shows the relationship between RAPID and ITSM—defined in terms of the major activities surrounding the actual production *deployment* or infrastructure. One goal of RAPID is to let you take complex ideas such as ITSM and organize them according to the relevant parts of the IT organization. The RAPID model itself allows you to break the components of a decision or solution into its core components, primarily to help you make sure you understand the relationships involved.

Figure 3.7 *RAPID and IT services management (ITSM)*



In HP, the ITSM department further assigns each component of IT service management to a set of services that align a customer's needs with an ITIL or HP best practice, much like RAPID assigns ITSM to its own components. By creating valuable and distinct services, ITSM allows the customer to understand and direct how best to make IT investments. Because the services consist of technology management components, ITSM uses IT-specific nomenclature to describe each best practice individually.

Readiness in HP ITSM: Business–IT alignment

The upper-left quadrant of ITSM (Figure 3.4) represents business–IT alignment. The group responsible for business–IT alignment deals with the customer, IT value, and IT strategy. To align business and IT needs, Hewlett-Packard ITSM addresses these key areas and related questions:

- *Customer management:* How can IT become a valuable business partner?
- *Business assessment:* From a market perspective, what opportunities are there for IT to add value?
- *IT strategy development:* What is the IT value proposition?

In ITSM, readiness is the parts of the IT organization responsible for aligning the solutions with actual customer needs. Readiness resources are translators that help customers understand IT-speak while assessing business needs.

In RAPID, "R" is for readiness: Being ready to make a decision Readiness is about identifying the common decision-making factors that are important to all stakeholders. To be ready, you must communicate—to find opportunities for common benefit. Readiness is about assessing the problem, developing a common vision, understanding a customer's priorities, and getting ready for work. Readiness also entails helping the customer to justify the investment. In RAPID, three primary activities are associated with readiness:

- Understand the vision and risks
- Prepare appropriate resources
- Gather statistics

It may be useful to agree to some high-level goals for assessing the needs of the customer. You can do this using a RAPID scorecard (like a balanced scorecard) to agree to the first set of RACE criteria, called readiness goals. (See Table 3.1)

Table 3.1
RAPID readiness scorecard example

READINESS	**R**esources	■ Business and IT project sponsor(s) ■ Project manager and technical lead ■ HP Services Consulting and Integration
	Activities	■ Survey and report on customer resources ■ Survey and report on customer needs ■ Develop a strategy for delivery of services ■ Develop and submit a business plan
	Constraints	■ Survey must review current and planned users ■ Business plan must address planning, implementation, and operational costs ■ Security plan must be addressed in needs assessment
	Expectations	■ Focus on updating aging infrastructure services ■ Increase satisfied customers by given percentage ■ Decrease operational costs by given percentage

For each component of a project, the project team must convey the basic *activities* of creating or changing the service—in terms that address the IT elements required to assess the *resources* and *expectations* of this particular customer. Finally, you must identify clearly any assumptions or *constraints*. For one project, readiness activities included the systems assessment, plus an application development framework and a project roadmap. The example scorecards provided in this chapter are not prescriptive guidance; you must find your own answers to each of the RACE components based on the needs of your customer and your IT organization. Chapter 4 presents additional detail describing to define and implement readiness, according to the key components described by ITSM, RAPID, and RACE.

Architecture: How do you design and manage valuable IT services?

Despite the significant economic recession that followed Y2K, the pace of business and technology never really decreased, measured on a transactional level. Each day our customers receive, store, process, and transmit more information and in more ways than the day before. Because each transaction means we have more information available, the requirements of processing are cumulative. Our collective "disk drive" keeps growing, as we store records of the trillions of daily financial exchanges, Internet site visits, electronic communications, and other virtual forms of knowledge. In fact, even the size of the knowledge "atoms" themselves are growing as we begin to digitize new types of information—such as audio, video, and handwriting. Even as the business has become more frugal, it has simultaneously become more demanding.

Architecture in HP ITSM: Service design and management The lower-left quadrant of ITSM (Figure 3.4) represents service design and management. To develop valuable IT services, HP ITSM addresses these key questions:

- *Service planning:* What current and future services fulfill IT's value proposition?
- *Service-level management:* How does IT define and measure service success?
- *Security management:* How can IT secure the infrastructure and data?
- *Availability management:* How will IT provision services for maximum uptime?
- *Capacity management:* How do services manage changes in demand?
- *IT financial management:* How much does it cost to deliver services? What is the budget?

In ITSM, architecture is the parts of the IT organization responsible for helping the customer build IT services that meet the business requirements. Architecture resources are innovators that work with customers to address business needs while simultaneously building good IT solutions.

In RAPID, "A" is for architecture: Building a technically sound solution After building a great business case, it is important to create an effective technical case as well. Architecture describes the technical elements envisioned within the business solution. In RAPID, three primary activities are associated with architecture:

- Identify relationships
- Plan for the architecture
- Measure and prioritize requirements

The scorecard for architecture organizes these requirements according to RACE. One method of adapting RACE to meet the requirements of setting priorities is to arrange the resources, activities, constraints, and expectations according to priority order. You can do this for any of the RAPID scorecards. (See Table 3.2)

Table 3.2 *RAPID architecture scorecard example*

ARCHITECTURE	**R**esources	■ Customer IT architects and management ■ Proof-of-concept Lab ■ HP Services Consulting and Integration
	Activities	■ Map customer needs to existing and planned services ■ Identify design and measure performance of existing solutions ■ Determine the appropriate solutions architecture, including the required technology components
	Constraints	■ Use existing components where possible ■ Justify changes to existing components where necessary
	Expectations	■ Document Business, Functional, Technical, and Implementation views (i.e., Compaq CSAM, HP Global Method Architecture) ■ Create service-level agreements for key services

For IT developers and integrators alike, architecture is an extremely important component of building a solution. HP Services, for example, provides solution architects capable of building an architectural model of the technology systems necessary to support a particular IT service. You must seek out technical thought leaders within your organization to provide architectural guidance. These resources may not be easy to find, however. Nevertheless, a good architect is important to a good solution.

In fact, HP ITSM is architecture for describing the best practices (processes) of IT service management. It uses a visual representation of the relationship between services to group services into architectural elements, also called “IT solutions.” The architecture, in fact, is the solution. It does not stand on its own; rather, it requires good processes and efficient integration to achieve its goals. Chapter 5 presents additional detail describing how to define and implement your technology architecture.

Process: How do you develop and deploy IT services?

For many years, businesses have focused on process improvement as one method of increasing operational (“a better hammer”) efficiency. By developing and deploying improved processes during every IT project, it is possible for technology customers to optimize the potential return of an IT investment. For example, HP Services helps customers build “standard build frameworks” to define the repeatable processes needed to build and deploy servers, workstations, and other computing devices. These frameworks are important: They allow customers to build, rebuild, modify, and optimize service elements within their computing architecture.

It is not enough to “put up” a solution. Too often, IT workers build systems in production without adequate opportunities for testing. These are often the first systems to fail, and worse yet, they can affect the production IT systems long before implementation is complete. For this reason, it is important to have a testing/lab facility available to develop the best practices for deploying a solution. Although building a testing facility may seem a substantial investment, it is necessary. Often the costs can be offset by using scaled-down or aging equipment (although this too can introduce problems), or by using the lab equipment as an inventory available for disaster recovery.

Process in HP ITSM: Service development and deployment The lower-right quadrant of Figure 3.4 represents service development and deployment. To build and deploy quality services, HP ITSM addresses these key questions:

- *Build and test:* How will information technology develop and assemble services?
- *Release management:* How will IT deploy and activate services?

In ITSM, process is the parts of the IT organization responsible for creating and refining IT processes. Process resources are business experts that help IT understand customer-speak while developing efficient and valuable services.

In RAPID, “P” is for process: How the solution works Customers can improve the effectiveness of IT management by organizing the existing service activities according to measurable processes. Building good processes is critical. ITSM, for instance, is an architectural representation of the processes (“best practices”) defined by ITIL. The improvement is measurable in the form of better processes for enforcing architectural standards. In RAPID, three primary activities are associated with process:

- Test the architecture
- Address feedback
- Measure changes (See Table 3.3)

Table 3.3
RAPID process scorecard example

PROCESS	**R**esources	■ End users ■ Customer IT staff and management ■ HP Services ■ Build and Testing lab
	Activities	■ Develop repeatable processes for service build frameworks ■ Develop repeatable processes for service provisioning
	Constraints	■ Existing processes must be optimized first ■ Processes no longer in use must be carefully decommissioned
	Expectations	■ Develop process guides (i.e., HP ITSM, ITIL, Microsoft Operations Framework) ■ Decrease implementation costs by given percentage

Building measurable, repeatable processes is important. You need to be able to find out when a process starts and stops. Creating configuration policies, for instance, lets IT both manage and measure the variety of services deployed to customers. A timer placed at the beginning and end of the "configure a computer" process can measure this effect. As IT becomes more sophisticated, it is important to both update and optimize processes. Chapter 6 presents additional detail describing how to build and manage efficient IT processes.

Integration: How do you change to support IT services?

There sure is a lot of IT stuff out there. When you consider the challenges for HP represented in the integration of the product lines and resources for HP, Compaq, Digital, and Tandem, then you must consider that integration is of significant importance to everyone at HP. Chances are, you may have similar concerns due to your own history of business growth, decline, acquisitions, mergers, divestitures, and other changes. These items all work in concert to increase the importance of integration.

Integration is how you make things work together. Integration is how you map people and technology to processes and services. Integration is how the infrastructure supports each component, and it is how related components communicate. For an IT project, integration is the process that moves a developer's implementation and transition forward to production operational support.

Integration in HP ITSM: Operations bridge The upper-right quadrant of Figure 3.4 represents the ITSM operations bridge. When getting ready for deployment, HP ITSM addresses these key questions:

- *Operations management:* How will IT run and maintain services (day-to-day)?
- *Incident management:* How will IT quickly respond to service outages?
- *Problem management:* How will IT eliminate root causes and prevent future incidents?

In ITSM, integration is the part of the IT organization responsible for meeting day-to-day service requirements while responding quickly to unplanned service issues. Integration resources are translators that help IT and customers effectively share the responsibilities for daily service and support.

In RAPID, "I" is for integration: Where it plugs in Integration is about timelines and infrastructure. Once you have the resources—good architecture and processes—available, it is time to organize them intelligently. Much like the operational management of any business, IT operations is responsible for the day-to-day activities. These activities, in combination with managing service incidents and problems, make up the majority of costs in technology management. In RAPID, three primary activities are associated with integration:

- Integrate systems
- Address issues and concerns
- Measure the impacts

Integration is what happens after IT engineers implement new services. After a service proves its value to customers, it transforms to a group responsible for day-to-day management. This group must be able to perform tasks that occur on a predictable schedule or in response to particular events. There are also technology solutions, such as HP OpenView, that provide tools to help in monitoring service, tracking incidents, and resolving problems. (See Table 3.4)

A solution that does not integrate with the infrastructure is of limited value. You must determine how you will combine existing and new resources to provide the services you define. Without integration, you will find yourself with no means to support a solution other than retaining

Table 3.4
RAPID integration scorecard example

INTEGRATION	**R**esources	■ End users ■ Customer IT operations ■ HP Services customer support and managed services ■ Integration testing lab
	Activities	■ Identify key milestones and schedule for service transition ■ Transition operational processes affected by new or changed services; update the support team ■ Identify trends in incidents and manage problems
	Constraints	■ Existing help desk must be used ■ Changes to existing operational structure must be justified
	Expectations	■ Develop support guidelines and knowledge bases (i.e., ITIL,Microsoft operations framework) ■ Decrease time to recover from failure by given percentage

developers for the long term—an expensive proposition. Chapter 7 presents additional detail describing how you can use integration to optimize your IT strategy.

Deployment: How do you assure continued delivery of quality IT services?

Information technology is not a playground, but unfortunately is often managed like one: Someone stands in the middle as everyone runs off to work on a favorite service, because everyone has something different at which he or she is talented. This "smart people" approach to IT can often fail when the smart people forget to document a change made in response to a recurring problem or feature request. When this happens, we tend to forget who the "smart person" was, and with that, we forget how to deliver the service.

Too often, long after we design an efficient service, someone decides we must be able to manage it as well. For example, ask a customer about the last time someone installed a service pack—with the subsequent reboot—at the beginning of the business day. The result is always a failure in delivering services. To assure quality service delivery, customers and the information technology firm must avoid these failures.

Deployment in HP ITSM: Service delivery assurance The center of ITSM (Figure 3.4) is service delivery assurance. To maintain valuable deployment, HP ITSM addresses these key questions:

- *Change management:* How will IT control production changes?
- *Configuration management:* How will IT track infrastructure components?

In ITSM, deployment is the part of the IT organization responsible for ensuring that IT maintains the systems, processes, and knowledge required to provide reliable IT services. Deployment resources keep IT running in a predictable and valuable fashion.

In RAPID, "D" is for deployment: How it works once it's out there Once deployed, a service must be subject to a very different level of control and management. Because changes to configuration can affect the abilities of a system (availability, capacity) or the processes used to support a system, it is important to manage these changes carefully. In RAPID, three primary activities are associated with deployment:

- Maintain systems and processes
- Enhance and optimize the architecture
- Measure the system value and performance

Implementing this level of change management into a deployed system means you must first clearly understand the architecture, processes, and

Table 3.5 *RAPID deployment scorecard example*

DEPLOYMENT	**R**esources	■ Customer IT and business operations ■ Customer IT staff ■ HP Service
	Activities	■ Implement and update change and configuration management processes ■ Keep change management resources informed ■ Notify stakeholders of service changes
	Constraints	■ Existing support resources must be used to support management tasks; additional resources require a business case
	Expectations	■ Reduce operations costs by given percentage ■ Decrease overall service downtime by given percentage

integration aspects of the system. Tools such as HP OpenView can help supplement and enforce this level of strict change and configuration management. (See Table 3.5)

Deployment is what happens once the solution is out there in the wild. It focuses on how to manage a service or solution once it is actually out there. It is not, however, analogous to the operations roles described for integration. Deployment requires implementation of a long-term strategy for managing the change that is inevitable in any IT solution. Chapter 8 presents additional detail describing what the next generation of IT infrastructure should look like during deployment.

3.3 Conclusion

When you build an information technology solution, remember that it must fit within the constraints of business value. The value of IT services to business is the service we provide: information technology. Much like the accounting department provides a business service by receiving funds and paying employees and vendors, IT's core function is the technology it provides. For technology, like many other business services, the secret to understanding customer value is the knowledge that you would not be around if business had not wanted the service in the first place. What is valuable to the IT department, in many senses, is valuable to the business as well—but we have been so caught up in finding "customer value" that we often forget our own needs are the needs of our customers.

Our customers are demanding IT that is valuable, elegant, operationally effective, well managed, and valuable. Value is the beginning and end of that equation, but you cannot provide IT value without addressing the technical requirements that support it. You must address these requirements in terms of how they affect the business, remembering that the IT department is part of the business as well. In IT, for every solution you must address five essential components (or classifications):

- *Readiness:* What does the business need to do to prepare for the decision?
- *Architecture:* How will the solution's architecture support technology needs?
- *Process:* Where are the opportunities for process improvement?
- *Integration:* How and where will the solution fit?
- *Deployment:* How do you measure success?

The next five chapters of this book explain methods and examples of how to address these and other important IT service-planning questions. In fact, you can classify this entire book as readiness for understanding what business needs from IT. Deployment is what happens later, after you have gone through the process of reinventing IT.

References

1. ITIL (Information Technology Infrastructure Library) (2002), available on the Web at www.itil.co.uk/about_itil/concept.htm and accessed on June 25, 2003.
2. Net Ready: Strategies for success in the E-conomy; Amir Hartman, John Sifonis, John Kadar, New York; January 2000.
3. Reprinted with permission from Paul Johnson, "Sci. Skeptic FAQ: The Frequently Questioned Answers" (1994), available on the Web at www.faqs.org/faqs/skeptic-faq/ and accessed on May 2, 2003.

4

Readiness: Solution Strategy

Action springs not from thought, but from a readiness for responsiblity.
(Dietrich Bonhoffer)

4.1 Introduction

Changing information technology means changes to people, processes, and technology. If the change is significant, such as changes to the structure of the organization itself, you must first make a significant investment in getting ready. In this chapter, you will learn about three key elements of preparing for any major IT change. The section on solution state teaches about how to find the starting line by identifying stakeholders, reviewing current solutions and customer needs, and gathering statistics. The section on solution requirements shows how to begin to understand your current situation by modeling the organization to clarify business needs. Finally, the chapter shows how to develop a solution strategy, to explain the vision, risks, opportunities, threats, and strategy for each of your primary management solutions.

Why is readiness important?

Standards and strategies are useless unless you implement them—and then continue to use them—at every level. It is futile for a CIO to set a "customer satisfaction" strategy in place without making sure all the staff understands what this strategy means for doing their jobs. Success requires you to get ready by first organizing the troops: cascading standards and policies throughout IT. When changing, you must first understand the consequences of the decisions ahead:

- The demands of the Internet age are only beginning. Children born in the Internet age without fear of technology will represent an increasingly difficult group of consumers as new electronic services compete for the traditional consumer dollar. As we increase our reliance on technology, hackers born in the Internet age will have the capability of wreaking havoc on any organization with a technology base. Lawsuits in the Internet age dig up emails and electronic copies of memos that can drag down the largest organizations. In fact, with the Internet age comes the Age of Terror—again making information technology a very attractive target. Now that we have made IT and the Internet such a part of us, we must be responsible enough to use information technology only where it enhances the experience of each organization's customers. Only if you manage according to an accurate model will you be able to understand, and in the long term mitigate, these risks.
- Technology must act less as *a part of* the business; information technology must instead *become the business*. This is a challenging jump for most executives, but is nevertheless required in the Internet age. For this to happen, however, information technology must align itself with very modern yet businesslike roles, extending accountability to the individuals. You must also make the business responsible for IT by enabling your customers to become a part of your decision making. If the technology department wants more investment, it must get incredibly close to the customers. This does not mean, however, that IT should give up its role in technology strategy. The balance between CIO and CEO is very important, because one individual holds a stake in the information-processing strategy and the other is a key stakeholder in the business. The best practice of IT service management is simple: Make the information technology department accountable for the service it provides.
- In the readiness phase, you explore the most important parts of IT and begin to prepare for IT change by building models of your IT solutions to help guide your efforts. It is not enough, however, to build a model. You must implement it. You must link the model to reality, and manage the IT world according to the IT model. Over time, the model becomes reality. It is not easy, it is not cheap, and it most certainly is not always fun. To make matters worse, the requirements are forever changing as business changes—requiring you to build, modify, and rebuild the model frequently. Unless you keep very detailed records of these modifications, you will never be able to maintain the pace. The alternative is to manage IT without standards and models—at your own risk.

To focus your view of IT, you must also understand the ever-growing importance of software to the IT department. Software is the part of computing that teaches the computer equipment to perform processes valuable to its human owners. Some of the most recent research on software indicates that one of the chief challenges in IT is the relationship between architecture and process. In abstracting his thesis on case studies of architectural changes, Josef Nedstam observed,

> *In order to cope with a changing market, software-developing companies need to be able to change not only their products, but also their development processes. . . . These changes do not only have impact on products and technology, but also on the organization and the processes used, a fact that should not be overlooked when performing what is thought to be purely technical changes.*[1]

Stakeholders are sensitive to IT change, yet they are often not aware of or concerned about the issues affecting the IT industry. These issues are core to technology change. We must make significant effort to express IT value in terms of what IT does for (and to) the business and not according to often-artificial or virtual boundaries around technical function or product. The fundamental organization of IT and its stakeholders must not change with every breakthrough product or new generation of technology. This is the most important point of ITSM, ITIL, and any other process-focused IT management architecture: You must empower your existing IT environment to respond to changes in both business and technology without always requiring the overhead of massive projects, expensive programs, and unpredictable effects on stakeholders.

The first three chapters of this book have presented information on "Business Cost of Ownership" (BCO) and the RAPID method for business and management readers; the rest of this book helps prepare you to use these methods. In practice, the issues IT faces are of much less daily concern to a chief operating or financial officer than they are to the IT professional. The decisions made by organizations regarding IT services will affect not only the professional lives of these workers, but their personal values as well. Some people want to use a version of Java, some want a certain platform for C, some solutions need Linux, and some need Windows.

Finding solutions through the barriers that are natural and common to IT requires some preparation; it means making decisions about how IT business is changing: Which technology direction is better or best? Which solutions, vendors, and resources fit the organization? How will the organization pay for the solution? Solution modeling with RAPID requires you to focus on your

IT organization at five very important decision-making levels. At the highest level, solutions, you see the IT organization from the CIO's point of view: as a whole composed of various component parts. At the lowest level, components, you see objects composed of the rules created by any number of solutions, services, processes, and operations. For example, your organization can use RAPID to implement these rules in a software application that helps you focus on the right information to make the right decisions.

How do you focus on the IT challenges of IT investment?

The most important result of this new focus on how processes affect architecture and vice versa is that IT stakeholders must understand not only both topics but also the opportunities and challenges of integrating software architecture and process improvement strategies. It is not just that software-developing companies are subject to the process–architecture relationship, but that all organizations purchasing the software in turn are sensitive to changes in software architecture and process, in a cascading or "trickle-down" fashion. These changes are not only changes to process and technology, but as Nedstam observed, they affect an organization at its core.

Especially during any uncertain economic time, business must invest carefully. This is notably true for IT investments, which have a tendency to grow quickly over time. A business does not often just "buy" a computer—rather, the business buys hardware, software, service contracts, IT staff, and any number of other related technology items. Enter the Internet. Once you connect, the rules regarding IT spending change dramatically. Whether you have a Web site as big as eBay, support Internet customers, perform commercial Internet transactions, or perhaps provide a virtual network connecting employees over the Internet, you must do what you can to support customers and minimize interruptions at any time. It costs a lot to run a business on the Internet, period. To mitigate the potential issues, the IT department must provide a specific and measurable set of services to each customer in return for the annual IT budget. This is important, because it supports our customers' trust and respect.

What is a RAPID model?

A RAPID model lets you take a major portion of the IT services provided by any organization and arrange them into manageable parts—and then manage those parts. You use these models to monitor and manage the performance of the most important things IT does for the organization. RAPID helps you align all the key technology elements in a way that is much more oriented toward business by using the best practices of balanced scorecards and ITSM (information technology services management) integrated with the Internet

Table 4.1 *RAPID modeling elements*

Element	Model	Best Practice/Standard	Purpose
Readiness (Chapter 4)	Solution	*Balanced scorecard*	Aligns individuals with technology and business strategy through accountability and rewards
Architecture (Chapter 5)	Service	ITSM	Aligns technology roles with known best practices for IT business management
Process (Chapter 6)	Process	UML	Documents technical elements of repeatable processes to enforce standards and retain knowledge
Integration (Chapter 7)	Operation	HTML and XML	Organizes, controls, and provides information to stakeholders about performance, roles, development, communication, and change
Deployment (Chapter 8)	Component	(all combined)	Empowers stakeholders to take part in decision making while enforcing efficient and effective change management

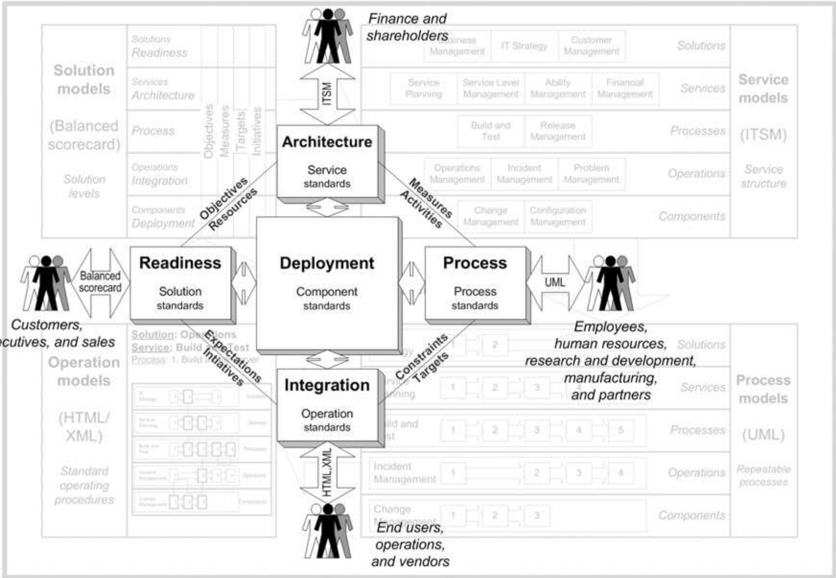
standards HTML, XML (eXtensible Markup Language), and UML (Unified Modeling Language). You align these elements to create the five key elements of a RAPID decision-making model, as shown in Table 4.1. (See Table 4.1)

How will RAPID help?

Using RAPID, you model and organize IT according to object-oriented representations of the rules of your IT organization. At the base is the component. Each object in RAPID inherits from the component; even the largest solutions. Components are the resources of your organization, packaged according to rules set at each level of the organization. The model is both hierarchical, using classes such as describing an automobile with "make", "model", and "year." It also reflects organization layers to allow for clear support for horizontal and cross-functional processes that are required to maintain the computing (and often business in general) infrastructure.

It is not "good enough" to have success in individual information technology projects anymore; it is of highest importance to have every IT component be consistently successful. This requires not only that all the stakeholders share a common definition of success but also that they become capable of differentiating and valuing each solution component according to its unique measurable merits. In a similar sense, RAPID first identifies the problems and then focuses your efforts on the most manageable components of IT likely to provide the greatest returns to customers.

Figure 4.1
RAPID modeling



RAPID is a modeling methodology (See Figure 4.1 and Chapter 10) you can use to audit a technology investment. The component model illustration (See Figure 4.1) is one example of the lengths to which you must go to ensure consistent and positive IT investment returns—your organization must build or rebuild every component according to common standards, with only documented and approved exceptions. Whether the investment is as large as your annual IT budget, or as small as implementing a single server, you must at least address every component discussed in RAPID. It helps you apply some software skills to organize and manage with ITSM or any other methodology based on service management. To actually benefit, you must manage IT in a consistent manner. Put simply—assuming you can measure what the change looks like each moment while you change IT—you can plug all your technology and management together and view the composite performance of every part of IT.

RAPID is not, however, a prescriptive methodology for IT management; it is only a descriptive method for identifying, organizing, measuring, and analyzing the most important measurable information locked within your IT investments. If you want RAPID to help you successfully manage IT solutions, you must combine the framework provided by RAPID with the reality of your own unique IT solutions, as they exist today. In this chapter, you will learn about

- Describing and classifying your current IT solutions
- Understanding the requirements of any change to an existing IT solution
- Building a strategy for IT solutions management
- How to serve your customer using RAPID models

4.2 Solution state

One of the most important parts of providing services in any industry is to understand the similarities between the unique needs of each particular customer. It is a standard part of consulting engagements, for example, to perform analysis and discovery by reviewing the stakeholders, solutions, customer needs, and other current statistics prior to developing a contract for service delivery. This analysis should result in a document describing the current solution state and the situation and conditions currently facing a particular organization. The solution state is very important, because it is impossible to measure changes without knowing where you are starting.

Determining the solution state is often very time consuming, so many service contracts are either very specific or very vague, depending on the maturity of the customers, vendors, and industry. It is very difficult, for example, for most IT customers to understand the technical details involved in a given IT solution. Most end users do not care about Linux and Windows or Java and C#, or even Sun and Microsoft. They just want their computers to do as they are told, when told to do it. Change is fine, but it had better not slow down a worker's day—workers have a job to do, and the impact of impaired job performance is sometimes disastrous.

To determine the solution state, you must identify the following:

- Stakeholders
- Current solutions
- Customer needs
- Statistics

While determining the solution state, record for later use any information that will help you in the remaining parts of achieving readiness, including items related to solution requirements and strategies. Although it may be tempting to discuss these elements, it is counterproductive at this point. Until all the runners agree on the starting line, it is not very useful to start talking about how to win the race.

4.2.1 Identify stakeholders

If you want to redefine information technology, you need to understand what it really means to the current stakeholders—the people affected by change. Unless you carefully manage change, chaos (in the form of downtime and other degradations of technology support levels) can ensue. In describing IT changes, this is especially true for all the human stakeholders: the IT customer, IT business, IT staff, IT support, and IT management. To mitigate the risk of omitting stakeholders, you must learn to identify stakeholders and communicate effectively. To minimize the impact of change, you must learn to communicate with each of them effectively. Communication is implicit and imperative, and all communications must be available to the right stakeholders at the right times.

Identifying stakeholders becomes much easier when you communicate often. Ensuring effective communications may require some unconventional methods: advertising on the corporate home page, meeting with business managers individually and as groups, and discussing the impacts of change decisions with IT groups. The group managing a particular IT change decision must be responsible for making sure the correct stakeholders are involved in any decision-making process. The alternative is to make IT decisions without involving the stakeholders, without communicating, and with poor marks for customer satisfaction.

Are people your only stakeholders?

People are not the only stakeholders, but the rest of the stakeholders must come second. Whether your goal is consolidation and reengineering of IT, or simply upgrading a desktop operating system, there are usually stakeholders in the form of existing and planned technology solutions affected by the change. If you add a server to your network, you also must consider many factors: software, implementation, support contracts, and even hard-to-predict factors such as additional network traffic and server scalability.

In contrast, the IT organization often hinders this kind of horizontal analysis: There are generally "walls" (in the form of management hierarchy, resource ownership, and even vendor loyalty) separating the server team from the network team from the desktop team, and so on. These walls can prevent effective communications by discouraging an implementation team from identifying and working with owners of related solutions. Although many other candidates may claim the title "stakeholder," the people are the only class that can communicate their stake. To simplify the class of stakeholders, focus on the human stakeholders.

How can you organize stakeholders and track their changing interests?

Hewlett-Packard, for example, uses groups in electronic directories (see Chapters 5 and 11) to identify users in many important roles. This electronic version of our organization and our workgroups is essential to both providing IT services and running our business. The groupings allow the organization to secure information for groups and individuals:

- managers to communicate with their reports: and users to find technology and human resources, benefits, and other management
- control notifications to users of an individual server of scheduled changes in service level (such as planned downtime).

Our directories support payroll, benefits, knowledge sharing, information access, horizontal and vertical workflow, and virtually every aspect of commerce between our business units, business partners, and in some cases even our customers.

Directories are an important part of providing IT solutions, because they support integration and consolidation of both architecture and process. For example, a directory-enabled workflow application can identify to whom to send a job in case of a worker's vacation or other absence. A directory can identify the chain of management or the managers to call for support of a particular solution. In fact, many IT applications use some sort of directory—but few share a common directory. Common or "enterprise" directories are important, because you can use them to consolidate business-processing rules and changes to information such as stakeholder names, addresses, and telephone numbers. Without a common directory, these data may be scattered in multiple databases—and even in different information update processes for the same function. In Chapters 5 and 11, you can learn more about how to use electronic directories to organize and locate information.

4.2.2 Review current solutions

You must also review your other resources, including the hardware, software, networks, and other components that make up IT solutions. Each project may require a different subset of this information: for instance, you would need a very different set of information to plan a desktop migration from the set you would need for a new line-of-business application. Before moving forward, you need to understand how changing current solutions will affect your customers. To measure the effects of change, you must be able to measure the current solutions.

But measuring current solutions is not always easy; one HP consulting customer started a desktop upgrade project with data from three different systems—each reporting a different view of their 800 to 1300 installed applications. Because the customer needed to test and prepare each application for redeployment, it needed an accurate list of applications to package. There were three choices: Two of the lists were available from the asset management department, and one was available from Microsoft's Systems Management Server. Each of the lists, unfortunately, referred to products using different names and often without version numbers. This chaos is a common challenge for IT projects: identifying reality although multiple databases disagree on the same types of information.

The project team found reality not within the tools but in face-to-face discussions with many different classes of stakeholders: IT managers, business managers, end users, and user support specialists who supported specific business areas. What they found was that although the vast majority of users needed similar applications within a department, the individual applications servicing each individual user were also customized. Some applications that were installed on user workstations, the users never used; and for some applications the company had no purchase record. For some applications, no user or IT manager claimed ownership. Despite corporate and IT policy, business areas had installed software without the help of the IT department. This project team needed to review each department and user individually.

4.2.3 Identify customer needs

Customers define their relationship with any organization on products and services. Whether you are a tax collector or a customer service representative, people define your relationship to an organization by describing what and whom you know. The individual people who sell, manage, develop, support, and control the products and services you purchase define your experience with any organization. Nevertheless, customers do not care about organization or hierarchy, they just want to

- Purchase a product
- Respect the price
- Find the product useful

- Know they have support
- Have a quality experience

This set of expectations is the same for widget purchasers as for purchasers of IT. People do not buy mobile phones and pocket computers simply because the devices *look* good—rather, they want to know a product and the services behind it *are* good. The only way to find out whether your product is good and how to improve products and services is to ask your actual customers—frequently! To do so, you must identify how you will communicate with each class of your customers. Your customers may include executives, managers, developers, end users, and many other groups. If you service the Internet, often your organization's customers become yours as well.

You can engage the IT customer in many ways: Web surveys, personal interviews, and physical or virtual "town hall" discussion forums. You must engage your customers when making big decisions, because often they can help you create both common and innovative solutions while still meeting their individual needs. One of the most dangerous ways to manage a large IT project is to wait until the solution is finished to communicate with customers. This is because customers are quirky: Not everyone wants the same thing. It is not sufficient to create IT service agreements with the business as a whole; you must manage customer needs and expectations at the individual business unit level. Not every part of an organization requires the same IT components. A desktop using Linux may not make sense in the marketing department, whereas it may make perfect sense for data entry clerks using only a Web browser. This kind of analysis requires a very specific model for creating and managing the individual service levels required by individual business units.

In contrast, finding customers for IT services requires you to expand your definition to include new groups of customers. In an IT services model you must be very specific about where you allocate costs, so IT services can also exchange costs. You need to be able to record these itemized exchanges for use in IT financial management. Chapter 5 discusses the financial documentation requirements of IT services in more detail. For now, simply remember that an IT Service Level Agreements financial model has two cardinal financial rules:

- Everyone is a potential customer.
- Customers always want to know *specifically* what they are buying.

When delivering services, remember that everyone can be a customer. Customers include the CEO, CIO, programmer, systems integrator, vendor, and end user. Everyone can purchase services; but the only way to make this work is that everyone must also have customers. In an IT services model, every manager (and even some individuals) becomes a potential buyer of the services you provide. You can easily expand this model to business in general. At HP, my boss (or more accurately, my unit's cost center) pays for my home cable modem, part of my car lease, my paycheck, and even my supply of business cards. HP gives each manager a budget, and the manager spends this budget as a consumer, using internal transfers, corporate cards, our Intranet, and the Internet. In fact, I also have a personal stake in this. I pay a personal use charge on the car, I pay for my benefits (although the company kicks in quite a bit), and these decisions, although often detailed and requiring much thought, are mine to make—as a customer.

Table 4.2 *Model statistics*

RAPID Element	Solution Model Element	Statistics
Readiness	Solutions	Number of stakeholders Percentage of needs addressed by current solutions Percentage of critical risks evaluated or resolved Annual IT budget
Architecture	Services	Percentage of requirements met by proposed design Number of services defined Percentage of existing services to be reused Annual IT service-level (program) budgets
Process	Processes	Number of new processes created Percentage of new/existing processes Percentage process improvement Annual IT process-level (project) budgets
Integration	Operations	Number of incidents reported Number of problems pending resolution Percentage of resources available for tasks Percentage of satisfied customers Annual IT operations budget
Deployment	Components	Number of changes pending Percentage of changes without incident Percentage of systems outside specifications Annual IT component budgets

4.2.4 Gather statistics

If you want to measure the long-term effects of change on your customers, you must have reliable metrics with which to measure. Metrics in IT are common, but often relate to a single-vendor platform or an individual project scope. To identify valuable or profitable IT services, it is crucial to develop a set of metrics that provide both continuity and flexibility in measuring IT change cycles. Base statistics on real, measurable components of IT.

Auditing IT with RAPID requires you to choose certain statistics for each measurable component. These statistics, however, vary between organizations because of differences in IT architecture and process.

As with all the RAPID models, these examples are only suggestions for where to start. Each organization and its IT department will have a unique set of concerns on which you must agree. Once you and your stakeholders have agreed to some of the more important statistics, you must actually describe how the solutions will fit together according to these rules.

4.3 Solution requirements

Solution architecture is critical to an IT project, because architecture defines both the requirements and the standard structural model of any IT system. In RAPID, an IT system is the IT technology view of an IT component. In the same sense, the business requires you to define *budgets* that serve as the IT business view of a RAPID IT component. RAPID helps you use models to relate your IT systems to each of the business components you define. For example, although ITSM gives advice on how to manage IT services, it does not actually explain how an IT services organization must define individual technology services (such as directory services or email services). To address this need, RAPID uses a simple architectural description to define the relationship between the business requirements and technology structure of an IT solution (See Figure 4.2).

To determine the solution requirements, you need to

- Understand the situation
- Model the organization
- Clarify business needs
- Prepare appropriate resources

You must build models that integrate both the business and technology requirements of each IT solution by matching the architecture with the resources needed to build it, and prepare those resources for building the solution. The relationship (and subsequent constraints) between technology and business views is not always as clear as you might like. Individual IT systems often service more than one business budget, and often the reverse, as well. To prevent confusion, RAPID requires you to place constraints, called *solution requirements*, on these relationships. Each of these solution requirements is important, because they define the constraints used in a solution model. To describe the best architectural model for your organization, identify and address all the known business and technical constraints.

4.3.1 Understand the situation

To understand a situation requires you to integrate your knowledge of the current state of a solution with the requirements of the new solution. Project managers call this *gap analysis*. The gap you analyze is the distance between a current solution and a new or changed solution. Gap analysis is a challenging task, because it requires knowledge of many different types of IT components. A growing role within IT services organizations is that of the IT architect. Architects are generalists who understand the relationships between large IT components. The thesaurus explains this well; it ranks architect with designers, engineers, and inventors. The architect is responsible for working with customers to describe technical solutions that meet or exceed business requirements for value. Because of their expertise in IT solutions, information technology architects provide a significant benefit to organizations while defining requirements. Architects help identify and classify related parts in order to build a consistent and flexible model for a given set of IT solutions. This is no small task: The model they develop must clearly define both the current situation and the proposed requirements in a manner that will motivate stakeholders to embrace each upcoming solution.

To complicate matters, the situation may be different for each group of stakeholders. For example, the end user's expectations often differ from those of the help desk. Each group's expectations are important, because stakeholders prefer solutions meeting their unique needs. In most projects, the different needs of stakeholders may require some customers to accept additional work, resources, downtime, or other costs. To understand a given situation fully, identify, record, and analyze the difference (or gap) between the current situation and the planned solution for each stakeholder.

There is an important difference between previous IT management disciplines and those required when providing up-to-date IT services. To deliver services most efficiently, you cannot make service agreements only at the organization level; rather, you agree to service-level standards for each customer/business unit. Base these standards on a specific model of measurements that are valuable to your customers. Once the model is developed, it becomes the yardstick for your customer's contract; the organization must continue to control and record changes to the model. Each change has an associated cost when supported by IT service-level agreements (for more on SLAs, see the section on service-level management in Chapter 5). It is not easy to develop an SLA, which is, in effect, a contract; an agreement between you and your customers to provide IT services according to the specified standards.

4.3.2 Model the organization

IT services is a very new model for most IT departments. Providing measurable services entails keeping and publishing very detailed records of IT performance. It also requires you to work closely with your customers—often much more closely than ever before. Providing measurable services is political, it is personal, and like it or not, it is often very controversial. Changes to the organization itself and its financial models may not all happen at once; due diligence is required to avoid making these models too complex or difficult to manage.

One of the most important parts of getting ready is to build a model of the relationships between the organization and IT itself. Develop a *consistent* model for structuring the organization, and continually measure changes affecting that model. Base the model on the needs of both business and technology. For organizations unaccustomed to formalized ITSM, building and implementing a model will help prepare you for the challenges ahead. RAPID solution modeling lets you start with a virtual model for services. By using a model to measure the effects of smaller (but real) changes, you can learn how to scale your solutions to include all of IT.

Both the structure and function of these relationships are extremely important. During the Internet boom, software developers widely started using a visual language called Unified Modeling Language (UML) to define the structure and hierarchy of software systems. Eleven companies, including HP, 3Com, Sun, American Airlines, Canon, and Unisys, founded the Object Management Group (OMG) in April 1989 "to create a component-based software marketplace by hastening the introduction of standardized object software . . . industry guidelines and detailed object management specifications to provide a common framework for application development."[2] UML lets you build models—pictures of current and future states, and lets you develop standard diagrams of architecture and process, as depicted in Figure 4.2. You can use UML to show the relationships between components in an IT services model.

The smallest changes in RAPID occur in objects called *components*. One important part of implementing ITIL best practices is to define configuration items (CIs). CIs are the individual measurable parts of ITIL, such as a configured server or a software application version. RAPID classifies CIs as *components*. Components are not simply IT assets; they are complex technology objects that support ownership, relationship, properties, and even state (past, present, future). In RAPID, you arrange components according to a standard model for presenting the structure and relationships between IT objects, called a *schema*. By developing a common schema, you can classify the operational properties of components (the individual working units) of information and communications technology into successively larger groups:

- Components into operations
- Operations into processes

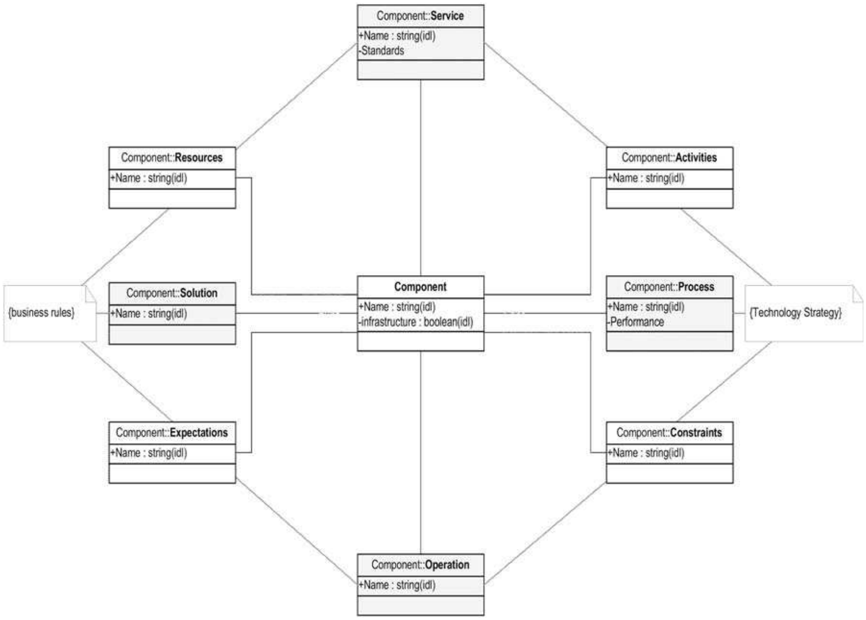


Figure 4.2 *Unified Modeling Language (UML) view of RAPID models*

- Processes into services
- Services into solutions

4.3.3 Clarify business needs

The IT business must be able to act as a valuable partner to each of its stakeholders. As additional services become available, the needs of each stakeholder may be unique. Stakeholder needs change often over time, so the model must be flexible to accommodate these changes. To minimize impacts across services, each IT service should operate as a unique part of an IT business. In a world of infinite choices, the IT business must demand that each service be profitable of its own accord. Inherent in this demand is the requirement that IT SLAs (service-level agreements) must account for a detailed set of options negotiated between IT provider and customer. The sum of the selected options allows the organization to anticipate the resources required to provide each valuable IT service.

How will IT service management affect your business?

IT service management is a very different model for IT, and depending on how you manage IT today, it may not be possible to implement all this change at once. The key of understanding ITSM and other IT best practices are that they are descriptive, not prescriptive. They empower the organization to invent a specific and measurable, yet individualized, model for effective IT management. You must understand the distinction—IT best practices, such as ITIL, leave the implementation of their standards to the organization. Your responsibility is to create realistic goals and actually implement the best practices. Expect results only when you fully implement one or a few practices. The results should be worth the risk: Significant IT cost reductions and measurable efficiency improvements. No matter who you are in IT, consider measurability important. It is your inevitable future—one way or another, your organization will measure you.

To observe the benefits of implementing best practices, therefore, you must actually measure them. Because ITSM is descriptive rather than prescriptive, you need to determine how to perform these measurements. ITSM demands you perform these measurements perpetually, in the form of itemized bills for individual services. To provide services, you must actually bill customers for each individual service. Some organizations may find this thought almost threatening. If you could detail how you manage and bill customers for IT solutions today, what would it look like? You could sell packages containing (1) all the servers, (2) all the software, and (3) all the support. You could also sell packages of mainframes, minis, and micros. The combinations vary widely between organizations.

These models, however, are neither specific nor measurable from a customer point of view. To address this need, IT services arranges services around very businesslike IT processes for ordering technology, paying vendors, finding new ways to apply products and services, supporting customers, and managing change. You must explain your bills in terms that both IT and the customer can understand. Doing so requires patience, cooperation, and resources. Remember that in IT services your business and your customers can be very picky; your competitors can attract their business in a mouse click. You must be able not only to compete but also to win, by being the best at helping your customers compete.

4.3.4 Prepare appropriate resources

An IT service does not operate without resources. IT resources include many different kinds of measurable assets. Some of the assets are human, some are hardware, some are software, and some are money (usually in the form of an annual IT budget). Each resource has an associated cost, and the services purchase resources at these costs. In IT, resources (money) for major projects are most often set at the top and negotiated throughout the management chain. IT services treat resources as a decision between customers and service providers; except in the case of infrastructure.

Resources are everywhere in IT. Hardware is the easiest to find; you can trip over it. Hardware is the physical components of IT, such as PCs, servers, and mainframes. Software is harder to find. Software includes the procedural components of IT, the computer codes for the embedded processes that make IT useful. The human assets of IT are the most important, because they are the most expensive. Many technology companies manage the human resources separately from the technology services they provide. This is because a single person

- May fulfill multiple roles in a given organization
- May leave the organization but should not leave the model because if the person leaves the organization, their responsibilities (role) must still be reflected in the service model to ensure that the job gets done
- Has a predictable cost per hour

One more resource (that organizations commonly undervalue) is documentation. Documents are one of the most important components of IT services; they are the knowledge assets that help you understand, support, and extend your current services. Before developing a strategy for solutions, first verify that you have the appropriate level of documentation resources on your current state:

- Stakeholder groups and decision makers
- Current solutions and solution owners
- Identified customer and business needs
- Statistics on current services

Allocate each resource to a single component, and allow for transfer of service resources (and cost) between both internal and external customers. Each service is responsible for completing an accurate financial strategy based on common standards for billing customers. Remember, your financial strategy must enable you to provide detailed service billing for each of your customers, expressed as a cost per unit of use. The cost should be relevant to the service, such as dollars per month. This level of detail is often difficult to add up, so for now, focus on finding the resources. Chapters 5 and 9 and explains IT resource and financial management in more detail. After identifying and preparing the resources, you can begin to develop a strategy for delivering IT solutions.

4.4 Solution strategy

It is important to negotiate, refine, and communicate a strategy to align the stakeholders toward common goals. Align these goals with each stakeholder's vision of the solutions, and the risks of implementing each particular solution. To build a solution, first understand your individual technology opportunities and the corresponding threats. It is important to build strategies on consensus; to make decisions by consensus, develop a strategy addressing each viewpoint of vision, risks, opportunities, and threats.

Technology vision must be flexible enough to account for inevitable change in IT products and services, while consistent enough to learn from the experiences of change. Vision is more important than just where you think you want to go; it is a view of what the organization will look like in the future. Vision is the CEO and CIO announcing the role of the Internet in an organization; vision is the IT directors choosing vendors and platforms.

You must communicate the vision widely, and it must be clear and to the point. Also make sure to address the different views of vision appropriate to each stakeholder. The key to setting a vision for technology management is that you must not stop with envisioning—your team must also implement your decisions.

There is no point in "beginning" to implement ITIL, ITSM, MOF (Microsoft Operations Framework), Web services, or any other Internet style of technology management. You must commit to and complete the tasks of both implementing a new methodology and investing in upkeep over the life of each solution. The vision must address the lifetime risks common to different views of vision, while providing a risk mitigation strategy that addresses the different impacts risk can have on unique stakeholders. Each stakeholder must share some common vision to achieve consensus. Achieving consensus usually requires you to break through many barriers. The benefit is worth the risk: IT accountability and customer satisfaction. You will know if you succeed, because you will have evidence to prove it. Enact your vision in concrete.

4.4.1 Understand the vision and risks

If you have no experience in service management, no documentation on current solutions, no identified stakeholders or subject matter experts, suspicious customers, and/or a nervous IT staff, then you likely have a much larger list of risks to compile than an organization that does not (very rare). It is important to encourage customers and other stakeholders to participate in risk analysis, especially if the customer will be paying to mitigate each *risk* (see Chapter 9 for more on risk). Involve your customers and other stakeholders in the IT decision-making process. The only way for the participation to be measurable is to measure it. Once you become this detailed, however, remember that every single customer's detailed opinion of you is important. You cannot just go to your largest stakeholders; you must involve the smallest ones as well. The only way to make customers truly happy is to give them the ability to guide you through the ever-changing maze of their needs.

One vision for service management in IT is happy customers (customer satisfaction). This is only one of many measurable traits, along with scalability, availability, performance, cost efficiency, and a variety of other abilities (ITYs) that may or may not be important to your organization and its customers. The important part to remember is that not everyone agrees on what is or is not important. You must remember the 80/20 rule. In one hundred years of Pareto's principle, we have rediscovered its truth in many aspects of life: that 20% of your efforts will provide 80% of your results, and 20% of your results will absorb 80% of your resources. This rule tells us that some parts, the vital few, are the customers who pay the lion's share of the bill. It also tells us that the majority, the trivial many, are the biggest opportunities but the hardest to satisfy. In fact, the 80/20 rule is auspicious as it even applies to nature itself: It describes the gravitational relationship among the Earth, the Sun, and the Moon. Within each split of 80/20, however, is just another classification waiting for you to discover it. No single ability is most important—except the ability to find new relationships. This ability requires vision, dedication, and most of all, measurement.

How do you build a vision?

HP consultants experience consistent success using design workshops to help their customers build solution strategies. Customers invite key stakeholders to two- to three-day knowledge-sharing sessions, to discuss the most common issues faced while planning a particular solution. These sessions are done in both large-group and focus-group formats, always with someone as facilitator to keep discussion on track. The goal of a design workshop is to discuss the most important success factors from previous projects of similar nature and to use this information to develop and document a conceptual framework for building a new solution—a strategy.

If your solution workshop is the first step to implement ITSM, you are in for a big surprise. You will go through the trouble of identifying the solution state and requirements—you will invite all your stakeholders—and the first thing you should expect is that not everyone will show up. Even if you have a way of forcing them to attend, it takes something radical to get everyone with an opinion to speak up. When cornered, people tend to guard information with the voracity a hungry dog may guard a steak. If you bring a bunch of technology people together to tell them that you are about to start measuring them, many are bound to be very quiet when you ask them how to do it. (I would imagine that the primary reason SLAs are so useful is that they force you to discuss your solutions and services with every stakeholder—to understand and address each set of concerns. It is there you must start, by finding where stakeholders draw their own lines in the sand.) Start with something simple, like a list of your most important principles. Do not forget to agree on and document what you mean by them.

Or you could try modifying the list from HP's IT department:

Principle	Meaning
Common	Be HP's best reference account
Business processes	IT delivers against defined business processes
Information	Data are managed as an asset
Applications	Buy, don't build; deploy "good enough" solutions
Application infrastructure	One is better than more
Core infrastructure	Continuously improve efficiency

Hewlett-Packard IT principles

What are the risks?

The purpose of these sessions is not to dig deep, rather to stimulate discussion of the most important issues and risks affecting the lifetime of a solution. The most successful solutions document is a frequently updated list of potential "gotchas" as well—risks that we must mitigate, and the resulting requirements. The risks of changing your organization to an ITSM model depend on how far away your current state is from your target model. Risks are events—things that happen—that you might have to do something to prevent. Once they have happened, ITSM calls them issues or problems. In RAPID, risks are the *constraints*. Risks cause you to constrain your view of an event because you do not really want it to happen. There are many ways to classify risks to the IT department; the important thing is that you use the same classifications as you do for the IT department itself. You classify risks according to the solution, service, process, operation, or component (collectively, "Component") a particular risk most affects. Chapter 7 provides more information on the day-to-day management of risks and other continuity issues by integrating this task with the overall management of IT operations.

4.4.2 Review technology opportunities and threats

Inherent to business are both opportunities and threats. Each opportunity is a decision that you expect to produce a positive result. Each threat is a decision that may produce a negative result. Both opportunities and threats can come from sources both within and outside of a given organization or its IT department. Opportunities and threats not only should include those you intend to address, but should also present opposing views and other stakeholder comments.

Start by looking at your current technology, especially the parts the user sees. Ask questions about current IT solutions of the users, the business, and other stakeholders:

- What would you do to make it work better?
- Are there specific parts that need the most improvement?
- How does it affect you?
- How can we make it easier to use?
- How can we reduce the impact of change?

If you talk to many different groups, you will find that every group of stakeholders will give you different answers. Although some elements may be important to many groups (less downtime, more training) others will be more specific. The manager of the finance department, for example, may focus on the needs of managing information assets. Ask a security guard to count how many people bring laptops to work, ask technicians what breaks most often, ask developers where their skills need improvement—think outside the box. Not only ask your customers questions, but also encourage them to ask questions of each other. By meeting with representatives of many different stakeholder groups together, you may find that in the mêlée of voices is a common song—some things will likely rise to the top as opportunities and threats affecting everyone alike. Record these lists, and make them available to stakeholders for comment.

4.4.3 Develop a strategy for solutions

Base your strategy on the consensus of stakeholders, if possible. If you can't find consensus, maybe you can identify the key points of contention between stakeholders. Some solutions require you to address an individual need in multiple ways to address severe disagreements or incompatibilities. The strategy must address multiple viewpoints that correspond with those of the decision makers. If you cannot support a given strategy with consensus, it may be necessary to use executive order or expand communications to clarify further the vision, risks, opportunities, and threats.

An IT solution is a budget itself, owned by an IT solution owner. The solution owners may be an individual or a group of individuals. This is an important difference from typical people hierarchy. In HP, for example, each manager has many roles and many roles have two managers. These management "buddy" teams have important horizontal as well as vertical functions. There is only one way to reflect these types of complicated relationships: You must have an accurate model. A RAPID solution model is an electronic representation of the hierarchy and components in a particular IT budget.

How will you build your strategy?

Many of the best practices for IT management are observed in the HP IT department. Considering some of the challenges this group overcame during the HP/Compaq merger, it is truly an amazing group. Directories, networks, Web sites, databases, systems, devices, and other technology integrated during the mergers of Digital, Tandem, Compaq, and HP (plus many other smaller acquisitions) represents a scope rarely seen in any IT department. Part of the success in IT integration at HP is due to the enterprise architecture. (See Figure 4.3)

HP uses a standard enterprise architecture based on a set of guiding principles. These principles underpin the architecture to aid technology decisions. The principles direct HP to where it wants to be. The enterprise architecture group keeps the number of principles to a minimum. (See Table 4.3)

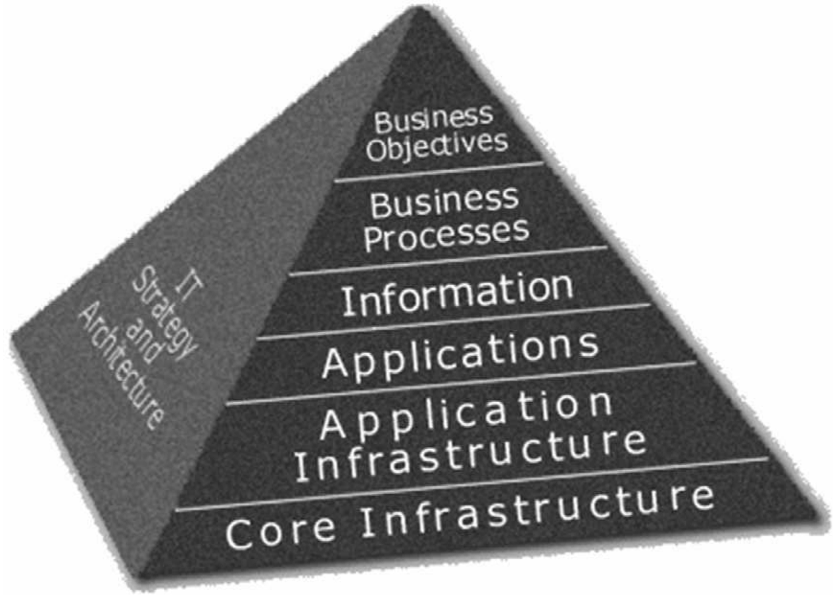


Figure 4.3
HP enterprise architecture

Table 4.3 *Architectural principles (Source: HP)*

Focus	Principles
Common	Use HP products and leverage our strategic alliance partners
Be HP's best reference account	Reuse solutions and components
	Align with industry standards
Business Processes	Core business processes will be maintained end-to end as a company-wide asset and will define value delivery chains around a total customer experience
IT delivers against defined business processes	As-is and to-be processes will be defined prior to any project delivery IT investments
	A single process will be used where business outcomes are essentially the same
	Business process taxonomy will drive organization of architectural objects
Information	Data are a corporate asset and have a definitive source
Data are managed as an asset	Enterprise data models are based on those purchased with core applications
	Enterprise standard data definitions are used and reused by all applications
	Business process lead(s) own subject area metadata definitions
	Data quality is owned by the business, measured and reported by the IT department
Applications	Commercial solutions from a few major software suppliers will be broadly used to increase capability and we will partner to deploy solutions that maximize competitive advantage
Buy, don't build; Deploy "good enough" solutions	Target 80% solutions to improve time to delivery and at a lower cost
	Applications will expose process-level statistical metrics
Application Infrastructure	Deploy global solutions using minimal images
	Unless business process dictates otherwise; use interapplication interfaces that tend toward real-time messaging technology
One is better than more	
Core Infrastructure	Drive simplification to reduce complexity
	Manage capacity across the IT infrastructure worldwide
Continuously improve efficiency	Deploy scalable, high-performance systems capable of supporting volatile workloads

The HP architecture management team reviews the principles, and the CIO approves them. This process allows HP IT to focus on the critical objectives—the ones most important to the business. HP's strategy, at the most basic level, is the principles on which HP focuses its enterprise architecture. Because HP is primarily a technology company, these principles drive not only organization of the HP IT department, but also the entire organization in turn.

4.5 Conclusion

There is a limit, however, to how much you should invest in getting ready and deciding to move forward. An important part of getting ready is to build a plan. In IT, we call this plan architecture.

References

1. J. Nedstam (2002), *Changes to Processes and Architectures; Suggested, Implemented and Analyzed from a Project Viewpoint* (Lund, Sweden: Lund Institute of Technology, Department of Communication Systems, Lund University), p. 5.

2. Object Management Group (2002, OMG "Background Information"), available on the Web at www.omg.org/news/about/ and accessed on May 10, 2003.

5

Architecture: Foundation Services

Noble life demands a noble architecture for noble uses. . . . Lack of culture means what it has always meant: ignoble civilization and therefore imminent downfall. (Frank Lloyd Wright)

5.1 Introduction

Although one might like to think information technology departments are experts at architecture, this is most often not the case. Too often, the architecture of organizations reflects technology that changes cyclically with the IT industry. If you base your technology investments on vendors' platforms, operating systems, applications, or any other technology component, you must be extremely cautious. Technology-based architecture for IT business management can lead to an untenable situation: Unlike the architecture of a skyscraper, technology architecture changes often to meet changing technology needs. In reality, you can structure the IT services along more business-focused lines, to minimize later changes to architecture. In this chapter, the section on service planning discusses how to design and manage IT systems according to a standard architectural structure. The discussion of readiness defined the layers within IT; here, the section on service design can help you organize the services within each layer in an architectural manner, to reflect the relationships between key services. In service-level management, you use services to reflect the priorities of your stakeholders by developing service directories and service-level agreements (SLAs).

Why is architecture important to IT?

Bob Napier, HP's CIO, made a number of comments about the value of enterprise architecture (EA) during an October 29, 2002, coffee talk. As EA is one of his more passionate issues, his comments provide a great look into our own strategies for EA management:

> *We're going to publish an enterprise architecture, and I would expect that everybody as an IT professional and part of this organization will . . . look at the enterprise architecture as defined. . . . Do you ever wonder why—and this has been going for as long as I know, long before I was born—is that people start construction projects. . . . They say, okay, I'm going to build this building and I'm going to invest $20 million dollars. And somehow it gets built, usually on time and on budget—sometimes not, when it's a new technology like the Golden Gate Bridge, which was a breakthrough in engineering. But by and large, your average, rectangular, eight-floor office building comes in on time and on budget. Why? Because in enterprise architecture everything is built from standards. There's a standard that says, every six feet on a wall, there's got to be an electrical outlet. There's a standard for everything. . . . There's architecture all the way from an artist's rendering of what this thing is going to look like sitting on that pile of dirt, all the way through to exactly where the beams are going to go.*
>
> *There's a very interesting, scientific engineering discipline of how you get from one end to the other. . . . Then you look at our [IT] industry . . . and you say, why is the percentage of systems projects have such a high level of failure? . . . Because we just start down this path building whatever this solution is going to build, and we don't necessarily start with a blueprint. We don't start with architecture. We don't have all the disciplined pieces around it . . . we just go. So would you call that a science or an art form? I'd call it an art form. . . . It may turn out pretty but it's not predictable. When you're investing the hundreds of millions of dollars on things we're investing in, and the number for the first half is pushing in excess of $300 million dollars worth of investment—when you're investing that kind of money, wouldn't you personally like it, if it was your money, to have some level of predictability to the outcome?*

Architecture is the study of how we structure the things we build. Every product has architecture; it is all the features that differentiate one product from the next. When you build a house or a skyscraper, you use an architect—a person who knows the practice of architecture—to help you envision what the house will look like. Given the constraints of finances, physics, and building codes, and the expectations of purchasers and occupants, the architects help us model the best structure with the resources we can afford for a given investment. When you build a number of structures and relate them to each other, this too is architecture. Whether dealing with the neighborhood of buildings, or the IT supporting a particular solution, you must be concerned with how the parts fit together to build a whole. To manage IT services, build a model to define the architecture of each IT component.

5.2 Service planning

Choosing from the many potential architectural models for IT services, you must align the right model with the values of the organization. In this section, I discuss some of the challenges in planning IT services. The architectural model of these services is very important, because it defines the structure and relationship of financial transactions within the information technology department. This is a crucial point in every service management model: providing for the possibility of individually managed and contracted IT services. One of the most important common factors of all service management models is effective management of SLAs. These service-level agreements often differ greatly according to the needs of the individual organization.

Have you planned SMART objectives?

Before you can build and manage an SLA, you must first build manageable services. There may be many different SLAs between IT and customers, the organization, vendors, and even users. Base each SLA on common objectives agreed to by the stakeholders. Each service must be unique, composed of parts owned by individual decision makers. These people negotiate SLAs among IT, customers, and other stakeholders. You can summarize the abilities of a particular service by defining service-level objectives (SLOs). To let organizations allocate appropriate budgets to individual services, the objectives must be SMART—specific, measurable, achievable, realistic, and time-constrained, as shown in Table 5.1.

Designate a team responsible for service planning in your organization. In small organizations, this can be a small group of customer representatives and financial and IT planners. In larger organizations, your service-planning group is often a dedicated group of individuals concerned with the long-term success

Table 5.1
SMART objectives

Specific	Define services with an unambiguous (specific) range of special and unique abilities. Make services responsible for a particular set of processes, operations, and components that it sells to IT customers.
Measurable	Build SLAs on standard (measurable) performance metrics and costs. Implement monitors to measure service performance.
Achievable	Define the responsibilities of service owners and supporters. Provide service owners appropriate resources based on previous experience.
Realistic	Services should plan to meet or exceed performance goals. Services must maintain a list of current risks and other issues.
Time-constrained	Review service strategy and metrics every year or two.

of IT services. In either case, your service-planning team must define valuable services, identify service relationships, and clarify service-level objectives.

When planning the services, consider three things:

- Have you defined services that customers will find valuable?
- Can the stakeholders agree to clear objectives and metrics for service levels?
- Do you understand the relationship between services?

5.2.1 Describe valuable services

One of the first parts of planning services is to define the services you intend to deliver. Not everything can be an IT service; you must be selective, because there is a significant investment in providing services. The overhead of providing too many (or too few) services can quickly overcome the potential benefits, so carefully consider the potential value of each individual service. Some services are more valuable than others. The service is the focal point for both architecture and financial management. Because you deliver each service to each customer for a specific cost, you must have a specific method for financial management that enforces your organization's unique description of IT service value. (See Table 5.2)

Your definition of services must be very clear. Remember that each component, including services, is vital to the delivery of other components.

Table 5.2 *Common architectural goals*

Solution/Readiness	Manage long-term costs by enabling calculation of actual TCO/ROI. Identify customer-valued services.
Service/Architecture	Enable TCO/ROI (total cost of ownership/return on investment) calculations. Identify actual service costs to appropriate stakeholders and support cost recovery where it makes sense. Provide financial information to decision makers.
Process	Encourage more efficient use of resources. Define repeatable processes for financial management.
Operation/Integration	Provide incentives for quality services. Provide better financial data on external service value.
Component/Deployment	Assist change management.

Consider developing a list of architectural goals that IT services will provide to your organization. Specify how each service provides value at each level, from solutions to components.

What should you consider when selecting valuable services?

When building a list of potential services, consider these points:

- The important part of selecting valuable services is that each must remain valuable. You make a significant investment in each service. If your SLAs enforce penalties for nonconformance, you may also take on additional risk for service failures. When describing services, it is important to define services in a flexible yet standard manner. All stakeholders must agree on the basic definition (standards) for each service, so be able to focus your definition of what processes, operations, and components you include in a valuable service. To avoid losing focus, remember to take out the trash.
- Your stakeholders will demand that your definition of each service be very detailed, and will often require that you provide services configured to their individual standards. Some CIOs of IT firms have accomplished this by creating “basic” services and then allowing customers to pay for service add-ons. RAPID lets you do this by defining the cost of not only services but of every component. By being this specific, you can focus on the most valuable services—the services your customers agree are worth paying for.

5.2.2 Clarify service objectives

A U.S. state government IT department was building a standard IT service for "industry standard servers." To develop the metrics, this department asked its customers—state agency CIOs and their staffs—to respond to a survey asking them to rate various indicators of their IT objectives.

Customer List of Objectives (servers)

Availability/integrity

- Disaster recovery
- 24 hours, 7 days a week
- Minimum availability requirements
- Change control
- Guaranteed delivery of data between systems

Privacy

- Private industry
- Remote access

Confidentiality/security

- Antivirus protection
- Compliance with state of security policy
- Data security and confidentiality
- Physical security
- Security controls to comply with the Health Insurance Portability Access and Accountability Act (HIPAA)
- Meet federal requirements and compliance with federal regulations
- Compliance with federal regulations
- Compliance with local control requirements and data security

Services and support

- Decentralized IT support
- Maintenance and support contracts
- Network and mainframe printing
- Problem resolution

- Access to data
- Storage
- Directory services
- Client OS Support
- Application distribution
- Dual network management

Costs

- Billing system
- Billing for services rendered
- Maintenance cost
- Solution cost
- Tiered
- Training
- Current system migration

The state posted the results of this survey on the Internet for project stakeholders and other interested parties to solicit feedback and guidance for implementation planners to use when planning IT services. You can start with this list, you can get a list from a vendor (most vendors have standard lists), or you can build your own requirements survey using the RAPID tools. Work with customers to develop the service objectives required for each of your IT services. One of the best ways to determine these objectives is to evaluate the customer needs through electronic surveys and face-to-face interviews. The objectives are simply the list of categorized topics (as just shown) discussing the principles of the IT organization, whereas the requirements are the actual results of the customer survey discussed later in this chapter.

Service-level objectives (SLOs) are the business performance targets you use to deliver individual services. Define metrics for each service, and then use the metrics as performance targets when defining SLAs. Each component in your models will include both shared and individual measures of performance. Operations (solution level) may track the number of daily incidents resolved, whereas the incident management (operation level) group may also track the average time to resolve incidents. Each layer of your model requires you to add detail to your performance metrics. Base each of these metrics on a particular objective or requirement of your customers (see the section on service-level management).

5.2.3 Identify service relationships

Deciding how to organize these components-selecting whether a particular component is a process or just an independent component—requires you to think like a computer programmer. In the 1970s, people began to refer to this handling of components as "modular programming." In a classic article from the Association for Computing Machinery, D. L. Parnas quotes Richard Gauthier and Stephen Pont:

> *A well-defined segmentation of the project effort ensures system modularity. Each task forms a separate, distinct program module. At implementation time each module and its inputs and outputs are well-defined, there is no confusion in the intended interface with other system modules. At checkout time the integrity of the module is tested independently; there are few scheduling problems in synchronizing the completion of several tasks before checkout can begin. Finally, the system is maintained in modular fashion, system errors and deficiencies can be traced to specific system modules, thus limiting the scope of detailed error searching.*[1]

Using this definition, Parnas extended the discussion of modularization to consider the criteria to use when decomposing a software system into modules. By reviewing two different modularizations of a single system, the article could explain why although there are often many ways to accomplish the same programming task, not all these methods produce equal results. One of the major contributions of this work was to move modularization beyond the simple process flowchart. Parnas suggested the following criteria for decomposing software systems:

- *Each module hides some design decision from the rest of the system.*
- *A data structure, its internal linkings, accessing procedures and modifying procedures are part of a single module.*
- *The sequence of instructions necessary to call a given routine and the routine itself are part of the same module.*
- *The formats of control blocks used in queues in operating systems and similar programs must be hidden within a "control block module."*
- *Character codes, alphabetic orderings and similar data should be hidden in a module for greatest flexibility.*[2]

These criteria are still in use today by software developers, but how do they affect IT service designers? When you consider organizing your IT services, you will surely find that many of the same criteria that apply to developing software also apply to the organization of information technology itself. A service, for example, will often hide many of the processes supported and components used. IT services, when implemented as organization-integrating software, are also subject to each of the other Parnas criteria. These criteria, however, are too much like IT—hierarchically structured software based on vertically structured organizations. The demands of the Internet age have made the horizontal (functional) capabilities of software organization as or more important than the vertical ones. Consider why Parnas suggested a model beyond the simple flowchart:

> *It is an outgrowth of all programmer training which teaches us that we should begin with a rough flowchart and move from there to a detailed implementation. The flowchart was a useful abstraction for systems on the order of 5,000–10,000 instructions, but as we move beyond that it does not appear to be sufficient; something additional is needed.*[3]

We should clearly review this statement again to mitigate the challenges of Internet-era systems, when it is common to have millions of instructions each second working in concert—simply by connecting a few PCs together in a network. In fact, it is interesting to note that whereas Parnas advocated a shift from functions to hierarchy, we must shift again now, to focus once again on the functions we were previously trying to "move beyond."

How can you organize your IT components to make service relationships clearer?

Parnas concludes,

> *Since, in most cases, design decisions transcend time of execution, modules will not correspond to steps in the processing. To achieve an efficient implementation we must abandon the assumption that a module is one or more subroutines, and instead allow subroutines and programs to be assembled collections of code from various modules.*[4]

The views and organization of different groups of individuals—the horizontal organization—may also have a stake in IT decisions. A simple organization chart does not explain these relationships (although it's a great place to start). In these "org" charts, any position may be an individual or a group, and individuals and groups may own multiple types of resources. Services are related by the processes, operations, and components they support or use. It is important to identify and record the relationships among all components. (See Figure 5.1)

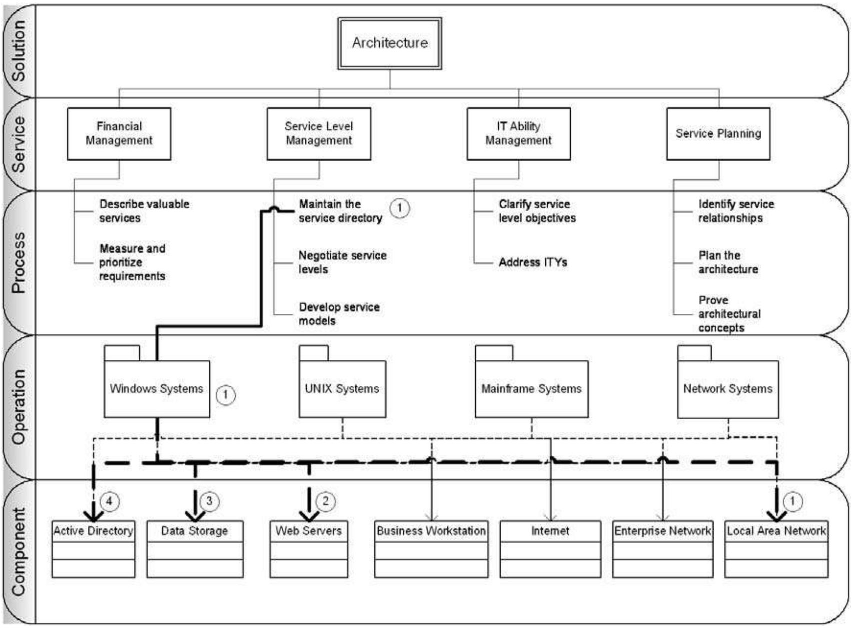


Figure 5.1
Finding relationships between processes and components

The first two relationships you must define are ownership and support. To maintain some sort of business sense out of IT management, you should indeed start with organization charts that define how you manage IT resources. These organization charts should define reporting paths and other relationships not only between IT human resources but also between technology and information resources. One benefit of having a detailed electronic model is that you can view the relationships not only of solutions and services but also of components. When designing IT architecture, determine many of the processes, operations, and components supporting each service.

The important thing is to document exactly how all the components relate through architecture. In building your models, you will find that the structure of each component changes over time as you learn more about these relationships—they may not all be obvious at the start. The benefit of building a model is that as you discover these relationships, you will have a place in which to note them. A component must be able to record its relationships; a single repeatable process (such as maintaining the service directory) may use many different components.

5.3 Service design

The service design defines the architecture, expected abilities, and proven concepts that comprise a given service. The architecture is a structural representation of a set of IT components, organized in a useful manner. Each service has a set of abilities that are a measurable part of service contracts. Base services only on architectural concepts that are proven and reliable. If a service is new, or if you must support new and old services in parallel, you are often required to prove architectural concepts by first addressing the processes, operations, and components that support a particular service.

When planning services and architecture, organizations must be creative and think outside the box. Like the blueprints of a building, these designs define the expected structure and relationship between IT components. Architecture is also the part that defines the foundations and other structures that keep your house standing, so it is important to build IT foundations that will meet the long-term needs of your IT environment. Architecture is luckily not new to IT, so there are many different ideas about how to design services well. One of the most important points that I can make on architectural design is that often the needs of the organization—both business and IT—define the most efficient design for IT services.

Approaches to design often relate to the needs of the organization:

- It is rather common to ask different vendors to come in for architectural design sessions. One medium-sized publicly traded steel products manufacturer had a paid design meeting with two competing vendors—at the same time. It was an interesting discussion; our views of best practices were very different; yet on some points, we vehemently agreed. The result for this customer was that its managers were able to pick out the common best practices of vendors with very different experiences.
- Larger organizations often take extensive measures to communicate between different IT groups when planning new services. One U.S. insurance company took a unique approach when beginning its Microsoft Windows 2000 Server and Windows XP design process. They assigned the same project manager for both the server and desktop projects—allowing sincere interaction between these very important teams, because a single individual was responsible for both. The organization also assigned members of both the development and

operations teams to work in both projects, allowing the operations organization to be better prepared for supporting the new platform and allowing the development organization to benefit from the "in the field" experience of operations.

- Still larger organizations often involve architectural committees in the design process. These groups include managers and/or architects from multiple technology areas. Some services, such as electronic directories, require interaction between every platform supported in the organization. In the largest organizations, you must have discussions (and even projects) involving members of the mainframe, UNIX, and microcomputer teams all together—especially in larger infrastructure projects and wherever Internet services are concerned—because the standards on an architectural level should cover all information technology.

How does HP design service architecture?

When customers ask HP to help them design service architecture, we use the HP Global Method for IT Strategy and Architecture (HPGM for ITSA, formerly CSAM). This approach to creating enterprise and solution architectures has proved effective through many years experience in defining, guiding, and evolving complex information systems in multiple application domains. Our consultants and customer stakeholders participate in developing a four-view model:

- *Business view:* Why is the project being done? What are the motivations and business drivers?
- *Functional view:* What will the system do? What information will it provide?
- *Technical view:* How will the system be realized with IT components?
- *Implementation view:* With what specific products and other components will the system be implemented? In what organization? According to what plan?

HPGM defines each view using principles, models, and standards appropriate for the business domain. When combined, the four views let us understand the needs of all stakeholders and create a snapshot of what the solution should look like. An extensible framework of methods, tools, and techniques supports this methodology. It is the foundation for a standard set of architectural services with well-defined deliverables across multiple scopes (discrete project, initiative, and enterprise). (See Table 5.3)

Table 5.3 *Primary components of HP Global Method (HPGM) (Source: Hewlett-Packard)*

(HPGM)	Description
Project Management	HPGM is the methodology for defining project management based on more than 30 years of experience from Hewlett-Packard and Compaq. It is consistent with the Project Management Institute's (PMI, www.pmi.org) *A Guide to the Project Management Body of Knowledge* (PMBOK®)—*2000 Edition,* (December 2000)
IT Strategy & Architecture (formerly CSAM)	ITSA is the framework for structuring architectural descriptions and an integrated, complementary set of services that are characterized by shared stakeholder understanding and commitment, clearly demonstrable business value and low development risk.
Application Development	Application development provides a versatile and expanding range of methodologies that address the many types of application development, including a classic waterfall approach and a distributed objects methodology.
Implementation	Implementation contains proven, widely used IT service management (ITSM) based on the industry-standard IT Infrastructure Library (ITIL), including the mature Rapid Solutions Delivery.
Integration	Integration includes the Services Oriented Application Lifecycle (SOAL), plus Package Integration, Data Warehousing, B2E (business to employee), and CRM (customer relationship management) and the Systems Engineering Delivery Framework.
Support	Support contains the Service Delivery Guides and Technical Service Kits.

5.3.1 Plan the service architecture

When you decide to build a house, whom do you go to first? The bricklayer, or the interior decorator, or to the kitchen-supplier? Most would go to an architect, with whom you can discuss all the key factors and who can help you do the following:

- Look at your house from all angles in a coherent way.
- Identify the decision steps to take.
- Build and maintain an integral picture of what you want to achieve; that helps you make decisions and communicate with the different parties who will be involved.

What are some examples of architectural services?

In a similar way, HPGM provides a systemized approach of services the solution architect can deliver to capture IT information at an architectural level. Develop similar services for your service design staff to deliver. (See Table 5.4)

Table 5.4 *HP Global Method Services (Source: Hewlett-Packard)*

Vision and Strategy Services	*Discovery Workshop*—Helps clients understand the opportunities that early-mainstream technologies create for their business, and develop an actionable strategy and roadmap. *Art of the Possible*—Helps clients understand the best way to apply an emerging technology to their business (such as the Wireless Art of the Possible workshop). *Next-Generation IT Strategy and Architecture*—Provides insight into emerging IT and application trends, and formulates strategies and architectures for applying these technologies. *Business Value Assessment*—Helps clients identify opportunities to maximize the value of IT-related investments by fully leveraging IT capabilities and achieving operational excellence.
Architecture Definition Services	*Architecture Scan*—A consulting workshop that sets overall scope and direction for the architecture, aligning business needs with proposed IT solutions; creates a well-defined solution concept and secures buy-in from key stakeholders. *Architecture Assessment*—Investigates current information solutions and infrastructure, evaluates them against business needs and best practices, and recommends changes. A*rchitecture Blueprint*—Develops a fully defined architecture with sufficient detail to guide investments and implementation.
Architecture Realization Services	*Architecture Services Labs*—Creates proof-of-concept solutions and prototypes of key mechanisms. *Architecture Guidance*—Provides guidance on the implementation of an architecture, including interpretation, evolution, just-in-time architectural decisions, and/or program/project management. Can optionally provide a consultant with industry, technology, operational, and/or implementation expertise to provide objective guidance. Guidance can be for issues of architecture interpretation, evolution, just-in-time architecture decisions, and/or program/project management. This service can provide a one-time analysis with recommendations, periodic review and reporting, or ongoing program/project management. *Acceptance Verification*—Measures compliance of a system with established acceptance criteria and provides recommendations.

5.3.2 Address ITYs

You can measure the architecture in terms of abilities (ITYs) that are valuable to customers: scalability, availability, reliability, and the like. You must match appropriate abilities with each service; not every service customer desires perfect reliability, infinite scalability, or the corresponding extreme for any ability.

More often, IT customers require a specific set of abilities and capacity related to their own business function(s).

Availability

- Accessibility (also in usability)
- Continuity
- Fault tolerance
- Recoverability
- Repairability
- Resilience

Cost

- Affordability
- Efficiency (also in performance)
- Reusability (also in evolvability/adaptability)

Evolvability/Adaptability

- Compatibility
- Configurability
- Enhanceability
- Extensibility
- Flexibility
- Internationalizationability
- Localizability
- Portability
- Reusability (also in cost)
- Stability (also in manageability)
- Tunability

Goodness of fit

- Acceptability
- Accuracy
- Capacity
- Completeness
- Effectiveness

- Generality
- Precision/Resolution

Implementability/simplicity

- Economy of mechanism
- Feasibility
- Simplicity
- Testability

Interoperability

- Connectivity
- Manageability
- Maintainability
- Repairability
- Serviceability
- Stability (also in evolvability/adaptability)

Performance

- Efficiency (also in cost)
- Latency
- Optimizability
- Responsiveness
- Throughput

Reliability/correctness of operation

- Auditability
- Consistency
- Correctness
- Repeatability
- Reproducibility
- Robustness (also in security)

Scalability

Security

- Authentication
- Authority

- Authorization
- Confidentiality
- Exclusiveness
- Integrity
- Nonrepudiation
- Privacy
- Robustness (also in reliability)

Usability

- Accessibility (also in availability)
- Ease of learning
- Ease of use
- Intuitiveness
- Predictability
- Understandability

How do you set ability targets?

Abilities each have a different set of measurements. One of the easier abilities to understand is availability: how often the system is available. The calculation is mathematic; based on the amount of time the system is available, divided by the amount of time the system could be available. These measurements are rather easy to understand, yet still difficult to measure. The architecture is responsible for helping customers define the abilities required by each IT component. (See Figure 5.2)

Other types of abilities are not mathematical; they are simply either followed or not followed. Security is a good example. HP's IT department defines much of our ability targets for security using different types of access control that users can select for their particular type of information.

Figure 5.2
Availability nines

	Fully Available	*8,766.2400*	*525,974.4000*	*31,558,464.0000*
			Downtime	
Nines	Availability	Hrs	Mins	Secs
2	99.00000%	87.6624	5,259.7440	315,584.6400
3	99.90000%	8.7662	525.9744	31,558.4640
4	99.99000%	0.8766	52.5974	3,155.8464
5	99.99900%	0.0877	5.2597	315.5846
6	99.99990%	0.0088	0.5260	31.5585
7	99.99999%	0.0009	0.0526	3.1558

Table 5.5 *HP Enterprise Directory—security abilities (Source: HP)*

Ability	Definition
Access Control	Control access to the directory data is through access control on the data themselves. That is, the access control is stored in the directory. This lets the directory server make access decisions independent of the client. Access is granted to either an individual or a group. Access to privileged attributes is protected with access control lists.
Anonymous Access	Anonymous access is granted to all public attributes for anyone within HP. Within the firewall there is no need to authenticate to the directory to access the public attributes.
Authenticated Access	Authenticated access is granted based on the identity of the client. The directory supports two forms of authentication: distinguished name and password, or certificate-based authentication. All forms of authentication are conducted over an encrypted channel. Passwords sent to the directory using an unencrypted connection are not accepted. Password-based authentication is acceptable for gaining the right to read privileged attributes.

During SLA negotiation customers select many abilities, but some may be more effective when selected for particular components or during critical periods of business. Work with your customers to explore their required abilities. Gather this information and document it in a way that is useful to both your customers and other IT stakeholders—especially the IT staff.

Why are abilities difficult to measure?

Security provides a compelling illustration of why it is difficult to measure abilities, especially IT abilities. How do you measure the application of skills you consider "soft"? How do you measure best practices? Without a clear method to measure the impacts, how do you manage items and investments that "feel important"? Countless problems seem to plague the IT department when providing abilities such as security.

In fact, a few key issues affecting security management are clearly not specific to particular computer operating systems or business strategies. Consider the cross-sectional view of IT risks and some of the ways to address these issues as shown in Table 5.6.

In the IT world, many of these concerns are recordable events (for example, "number of virus outbreaks per month") and measurable activities ("average time to evaluate and put priorities on virus response

Table 5.6 *Common security concerns*

Security Risk (Architecture)	Common Risk Mitigations (Process)
Data interception	Secure (encrypt) information moving across networks to prevent unauthorized monitoring or recording of information.
Denial of service	Create procedures for limiting (through use of network routers, firewalls, etc.) the effect of intruders flooding components with information requests. Maintain availability and reduce the risk of server and other component crashes by keeping security patches up-to-date. This may also prevent other types of attacks that are easier after the component is overwhelmed.
Identity interception	Require users to secure their electronic credentials and passwords from discovery using social or technical means.
Macro viruses	Application-specific viruses could exploit the macro language of sophisticated documents and spreadsheets.
Malicious mobile code	Use firewalls or proxy servers to restrict internal users from executing code (Java, Microsoft COM) on unknown Web sites.
Manipulation	Prevent modification or corruption of information by using electronic signature and encryption methods. Use antivirus software on computers to prevent viruses from corrupting data.
Masquerade	Monitor IT components for unauthorized *and* unusual access. A masquerading intruder can pretend to be a valid user (or even another IT component) by mimicking login credentials, network address, or other authentication factor.
Misuse of privileges	Limit use of "full privilege" administrator accounts; intruders can use these accounts to obtain access to or destroy unauthorized information.
Replay attack	Use intrusion detection components to identify intruders "playing back" recordings of network information.
Repudiation	Use public key to digitally "sign" important transactions and ensure that the recipient can verify the sender.
Social engineering attack	Educate users and enforce password management policies; sometimes attackers use the more direct approach by simply asking users to verify passwords.
Trojan horse	Monitor computers and networks for applications that you have not authorized or cannot identify; some of these may be malicious software hiding its presence or acting like a useful utility.

strategies"). The concern of architecture is that you must find the most important (often the most common) issues and address them first.

5.3.3 Prove architectural concepts

Architectural concepts must be tested, documented, and reviewed to ensure viability of a particular service design. Adopt services from existing components whenever possible. Services built from existing components and other proven concepts are generally less costly but may require significant integration effort to integrate with newer services. When existing services are not available, model and prove new service concepts in a manner not disruptive to existing ones.

For architectural concepts (or hypotheses) to be "proven," in a scientific manner, there are some basic requirements:

- Build test plans and thoroughly test each concept to ensure that it is appropriate for the potential customers.
- Document and publish your architecture, test plans, test results, stakeholder feedback, and resulting changes to the concept.
- Provide a means for stakeholders to discuss the published results. Some of the most important feedback is peer review.

How can you organize proven architectural concepts?

HP organizes information in the enterprise architecture library in a very specific manner. Enterprise architecture consists of a number of main content types, as shown in Table 5.7.

Table 5.7 *Categories of architectural knowledge assets (Source: HP)*

Topic	Description
Domains	Domains or subject areas define direction, approved products, etc., for IT
Policies and standards	This page lists policies and standards by domain
HP's IT principles	HP's IT principles guide actions, decisions, etc., within the whole IT department
Products	Detailed product information for software used within HP
Initiatives	The major projects with which the enterprise architecture team is involved
Product evaluations	Evaluations of products that have been looked at within HP
White papers	Lists the white papers submitted to the enterprise architecture team
Glossary	Definitions for terms used within the enterprise architecture program

5.4 Service-level management

Service-level management is the practice of using SLAs to structure contracts between service providers and customers. These agreements allow customers to prioritize their requirements based on measurable factors of IT services. As the complexity of IT services grows, it is important to develop a directory of services that lets your customers find valuable services easily. Once a customer has decided on a set of services, you must negotiate an SLA that is fair for both the provider and the customer(s).

5.4.1 Measure and prioritize requirements

Develop your own list of potential requirements (based on the ITY needs discovered, as listed earlier), and ask your customers to help you rank them. The ranking is very important; it helps you determine where the customer perceives the most value in each particular investment. By reviewing its customers' answers, the state government (in the example earlier in Chapter 9) developed a very clear picture of its customers' business and functional requirements. Here are the areas that customers of the state's IT department felt were of "high" importance: (See Table 5.8)

Table 5.8 *Business and functional requirements (examples)*

Business Requirements	Availability and Integrity	Local systems not being down for more than 24 hours; compliance with standards for backup, restoration, and business resumption; backups occurring at night; restoration of services occurring in a timely manner; Internet access; 7 A.M. to 7 P.M., Monday-through-Friday availability; and network response times at least as good as the current environment
	Privacy	Protection of privacy
Functional Requirements	Confidentiality/security	Deflection of virus outbreaks; policy-based management of antivirus software; compliance with IT department security policies; compliance with federal, state, and local program requirements; and secure locations for servers and storage media
	Federal and State Mandates	Compliance with federal regulations
	Services and Support	LAN-based printing; access to agency and customer data; and network shared access to documents based on security levels
	Costs	A clear, user-friendly billing process; low cost for system maintenance; and financial viability of solutions for affected departments.

(Continued)

Table 5.8 *(Continued)*

Infrastructure	User account administration; Universal Power Supply (UPS) protection for servers; timely backups for all servers; Structured Query Language (SQL, a computer programming languages for databases) 7.0 platform; O/S support of existing platforms; adequate storage for electronic documents; identification of storage capacity, current databases, and file-and-print servers; printer management and support; file-and-print services; preservation of existing share design and permissions; support for ongoing changes to shares; quantification of impact on network regarding moving applications from a local network to a remote network; definition of IT department's role in management and delivery of networking; matching of current network performance; support for existing networking technologies; and secure network solutions
Environment	No item was singled out as having a high level of importance
Applications	Having a process to determine what applications/servers that are unique to agencies that will remain with agencies, and a tiered-criteria approach for agency ROI decision process; support for all current client/server applications, whether supported by agency or IT staff; and support for existing database platforms and versions
Security	Support for a multitiered, highly customizable security architecture; firewall security and encryption support for Web hosting of Internet applications; ability to disable a user immediately because of termination; expanded ability to limit access of confidential data by users; assurance that applications are accessible only to authorized users; use of password with unique ID to access system; automatic deployment of virus protection to desktops; maintenance of security patches and updates; and end-to-end protection of data

You must consider the requirements of your own stakeholders. These requirements should meet all the requirements of each service itself, but you will also use SLAs to reflect the priority of customer requirements as well. It is up to each service to define the expected service levels, because they will vary between services. It is also useful to define multiple levels of each service (for example, "Silver, Gold, and Platinum Service") to allow more choices for your customers. Software companies often use this method for providing support to end users. By providing multiple levels, you will be able to provide cost advantages to your customers. You will also use metrics for each service level to help you predict resource requirements. (See Table 5.9)

Table 5.9 *Service levels (example)*

Service Level	Included Components	Downtime (availability) Goal	Capacity Goal	Incident Goal
Platinum	Dedicated server	5 min/year (99.999%)	Per 500 users	15-min response, 1-hr resolution
Gold	Dedicated server	52 min/year (99.99%)	Per 100 users	60-min response, 4-hr resolution
Silver	Shared server	525 min/year (99.9%)	Per 1 users	360-min response, 24-hr resolution

5.4.2 Develop an enterprise service directory

The service directory (ITSM calls this the service catalog) is an electronic representation of IT components that describes service architecture. It helps you organize the architecture of your services to describe these services to customers in an easily understood, shared way. It provides customers with current information on what services are available, what components comprise each service, other features, billing and charges, and so on. You may need many directories to address internal and external needs, but when possible tie each directory together in a central location. One example in HP is the enterprise directory (ED). The ED is a highly accessible, highly available central repository of up-to-date general directory information of value to HP. ED is accessible: It provides standard programmatic interfaces, easy-to-use outgoing data feeds, and useful security/data access restriction mechanisms for groups and individuals within HP. The enterprise directory uses the Lightweight Directory Access Protocol (LDAP). LDAP is an evolving Internet standard for storing, retrieving, and managing directory data. LDAP provides a mechanism for rich searching capabilities and authentication.

What kind of information is stored in the enterprise directory?

The enterprise directory supports numerous applications, including information retrieval, authentication, authorization, group management, and email translation. The directory stores identification information such as name or employee number, credentials such as X.509 certificates (Internet Standard X.509 digital certificates, commonly used to encrypt email and other information) or login account names, verification information such as certificate revocation lists, affiliation information such as group membership or employment

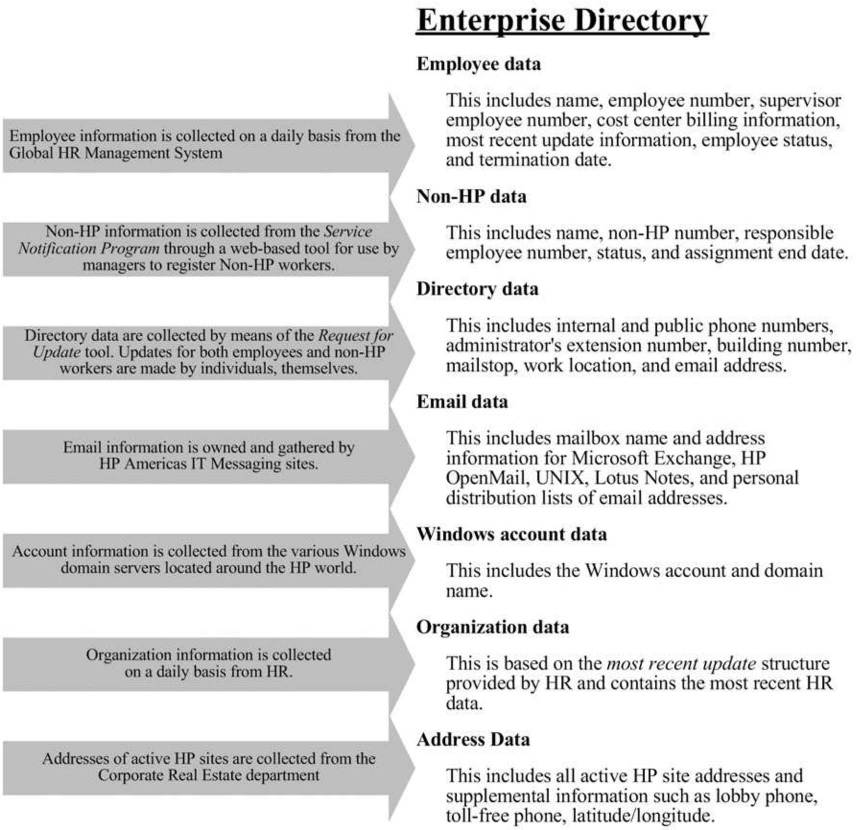


Figure 5.3 *HP enterprise directory—data model (Source: HP)*

status, and contact information such as email address or telephone number. This information is very important, because it helps critical services such as email find their way through a very large organization. In fact, email is based on use of two directories: the enterprise directory, and the information stored in the domain name system (DNS) directory. (See Figure 5.3)

How do IT components access the enterprise directory?

Access to the enterprise directory uses the specifications outlined for LDAPv3 (Lightweight Directory Access Protocol version 3). These specifications, in Internet RFC 2251, use a standard protocol enabling off-the-shelf servers and applications to access information in a consistent manner.

Table 5.10 *HP Enterprise directory schema*

Schema	
Naming	LDAP Enterprise Directory Distinguished Name (DN)
Data Ownership	The owners of various directory data
Indexes	The types of indexing available
Data Classification Policy (DCP)	The security levels protecting data (public, privileged, protected, private)
Operational Attributes	The standard attributes defined by LDAP that are retrieved by request only (time stamps, creators)
Employees	People who can demonstrate a pending, current, or past employment relationship with a recognized instance of Hewlett-Packard.
Non-HP	People are identified by the non-HP registration system owned by the Service Notification Program
Partners	The "Partners" branch is used to store entries for HP business partners.
Groups	Groups are lists of distinguished names
Email, Services	Populates services subtree with "generic" email addresses. These are email addresses associated with an application or a personal distribution (email) list, and not used by a single individual
Locations	Recognized HP locations have been assigned entries in the "Locations" branch.
Organizations	Recognized HP organizations have been assigned entries in the "Organizations" branch.

There are three basic methods for accessing information from the enterprise directories:

- *Data files* are files containing large amounts of enterprise directory data (such as general data for every employee in the company) that you can retrieve on a regular, consistent basis. You would use this method to merge enterprise directory data with an existing data set (for instance, if you need to load specific fields from the enterprise directory into your own database to supplement the data you gather for your application). The ED data files enable business applications to incorporate directory-related information about HP employees and non-HP workers, including names, billing information, building and mail stop, telephone number, email address, and simplified email address (SEA).

- *Programmatic*—You can pull information out of the enterprise directory in real time from your own application using a programmatic interface called an LDAP SDK (Software Development Kit). This method is good for looking up small quantities of information on an as-needed basis and can also be used for activities such as client-side authentication (verifying a user is who he or she claims to be) via Microsoft® Windows NT passwords or digital certificates.
- *Lookup tools*—PeopleFinder is a Web-based lookup tool that lets you view information about individuals, locations, and organizations. Use this method for occasional lookups of specific data performed on an as-needed basis by a person.

In January 2003, HP's enterprise directory used the basic architecture outlined in Table 5.10.

5.4.3 Negotiate service-level agreements

Service-level agreements are a contract between you and your customer; they represent the service you will provide and the costs associated with the service. Choose long-term measurements, because you will use SLAs to monitor performance, report on service levels, and encourage proactive enhancement of service levels. As a result, be careful to choose measurable elements for variable service contracts.

Service-level management is a process for helping both you and your organizational network to make better decisions. Your customer, in the Internet age, may be an internal or external customer or even a partner or competitor. I often find myself helping customers get a good deal from IBM, even though IBM is a competitor. This is part of providing good service.

5.5 Conclusion

Because an SLA is an architectural contract, there are likely as many strategies as there are attorneys. At the least, count on involving stakeholders in the following areas:

- SLA pricing
- SLA structure (service-level objectives)

- Measurement
- Reporting
- Change and configuration management (for example, provisioning, business need changes, contract renewals, and terminations)

Make cooperation your goal. In the Internet age, you need to remember the long-term nature of your relationship is as fragile as a popular "Anybody But ABC Corp" Web site. Service-level management is a method of sharing joint responsibility by focusing on stakeholder needs, structure, processes, constraints, and decisions. Some of the common elements include the following:

- Description of the stakeholders, service, deliverables, links to more information, and identified responsibilities
- Objectives for special abilities (such as number of customers, amount of data, and so on)
- Agreed schedules for service delivery and support (such as response to incidents, problem resolution, changes) and customer schedule of sensitive business activities and locations

Also consider implementing "back to back" agreements with different stakeholders—your customers and partners. In short, sit down with everyone and determine a governance (management) model that works. Test it, measure it, and if it does not work the way you expected, fix it! Just remember to give yourself enough time; to do this requires a clear view of your processes.

References

1. D. L. Parnas (1971, 1996), available on the Web at www.acm.org/classics/may96/.

2. Reprinted from *Communications of the Association for Computing Machinery (ACM)*, vol. 15, no. 12 (December 1972), pp. 1053–1058 Copyright© 1972, Association for Computing Machinery Inc. Parnas references Richard Gauthier and Stephen Pont (1970), *Designing Systems Programs* (C) (Englewood Cliffs, NJ: Prentice-Hall).

This Page Intentionally Left Blank

6

Process: Best Practices

Life is like riding a bicycle. To keep your balance you must keep moving.
(Albert Einstein)

6.1 Introduction

In this chapter, you will learn about the most important part of any best practice: process. It is not only the stuff of IT best practices—process is required to sustain life. Processes such as breathing, consuming food and water, sleeping, children—all these are the most basic in our nature, because they allow us to live. To maintain a supportable IT environment, you must understand the processes required to sustain the life of your infrastructure and applications. This chapter explores different kinds of processes that are important to information technology. The chapter discusses how to develop and test repeatable processes for both the infrastructure and application systems, in the section on process development, and then how to apply them to the existing IT environment, in the section on process deployment. It is not sufficient just to build and deploy processes. The section on process measurement can help you define ways to define metrics and monitor your performance of repeatable processes. To understand the importance of process to IT, this chapter requires you to ask only one question:

How would you start from scratch?

The current age requires you to consider many possibilities. Whether reengineering because of business need, management fad, natural disaster, acts of terrorism, or dumb luck—what would it take you to begin again, developing IT from scratch? Where would you start? What resources would you need? Most of all—would you even have much of this information? The answer for many organizations is a resounding no that would shock most executives. If we know that all it takes is one event to bring down an entire company, sometimes leaving nothing but worried shareholders and customers, why would anyone consider disaster recovery of such low regard? Are your backups so useful that they could restart you if the unthinkable actually happened? It is very difficult and expensive to rebuild an IT environment. Understand the repeatable IT processes of your organization—and document them thoroughly. This chapter teaches how to record and manage processes in a manner that lets you see the actual processes consistently—helping organize processes in a way more likely to stand the test of time.

6.1.1 The growing importance of process

One of the first reasons why process is so important is the concept of improving organizational processes or business process improvement (BPI). The concept is simple: We do many things when we work, but not all of them are necessary. When we improve these processes, there should be an overall improvement in the efficiency of the organization. BPI was a popular management technique during the 1990s. Other writers have called this business reengineering. The process of BPI is still a very misunderstood concept, especially for IT managers. One of the most often missed aspects of improving or reengineering processes is that you must do so

- Across many different processes
- In constantly changing ways
- According to a repeatable process

It is not sufficient to simply improve processes; you must actually make process improvement a part of your basic business process. There are many books about improving processes. According to one online bookseller, over 16,900 books have the word “process” in the title. Books are available about business processes, research processes, scientific processes, and biological processes. Every profession and industry has some book about its processes. In fact, you can learn a process for almost everything—with one glaring exception—information technology. Contained within the same bookseller's almost 2000 titles on information technology are a lonely 41 titles containing the natural combination of "information," "technology," and "process." A closer review shows that most of these books are actually about "information-processing technology." In fact, only about three loosely related technical books exist on the critical relationship between process and technology. At the end of the technology revolution and beginning of the information age, one might argue, our collective literature seems to indicate we are either not concerned about or completely unprepared for the massive changes required to support the process effects of information and technology.

The effects of this missing process are spectacular—similar in many ways to the effects of the cotton loom on the textile industry, or the effect of air cargo on trains and ships. Entire industries change in only a few years under the impact of new technology. Now we face the effects of technology on our most precious possession: information. The growth of information-handling technologies during the IT boom of the 1990s has created a universe of information contained in each organization's technology products and solutions. The result is that there are likely more unused PCs in the world than there are people, and more discarded floppy disks than grains of sand on your favorite beach. For better or worse, business has changed because of technology. It is uncommon to find an organization, of any size and in any industry that has not been subject to the effects of the personal computer, email, and the Internet. The Internet has created entirely new classes of products, workers, companies, industries, and laws while fueling an economic boom—until it burst like a soap bubble on hot cement. Now that the bubble has burst, what comes next for technology workers and the industry? In fact, what comes next for the sellers and purchasers of technology? Part of the answer lies in the technology we use today.

Why is managing processes so challenging for a data processing organization?

IT professionals designed each of these new solutions to meet the needs of a particular implementation—related to a unique set of needs—without considering that the very technology they implemented might actually change those needs over time. The result is that each organization, each industry, and largely each individual, have developed generally inconsistent and hard-to-understand technology implementations. Put simply, few organizations required IT to draw blueprints or get permits for any of their buildings. It is not uncommon to see the effects of this inconsistency within a single organization—a model in which so many different groups are responsible for technology purchasing and operations that no one group can claim responsibility for anything. By the time you get to the desktop applications, nobody is in charge. Nobody is responsible if it breaks, and nobody knows who put it together.

Technology has barely begun to change to meet the needs of business. Organizations are not prepared for the inevitable future where their customers expect anywhere, anytime, personal responses to their inquiries and demands. Most organizations have been able to live under the assumption that they could always reboot servers or submit large-batch jobs any night of the week—an assumption that falls apart immediately when you put up an e-commerce Web site or become part of a global organization. Few if any organizations will be immune to the demands of technology; these changes are deeply rooted in the global economy: in governments, financial institutions and exchanges, every industry, and every form of communications. On a typical day, the inhabitants of Earth use technology to store and process uncounted trillions of bytes of data—or more. The statistics keep growing—and so do the storage requirements. Bandwidth needs increase, as do requirements for processing cycles. Technology affects us at the epicenter of our economy and in our daily lives, but although popup weather messages are handy, not everyone agrees on the benefits of this effect. (See Figure 6.1)

It is, therefore, imperative that we learn new ways to manage the business parts of technology. On one fact we all agree: Business is demanding more measurable value from technology every day. CEOs are desperately searching for how to use IT. Shareholders and citizens are only beginning to

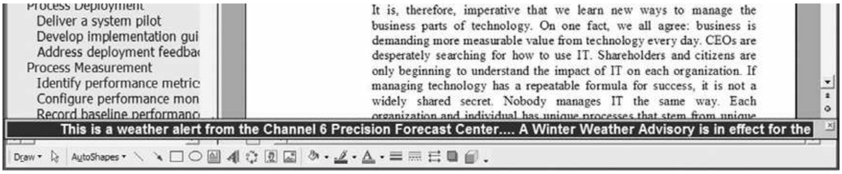


Figure 6.1
Web service example: Popup weather notification

understand the impact of IT on each organization. If managing technology has a repeatable formula for success, it is not a widely shared secret. Nobody manages IT the same way. Each organization and individual has unique processes that stem from unique technology systems and solutions—and unique business problems. Everyone is concerned about IT—but nobody really knows what to do with IT. The formula for technology success, it would seem, is missing.

6.1.2 RAPID processes

In RAPID, process is the sum of all the things you would need to do to rebuild an IT environment from a shopping list—all the parts you would need, and all the processes you would use to build it and support it. This is probably not all information that you have today—although you undoubtedly have enough to begin with. Organization of this information is critically important, and you must record it in a way that ensures the survival of your IT organization; you simply do not have time to rebuild all your IT from scratch without directions. If you depend on technology for business survival—especially if you support stakeholders over the Internet—you must sustain the life of your IT. You cannot possibly expect to measure every process in IT. You must focus on certain processes for survival: the repeatable ones (like breathing, eating, and building servers). One of the most important benefits of ITSM (information technology service management) is that it allows you to organize and set priorities on the processes that are inherent to IT. Structure, however, is not enough—you must actually record your processes in a structured manner. Then make someone responsible for maintaining those processes. (See Figure 6.2)

Another way to put this in perspective is to examine, in business terms, one of the most costly processes supported by any organization: new hires and internal transfers. The following figure explains this concept in terms of business process flow (solid lines) and the IT process flow (dotted lines). From a business perspective, starting a new hire is rather easy. In Figure 6.2, the employee fills out a few forms and the hiring manager forwards these forms to the appropriate local departments for processing. The manager requests assets such as a desktop, productivity software, and information access permissions. Each department forwards the forms to other parts of the organization that perform increasingly specific services related to the form. Depending on how efficient the organizations' processes are, the required

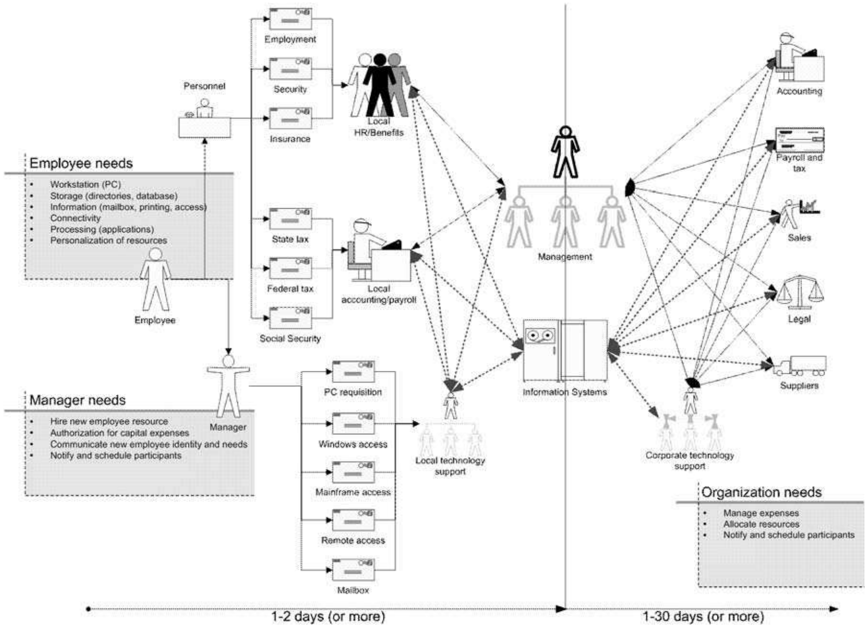


Figure 6.2
Business view of hiring process (large organization, decentralized IT)

services become available in a few days or weeks. Changes requested after the initial hire or transfer (such as a new account, application, or permission) cycle through a similar process. (See Figure 6.3)

For many organizations, especially those created before the dotcom era, IT processes serve as the channel for communications between local and organizational IT resources. The local computer guru was, since the beginning of the PC era, the business' primary line of communication between IT needs and IT solutions. The typical IT provisioning process, as explained in Figure 6.3, begins with a hiring manager selecting an applicant for a particular position or set of responsibilities. The manager contacts a local IT resource who sets off a similarly unimpressive sequence of events. The IT resource assigns LAN and mainframe access accounts, a desktop computer, and resources such as applications and storage. Each organization, according to its needs and complexity, may automate all or part of the provisioning and configuration processes of individual products.

How do you classify repeatable processes?

Luckily, IT provides us many opportunities to record repeatable processes such as hiring or desktop deployment. Technology-dependent organizations (much of the financial industry, for example) have built very detailed

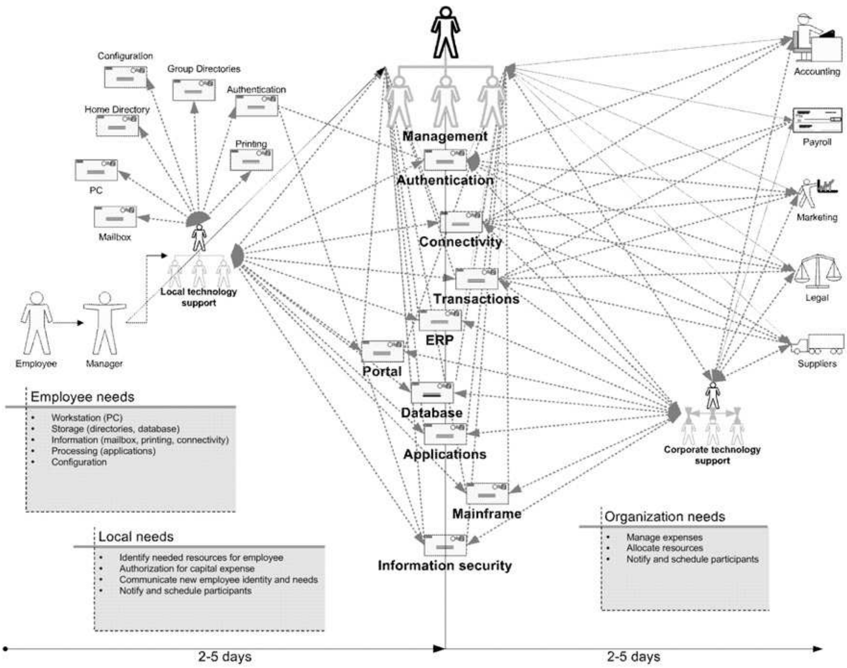


Figure 6.3
IT view of hiring process (large organization, decentralized IT)

processes that describe their IT operations—how to configure things, how to change them, whom to notify, and so forth. These standard operating procedures, operations, consist of parts of standards for processes, executed in a specific order. (See Figure 6.4)

You can differentiate processes and operations in RAPID (although the dictionary and most of the IT industry often treats them as equals) by using the following criteria:

- A process is any repeatable sequence of events (often called a repeatable process) that is generally consistent globally, or across an industry. In the IT industry, for example, processes include sequences such as "IT projects" and "pilot-testing" that are often similar in many different organizations. An IT project is how many organizations build new solutions, and pilot-testing is the process of user-testing a new IT system. IT processes include standards for exchanging information, such as the Internet standards. IT processes are the things information technology does repeatedly, without regard to vendors or specific technology and business needs.

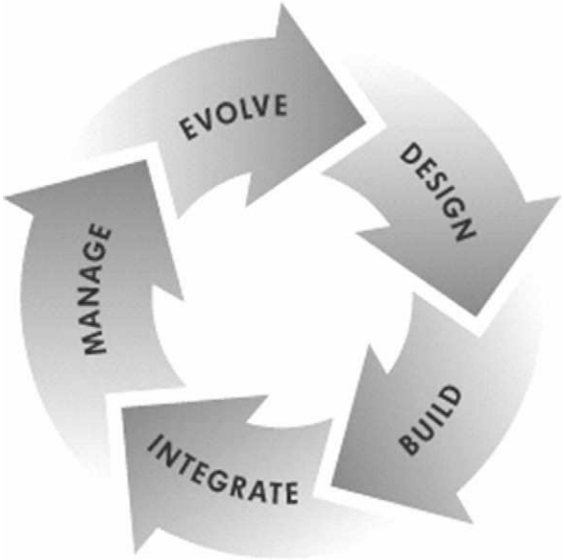


Figure 6.4
Repeatable process for adaptive infrastructure (Source: HP)

- An operation is the application of a repeatable sequence of events (a process) using the resources and methods of a particular organization. Where one organization may perform pilot-testing only in production, other organizations have specific test facilities—some even have actual remote buildings and network connections specifically for testing of disaster recovery, global applications, and even new IT systems. IT operations are the things that information technology does in your particular organization to support your unique business and technology needs. Unlike processes, however, some operations are not always repeatable. Troubleshooting a new customer support incident, for example, uses parts of many repeatable processes to accomplish a goal. The operation combines components providing resources and components (subprocesses) of the repeatable processes, sometimes in a process that itself is repeatable or affects components in a repeatable manner.

This chapter focuses on the repeatable version—the process. Again, separating process from operation (or even sometimes from architecture) requires standards. Often these standards depend on the needs of each organization. Another way to look at it is to consider where your commitments lie. Whereas some organizations are prepared to make Windows, Linux, or any other specific technology an internal standard, you may want to consider processes involving specific technology as an operation. This provides a more predictable definition of process that will let you focus on how the more important processes—the ones you must always do—affect each operation. HP ITSM is useful in identifying repeatable processes, because it not only defines the architecture of IT processes, but it also describes the processes themselves. In ITSM, the processes are two "how" questions:

- *Build and test:* How will IT develop and assemble services?
- *Release management:* How will IT deploy and activate services?

Why are processes and operations different from each other?

A clearer example may be the distinction between "Build and test," a process, and "Build a Linux server," an operation. The manner in which you build a server, or whether you build "servers" in the first place, depends greatly on the state of the technology industry. As we progress, new technologies will help you manage computing in entirely new ways—for instance HP's Utility Data Center allows you to build services as you manage a storage area network today—component by component, without much regard for the individual hardware. IBM is also a leader in this field with its Linux-on-mainframe initiatives: You can build hundreds of Linux servers and run them all on the mainframe as individually or collectively managed instances.

Process development—"Build and test"—is an entirely different beast. Process development, for IT, is simultaneously the place where we have the most experience and the road most not traveled. Data processing, information systems, and information technology departments have all developed millions of computing processes. There are processes for credit card authorization, customer tracking, e-everything, and even some that describe the software development process itself. ITSM and its parent, ITIL, have led the field in IT process development. Chapter 5 discussed how to use these processes to develop architecture.

6.2 Process development

Since the early days of humanity, someone has always been out there trying to find the best way to reduce work using some form of technology. From the abacus to the printing press to the PDA, people have used technology to perform a simple feat called engineering. In practice, it is not that simple. Electrical, chemical, mechanical, and civil engineers are only the beginning of the list. The term engineering belongs to an entire genre, and applies to many different forms and media, but generally people agree that engineering enables a benefit by using an improved process. The stone became a wheel, and people needed a cell phone and a PDA (personal digital assistant) to manage their newfound mobility. At some point, someone always decides we must organize each group of our habits to become either a standard process or a "bad habit." Like all good habits, we gave this process a name: business process reengineering. It was a very popular business concept during the Internet boom, only waning for a few moments as the world finished building the 20th-century Internet and prepared for Y2K apocalypse.

How do you reengineer IT processes?

Business process reengineering, one form of finding roadblocks, is not a new concept for IT professionals, but never one we have had to address internally until now. We have been the process makers, the process recorders, and the process supporters. Our software, hardware, and passion for technology have brought significant change to business while slowly alienating each of us from our business roots. We developed calculators and spreadsheets, databases and knowledge management. We glued each new invention of technology into its place through the tireless acts of "midnight migration crews" and insistent information officers. We helped business gather information faster than it could possibly use that info. Some of our best engineering efforts come from eliminating parts or processes that are more challenging than necessary.

One of the things HP's team has done to help customers reengineer processes is to develop detailed project plans. Although these project plans may have been very helpful to the project team, is it necessary for the IT department to reinvent the wheel each time its members approach a new project? Could there be better ways to manage projects than traditional project management disciplines such as the project plan? The project-planning process is very valuable, because it lets stakeholders see not only the overall process flow but also the dependencies between activities and resources. (See Figure 6.5)

Before you can use IT services, you must build repeatable processes to address the needs of both services and stakeholders. These processes include those needed to build and test infrastructure and application systems, as well as standard methods for addressing stakeholder feedback. Infrastructure systems are the basic common service components required to deliver IT services to users. Application systems include the software used to provide direct stakeholder value. Testing must address the unique needs of infrastructure and application systems. Involve stakeholders during the development and testing process, to address critical feedback provided during this time.

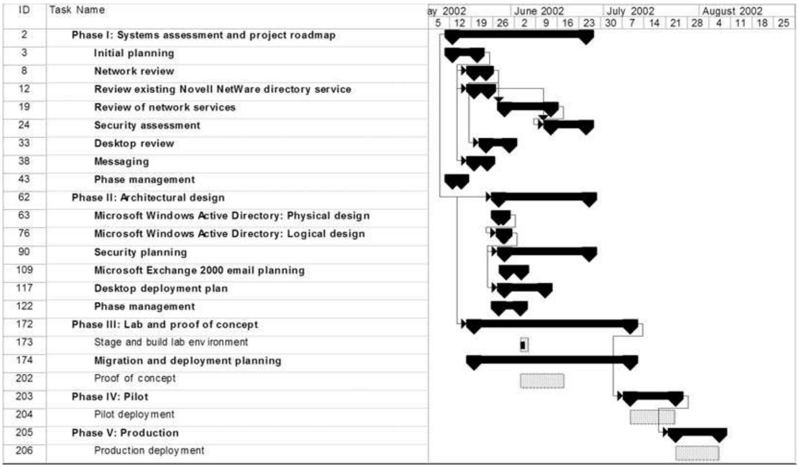


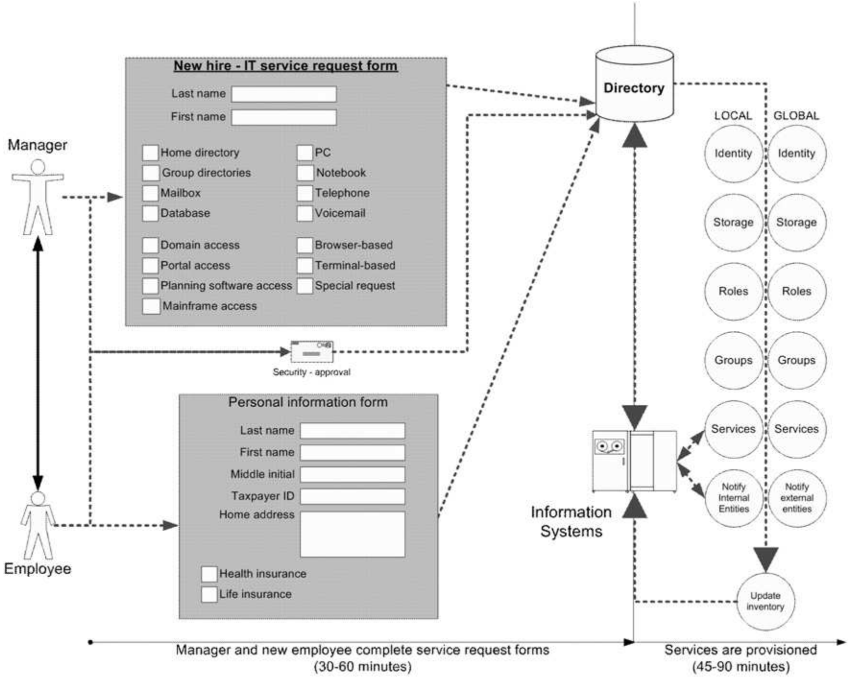
Figure 6.5 *Reengineering project plan for technology infrastructure*

Look not only for processes to fix but also for processes to eliminate. If you glance around your organization and see people doing a lot of busywork, this may be an important indication that some processes need to be de-engineered. How many people actually need a paper copy of that report you print every day? How often do you print out email? Moreover, why does your car need a checkup every 1000 miles when others only need 100,000-mile checkups? One critical distinction you must often make in process development is how you define your infrastructure. Maintaining your infrastructure is critical—it is the things that help hold your systems together. Because infrastructure is common to many stakeholders, you should create easily repeatable processes; the more detailed your processes, the more successful you are likely to be in actually repeating them. The Rainier group worked with Microsoft, for example, to help it develop a “process knowledgebase” it could use to centralize project planning product releases. The system learns about the common tasks for managing software releases by finding similar tasks in projects and making these tasks available to each new release project. Again, automating the project-planning process itself gives Microsoft a way to reduce work for project planners. Microsoft employees in general loathe doing any work they can find a way to automate.

How will you improve processes during development?

Another example of how to develop integrated repeatable processes is to look again at the new-hire provisioning process. In organizations using directory-enabled processes, such as HP and Microsoft, the process of managing new hires looks very different from the way it looks elsewhere. Although the information still comes from managers and employees all over the company, these companies manage the entire process as a series of Web operations that route the appropriate documents to the appropriate stakeholders. The electronic directory allows applications in different divisions throughout the company to find employee information, current reporting structure, and other related applications. When a new employee comes to any organization using directory-enabled processes, these groups allow the various applications to detect and handle the "event" of a new hire. When the manager requests a new hire, a directory creates the initial electronic identity of the new employee. This identity grows as various information publishers throughout the organization process information and register new records in the enterprise directory. The employee may sign a few forms, depending on regional requirements, but then registers by web for the majority of business services. The central directory combined with a "single sign-on" web site (such as Microsoft Passport) allows users to use a single logon and password to access web sites throughout the organization. (See Figure 6.6)

Figure 6.6
Technology view of hiring process



Organizations using the Internet standard Lightweight Directory Access Protocol (LDAP) accomplish this process by creating groups of information subscribers (represented by circles in Figure 6.6). These groups allow interested parties such as payroll and human resources (HR) to request electronic notification by various processes in the organization. As roles change, the directory lets applications send the right messages to the new holder of a specific role. For instance, I belong to groups in the HP enterprise directory that identify me to different organizational processes. I have groups that let me receive information addressed to my various affiliations: pre-merger Rainier, pre-merger Compaq, and HP Services. There is a group for my office building, one for my city, and another for my assigned "HP Services/Consulting and Integration/Enterprise Microsoft Services/Windows and Messaging/practice" (team names get very long in directory-serviced environments!) that includes my manager and a group of individual contributors. I have a number of groups that are mailing lists for my various technical interests: messaging, directory integration, .NET, mobile computing, and security. I even have groups that send me notifications if someone schedules my particular Microsoft Exchange mailbox server for maintenance. (See Figure 6.7)

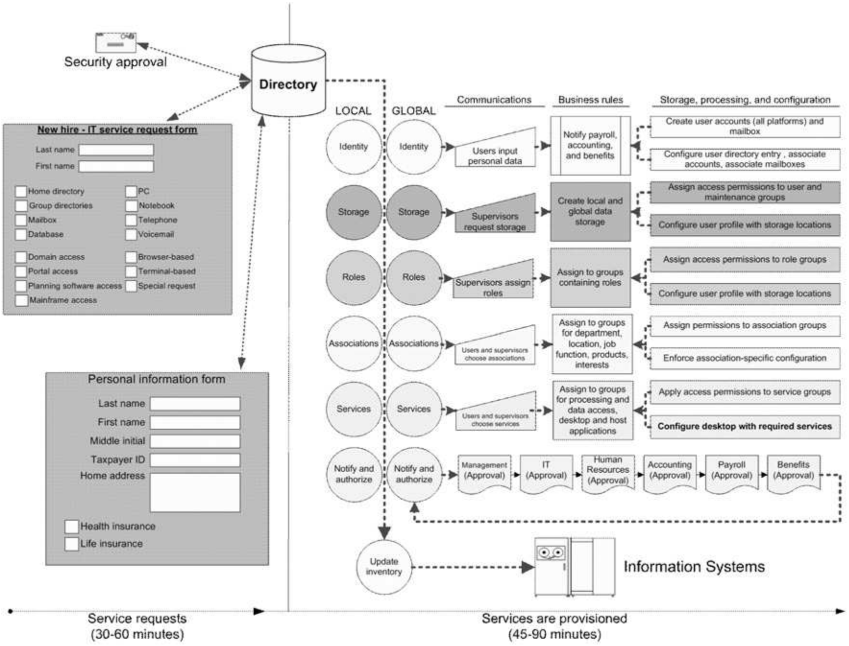


Figure 6.7
Operational view of hiring process

For the various groups involved in the new hire process, the result is that you can build business processes that do not depend on every individual IT component making (and subsequently needing to communicate, process, and keep updated) a copy of basic identity and location information for business objects. The HR data go to HR databases, the payroll data go to payroll databases, and systems publish small portions of this information to the directory, which advertises where all the information is stored. For the new hire, the result of this tight integration is a set of integrated components forming a very expedient and efficient hiring and starting process. Consider the value of some of the effects:

Table 6.1
Cost effects of process changes

Effect	Metrics
Eliminates most paper forms for new hires	*Reduced costs:* hiring manager involvement, paper, printing, distribution, collection, data entry, data storage *Increased costs:* electronic forms development
Eliminates data duplication and scattering	*Reduced costs:* data storage and processing for duplicated identity information: name, address, city, state, zip, position, manager, etc. *Increased costs:* management of security, privacy
Unknowns	Network traffic, web server processing, web server storage, desktop processing

Note that using integrated information directories has security and privacy implications. This is a rather complicated process to do from the top down. Each group should designate an owner, responsible for administering security, privacy, and other ability requirements. You must then teach owners to manage the levels of information access that other individuals and groups should have to their data. The directory provides the initial "Who is this user?" information, and each component in the organization is then responsible for enforcing data access according to business architecture and processing rules stored virtually in the enterprise directory.

6.2.1 Build infrastructure systems

To build IT infrastructure, you must first understand the history and importance of infrastructure. If early civilizations had not quickly understood the value of infrastructure, it is doubtful humans would have ever made our way up into civilization in the first place. Frontinus, a water commissioner in the Roman Empire, wrote:

> *I think no one will doubt that the greatest care should be taken with the aqueducts nearest to the City (I mean those inside the seventh mile-stone, which consist of block-stone masonry), both because they are structure of the greatest magnitude, and because each one carries several conduits; for should it once be necessary to interrupt these, the City would be deprived of the greater part of its water-supply. (LacusCurtius: Frontinus on the water supply of Rome, translated by Charles E. Bennett, 1925 accessed in May 2003 at http://www.ukans.edu/history/index/europe/ancient_Rome/E/Roman/Texts/Frontinus/De-Aquis/Text*.html a124)*

The utility infrastructure supporting and connecting consumers in the United States is similarly critical. Gas, water, power, roadways, railroad, and now the Internet—each stretches in a semiredundant manner to connect every city and county, from producer to consumer. These pathways are both the security and the artery of the economy. Whereas the loss of a single power station rarely disrupts power for anyone, the loss of many could bring disaster. Some of these pathways are so crucial that we have no choice but to protect them at all costs. Reliability, in utilities, is an important goal of all primary stakeholders: producer, distributor, and consumer.

How does infrastructure affect IT?

The problem with delivering IT services is that, unlike traditional utilities, there are simply too many components to track using conventional methods. Imagine, for example, a detailed monthly telephone bill of futuristic IT services: at the top would be a total, and then a breakdown of cost by component category. For example, one line item might read "Data storage—$4 million." Each service would be further broken down into individual transactions, subscriptions, and measured services provided. Eventually you would reach the level of individual software applications and hardware devices—with charges for each of these services. At the bottom would be a total. At some point, the business will notice the charges on everyone's corporate cards. These bills, from each IT utility provider, may then become your responsibility to sort out and understand.

One of the key methods of simplifying IT services is to protect the role of the infrastructure. Certain solutions, as they grow from (hopefully) a test lab into production, are indeed as important as the water we drink and the power we consume. The Internet, email, Web browsers and services, desktop PCs, and information storage have come to the forefront of the Internet age. They are ubiquitous in society and in most businesses. For many organizations, losing any of these core services for an extended time could spell destruction—or at least the implementation of a disaster recovery plan. In fact, that may be the best definition of infrastructure: what you must recover in an emergency. Another important role of infrastructure is to account for certain investments. If you implement infrastructure as an operation, you can attribute infrastructure maintenance investments to that operation. Because many infrastructure operations are common to many different services and processes, you will find you have spread out the costs of maintaining infrastructure evenly among the correct services, and therefore to the correct customers.

What part of IT is infrastructure?

Only accept a few instances where you say, "That's infrastructure, everyone should pay for that." Otherwise, this dictum begins to sound like just another way of loading overhead on customers. Only a certain part of IT fits within the role of infrastructure, just as in any utility. Some applications as well, such as productivity suites and email, may be important enough that users would notice significant downtime. Custom applications on each desktop are usually important to a single group or role only. In this case, the application is only infrastructure if downtime would

- Affect a majority of the overall users
- Affect other infrastructure components
- Be financially expensive to business
- Affect other critical business operations
- Directly affect influential stakeholders

Not all parts of the enterprise network (nor all parts of the Internet) are required in order to maintain status quo. This is not to say that these services are not important—it is simply that the majority of users will continue to function without them. Parts of the environment that provide service to every user are definitely infrastructure. Networks, servers, data storage, and mainframe computers are certainly infrastructure. Any service directly supporting the majority of users is suspect. To identify your infrastructure, look first for the pieces of computing that most users need and then for the pieces of computing that connect other systems together.

How do you build and manage infrastructure processes?

Each time you build a new component, you will decide whether it is infrastructure. The infrastructure switch on a component lets you do a few key things to your IT management environment:

You must manage infrastructure systems separately. If any of these systems fails, you will certainly have to respond more quickly than for most application systems. These failures often affect many users and even other stakeholders. You may want to set priorities on some systems even within the infrastructure, because some outages are more visible than others. You can enforce the priority of individual components by creating operations to manage them.

You must manage changes to your infrastructure processes carefully by communicating thoroughly with stakeholders. In infrastructure, unscheduled outages—or even changes to schedules—can have significant impact on users. If you provide a help desk service, you should also define unique operations for informing the help desk of infrastructure change schedules.

Create a repeatable process for managing infrastructure. This process will define the life cycle of the infrastructure: service availability schedules, change blackouts, critical times for individual stakeholders, service maintenance schedules, and anything else you regularly do to manage and maintain your infrastructure. You may create operations for each of these items.

One way to look at processes is to split them up into flowcharts. This very mature practice of computer software development lets you see the workflow of a discrete process. A more modern view of a flowchart is called a "cross-functional" flowchart. A large U.S. financial services company worked with my team to develop detailed repeatable processes for deploying Windows XP. Processes such as "deployment" were broken down into multiple operations such as "creating desktop images" and "installing workstations." Each operation consists of component activities, such as "preparing the workstation." (See Figure 6.8)

By documenting each operation and how it relates to the overall process, the customer hoped to enable support resources to understand the relationships between the operations and components in a single repeatable process. This type of classification is important, because the relationships become more complex at each level of depth in an organization. The insurance company did not stop there, however. It also worked with HP consultants to develop a software application that would enforce its business rules while automating the distribution of software. The application was intentionally modularized to allow quick migration from Windows-based to Web-based operation management. Chapters 7 and 8 explain more about how to manage infrastructure and component changes, and these chapters also review the important aspects of rules that affect maintenance of the infrastructure. For now, create a list of the items your stakeholders consider infrastructure; focus on making sure the list is rank-ordered to account for the resources and activities of each item in the infrastructure.

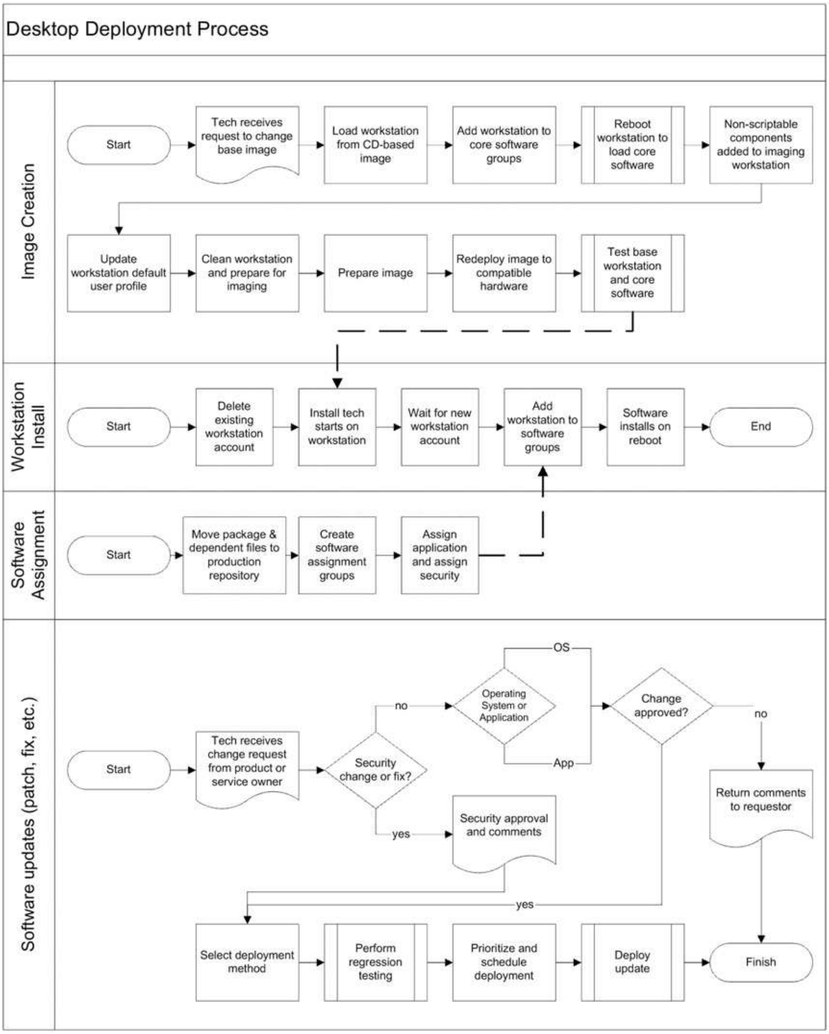


Figure 6.8 *Desktop deployment process*

6.2.2 Build application systems

After you create repeatable processes for maintaining the infrastructure, you will also need to create a process to maintain applications. Some organizations define applications as "anything that runs on the operating system" or "components other than the PC and the operating system"—but remember that some software in addition to the operating system has recently become infrastructure. Email was once an application that had significance only to individual departments; now most of the organization uses email. In this respect, email applications have become infrastructure. Other components may be part of the infrastructure as well; your organization may classify antivirus applications, productivity applications, and others as infrastructure. The rest are applications, and they too need a process for management. Applications are the components that run "on the infrastructure." Each application is also subject to a process. Many components can benefit from automatic setup: Web sites, server and desktop applications, user profiles, and others. Each application should have a repeatable process for implementation. One example of a process for building application systems is the application installment script or package. The package stores rules and a process that defines how software is installed on a given operating system. Each of these packages is an operation, but the overall process to build these application packages is a process.

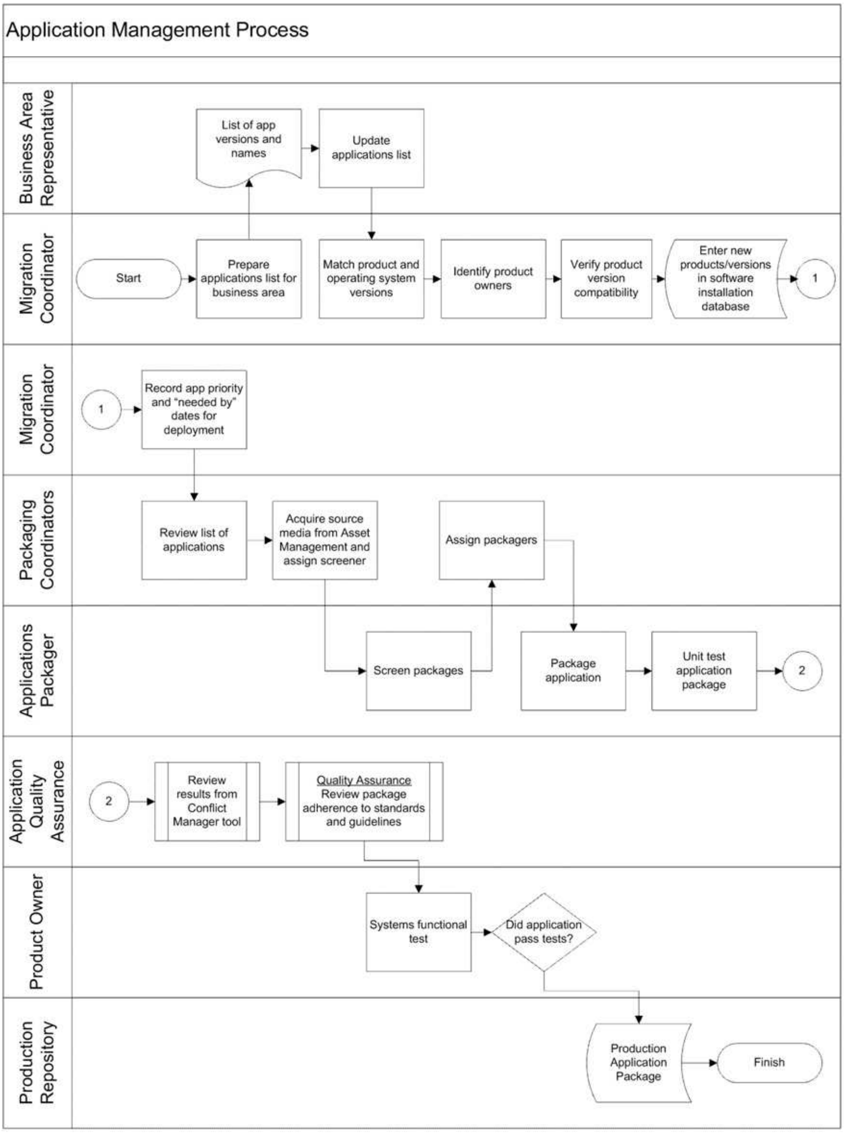
What does a repeatable process for application systems development look like?

On the Windows platform, for example, you can accomplish the installation, removal, and configuration tasks (on a single computer) by packaging applications. Microsoft has created a system for installing applications on Windows systems using MSI (Micro-Soft Installer) packaging technologies. Choosing to implement packages is itself a financial choice. It costs money to buy the third-party tools such as Wise Installer, WinInstall, and Installshield. It may not make sense to invest in creating packages for software that is no longer used, or if only a few users will use the package. Set thresholds (constraints) in your application systems process that let you determine when and if your team should package an application. The insurance company mentioned as an earlier example built a great standard process for managing the packaging of its 1300 desktop applications during migration from Windows NT to Windows XP: (See Figure 6.9)

Again, the company implemented this process in software; in this case, the company chose to build a Lotus Notes database to control the workflow of software packagers. This let the migration project stakeholders look up the status of a particular package at any point in the migration process. Build repeatable processes for at least the following aspects of application systems:

- Requirements identification (often per business unit)
- Application source distribution
- Installation and removal
- Configuration
- Testing
- Stakeholder communications

Figure 6.9
Application management process



6.2.3 Test the architecture and repeatable processes

Just as in any other form of learning, repetition of IT processes is essential to understanding and documenting them. It also helps you find "sporadic issues" that may not appear if you only test a process once. First test your architecture and processes in a test environment, then rebuild them for a pilot, and sometimes rebuild them again for the production environment. Whenever you make design changes during testing, restart the testing process to ensure that the entire repeatable process is tested. One of the most important investments you can make in your IT systems is the development of systems specifically for testing. Of course, it depends on where your investment priorities are—but execution of testing plans is still critical in any case.

How do you test architecture and process?

Test each process you develop and the architecture connecting them to other systems. Develop standards for how you perform the testing process, and enforce these standards. For example, when testing an infrastructure component you may require that a group of testers (independent from the project team) test the component. Network and server infrastructure may require performance and scalability testing, whereas for PC workstations you may only require performance testing. (See Figure 6.10)

One of the ways IT departments can ensure failure is by failing to test major change processes in a test environment before placing them in the production (customer-supporting) infrastructure. You can prevent this from happening by testing your changes thoroughly before they ever hit

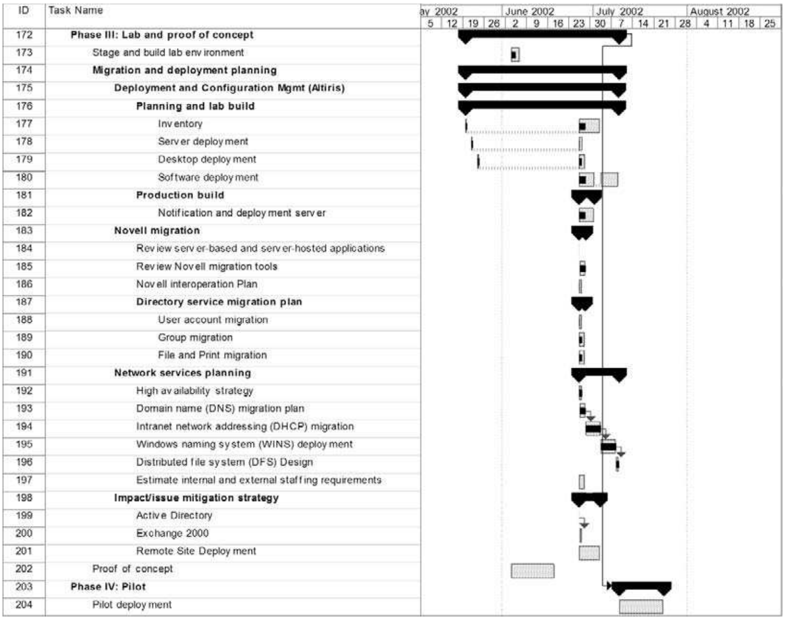


Figure 6.10
In-project testing plan

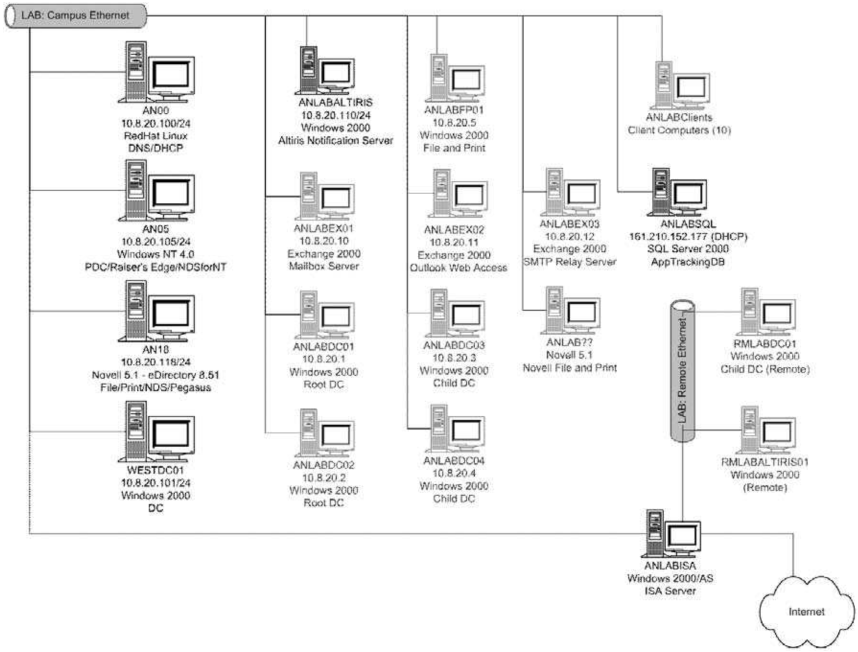


Figure 6.11
Process-testing lab architecture

the production network. Whether you support a network of 10 servers or thousands, you must invest in the ability to test without affecting your production users. For large projects, you may need to develop a unique test lab infrastructure that survives throughout the project. One of our customers, a state community college system supporting 20,000 students, used a detailed testing plan combined with a few Compaq-provided demonstration servers and a classroom full of desktop computers to simulate the operations of their network during their Novell to Windows migration process. Although they were not able to mimic their production hardware 100%, they were able to mitigate challenging issues such as student registration, security migration, and directory integration between Novell NDS (Novell Directory Services) and Microsoft Active Directory. (See Figure 6.11)

Why is it important to test processes in a lab?

One telemarketing company our team worked with found out the importance of testing the hard way, shortly after the consultants left, as one of their network engineers accidentally trashed the production Windows Active Directory by installing an early-release version of one particular software system for telephone call management. The change was at the schema level, irreversible, and caused many months of service inconsistencies for the customer as the two behemoth vendors fought out how to resolve the issue. The result was that after having successfully designed and deployed parts of Windows 2000 and Active Directory to support thousands of customers in just weeks, the customer then spent months without the ability to complete the migration because of the inability to deploy new directory (domain controller) servers. The root cause of this issue was clearly skipped testing processes and failure to enforce standards through limiting access to directory updates. (Luckily, the company is now much safer, as it subsequently fired many of their information technology managers and hired away one of our more knowledgeable Windows 2000 resources as a replacement.) It is not sufficient to build and test great processes; you must also involve the stakeholders in obtaining approval to move forward before beginning process deployment.

6.2.4 Address development feedback

Process development should always create feedback. Developers may find areas for improvement; testers may find “bugs” and other issues. Record and address all this feedback. Sometimes it will require changes to the processes that need retesting; sometimes it will require you to publish an “issues list” that allows users and support resources to quickly find issues on common unresolved processes. Some development feedback, however, does not affect the operation of the system or is outside of the original scope. Unless an issue is both in scope and affects the operation of the system, postpone the change by having the stakeholder submit it as a change request.

6.3 Process deployment

Once you have developed your new IT systems, you can then begin deploying your new systems into the existing production environment. To release new systems carefully, start with a small, measurable group of customers. Focus on success during the first production release to the small group of customers: the system pilot. The pilot provides an excellent opportunity to focus on documentation such as implementation guides. Also measure changes to existing processes, and provide feedback to the service teams if these changes affect SLAs. During each customer release, solicit customer feedback and address any issues that affect expected service levels.

6.3.1 Deliver a system pilot

A pilot is a test of an IT system in the production environment that involves users doing actual business work. You will usually start with only a sample group; it is difficult for the development group to have sufficient resources to support testing for all the stakeholders. When developing an ERP (enterprise resource planning) system, for example, include a sample of all the stakeholders using the system. If the project is large enough, you may want to consider asking your stakeholders to help you build an ongoing support operation for deployment testing of additional users. Some extremely large projects, such as desktop deployment and infrastructure upgrades, will require you to deliver a pilot for each group of stakeholders. Pilot processes should be a standard part of your organization, to enforce consistency between projects and between programs. Retain and organize the documentation and other artifacts created for use during pilot delivery:

- *Development feedback and design reviews*
- *Implementation guides:* The main navigation aid to help you deliver a pilot. This document contains the necessary information, guidelines, tools, and references to review an existing environment and help prepare and plan the migration to a new one.
- *Prerequisites checklist* to make sure essential information is available to conduct the pilot effectively
- *Service sign-off forms* used by the customer to convey acceptance of the completed pilot deliverables
- *Project plans or process models* for service delivery activities (and reserving the client's time for interviews)
- *Presentations* to show what the pilot will achieve and deliver, describing scope, deliverables, and benefits of the service.
- *Questionnaire forms* to assess the overall customer strategy and situation (may come from the readiness phase; this document will help gain time and influence the approach to delivering the service)
- *Templates* used by the project manager to complete the statement of scope and delivery schedule
- *Administration guides*
- *Acceptance test plan templates* for planning pilot tests and documenting the results

- *Detailed design specification reports* for the pilot hardware layout and configuration
- *Lists and comments* of participating stakeholders
- *Specification worksheets* for resources participating in the pilot

6.3.2 Develop implementation guides

One of the primary responsibilities of any group developing and deploying repeatable processes is to gather sufficient stakeholder comments (input) and create sufficient documentation (output) to repeat each important process. Implementation guides allow you to create repeatable processes that are extremely detailed and easy to recreate. Your goal should be to document each process completely, from start to finish. Implementation guides need not be only documentation in the "typed it in Microsoft Word" sense but can be any publishable form of electronic knowledge, from the word processing document to the XML (eXtensible Markup Language) representation of a business process. In fact, there are benefits to having multiple dimensions of knowledge; a process may be much clearer to your implementation teams if you document it in Word, but the process is much more repeatable if you also develop and record an automated installation script. (See Figure 6.12)

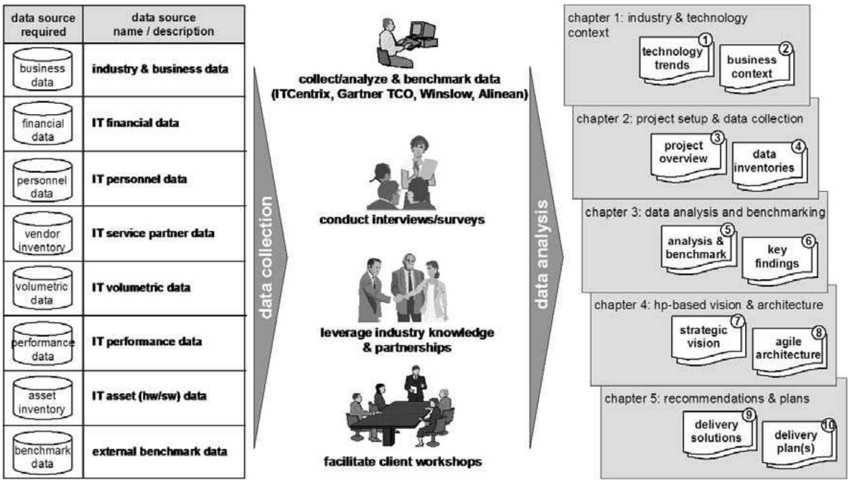


Figure 6.12 *Business value assessment-inputs, processes, and outputs*

Consider Figure 6.12, illustrating the process implementation guide for teaching consultants HP's business value analysis (BVA) service. Documentation is not only a book describing the process you have developed; it also includes definitions of the processes and resources you use to perform particular activities. You must ensure that not only each process is repeatable but also that the process is capable of handling exceptions during execution. To avoid support and reengineering nightmares the next time around, make sure you describe the input and output of each process clearly.

6.3.3 Address deployment feedback

Again, feedback is important. Each time you execute a process you must request feedback—whether you are developing a process for the first time, releasing a process into production, or making changes to an existing process. Processes only become repeatable and standard when they are important to a large group of stakeholders. It is important to assign owners to deployment issues; often deployment issues are problems. After stakeholders have accepted a system pilot (often called a "customer acceptance test"), most organizations handle feedback as incidents, problems, or change requests. Chapter 7 explains incident and problem management, and Chapter 8 can help you implement change and configuration management.

6.4 Process measurement

It is not enough to deploy processes that "look" effective. You must also measure the common processes and operations, repeatable or not, used in service-level agreements. When you first implement a new process, you must help stakeholders define the metrics to measure performance of processes. Often this includes implementation of monitoring components that gather and store performance information on the repeatable processes. After recording a baseline of performance, you can measure and report on the changes in individual processes.

6.4.1 Identify performance metrics and measure changes

Processes not only change during development or deployment—they are always changing, according to the needs and capabilities of your people and the technology. By managing and measuring the changes to individual processes, you can learn a lot about how you manage processes as a whole. Choose metrics you can use to measure process change, metrics such as time/cost to develop a new process, time/cost to change an existing process, and time/cost to execute an existing process. If you record performance statistics with each process, you can easily build sophisticated models to explain the impacts of process management.

Performance metrics are a critical part of using balanced scorecards. Each process is subject to a certain number of measurements (metrics). These metrics let you appraise the processes and tie the processes to expected levels of performance. To create measurable services, choose one or more performance metrics with which to measure the processes. Choosing the right metrics is important, because there are many potential metrics for each investment. HP advises "The 3 Ms" for metrics: meaningful, measurable, and manageable. By answering the following questions, you can adapt the guidelines for selecting effective metrics for balanced scorecards at HP to meet the needs of your organization: (See Table 6.2)

The balanced scorecards reflect the business needs of the technology organization, but you must also choose internal metrics to measure each component. It is not the monitoring itself that is important, but rather the application of monitors to enforce performance agreements. Put simply, it is useless to guarantee 100% availability if you have no way of measuring availability. To your benefit, IT gives many methods of monitoring performance. Once you have selected the metrics appropriate for each component, you must record them and enable monitoring of each affected process.

6.4.2 Configure performance monitors

Technology processes allow for detailed, automated performance monitoring. There are many ways to monitor the performance of processes. One of the most important measures of a process is the amount of time it requires of people and technology. Each process should be responsible for exposing some metrics. You can use these metrics when writing service-level agreements and change requests; they are also useful when defining the metrics of operations (see Chapter 7).

Some organizations may choose to record performance by monitoring servers and user workstations. Many devices can use the Simple Network Management Protocol (SNMP) Internet standard for accessing configuration

Table 6.2 *Balanced scorecard metric selection*

Questions to ask	HP BSC Requirement
Meaningful	
Does the metric correspond to your balanced scorecard strategy?	Linked to the organizational strategy and the delivery of shareholder value
Does the metric clearly address the needs of stakeholders?	
Will the metric be supportable in service-level agreements?	
Will your stakeholders understand the metric? Do other processes use the metric?	Rationale for selecting the metric should be intuitive and easy to communicate
Will changes to the metric affect other processes?	
Measurable	
How often will you collect performance data? How often will you review performance data?	Available within a reasonable time period after close of the scorecard period
Can you record statistics without modifying existing systems?	Collectible with minimal technology/ resource investments
Where will you collect information? How will you record and implement changes to metrics?	The metrics should not be changed within the measurement period
Manageable	
How many metrics will you allow?	There should be no more than 12 metrics
How will you identify and address "out of date" metrics?	The metrics should be fixed within the measurement period

data for network-connected devices. Using SNMP, you can usually access information about the device, its operations, and performance statistics gathered since the device last restarted. The Windows operating system exposes performance-monitoring capabilities using standards for WBEM (Web Based Enterprise Management) and Windows Management Interface (WMI) as in the following example. WBEM "is a set of management and Internet standard technologies developed to unify the management of enterprise computing environments."[1] You can use WBEM both to monitor and later to manage systems. XML standards for WBEM are still in development. You can create performance monitors for each IT process using standards such as SNMP and software programming interfaces such as WMI. Your processes are important, but you will often find value in monitoring each component.

6.4.3 Record baseline performance

You generally first record performance baselines at one of two times:

- Immediately after user acceptance of a system pilot
- Immediately after full deployment of a system to all users

Major changes, such as migrating additional customer groups, may change the expected performance of a system, and should therefore update the baselines. The baseline performance is a measurement of current expectations for system performance. If you are measuring a process according to cost and time-to-execute, for example, then you must record the performance of the process against those metrics.

6.5 Conclusion

As much as we all rely on repeatable processes to manage our daily lives, it seems that quite often we live in a process-averse culture. Not everyone can or wants to do the same job in the same way; in fact, it is only through differentiation, creativity, and "thinking outside the box" that we consistently learn new ways to improve on our processes. The process engineering and reengineering work continues; even after investing in the development and deployment of well-documented processes, consider the processes you will use to keep moving forward. Not every task is repeatable, and therefore not every task is a process.

Consider the following guidelines when implementing repeatable processes:

- Invest in helping stakeholders understand why repeatable processes are important. Governance alone will not cause people to follow a process; enforce processes by creating measures that turn the processes into desirable habits.
- Structure processes carefully to ensure that cross-functional relationships and processes are considered. Whenever a process is transitioned from one decision-maker to another, consider the time it takes to communicate the hand-off.
- A process is only as good as the architecture that supports it. When the architecture changes, you should review the effects on related processes.

- Some processes are not processes at all; be consistent in your definition of the difference between processes and operation. If you find that a process impedes performance, it may make sense to make it an operation and remove some overhead.
- Find ways to make self-sustaining processes. A repeatable process should define the manner in which the organisation will control change to the process in the future.

Process is not where you begin, and it is not where you end. Process is the glue that binds the architecture and operations together. The clearest definitions for development, deployment, and measurement are not sufficient to define all of the processes in use in modern enterprise-level IT architecture. Beyond the repeatable processes are the day-to-day operations that actually allow you to deliver valuable services to your customers.

References

1. Distributed Management Task Force (2003), available on the Web at www.dmtf.org/standards/standard_wbem.php.
2. James H. Harrington. (April 1991). *Business Process Improvement*: The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness.
3. Michael Hamneer, James Champy, James A. Champy. (June 2001). Reengineering the Corporation: A Manifesto for Business Revolution.

7

Integration: Coherent Infrastructure

7.1 Introduction

Integration is what happens when you bring architecture and processes together with your resources. Your model for integrating your IT enterprise must focus on the needs of the most integrated technologies. Infrastructure is the parts of a solution used by more than half of an organization. These components are so important that without them, you *have* no solution. The network, directories, email, and other Internet operations are often of such great importance to customers that the organization itself must agree to accept the standard definitions—often called *infrastructure.* Maintain your infrastructure consistently; these components are your heart and lungs. This chapter can help you learn to model integration of technology components, by discovering how the relationship between architecture and process affects the day-to-day operational components of IT. The section on operations infrastructure explains how the infrastructure supports core services and the integration needs of all IT systems and processes. The section on operations management can help you manage the day-to-day operations of your IT environment, and the section on operations support addresses the needs of incident- and problem-based support for stakeholders.

What are operations?

Operations are the day-to-day processes; the ways most IT support staff spend their workday. It is important to continually improve these infrastructures and remove complexity. What does it mean to reduce complexity of your infrastructure? Why is it important in the first place? One HP

Table 7.1 *Challenges to IT operations*

Challenge	Indications
Reducing IT complexity	Exponential vendor hassles with duplicate costs for management and integration across your global locations Inefficiencies around islands of disconnected, incompatible, or multiplatform environments and proliferating systems Paying your staff to solve problems they weren't hired to solve Lack of ability to predict, reduce, or report on management and support costs
Ensuring cost-effective stability and flexibility of your infrastructure	Inefficiencies supporting multiple or incompatible platforms, all with separate service contracts Accelerating management costs across islands of disconnected IT environments Lack of alignment between your business continuity and recovery requirements and your current IT capabilities Underused systems and struggle to meet ROI and TCO goals
Optimizing today's assets for tomorrow	Adopting the Web as your business channel, yet your internal systems are not connected to your Web delivery methods Meeting the demands of your mobile workforce with existing assets and minimal investment Under pressure to provide ROI proof before management will support a full-scale new technology implementation Looking for a strategy that allows for long-time cost savings despite unpredictable computing needs Budget and resource limitations are forcing you to support end-of-life or outdated proprietary systems

presenter explained some of the most common customer concerns: (See Table 7.1)

How do you manage operations?

Operations are systems—they are one in the same. Operations are when you integrate architecture and process with the actual components; putting resources to work on activities under known constraints and well-communicated expectations. The answer to effective operations is having the right resources make the right decisions at the right time, just in time. Too often, we consider that the only way to manage operations is to have a single person or vendor responsible for a number of computer "boxes" that happen to be running software. IT departments change focus, almost cyclically, between architecture and process—without maintaining the investment between cycles. In fact, the only method of managing IT operations that has proved effective over time is the balanced application of infrastructure and software management.

Carly Fiorina put it simply to HP's operations staff during the December 2002 meeting; you cannot have one without the other:

> *Finally, strategy, structure and process, and rewards and metrics are what I would call the hardware of a system. Culture and behavior are the software of the system. And like any system, whether it's Superdome or a $49.99 printer, the hardware and the software has to go together. You can't have an effective system, one without the other. So, how we get things done is as important as what we get done. It is vital to delivering on the entire architecture . . . This is a framework that the Executive Council and I used to measure where we are and where we're going. It's a framework I've talked a lot about with the senior leadership team. This is how we managed the business and it is how we will manage the business going forward.*

7.2 Operations infrastructure

How you address these concerns—based on the reality of your organization's current state—is a primary component of how you manage your infrastructure. Some organizations (such as HP) even divide the infrastructure part of operations into two parts: core infrastructure and applications infrastructure. Depending on the size of your organization, you may also need to maintain the core and application infrastructures differently. Keeping the applications infrastructure under a separate set of rules lets you focus on maintaining the heart: your core infrastructure. Create lists of the components in your core and applications infrastructure, as shown in Table 7.2.

7.2.1 Maintain an adaptive infrastructure

A certain part of your infrastructure, the core, is a lifeline for your business. It serves objectives by generating immediate and ongoing value through better communications—connecting systems, processes, and people. As technology becomes a more standard part of our lives, these expectations will only continue to test the abilities and limits of customer-facing applications. As the economy changes, the organization will require greater ability to use information technology assets. This is not to say that return on investment and total cost of ownership are the only important expectations; rather, customers will expect positive IT financials for systems that provide consistent and balanced growth in many abilities: security, continuity, and others. (See Chapter 5)

Table 7.2 *Infrastructure operations (Source: HP)*

Core Infrastructure	Disaster recovery Distributed infrastructure services Enterprise management Solutions Enterprise operations High availability Information security	Internet infrastructure services Mobility Networks Personal computing environment Platform infrastructure Printing infrastructure
Application Infrastructure	Application runtime Environments Business intelligence Platform and tools Business process Management Collaboration Content management	Databases Development framework and tools External application Integration Internal application integration Portal integration framework and technologies

Your IT infrastructure will increase in complexity and diversity, and could become more difficult to manage if your core infrastructure services cannot change with the changing needs of stakeholders. Part of HP's strategy for helping customers meet these needs is an initiative called *adaptive infrastructure.* An adaptive infrastructure links business and IT for increased business agility with real-time access to information for analyzing the demand of resources and ability to adjust quickly the supply of your computing resources to take advantage of new business opportunities or react to changes in demand before your window of opportunity disappears. (See Figure 7.1)

HP uses the term *adaptive infrastructure* to describe a way to make business and technology more agile, that is, more responsive to changing needs. To carry to the extreme the paradigm of adaptivity, the infrastructure must be able to dynamically adjust and modify itself to meet your business needs in real time. Resilient and reliable infrastructure ensures that critical resources are there when you need them. The infrastructure should also serve as an open, extensible foundation that will meet your needs today and tomorrow.

IT infrastructures today have basic abilities for management and monitoring. An adaptive infrastructure, however, understands the nature of this information and responds automatically, based on business rules defining operation of the infrastructure. To create an adaptive infrastructure requires you to create core services—the ones you think are vital to be able to respond quickly to changing business needs. From a technical perspective, an adaptive infrastructure should provide the elements shown in the figure. (See Figure 7.2)

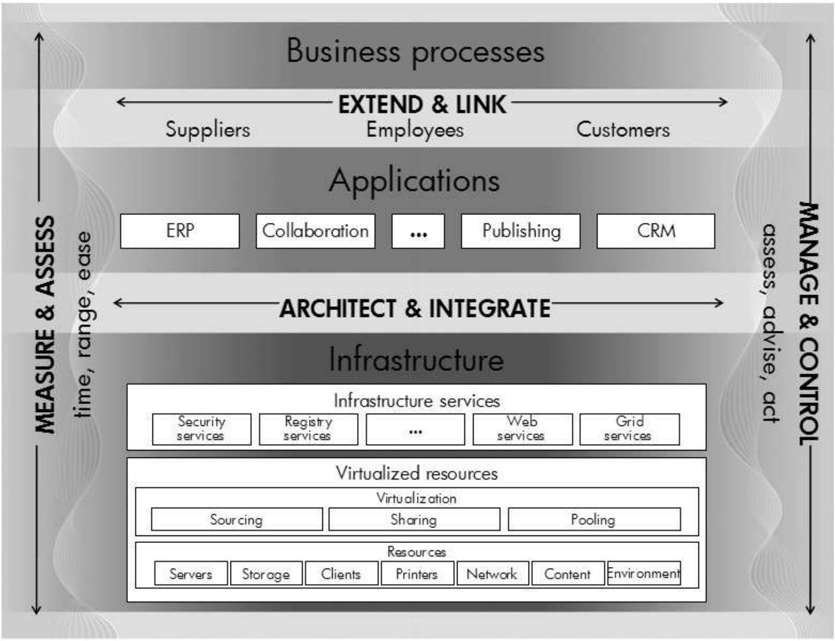


Figure 7.1 *Definition of adaptive infrastructure (Source: HP)*

This adaptability delivers a completely new level of flexibility by allowing rapid allocation of resources with minimal effort and cost: automating routine maintenance and resource allocation tasks, monitoring systems and processes to proactively identify and prevent faults, optimize performance, and manage service delivery end-to-end. Integrating all three areas will help you enable truly adaptive infrastructure. For IT, integration is the process of bringing together the architecture and processes. This happens when you manage the infrastructure; in fact, it happens every time you plug one thing into another. Integration happens when you are loading software onto a PC, or connecting a wireless bar code reading device to a network. Integration is the point at which both the user and the infrastructure have a significant stake in the operations components of every IT department.

How do you create an operations infrastructure that adapts to changing conditions?

A key concern in creating adaptive infrastructure is how to integrate the architecture of computing systems with the processes used to manage them:

> *There are a number of similarities and related issues between software architectures and software processes . . . From a technical viewpoint they can both be decomposed into components that treat information, and*

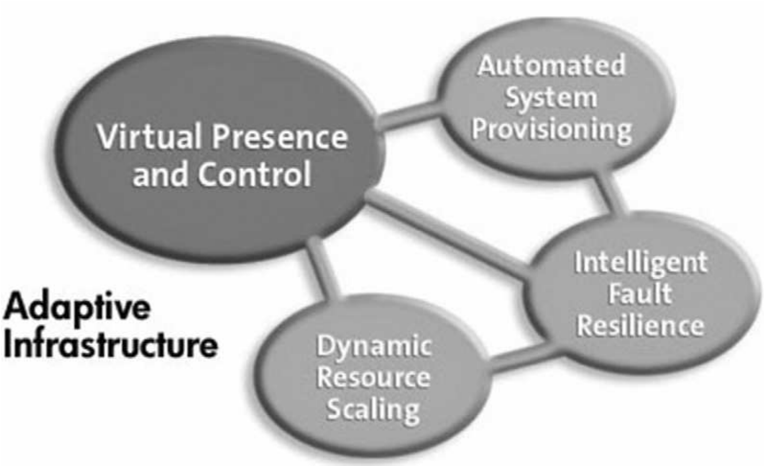


Figure 7.2 *Components of adaptive infrastructure (Source: HP)*

Intelligent Fault Resilience	Ensures that your business continues to operate uninterruptedly, even in times of unpredictability and threat
Dynamic Resource Scaling	Uses virtualization capabilities across servers and storage
Automated System Provisioning	Incorporates intelligence at every level of the infrastructure to give you better control over the delivery of essential business services

output some type of data. This view may be useful especially when trying to automate parts of the software process. It may also be useful when trying to find strategies for decomposing a software process into activities. A problem with this view is that whereas software components can be controlled and behave deterministically, the teams and personnel in a software project present challenges in social and cultural behavior.[1]

7.2.2 Create standards to integrate architecture and processes

The integration of architecture and process is significant to IT, because all sciences are again exploring the relationship between structure and function. This relationship gives IT the ability to store, process, and communicate information. It dictates the required "protocols" or languages supporting information transfers. Integration is how IT classifies relationships between architecture and process.

There has recently been a renewed interest in the relationship between structure and process. Researchers have looked at the way that the structures underlying information systems are (re-)shaped and how they in turn have an influence on our lives . . . The picture that emerges from such research is of action being constituted by—and constitutive of—structure, with structure being [sic] a product of action and both an enabler of and a constraint on action. (Workshop description for "W8: Structure and Process: The Interplay of Routine and Informed Action," at the Seventh European Conference on Computer Supported Cooperative Work, 18–20 September 2001; Bonn, Germany, available on the Web at http://cuecscw2001.gmd.de and accessed on May 2003.)

One of the information technology ways to explain what it means to integrate architecture and process is to consider the change of the Ethernet networking standard from transmission rates of 1 gigabit (1 billion information bits per second) to 10 gigabits (10G Ethernet). The 10 gigabit per second Ethernet protocol Institute of Electrical and Electronics Engineers (IEE) standard 802.3ae makes a critical change in Ethernet: For the first time, the network signals use multiple "channels" of information. Information in the next superhighway will, instead of hanging out in one lane waiting for a chance to pass, enable four lanes of data-moving "pavement." The result, however, is somewhat challenging. Ethernet depends on a space between information, called the *interpacket gap (IPG),* that provides idle time between data "frames." (See Figure 7.3)

The 10G Ethernet uses a round robin pattern to shift information into four channels (the "lanes" in a "data" highway). This is somewhat like filling four buckets with an even number of apples—you put one apple each in bucket 1, 2, 3, and 4, and then go back to bucket 1. To maintain compatibility with the basic standards—and the underlying idea of knowing when to expect data to arrive if interference happens—designers of the 10G standard would need to find a way to maintain that useful gap. The challenge for the designers of the new 10G standard was that the architecture of the Ethernet data-moving process had to change. You cannot predict how long

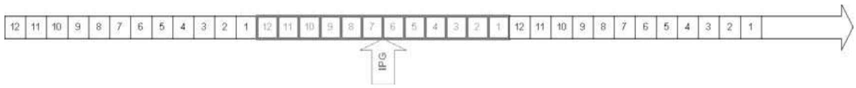


Figure 7.3 *Traditional 1G Ethernet interpacket gap (IPG)*

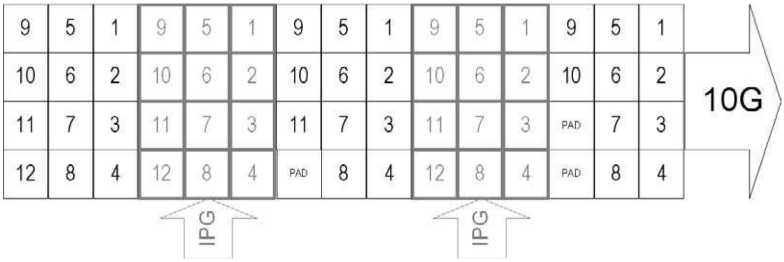


Figure 7.4
10G Ethernet interpacket gap

each transmission of data will be. Their response was simple, but effective: (See Figure 7.4)

To ensure that the next bit after a packet gap was always in the right channel, the designers considered inserting "padding" into blank bytes between frames and required the first character of a new frame to come in the first lane. The problem was that this padding was unused space, which directly affected performance. To avoid giving up 10% of the performance, the designers allowed network product vendors to address the issue in three ways:

- Add the padding, making the IPG 12 to 15 bytes.
- Shrink the size of each packet gap by up to 4 bytes, making the IPG 8 to 11 bytes.
- Average the sizes by combining methods 1 and 2 with a counter, making the IPG average 12 bytes.

This is what happens when you integrate architecture and process: Changes to the standards affect all your stakeholders. Just like the effects to Ethernet equipment manufacturers and customers, change in your architecture can degrade performance or increase errors, but changes in communications processes can result in incompatibility. Your response, like that of the Ethernet standards group, should be to consider both the architecture and the process. The Ethernet standards group, however, had an advantage when making decisions: Packets are much easier to analyze than people are.

How will you integrate architecture and processes?

If architecture and process are clearly related, then the next logical question is how to handle the relationship. In fact, careful integration of architecture and process allows the organization to identify and classify relationships that it would more often find by accident. This allows your IT organization itself to become adaptive, so that it can constantly evolve to meet the needs of your stakeholders. One adaptive feature of HP's internal enterprise architecture

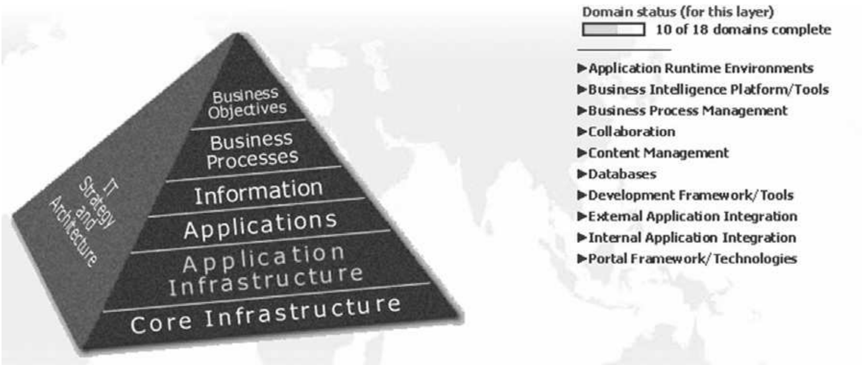


Figure 7.5 *HP Enterprise Architecture Navigator (Source: HP)*

(EA) is that through the Web site you can actually watch EA evolve as the "new HP" becomes "just HP." The IT group provides an extremely intuitive user interface; you simply move the mouse over an area of interest and your Web browser displays a menu of the particular concept and each of its subcomponents. You can instantly see which areas are nearing completion, or click on an area to direct you to a new page providing specific information about the ongoing development of a particular topic. (See Figure 7.5)

By integrating the IT enterprise architecture with each component and the actual process of developing and maintaining the architecture itself, stakeholders can "check in" to see how the actual process of documenting and maintaining our architectural rules is progressing. Putting together large IT organizations such as Compaq and HP means a lot of compromise and a lot of new documentation. If you find you do not have much documentation of your current infrastructure, don't miss this opportunity to document it now.

Even if you argue that you have plenty of documentation today, how do you organize it for tomorrow? Everyone agrees we need to do something different to manage IT services for tomorrow, yet nobody seems to agree on exactly how to integrate ITSM with the actual IT organization. RAPID recommends you manage this relationship close to the technology and technology people. Today, most infrastructures have the basic ability to manage and monitor any number of technical parameters. Do you know what the parameters are, and what they mean? Understanding what that information means, and having the ability to quickly act on it, allow you to proactively support your organization—in effect allowing you to plan the future by setting standards. (See Figure 7.6)

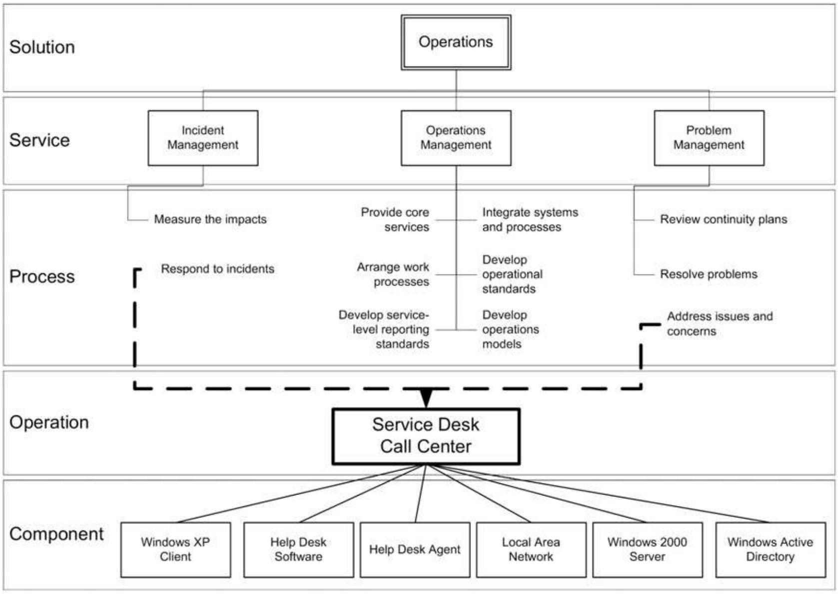


Figure 7.6 *Integration of call center operation*

You create operations to support integration of the architecture (services and service components) with the repeatable processes. The operations are the standards your organization uses to manage day-to-day IT needs, such as support call management. The service desk or call center receives telephone calls from users. These users expect the service desk to help resolve their problems. The figure depicts a service desk that is responsible for incident response and addressing stakeholder concerns. To perform this operation, you must integrate certain portions of multiple services and processes with a number of different components. Although this may seem obvious, consider the financial impacts of this transaction in an IT services model. Simply answering the telephone call in the service desk requires you to use many different components. Is each component responsible for charging the user according to usage, or is there another measurement that matters? You must define standards to help you manage these operational relationships.

How do standards affect integration?

Just as standards were critical to the growth of the Internet, pervasive industry standards throughout your technology infrastructure will be equally important to the future growth and success of your IT infrastructure. Consider how your organization views integration standards. Depending on your needs, you may handle these challenges differently:

- Government agencies are often interested in openness of standards, to facilitate citizen and cross-agency communication.
- Corporations are often interested in protecting past investments by integrating aging and new technologies.

Starting with industry standards protects your capital investment in technology, ensures that your infrastructure will be compatible with future technologies and prepares you to take advantage of those technologies more quickly than your competitors. Industry standard technologies increase your business agility and improve integration with the rest of your technology environment, partners, and customers.

HP, for example, concentrates commitment to industry standards in three primary areas, as shown in Table 7.3. (See Table 7.3)

7.2.3 Optimize technology assets

Integration and infrastructure, because of their importance to the overall IT budget, are a key component of IT value. Managing this part usually requires some form of executive order, because the infrastructure is a basic ownership

Table 7.3 *HP focus areas for commitment to industry standards (Source: HP)*

Fabric-based interconnects	With industry standard interconnect technologies; new standards that are currently in development will be the driving force behind the convergence of storage networks, which primarily use Fibre Channel protocols today, with Ethernet networks. One example of such a standard is the Internet Small Computer Systems Interface (iSCSI) protocol that unites SCSI commands for storage and IP protocols for networking. These standard interconnects have the potential to bring you closer to your partners and customers for better relationships, service, and customer satisfaction.
Hardware architectures	The use of IA-32 and I-64 architectures throughout the computer industry has propelled innovation within server and storage design because of the focus on a single architecture. These standards have helped to bring the benefits of enormous computing power to companies of all sizes.
Software interfaces	The use of standard software interfaces, for example, XML (eXtensible Markup Language) and SOAP (Simple Object Access Protocol), provides a framework for intelligent middleware applications and management tools. Most importantly, unifying standard interfaces enable more modular software applications that will reduce the time-to-market for the construction of business-to-business applications and speed integration with future customer and vendor networks.

cost subsidized across the entire organization. In fact, this cost is the baseline of BCO. It is not sufficient to simply invent and implement standards; your standards must support optimization of your technology assets over the long term. One of the most effective ways to optimize a service is to focus and concentrate your efforts. You cannot keep focus if you try to do everything at once. You must choose your battles carefully, so that you do not overinvest in assets that do not provide the required customer value. Remember that the value of assets changes over time; many technology assets that were valuable yesterday will not be valuable tomorrow.

What assets should you concentrate on?

There are many standards for integration, but not all make sense for every organization. Your standards will depend on the needs of your organization, the needs of your industry, your partners, and customers. You cannot expect to support every standard; rather, you should focus on the ones that make most sense and support your organization.

Finding ways to improve productivity in daily tasks is one of the most important parts of managing the operational costs of the IT environment. The day-to-day costs of operations management are the most expensive part of IT. By focusing on consistent and constant optimization, you can use the efficiency gains to reduce your operations workload and costs.

Table 7.4 *Architecture improvement programs (Source: HP)*

HP IT Web Services Architecture	The HP IT Web services architecture site provides information, documents, contacts, and links describing work results, guidelines, and activities of cross-departmental efforts to define the future Web services-based architecture for application deployment, customization, integration, and development at HP.
Technology Watch	The Technology Watch site examines long-term technology trends, makes predictions as to which trends are the most viable, and analyzes the potential impact that such trends could have on the enterprise in the future (typically 2 to 5 years). The Technology Watch site also contains information about regularly scheduled audio conference forums on technology trend topics.
Sunset Applications Retirement	The Sunset Applications Retirement (Sunset) program seeks to identify applications deemed fit for obsolescence by the business. An outcome of the project is a plan for targeted application obsolescence and execution of that plan.

How can you optimize assets?

The HP IT department started with three architecture improvement programs designed to support long-term optimization of assets in support of normal IT life cycles, as shown in Table 7.4. (See Table 7.4)

7.3 Operations management

One of the greatest challenges of integrating architecture and process is that of integrating the information you need to maintain your infrastructure effectively. The management of computing operations requires you to integrate people, processes, and technology in ways that increase in complexity as new relationships are explored. To manage rapid changes in operations, you must find new ways to manage the relationships between IT operations without ending up with independent connections between components. The reason for this need is mathematical: Consider the number of potential connections between each of your operations. One of the most important roles of operations management is to reduce the complexity of these connections whenever possible. (See Figure 7.7)

7.3.1 Develop operational standards

IT operations work in many ways; some methods are well thought out, but many are not. How do you organize the tasks and coordinate each task with the appropriate resources? What processes are priorities, and how do you make sure you give these processes the deserved level of focused effort? One of the primary roles of the operational management of IT is to arrange the

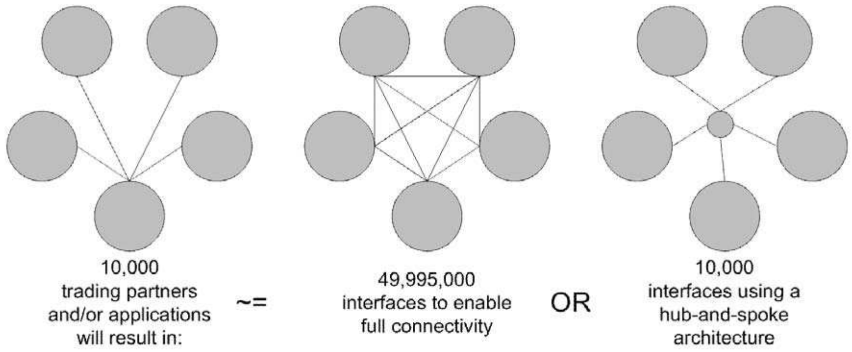


Figure 7.7 *The communications effects of integration (Source: HP)*

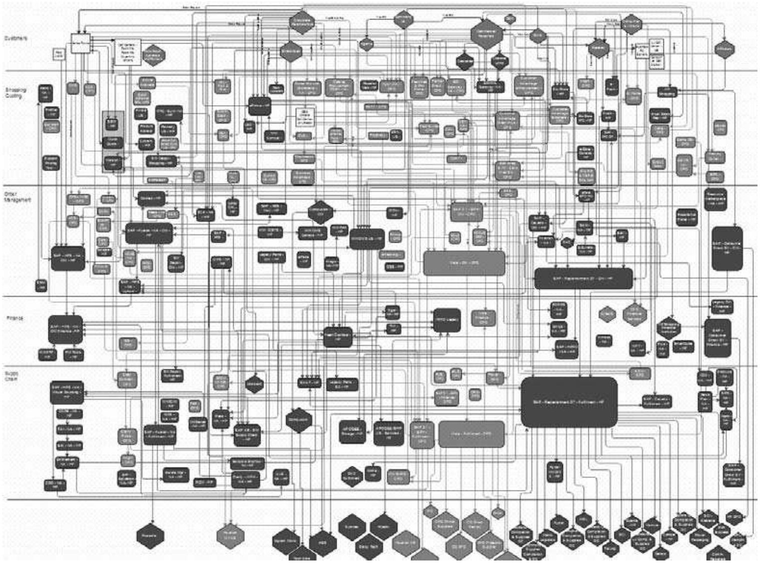


Figure 7.8
Megaprocess

day-to-day standard work processes of the IT organization and its resources. The complexity of these standards depends on your organization, but on completing an operational model most organizations find that even a bird's-eye view of the operational standards is much more complex than ever imagined. One concept to come out of the integration of HP and Compaq is called the "megaprocess." Put simply, the megaprocess is one map of the HP digital nervous system. The integration team put these together, all planned from "Day 1"—that is, whenever the merger closed. At first glance, the picture of "Day 1—Customer to Cash" is a convoluted mess: (See Figure 7.8)

The megaprocess is not, however, as confusing as the first glance (it is, however, confidential, so this book makes no effort to make the text visible or to explain it; rather, it is used to illustrate the complexity of a megaprocess). If you could look inside each of the processes, you will find that HP implemented them according to a much more palatable architecture called *process mapping*. The process map allows you to look at related processes in multiple layers of an organization. Consider the difference between sales responsibilities of a manager at level 0 and at level 1:

- The level 0 manager is responsible for selling products and services
- The level 1 manager assists the level 0 manager by developing a sales plan and quotes and then by performing direct sales tasks.

What do standards look like?

HP actually defines two different types of standards, called *standards* and *policies*. To help clarify the difference between policies and standards (as often both may exist for a particular topic), consider a side-by-side comparison. You will find that the primary difference between standards and policies is how HP implements them to meet its unique business needs. (See Table 7.5)

Table 7.5 *Policies and standards for password management*

Topic	**Password Policy/Standard**		
Summary	The HP password (policy/standard) establishes the minimum requirements for generating and managing passwords used by operating systems, database management systems, or applications on all systems owned by or operated on behalf of HP throughout the corporation. Passwords protect identifiers that have been issued to users or services for the purpose of obtaining access to a system or an application. This policy/standard addresses the use of passwords to protect identifiers used as credentials in authentication protocols. In this form of authentication a password is used as verification of claimed identity. A global policy/standard for password management is needed to ensure that a consistent level of security is maintained across all HP operating systems and applications.		
Scope	This policy/standard applies to all applications, operating systems, database management systems, network operating systems, and devices that require a user ID and password. Passwords issued for temporary IDs, password resets, and locked-out IDs, must also conform to this policy/standard. This applies to any password used as credentials in authentication protocols.		
Policy/Standard	**Policy**		**Standard**
	Protection of passwords		Password selections
	Password selections		Password lengths
	Password expiration		Password expiration
	Password transmission and storage		Account lockouts
	Account lockouts		
	Password uniqueness or minimum age of passwords		
	Password hacking		
	Password display, logging, and printing		
	Breach of password security		
Authority/Responsibility	Information security has the sole responsibility for maintaining this policy/standard.		
References	**Policy**	**Standard**	
	ISO 17799 9.2.3, 9.3.1	ISO 17799 9.2.3, 9.3.1	
	Password management policy	U.S. Department of Defense Password Management	

(*Continued*)

Table 7.5 *(Continued)*

		Guideline, CSC-STD-002-85, Library No. S-226,994 Password Management Standard Information security exception policy Information security exception standard
Constraints/Waivers	Requests for exceptions to this policy/standard are reviewed and approved or denied by the information security exception review team on a case-by-case basis according to the information security exception policy and standard.	
Implementation Plan	This policy takes effect immediately following approval.	This standard takes effect immediately following approval for all new implementations. Compliance of the existing environment is required within 90 days of the approval of this standard.
Compliance	**Policy**	
	It is essential that any violation of this policy be reported immediately to a supervisor, IT director or immediate staff, and/or HP's legal department so that an effective response can be undertaken. Violations will result in appropriate disciplinary actions up to and including dismissal.	

7.3.2 Review continuity plans

The operations group is responsible for maintaining the day-to-day continuity of the IT enterprise. Chapter 4 noted that risks relate to the organization's unique vision and strategy. In Chapter 9, you can learn more about identifying and measuring risks. How do you manage risks day to day, and what do you do about downtime and other impacts to customers? To do this, HP uses a process called business continuity planning (BCP). BCP addresses the following topics:

- Risk identification
- Risk quantification
- Risk mitigation

- Emergency response
- Crisis management
- Disaster recovery
- Exercising, monitoring, and improving

You must develop a similar process to manage your risks; it is not an affair for each individual technology project alone, it is an important part of continually maintaining your organization's overall IT health. As the organization changes, so change the risks. Make a conscious effort to mitigate risks and maintain continuous business. The process HP consultants use to manage risks consists of four phases: (See Table 7.6)

How will you manage risks to continuity?

Capt. Edward A. Murphy said, describing an issue in a project "designed to see how much sudden deceleration a person can stand in a crash," that "If anything can go wrong, it will."[2] Nowhere is this more true than in the IT world. Much like the original Murphy project, IT seems to be a significant cause of crashing businesses and other continuity problems. In the early 1990s, the primary cause of these problems was downtime. Toward the late 1990s, the problem had become Internet viruses. By the turn of the millennium, the problem was an economic bubble burst followed by an IT bloodletting and more than a few dotcom bankruptcies. The reality is, things do happen in the IT environment, and many of these things are not as easy to predict as we might like.

Take an integrated approach to mitigating your risks. Involve each owner of each component in this process. Manage not only the risks but also the relationships between risks in different components. Put simply, if the network goes down the continuity effects apply to much more than just the network maintenance department. Discontinuity can affect a lot of people, processes, and technology, so manage risk mitigation by finding the problems at the source. One of the best ways to manage risks is to make each IT component responsible for managing its own risk. By maintaining a standard model of your IT environment, you can assign component "risk owners" to take on this responsibility, and by making risk management a consistent process you can better ensure that each component is actually managing risks.

Similarly, not only the people, but also the components themselves must become capable of addressing pending continuity problems. If a database is

Table 7.6 *HP Consulting risk management process (Source: HP)*

1. Identify business needs	
Business continuity audit/risk analysis	**Business impact analysis**
Audit your current environment from a risk and recoverability perspective Ensure infrastructure is adequately protected from preventable risks, and that supporting processes and procedures are in place to effectively recover from service interruptions Identify areas of risk with recommendations for prevention and management	Identify business processes that are essential to the livelihood of your company Identify the tangible costs, such as lost revenue, market share, and penalties; and the intangible costs, such as reputation, image, and customer satisfaction that could result from downtime Identify the target recovery time objectives (RTOs) for critical processes
2. Design and architect	
Business continuity strategy definition	**Technical infrastructure design**
Provide strategies for various scenarios and durations of potential disasters Translate recovery time objectives and other continuity requirements into supporting continuity and recovery strategies	Provide a design or blueprint of the architecture that will ensure your recovery time objectives and business needs are met
3. Plan methodology	
Business continuity plan development	
Design and develop a business continuity plan tailored to your business requirements Outline the processes, procedures, and people necessary to recover and continue critical business processes after a service interruption or major catastrophe	
4. Rehearse and review	
Business continuity plan rehearsal	**Business continuity plan review**
Rehearse and critique your continuity and recovery plan Pre-plan your rehearsal by jointly developing test plans and objectives Record feedback during plan implementation	Assess the plan against industry best practices and your current business needs

slowing down, an automated notification should be sent to the individual responsible. If a server is experiencing network problems, or in some cases is simply measuring network slowdowns, a similar notification must be sent. If an incident—a measurable event—persists, it becomes a problem. In any case, however, these situations each affect a unique set of stakeholders. The key is to find and measure the impacts.

7.3.3 Measure the cost of impacts

The reality is, things happen in the IT department; you cannot escape this fact. Downtime because of bad equipment, bad software, viruses, and poorly scheduled IT changes are not new to the Internet era, but the impact on business has certainly amplified. Bad business decisions are not new either, but again, the use of email on business litigation magnifies the impact of such decisions. In fact, one of the harshest realities of the Internet for the technology worker is that now that technology is so important to business and the economy, the cost of downtime has increased dramatically. What is the real cost of downtime? Does it include work to restore service; does it include user costs such as lost business productivity? Where do we draw the lines? One of the greatest benefits and challenges of managing IT services with a model is that you can build very complex methods of assigning these types of costs. If you chose to implement service-level agreements at the level of individual IT components, your model for measuring impacts to each line-of-business customer might look something like Figure 7.9, with each service-level agreement related to a particular business case at the intersection of components and internal line-of-business customers (LOBs). (See Figure 7.9)

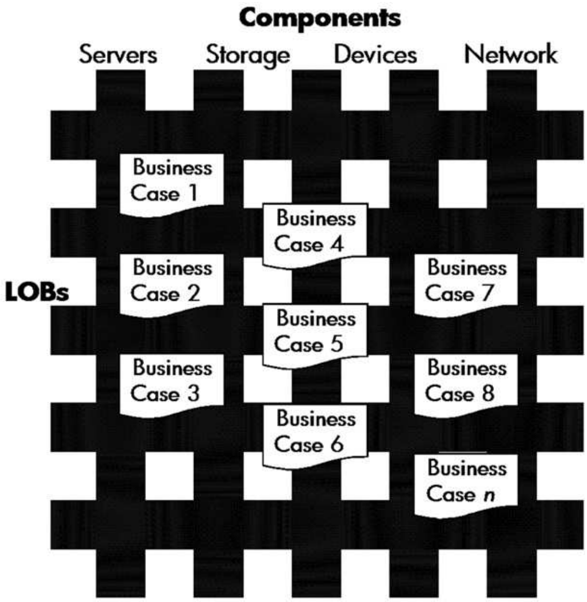


Figure 7.9
IT infrastructure value capture tapestry

The fact is, technology affects each of us in a different way. The impact of a longstanding IT problem to a user is definitely not the same as the impact to a shareholder—but is one more important than the other? The answer is that you must understand and manage the relationship between customers and impacts. If your email is down, that is much more likely to affect the CEO than the delivery driver—until you give your delivery driver a Blackberry or some other messaging device. We must develop advanced methods of measuring the potential and realized costs of downtime.

Where are the costs of downtime and other IT impacts?

In an HP Labs report, Maher Rahmouni observed,

Detecting and diagnosing a fault is challenging in general because

- A single fault can give rise to a multitude of symptoms
- Each symptom viewed in isolation may be explained by a wide range of faults. It is difficult for a human operator to extract the correct root cause from all outstanding alarms in (near) real time.

This indeterminacy results in operator inefficiency and slow recovery from failures. An ideal alarm correlation system would uniquely identify any fault (or set of faults) in the network unambiguously and in real time. This ideal has remained elusive, for several reasons:

- The number of possible faults is extremely large.
- The system model that describes how a fault results in a set of symptoms is difficult to describe accurately and exhaustively. This system model depends on the physical properties of network elements, the software running in different components of the network, interdependencies among subsystems, and so forth.
- The management system itself can introduce noise and delay into the alarm stream. For example, an alarm storm caused by a severe fault can overload the management system, leading to dropped and delayed alarms. Also, the fact that in IP networks, management information is usually carried in-band (by the managed network itself) is problematic, because some faults (such as a network partition) may actually result in the suppression of some alarms.

The biggest challenge comes if you decide to refund and/or charge stakeholders for component incidents. Then you are responsible not only for spending money to find the malfunctioning components and repair them but also for spending additional money to manage the finances of refunding customers and charging vendors. You cannot underestimate the challenge of this analysis in a large IT environment. When a user calls the help desk to report a "problem with an email," who or what is to blame for the inconvenience: the email client, the desktop operating system, the network, the server, or the Internet? Are you prepared to take responsibility for the health of Internet services, when the consensus in January 2003 (right after the latest Microsoft-based Internet worm ate up 20% of the available Internet bandwidth for days, spreading in a manner of minutes) is that the Internet itself is critically insecure for maintaining business continuity? Will your customers let you make this distinction in your responsibilities if you are asking them to pay for services that stopped working for minutes, hours, or days? Do you have the appropriate reciprocal agreements to recover these costs with your Internet service providers (ISPs) and other telecommunications equipment and service vendors?

7.4 Operations support

The first job of many IT workers is in the support department. Countless thousands of us began our professional IT careers at a help desk. Often unnoticed and unappreciated, the help desk is the "one stop" for technology stakeholders with incidents, problems, issues, concerns, and often gripes and general bitching. If you have ever worked on a help desk, you certainly know a few key rules:

- The customer is always right; even when the customer is a blooming idiot (or simply an innocent) who truly believes the CD-ROM tray is a coffee cup holder.
- Sometimes the customer really is right.
- The customer is always upset—and has called plenty of people before you (who are still trying to fix the problem).
- The customer is always in a hurry—and would not have called otherwise.
- If you don't know the answer at the start of a support call, the cause is most often to be found in the one place you would have never thought (or had never been trained) to look.
- You never have enough information to make every customer happy.

The result of these rules for the help desk is a world full of procedures for average hold time, mean-time-to-resolution, and escalation, and other very painful-to-remember policies that change every few hours. This is not an exaggeration; the rapid rate of technology innovation has demanded more from the help desk and other operational support workers than arguably any other group—except maybe their managers. The rate of technology innovation increased significantly during the Internet boom and has continued high since the dotcom bust. Intel and AMD (Advanced Micro Devices) are still locked in processor performance wars, and hard drives keep getting bigger. Even though many organizations cannot fill a 1GB network connection, now 10GB networks are available. The growth is not stopping, and the investment has not either. And all this innovation, to the help desk, is often noise.

The question is, how can we organize operations support to support the new IT operations? To make information available is not sufficient if the information is constantly stale or poorly organized. The benefit of integrating your architecture and processes—your entire IT department—is that you will be able to find the right information in the right place. If you cannot find it, you should be able to notify someone, and someone should be responsible for remedying the situation. Similarly, if someone leaves, you must retain his or her knowledge.

To be successful, you must

- Respond to incidents
- Resolve problems
- Develop service-level reporting standards
- Address issues and concerns

In Appendix B, you will find a list of operations that HP considers critical to the success of HP IT infrastructure. This list can help you identify and organize potential impacts to your IT infrastructure. This list is not static or universal; you must change it to meet the needs of your organization. Also consider adding your own operations for the rest of the solutions in the HP architecture: business objectives, business processes, information, and applications.

7.4.1 Respond to incidents and resolve problems

The primary concerns of support resources are access to information about customers and IT components. To be valuable, this information must be current. Here are some possible strategies to address the rules listed earlier: (See Table 7.7)

Even if you are not paying your customers for downtime, many organizations require reporting of IT incidents. If nothing else, incident and problem history is invaluable when supporting customers. With the

Table 7.7 *Customer service strategy*

1. The customer is always right.	Find methods to provide answers to frequently asked questions (FAQs) *before* customers call the help desk.
2. The customer is always upset.	Provide the support resources with support call history and/or customer relationship management software to satisfy the right customers.
3. The customer is always in a hurry.	Capture knowledge for each IT component in shared knowledge databases, document management systems, discussion web sites, etc.
4. If you do not know the answer . . .	Provide clear escalation procedures, access to automated component support information, access to testing tools, etc.
5. You never have enough information.	Integrate information and processes in Web portals to identify common incidents and problems quickly.

appropriate model for IT cost management in place, assign impact costs to the appropriate components, performing root cause analysis *to find, fix, and charge the original source of a problem*. These observations are critical to understanding the challenge of managing incident and problem costs. It is urgent for you to be able to find and repair "root causes," but the vast complexity of the typical large IT infrastructure frustrates this process. More often than not, the answer is to restart a system or process—further frustrating the process by introducing new variables in the fault detection process. The response to this is new sets of software tools for performing real-time root cause analysis.

How do you integrate support operations?

In an HP Labs article, Maher Rahmouni presents this solution:

> *There are several elements that [have] to be part of the system to qualify the product as a true root cause analysis tool:*
>
> - The object data store, *arguably the richest area of root-cause-software innovation, where information relevant to managed entities (network devices, applications, and so on) is stored. . . .*
> - Topology, *which helps you understand how networked, system, or application elements interrelate. Network topology has become a hotbed of contention, because different vendors have different implementations for layer-2, layer-3, and VLAN (Virtual Local Area Network) capabilities. Knowing where problems are in the*

networked environment is no longer simply a matter of downstream alarm suppression.

- Event/alarm correlation intelligence. *Root-cause analysis requires a way to correlate across events, topological change, and other changes within and among devices and applications. Here, the answer is usually an algorithm or set of algorithms.*
- Presentation and context, *which range from simple notification to a full set of service-level solutions. What about the reporting of real time alarms and alerts? These need to be processed on a near real time basis. The data needs to be disseminated as fast as possible to the concerned parties in a meaningful manner.*

ITSM defines the two most important processes you must integrate with your architecture: incident management and problem management. These processes, based on the ITIL best practices, are key to managing IT. Incident

Table 7.8 *Incident and problem management activities (Source: HP)*

Service	Quality Control Activities	Process Delivery Activities
Incident management	Establish help desk structure Establish incident control system Develop management reports Perform continuous process improvement	Accept calls Log incidents Categorizeincidents Prioritize incidents Isolate incidents Escalate incidents (within the process and/or to management) Track incident progress Resolve incidents Notify customers Close incidents
Problem management	Establish problem/known error control system Setup and maintain support contacts Establish preventive maintenance procedures Establish known error verification facilities Establish supplier support interfaces Develop management reports Perform continuous process improvement	Analyze incident trends Log problem Identify root cause Track problem progress Verify known errors Control known errors Resolve problems Close problems/known errors

management is the part of your IT support responsible for reactive processes, often called the help desk or service desk. Problem management is the part of your IT support responsible for proactive processes. Problem management is the second and third levels of your service escalation procedures; they are the groups responsible for implementing proactive fixes and fixing long-term problems.

7.4.2 Develop service-level reporting standards

It is important to provide customers with information needed to make the right service choices. You must develop standards for reporting on compliance to service-level agreements. Each report should detail promised versus delivered levels of service for each individual component. Remember that reporting requires work itself, although it is generally a requirement of most services. Try not to do more work than you need to do; you may want to take the cell phone company route here and ask your customers what level of detail they need; it makes no sense to provide more information than your customers actually require (and will pay for).

7.4.3 Address issues and concerns

Do not forget that one of the most important parts of managing operations is how you perform those operations in the first place. You must provide the ability for customers to express their issues and concerns, but you cannot do this without understanding the importance of individual issues and concerns. These encounters are incidents like any other; respond to the individual concerns as they come up, and maintain a list of common issues that you can manage as problems. The next chapters show how to focus operational efforts on taking the right scope of action to address the issues and concerns of individually managed components.

7.5 Conclusion

Integration is a road, not a destination. Do not assume that everything will continue to sick together just because you have created an adaptive infrastructure with operational standards and efficient support. Careful monitoring of your operational results will provide the best opportunities for long-term user satisfaction. Operations are what the user sees on a day-to-day basis, and the user's perception of operations is quite often the deciding factor in ongoing investments.

Another important factor that many organizations seem to ignore is the relationship between operations and the decision-making impacts caused by the rest of the IT organization. Many of the most successful architects came from the operations group, but how many of these same architects remember the pains of receiving a phone call from a business customer with an urgent problem or change request? How many of the same architects routinely consider the long-term operational impact of their design decisions? To ensure coherent infrastructre, remember that the engineers and support resoutces should communicate openly, frequently, and work together for the benefit of the customer. To help open these lines of communication, you may find success in first changing the way that the organization makes decisions. Put in a few roadblocks, and force the stakeholders to consider and document the needs of their peers. It is important to consider that when delivering IT services, no single decision-maker or customer can be the "most important" or have the "final word;" decisions must be made as a consensus, and if a consensus can not be found, then the scope of the decision should change out of necessity.

Creating a cohesive infrastructure in an IT services model requires you to consider the needs of all stakeholders, and to prioritize the decisions according to objective requirements. Consensus is important; without it, you cannot expect your stakeholders to work together for the benefit of the organization. Development, like integration, is not a destination. Once you deploy the technology solutions, consider the ongoing effects of change and decision-making on the solutions you create.

References

1. Nedstam, J. (2002), *Changes to Processes and Architectures; Suggested, Implemented and Analyzed from a Project Viewpoint* (Lund Institute of Technology, Department of Communication Systems Lund University) p. 18.

2. AFFTC History Office (2003), available on the Web at www.edwards.af.mil/history/docs_html/tidbits/murphy's_law.html.

8

Deployment: Questioning Change and Decision Making

Life is a process of becoming, a combination of states we have to go through. Where people fail is that they wish to elect a state and remain in it. This is a kind of death. (Anais Nin)

8.1 Introduction

Whether you are a CIO changing strategies or a PC changing operating systems, you must execute each individual change responsibly. Although at the lowest level of the organization, the component has a very important job; it is responsible for understanding the impact of changes throughout the organization. To be successful, your resources must understand the impact of changes. Change and configuration management lie at the core of ITSM because they require you to ask a very important question every time you make a decision: What are the impacts? In this chapter, you can learn about how to keep track of standard component configurations to ensure effective component delivery. Once you deliver each component, you will want to manage future changes using best practices for component assurance. Finally, in the section on component monitoring you can learn how to tie everything together by reporting on the costs, performance, and efficiency of each individual IT component.

What are the potential impacts of each IT decision?

When the CIO sets a strategy, it may affect the operation of every PC in the enterprise—including the PC about to get an upgrade. It may be less expensive to just erase the PC contents and start with a fresh computer operating system.

But if the CIO set a strategy focused on customer satisfaction, does it really make sense to save time during an upgrade by not backing up all the user's files first? Change is useful—as long as it moves you forward. Each decision has risks, and some of these risks, if not mitigated, can result in some much-undesired outcomes. You are responsible for making decisions every day, from service packs to vendor selections. Each of these decisions has a number of potential impacts, depending on how your particular organization manages technology. It also depends on the role of the particular technology. It is crucial for you to know the relationships between each decision and its components. No matter your role in an organization, each IT decision has potential impacts beyond you:

- *Decisions may have widespread consequences*. If you are an IT executive, your decisions may change many layers of your solutions; if you maintain the servers or other IT components, remember that you are responsible for enforcing the decisions made by every level of the IT organization. In fact, there is an important relationship between CIO and IT staff: The CIO's strategy, in many organizations, is close to useless when a server goes down. It is one thing to promise your stakeholders 100% availability, scalability, or security—it is quite another to provide this level of service for every information technology component. Priorities must be set, and stakeholders must buy into decisions.
- *Decisions may require long-term investment*. Some IT components, like any other technological innovation, need regular repair and maintenance. PCs are infamous for this trait. Other components, such as networks and email servers, are infrastructure: They are so important that their failure will measurably impact many users and other stakeholders. Organizations have responded with new hardware, new software, and operating systems, new processes for deployment and maintenance, and many other strategies—yet none seems to address the key issue: making sure we understand what we require of each PC in the first place. Evaluate the long-term cost effects of changes.
- *Decisions may not affect all stakeholders equally*. If you find out which of IT services your end users actually use, you will find little accord. Depending on role and position, there are two classes of users: those who use a little of everything, and those who use a lot of one thing. If you ask your IT financial wizard how you license software, however, you will find we often manage purchases according to "least common denominator" (for example, "if we buy 1000 copies for our 1000 users, we will certainly have enough" as opposed to "How many users will actually use this software component?"). Understand how your decision affects *each* stakeholder.

- *Decisions may be parts of standard processes*. Some changes are not as valuable to manage as others. For example, if a change will only affect a few users and those are the users requesting the change, it will likely cost more to use a formal change management process than to simply make the change. Implement each component in a way that lets you understand the scope of its potential effects to empower decisions made by the right people at the right time. This takes practice.

8.2 Component delivery assurance

At the core of the IT organization is a set of components. Each component is an individual configured set of IT resources; for instance, you may consider “a server with an operating system acting as a security provider by performing user authentication” as an individual component. Although the concept of component management may be straightforward enough, the actual implementation of this practice is challenging indeed. It requires you to be very detailed in decision making and assigning accountability. Strike a balance between democracy and bureaucracy, flexibility and efficiency, architecture and process. This relationship is apparent not only during integration, but also when managing integrated components. To enforce standards, it is important to perform two critical ITSM functions: configuration management and change management.

How do you assure delivery of IT services?

HP calls the ITSM core *service delivery assurance*. It is a rather easy concept, explained well by HP’s internal Change Management Team:

> *Change Management supports the HP infrastructure environment by providing a service that monitors, tracks, communicates, schedules and coordinates activities that have the potential of impacting the production environment.*

The primary function of Change Management is to analyze all change requests, taking into consideration the benefits to be gained from the change, while keeping the risk of impact to Services and Operations (OLAs, SLAs) to a minimum.

This requires a balanced approach in applying formality to changes without bureaucracy or undue delay. Change Management ensures that changes are planned, coordinated and clearly communicated. Consequently, no change can be "cut over" to production except through the Change Management process.

The change management representatives on global, regional and subsidiary levels were given the authority by the respective Managers to determine whether any changes can be implemented and whether a change should be withdrawn at any stage prior to acceptance sign-off.

To document the changes being made, the Change Management Team compiles data from change requests, which then provides a consistent management reporting structure for service-related matters. (Source: HP)

How can you assure delivery of IT components?

Like living cells, IT components grow and change according to the surrounding circumstances. Computer networks, for example, have become more important, therefore larger, and more complicated to manage. Also as in biology, each IT "cell" can reproduce or divide; consider what happened in the 1980s when personal computer use grew and business departments bought PCs, or what happened in the 1990s when the electronic network became bigger than the computers it connects. One often manages networks and computers differently; they serve different functions. When delivering components to support integrated services, however, common goals govern management. At the core of these goals is the individual change—often the result of decisions made by human components in the IT organization. How you make each decision, and what happens after, are arguably the most important aspect of providing valuable IT services. Components relate to each other in a spider web-like fashion, and breaking a critical strand can bring down the entire web.

8.2.1 Enforce configuration management standards

Every component in an IT services environment must prove its own adherence to standards; to enforce standard configuration, you record the actual implementation of the component in a configuration management database (CMDB). The CMDB is the central repository for information about every individual component in the IT organization; it is also, therefore, the central schema for everything IT can do to a component. Do not underestimate the importance of the CMDB; to make long-term assessments of the effects of IT decision making, you must maintain standards for configuration management.

The team responsible for HP's OpenView ITSM Configuration Management software explains some of the important features of configuration management:

- Specify the versions of configuration items in use in the IT infrastructure.
- Provide information on the status of configuration items, for example, "scheduled for live use," "in live use," "scheduled for upgrade," and so forth.
- Determine who owns each item (the individual with primary responsibility for it).
- Determine the relationships between items (components, connections, and so on).
- Control changes to the configuration items by making sure changes are made only with the agreement of the appropriate authorities.
- Audit the IT infrastructure to make sure it contains only authorized configuration items.

How will you enhance and optimize the architecture?

The task of enhancing and optimizing the architecture will require increasing amounts of information as the complexity of the IT environment grows with each generation. Develop models to explain your standards for storing this information and maintaining these standards. Develop a component responsible for periodic maintenance of the CMDB, and keep it clean enough to manage changes along the way.

8.2.2 Enforce change management standards

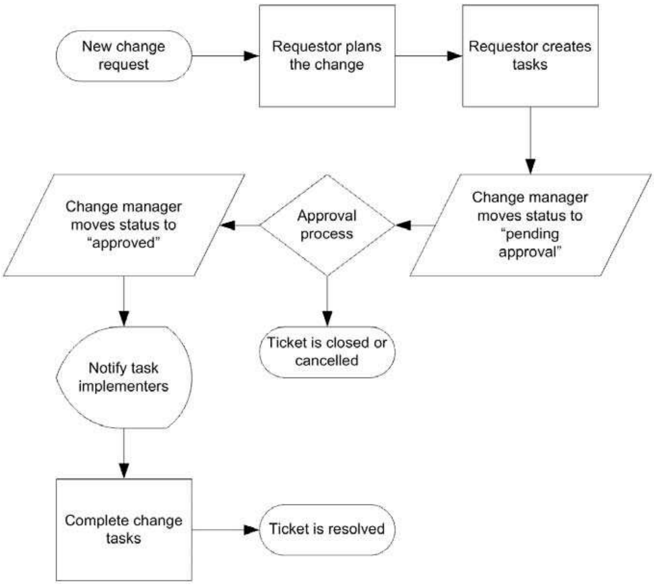
To make the right decisions regarding change, it is important to have the right information when you need it. To implement changes effectively, you must understand the scope and potential impact of each change. The RAPID decision-making model lets you view and manage 360 degrees of your organization and its technology. In the center of the model is your component—the thing you need to change. The component is the center of the decision for a few key reasons:

- *Changing components requires stakeholder buy-in*. Change management requires you to involve all the appropriate decision makers.
- *Changing components may affect long-term costs*. You must evaluate changes to ongoing support costs of each change.

- *Changing any component may change other components*. Changes that affect other components (including solutions, services, processes, and operations) are subject to the rules of the other components.
- *Changes may affect the support requirements of a component.* (For example, you may promote an application to become infrastructure, or demote an infrastructure component to an application.)
- *Changing components requires change management.* ITSM requires that the change management group be responsible for any changes outside of a project. (See Figure 8.1)

One of the most important aspects of a service-level agreement is that you are making an agreement—a contract—with your customers to assure appropriate levels of service. It is not sufficient to provide assurances for the services; you must assure the quality of every component provided by managing changes and providing maintenance. Uncontrolled changes, especially during the typical workday, can have disastrous effects on user satisfaction, system availability, and other key metrics of the IT department. In an IT services environment, any substantial effects on metrics are noticeable—quite often on the bottom line. Most of IT will not be happy about having the additional work required by change management, they do not want "big brother" or anyone else looking over their shoulders as changes occur,

Figure 8.1 *Change management process (Source: HP)*



and in general are skeptical of change in work processes. Regardless of these concerns, you must not only implement but also enforce change management. It is imperative to control changes to your production IT environment. Often the smallest changes can have the biggest impact—and be the most difficult to find. Any stakeholder can request component changes, and you must address his or her needs in a manageable and supportable way. Change management standards describe the process for controlling changes (the effect of decisions) to the IT environment.

What level of change management is most effective?

The last is one of the most critical points of ITSM: You must manage changes to configuration carefully. ITSM does not define the size, scope, or shape of a component; this task ITSM leaves up to the organization. The general rule is that you should size each component in a CMDB to the level of "independent change" or "maximum control with minimum records."[1]

Finding the right components to include and pinpointing the appropriate level of detail is the ultimate challenge, but generally you can start with an asset inventory and justify your way simpler or more complicated. If you develop software internally, you will need to account for changes to these components as well. With a model, you can see the effects of architecture and process on each individual component to help you make decisions. A model can help you

- Understand the relationships
- Estimate the impacts of each change before making changes
- Ensure that resources do not make untracked changes

HP's Change Management group defines five principles of change management. The following principles are the basis for the Global Change Management Standards and Procedures. (See Table 8.1)

How do you measure the effects of change?

In measuring the effects of change, be specific. Customers will not accept classifying all computing dinosaurs as all being "big and green" when they are in truth different species of theropods. For this reason, to be effective in IT services you must first learn to classify your IT service investments and identify their similarities. Once you have done so, you will find it much easier to see the trends inherent in the growth of any technology—to see the forest despite the trees. Remember that the classifications will change over time. Just ask the Brontosaurus; known to children for decades, it is a species now declassified and relegated to the history of science.

Table 8.1 *Change management (Source: HP)*

Cut over/under Information Services Operations control	Cut over is done under GBS control, activated by the change manager's approval of the change and its implementation plan, and the approval of the appropriate approving manager. An approving manager can delegate the approval function as long as this does not conflict with the change management standard/policy.
Project manager's responsibilities	Changes are to be submitted by the project manager/change owner in charge of the change on the mandatory quest change request form. The project manager/change owner takes responsibility for ensuring all tasks required to implement the change have been identified and are acted on correctly. He or she is also responsible for ensuring users' understanding for the necessity of the change, representing customers in the change management process, and coordinating changes with customers.
Changes are approved, rescheduled, or canceled by the change management team members	Changes and schedules are accepted and approved for implementation by approvers and the change management team. This is based on risk and benefits analysis of the change. In case of conflicts, changes may be rescheduled or canceled on the basis of unacceptable risk to the company's shared systems environment after consultation of all involved parties.
Change request approval deadlocks and appeals	As the change manager has the authority to approve or cancel a change, he/she must not be subject to pressure from any project manager/change owner. In case of disagreement, the project manager/change owner needs to involve the respective level of GBS management for final decision.
Change management planning/review meetings	Scope, scheduling, and completion of all changes are reviewed periodically. Change management is responsible for facilitating meetings as prescribed in the global change management procedure document. The project managers, requesters, and implementation managers are committed to contribute to the planning and review process as required.

What are some examples of change management?

HP's OpenView ITSM change management software provides a good example of some of the important features of managing change: (See Table 8.2)

8.2.3 Maintain systems and processes

Do not forget: Machines break down. Update the processes as the structure changes. Put each process on the right calendar.

Table 8.2 *Change management features*

Initiation	Initiates the change
Tracking	Tracks the changes and information throughout the life cycle
Authorization	Schedules change advisory board meetings and sets up approval cycles
Risk and impact assessment	Identifies the impact of the change to the overall IT service provision
Scheduling	Schedules change timeframes with minimum impact on overall IT service provision
Building	Identifies a backup plan, any dependencies on the change, and so forth
Coordination	Coordinates the work related to changes
Verification	Verifies if the change was implemented as planned
Review	Runs reports and aggregates information for decision support

8.3 Component monitoring

One of the most useful applications of the RAPID methodology is to create a model for implementation of an integrated CMDB and definitive software library (DSL). (See Figure 8.2)

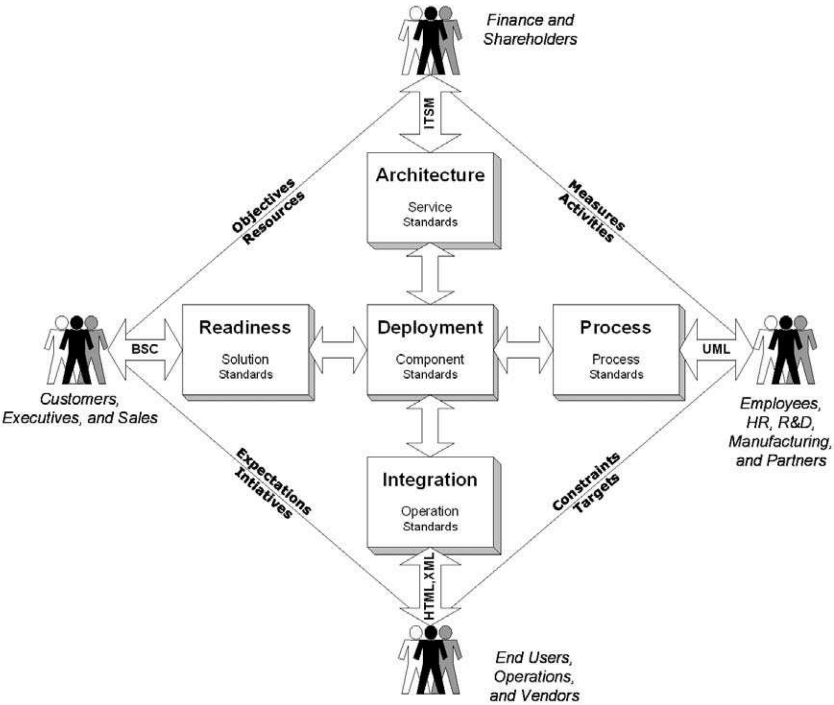


Figure 8.2 *RAPID decision-making model*

The advantage to a configuration database is to provide a centralized location for storage and auditing of

- System costs
- System performance
- Service efficiency reporting standards

8.4 Conclusion

The most important thing to remember about decision-making is that you need to work together as a team with each component's unique set of stakeholders. Decisions, especially changes, tend to have far-reaching effects that often seem unpredictable. You need to understand the impact of changes, especially infrastructure changes, in order to mitigate risk. You also need information from stakeholders in order to understand the impact of a change to customers; for example, it is a poor practice to schedule a major change right before a financial reporting period. In order to make good decisions, you must have a good record of the "seams" between each component.

If you manage change and configuration at the discrete level required by ITIL, ITSM, and RAPID, you will find that the number of items that you consider components will compound the confusion in finding the right decisions. Do not let change and configuration management become bureaucracy, make decisions "just in time" with the right stakeholders for each component. You cannot assume that a single "change management group" is appropriate for the entire IT department. The rest of this book includes some high-level recommendations about how to build and track all kinds of components: solutions, services, processes, operations, and individual discrete components. You must remember that the level of detail in this book is not sufficient; you must add your own point-of-view in order to implement an effective model for service management. This requires patience, teamwork, and dedicated effort; your team cannot do this all at once. Work together to find the right information you need to make good decisions.

Reference

1. Macfarlane and C. Rudd (2001), Configuration management, in C. Cambray (ed.), *IT Service Management Version 2* (Reading, UK: itSMF Ltd), Chapter 8, pp. 23–24.

9

Business Cost of Ownership

Science is not an encyclopedic body of knowledge about the universe. Instead, it represents a process *for proposing and refining theoretical explanations about the world that are subject to further testing and refinement. No. 92–102. Argued March 30, 1993—Decided June 28, 1993) [emphasis in original] Daubert v. Merrell Dow Pharmaceuticals (92–102), 509 U.S. 579 (1993)*

9.1 Introduction

No matter where you look, the information and communications technology industry (ICT or IT) is talking about value. Ask HP, Microsoft, IBM, Gartner, Oracle, Sun, or even your government. Everyone is talking about the importance of IT value and desperately trying to find such value. It is great that you can find so many valuable IT solutions, but how do you go about selecting one for your business? Whether you are a help desk agent or a CIO, you had better be concerned with value. After Y2K and after the dotcom bubble burst, value is really the only thing that counts. Technology vendors, customers, and shareholders alike now know we must focus not on finding either the most functional or the most efficient solutions but rather on the solutions that are valuable according to the customer's current definition of business needs. To help the reader prepare for a future in value-oriented information technology, this chapter discusses one important component of business value: methods for measuring BCO, the business costs of operating IT. (See Figure 9.1)

IT already has many cost and value models. BCO does not argue that the other models are wrong (in fact, they have their place in this book), but

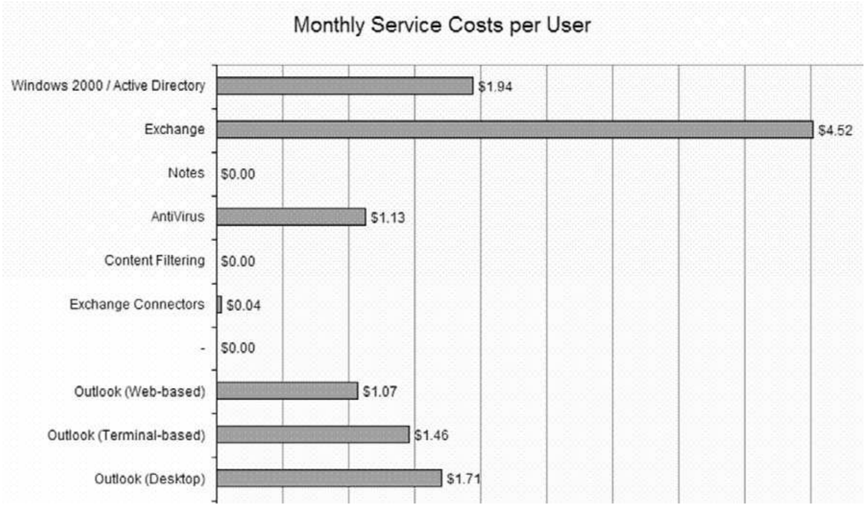


Figure 9.1
BCO service costs for case study organization

rather that the current models are much too specific and too complicated to be universal. BCO is about how you can measure the actual costs of providing your own IT services today—and how you can compare your costs to similar costs of your partners—with less focus on any complicated measurements of scalability, downtime, or "futz factor." The *American Heritage Dictionary* defines "futz" as "to waste time on frivolities; fool." BCO is a cost for a specific service, as in Figure 9.1. BCO is what you can see; it is what happens to TCO (total cost of ownership) when you eliminate the guesswork. Our team at Rainier, Compaq, and then HP created BCO out of the same frustration many organizations feel when they try to find the best places to invest IT dollars: There is simply no standard. We created BCO because we felt our customers deserved a model that would help them see not only what IT dollars could bring them today but also what IT investments would mean to them in the future.

BCO is the basic (or business) cost of ownership for an IT solution. You measure it by dividing the real dollars that business invests in IT (the IT budget) accurately among the services IT provides in return. BCO is the highest-level view of IT as accurately as you can build it today. The reason why nobody can agree on a model for IT value is easy to understand: Every organization has different needs according to the needs of its stakeholders. This chapter introduces IT financial modeling, with a specific focus on how BCO can help simplify both the provider and customer sides of IT services.

9.1.1 IT financial models

When considering financial analysis and management of information technology, one must discuss a few primary models. These models can be classified into two major groups: cost models and value models. Cost models are concerned with the monthly cost to support a user of a given service, such as TCO, the total cost of ownership of a certain segment of information or communications technology. Value models, in contrast, find a basis in traditional economic analysis, including the calculation of a ROI or return on investment.

9.2 Changing needs in IT economics

The first CIO to ask me, "What's wrong with TCO that we need BCO?" was a technology leader responsible not for a Fortune 500 company or a dot-goner, but for a new breed of government. His boss, the state governor, had charged him with the nightmare responsibility of consolidating the majority of technology resources and functions within their state. Within the state government, as in many other organizations, the natural response was to immediately carve out a few big pieces and ask the largest vendors for their TCO estimates. When we first learned of these pieces, the "other" vendor had already responded with a proposal for a large mainframe and some "very rough numbers" for consolidating email services. By the time our team first presented our vision to this CIO, we had only begun to question fully the concept of using these disconnected solutions to address integration needs. Our Compaq (at the time) account team tried to address the challenge in a naive and unique way: We tried to put it together in our minds and add it all up.

9.2.1 Technology of highest importance

In the fall of 2001, while working with the IT staff supporting the state patrol agency, I first heard rumors about an important upcoming project. A few of the people I was working with were concerned because the state's central IT department was developing standards to mandate a single active directory forest for the entire state, containing every agency. (To those without a background in Microsoft's new enterprise computing directory, this act is akin to each state agency handing the state IT department the administrator password and keys to their offices.) By January 2002, we found that the state's governor and CIO had decided in the midst of chaos to consolidate PC servers and services for all administrative agencies—about 20,000 state workers—including servers, storage, email, data warehouse, and Web hosting.

Their vision was to build shared technology services that quickly would grow to meet the needs of many state employees and other state customers. The state IT department would resell technology services to agencies with competitive features and prices. Who would have guessed that the statewide active directory standard (an obscure technical standard in any case) foreshadowed a wholesale desire within state government that would become clear shortly after New Year's? It is incredible that a CIO and a governor would be at all concerned about counting servers at such a pivotal moment in our recent history. Perhaps, in retrospect, it is actually no surprise they chose these important tasks. In a research note at the same time, Gartner Group predicted that

> *In 2002, more than 70 percent of governments will establish or strengthen central IS (Information Systems) organizations to coordinate e-government developments at the national or regional level . . . [E]-government spending will shift from front-end to back-end activities as a consequence of assessments and prioritization of ongoing initiatives driven by centralization.*[2]

These challenges are not unique to state government, or even to IT. In fact, if you are even "somewhat" involved with a technology consolidation project in your own organization, it is critical to consider your own answers to the challenges this CIO posed to our team, the competitors, and this state:

- How do we identify and set priorities on the IT needs of our organization and its customers?
- How do we invest scarce resources in IT solutions that will provide the greatest value?
- How do we build support for these solutions in administrators, users, and other influential stakeholders?
- How do we build solutions that will grow to meet our uncertain future needs?
- How do we measure the results in a factual or even scientific manner?

BCO is not a simple idea, but it is the answer to the final and arguably most important question posed: How do we measure success?

Why BCO?

The answer to the question "Why BCO?" is simple: Unlike TCO, you can measure BCO. In fact, that is all you need to remember to start analyzing BCO—look for what you can measure. BCO is about using metrics—measurements—to evaluate, design, implement, and maintain your systems. For IT professionals, the following list of common metrics should look familiar:

- How many seconds of processor time does the task use?
- How many bytes of disk storage does the user need?
- How many bytes of network information does the server exchange?
- How long does the backup process take?
- How many engineers and operators will it take to support this service?
- How long do tasks wait for users?

That familiarity with the terms is also the problem. Metrics are easy to find—but not easy to put priorities on, aggregate, predict, agree on, and use. The state's CIO needed to evaluate, consider, and address at least the following variables:

- He is charged with supporting 250 different categories of agency, branch, department, board, and other state divisions. In the future, legislation or necessity may increase these responsibilities to include counties, cities, other forms of municipality, schools, universities, and every resident of the state. From online driver licensing to counterterrorism to state tax records, the CIO office may need someday to support it all. However, if the state waits too long to compete, it may lose the opportunity to compete with external vendors.
- In general, each state agency is responsible for supporting some component of its information technology. In the past, rapidly changing needs have required local support of items such as PCs and common desktop applications. In larger agencies, full-time development staffs create and purchase line-of-business (LOB) applications for processing transactions such as licensing, taxation, registration, and information. Purchasing, and even contracts, with IT vendors had decentralized throughout the state.
- The state's IT department does not control the Internet infrastructure. An aging centralized fiber-optic Internet backbone is regressing—moving from innovative to restrictive—while needing updates to bring both technology and architecture current. The department controlling the state Internet infrastructure is subject to frequent and intense

scrutiny by other agencies and is under pressure to compete with larger telecommunications vendors.

- From the CIO's desk, much of the state's IT infrastructure is unknown and undocumented. The administrative branches, for which the governor and CIO are responsible, consist of 14,000 users and almost 50 independent IT departments. For example, separate email products supported by these departments include Microsoft Exchange, IBM/Lotus Notes, Novell GroupWise, and a variety of mainframe and midrange host mail services. The only electronic directory connecting these systems to provide a semicomplete state directory was aging rapidly and had fallen far out of the mainstream.
- Nervous state employees, legislators, vendors, executives, and constituents are very wary of consolidation. Employees worry about losing jobs, about supporters losing confidence, about customers becoming unhappy and long-term costs of the "solution" skyrocketing beyond any expectations.

All these challenges were still trivial compared to the CIO's biggest challenge: finding a way to authorize and pay for the required improvements. His desired model is to create a "shared services" support structure in which the state IT department would become not only a revenue generator but also a more efficient competitor in the IT services market. To realize this vision, systems and staff supporting the current systems need drastic change. To achieve the required economy of scale, the customer base for the state IT department would have to grow quickly—without previously erected legislative and procedural barriers.

Why would you want to know less?

Part of the reason why our knowledge is diminishing is that with TCO you list everything these vendors claim to know—and then you add it up. By doing this, you assume your vendors know what is important to you and what is not. You assume that they actually understand your needs or that your needs are not unique. Dangerously, you also assume with TCO that the vendor or researcher providing the TCO estimations is highly concerned with your long-term results and that its definition of results matches your own. But that vision and yours will rarely agree. BCO has a different goal in that it is based on your own vision and the unique measurable elements you identify when building a strategy. Moreover, BCO wants you to add it up frequently. To make this work, you must focus your knowledge on things you can control. As a consequence of the shift, a few governments will start developing a business process sourcing strategy for electronic service delivery, aiming to engage the private sector and nonprofit organizations in delivering value-added electronic public services.[2]

The notion of collaborating to provide "public services" is important, because it implies the services in your BCO vision will be of value to your "public." BCO, unlike TCO, focuses on services and metrics valuable to the largest group of users. BCO is most importantly about finding the metrics we can commonly measure—and continually measuring them. You must be incredibly clear and concise. None of the parties in our case study started or ended with a complete picture of the costs and challenges associated with each agency's support of the state's current services; forget the notion of predicting the costs and needs of the future. Unlike TCO initiatives, BCO focuses you clearly on your situation and the parts over which you have the most control. In that sense, it is desirable to know less.

9.2.2 Challenges and opportunities

Metrics in IT are not limited to dollar costs of capital items. When approaching a large activity, such as upgrades or a new solution, you must determine the following:

- How will this solution affect your existing systems and processes?
- What positive impacts (opportunities) and negative impacts (challenges, risks) do you expect your solution to produce?
- How will you identify and mitigate undesired or unforeseen effects?
- How will you measure the impacts of the solution?

To understand BCO, it is important to know first why cost is so important. Business cost of ownership is the part of computing costs and benefits that is measurable. BCO accounts for the largest single component of an organization's computing costs. You can monitor, collect, aggregate, predict, and treat BCO like any other type of computational information. It is not the opinion of others; it is the facts, as you need to see them.

What are the risks affecting the solution?

Finding the risks and other challenges affecting your potential solutions means finding and asking questions that are not always easy to ask or answer. Most people do not quantify "risk" well, associating it only with "danger" and other negative connotations. In fact, danger is an extremely sensitive level of risk that requires attention to prevent generally undesirable results. To make your evaluation a little easier, start with the basics and begin to search for pieces of information about your current and future systems—look for things that you

- Know
- Do not know
- Wish you knew
- Need to know

To make this job easier, start with your users. In fact, it seems everyone has a few opinions to share on how well the technology suits the needs.

The key to finding risks is to ask clearly and frequently:

- You must ask *clearly,* as stakeholders often wonder why you are asking the question in the first place. Users, in particular, often distrust any change. Find a way to make the answer important to them. "How would you change the network?" may not be as effective a question as "How often do you find you can't get to a resource, and after calling the help desk you find that part of the network is down?" You may not find an answer to that specific question initially or ever—in fact, don't ever assume you have finally found the right answer, because even the questions should change.
- You must also ask *frequently,* because you may need to ask the same questions a few times to get a valid answer. Don't assume the first answer, or even the most frequent answer, indicates the most important risk.

For this chapter's case study, there were many different risks. Each set of risks directly related to a particular set of stakeholders and potential outcomes. For instance, the governor and CIO could not guarantee that the executive branch had the ability to require consolidation of its agencies or services. They also had no clear method for the agencies to pay for IT utility services in the new model, because the state provided funding for IT services through allocation of valuable capital resources (hardware and software) and full-time equivalents (FTEs). The staffing positions (FTEs) are identified as a head count with a certain budget for operation. If they were unable to reach this compromise, it was important to understand the risks of keeping the current distributed services model. As always, there was the sticky issue of any personnel restructuring. By evaluating each risk individually, the state was able to determine which risks were most important.

In the risk department, BCO brings some unique challenges to the IT circle. It is uncanny how many organizations find it acceptable not to document the most fundamental components supporting their users. As a result, vendors frequently spend valuable project dollars to re-review the network or other primary systems in order to avoid risks. To be clear: If you are determined to manage BCO, then you must *continually document your systems.* There is no other option, and it is of critical importance—BCO assumes that you know what you are measuring and that you are measuring it for a reason. (If you have not already started taking notes on your own systems, keep reading this paragraph until you do.)

Critical risks

For most organizations, finding the risks is only the first battle—after you find them, you still need to measure and set priorities on them. When measuring BCO risks, start by identifying the most critical risks of each stakeholder, particularly those that directly support revenue generation. Each stakeholder should be encouraged to find distinct and measurable risks, because this is a primary goal of BCO. For many risks, when developing the list you will not know the cost and frequency of the impact. This list will change and improve over time, so focus first on discovering and setting priorities on the most comprehensive list of potential impacts and mitigation strategies available. (See Table 9.1)

One particular technology concern for the state was an aging and expensive X.500 directory server, essentially a computer phone book of email addresses, which connected the various email systems. The state agencies used Microsoft Exchange, IBM/Lotus Notes, and Novell GroupWise. The IT department supported both centralized Microsoft Exchange and IBM/Lotus Notes services. Many brands and models of hardware were in use, especially within the ranks of PC servers. Inconsistent policies for sharing directory information and the high support cost for the aging directory product constantly annoyed many of the larger agencies. The state agencies, awaiting a consolidation decision, had delayed upgrades of major products, including Microsoft Windows 2000, Active Directory, Exchange 2000, Windows XP, and IBM/Lotus Notes. (See Table 9.2)

Risk management

The first component of BCO is risk. A risk, in the context of BCO, is a potential negative impact that is predictable and measurable. We use risk management as a standard process when delivering consulting projects, primarily to enhance communication with our customers. To categorize these types of risks, we review the business and technical challenges facing the

Table 9.1 *Business BCO risks for case study organization*

Stakeholder	Impact	BCO Metrics	Possible Mitigation(s)
Executives	High initial capital costs will affect legislative acceptance of the consolidation plan.	Purchase costs Recurring costs	Evaluate TCO, find business process improvements, and create a plan for effective reengineering.
Executives	Delays to system upgrades will increase implementation costs for homeland defense improvements.	Time to implement Cost to implement	Require upgrade projects to implement currently available security standards.
Agencies	Agencies will experience reduced service levels by using the IT department in a third-party manner.	Service availability	Involve agencies in development process. Ensure appropriate review of plans by all stakeholders.
Staff	Consolidating technology products will require retooling of solutions and retraining of users and support staff.	Skills assessment Training costs	Evaluate training requirements, build plans and cost estimates for migration and training.
Vendors	Product consolidation will affect key vendor relationships.	Vendor cost Partner satisfaction	Involve major vendors equally to ensure perception of fair competitive environment.
Staff	Reorganization may involve staff reductions.	Skills assessment Job performance Job satisfaction	Communicate clearly with staff. Identify staffing expectations and criteria for filling positions.
Citizens	Negative impacts during changes to services may affect core government processes and in turn, citizens.	Service availability	Communicate freely during times of change and use "distributed" migration tactics to minimize localized disruption.

organization. By telling the customer where we see risks, we can communicate our concerns in a standardized manner.

One common method of risk management used by our team ranks risks according to severity and probability. The first step is finding risks attributable to technical issues, potential events affecting the schedule, and expected administrative roadblocks. We measure severity according to potential impact, from 1.0 (invisible impact) to 5.0 (complete chaos). Probability is subjective and determined by the consultants according to their own experience. (See Table 9.3)

Table 9.2 *Technology BCO risks for case study organization*

Stakeholder	Impact	BCO Metrics	Possible Mitigation(s)
State IT and agencies	Aging directory server needs replacement or upgrading.	Cost to support Cost to replace and support	Implement new central directory service. Integrate departmental directory services.
State IT and Agencies	Directory service architecture needs updating.	Cost to update Cost to replace and support	Implement new central directory service. Integrate departmental directory services.
Finance	Multiple email systems increase hardware, software, training, and support costs.	Hardware cost Software cost Training cost Support cost Other related costs	Consolidate to one or two primary supported mail systems. Consolidate email systems to centralized maintenance facilities.
Finance	Costs for maintenance for aging software may increase if not replaced or upgraded.	Current maintenance cost Maintenance cost if replaced Maintenance cost if upgraded	Upgrade software to newest version. Replace software with changes to other systems. Choose and implement new software.
Agencies	The organization will not fund projects requiring support of new Microsoft or IBM technologies until standards are determined.	Time to determine standards	Develop complete standards and processes for implementation. Develop core standards for interoperability.
All	Existing solutions (especially applications) may require change to meet future technology standards.	Number of solutions requiring change Time to implement change	Identify and document factors in the solution likely to change over time.

Table 9.3 *Consulting risk management plan*

Risk	Severity	Probability	Owner	Date	Mitigation
A description of the risk	1.0	100%	The person responsible for risk mitigating the risk	4/3/02	A description of how the risk is being resolved

How do you measure risks?

BCO is concerned with distinct measurable costs, and as such, the risk measurement process involved with it must be similarly concerned. Severity, without a metric such as cost, is a difficult measurement to apply when considering the broad range of potential and changing impacts that affect a business. Probability, implemented as a guess without strict data gathering, is rarely questioned but often should be questioned. When the risk that had 10% probability actually happens and affects your revenue-generating business units for six months or even a day, you will certainly understand why a project-centric risk plan does not fully address the needs of a large system such as consolidated IT services.

When computing BCO, consider both direct risks and indirect risks. Thus it is important to review and correlate BCO effects between different solutions. For example, what happens to each system when the network goes down? How does network downtime affect server availability? The loss of which connections would produce the greatest effect? Most crucially, what do the users think is important that you have not considered? Because BCO focuses on the most important components of your systems, such as the network infrastructure, it's not enough to consider the cost of email server downtime alone. You must also consider network interruptions, Internet disruptions and attacks, viruses, lost documents, disk replacement, and a large variety of other costly issues that should be easier to measure than just "downtime." In BCO, it is not necessary to dream up scenarios that are likely to strike "every company someday"; rather, find the scenarios that will affect your organization today.

Security is a perfect example of risk affecting BCO. Consider the examples of evaluating the impacts of different security incidents that are shown in Table 9.4. (See Table 9.4)

Before running to measure your potential security hazards, customize the figures in this chapter's tables to meet the individual needs of your organization. For most of us, this will not be easy. The IT department rarely keeps the level of detail necessary to address these types of incidents. For that reason, in BCO we need to separate the known from the unknown. If you cannot actually look at a report of the measurements today, do not guess. With BCO, you must identify the unknown so you can decide whether to start tracking it in the future. There will be time for guesses later, but only in certain solution-modeling exercises.

Is downtime really a risk?

To be clear, downtime itself is not a risk in BCO. The processes, procedures, equipment, and personnel required to make service available following a disruption are an extremely important part of BCO. This is an important

Table 9.4 *BCO impact management plan with example data*

Impact	BCO Cost	Frequency	Owner	Standard Mitigation
Description of the impact	Average external and internal cost in dollars to recover service	Expected periodic frequency of impacts (e.g., once per year)	Group of responsible support resources	Documented processes for preventing and responding to impacts
Virus (major)	$40,000 +	2 events per year	Corporate IT	Antivirus software Consulting Data recovery System recovery
Virus (small)	$40 +	2000 events per year	Local IT	Antivirus software Data recovery
Denial of service (Internet)	$2,000 +	1 event per year	Corporate IT	Internet firewall ISP support Consulting System recovery
Destruction of data (internal)	$__	__ events per year	IT and security	Forensic analysis Data recovery System recovery
Unauthorized release of information	$__	__ events per year	IT and security	Forensic analysis Data recovery System recovery

contrast between BCO and TCO. In their summary study on TCO in the enterprise environment, the Radicati Group Inc.[1] provides the following guidance on how downtime is measured:

> *Downtime Costs – These include any lost productivity caused by failure (i.e., downtime) of the messaging systems. These include both scheduled and unscheduled downtime. . . . For the purposes of this study, we assume that unscheduled downtime affects 25% of the total user population, whereas scheduled downtime affects only the messaging IT staff.*

Assuming most readers do not purchase the full study, this is the extent of background information provided by Radicati for enterprises considering $20 to $70 of downtime for each user per year. The entire Microsoft or IBM email solution, by Radicati's estimate, costs each organization about $279 to $285 TCO per user, per year. No wonder TCO is confusing, because the brief explanation describes a potential downtime cost in the range of 7% to 25% of the total solution cost. (Incidentally, if we ever implement a system where downtime costs that much, someone is losing his or her head over it!)

In fact, Radicati seems to agree. The same document later provides a review of service provider environment systems. That study gives the following definition for downtime:

> *Downtime Costs – These include time spent by full-time administrators dealing with system failures (i.e., unscheduled downtime) as well as scheduled downtime. We assume that both scheduled and unscheduled downtime affects all full-time messaging administrators. We do not attempt to measure the effect on the subscribers, though here the impact of higher downtime probably translates into higher subscriber attrition.*

The dichotomy here is stunning: In an enterprise, you must pay for a user's downtime, but as a service provider you may summarily dismiss this expectation (although you may lose subscribers if service is poor). In reality, few organizations find themselves accepting either the black or the white explanation, most organizations find themselves considering a shade of gray. In BCO, you must address the second definition of downtime just listed—"full-time messaging administrators"; in other words, people and process time. Unlike "lost productivity," these variables can be easily measured by recording how long it takes to restore service.

What is your vision?

Another way to view risks is to consider them opportunities. The state CIO and his financial officer, with the support of the governor, asked the state's agencies to consider consolidating some of their systems and personnel into the centralized IT department. By consolidating support and hosting for key systems, they argued, the cost to the state for supporting these services would certainly decrease. In early 2002, this state like many others was facing an unpredictable recession and large budget shortfalls caused by decreased tax revenues.

The economic condition and the new needs for homeland defense were of highest administrative priority; however, aging and poorly connected systems frequently affected both of these core challenges. The agencies were facing moving targets when trying to implement the frequently changing and wide-ranging new standards of government, especially those involving security and technology. By focusing investments on common and critical infrastructure, one could argue that these shared improvements would positively affect many different projects.

The state IT department also wanted to create an internal service provider, of sorts, to respond to the universal needs of each department and agency. The new centralized IT department would support the common infrastructure by purchasing products and developing valuable IT services for the agencies. Agencies would pay a standard monthly fee to the IT department for each user connected to a particular service. The new IT department would need to hire and manage most of the state's IT employees, expecting some cost benefits from support consolidation. The targets they had proposed for consolidation were mission-critical for the government, education, private organizations, and citizens. The targets were the core components needed to support these stakeholders' vision for the next-generation Internet-ready infrastructure.

9.2.3 Critical investments in IT

It was a recession year. The state IT infrastructure was aging, but the agencies in contrast were already supporting citizens by creating innovative Internet solutions. Seemingly random issues in state networks and systems created frequent and sometimes enormous impacts. The CIO in our case study decided, as did many organizations, that it was time to optimize and improve the computing infrastructure. The risk was significant, but the expected outcome of building modern and cost-effective systems was critical. The governor, facing a reelection campaign and needing a big success, was more than receptive to the innovative vision of the CIO. The agencies, however, were skeptical from the start.

What are your priorities?

To the agencies, previous experiences with the state IT department had taught that any consolidated or centrally hosted services meant losses of

- Design control
- Cost control
- Innovation
- Support

From a BCO point of view, finding this discomfort means you found a great place to start. In those topics where the stakeholders are most dreadfully concerned about flexibility, cost, value, and "trustability," you will often find the critical information you need to start measuring and optimizing your organization's most important costs. In fact, when you confront these concerns directly, many stakeholders will tell you at great length why these particular investments are so important to them.

Once you find and weigh the challenges and opportunities, the awkward task of ranking your organization's needs begins. By identifying a solution or component as a priority, you have taken the first step in making the investment. Setting priorities requires a fairly clear understanding, and thus you will find that setting them produces a significant amount of information. If you began this BCO process by making a wish list, the priorities are the parts that "float to the top," because they are so important that you required a subproject just to evaluate each one of them and come to this point.

Once you identify your list of priorities, it is important to make sure the majority of stakeholders agree on the importance of those topics. To communicate their vision and priorities, the governor and CIO in our case study created "charters" to describe and enforce their perceptions of the state's most important IT consolidation needs:

1. Data warehouse
2. Server farm
3. E-mail
4. Storage
5. Web hosting

What is the potential scope of each priority?

Often priorities overlap. To meet the BCO requirement of being specific and measurable, evaluate both individual and grouped priorities. In practice, this means you must be specific about how each stakeholder views a certain priority. Email, for example, may be extremely important to an individual because of more than one job function—users may need email to communicate with both customers and vendors.

Each department or agency in the state had created its own networks and applications, and they were fighting consolidation and the implementation of important standards for managing technology systems. Many of the state's technology workers had been there for a decade or more. The state supported an advanced Internet infrastructure, with fiber-optic connectivity to many or most locations. Some of the larger agencies, supporting services such as health, public safety, and welfare had made significant investments in their IT systems and applications. Many firewalls separated the state's organizations, which each agency had interconnected using the state's Internet and mainframe links.

In many solutions, it is possible and normal for loosely related components to have relationships. Here's a frequently used example: When a construction team cuts a fiber-optic network cable separating the data center from the users, neither the email nor the SQL database is very valuable. For a mission-critical Internet site or Web service, this type of loss can be excruciating. For organizations with consolidated IT services, this situation is similarly traumatic. If you lose your network, your customers lose your services. Do not undervalue that sentiment, because it relates to one of the most important new realities of IT: If you lose your network, you lose your customers.

9.2.4 Finding solutions

Once you have identified the needs, you must go on to build good IT services. Don't start with the individual services, however. Start by classifying groups of services into solutions. A desired result of building good services is to build good solutions to customers' needs, so start by answering these questions:

- Who will participate in identifying and selecting solutions?
- Who should own and maintain the solutions?
- What solutions are available?

When you talk about solutions, remember that solutions are focused first on business needs and then on technical ones. The solution is how we explain it to a customer, and the services are how IT delivers the solution. Desktop deployment, for instance, is a solution. It requires services such as hardware and software procurement, disk imaging or scripting, information backup and restoration, and customer support. A solution requires a set of services, and a service often supports many solutions. In database developer parlance, this is a "many-to-many relationship." It is also the hardest kind of relationship to work with.

Who will participate in identifying and selecting solutions?

In general, a "stakeholder" is anyone with a stake or interest in the solution—such as customers, support staff, and management. Classifying stakeholders is an important role of the executives and IT management responsible for sponsoring and managing implementation projects. The team implementing any IT solution may require participation of many different stakeholders. It is very important to consider these stakeholders as individuals, as well as in their role as members of a group. (See Table 9.5)

Because of the many-to-many relationships between services and solutions, it is important to classify stakeholders. Once you identify the stakeholders, they become a permanent part of a solution. From vision setting to retirement of individual components, these stakeholders are important. New stakeholders and categories may appear over time, so it is important to maintain and update the list. The importance of identifying stakeholders should be apparent: You must include them in tasks that affect them.

Who should own and maintain the solutions?

Some solutions, such as PC software, are easy to sweep under the rug. After implementation, the development team generally moves on to new opportunities while the operations team takes over maintenance. This works well as long as you know the original developers and application owners. But during large projects—especially consolidation—applications without owners are troublesome. If the solution is old enough, the operations team may not know who built it or how to change it to meet new PC standards. Each solution should be associated with at least one owner and a group of maintenance/support resources.

At minimum, identify the following roles for each service, and keep the role assignments current despite promotions, reorganization, turnover, or other changes:

- Owner(s)
- Developer(s)
- Support contact(s)
- Solution dependencies

Table 9.5
Classification of stakeholders

Service/Solution	Business Group	Function
Desktop deployment	Customers (“end users”)	Operations staff
Server deployment	IT Staff	Support staff
Service provisioning	IT Management	Windows specialists
Service monitoring	Executives	UNIX specialists
		Mainframe specialists

One practical method to track solution resources (such as owners) is with an electronic directory. Chapter 5 discusses architectural components such as directories. By maintaining a directory of services and solutions, it is easier to find an owner or support resource when you need one. By keeping important information such as developer/vendor contacts, you can often expedite future evaluations of a particular service. In many cases, it is also handy to keep track of the solution's dependencies: hardware, software, operating system, or even other solutions.

What solutions are available?

Knowing that large purchases were in the works, the state in this chapter's case study also selected a few options of popular application services and asked the vendors to provide TCO estimates based on "industry standards" for the complete support of their solutions: Microsoft Exchange versus IBM/Lotus Notes. The local account team for Compaq, sensing the importance of the upcoming decision and noticing the increased activity of their competitors, formed a team to respond to the state's request for pricing information. At Compaq's request, the state's IT department and agencies held a series of onsite "knowledge-gathering sessions" to help the vendors build a more complete picture of the state's requirements. To ensure that no one had an unfair advantage, the state invited vendors to deliver separate sessions. The vendors, in turn, helped the state IT department and agencies to begin discussing their environment and needs.

Compaq had long been a leasing partner of the state, and the majority of state organizations had selected Microsoft technologies such as Exchange email. The state's central IT department used Compaq hardware and Microsoft software; however, the department had been a longtime IBM partner, and the relationship with IBM was strong. The standards selected for Web development included WebSphere and other IBM products, whereas many agencies were using Visual Basic to develop Microsoft-based solutions. In fact, every government agency or department supported many different technologies. This mixture, combined with the innovative but combative nature of each IT department, led to widespread mistrust and a very expensive technology support bill for the quickly aging IT systems of the state.

In the same way, evaluate and select solutions based on both the customer's needs and the appropriate solutions. Here are some lessons from the case study:

- Often the best solutions are the ones you already own and maintain. The advantages in learning curve and integration time are measurable and often significant.

- It is important to consider multiple options when appropriate—adding to the difficulty of finding the right solution.
- There is sometimes no "one right answer"—the best solutions often let your customer choose from a menu of service options.

This entire book deals with finding the best solutions to IT problems. The next section describes our first BCO solution, the "solution calculator."

9.2.5 The first BCO solution calculator

As part of our effort for the state, we built a rather large worksheet that would allow the state stakeholders to consider the cost impacts of certain service solutions. Conceptually, we intended the worksheet to describe the cost per user, per month, for specific services. The cost would be high level; addressing the most important and predictable IT costs for supporting the service. We arranged the costs according to the state's request for "apples to apples" comparisons. Specifically, the state wanted us to address email, but the arrangement seemed valuable for other services as well. This spreadsheet, in effect, is the basis for BCO.

This section both explains the BCO spreadsheet as originally developed and points out some glaring difficulties with the specific model—areas for improvement as you make your own BCO measurements. I also discuss some of the considerations that concerned our team, and some potential solutions.

What is the formula for BCO?

It is a challenge to add up IT costs. There are many different models, but the two general formulas for BCO are as follows:

BCO = [IT purchase costs] + [IT leasing/subscription costs] + [internal IT costs]

BCO = [hardware] + [software] + [internal staffing] + [external services] (See Figure 9.2)

Other variations of these formulas are acceptable. The key is that for your organization, BCO is a formula that allocates costs to individual services as in Figure 9.2. Throughout this chapter, the formulas given are the ones that made the most sense to HP/Compaq and the customer. Your own formulas may differ but will generally include at least the items in Figure 9.2. The key to BCO is that you include only the measurable components.

Figure 9.2
BCO service cost allocations

Consolidation Costs (purchase	Hardware	Software
Windows 2000 / Active Directory	$62,443.81	$468,286.80
Exchange	$1,001,613.93	$228,516.18
Notes	$0.00	$0.00
AntiVirus	$17,948.70	$168,432.00
Content Filtering	$0.00	$0.00
Exchange Connectors	$8,833.44	$11,720.00
-	$0.00	$0.00
Outlook (Web-based)	$13,885.41	$5,860.00
Outlook (Terminal-based)	$44,871.75	$14,650.00
Outlook (Desktop)	$0.00	$0.00
-	$0.00	$0.00
Configuration and Migration	$0.00	$0.00
Additional Staffing		
	$1,149,597.05	**$897,464.98**
Annual Maintenance Costs	**Hardware**	**Software**
Windows 2000 / Active Directory	$3,410.62	$118,866.16
Exchange	$58,192.46	$217,023.36

How do you measure cost?

The calculator measured cost in dollars. The primary concern of the state was to determine a cost per user, per month, for the given service. Dollar cost is generally one of the most important considerations of any organization. This is the investment, in both one-time and recurring charges, required to support IT.

In reality, additional measurements of cost are important. In fact, cost is only one way of looking at the impact of BCO. In BCO, cost reflects the impact caused by a particular approach: in dollars, in customer satisfaction points, in availability, or in any other measurement important to your organization. In fact, the importance of these factors is likely different for each group. The state patrol, for example, may define its user support agreements to include extremely high levels of guaranteed availability. In return, it may not require nearly the level of user satisfaction as the finance department or a customer service team.

What about the futz factor?

BCO does not include the user costs found in the TCO model. The primary assumption is that IT is a service provided to the user, not the other way around. The user, in a service model, pays a certain amount to use the service when it is available. Billing begins on installation and terminates on disconnection. Just as with your telephone and cable service, when service is unavailable, users may request a refund. The incentive to IT is clear: Keep the service up, and you may keep your job.

9.3 Monitoring IT with BCO

Monitoring is important. Every utility uses monitors of one kind or another. The most important public utilities (such as fuel, electricity, and phone service) use monitors for many reasons. The monitors are where the bills come from, often a meter or some other physical device used to report consumption. They provide feedback to let the providers ensure that services are

- Running
- Available
- Efficient
- Measured
- Recorded

Monitors record the frequency or intensity of events, according to a standard set of units. Monitors are like the nervous system—used both for managing critical function and for communicating important messages. Some IT departments use *service-monitoring* tools such as HP OpenView, IBM Tivoli, or CA Unicenter to manage and communicate about important servers and services. These tools are important in that they provide measurable feedback that, when aggregated, IT management can use in developing metrics for IT services.

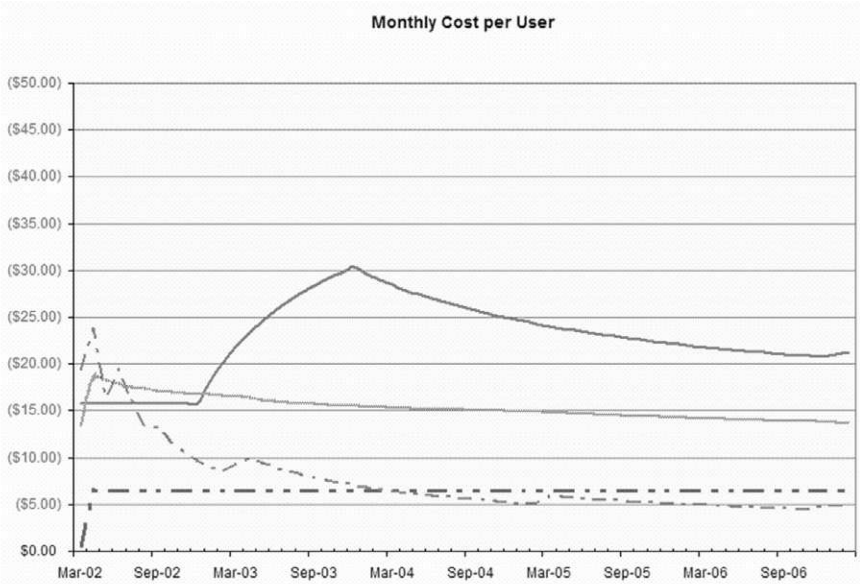
9.3.1 Architecture: investing in value

The business and IT customers, valued as individuals and as a whole, must be involved frequently and deeply in defining, implementing, and operating technology solutions. To ensure long-term success of IT investments, it is critical for business and IT managers to work together while creating solutions that provide internal and external customers with measurable business value.

The customers of IT include all the users of an organization's technology, including managers, workers, customers, vendors, and partners of the organization. Pulling the plug on their technology would immediately stop many organizations from functioning. No department, agency, group, team, or individual is immune to the effects of the IT department. Love it or hate it, the department controls much of our lives. Business and its employees rely on IT to provide services such as a paycheck, a computer, a connection to the network, mail services, software applications, and every sort of information. (See Figure 9.3)

Too often, organizations assume that value will magically appear in response to implementation of new software, faster hardware, better programming languages, or higher bandwidth. Adept vendors, with experienced sales and marketing staffs, frequently describe the "nirvana" experienced by other customers on installation of their newest product or service. What they are not mentioning is the staggering investment many of those organizations made in properly aligning the solution with the business. They are also not mentioning the customers who often failed at implementing an identical solution. Figure 9.3 presents an alternative provided by BCO, a forecasting chart you can use to develop projections based on changes to your services model.

Figure 9.3
BCO forecasting chart



Value is a measurement of impact to business, not speed of processing or ease of implementation. If an organization fails to properly *plan, build,* and *operate* even the simplest solution, the business will likely never see measurable value. In fact, many very small solutions have turned into business nightmares. For example, just look for 8-bit and 16-bit applications in your Windows environment. Residue of the popular DOS and Windows 3.1, providing support for these pesky little critters sometimes means the difference between staying on Windows 95 and being ready for Windows XP. In business terms, the value of a 100% current desktop environment is its simplicity: Organizations with a consistent desktop platform only have one platform for testing when implementing new applications and software upgrades.

Naturally, vital questions arise when determining a solution's value:

- Who should be responsible for identifying, selecting, designing, implementing, and operating valuable solutions?
- What are the critical factors for success?
- Technology solutions can affect every part of our value chain and business processes. How do we measure the impacts of a solution, and what impacts should we look for?
- What is the value of using technology as a change agent?
- How can technology help us wisely reengineer our organization?
- What is the measurable value of technology? Where does technology really fit in?

To answer these questions, it is important for technology and business to work together. Neither the IT department nor the business can answer these questions in a vacuum, and both must understand that the answers are subject to constant change. The companies, governments, and workers that rise to meet the demands of the Internet generation may eventually succeed. Those who ignore the information age will likely find themselves unable to communicate with much of the next generation of consumers. In any case, a service is only valuable as long as it is used—so assume your answers about value will change over time.

9.3.2 Process: coordinating best practices

The first Internet boom taught us all just how powerful a global market can be; the second Internet boom may bring significant advances in the way we each live our lives. To build the next-generation computing platform, it is important for every employee and each customer of an organization to participate in the process of getting there. Finding challenges and risks along the way is a certainty, but minimizing the actual impact of these issues is, as MasterCard would say, priceless. One method of minimizing impacts is by having efficient and well-practiced processes: *best practices*. These repeatable formulas for success require us to manage a product using a specific service—Stephen R. Covey would say they are ways of sharpening the saw. In Covey's *Seven Habits of Highly Effective People* this is the last habit—the one that keeps the others running smoothly.

Enterprise directory services

If you look inside HP, you will find impressive BCO practices everywhere. Compaq grew significantly by acquiring Tandem, Digital, and other smaller technology companies. To connect the myriad of systems and people within Compaq, they placed an early focus on enterprise directory systems. One proud moment for former Digital employees is the integration of directory services that occurred immediately after the merger.

Directories are an important example of a solution that facilitates communication and processing between other solutions. In BCO, directories serve not only as phone books but also as the focal point of every user, system, application, process, and service in an organization. Compaq implemented intranet-based applications using directory to service benefits enrollment, resource management, employee reviews, payroll advice, and virtually every other activity within the business. Because the directory reflects the current state of complex relationships within the organization, inclusion within the directory is the start of business as an HP employee. You get an employee number, a network login, access to the corporate intranet, an email address, and a standard employee identification badge.

For Rainier, our acquisition by Compaq in October 2001 involved integrating two "native mode" Windows Active Directories (bringing together two or more critical [and often, thank you Microsoft, inflexible] directories). Both companies being early adopters and supporters of Microsoft's Windows 2000 platform, we were somewhat pushing the technical envelope. For those not immediately impressed, some capabilities for managing multiple directory "forests" have been added to Windows 2003, but many features are still in development. The result of Compaq's expertise in directory services for our organization was a methodical integration process so smooth that our customers have not noticed. Starting with our executives, Compaq quietly interwove our organizations. The focus was on getting critical parts done before January: the computer networks, employee benefits, directory services, payroll, and then sales. These business-critical services are just the beginning of what makes a good BCO solution. In addition, the upcoming merger with HP meant that if Compaq did not integrate us before the merger, HP would integrate us after the merger. To avoid billing complications, Rainier, Compaq, and HP settled for a mixed approach. Our badges and voice mail changed, as did our organization—but the billing system remained the same and Compaq used its internal cost transfer system. We now call this practice in HP "adopt and go"—if you have to make a choice between similar options, pick the best, and leave the rest.

Intranet solutions

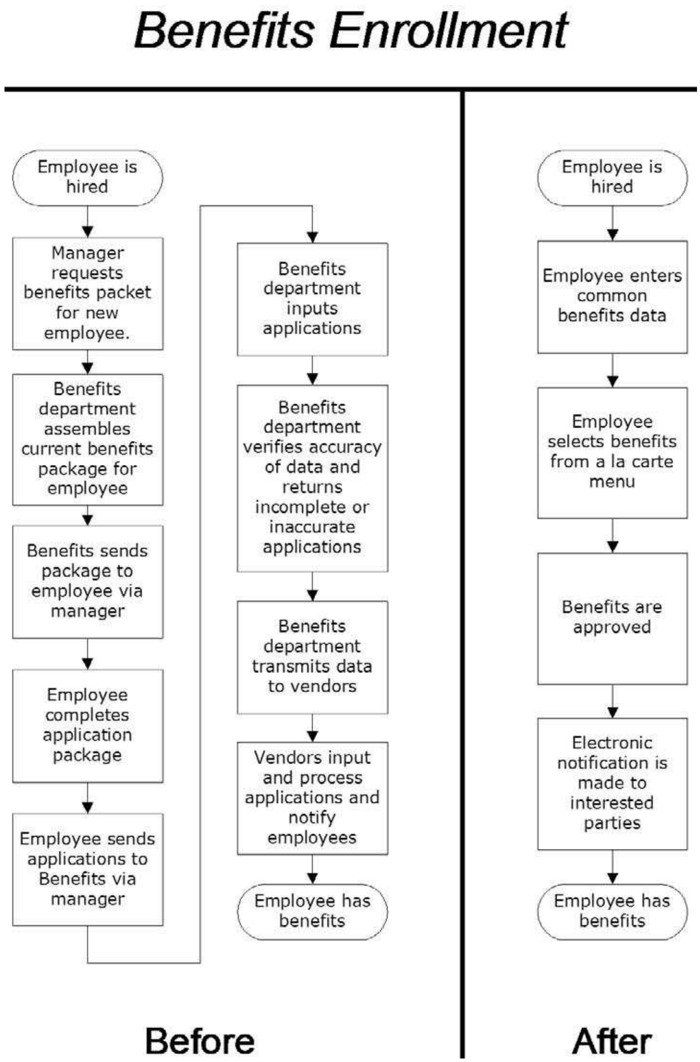
Intranet solutions are a great way to integrate an organization. Many companies are organizing information and resources in an easy-to-navigate web interface. Some organizations have set directives that web-based applications are the standard for development of new applications. Many large companies are beginning to invest in portal solutions that allow an individual to browse and search for these applications across their organization-wide Intranet. At HP and Microsoft, these solutions are our primary interface with every service and with all the knowledge within our organizations.

Another excellent example of a good BCO solution is Microsoft's Human Resources Web. HR Web is an internal solution used by Microsoft employees to perform common tasks related to human resources and benefits administration. The impressive part of HR Web is not its technology; in fact, Microsoft constantly updates the components supporting HR Web to use the latest technologies. (See Figure 9.4)

The value of HR Web is that, according to Microsoft's Web site, it "streamlines processes, information, and transactions for Microsoft's Human Resources department." In short, it allows benefits staff at Microsoft to focus on the business function of HR by giving them time to explore new solutions and provide customer service should problems arise. The application includes some of the most time-intensive, frequent, and costly processes Microsoft could identify: benefit enrollment and maintenance, time sheets, absence reporting, pay statements, and other common paperwork. Started in 1995, the HR Web eliminates over 200 paper forms and saves, according to Microsoft's own estimate, over $1 million per year.

The worker wins even bigger. The concept of HR Web became so popular that virtually all internal forms processing within Microsoft happens

Figure 9.4
Microsoft benefits enrollment and HR Web



on the Intranet. A check for an expense report that formerly took three weeks to process is now in the employee's bank account within three days.[2] For organizations wanting to explore the value of the Internet without jumping in head first, HR Web serves as a perfect example of using available technology to empower workers to optimize processes and communications. By constantly growing a simple intranet solution, Microsoft hit a home run in addressing extremely time-consuming tasks that normally distract people from doing their jobs.

Assuming each of their 50,000 workers only fills out half of the 200 HR forms once, Microsoft has avoided the manual processing of approximately 5 million forms, or approximately 200 gigabytes of electronically imaged paper forms. When you consider that employees fill out many forms annually, quarterly, or even weekly, you quickly find Microsoft has avoided the processing and storage costs of hundreds of millions of forms. In addition, by using the directory and messaging system to route electronic approval forms, Microsoft has significantly reduced the time required by managers to enforce standard business protocol. For them it is simple: The manager clicks Approve or Reject when the email arrives.

9.3.3 Integration: enabling optimization

It seems valid to assume that the value of technology solutions should grow; however, most solutions rarely change after implementation. Because of the massive costs for product selection and implementation, many organizations treat technology solutions as long-term capital investments rather than as services and utilities requiring constant improvement. In many cases, software developers and IT support engineers are subject to an endless stream of new products and services. In most cases, the only opportunity to review an existing solution comes many years later when implementing a product upgrade.

For example, consider the number of companies still looking for the best way to retire Windows 95, almost 7 years and three major Microsoft business platform releases since its introduction. Most users agree that Windows NT was faster, Windows 2000 is more reliable, and Windows XP is more user-friendly. Incredibly, many Microsoft customers support all four operating systems. In the worst cases, customers support 10 or more different combinations of desktop operating systems, versions, releases, and service packs for the Windows platform alone. This does not include the tens, hundreds, or thousands of variations caused by individual desktop applications.

Understanding the negative business impact of a nonstandard desktop is difficult unless, of course, you happen to be responsible for supporting and developing solutions on each of these different desktops. Solutions are growing and living entities. To ensure that organizations realize technology value in the long term, both the solution and its owners must be able to support rapid change and address inconsistency in the needs and habits of the user. For obvious reasons, we expect machines of every type such as tools, vehicles, and appliances to require and receive regular servicing. In fact, sometimes it is easier just to buy a new car, especially when trying to use a rusted-out two-door sports car for a family of eight.

There is no inherent value in technology unless it efficiently fills a current need for the majority of users (a common issue for current owners of 8-track and phonograph players). Executives, managers, engineers, operators, and users must constantly look for better ways to use and support technology. Computing solutions, like cars, eventually get old. They all expect occasional servicing, are often improvable, and eventually need to be retired. Deb Best, an account manager for Compaq's government and education market, stated it well:

> *It is funny we don't get mad at the car companies for refining and improving their products. We understand that we don't get credit for the last car we bought either.*

Why is technology different? Technology optimization is everyone's job, always.

What are the right answers?

Although BCO focuses on costs, it is not always desirable to reduce BCO costs. You must weigh solutions such as disk imaging and bit dropping in terms of their overall cost, not just the cost of the "point" solution. No wonder IT costs so much—we often try to reinvent the wheel every time we implement a new server. Develop standards for assessing and monitoring IT, and then move on to more complex tasks like reducing TCO and providing valuable solutions to stakeholders.

There are right and wrong places to save costs. The historical proposition of late 20th-century programmers, in retrospect, seems almost ludicrous: Only storing two digits for the year field of the date of any event would allow the entire system to save 25% (or two out of eight digits) in a common date such as 01–01–2000. The programmers of the 1960s, 1970s, and 1980s simply never considered the possibility that the dropping bits in their code would ever become a problem in 2000, although in reality many parts of those programs are still in use today. This is a curse of IT: The most useful—but not necessarily the best—systems often outlive their designers. (If you do not believe this, go find any COBOL programmer and ask how busy he or she was before Y2K.)

In many cases, our team has observed, organizations prefer to base IT improvement on small parts of large processes, such as desktop deployment. This is a dangerous ploy. One common and established solution for workstation TCO reduction, disk imaging, required organizations to purchase disk space or CD-ROMs to store computer images for each new hardware and/or software combination. Each new PC model, network card, or software requirement would create a cascade effect, increasing requirements for disk image storage. Purchasing departments introduced additional challenges by ordering low-cost hardware rather than standard hardware. Worst of all, customers rarely cleaned up old images.

We generally preferred scripting (various computer languages used to automate common tasks in a computing environment); although it did require us to maintain some programming skills, it was easier to make changes when hardware, software, or configuration changed. In most cases, the time or storage spent creating each new image was much more expensive than the time to create and integrate automated installation *scripts*. These scripts allow engineers to automate processes in a customized way. For example, one function of a login script creates connections between drive letters on a PC ("N:") and network storage on a Windows, UNIX, mainframe, or other system. We used automated installation scripts to perform configuration and software installation during automated installation of PCs. Most customers learn to customize these scripts with very little instruction, and then create a common place to store these scripts.

9.4 Conclusion

As information technology becomes more complicated, current technology value and cost assessment models such as TCO will become much harder to implement and maintain. Understanding how and most importantly where to use these different models is one important component of managing technology investment. Without considering the widespread and long-term effects of an IT decision, it is impossible to make an informed decision likely to stimulate stakeholder perception of IT value.

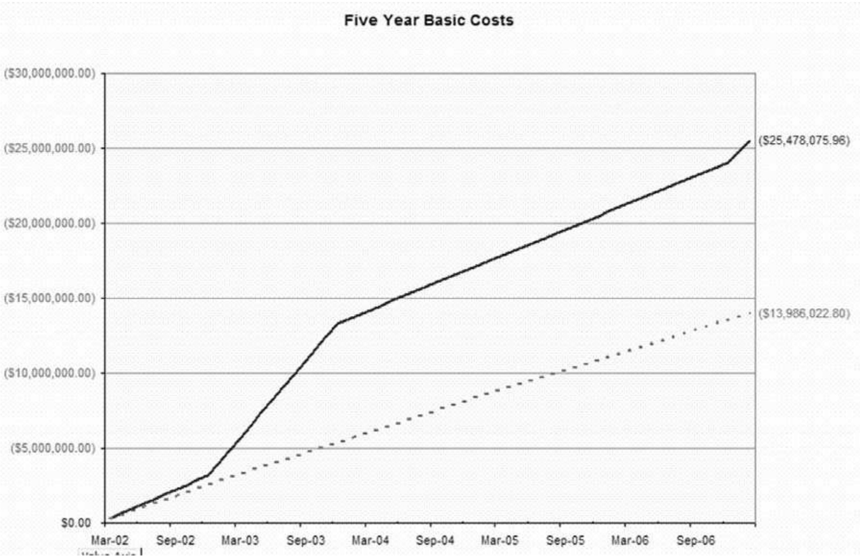
TCO includes all of the organization's technology components, such as hardware, software, consulting, internal staffing, and training. It is also all the effects caused by the solution: informal learning, co-worker time, futz factor, downtime, management, administration, and so on. The list goes on to include any cost that a business could attribute to technology. If business found a problem, someone in IT could find a solution that would "reduce TCO costs and make happy customers." TCO has been a major selling point used by every technology vendor and every CIO. The mission was clear: Let business know that IT is an investment. The goal of BCO, as shown in Figure 2.5, is to see the potential long-term differences between different delivery options—to avoid taking a wrong turn in an expensive direction.

Using BCO, you can measure the distance between two solutions, such as two methods of desktop deployment. Assuming most applications are installable from the network, a basic image of the operating system and office imaging required at least 600 MB (megabytes; 600 million bytes), compared to a 10KB (kilobytes; 10,000 bytes) installation script. The relative storage used between a disk image and an installation script is at minimum 60,000:1. A disk image, however, is often specific to a hardware and software combination. For a company with 25 different hardware/software combinations, this equates to 15GB (gigabytes; 15 trillion bytes) of storage, just for installation images. However, automated installations required 30 to 45 minutes, compared to the 5 to 10 minutes required for cloning. In addition, you must maintain and often modify automated installation scripts, requiring you to retain scripting skills or outsource hardware and software changes. (See Figure 9.5)

Measuring unique components of a single task such as desktop deployment has its own challenges:

- Because monitoring is a long-term form of measurement, you must develop a flexible model for monitoring IT.
- It is not sufficient to gather the raw statistics as discussed in this chapter; you must also organize this information according to priority and structure

Figure 9.5
Five-year BCO projections



and convert technology units such as gigabytes, servers, users, and processes into common dollar and time costs. BCO does not address these two challenges; rather, it uses RAPID to organize BCO and perform analysis.

Most importantly, remember that the API—architecture, process, and integration—fits in the middle.

References

1. The Radicati Group Inc. (2002), available on the Web at www.radicati.com/cgi-local/brochure.pl?pub_id=127.
2. Christopher Baum (2001), available on the Web at www.gartner.com/DisplayDocument?id=351064. In this research note, GartnerGroup discusses the challenges and "different reality" facing governments in 2002.

10

Shaping RAPID IT Solutions

Hamlet: Do you see yonder cloud that's almost in shape of a camel?

Polonius: By the mass, and 'tis like a camel, indeed.

Hamlet: Methinks it is like a weasel.

Polonius: It is backed like a weasel.

Hamlet: Or like a whale?

Polonius: Very like a whale.
(William Shakespeare, "Hamlet," Act 3, scene 2)

10.1 Introduction

Defining independent computing modules (components) is a critical part of managing the next-generation IT environment. One popular method of modularization in science is the definition of structural models, such as the classification of species, elements, biological components, and social structures. To do this in a technology management schema requires coordination of many IT decision-makers and stakeholders; it is a team effort. In the section on **Solution Modeling**, you will learn how to reflect your organization's priorities in IT strategy and communications. The section on service modeling shows you how to use ITSM and HPGM to understand how your services relate, assign objectives, set priorities on requirements, and continue to develop your IT service standards by enabling service-level agreements (SLAs) and architectural standards. The section on process modeling shows you how to use UML to document the repeatable processes affecting your IT components. The section on operations modeling shows you how to use HTML and XML to integrate stakeholders with your IT components.

How do RAPID models work?

The goal of enterprise modeling is to model your organization's high-level requirements so you understand the fundamentals of your organization and its environment. To do this, you must develop three artifacts: an enterprise-level use case model, an enterprise-level business process model, and an enterprise-level change case model.[1]

RAPID defines a hierarchical system for defining the classification of a system of IT components. A RAPID model is a picture of an IT component in the context of the specific role it plays in the organization. At the top of the RAPID pyramid representing management strategy are solutions, the most complex RAPID components. At the bottom are the individual basic changeable components that comprise IT. (See Figure 10.1)

The models allow you to record multiple views of the information critical to IT decision making, such as the functional hierarchy of services or the cross-functional nature of processes. One way RAPID helps you balance goals is by supporting multiple views of a given situation. Each RAPID model helps you understand the relationship between these different views. Of greatest importance in managing IT is the relationship between business and IT itself. The solution model, for example, represents this balance by actually implementing the business view as a technology solution.

Before you build your model, you must know a few things about the best practices of IT service management:

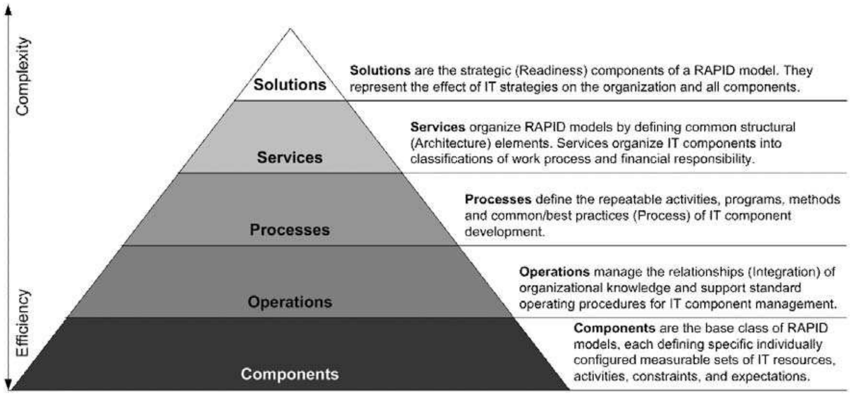


Figure 10.1
Layers of RAPID solutions

- Your model will not be organized by mainframes, minicomputers, supercomputers, microcomputers, Microsoft, HP, IBM, Sun, Oracle, Linux, or any other product. None of these has proven consistently to be "the key" for every organization. In fact, if you use them you risk exposing yourself to vendor failures. To avoid technology-specific solutions and services, RAPID is like ITSM: The model uses these items (and the people who support them) as resources.
- Your solution model will not necessarily look like your organization or the current reporting structure. Although some organizations already use an ITSM structure, few actually charge and manage for individual services according to business unit–specific SLAs. RAPID can help you learn how to make these changes, but many organizations do not choose to actually restructure the IT staff according to a service management model. In fact, that is part of the beauty of ITSM; even if your boss doesn't change, his or her responsibilities certainly do. You can manage resources separately from services. A model helps you relate them.
- Your model will continually change as you build and use it. Expect this and record these changes, because it is your responsibility to keep the model correct. Recording knowledge is mandatory. (What happens when John Q. ITStaffer wins the lottery or gets run over by a bus?) If the information used to make IT decisions is not recorded, there is much less benefit to building the model in the first place. IT needs change . . . get used to it.
- Your model is worthless if you do not use it. The value of a solution model comes from being able to monitor and predict changes in your IT costs, thereby making better decisions. Unless you have detailed accountability, feedback, and rewards for the goals in your model, you will fail. This use of the model will require buy-in from the executive office to each individual contributor.
- Your model may look different from RAPID. As the organization changes, its needs will often change as well. It is important for descriptive architectural models to be flexible enough to meet the needs of your specific organization. Discuss with your team the ideas presented in this book; it is full of points to dissect and argue about until you find your own solution model that meets the most common needs of all your stakeholders.

To meet the preceding requirements, you will need to implement your business-facing IT solutions model (your strategy) in a manner that your business and technology customers will both understand.

10.2 Solution modeling

Solutions are the strategic (readiness) components of a RAPID model. They represent the effect of IT strategies on the organization and all components. Every IT investment decision is a solution subject to a budget model addressing the availability of resources such as existing assets and individual program and project budgets. The most important model for IT is the annual IT budget and business plan. IT stakeholders often have a few certain expectations on which they actually insist, and if you do not meet these expectations, you will be unable to obtain funding to proceed. The sum of the expectations for each major division of the IT budget defines a solution model, which acts as a communication tool among stakeholders. A solution model is a decision-making (or auditing) plan for delivering services. The solution model divides a given IT investment into a certain number and arrangement of resources, and enforces accountability for every resource. You use these resource items to perform activities that are valuable to stakeholders.

10.2.1 Balanced scorecards

One of the most effective ways to implement and monitor accountability for strategy is the balanced scorecard model. Based on work by Robert Kaplan and David Norton at Harvard University, scorecards allow organizations to manage performance at all levels according to a core set of shared objectives, measurements, targets, and initiatives. Hundreds of organizations, including HP, Siemens, Mobil, CIGNA, and even the U.S. government, use scorecard-type management to align objectives, resources, and rewards. RAPID uses the Balanced Scorecard Functional Standards™ Release 1.0a as defined by the Balanced Scorecard Collaborative Inc. at www.bscol.com. A scorecard is a device that allows you to align resources with stakeholder strategy, and actually measure the success and likelihood of future success by implementing a particular strategy. Each person in an organization, from the executives down to the individuals, builds a scorecard that reflects his or her personal contribution to the organization's goals. (See Figure 10.2)

Each of these models works together in a RAPID decision-making (solution) model to help your organization maintain IT service management in a scalable manner while enforcing accountability with scorecards. You may choose a model other than ITSM, and you may choose a performance management system other than scorecards. For the purpose of focus, clarity, and

Figure 10.2 *RAPID balanced scorecard*

Element (BSF Perspective)	Resources (and Objectives) Strategic Intent	Activities (and Measures) Performance Metric	Constraints (and Targets) Quantifiable Goal	Expectations (and Initiatives) Action Program
Readiness (Added perspective)	➢ Communicate strategy and prepare for IT change	➢ Identify units for each performance metric (documents, requests, cost)	➢ Schema identified for collection of monitoring data	➢ Solution models (RAPID Balanced Scorecard)
Architecture (Financial)	➢ Reduce IT costs	➢ Implement IT Service Management (Annual total cost for IT)	➢ Customers using SLA (100% of business units) ➢ Reduce total IT expenditures (2% of current total)	➢ Service models (Service Standards, Service Level Agreement DB)
Process (Internal)	➢ Improve IT knowledge	➢ Document processes (# of documents in process KB)	➢ Processes documented (1 KB / program or project)	➢ Process models (Process Standards, Process KB)
Integration (Customer)	➢ Maintain satisfied customers	➢ Document support (# of documents in support KB)	➢ Customers indicate satisfied (100% of surveyed) ➢ Reuse current KBs (80% of support incidents)	➢ Operation models (Standard Operating Procedures, Support KB)
Deployment (Learning and Growth)	➢ Enable quality growth and accountability	➢ Manage changes (# of pending change requests)	➢ No-impact changes (80% of changes) ➢ Current documentation (100% of components)	➢ Component models (Change Management Standards, Change and Configuration Management DB)

IT = Information Technology, SLA = Service Level Agreement, KB = Knowledge management database, DB=database

abundance of examples, this book focuses on a model that uses both. At the very least, you should choose standards that will allow you to work effectively with your stakeholders, especially your partners. You may find that you want to rename the key solution models (an ability required by the Balanced Scorecard Standards) to attributes of more importance to your organization. Because the models are all descriptive methodologies, you will find many opportunities to customize models for any organization. The solution models describe the most important things IT does for a particular business, according to your organization's unique needs. Your methods may classify these categories differently; you manage services by organizing them with a few key results of your organization that are both important and measurable. (See Figure 10.3)

How does RAPID implement balanced scorecards?

According to the functional standards for balanced scorecard design, you may need to customize the scorecard to meet the needs of your organization. RAPID also requires some customization of the scorecards, because there are some additional requirements of the RAPID models:

- Resources implement objectives.
- Activities implement measures.
- Constraints implement targets.
- Expectations implement initiatives.

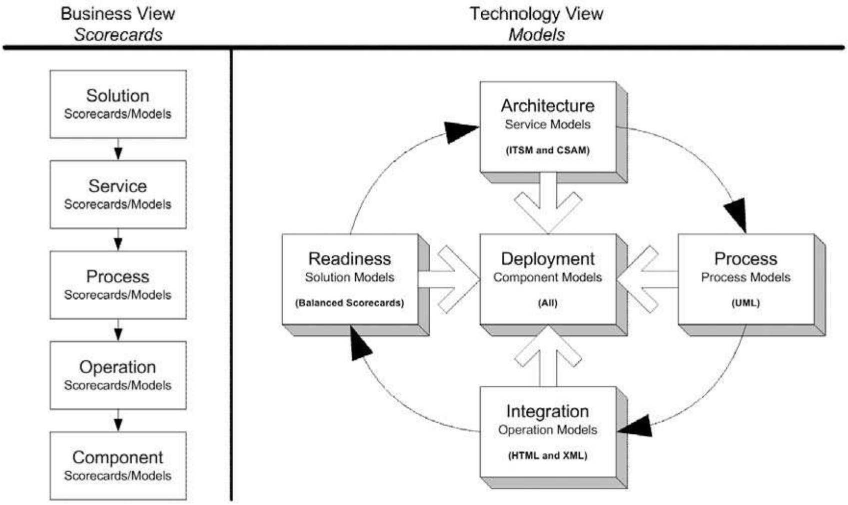


Figure 10.3 *RAPID views for business and technology modeling*

These distinctions are very important, because the balanced scorecard is only half of a RAPID model. The other half is the technical implementation linking actual resources and systems with the scorecards. Again, it is not sufficient to *have* scorecards. You must actually *use* scorecards—at every level. "To execute strategy is to execute change at all levels of an organization."[2] In the IT department, this includes not only workers but also the systems they manage. To ensure effective management of all levels of the organization, you will use a concept from balanced scorecards called *cascading*. When you cascade your balanced scorecard strategy in RAPID, you create models for not only the business execution, but for the technical standards as well. Scorecards extend a business strategy and accountability for execution from the executives to the technicians. Using ITSM allows you to extend the strategy specific to the IT department further, to the end user. Where the scorecards support the performance management needs of a vertical business organization, the solution models support the horizontal and vertical communication requirements of IT solutions.

How do you align your resources to meet strategic objectives?

When using scorecards as a management tool, it is important to choose specific objectives for each scorecard. According to the functional standards, objectives

- Describe the intent of a strategy
- Explain how a strategy becomes operational

- Join to form the building blocks for the overall strategy of the organization
- Allow strategic objectives to be aligned with perspectives

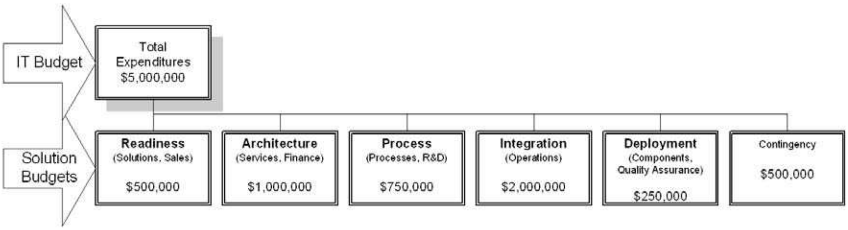
10.2.2 Solution models

RAPID also requires you to align resources with the objectives of each solution (the scorecards call these *perspectives*) in your solution model. In IT services, every resource is responsible for the resources it consumes—the ultimate peer-to-peer economic relationship. As a result, each relationship is subject to constant change as stakeholder peers change. Without a clear and consistent model, relationships are unmanageable—especially when you consider that IT services requires you to support multiple views of a given solution. Services align the IT organization along very businesslike lines that are much less subject to the churn of technology relationships. For example, the "incident management" group does not care if you use Windows, Linux, or the mainframe—the group is responsible for managing incidents of any type that affect IT customers. Only after you record an IT incident do you worry about finding the group that can fix it.

The primary measurement of resources is cost. Resources are any component useful to an IT solution with a known cost. This includes costs that are classified as capital and operational, direct and indirect, fixed and variable.[3] You can classify resources as hardware, software, staff, internal and external services, facilities, and transfers between internal cost centers. For accuracy's sake, start RAPID with a known budget and manage that budget by dividing it and assigning it to each solution. (See Figure 10.4)

To apply RAPID to the annual IT budget, for example, you must account for every resource paid for with a given budget. In your budget, account for each of the primary services each solution provides. These solution budgets become part of the constraints for activities at the next level. Using these resources, each solution proceeds to define the valuable service

Figure 10.4
Budgeting with RAPID



activities it provides to meet objectives. Again, clear accountability is required; each solution must have a group or individual owner responsible for making decisions. Once you assign resources to meet the objectives, identify the measurable activities each resource will perform.

How do you structure activities to implement strategy measures?

Like everything else, remember that even your models for IT budgeting may change over time. Some IT industry analysts (for example, Gartner in its business management of IT predictions for 2003) indicate that advanced budgeting models are on the way as "organizations will begin to implement Just-In-Time (JIT) funding for IT projects."[4] This will not happen overnight. To achieve a JIT model for IT services, you must manage IT services in a manner detailed and consistent enough to explain to business units where they best support individual IT investments. Control of the overall budget is still important in IT service management: Even if you manage every component by customer-specific SLAs, a group must continue to be responsible for IT strategy and communications. Your goal, therefore, is to find the right balance between top-level and JIT budgets for IT solutions. Using scorecards, you will measure actual periodic results for each solution.

According to the balanced scorecard functional standards, measures:

- Act as performance metrics for quantifiable progress toward objectives
- May be "lagging measures" (such as revenue) based on measured outcomes or "leading measures" (such as customer satisfaction) that predict future performance.
- Communicate the behavior needed to achieve an objective
- Become the actions used to accomplish one or more explicitly linked objectives. (BSC.1 Functional standards, available on the web at http://bscol.org/bsc_online/technology/standards/)

Again, it is not sufficient to have scorecards; you must actually use them to measure performance against objectives. Your activity view is the "hierarchy of decisions" component of RAPID. Each of the solution models (the perspectives for balanced scorecards) is an activity that supports any number of subactivities. The activities help form the next level of RAPID hierarchy as well. For instance, the activities of a solution are the services. Each solution supports a number of services. Likewise, the activities of a service model are processes. As you cascade the model, each manager has direct reports until you reach the level of *individual contributors* (nonmanagers), according to the vertical management needs of the organization. You may

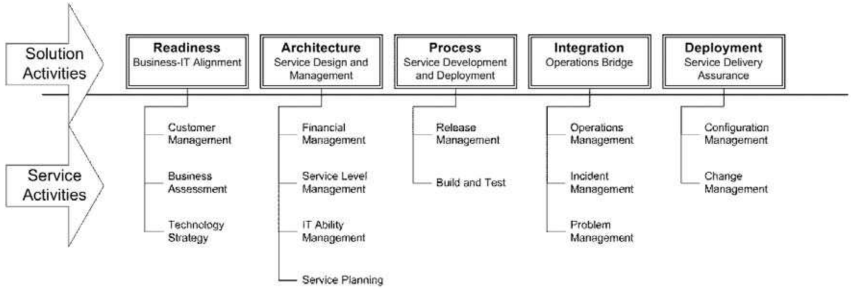


Figure 10.5
Structuring services

not have enough people to put in five layers of management, or you may have so many managers that five levels will not do. You may also choose to limit yourself to five primary management levels (as HP did), and allow managers to take multiple roles and share roles. For now, you only need to define the activities you will use to accomplish your goals in the next model. (See Figure 10.5)

Assign each resource to a certain set of activities, typically defined as being either *programs* or *projects.* Projects are short term, such as designing or implementing a new technology solution. Programs are long term, such as maintaining or managing a particular technology platform. Most activities are either performed on demand or recur on a set frequency. Allocate costs from individual resources to unique measurable activities, subject to the constraints and expectations of each stakeholder.

How do you choose constraints to implement strategy targets?

Scorecards measure strategy accomplishments by setting targets for execution. According to the functional standards, targets

- Quantify goals for measures
- Assign a specific time frame to each target
- Help the organization succeed by communicating both expectations and progress toward common goals. (BSC.1 Functional Standards, available on the web at http://bscol.org/bsc_online/technology/standards/)

Constraints are the expected limits of IT resources and activities, expressed as targets. For example, a CIO could set a constraint that annual IT service budgets will decrease by *x* percent per year, or will be *y* percent of the business revenue. You may base some targets on an expectation of 100% compliance; a CTO (Chief Technology Officer) could set a constraint that indicates an organizational preference for a particular platform or development language. You must address constraints in each component of a given IT solution: it is not sufficient to control the IT budget if this control leads to inefficient or inoperable IT solutions. To reach consensus on the more general requirements of a given solution, stakeholders must set priorities on constraints.

How will you organize expectations to implement strategy initiatives?

Scorecards execute strategy by implementing strategic initiatives. According to the scorecard functional standards, each initiative

- Is an action program to drive performance
- Focuses the activities of resource groups to support stakeholder objectives
- Must align with the strategy in the scorecard

An expectation is an outcome desired by a set of stakeholders. Each stakeholder has expectations, but the number of overall expectations should be limited and used for priority items only. You can meet many expectations through the scope of an activity or group of activities. Expectations are an explicit definition of success, and as such must be SMART: *specific, measurable, achievable, realistic,* and *time constrained.*

At the top of the reporting chain in the scorecards are solutions. They help you manage IT by organizing service management into a few key building blocks for decision-making; each of these blocks is a unique solution model. You choose solutions to match the way you communicate and manage decisions in your IT organization. Although a stakeholder may play multiple roles in different decisions—customer and end user for example—you must arrange the solution models in a manner that provides clear paths for both accountability and communications. Without these clear paths, it will be difficult to assign accountability for decision making. The stakeholders in your model are not individuals; rather, they are roles implemented as groups containing individuals. In the same way, solutions contain groups of services that facilitate communications with your stakeholders, enabling them to be accountable for decision making. (See Figure 10.6)

A solution model is the electronic representation of a balanced scorecard. It is the basic schema of your organization's performance standards as you choose to represent it in technology. You use the scorecard to manage not only performance but also the technology. It allows you to cascade

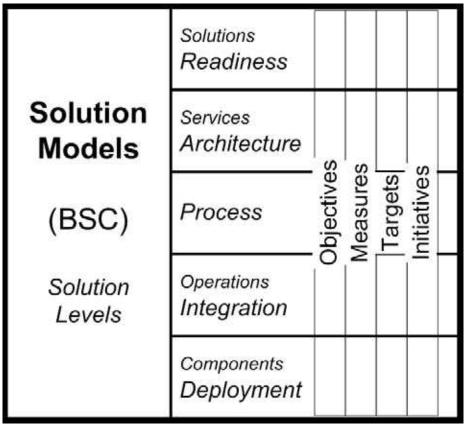


Figure 10.6
Solution model

a strategy throughout an organization. To be effective in IT, the strategy must be more than just business performance management; it must organize in a way that IT can use it to manage individual components of technology. You combine the models to create standards for IT execution, until you reach a level of detail where every change is a managed event of known proportions. Again, this takes time and effort, but the result is that you will be able to observe performance as it happens—where it happens—just in time.

How do you build a solution model?

Although it may make sense to start discussing how to implement this model in technology at this time, your only goal in the solution model is to work with stakeholders to agree on and document your top-level scorecards for your organization. Choose the four or five things you all agree are most important to focus on, and describe what it means to be successful when they are implemented. Make sure your goals are SMART and that you involve all the right stakeholders. Napkins and garage meetings are acceptable.

10.3 Service modeling

Services organize RAPID models by defining common structural (architecture) elements. Services organize IT components into classifications of work process and financial responsibility.

10.3.1 IT service methodologies

A service model is an architectural definition of the organization of a particular service. The most important service models are the service directory and SLAs. The service directory is a hierarchic catalog of IT resources and activities organized according to the needs of each stakeholder. SLAs are contracts between an IT provider and customer. Service models allow you to build the basics for managing the architecture of services.

How will you improve IT service management?

You can apply the scorecard system to your IT management by integrating it with ITSM. You can combine the scorecard system and the ideas of IT Service Management by joining these sets of best practices with a model. The model should allow you to relate best practices of ITSM, such as Incident Management, with the concrete goals of a balanced scorecard. For example, "The Incident Management team will improve customer satisfaction by 2% during the 2004 fiscal year." or "By hiring an outside consulting firm to complete the project backlog, the Service Design team will increase revenue by 4%." Scorecard types of systems allow you to set goals, measure your attainment against the goals, and change goals according to a scheduled process.

To reflect structure of an object in a model, you must be able to see the hierarchy used to organize the object. Architecture gives buildings front, side, and top views, floors, layers of information from the foundation walls to the interior wiring. Architecture is not everything, however. Some building models also reflect the living space in terms of the processes and usage of its inhabitants, such as requiring a certain number of bathrooms. RAPID uses an architectural description to organize the processes involved in IT management, but also uses architecture to separate and organize the layers of management and decision-making knowledge (including architecture and process).

How can you use RAPID to align the architecture with ITIL best practices and ITSM?

Service architecture defines the structure of and relationship between different views of a given IT component: the sum of the different views of stakeholders. It makes a lot of sense to start with a model on which your own IT

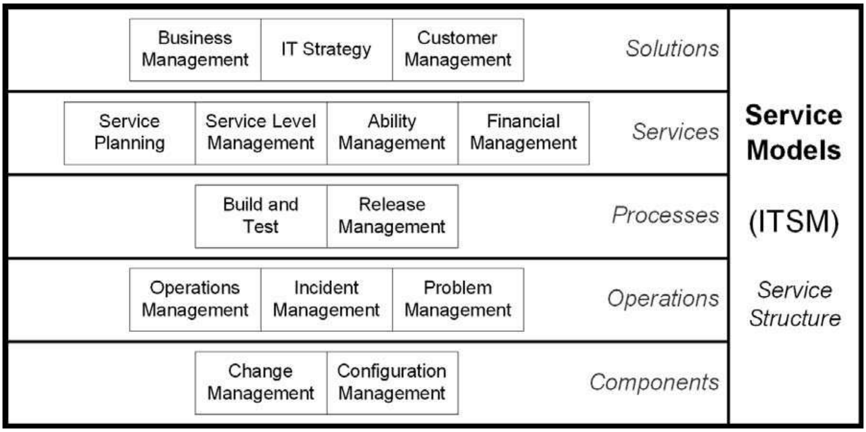


Figure 10.7
Service model

managers agree. To simplify and organize your definition of architecture, you can start with ITIL, ITSM, or RAPID to develop a list of your services. Note that although ITSM and RAPID are both focused on IT process improvement, the major contribution of these methodologies beyond ITIL is the structure—an architectural model—that lets you group similar processes into manageable services. Start from this list of services, and work toward your own categories that make sense for your organization. (See Figure 10.7)

The solution model (see Figure10.6) defined a structure for five IT solutions, including solutions readiness, services architecture, processes, operations integration, and components deployment. Each solution supports a number of services related to the overall goals of the solution. The services/architecture solution, for example, is responsible for services called *service planning, service-level management, ability management,* and *financial management.* This is somewhat confusing; at this level, services are still somewhat abstract. A logical question might be "What customer is going to buy financial management from the IT department?" The answer might be simpler than you think: Remember that every IT service is responsible for managing finances by selling services to customers. By creating a service with deep expertise in finances, most other services will choose to purchase these services from the financial management service. In the same sense, a single development project may use many services to complete the development of any new component. To focus solutions on services requires you to see each component clearly, from the solution itself to the individual components, classified and organized in a manner that accounts for the details required to manage IT according to ITSM:

- Changes are subject to both the decision hierarchy of the organization and the needs of the technology component itself. The IT budgeting process must account for the expected lifetime cost of each change.
- Each component is subject to a known set of standards called *configuration*. Changes of (or exception to) these standards are managed events that require you to consult with each decision's stakeholders.
- Each component in a model has four primary interfaces, to organize the communication needs of both the components and their stakeholders. Components must have clear interfaces (communications paths) to each stakeholder.

To implement this in software requires you to use concepts of object-oriented computer programming, such as inheritance. Inheritance is the property of a software object that allows it to have "children": subobjects of a different software class that support all the features of their parent. For example, you could start with a base class of *vehicle* and create from it classes of *car, sport utility, Cadillac, Escalade.* Object orientation lets us arrange these classes in a manner that allows the car to inherit all the properties of a vehicle, such as an engine, while creating special new abilities, such as a frame suitable for off-road travel.

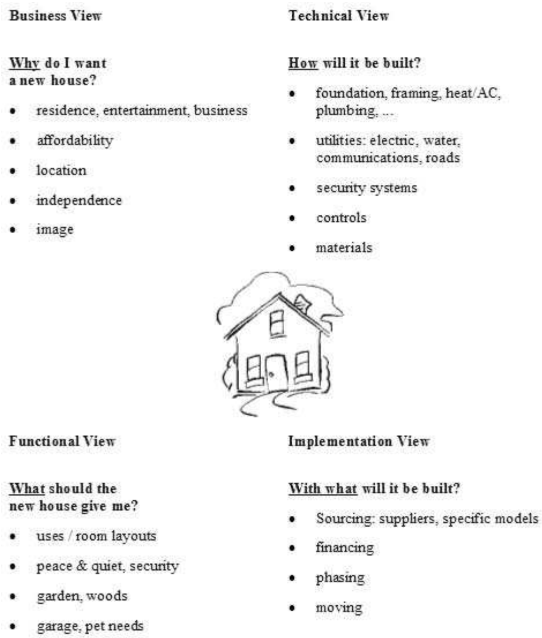


Figure 10.8 *HP Global Method Views (Source: HP)*

How can you use the HP Global Method to improve your service architecture model?

An architect knows the questions to ask, how to organize those questions, and can provide useful knowledge about the answers. The architecture is a high-level picture, organized in different views that represent major aspects of the house (the architectural concept). HPGM helps HP understand and explain customer architectural requirements. (See Figure 10.8)

10.4 Process modeling

Processes define the repeatable activities, programs, methods, and common/best practices (process) of IT component development. This is some of the most important knowledge in the IT department, because it represents the results of the individual component development projects occurring throughout IT. You must capture this information for later use; it is a valuable description of the requirements and relationships between different projects. Once you start using RAPID, you will want to model your IT processes according to flexible business modeling rules, such as those defined in the Unified Modeling Language (UML).

10.4.1 Unified Modeling Language

Standards such as UML are important when modeling processes because they help enforce consistency of language and grammar between dissimilar processes; in short, you can use standards such as UML everywhere. It is not limited to use for IT processes; rather, it is important for use in standard business processes. RosettaNet is an association of more than 400 leading companies manufacturing electronic components, information technology, and semiconductors working to design, develop, and promote open e-business process standards. Much like the Rosetta Stone, which helped modern scholars decipher ancient Egyptian hieroglyphics in the 1800s, RosettaNet establishes a global language for electronic business information transfers. Participation in RosettaNet opens the lines of digital communication so that companies can be part of the digital economy. RosettaNet uses standards such as UML and XML to create dynamic, flexible trading networks, leading to increased operational efficiency and new business opportunities. UML lets you document complex processes with multiple views according to commonly accepted standards of IT architectural modeling. Chapter 4 presented the UML static structure view of the RAPID model components, and this chapter explores ways of using UML to document your important processes.

How is UML valuable to the IT department?

UML has many different views; you use different views to model different types of components, as shown in Table 10.1.

10.4.2 Process models

One of the challenges in UML is how to organize the many different types of information you can store in UML documentation:

Table 10.1 *Object Management Group standard UML diagrams*

RAPID Class	UML View	Description
Solution	Use case	Use cases describe how stakeholders and other actors (e.g., time, other components) interact with each other
Service	Static structure	Static structures make up diagrams representing real-world concepts and the relationships between them, or classes that decompose a software system into parts
Process	Sequence	Sequences show the resources participating in an activity and the activities they generate, arranged in a time sequence
	Statechart	Statechart diagrams show the normal process of unique states (milestones) a component may exhibit during its life
Operation	Collaboration	Collaborations show the relationships (such as class, dependencies, usage) between component roles in terms of information exchange and data flow
	Deployment	Use to show the structure of operations and communicate how you deploy components that make up an operation
Component	Activity	Use to define the internal process flow of components; each activity should be an independent process, such as taking an employment application or adding a document to a knowledge database
	Component	Use to document relationships between individual component

10.5 Operations modeling

Operations manage the relationships (integration) of organizational knowledge and support standard operating procedures for IT component management.

10.5.1 Internet standards, HTML, and XML

Internet standards are important, as they are the *lingua franca* of global communications. To remain competitive in the Internet age, it is important to speak the Internet language. Some organizations will choose to use more of these standards, and some less, but formatting data and processes with the Internet is certainly important.

10.5.2 Operation models

How do you arrange work processes?

Work processes are the things the operational IT department does every day. One way to model this information is in process flowcharts. Structure your architectural components into detailed processes that reflect the responsibilities of each stakeholder. (See Figure 10.9)

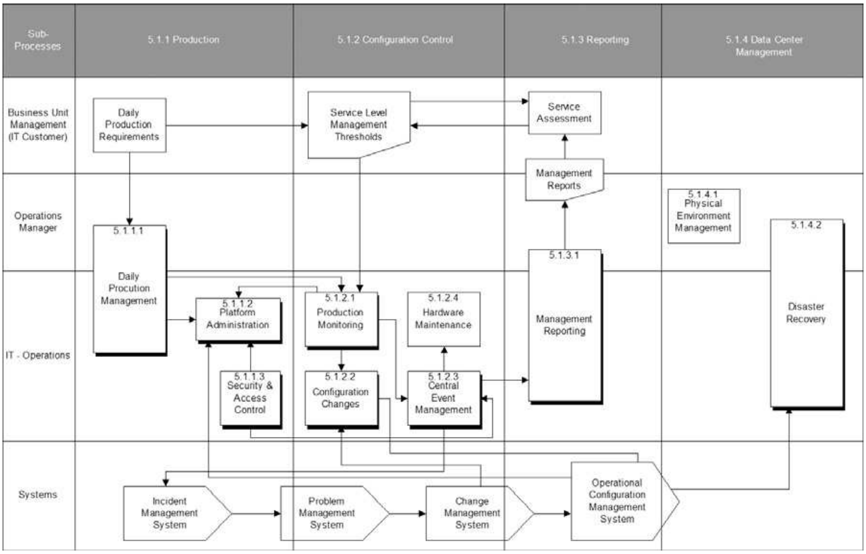


Figure 10.9
Work process (example)

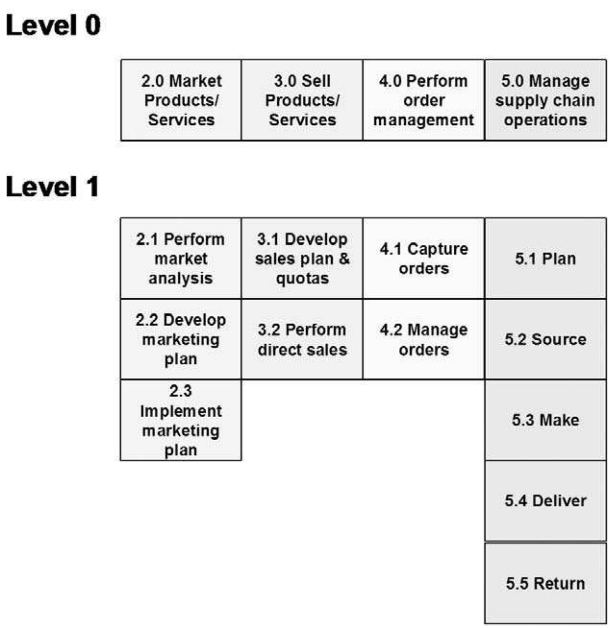


Figure 10.10 *Layering models (Source: HP)*

Once you have developed your work processes for the day-to-day operations of the IT department, you must move forward to implement the operational standards that will govern IT.

How do you organize standard operations?

HP uses a methodology called ValuePrint. The model allows you to relate multiple layers of a complex business process (say the integration of architecture and process itself). You can represent this relationship graphically using a very simple method of boxes and hierarchical numbering. You use a similar method when displaying RAPID models. (See Figure 10.10)

You may need to build many layers, depending on the complexity of your organization. Once you have organized your standards, then continue to define the substance of each policy and standard. (See Figure 10.11)

How can a model use HTML and XML?

Again, the web provides an excellent medium for publishing and maintaining standards. HP organizes standards into "domains" that define each operational component of IT. Each domain of knowledge classifies similar types

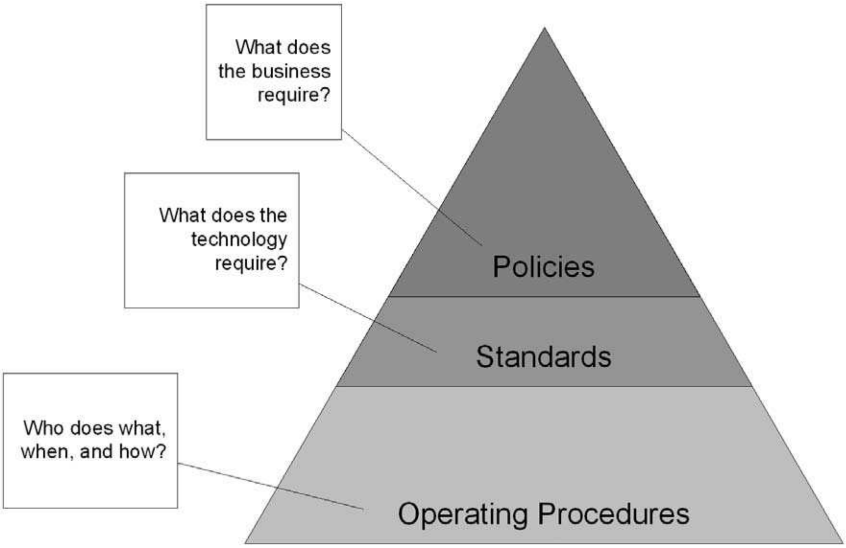
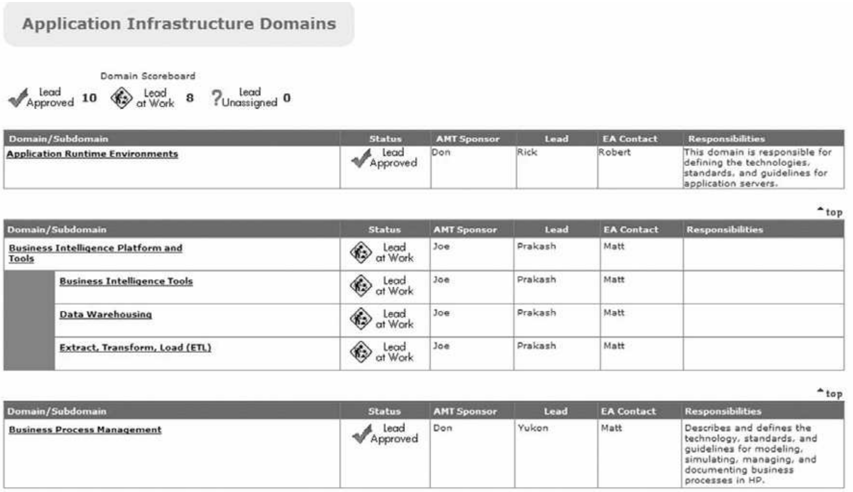


Figure 10.11
Policies, standards, and procedures



Application Infrastructure Domains

Domain Scoreboard

Lead Approved 10 Lead at Work 8 Lead Unassigned 0

Domain/Subdomain	Status	AMT Sponsor	Lead	EA Contact	Responsibilities
Application Runtime Environments	Lead Approved	Don	Rick	Robert	This domain is responsible for defining the technologies, standards, and guidelines for application servers.

top

Domain/Subdomain	Status	AMT Sponsor	Lead	EA Contact	Responsibilities
Business Intelligence Platform and Tools	Lead at Work	Joe	Prakash	Matt	
Business Intelligence Tools	Lead at Work	Joe	Prakash	Matt	
Data Warehousing	Lead at Work	Joe	Prakash	Matt	
Extract, Transform, Load (ETL)	Lead at Work	Joe	Prakash	Matt	

top

Domain/Subdomain	Status	AMT Sponsor	Lead	EA Contact	Responsibilities
Business Process Management	Lead Approved	Don	Yukon	Matt	Describes and defines the technology, standards, and guidelines for modeling, simulating, managing, and documenting business processes in HP.

Figure 10.12
HP Enterprise Architecture—domain view

of information. The knowledge domain is a set of principles, technologies, standards, products, and configurations that is part of HP. Each domain's information is available through the HP Enterprise Architecture site by selecting a layer of the architecture. (See Figure 10.12)

Each domain includes a *domain definition template* allowing domain owners to describe the unique information required in each knowledge domain. The template divides the focus into three areas:

- Domain level: scope, key initiatives, audience, contacts
- Product level: governing technologies, products, product positions
- Background: supporting documentation, links

Domain owners (called "leads") complete information about each of the most important domain knowledge areas. This information allows stakeholders to find the information they need in order to interact with the knowledge domains. This is an important knowledge contribution, because this information allows customer-facing HP staff to see the effects of this particular style of IT management in a working model they can use with customers. Domains describe the information given in Table 10.2. (See Table 10.2)

By selecting a particular domain at the web site, stakeholders are presented with a detailed description of its current state, based on the topics just shown in the table. This web-based interface allows anyone on the internal HP network with a web browser to interact with the defined architecture of the IT department. This interaction allows HP to enact well-documented standards that provide direction to everyone working with the organization's IT department. (See Figure 10.13)

How do you set priorities on your standards?

It is important to maintain standards according to commonly understood levels of priority. These priority levels help stakeholders understand whether a standard is "just a suggestion" or "the immutable law." The initial HP domain definition template requires a domain's creator to provide information critical to other stakeholders of the domain. As this information is completed, each standard is "positioned" by the HP Enterprise Architecture Group according to its maturity and supportability levels. This allows HP staff to identify immediately how a particular standard affects them.

Position	Description
Approved	Current standard; support available.
Evaluation	Early assessment activity; not for production use. Re-positioning will occur after the evaluation.
Pilot	Limited production use. Approval from architecture management team required.

Continued

POC (proof of concept)	In research phase; not for production use. Approval from architecture management team required.
Preferred	The favored choice when multiple approved standards exist. When multiple approved products exist, prescriptive guidelines are provided to direct usage.
Prohibited	Use not permitted. The prohibited position is provided to explicitly define Compaq's restriction on use of a given product. Any product without an explicit position listed is deemed prohibited.
Restricted	Continued use for business critical applications is allowed, but new uses are prohibited without express written consent of the domain architect.
Sunset	Being phased out; no future support or development.

Table 10.2
Primary topics of knowledge domains (Source: HP)

Topic	Description
Domain name (required)	Domain names are assigned by the Enterprise Architecture Group.
Domain architect (required)	This is the person primarily responsible for the content and direction of the domain. Name and email address must be provided.
Key contributors (optional)	Key contributors are those who provide major input and direction to the domain architect.
Scope (required)	The scope defines the activities that are the direct responsibility of the domain.
Key initiatives (required)	Key initiatives represent the direction and task(s) essential for the progression of the domain and should encompass a 12- to 18-month horizon.
Audience (required)	The audience is defined as those who are affected by decisions and actions of the domain.
Additional relevant domain documents (required for applications)	Provide documents for such things as application landscapes, roadmaps, etc. These are placed in a file share—secured such that domain owners will have update authority.
Primary domain interactions (required)	Domain interactions are defined as other enterprise architecture domains and/or subdomains in the architecture that must interact or coordinate with the domain. This interaction is upstream or downstream from the domain.
Links and references (optional)	Links to supporting documentation, web site URLs, or other relevant information. Provide a link name and an address (UNC, URL, etc.).

Continued

Table 10.2 *(Continued)*

Governing technology and products (required for infrastructure)	Each domain spans a number of technology categories whose usage is governed by the domain. These groupings should be logical and may depend on the organization. There are two categories of products, those owned by the domain and those referenced by the domain. Owned products are those for which the domain defines the direction, usage, and position. Referenced products are those that are used within the domain but whose direction, usage, and position are defined in another domain.

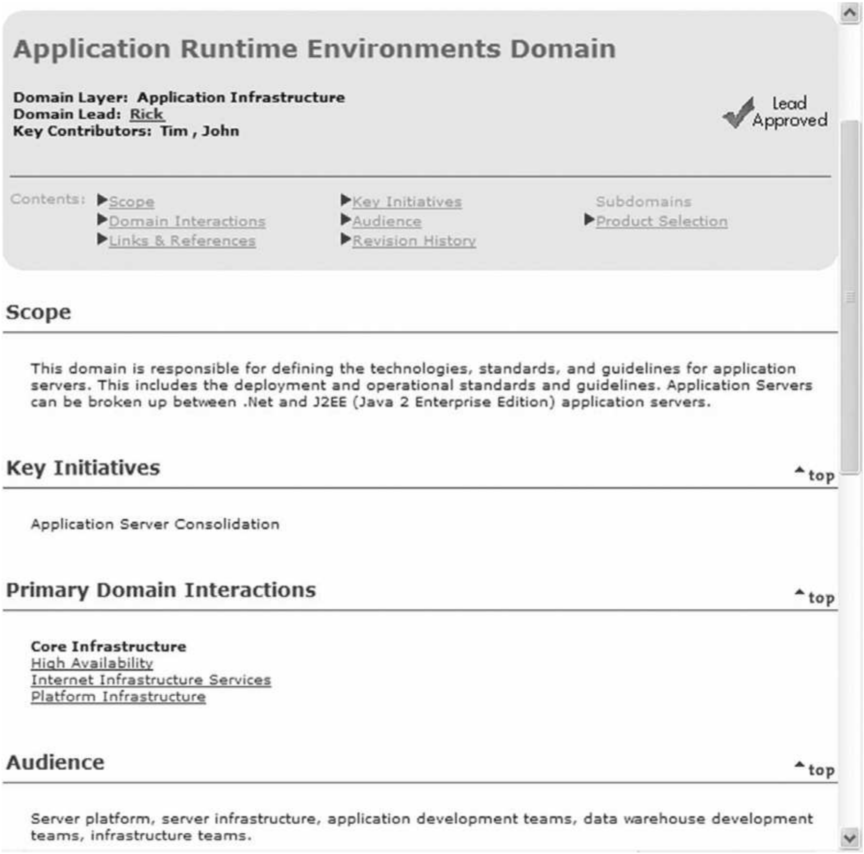


Figure 10.13 *HP Enterprise Architecture—domain layer view (Source: HP)*

What else is important for standards?

One important part you cannot forget is to document your standards. RAPID models provide one method of accomplishing this task. Remember to maintain your documents in a way that lets you keep them up to date. At the bottom of each document in the HP Architecture Library is the following information: (See Figure 10.14)

Figure 10.14 *Document metainformation*

Revision History ^top

Type	Domain	Initiation Date	Tracking ID#	Title of Document	Owner	Key Contributors	Strategy Lead
P	Information Security	26 Feb 1997	ITP-SEC-004	Password Management Policy	MSDD Information Security Policy Team		Sherry

Document Information ^top

Issuing Organization: Information Technology	**Document ID#:** ITP-SEC-004
Sponsor: Fred	**Last Review Date:** **Revision Date:**
Contact: MSDD Information Security Policy Team	**Effective Date:** 25-Sep-2002

References

1. Scott Ambler (1999), Software Development Magazine, September 1999, available on the Web at www.sdmagazine.com/documents/s=826/sdm9909p/.
2. D. Norton (2002), Managing Strategy Is Managing Change, *Balanced Scorecard Report,* 4(1): 1. Available on the Web at www.bscol.com or from Harvard Business School Publishing online.
3. I. Macfarlan, and C. Rudd (2001), Financial Management for IT Services. In Derek Cambray, (ed.), *IT Service Management* (UK: itSMF Ltd), Chapter 13, p. 49.
4. B. Gomolski and R. Mack, (2002), available on the Web at www.gartner.com/DisplayDocument?doc_cd=111560 and accessed on.

This Page Intentionally Left Blank

11

RAPID IT Components

> *What we need to do is learn to work in the system, by which I mean that everybody, every team, every platform, every division, every component is there not for individual competitive profit or recognition, but for contribution to the system as a whole on a win-win basis. (W. Edwards Deming)*

The component is the "cell" of RAPID. As in nature, the cell is the basis for the overall functioning of the complete system. Some components (blood transportation or network infrastructure) are extremely important to the functioning of the system; loss of these components can have devastating effects. Some components are more sensitive to the effects of change than others are; speeded up to the point of exhaustion, these components become cancerous. Your goal is to build healthy components in healthy system.

The biggest challenge of components is determining how to define them. It requires a great deal of agreement between stakeholders to draw the "lines" around individual components in a model. Components address the decision-making capabilities of every stakeholder, and the nature of components (like any other computing "object") is that each one you define may appear quite different in makeup:

- A personal computing device (with operating system and core applications)
- A high-speed network connection (with 1-gigabyte data transfer and security smartcard)
- A personal storage folder (with 100-megabyte storage limit and web-based access)
- An email address (with 100-megabyte storage limit)
- Access to standard employee web services (including News, Human Resources, Payroll, and Performance Management)

You also cannot afford to manage every object in the IT environment as a component; the overhead would be too great. To be effective in any form of IT Service Management (ITSM) you must have a consistent definition for each component in the organization's Configuration Management Database (CMDB); the effectiveness of Configuration Management as a whole depends on good CMDB design. The component of a RAPID model is used, among other things, to define "where you draw the line" around decision-making, Configuration Management, and ITSM in general.

Component definition should consider the needs of all stakeholders and involve them in component management, but define the stakeholders selectively for each component. The impact of decisions may vary among stakeholders, and roles of individual stakeholders often change over time. Agree to assign priority to certain groups or classes, such as the business and technical decision makers and the users.

There are two primary classes or groups of stakeholders you should consider when designing or changing components: the "generic" stakeholder(s) and the decision maker(s). They are not always individuals. The stakeholder often includes users, organizational or business units, technicians, and other large groups of individuals. The decision maker is usually a small group of IT personnel and a customer contact. In some cases, such as routine responses to incidents and problems, remember that the decision-maker for a particular component may be a single individual.

Keeping track of all the parts of a RAPID model-components, operations, processes, services, and solutions—is a challenging task for even the most agile IT organizations. Many organizations will find these business and technical rules embedded in the years of previous IT applications and infrastructure. For organizations requiring ongoing measurable response to stakeholder and component issues, it is necessary to describe an electronic method of maintaining this information. One particular technology solution stands out as providing the basis for effective configuration and component management.

11.1 Finding the right components: Electronic directories and the agile enterprise

Most organizations use tree structures to describe governance and communications; a look at any organization chart will confirm this. The electronic directory service provides a unique ability—tree and leaf management—that is critical to supporting complex electronic decision-making models. Decision-making trees are not new to business, and directory trees are not new to IT; the new part is the creation of "forests" that include many electronic information trees.

Enterprise use of early electronic directory technology, like Novell Directory Services (NDS) started with basic "log on" and server security. Now, companies such as HP and Microsoft use electronic directories everywhere. Directories are the glue that binds together security models, benefits, purchasing, expense reimbursement, performance management, and electronic communications. The Lightweight Directory Access Protocol (LDAP) Internet standard allows organizations to create flexible *enterprise architectures* by defining a logical "central hub" to organize the various sources of information in an organization.

At least five challenges face the next generation of enterprise directory architecture:

1. *Readiness:* To service an e-society, it will be necessary to organize electronic services into a hierarchy of recognizable "packages" or solutions that consumers can purchase and vendors can support. Organizations may be concerned about tying the business so closely to technology. The business value of directory services as an integration tool will be particularly difficult to explain in organizations without prior experience in both directory management and enterprise application integration.
2. *Architecture:* Many organizations do not understand the value of their information, especially their architectural standards and best practices; it is rare indeed to find an IT organization (outside of some financial institutions) that have a good history of definitive IT policies and solid, current IT documentation. The architecture of an electronic directory may affect the future operations of an entire organization, as it may enforce business and security rules. Groups and individuals in the organization may not be able to agree on common practices for owning and managing information. For instance, the HR department usually owns the primary electronic record of every employee (for good reason, as they often own the risk for violating rules involving employee information).
3. *Process:* The process for implementing directory-enabled management is always customer specific, and requires participation of diverse stakeholders. As you provide more electronic services directly to the consumer, you will find that directories often store the massive volume of employee and customer information used by web sites. For this reason, directory information may be subject to the impacts of hard-to-predict future events like mergers, acquisitions, divestitures, reorganizations, and other changes to reporting processes. Each organization has a different formula

for success, and the Enterprise Directory strategy must be able to adapt to changing strategies or other "process flux".

4. *Integration:* Creating a strategy for maintaining such a primary organizational role may be challenging for IT staff and management. Much like accounting software and electronic mail, the directory is becoming a mission critical application. Unlike accounting and mail software, every physical site and every computing device requires instant access to directory services, if integration of physical and logical security is important (such when you use electronic keys or smartcards to restrict access to physical or virtual areas). For the directory to enforce security and business processes, directory structures must be predictable and well managed. Integrate the directory services carefully, as you should assume that the structure of the directory changes over time.
5. *Deployment:* Creating a directory-enabled management environment requires extensive cooperation, often starting with the core infrastructure and then iteratively integrating enterprise operations.

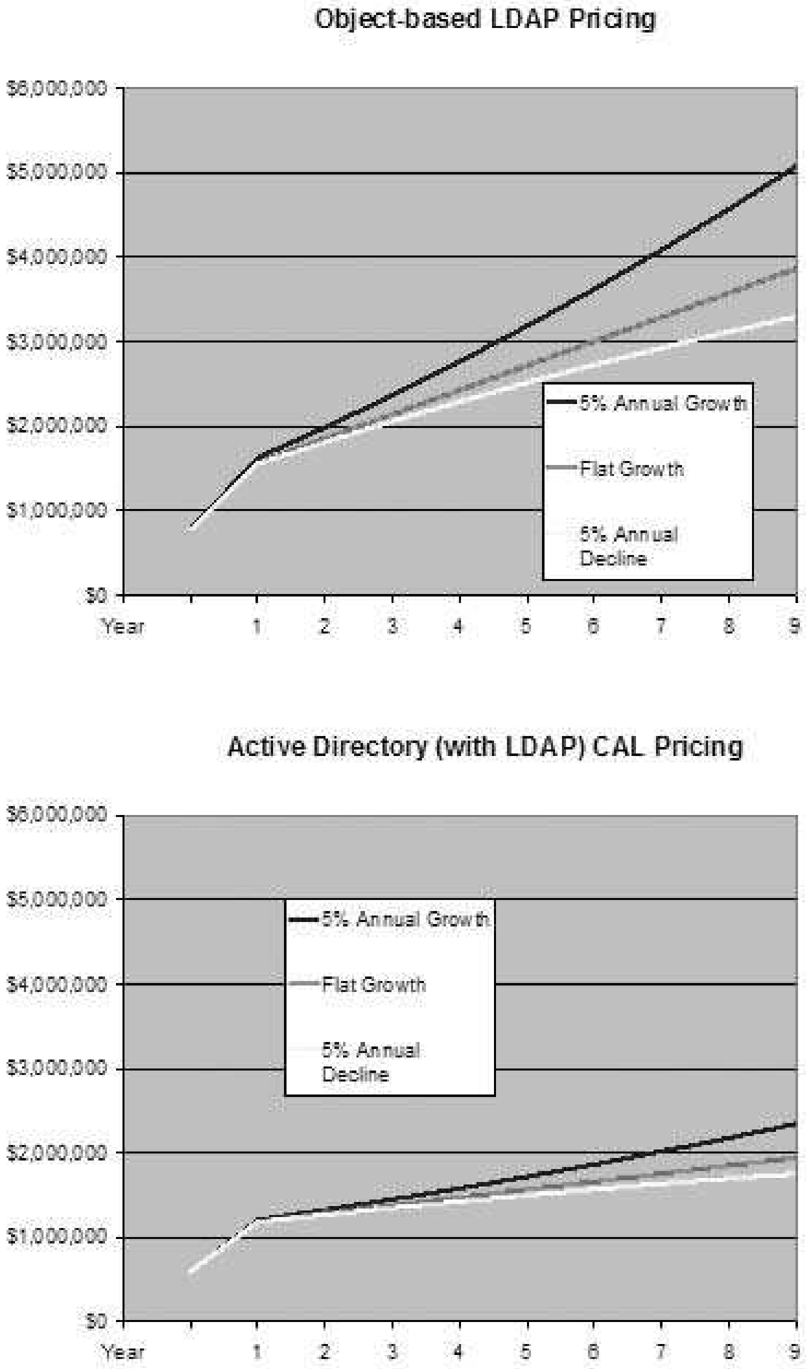
11.1.1 Readiness: Business value of directory integration

In late 1997, Microsoft Consulting Services began to focus its first resources on Windows NT 5 and the Active Directory. Since that time, Windows 2000 has moved Microsoft from being the "vendor without a directory" to a leader in providing distributed directory services. Windows Server 2003 expands the technical abilities of the Active Directory, and continued deployment of LDAP-enabled Windows domain controllers may help Microsoft's drive forward in this space. The potential impact of this change on the directory services market, coupled with that of Linux-based OpenLDAP servers, is that any vendor currently selling "directory only" products will be severely limited in the ability to maintain market share in accounts with large Microsoft or Linux environments. Some of these vendors are already in crisis, affected by a "double whammy" of the economy and the technology squeeze.

The graphs in Figure 11.1 compare the March 2003 proposal pricing provided by an HP partner specializing in directory and metadirectory products, and the estimated cost of a similarly timed strategy using Windows Client Access licenses: (See Figure 11.1)

The business value of directory services as an integration tool is not widely agreed on, but the industry is certainly building an interest in understanding it.

Figure 11.1
License pricing: LDAP vendors versus Microsoft



It is not difficult, however, to explain the potential business value of consolidating directories and the information they store. One of the most interesting effects of Microsoft's strategy of product integration is the consolidation of applications this strategy causes in customer organizations. Using platform-integrated directory services to eliminate our customers' cost for supporting multiple LDAP vendors significantly reduces software licensing costs. Many customer projects, for example, can share the costs (software, hardware, implementation, and support) associated with the Active Directory, as many Windows-based products (such as Microsoft Exchange) either require customers to purchase Active Directory and/or Windows Server licenses now, or in the future. Customers also benefit from Microsoft's server-based licensing for support of Internet-facing customer directories. Many customers will prefer the flat US $20,000 per server for a Microsoft Windows Internet Connector license, compared with the LDAP vendor's $0.75 per user option, when they exceed the breakeven point by supporting more than 27,000 customer entries.

Organizations using Microsoft or Open Source technologies must consider the potential value of consolidating all their directory services, based on software costs alone. However, the potential benefits do not stop there. Hidden in directory services is a powerful business value tool that only a few customers have seriously explored. The biggest potential business value of a directory is the integration of enterprise architecture and processes.

11.1.2 Architecture: Advanced directory structures

Advanced directory service architectures require careful planning. When designing these models (especially those for larger organizations), it is important to develop methods for managing both the horizontal and vertical processes the organization supports. Groups and individuals in the organization may not be able to agree on common rules for information ownership, transfer, privacy, and security. To extend directory services to support this wealth of information requires a simplified storage hierarchy. (See Figure 11.2)

Each organization defines a unique and specific "organization chart" to structure its enterprise directories. However, there are naturally many ways to look at the same information; the point-in-time organization often seems arbitrary, in the sense that people's roles and corresponding directory memberships often change.

The enterprise directory allows organizations to store and manage many different types of enterprise information. In the organization shown in

Figure 11.2
Enterprise directory structures

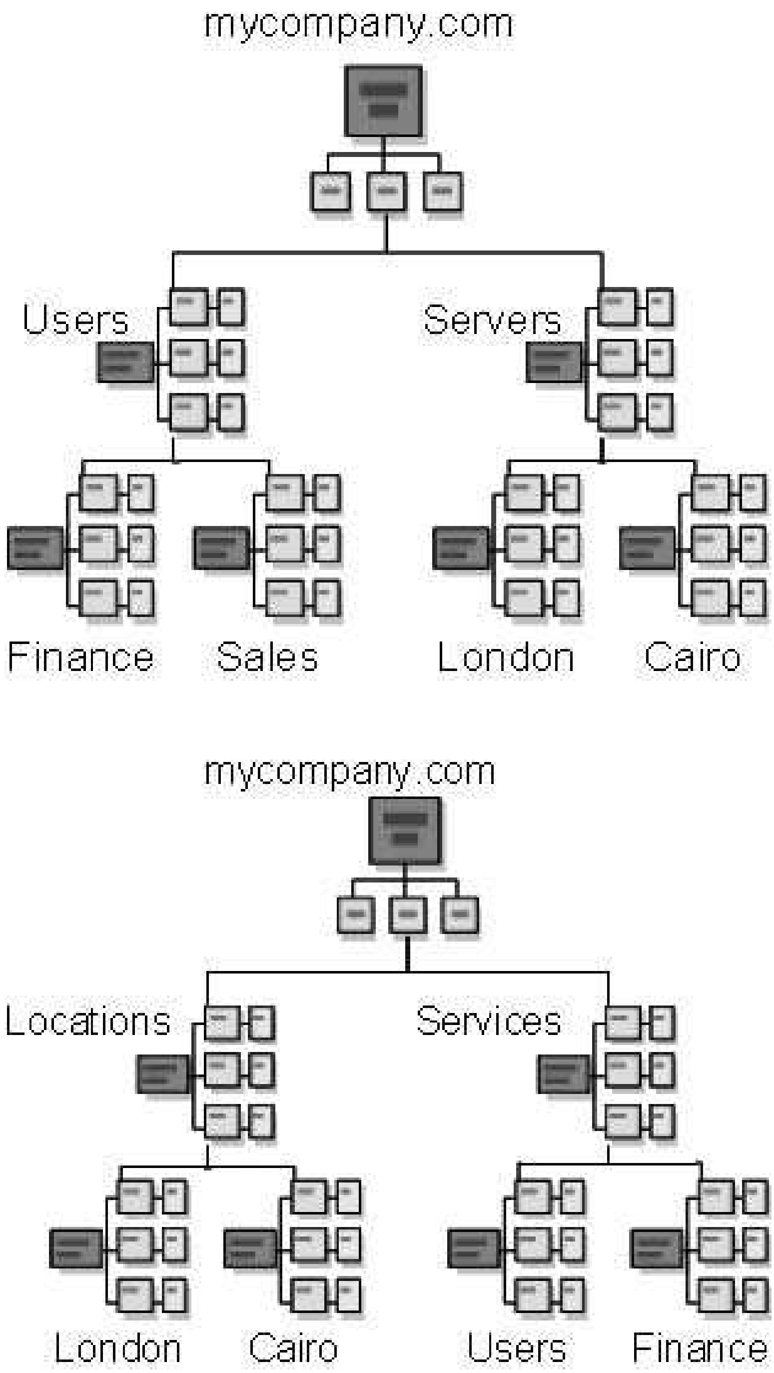


Figure 11.2, the directory allows you to move users from the London organizational unit (OU) to the Cairo OU. You can also move servers from Cairo to the finance department, but how would you troubleshoot a problem involving the "finance servers in Cairo"? Even more complex is the natural tendency to focus on either a centralized or decentralized management model; in fact, the directory allows you to do both. When designing directories, it is important to consider where you will store both centralized management policies and the localized policies that vary by business unit, technology platform, location, and so forth. Consider the possibilities provided by both models; an advanced strategy may include elements of both centralized and decentralized models.

Directories are flexible—made for change. As more applications integrate with LDAP by storing key enterprise management information in a directory, customers can change business structures and processes without requiring frequent changes to technology applications. A single user can be associated with any number of organizational resources and business and technology services such as hardware, software, benefits, and payroll. Use directories to assign users or groups certain rights, such as the ability to update personal information, manage group membership, or to limit a user's ability to read or change information. (See Figure 11.3)

The directory also allows you to group these items. For example, a user can belong to groups providing him or her with all the standard resources for that user's location, role, and position. Many customers have seen benefits in

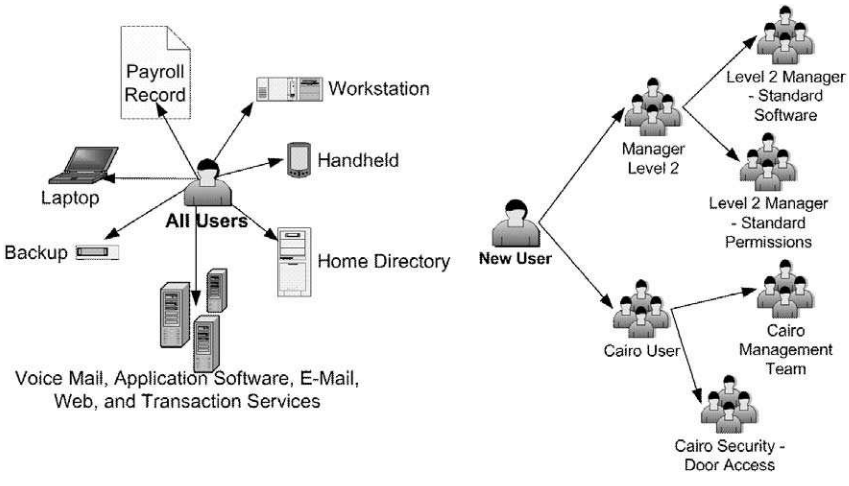


Figure 11.3
A flexible directory

grouping software—for example, creating a "core desktop software" group might contain groups for Microsoft Office, Adobe Acrobat, Internet Explorer, and Windows. The benefit of storing this information electronically (and determining ways to enforce directory policy for each of these resources) is that you can also query it—much like a database—to find out how many users the organization in fact approved to use those expensive copies of Office.

The best architecture for a directory service depends on the organization, but consistent rules affect directory design. Some *long-running processes* (cross-functional ones such as hiring, transfers, exits, address changes)—are useful references when considering the most effective structure for an enterprise directory.

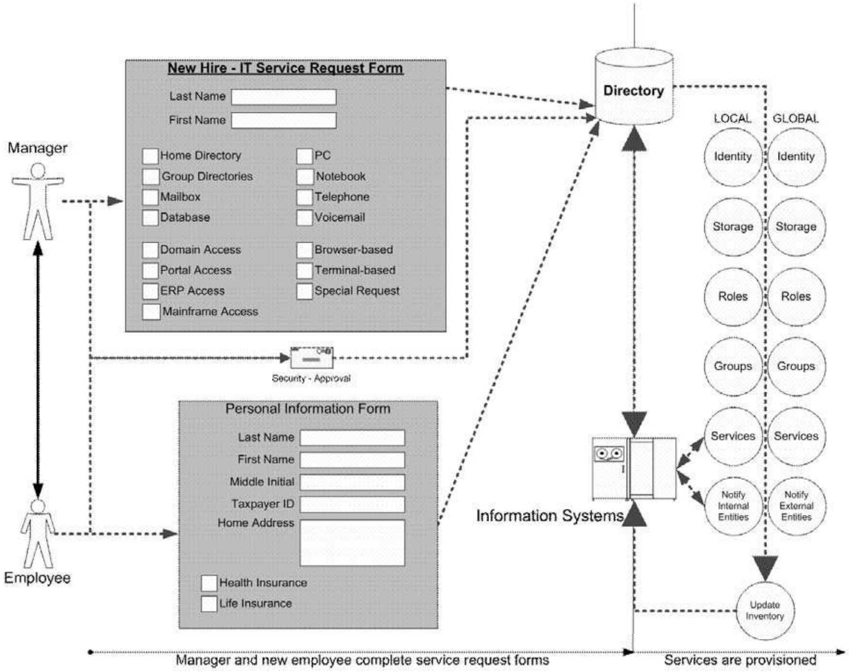
11.1.3 Process: Managing change

A long-running process—such as hiring a new employee—provides an excellent reference for defining enterprise directory strategies. For example, consider the number of forms it takes to process the hiring of a new employee: management requests, advertising, interviews, new hire authorization, government forms, human resources, benefits, payroll, and other forms supporting business and technology assets. Data entry clerks (in any number of departments) must often transfer this information from paper forms completed by the new employee or their manager. For forms that departments must share, the routing and approval process for a single hire can take hours of work and weeks of time. One of the biggest challenges is that the information gathered during long-running and cross-functional processes (how many times must you really write your name and home address, and what happens if you transpose a digit on a single copy?) is stored in many different departments and their IT systems. Each time any of this information changes (and some of it—such as resource needs—changes often), you must track down all the copies or take the risk of using "stale" information.

An electronic directory allows you to centralize long-running processes, while increasing the integration and effectiveness of decentralized security and management models. Directories are adept at managing the constant change and churn of the business and the IT department. By using web applications to store the most critical information for each *information component* of business and technology in enterprise directories, organizations significantly improve business and technology agility. For instance, an organization can centralize all its employee and customer information in a directory. This allows a single point of security (and storage) when hiring new employees, updating customer names, addresses, billing and payment data, or other commonly used information. The directory also lets you share or assign control (security) of information and publish encryption certificates for use when safeguarding privacy and confidentiality.

Simplifying complex and long-running processes is by no means simple. It requires you to find common elements of tasks that cross boundaries between organizational divisions, departments, teams, and individual managers. Process consolidation also requires cooperation between the core business units and the IT providers. By integrating a customer's knowledge of its business and core business processes with HP's experience in technology solutions such as enterprise directories and web services, HP consultants help customers implement electronic forms and global provisioning pipelines to eliminate the form completion and data entry efforts of workflow in long-running processes. Integrating this information is invaluable, because it has the potential of reducing work throughout the enterprise. (See Figure 11.4)

Figure 11.4
Managing change through the directory



11.1.4 Integration: Global provisioning pipelines

Organizations can use the enterprise directory as a centralized pipeline to integrate information processing throughout the organization. This global provisioning pipeline strategy allows the enterprise directory to serve as an authoritative, organized source of information on important cross-functional processes.

Consider the following elements of an IT-integrated new hire process:

- Find all web services belonging to the "HR department" and the "IT department."
- Given a particular employee, find the hiring and approving manager's email addresses and send each message requesting authorization.
- Record the user's personal data in the directory for use by all authorized information users.
- Get the location of the web service(s) providing the "enterprise purchasing catalog."
- Get the location of the web service(s) providing "new hire setup services."
- Update the status of the "new user provisioning" process by adding to the directory a workstation object, a user object, and a set of desktop software application package objects.
- Update the status of the "new hire provisioning" process by adding to the directory an HR object, a benefits object, and a payroll object.

Storing this information in an electronic directory ensures that applications and services have a consistent place to find the latest information affecting the business rules of the organization. In this model, if a rule changes—because of changes in business process, stakeholders, or technology—the updates occur "just in time."

A provisioning pipeline (Figure 11.5) simplifies the process of new hires:

1. A hiring manager begins the process by using an electronic form in the management portal to describe the type of employee the organization wishes to hire and selects services from the directory such as PCs, email accounts, telephone service, an office desk, pay grade, and so forth.
2. Applicants enter information into electronic forms, generated by using the directory to find and integrate web services for each member of the "new hire process" directory group.

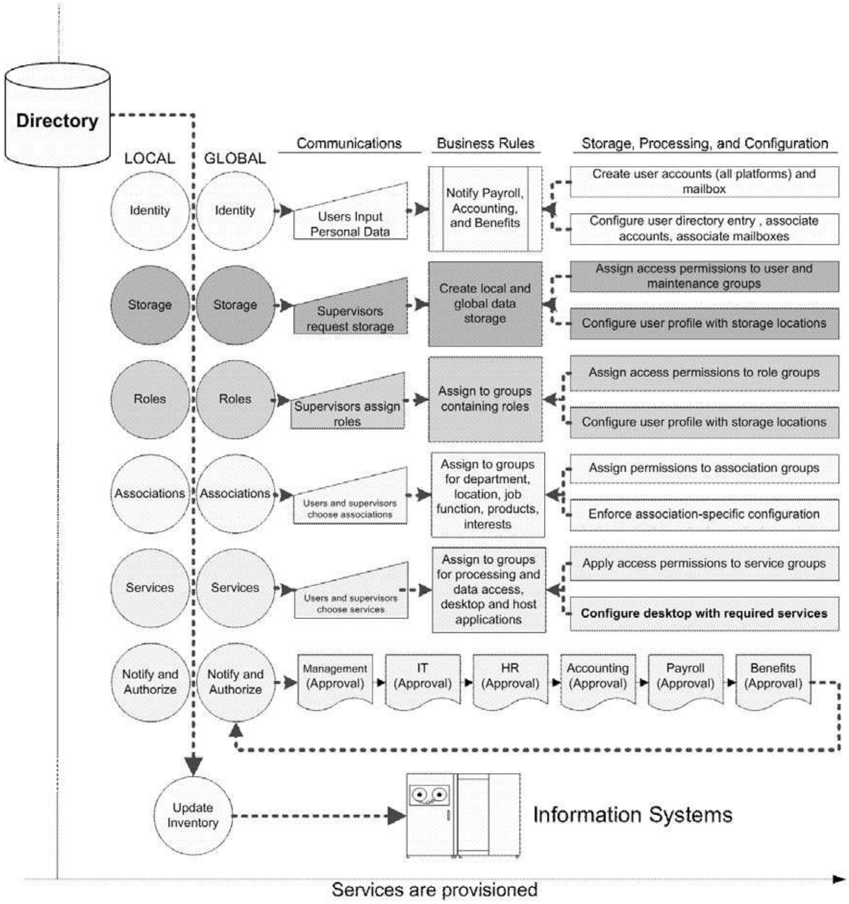


Figure 11.5
The provisioning pipeline

3. Each form processes this information according to its owner's business rules, using the directory as a source for information on the new hire. Each process should also publish a subset of new directory information for use by other processes.

The dance of provisioning that ensues is a choreographed electronic version of the former "mob rules" business process. The directory makes important "just-in-time" information available to all its stakeholders, not just the ones tied to the back office systems. For example, the directory provides a consistent location to store business rules like reporting structure, message routing, and workflow. (See Figure 11.5)

Table 11.1 *Forms completed in the new hire process*

Process Groups	Example Directory Subgroups and Web Forms
Critical information	Name, home address, telephone numbers, emergency contacts
Government requirements	Federal information, state/province information, local information
HR requirements	Payroll, benefits, demographics
Workplace requirements	Workspace, building access
IT requirements	Workstation, email account, home directories, group directories, database access, software applications, passwords

As outlined in Table 11.1, the list of forms completed by managers and employees for a long-running process such as a "new hire" may be quite complex. (See Table 11.1)

For maximum scalability and partner integration, manage relationships between directories (there may be many) using directory replication and LDAP referrals (the "call-forwarding" feature of directory services). Referrals are extremely important, because this function allows you to store a subset of your partner or customer's information. When you want to consolidate information, the directory can also track "pointers" that help applications locate updated or detailed information.

Organizations can also store universal resource locators (URLs) in the directory service. These URLs refer to network-available information and services, limiting the number of copies stored for larger information sources such as multimedia files and databases. Some operating systems, such as Windows 2000 and Windows XP, let you store configuration policies in the database as well; for instance, customers can use application policies to force workstations in the payroll department to install the latest payroll software.

11.1.5 Deployment: Adaptive enterprise architecture

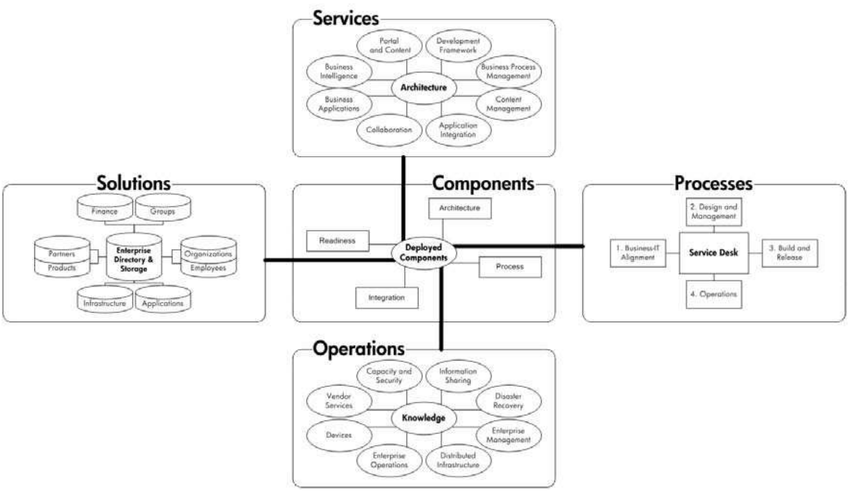
Imagine the nightmare of managing an organization with thousands (or millions?) of web services to complement each possible grouping of people who share a vertical or horizontal role! Although this nightmare may not be on us next week, as more vendors support integration with electronic directories the role and importance of the directory in each organization will grow. Consider the complexity of Figure 11.6, which defines a consistent method to integrate the typical enterprise directory, directory-enabled ITSM work processes, an ITSM configuration management database (CMDB), and the RAPID enterprise architecture. (See Figure 11.6)

One of the most important capabilities of HP is our "Adaptive" brand of IT solutions, such as Adaptive Infrastructure and Adaptive Networking Architecture (ANA). When integrating the architecture and processes of our customers using technology, it is important to remember that electronic directories give us the capability to add both organization and flexibility in the form of an Adaptive Enterprise Architecture. A properly structured and integrated enterprise directory is the core component of business and IT agility—providing the spinal cord for our digital nervous system.

HP has been "doing directories forever." We reinforced this message before and during the HP/Compaq merger with HP's implementation of an enterprise directory based on Sun iPlanet. This directory runs on Linux servers, integrating feeds from other directories as diverse as Exchange 5.5 and the Active Directory and PeopleSoft, as well as critical cross-platform software facilities such as SiteMinder and PeopleFinder. Leveraging our external and internal experience, the HP consulting and integration department helps customers define sophisticated yet supportable directory strategies that enable agility in their information systems.

Directories are not a panacea; but they may be the only way to channel the turbid waters separating business and IT. Remember, start simple but consider all the possibilities. At HP, the directory is a core element of the glue

Figure 11.6
The need to integrate



that binds us. HP could not continue work without it; a single entry in HP's enterprise directory gives us the rights to use our vast network of business solutions, technology services, electronic processes, focused operational knowledge, and every other directory-enabled component.

11.2 Conclusion

Components are the "unit" of technology measurement in a service level agreement. They are the independently measurable parts of an IT organization, and they must be well defined, to reflect the strategy of the organization while delivering measurable impacts to the individual user. If the organization and user have agreed that customer satisfaction and system availability are of equal importance, for example, the model for each component must reflect this strategy.

When creating components for use in RAPID, remember five rules:

1. *Create only valuable components.* Everything in your environment can become a component, but only things that are congruent with your customer's business values and strategy can become *valuable* components. Preserve value by periodically reviewing each component to see if it still provides the expected levels of value, or if it needs changing, replacing, or retirement.
2. *Design components to fit.* Choose only certain items to be components in your specific architecture.
3. *Build components to last.* You will continue building, releasing, and retiring components indefinitely.
4. *Seek out new relationships.* Many components are related; make the important relationships components in their own right.
5. *Everyone makes decisions.* Components are the basis of understanding the effects of change and other big decisions.

This Page Intentionally Left Blank

12

Beyond Cost of Ownership

> *The most important scientific revolutions all include, as their only common feature, the dethronement of human arrogance from one pedestal after another of previous convictions about our centrality in the cosmos. (Stephen Jay Gould)*

In this chapter, I touch again on many of the issues discussed throughout this book. This chapter, however, is notably different from the others. It is not about anything technology organizations do well; rather, it is about the bad decisions we make most often. It does not describe best practices or repeatable processes, cohesive architecture, or coherent infrastructure. In this chapter, I take a much less formal approach to explaining why moving "beyond TCO" is so critically important to everyone.

> *Like most useful military concepts, [the phrase] "fog of war" normally is attributed to Clausewitz, who receives the credit for the alliterative "fog and friction"*—friction *referring to a physical impediment to military action,* fog *to the commander's lack of clear information. (Eugenia C. Kiesling, 2001)*

You must similarly consider that a "fog" exists in the IT department that flows through the maze of components you can build and instrument with RAPID. The fog of IT is a broad spectrum of intangibles that you can summarize as "ability management." Abilities, such as scalability and adaptability—and especially agility and other forms of security—at first seem of little consequence to the average user or business manager. Most abilities appear at first glance to be fuzzy or at least difficult to measure and prioritize. Some abilities seem more important, while others are industry sensitive. A single ability, however, has recently become much more important to most technology decision makers. If you cannot provide *security* (and for nearly half a century, many of us have taken it for granted) you might as well go back to handwritten information locked in vaults with armed guards.

This single ability is a formative issue of humanity itself: Long before the Internet, before video games and computers, telephones, electricity, the printing press, explosives, firearms, and the abacus—even before nations, kingdoms, states, cities, and communities—a cold and scared person sat starving in the dark of a cave thinking about his or her very personal security needs. And, at the last moment before finally falling asleep, a simple idea illuminates the mind's cave and warms the vision of tomorrow: how wonderful it would be to control fire! To bring the fire inside the cave, blossoming and gentled, warming, cooking, warning off the tiger outside—perhaps even a weapon in the fight for life.

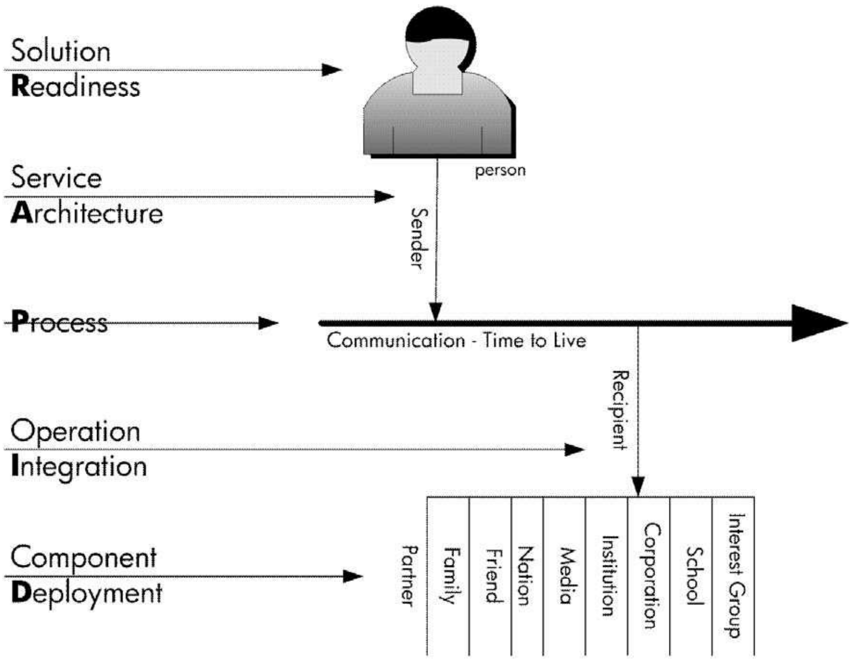
Controlling tomorrow's technology is just another form of our struggle to control fire. In the information age, we seek to control information and communications. Unlike traditional security, best practices for information security are rather elusive. Where do you place your guards? Who do you give the keys? The key to security in the information age is to distribute it. Too much of the information stored electronically today is organized in a way that makes the hacker's job quite easy. Gigantic databases, numerous backup copies, massive printed reports and the garbage they produce, and centralized disk storage make targets as attractive to the modern hacker as an unguarded stagecoach was to a villain of the Old West. Since the bubble burst, the hacker might not be your biggest concern; consider the impact of other forms of security requirements, such as the United States' Sarbanes-Oxley Act of 2002. How can you certify the accuracy of financial statements and disclosures, if you cannot audit the information systems, processes, and people that affect the majority of business information? A new Sheriff is most certainly in town, this one wielding the powers of email trails and old hard drives.

Standard business processes, it would seem, only compound this challenge; however, there are some modern opportunities to increase security without sacrificing manageability. Information is only valuable when you can understand its context; separate information from its grammar and you create a very effective method of security. For example, if you split your information into three parts before storing it (and remember where you stored it), then it is much harder for the hacker, competitor, or curious IT administrator to find value in the information. For example, hackers often find credit card numbers and billing addresses in the same database. Perhaps a better way would be to store the customer's credit card number with a customer ID number, and store the billing address in a completely different database? A better method might be to never store the credit card number in the first place, rather to use it only at the instant of transaction verification. But, decisions like this require the cooperation—and subsequent process change—of diverse groups like IT, accounting, finance, sales, and executives. These types of decisions are not easy to come by; as a result we often choose the 'politically correct' route before the right decision.

In fact, architected correctly you can even lose one part of the information and still be able to quickly recreate the whole, if you use a mathematical rule called *parity*. Very simply, parity means that you keep shooting for zero: $1 + 0 + -1 = 0$ (The mathematical explanation for this is somewhat more complicated, as it uses a binary logic operation where the XOR operation is true if and only if one of its operandas is true, unless both operandas are true then the output is false.) This rule is the key to a type of disk storage critical to modern computing, called "RAID" or *redundant array of independent disks*. It works by breaking information into pairs, storing each bit of information in one of two or more places and storing the parity of that information in a third place. The arrangement does not matter, and you can lose a copy of the information and still recover it from the two remaining copies. This is a dowside, however, as you must allocate additional space for the parity of information.

RAPID forces you to similarly break apart the structure and function of your IT decision making. To ensure parity in RAPID, learn to define the architecture of your processes in natural groups of communicating structures. The structures, represented by the RAPID solution, process, and component, are joined together by the functional roles of RAPID services and operations.

Put more simply, RAPID forces you to break apart structure and function into clearly defined layers. This exercise, while it may require more work for each individual component, is one of the key "missing elements" between technology and business value. There is no inherent value in a database (unless you are in the business of database software, of course); instead value is derived from the information contained in the database and how that information is used by the customer.



12.1 Security: Readiness, process, and components

A similar challenge faces everyone deciding on and preparing for the path to the "new IT." A single ability (securability) of agile business is, in this age, what Stephen Covey would call "A1, most urgent importance." Providing security is not a new ability; in fact, it is a concept as old as humankind; in its name countless blood has spilled on battlegrounds from which strong nations later sprang forth. Dictionaries call it freedom from risk, danger, injury, theft, failure, anxiety, fear—in general, freedom from bad. It is the guarantee you give the bank when you pledge your home for a mortgage; prisoners in every country throughout the world are held in the name of security. The guards, officers, and soldiers guarding our streets, borders, and heavens, do so in the name of this one all-important need. Privacy, scalability, adaptability, interoperability, accuracy, and capacity—each of these abilities and all others find their basis in the same fundamental desire of every human: security.

Why is security so difficult to provide for IT?

In short, security is difficult to provide for IT because we have come up with too many answers and too few questions. Security is about asking questions. Each time I ask a good question, I take a step forward in my own personal quest for security. A question allows us to interact with others for learning and other forms of enrichment. Many of my questions lately have started to produce another important asset: friends. Our connections to others—our ability to ask and answer questions—make us strong.

Consider the importance of a single question:

> *Since you made it clear just now that you don't think that Saddam has disarmed and [since] we have a quarter million troops in the Persian Gulf and now that you've called on the world to be ready to use force as a last resort, are we just days away from the point at which you decide whether or not we go to war? (Ron Fournier, in the Azores, March 16, 2003)*

If the best way to get a clear answer is to ask a clear question, then we need to learn to ask our customers some tough questions about where they feel technology and information should fit in their lives. We should not expect 100% agreement on any one answer. Some people choose to use new information and new technologies; some do not. Some people, in fact, never will. Both science and some of the strongest and oldest beliefs of the world agree that we need to balance our curiosity and quest for knowledge with a respect for the world. This need for balance affects us, because market strategies often do not consider if people really want what we are offering.

We often completely discount the fact that people could possibly choose not to want more information and more technology. Many people did not grow up with technology, and now they face an endless supply of unwanted information. The fight over the value of commerce and innovation must arise from the needs of some people to choose much simpler and (as some argue) more fulfilling lives. Do any of us really want our children to grow up in a world like that of our parents—with no time to enjoy our families and some of the conveniences we have created? How long is each "minor" improvement very valuable? Can we find ways to take apart the things we build and reuse them to spare our Earth? We cannot afford to leave any stone unturned, any question unexamined (especially a "tough" or "sticky" issue); we are still learning too much about the world around us. We know so much, yet we have so much to learn. *To answer the challenges of readiness and security alike, we need to learn how to ask better questions.*

What does this book have to do with business?

Security, in the logical and physical sense, is not an absolute requirement of every IT decision. Its greatest importance comes in the sense that we must be secure and committed to the decisions that we make; we must be able to follow-through. The most important point of this book is not about technology itself; rather, this book demands that the technology organizations find better ways to communicate with the business organizations. Too often, we get lost in the technology and its associated acronyms. There must be a better approach to communicating the long-term value of information, communications, and technology. Information technology needs to learn to ask structured business (formal) questions of our stakeholders, focused on managing the ever-growing complexity of the world we are creating:

1. *Readiness:* Do you really need what I am describing to you, or is it just a convenience—do we agree on price and delivery schedule? What will it take for each of us to get ready?
2. *Architecture:* Do we understand each other's needs, expectations of payment, and have we agreed to a long-term contract?
3. *Process:* Have we agreed on a plan to design and build each part of your solutions, understanding that your needs may change over time?
   a. The solution
   b. Each service I will bill you for
   c. The schedule
   d. Your support options
   e. The decision-making responsibilities we will each own or share and methods for improving our relationship
4. *Integration:* Have we found the seams? Do we understand where we will need to interact and where our teams will need to interact? Have we created an environment that lets us exchange information to ensure consistent support?
5. *Deployment:* Do we have a clear decision-making process? Have we established common priorities and made the right preparations for well-executed decisions? Do we have the right methods to audit and study global and individual decisions?

If something looks too good to be true, it probably is—but how can you be so sure of your point of view that you do not consider both possibilities?

This rule—consider both possibilities—is especially important to those concerned with providing security to information and technology. We built the majority of current systems in IT departments around the world without focused concern for security; the effect of this oversight is a creeping global insecurity. There are more concerns than answers, and the landscape of security changes with each new hacker exploit or monster virus. Do not believe that you are immune to the demands of information security. Each organization must build repeatable processes for enabling and managing security abilities; often the focus of these tasks must be on each individual component of information. You must understand not only the organizational challenges of security but also the effects on individual data. To address the breadth and depth of this challenge, classification systems (such as RAPID, networks, and directories) allow us to see our decisions as both the forest *and* the trees.

12.2 To what end? Using technology to search for answers in information

I cannot even pretend to have the answers to the question "To what end do we have technology?"; to do so would have me in the stockades, burning at the stake, or at the business end of a Unabomb in a second. I do have some of my own answers and quite a few more questions. So do you. So does everyone. I base my questions on my childhood fear of commitment, my adult understanding of the significance of my decisions, and my professional knowledge of the balance inherent to integrating the organization using the natural hierarchy of an electronic directory and a network.

For the first time in the history of humankind, we have the ability to share in exploring the eternal question "Why am I here?" by recording all our best thoughts for our children—and making sure those thoughts stick around as long as they are needed. One of the most pointed explanations I have read to date on the importance of making proper decisions in the struggle for life comes from the biologist Edward O. Wilson in *Consilience* (1998). The idea of "consilient" thinking, like RAPID, points out that we must balance our decisions. Wilson asks some very difficult questions about humanity by looking at similarities in a very diverse set of subjects:

biology, history, sociology, culture, and even religion. His question: Can the tree of knowledge connect everything to one common point in nature? Consider two possible responses. Einstein would ask, What if the answer is relative to your point of view? My question: What if *both* viewpoints were somehow correct?

There is no deep meaning to IT; it is just a container for today's thoughts, a language and a timekeeper that instantly connects all people. The important part is only that the technology connects people and their information. Information and technology help us express the views, opinions, and content created by a global picture of the world. Connected to the Internet

Figure 12.1
A knowledge tree

(at home, in the community, at work, and at school) are the thoughts of billions of people. These precepts, thoughts, opinions, buying habits, building practices, laws, and decisions are happening in real time. More information is generated today than any individual could study in a lifetime. Every day, as a larger percentage of the population connects, the size of this "tree of knowledge" grows.

This is not true, however, of the "truth" behind the information we have stored. People have recorded this "truth" throughout history: in personal and group impressions of the hunt, wars, lawsuits, inaccurate bills, tax collectors, misunderstandings, and other "best" practices. We all have better ways to work out our differences. We all have better things to do (as Tony points out in the introduction, Bermuda would be a great start!). Maybe the answer is much simpler than we have considered, and our only job is to prune our trees of change: in our constraints and expectations, standards and rules, architecture and processes, resources and activities, solutions and services, and questions and answers. Keep it simple; when you do not know where to go, ask questions while looking for your own roots. (See Figure 12.1)

The long life of a tree (as well as in every windstorm) demands flexibility; there are three basic options for the growth of every branch of a tree: (1) toward the radial center, (2) away from center, or (3) straight up. It has been argued that there are also three options in the human condition: (1) fight, (2) flight, or (3) freeze. Perhaps there is more between heaven and earth than was dreamt of by Hans Selye—more flexibility and potential to human beings than accounted for by fear. People arguably want security—freedom from fear—but many of us also desire a spoonful of creativity, a dash of innovation and invention, and the promise of new challenges and opportunities.

This Page Intentionally Left Blank

Appendix A: Hewlett-Packard ITSM Activities

1.1. Business Assessment

1.1.1. Process Delivery Activities
- 1.1.1.1. Define market segments
- 1.1.1.2. Characterize service opportunities
- 1.1.1.3. Review segment size and growth potential
- 1.1.1.4. Conduct segment value chain analysis
- 1.1.1.5. Conduct competitive analysis

1.1.2. Quality Control Activities
- 1.1.2.1. Establish research methodology
- 1.1.2.2. Develop analysis and recommendations formats
- 1.1.2.3. Develop management reports
- 1.1.2.4. Perform continuous process improvement

1.2. Customer Management

1.2.1. Process Delivery Activities
- 1.2.1.1. Develop marketing communications
- 1.2.1.2. Maintain proposal boilerplates
- 1.2.1.3. Sell IT services
- 1.2.1.4. Manage customer relationship
- 1.2.1.5. Survey customers
- 1.2.1.6. Identify service opportunities
- 1.2.1.7. Conduct executive briefings
- 1.2.1.8. Perform new customer introduction

1.2.2. Quality Control Activities
- 1.2.2.1. Establish liaison procedures
- 1.2.2.2. Establish customer introduction procedures
- 1.2.2.3. Develop management reports
- 1.2.2.4. Perform continuous process improvement

1.3. IT Strategy Development

- 1.3.1. Process Delivery Activities
  - 1.3.1.1. Determine IT budget
  - 1.3.1.2. Perform strategic analysis
  - 1.3.1.3. Define and document the IT vision
  - 1.3.1.4. Develop the IT mission statement
  - 1.3.1.5. Identify breakthrough objectives
  - 1.3.1.6. Identify key critical success factors, obstacles, and constraints
  - 1.3.1.7. Select service solutions for IT provisioning
  - 1.3.1.8. Identify enabling technologies
  - 1.3.1.9. Define the IT architecture
- 1.3.2. Quality Control Activities
  - 1.3.2.1. Establish IT business planning and control procedures
  - 1.3.2.2. Define IT business rules
  - 1.3.2.3. Determine the IT organizational structure
  - 1.3.2.4. Establish IT policies, standards, guidelines and procedures
  - 1.3.2.5. Determine service selection prioritization procedures and criteria
  - 1.3.2.6. Develop implementation control procedures
  - 1.3.2.7. Develop management reports
  - 1.3.2.8. Perform continuous process improvement

2.1. Service Planning

- 2.1.1. Process Delivery Activities
  - 2.1.1.1. Plan for new standard service
  - 2.1.1.2. Design custom service
  - 2.1.1.3. Conduct service risk analysis
  - 2.1.1.4. Define functional requirements
  - 2.1.1.5. Analyze capability gaps
  - 2.1.1.6. Make service "buy vs. build decision
  - 2.1.1.7. Determine ROI on service development
  - 2.1.1.8. Create internal design specification
  - 2.1.1.9. Develop strategic alliances
  - 2.1.1.10. Evaluate portfolio impacts
  - 2.1.1.11. Keep service current

2.1.1.12. Manage service value
2.1.1.13. Obsolete service
2.1.2. Quality Control Activities
2.1.2.1. Develop service specification standards
2.1.2.2. Develop management reports
2.1.2.3. Perform continuous process improvement

2.2. Service Level Management

2.2.1. Process Delivery Activities
2.2.1.1. Assess customer-specific service requirements
2.2.1.2. Map requirements to standard services
2.2.1.3. Identify need for custom services
2.2.1.4. Negotiate and document SLA
2.2.1.5. Establish service performance review cycle
2.2.1.6. Analyze customer-specific service level performance
2.2.1.7. Create customer reports
2.2.1.8. Conduct service performance review
2.2.1.9. Propose service improvements (customer-specific)
2.2.2. Quality Control Activities
2.2.2.1. Establish service level priorities
2.2.2.2. Maintain SLA version control
2.2.2.3. Develop management reports
2.2.2.4. Perform continuous process improvement

2.3. Security Management

2.3.1. Process Delivery Activities
2.3.1.1. Enforce corporate security policy (as it pertains to IT)
2.3.1.2. Promote security awareness within IT
2.3.1.3. Conduct security gap analyses
2.3.1.4. Conduct security risk assessments
2.3.1.5. Perform security audits
2.3.1.6. Evaluate security incidents
2.3.1.7. Assist with the resolving security issues found within other IT processes
2.3.1.8. Establish supplier relationships (related to security)

2.3.2. Quality Control Activities
- 2.3.2.1. Establish security procedures (includes virus control)
- 2.3.2.2. Select security systems and/or tools
- 2.3.2.3. Develop management reports
- 2.3.2.4. Perform continuous process improvement

2.4. Availability Management

2.4.1. Process Delivery Activities
- 2.4.1.1. Determine reliability and serviceability requirements
- 2.4.1.2. Determine contingency requirements
- 2.4.1.3. Analyze service availability risks
- 2.4.1.4. Conduct gap analysis (availability)
- 2.4.1.5. Develop buy vs. build recommendations (availability)
- 2.4.1.6. Develop buy and build specifications (availability)
- 2.4.1.7. Establish supplier relationships
- 2.4.1.8. Analyze service availability performance
- 2.4.1.9. Propose service improvements (availability)
- 2.4.1.10. Conduct supplier review
- 2.4.1.11. Rehearse and review contingency plan

2.4.2. Quality Control Activities
- 2.4.2.1. Establish supplier procedures
- 2.4.2.2. Establish contingency plan standards
- 2.4.2.3. Develop management reports
- 2.4.2.4. Perform continuous process improvements

2.5. Capacity Management

2.5.1. Process Delivery Activities
- 2.5.1.1. Inventory service resources
- 2.5.1.2. Characterize service workloads and demands
- 2.5.1.3. Configure service capacity profile
- 2.5.1.4. Determine service capacity requirements
- 2.5.1.5. Conduct gap analysis (service capacity)
- 2.5.1.6. Develop buy vs. build recommendations (service capacity)

- 2.5.1.7. Develop buy and build specifications (service capacity)
- 2.5.1.8. Analyze workload performance
- 2.5.1.9. Propose service improvements (capacity)
- 2.5.1.10. Manage service demand

2.5.2. Quality Control Activities

- 2.5.2.1. Establish a service capacity planning system
- 2.5.2.2. Establish service capacity planning benchmarks
- 2.5.2.3. Develop management reports
- 2.5.2.4. Perform continuous process improvement

2.6. IT Financial Management

2.6.1. Process Delivery Activities

- 2.6.1.1. Calculate expected service cost
- 2.6.1.2. Analyze projected revenues
- 2.6.1.3. Develop service budget
- 2.6.1.4. Analyze service usage and cost
- 2.6.1.5. Propose service improvements (cost)
- 2.6.1.6. Calculate invoice and bill customer
- 2.6.1.7. Receive payment
- 2.6.1.8. Track financial assets
- 2.6.1.9. Calculate total cost of ownership

2.6.2. Quality Control Activities

- 2.6.2.1. Establish cost and charging allocation structures
- 2.6.2.2. Promote cost-effective service usage
- 2.6.2.3. Establish cost management system
- 2.6.2.4. Establish investment appraisal guidelines
- 2.6.2.5. Develop management reports
- 2.6.2.6. Perform continuous process improvements

3.1. Build and Test

3.1.1. Process Delivery Activities

- 3.1.1.1. Acquire service components
- 3.1.1.2. Develop application provision guidelines
- 3.1.1.3. Develop applications
- 3.1.1.4. Certify hardware/software
- 3.1.1.5. Construct service support and control mechanisms
- 3.1.1.6. Develop test plan and procedures
- 3.1.1.7. Perform prototype test setup

3.1.1.8. Perform prototype test
3.1.1.9. Perform unit test setup
3.1.1.10. Perform unit test
3.1.1.11. Perform pilot test setup
3.1.1.12. Perform pilot test
3.1.1.13. Document recovery procedures
3.1.1.14. Develop support procedures
3.1.1.15. Develop training design and plan
3.1.1.16. Develop training materials
3.1.1.17. Develop "master blueprint" (i.e., “production plan”)

3.1.2. Quality Control Activities
3.1.2.1. Develop management reports
3.1.2.2. Perform continuous process improvement

3.2. Release Management

3.2.1. Process Delivery Activities
3.2.1.1. Procure resources
3.2.1.2. Conduct IT staff and supplier training
3.2.1.3. Assemble components
3.2.1.4. Distribute components
3.2.1.5. Implement service support and control mechanisms
3.2.1.6. Implement component, service function, or end-to-end service
3.2.1.7. Perform software administration
3.2.1.8. Conduct customer training
3.2.1.9. Establish production test scenarios
3.2.1.10. Perform production test
3.2.1.11. Perform customer acceptance test setup
3.2.1.12. Perform customer acceptance test
3.2.1.13. Activate service

3.2.2. Quality Control Activities
3.2.2.1. Develop management reports
3.2.2.2. Perform continuous process improvement

4.1. Operations Management

4.1.1. Process Delivery Activities
4.1.1.1. Schedule production processing
4.1.1.2. Monitor resource status and raise alerts

4.1.1.3. Manage output and print queues
4.1.1.4. Manage backups
4.1.1.5. Administer clients, servers, networks
4.1.1.6. Administer users
4.1.1.7. Administer internet protocol (IP) addresses
4.1.1.8. Administer databases
4.1.1.9. Manage voice infrastructure
4.1.1.10. Maintain a secure IT infrastructure environment
4.1.1.11. Coordinate preventive maintenance
4.1.1.12. Track service delivery cost data
4.1.1.13. Track service metric data

4.1.2. Quality Control Activities
4.1.2.1. Establish and maintain operability standards
4.1.2.2. Promote operational effectiveness
4.1.2.3. Promote operational efficiency
4.1.2.4. Manage the computing facility
4.1.2.5. Develop management reports
4.1.2.6. Perform continuous process improvement

4.2. Incident Management

4.2.1. Process Delivery Activities
4.2.1.1. Accept calls
4.2.1.2. Log incidents
4.2.1.3. Categorize incidents
4.2.1.4. Prioritize incidents
4.2.1.5. Isolate incidents
4.2.1.6. Escalate incidents (within the process and/or to management)
4.2.1.7. Track incident progress
4.2.1.8. Resolve incidents
4.2.1.9. Notify customers
4.2.1.10. Close incidents

4.2.2. Quality Control Activities
4.2.2.1. Establish Help Desk structure
4.2.2.2. Establish incident control system
4.2.2.3. Develop management reports
4.2.2.4. Perform continuous process improvement

4.3. Problem Management

4.3.1. Process Delivery Activities
4.3.1.1. Analyze incident trends
4.3.1.2. Log problem
4.3.1.3. Identify root cause
4.3.1.4. Track problem progress
4.3.1.5. Verify known errors
4.3.1.6. Control known errors
4.3.1.7. Resolve problems
4.3.1.8. Close problems/known errors
4.3.2. Quality Control Activities
4.3.2.1. Establish problem/known error control system
4.3.2.2. Setup and maintain support contacts
4.3.2.3. Establish preventive maintenance procedures
4.3.2.4. Establish known error verification facilities
4.3.2.5. Establish supplier support interfaces
4.3.2.6. Develop management reports
4.3.2.7. Perform continuous process improvement

5.1. Change Management

5.1.1. Process Delivery Activities
5.1.1.1. Request for Change (RFC) processing
5.1.1.2. Impact assessment
5.1.1.3. Change approval
5.1.1.4. Schedule and coordinate changes
5.1.1.5. Coordinate recovery from change failures
5.1.2. Quality Control Activities
5.1.2.1. Establish RFC submittal process
5.1.2.2. Define change category and priority schemes
5.1.2.3. Establish change "project" management process
5.1.2.4. Establish change advisory board
5.1.2.5. Conduct post-change review (retrospective)
5.1.2.6. Develop management reports
5.1.2.7. Perform continuous process improvement

5.2. Configuration Management

5.2.1. Process Delivery Activities
5.2.1.1. Maintain CIs
5.2.1.2. Conduct control and status accounting

- 5.2.1.3. Report CMDB data
- 5.2.1.4. Verify integrity of CMDB data
- 5.2.2. Quality Control Activities
  - 5.2.2.1. Load initial CMDB data
  - 5.2.2.2. Establish configuration management system
  - 5.2.2.3. Develop CI control policies
  - 5.2.2.4. Develop management reports
  - 5.2.2.5. Perform continuous process improvement

This Page Intentionally Left Blank

Appendix B: HP Infrastructure Operations

1. Core Infrastructure

- 1.1. Disaster Recovery
- 1.2. Distributed Infrastructure
  - 1.2.1. Directory
  - 1.2.2. Naming
- 1.3. Enterprise Management
  - 1.3.1. Asset & Configuration Management
  - 1.3.2. Consumption/Billing Management
  - 1.3.3. Enterprise Management
  - 1.3.4. Internet Specific Management
  - 1.3.5. Network Management
  - 1.3.6. Problem, Service & Change Management
  - 1.3.7. Security Management
  - 1.3.8. Service Catalog
  - 1.3.9. Systems Management
- 1.4. Enterprise Operations
  - 1.4.1. Change Management
  - 1.4.2. Enterprise Scheduling and Production Support
  - 1.4.3. Facility and Site Management
- 1.5. High Availability
- 1.6. Information Security
  - 1.6.1. Identity Management
- 1.7. Internet Infrastructure
  - 1.7.1. Internet Infrastructure (FTP, Content Mgmt, Proxy)
  - 1.7.2. Neutral Zone Architecture
  - 1.7.3. Remote Employee Access
  - 1.7.4. Remote Partner Access
- 1.8. Mobility
  - 1.8.1. Platforms & Infrastructure
  - 1.8.2. Platform Management & Personal Productivity Applications

- 1.9. Networks
  - 1.9.1. Call Center Telephony
  - 1.9.2. ISP Connectivity
  - 1.9.3. Overall Data Network Architecture
  - 1.9.4. Personal Connectivity (Cellular and 3G)
  - 1.9.5. Video and Multimedia (webcasting, streaming media)
  - 1.9.6. Voice Systems
  - 1.9.7. WAN/LAN/MAN
- 1.10. Personal Computing Environment
  - 1.10.1. Personal Productivity Applications
  - 1.10.2. Platform
- 1.11. Platform Infrastructure
  - 1.11.1. Grid & Planetary Computing
  - 1.11.2. Server Platforms
  - 1.11.3. Storage Platforms
  - 1.11.4. Utility Data Center
- 1.12. Printing Infrastructure

2. Application Infrastructure

- 2.1. Application Runtime Environments
- 2.2. Business Intelligence Platform and Tools
  - 2.2.1. Business Intelligence Tools
  - 2.2.2. Data Warehousing
  - 2.2.3. Extract, Transform, Load (ETL)
- 2.3. Business Process Management
- 2.4. Collaboration
  - 2.4.1. Mail and Messaging
  - 2.4.2. Team Collaboration
- 2.5. Content Management
  - 2.5.1. Document Management
- 2.6. Databases
- 2.7. Development Framework and Tools
  - 2.7.1. ERP Configuration and Development
  - 2.7.2. General Development
- 2.8. External Application Integration
- 2.9. Internal Application Integration
- 2.10. Portal Integration Framework and Technologies

Frequently Asked Questions

Best Process Questions

Figures and Tables

This Page Intentionally Left Blank

Index

T

About the Author

Drew Wigodsky is a technical consultant in Hewlett-Packard Services Consulting and Integration. His technology experience includes a myriad of roles as developer, designer, architect, project manager, executive, and owner. Before joining Rainier/Compaq/HP, Drew was a consultant with Microsoft Consulting Services, an executive in a small-business technology consultancy, a consulting practice director, and developed software for high-speed banknote image processing and telephone sales management. His customers include large and small multinational corporations, colleges, state governments, and non-profit organizations. Currently, Drew serves as an architect, planning IT investment strategy for a $3 billion HP Services outsourcing contract.


ASR(not)proudly PRESENTS
a re-release of this beautiful book
We are Anti-Scene releases group, we are free, independent, and not caring MAD people.