The FAST PATH to CORPORATE GROWTH

Leveraging Knowledge and Technologies to New Market Applications

MARC H. MEYER

The Fast Path to Corporate Growth

The Fast Path to Corporate Growth

Leveraging Knowledge and Technologies to New Market Applications

MARC H. MEYER

OXFORD
UNIVERSITY PRESS
2007

OXFORD
UNIVERSITY PRESS

Oxford University Press, Inc., publishes works that further
Oxford University's objective of excellence
in research, scholarship, and education.

Oxford New York
Auckland Cape Town Dar es Salaam Hong Kong Karachi
Kuala Lumpur Madrid Melbourne Mexico City Nairobi
New Delhi Shanghai Taipei Toronto

With offices in
Argentina Austria Brazil Chile Czech Republic France Greece
Guatemala Hungary Italy Japan Poland Portugal Singapore
South Korea Switzerland Thailand Turkey Ukraine Vietnam

Published by Oxford University Press, Inc.
198 Madison Avenue, New York, New York 10016

www.oup.com

Library of Congress Cataloging-in-Publication Data
Meyer, Marc H.
The fast path to corporate growth : leveraging knowledge and technologies to
new
market applications/
Marc H. Meyer
p. cm.
Includes bibliographical references and index.

ISBN 978-0-19-518086-2
1. Success in business. 2. Creative ability in business. 3. New products—Management. 4. Business planning. 5. Marketing research. 6. Market segmentation. 7. Corporations—Growth. I. Title.
HF5386.M564 2007
658.4—dc22 2006022341

1 2 3 4 5 6 7 8 9

Printed in the United States of America
on acid-free paper

To Olga

Acknowledgments

This book is the product of many helping hands. Three in particular deserve special thanks: Mark Anzani, whose steadfast friendship and knowledge of the computer industry lent early momentum to my efforts; Neil Willcocks, who has synthesized user-centered design with technology platforms in ways that few can match; and John Helferich, who has helped me think through the business processes needed for enterprise growth.

Others in industry and academia made important contributions to various chapters. These include Northeastern University colleagues John Friar, Fred Crane, Daniel McCarthy, Mario Maletta, and Dennis Shaughnessy; Al Lehnerd, for many years my mentor and dearest friend; and James Utterback of MIT.

Within industry, my deepest thanks go to George Walsh and Peter Tarrant of IBM; Art St. Cyr, Bill Easdale, and Sage Marie of Honda; The Math Works' Loren Dean, Peter Webb, and Eugene McGoldrick; Joseph Sawicki of Mentor Graphics; Harry Keegan of Braintree Laboratories. There are many associates at Mars, Incorporated that contributed time and thought to this project. I extend particular thanks to Ralph Jerome, Bob Boushell, Tom Collins, Michael Wilson, and Stewart Townsend. I also wish to thank Harry West of Design Continuum for teaching me much about user-centered design; Kristin Boyle for helping me develop methods for validating prototypes; and James Walter, who showed me how to view new services as modular platforms. Equally important to this effort were Terry Leahy, Martin McDonough, and Jeff Curran who all contributed to the chapters on business model innovation and development.

Special thanks go to my students in Northeastern University's High Tech MBA program who, throughout the years, have helped me refine my methods. For specific examples employed at various parts of this book, I owe special thanks to Chad Haering, Subhabrata Biswas, Nathan Butts, Ofer Michael, and Kevin Clarke. Teaching this cohort of student-managers has been one of the great honors of my life.

This book is also far better for the writing counsel of Richard Luecke. Appealing to my truest passion, Dick would often say, "Marc, if you can't wrap up that chapter in forty pages, just stop and go fishing." Since parts of

this book were written overlooking many fine waters here and abroad, I thank Dick for his sage advice.

I had the further good fortune to work with Oxford University Press editor John Rauschenberg. Truly interested in ideas and their application, John was a true partner—and OUP is an outstanding organization.

Perhaps most important have been the support and “encouragement” I received on the home front. I am sure that many authors have heard words akin to, “Husband, get back up there and finish that book. It’s going to be great!” Or “Pops, c’mon, let’s take a break and shoot some hoops.” Or, “Daddy, you work on the neatest stuff. I love those new M&M’s!” The bottom line is this: Olga, Max, and Rosa, your words are the gold of this venture I will remember most and forever cherish.

Contents

Introduction: The Fast Path to Corporate Growth

1 IBM Rises from the Ashes

2 A Framework for Action

3 Segmenting Markets for Growth

4 Understanding User Needs

5 Creating Design Concepts, Prototyping, and Validating Design Choices

6 How Honda Innovates

7 Product Line and Platform Development

8 Honda's Element Comes to Life

9 Business Model Innovation

10 New Brand and Product Line Development at Mars

11 Making the Business Case

12 Executive Decision Making

13 Leading Teams to Growth

Appendix: Underlying Technology Principles in IBM's Renewal

Index

The Fast Path to Corporate Growth

Introduction

The Fast Path to Corporate Growth

Growth is the goal of every ambitious business. Growth serves the well-being of the corporation and its shareholders. It provides employees with a dynamic work environment and opportunities for advancement. It provides customers with a stream of new products and services. Growth is what every CEO aims for but usually finds illusive. Many companies that do produce growth eventually find it slipping through their fingers. Indeed, we all know of enterprises that grew on the strength of innovative products or services, gained market leadership, but, for one reason or another, lost their lead and began a steady descent—some into the dustbin of business history. Yesterday's leaders in automaking, minicomputing, and photography come to mind. A comparison of the Fortune 100 of even 25 years ago with the current list will confirm how today's leaders often become tomorrow's laggards—if not losers.

Is stagnation and decline inevitable for companies in mature industries and markets? Must all great companies be bumped aside? Decline happens, but that does not mean that decline is preordained. If a company can create and apply technology to solve real problems at one point in its history, no law of nature dictates that it cannot do so again (and again) in the future. Pessimists should be heartened by the example of Apple, which, thanks to the now ubiquitous iPod, is enjoying a renaissance of growth and shareholder prosperity—and with the same driver at the wheel. The company seemed condemned to a niche market. But by applying its technical capabilities and customer knowledge to other needs—by thinking outside the confines of its existing business and toward the convergence of computing and music—Apple escaped its small niche and created a new and fast-growing business. Rivals in the consumer electronics industry were caught flat-footed.

Business renewal does not happen easily. It requires courage and leadership on the part of executives, focused and creative work by teams of dedicated individuals, and a target market waiting to be served. This book presents a framework and methodology, as well as encouragement, for that important work.

The Perilous Third Phase

FIGURE I.1 summarizes the challenge faced by many companies: the three-phase market life cycle. In the first phase, an innovative product (or service) is introduced, usually based on breakthrough technology. Initially, sales are low because the performance of the new technology is not perfected.

In the second phase, sales increase dramatically as price and performance improve, and as customers recognize the utility of the new product. Observing this expanding market, fast-following rivals rush in with products of their own. Some are me-too copies; others represent genuine value improvements. Eventually, however, the market is saturated or better substitutes appear, and the product category enters a phase of stagnation and decline. Profits in this third phase almost always fall faster than unit sales.

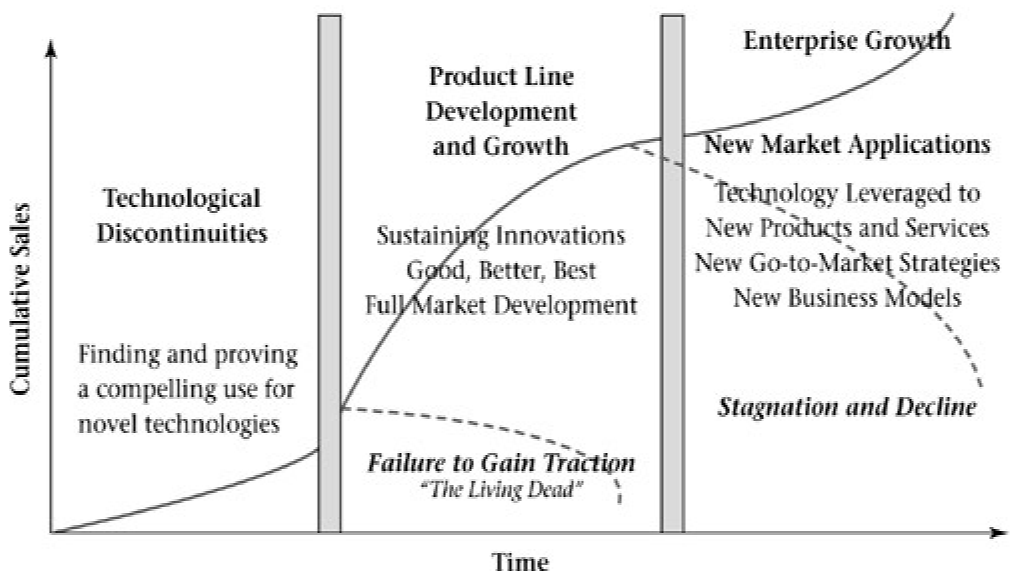

FIGURE I.1 The Challenges of Growth over the Market Life Cycle

This final phase is pivotal in the life of product categories and the companies that provide them. The question is, how can a company get off the downward slope of stagnation and onto the ascending slope of enterprise growth? One way is through acquisition: simply purchase another company that's in either phase one or phase two—perhaps a rival—and ride its coattails through market expansion. Appealing as this solution may be, the track record of acquisitions is discouraging; disappointments outnumber successes. The only sure winners here are the investment bankers who put these deals together, and who reap still more fees when the deals fall apart and are "divested."

Getting Back to Growth

The innovation literature and industrial practice point to two more promising paths to growth. The first path is the creation of a truly breakthrough innovation—a technological discontinuity. As described so well by James Utterback, breakthrough innovation brings something truly new to the world and represents a distinct departure from existing technologies or forms.[1] The first semiconductor chip fits this description, as do the first inkjet printer, wireless phone technology, the angioplasty heart procedure, and other things we now take for granted. Many of the products that fill our homes and workplaces today—GPS navigation devices, fiber-optic cables, hybrid-powered vehicles, digital imaging, and email—are based on technical breakthroughs. But as every CEO and R&D person knows, projects aimed at developing these breakthroughs are risky and expensive and may not produce revenues for 10 or 12 years, if at all. Those projects that succeed put their companies onto a rapidly rising growth path. But many fall short of technical goals or miss the mark with customers. This is not to say that every technology-based company shouldn't have breakthrough projects in its portfolio; it simply means that they will not stop the downward slope of sales and profits in a timely way. Companies caught in the declining phase of the market life cycle need relief in *two to three years*.

The second path to enterprise growth—and a well-worn path at that—is incremental innovation. This form of innovation improves on or branches from an existing technology or product design. It is close to a company's core business and serves markets and customers the company already understands. The products and services that result from incremental innovation generally require less time and money to develop than do their more radical kin. They are less risky because they are based on proven technologies and are directed to customers already familiar to the company. In many instances, they can be manufactured on a company's existing production line and distributed by the sales force through the distribution channels the company has already developed and refined.

This approach is innovation at the margin, improving something that already exists and enjoys a ready market: a cell phone with a digital imaging feature, a PC with a faster processor, a new version of office application software with more bells and whistles. All are incremental innovations, and to the extent that they address real user needs, they can stem or reverse the downward tide of sales. Brand extensions and product derivatives aim to do the same. As I and Alvin Lehnerd describe in *The Power of Product Platforms*,[2] the challenge here is to create next-generation architectures for these established product lines, architectures that share product and process platforms to reduce cost of goods and time to market. Although this work is less incremental in nature—the development of new architecture, subsystem technologies, and manufacturing processes—it is still aimed at better serving *current users and uses*.

So which path to growth should a company take? The quick answer would be "both." Every well-balanced R&D portfolio contains a mix of incremental and discontinuous projects. If well done, incremental innovation can give a boost to revenues and profits in the short run. The breakthrough innovation route may hold great promise for the future, but many companies cannot wait a decade or more to reap the benefits. They need new revenues right away.

Fortunately, there is a third, less understood path to growth. I call it *new market applications*. That path is the subject of the chapters that follow. New market applications are new product lines and services that leverage a company's technical and, in some cases, production capabilities to serve *new users* and *new uses*. Instead of trying to squeeze new sales out of existing customers with a "new and improved" version of an old product, a new market application focuses a company's talents on an unexploited market segment.

Successful new market applications have the virtue of getting a company onto the growth curve without the high risk, costs, and long development cycles associated with breakthrough innovations. Although the improvement and application of company technology is an essential part of this strategy, a new market application does *not* rely on a technological breakthrough. Projects that do are best left in R&D.

Success here requires management to consider a range of potential market applications—new products or services for new users and new uses. Management can then look at these potential applications through the lenses of *adjacency* and profit potential, as articulated by Chris Zook.[3] Some targets may be clearly adjacent to the current business. For example, a telecommunications company might create a stream of new value added services that leverage its fiber and wireless infrastructures. Other new target applications might be ventures to define and capture a new, emerging market space.[4] For example, a traditional confectionary manufacturer might venture into the bold new world of healthful, nutritious—and higher priced—snacks. Either approach can work, depending on the situation. Some of the firms I studied broke out of their doldrums with bold moves into new market applications, and then exploited their advantage with adjacent developments; others companies prospered my moving aggressively forward markets rich with new, closely related opportunities. Either way, successful companies were all able to leverage their core skills and technology into new product lines and services, making the quest for growth not only exciting but *achievable*.

The best feature of this strategy is that it *gets results quickly* within the context of overall corporate growth. That is why the title of this book is *The Fast Path to Corporate Growth*. In most of the cases described in these chapters, new market applications were launched roughly two to three years after conception; many were profitable within one year of launch. Better still, many new market applications are capable of constituting a *new business*—not just a product line extension.

What It Takes to Succeed

What does it take to successfully develop and launch new market applications? My studies indicate that innovation is required in three dimensions: marketing, technology, and the business model (FIGURE I.2).

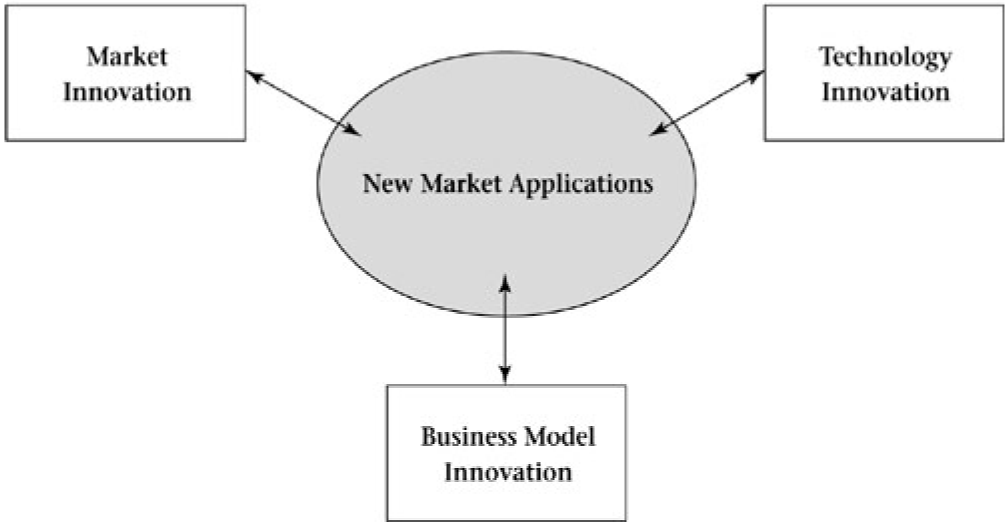

FIGURE I.2 Three Dimensions of Innovation in New Market Applications Development

— *Marketing innovation*. Innovation begins here with market segmentation to identify new target users and users to which a company can leverage its technology. The innovation then extends to user research that embeds the development team in the world of potential new customers. Traditional quantitative methods of market research are secondary; they are no substitute for walking in the shoes of target customers.

— *Technology innovation*. Innovation in this dimension concentrates on two levels. The first is transforming knowledge of users' needs and frustrations into new concepts and then turning these concepts into specific product and service designs. The second level is achieving implementation of those designs wisely. This means adapting working technologies to new applications through a strategy of modularity in architecture and the development of shared subsystems that can be leveraged across those applications. In this way, the shared technologies become enabling platforms for enterprise growth.

— *Business model innovation*. A business model describes how a company makes money. New product lines and services targeting new users or uses offer the opportunity to replace old margin-poor business models with better, margin-rich approaches. Shackling a new market application to an old business model may limit if not destroy its transformational potential.

There is, of course, an important human element to new market applications. These initiatives are driven by teams of employees from R&D, marketing, and other functions who aren't afraid to reach out to customers, observe them in their natural environments, and rapidly develop and test new solutions that meet their needs and remedy their frustrations. Successful new market applications also need executive sponsors who won't try to force square pegs into round holes. If a promising new product line needs a different approach to distribution or pricing or is better suited to outsourced manufacturing than to the company's own production lines, these executives must support what's best for the new product line. More about this later.

What's Ahead?

This book describes a framework for growth through new market applications. That framework—a series of activities—is explained in chapter 2. Chapters 3 through 11 explain those activities and provide extended examples from exceptional companies. Principal among these are Honda, with its development of the now-popular Element SUV, and Mars, which has launched new candy and pet food products. Many chapters contain templates readers can use in their own projects. Chapters 12 and 13 address the people side of new market applications. These provide plainspoken advice on what executives and innovation team leaders must do to execute the steps of new market applications development.

The Research Sample

The framework described in this book was synthesized from detailed research with more than two dozen companies over the past decade. These companies represent a broad range of industries, and each has created new product and service lines that have sold successfully to new users and for new uses relative to their traditional offerings. The experiences of these companies are stunningly similar in their planning and execution, organization, and business processes. They include IBM, the world's largest computer company; Honda, one of the most successful and fastest growing automobile manufacturers; Mars, the world's largest confectionery and pet food manufacturer; The

MathWorks, a market leader in mathematical computing; EMC, the leader in storage systems; Charles River Laboratories, the world's largest pharmaceutical services company; Mentor Graphics, a respected software company in semiconductor design and testing; and one of the country's largest suppliers of agriculture products. Service companies were well represented in the study group. They included the largest U.S. life reinsurer, the leader in custodial services for the mutual fund industry, and a leading U.S. provider of health care products, systems, and services. One could not have a better group of companies from which to learn methods for achieving internal, organic enterprise growth. In a number of instances, my work with these companies had less to do with survey research and case writing than with working directly with teams to develop new and exciting product and service concepts. These companies provide plenty of examples of how other enterprises can get on a more rapid path to growth—as either an alternative or complement to acquisitions.

Now, you are about to share this journey. Before we launch into the details of our management framework for corporate growth, however, we have a story —a true story of an industry leader that found itself in decline. In the early 1990s, this company's core business was imploding. With sales dropping and losses mounting, something had to be done—and quickly. In this company's reemergence to greatness lie the foundations of the concepts and methods for the rest of the book.

Read on.

Notes

1. James Utterback, *Mastering the Dynamics of Innovation* (Boston: Harvard Business School Press, 1994).

2. Marc H. Meyer and Alvin Lehnerd, *The Power of Product Platforms* (New York: Free Press, 1997).

3. Chris Zook, *Beyond the Core: Expand Your Market without Abandoning Your Roots* (Boston: Harvard Business School Press, 2004).

4. W. Chan Kim and Renee Mauborgne, *Blue Ocean Strategy* (Boston: Harvard Business School Press, 2005).

CHAPTER ONE

IBM Rises from the Ashes

There are few examples of reshaping a withering traditional enterprise as dramatic as IBM's renewal of its mainframe computer business. It is perhaps the greatest industrial turnaround story in American business during the 1990s. Also, it clearly shows the differences between reinvigorating an existing product to better serve *current users and uses* and extending that product line to capture *new users and new uses*.

To make this happen, IBM had to:

1. better segment its markets and research the needs of users in those markets;
2. more effectively translate that understanding into compelling new product designs powered by new technologies; and
3. redefine and then continuously enhance its core business model.

These three areas of innovation are the focus of successive sections of this book and are challenges faced by every corporation, large or small, in the journey toward internally generated growth.

IBM's turnaround is an amazing story of how a management team was able to integrate innovations in all three dimensions, first to save its core business in transactions processing and then to leverage new technology to new market applications. These new applications focused on Web-centric, on-demand computing across major industry vertical markets.[1]

Dark Days at IBM

The early 1990s were dark days for IBM's mainframe division, the computer giant's traditional powerhouse. The company had completely missed or ignored the client-server technology boom of the mid-1980s, and by the early

1990s, its traditional batch mainframe was sinking. Centralized mainframes controlled in the “glass house” DP shops of big corporations were being replaced with servers decentralized throughout major corporate customers. The engineers who had worked for so many years in IBM’s storied research centers saw their careers coming to an abrupt end. They felt adrift in the sea of technological change.

IBM’s R&D centers had been the hallowed ground of computing innovation. The Poughkeepsie center had been the heart and soul of mainframes, developing most of its hardware and operating systems technology (essentially multiple virtual storage, or MVS). The Boeblingen Development Laboratory in Germany had been working on mid-range mainframes (IBM S/390 class machines for small business or office environments).[2] Endicott, New York, had developed the mainframe service processor, and certain corporate staff functions were based in another facility in Somers, New York.

First launched in the late 1950s, IBM’s mainframes, operating systems, and databases had become the cornerstones of data processing for the world’s large corporations. In his classic *The Mythical Man Month*, Frederick Brooks described the development of OS360, which during the 1960s was a pioneering as well as proprietary operating system software.[3] This work, as well as advances in hardware, encryption, and storage, produced a great tradition of innovation in IBM. The mainframe became the king of computing, and throughout the 1970s and 1980s, the engineers who worked in mainframe development centers had every right to feel like kingmakers. Over the next 40 years, technological advances and the sweat of thousands of engineers led to successive generations of increasingly powerful million-dollar-plus machines.

Technological advances in computing over the following decades were breathtaking. By 1995, the effective price per million instructions per second (MIPS) was only 1 percent of what it had been at the launch of the System 360 in 1964. IBM’s earliest mainframes were priced at about $256 million per MIPS in the late 1950s; by 1970, about $2 million per MIPS; by 1980, $350,000 per MIPS; by 1990, $100,000 per MIPS; and by 1995, less than $20,000 per MIPS. By the turn of the millennium, the approximate cost per MIPS in a high-end mainframe—IBM’s z900 (a recent mainframe)—had tumbled to $2,500. From $256 million to $2,500 per MIPS: a 100,000 times improvement in price performance.

By the late 1980s, IBM’s S/390 division accounted for more than $10 billion in annual revenue, and more than 30,000 employees were working on the product line in one way or another. The division was, in its own right, one of the largest manufacturing companies in the country. Everyone within IBM felt that S/390 division remained the heart and soul of the corporation. As it fared, so did the rest of "Big Blue." But by the late 1980s, chinks had begun to appear in IBM's armor in the mainframe business and the corporation as a whole. Large customers were asking for features that the company could not provide in its mainframe products, such as Internet- and intranet-style networking and applications software for online business and collaboration. By the early 1990s, the chinks in IBM's armor had become cracks—cracks that grew into fissures.

Perhaps complacency had set in; the S/390 division might have been too dominant for too long. What is clear in hindsight is that IBM's organization, processes, and approaches to new product development were focused too much on incremental innovation for existing product line architectures. The majority of its resources were allocated to sustaining current product lines. Risk taking had given way to risk avoidance.

For IBM, the move from traditional batch mainframe computing to client server computing and then to distributed peer-to-peer Web-based computing for e-business seemed an insurmountable hurdle. Writing in *Business Week*, John Verity summed up the situation:

> There was a stern John F. Akers telling Wall Street analysts and reporters of a plan: shed 25,000 employees, cut $1 billion from product development budgets, cut $1 billion from administrative expenses, and take a $6 billion charge against 1992 fourth-quarter earnings to cover the terminations and asset sales.[4]

And a few months later, Verity wrote again:

> No question, 1992 was a disaster for IBM … the biggest net loss in American corporate history, a 50 percent plunge in its share price, and a hail of criticism that led IBM's directors on January 26 to seek a replacement for Chairman and CEO John F. Akers. Nearly lost in the headlines, however, was a particularly alarming revelation: IBM, it seems, has even lost its touch in mainframe processors and storage systems—a $50 billion worldwide industry that it has dominated for 25

> years. While overall sales of such equipment grew by an anemic 2 percent in 1992, analysts figure that IBM's mainframe processor revenues dropped 10 percent to 15 percent in 1992, to about $7.5 billion. At the same time, mainframe rival Amdahl Corp. posted a 48 percent revenue gain. Unisys Corp., the No. 2 mainframer, reported that its mainframe sales jumped more than 10 percent over 1991. Clearly, the mainframe is not dead. What is dead, though, is revenue growth from the old-style machines, such as IBM's current System/390.[5]

Major technological discontinuities were sweeping across industry, directly challenging IBM's traditional approach to computing. Mainframes had traditionally run the transactions-processing applications in large corporations. For users to get data from them onto their PCs, the PCs were made to emulate mainframe terminals, and then "screen scraping" programs grabbed the data for the user's spreadsheet or some other PC-based application. Client server computing offered a much more elegant and flexible solution for sharing programs and data between large and small computers. Programs and data could be seamlessly downloaded from central computers onto any other computer on a local area network and then run on those computers. By the early 1990s, Fortune 500 companies were adopting Unix- and Windows-based client server solutions wholesale, directly threatening the traditional mainframe solution. Independent software companies such as SAP and Oracle offered client server applications that covered nearly all major operating functions across the enterprise.

There was also a strong trend away from customer applications software to off-the-shelf packages. IBM, again, was not moving with the trend. As a result, these packages were being developed largely for Unix and Windows environments. Data communication between sites was also changing from IBM's proprietary networking standard to Internet addresses: "IP" (Internet protocol) was destroying "SNP" (IBM's proprietary systems network protocol).

IBM's sales force covered up for its aging technology. Its sales force and account management methods are probably the best ever developed in the computing industry. IBM's top salespeople literally became part of their customers' decision-making processes with respect to computers, software, and networking. As the 1980s gave way to a new decade, however, even the world's best sales force could not stem the tide. Customers' needs were changing, and IBM was not responding.

The bread-and-butter customer for the S/390 division was the large, global bank, trading house, insurance company, retailer, or manufacturer. But their needs and organization structures were changing rapidly. These customers were seeking reduced overheads, faster production cycle times, and partnerships with other corporations. These business demands called for distributed computing capable of more closely integrating and synchronizing operations on a global scale, including client-server ERP (enterprise resource planning) systems. Customers also needed an array of Web solutions and office productivity applications within a highly secure environment.

Managers at IBM were not oblivious to these changes. For years, however, they failed to marshal the collective energy and resources needed to address them head-on. And so the company began a long slide toward the abyss.

How deep was the abyss? In 1990, approximately 30,000 IBM employees were working in some way on the development of the hardware, software, and peripheral systems associated with mainframes. Reported net earnings were approximately $6 billion. A year later, the company reported a small net loss. In 1992, the loss approached $7 billion. By 1993, losses exceeded $8 billion! IBM's losses were as large or larger than the annual revenues of many other computer manufacturers. These problems were felt most strongly in the S/390 division, the home of mainframe, where the top thousand S/390 customers accounted for nearly 70 percent of the division's $11 billion revenue.

Hit from Above and from Below

IBM had two major traditional competitors, Amdahl and Hitachi, during the years of crisis. These rivals began offering features not found in IBM products. While Amdahl never beat IBM at the MIPS game, it consistently sold similar class machines for 20 percent less. Its market share hovered between 4 percent and 7 percent. Hitachi, on the other hand, was working hard on beating IBM at the bipolar large-scale uniprocessor game, just as IBM was abandoning its own bipolar technology for the first generation of mainframes based on CMOS microprocessors.

To hold a logical state (on-off) within a bipolar logic circuit, current is always running through the circuit. Within a CMOS circuit, a burst of current is needed to switch the logical state, but the state is maintained by a much smaller current. This has an enormous impact on power consumption. It also affects size. The continuous current in a bipolar requires circuitry large enough to handle the heat generated. Circuits that were too small would literally fry. At a chip level, a bipolar requires far more energy than CMOS, so these machines required water-cooling systems in addition to the traditional air-cooling from fans. These water-cooling systems were elaborate mechanical developments with a copper piston positioned on the back of each chip, held in place with a retaining housing, and cooled by chilled water flowing through a cold plate.

The analogy might be one house with very powerful lights always left on, day and night, versus a similar house with equal lights controlled by motion sensors, illuminating only when someone walks into a room and then turning off when the person exits. The catch was that while IBM's first CMOS machines were a lot more energy- and space-efficient, they were also a lot slower than the latest versions of the older bipolar chip architecture. And given its huge deficits, IBM did not have enough R&D money to develop both architectures in parallel until CMOS gained the performance advantage.

Hitachi's decision to make a "killer" bipolar machine had a predictable consequence: The next version of Hitachi's bipolar mainframe was even faster than its or IBM's predecessors, and it was at first considerably faster than IBM's first generation of new CMOS-based mainframes.

In 1993, Hitachi introduced the Skyline bipolar microprocessor in its mainframes. These new machines had literally twice the computing speed of IBM's own bipolar mainframes. IBM's large customers—airlines, financial services firms, and retailers—began buying Hitachi's computers for machine-intensive applications. Within several years, Hitachi had taken 9 percent of market share, reducing IBM's share to the mid-70 percent range. This translated into almost a billion dollars of new annual revenue for Hitachi, and a billion dollars less for IBM! That was a billion dollars forfeited to a competitor coming into the market with a better "high end" solution, albeit one based on an aging architecture.

As IBM was being attacked from an expected source, a new set of entrants were gaining even more ground. This was technology from "the low end," multiprocessor versions of RISC-based workstations packaged into small mainframes called "servers." These were running client server operating systems software, which at that time was some version of Unix. Today, that software is Linux, and it is the foundation of open systems computing. It was only a matter of time before these “small” machines would grow to become more powerful mid-sized machines. By the end of the 1990s, Sun Microsystems was offering a RISC-based machine (the UE-10,000), regarded by many as the fastest mainframe on the market for commercial applications. Sun, and other competitors such as Hewlett Packard, had beefed up their machines to achieve mainframe-like throughput.

Even the traditional mainframe had multiuser access to applications and could therefore be called a server. However, the mainframe environment was generally a closed one, living in its own glass house, and not well suited for heterogeneous computing environments. The new breed of servers offered by Sun, Hewlett Packard, and many others was far more suitable to the client-server architecture that raged through industry during the late 1980s and 1990s. Unix, and later Linux, was the key. These computing environments are naturally suited for the peer-to-peer intranet and Internet environments that mushroomed in the late 1990s and are dominant today.

These new machines were also far more scalable than traditional mainframes. A DP manager could start small and then cluster new systems within a distributed processing environment. By the year 2000, a Sun SPARC-Solaris workstation could be purchased for less than $5,000; its high-end UE-10,000 came in at about $1 million. An entry-level mainframe from IBM, in contrast, cost about $250,000, and its high-end machines cost several million dollars. While less powerful in terms of throughput than IBM’s biggest mainframes, these Unix-based machines nonetheless provided significant functionality at a fraction of the price.

Hit from above, savaged from below, IBM had to rearchitect not only its mainframes but also its entire approach to designing, engineering, and manufacturing its products.

By IBM’s own estimate, some 80 percent of e-business computing procurements by large corporations in the period 1997–2000 were servers from Sun, storage systems from EMC, networking routers from Cisco, and database software from Oracle. IBM was largely missing in action.[6] Collectively, these companies were staking claims to different segments of the e-business world, and their sales were growing. In 1993, Sun Microsystems reported $4.7 billion in revenue, Compaq $7.1 billion, and Hewlett Packard $20.3 billion. By 2000, Sun had tripled its revenues to $15.7 billion, and Compaq was running at a $38.5 billion clip (it was soon to be acquired by

Hewlett Packard), and Hewlett Packard itself reported year-end sales at $48.7 billion.

These competitors were not IBM's only worry; it had to face the reality that there was a new breed of CIO in its core market. Web-focused, e-business aware, this new CIO didn't demonstrate the vendor loyalty that characterized his or her predecessor. The DP people being hired out of college by the Fortune 1000 knew C, HTML, and Java. Schools were not even teaching traditional mainframe computing infrastructure and COBOL programming anymore! The "career customers" who would buy equipment and software only from IBM were disappearing.

These developments spelled disaster for IBM. It still had a strong share of traditional markets but was being attacked by mainframe competitors from one direction and by Unix server manufacturers from another. Worse, it hadn't seriously joined the industry-wide race for development of a new e-business infrastructure. Its once commanding 90 percent market share for all server applications had plummeted to 40 percent. Staff morale was terrible and layoffs were in full swing. By 1995, IBM was developing new mainframe computers and operating systems software with *one third* of the staff it had in 1990.

Mandate from the Top

Change at IBM started at the top. For decades, IBM had promoted career employees up through the ranks, even to the CEO level. Faced with what many thought was the prospect of bankruptcy, the board agreed that it was time for a change. They brought in an outsider, Louis Gerstner, to run the company.

Gerstner's challenge was massive: IBM was focused largely on slow-growing markets. It was also slow in bringing new products online. An internal study had shown that the average cycle time for developing a new mainframe model from start of R&D to launch was six years![7]

Gerstner demanded and drove change within the traditional bureaucracy. Under his command, IBM adopted a "live by the sword, die by the sword" approach to technological innovation. Small changes in products, organization, and culture would not work. Everything had to change, *without frightening the installed customer base*. Under the new regime, the guiding principle was to make the company's computer designs obsolete with better ones in increasingly shorter time cycles and then *to leverage those designs into new market applications* (which IBM came to call "e-business"). Combined with consulting, systems integration, and hosting services, new hardware and software solutions for these new e-business applications would grow entirely new streams of revenue and restore the company to profitability.

IBM's traditional home base, mainframe computing, became the focal point for change. Four key executives made it happen. First was Nick Donofrio, the division's senior executive; he made the decision to make CMOS pervasive throughout the architecture. Working directly with him was Linda Sanford, who implemented multifunctional team-based philosophies throughout the organization and made sure that new systems were delivered under incredible budget and time pressures. The Donofrio-Sanford team took the S/390 division from its old architecture (known internally as the H series) to its first new architecture, the G series. Dave Carlucci took over the division at the end of 1997 and drove tough decisions regarding open systems server architecture and extending the mainframe brand beyond its traditional online transactions processing (OLTP) roots and into the e-business domain. The result was a second new architecture, the zSeries. Last, there was Sam Palmisano, who became head of the overall server division, which by 2000 included the mainframe group. Palmisano championed Linux and made it both a technical and business reality for IBM. He subsequently became CEO of the corporation and was its chairman at the time of this writing.

Innovation in the Market Dimension

Target market strategy and a rich understanding of customer needs are the first steps in internal, organic enterprise growth. IBM had to change how it viewed its customers. This started with how it segmented its target markets; that segmentation would reveal specific solutions for different customer groups.

	Current Markets			New Markets New Users
	390	Unix	Cost	Web, Data mining
250 > MIPS	Traditional DP			
30–250 MIPS				IBM was not focused on these applications.
0–30 MIPS				

FIGURE 1.1 IBM's Traditional Market Segmentation

In the mainframe division, segmentation had been strictly product focused, as shown in FIGURE 1.1, as opposed to a truly customer-focused segmentation. Customer groups are inferred in the figure but not directly stated. In fact, a single customer might possibly use all of the products shown in the figure. This segmentation approach also encouraged product development silos, with one IBM division making large systems, another making mid-sized systems, another making small systems, and each making or licensing their own particular software. Integration between these different systems occurred largely at the customer site.

You can see the focus of the old S/390 group: large computers delivering more than several hundred MIPS. Although there was a category for *new stuff*, shown on the right-hand side of that figure (titled "Web, Data Mining, etc."), that category was seen largely as software applications and peripheral to division's core business. The S/390 division was all about engineering and selling "big iron." This view of emerging client-server and Web applications as peripheral to the core business virtually assured that S/390 engineers would not focus on e-business requirements when developing next-generation machines. In fact, the division's market segmentation grid from the early 1990s could be expressed as a single segment cell: high-volume online transactions processing.

A customer-focused, integrated approach would have shown the migration of customer needs toward client-server computing first and then distributed Web-based computing next. These needs would have then driven more rapid change and integration of IBM's hardware and software products. This was precisely the innovation in market segmentation and user needs research that occurred in IBM during the mid-1990s.

At the time, the S/390 division had no internal marketing staff. Management relied instead on consulting firms. The consultants tried a method that was highly sophisticated and considered state of the art. It differentiated between types of buyer behavior. More than 20 different decision-making categories were identified. Unfortunately, the division's engineers found this behavioral approach confusing. For example, two different types of decision-making behaviors could demand exactly the same type of machine, with the same type of operating systems, database, and data communications software.

IBM then decided to pursue what now seems like an obvious approach: segmentation by industry verticals and company size. The new market segmentation framework is shown in FIGURE 1.2. The division assigned its own product development team to the job of understanding user needs. This team, and the executives sponsoring it, soon learned that users in different industries had different priorities for features such as scalability, heterogeneous environments, and security. Customers within these vertical markets also had different needs, based on their size, in terms of throughput requirements, cost constraints, and the type and structure of complementary services to be provided. IBM set about getting its facts straight on each of these issues.

Market-Driven: Industry Group, Company Size, Incorporating Emerging Markets

	Financial	Distribution	Mft	Telecom	Life Sciences	*Etc.*
Fortune 1000						
Medium						
Small						

IBM focused its user research on the needs/problems for different size customers in each segment, looking at:

- Reliability of equipment
- Scalability of system (spikes in transaction volumes)
- Compression of data (for faster distributed computing)
- Security requirements (cryptography)
- Software needed for e-business and CRM applications

FIGURE 1.2 IBM's Market Segmentation for Server Technology

The vertical market approach also allowed IBM to get very specific about the needs of emerging customer groups, such as life sciences. In the old environment, a mainframe was a mainframe. In the new environment, IBM aimed to sell specifically tailored solutions to different customer groups. It no longer assumed that a pharmaceutical company was like a financial services firm, or a brokerage, or a manufacturer.

More important, the segmentation framework included small and medium-sized enterprises—not strictly industry giants. This encouraged IBM engineers to begin thinking about common architectures that could be scaled to accommodate the full range, from small servers to very large servers. The new market segmentation recognized that customers no longer viewed different-sized computers as different universes. This changed the language used within IBM. Instead of talking about mainframes, minicomputers, and workstations, people used the term "server." Using the same descriptor for different models had a powerful effect: engineers—like their peers at Sun, Dell, and Compaq—began thinking about a common, scalable architecture across the entire spectrum.

IBM's market research proceeded along very specific dimensions of user needs, some of which are shown in FIGURE 1.2. The company wanted to know precisely how customers thought about equipment reliability, how they viewed the dimensions of scalability, and how they thought about data and program security. For example, a large brokerage considered scalability as handling without interruption a 15X spike in standard usage on its online transactions processing systems. Just as important, IBM wanted to know how users in different vertical markets thought about mining data about their customers' purchases and how large corporations shared information with their suppliers and channel partners. The answers to these questions became the design drivers that guided the performance parameters of many of the hardware and software subsystems developed for the next generation of large-scale computers.

Innovation in the Technology Dimension

Technological innovation was IBM's next fundamental dimension for renewal and growth, and product line architecture was the point of the spear.

A product line architecture is simply defined as the major subsystems and the interfaces between those subsystems that are common to all the specific products or models within that generation of the product line. Good architecture, which is modular and scalable, provides the flexibility needed to meet ever-higher levels of performance and to run new types of applications. Poor architecture is limiting. IBM's old mainframe architecture cornered it into the box of online transactions processing (OLTP). IBM's new architecture runs ever-higher loads of OLTP and, at the same time, has separate, dedicated resources for zipping through Linux and Java applications. That's scalability and flexibility all in a single, robust architecture.

Within IBM, product line architectures are referred to as *reference architectures*, where numerous specific models adhere to the major subsystem stated in the architecture, as well as the interfaces between these subsystems. Subsystems that are made common across multiple products—and better, multiple product lines—are the true product platforms. Many corporations make the mistake of seeing their product line architectures as product platforms, because their customers view platforms that way. However, this often leads such firms to try to force-fit a solution into a new market application for which it was never designed and for which it is not really suitable. For example, in IBM, a microprocessor is considered a product platform that can be leveraged across the distinct architectures of mainframes, mid-range servers, and workstations, just as Honda leverages a common engine across a passenger car and an SUV.

The H Series Mainframe

During the 1990s, IBM shifted through three distinct product line architectures; prior to this, the company's product line architectures tended to last a decade or more. IBM entered the 1990s with a mainframe architecture call the H series, which was based on a 31-bit bipolar microprocessor.[8] Bipolar circuitry was at its core, making H series machines very large, energy intensive, and hard to cool. (Technophiles will relish the detailed description of IBM's technology strategies, mainframe architectures, and subsystem evolution in the appendix.)

IBM's mainframes were large, a thousand square feet large. FIGURE 1.3 shows an old picture of the glass house with a large mainframe and external storage devices that included disks and tapes for backup. Today, the same picture might show one single four- by eight-foot cabinet containing a System z computer delivering more than ten times the MIPS capacity and also a similarly sized cabinet containing a multiterabyte disk storage system!

These old mainframes had to be delivered on a large tractor-trailer truck. The frames were forklifted into the facility. A team of six to eight people put the frames together and connected the cables, power supplies, and necessary plumbing. The facility had to have a raised floor to accommodate the plumbing. A testament to IBM's field engineering capabilities, a massive water-cooled machine was up and running in one or two days.

FIGURE 1.3 Inside the Glass House: Series Bipolar Processor Mainframe Computers, Circa 1990–1994 (IBM. Reproduced with permission.)

The last derivative product of the H series, the H6, contained up to ten 60 MIPS processors within one system's physical package, and it delivered 450 MIPS. This "old" mainframe was a phenomenal piece of electronic and mechanical engineering. High-voltage electronics were in immediate proximity to copper pipes filled with chilled water to dissipate heat. Cables, as opposed to electronic buses, connected the processors, memory, and I/O channels.

The G Series: The CMOS Mainframes

By 1993, with the H series as its major hardware offering, IBM's losses reached $8 billion in a single year. Within the S/390 division, the writing was on the wall. IBM's investment in bipolar semiconductor fabrication was enormous, but the only engineers using these chips were in the S/390 group itself. Management also knew that bipolar technology could not keep pace with customers' performance requirements. The technology roadmaps were very clear. IBM was advancing its own bipolar processor speed by 18 percent per year in the early 1990s, whereas CMOS speed was surging ahead at 50 percent gains per year. Mainframes themselves would have to double in power every several years to keep pace with customer demands—so IBM believed. Soon, a bipolar mainframe would have to fill a small house! In an age where hardware budgets were forecast to level, if not decline, that path was not tenable.

In 1993, two very different design plans were floating around the S/390 division. One plan was to build a new mainframe that would provide all the power (450 MIPS) of the H6 but do so with CMOS technology. Staff in Poughkeepsie, the traditional hearth and home of mainframe engineering, championed this plan.

The alternative plan was to make a CMOS mainframe that was substantially less powerful, but for which development would be far less risky. The first CMOS machine would be followed quickly by upgrades to increase throughput. A design group in IBM's lab in Boeblingen, Germany proposed this plan. The German lab had already been working on CMOS microprocessors for smaller IBM machines.

IBM's senior management went with the German plan. It urgently needed a new architecture, and it liked the Germans' CMOS experience. Somewhat shocked, the Poughkeepsie engineers found themselves having to implement the European design.

Toward the end of 1994, after an incredibly intensive two-year effort, IBM released its first CMOS-based mainframe. This new machine—the G series—aimed to stem loss of market share to Hitachi and Amdahl—but it took almost four years of constant improvements to do so. This was a great technological step forward, setting the stage for future developments. However, IBM was still serving *current users and current uses*, that is, large corporations performing online transactions processing.

The first product, the G1, was only a tenth the size of the last H6 and used 13 percent of the electricity. It used an electronic bus instead of cables, and fans instead of pipes for cooling. These benefits translated into a much lower total cost of ownership for customers.

FIGURE 1.4 graphs the performance of the new G machines versus the old H series (as well as the subsequent zSeries). Note that the 450 MIPs in the H6 versus the 100 MIPS in G1. The new machine was four times slower than the last model of the old architecture!

Consider the fortitude needed during this transition period. At 100 MIPS, the new G series would be much slower than the H6. Several years of hard work would be needed before the new CMOS machines would catch up to their bipolar predecessors. At the same time, Hitachi was expected to introduce its Skyline bipolar machine, whose speed was anticipated to eclipse IBM's current H6. IBM would lose a billion dollars of revenue a year to this last of the giant bipolar mainframes. Management had to convince engineers and marketers alike that this fundamental architectural change was necessary and could no longer be delayed.

Another engineering team—outside both the Poughkeepsie and Boeblingen labs—came up with a way to beat the 100 MIPS problem: tightly couple and share loads across separate G computers. This coupling technology, called Sysplex, went a long way toward preserving the installed base during the transition from H to G. Sysplex allowed customers to attach G series machines to existing installed H machines to increase capacity without discarding all prior investments. A customer could achieve the power of an old H6 with four coupled G1s.

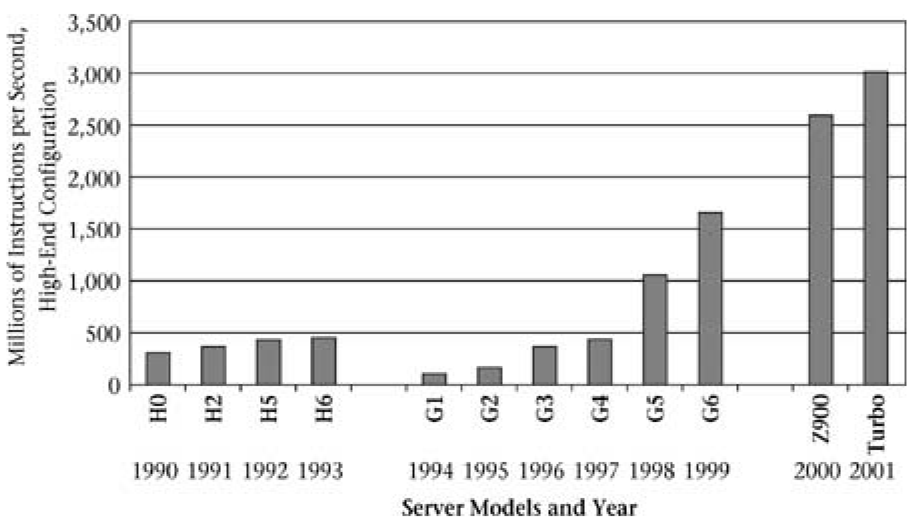

FIGURE 1.4 Three Generations of IBM Mainframe Performance: From H to G to Z

Despite the risks and pressure, senior management persisted with the CMOS mainframe design. At the time, there was no way that management could have fully anticipated its success. Had G development been seriously delayed, or Sysplex not worked, IBM would have lost many customers. Thus, G series development was a "bet the company" move. Fortunately, the G series architecture had a clear path to add more processors and cache memory. Subsequent models offered accelerating performance.

The G3 (released two years later in 1996) provided even greater benefits: a 95 percent reduction in power, a 98 percent reduction in cooling capacity, and a 90 percent reduction in floor space relative to the old H6. *And* it was as fast as any H machine made by IBM. By the time G4 hit the market, the tide had turned. The system was as powerful as anything in H and offered a much lower total cost of ownership. It had more microprocessor buses, more banks of shared data, and substantially greater capacity to read and write data to networks and storage devices. The G5 that followed in 1998 was twice as fast as the H6, putting both Amdahl and Hitachi under severe competitive pressure IBM pursued its advantage. By 2001, Hitachi and Amdahl exited the mainframe business altogether. IBM was now positioned to take on its other set of competitors, producers of Unix-based servers, and extend its technology to the new market applications of Web-based e-business.

Many readers may ask, "Why didn't the transition to CMOS occur ten years earlier? It was already at least a decade old, and many other companies were using it. IBM itself was using CMOS in other smaller computers. Why was the G series so long in coming?" The answer lies in the design of IBM's organization and its processes. The story of the organizational changes that management made during this time frame is well told in Gerstner's own words and those of a few others.[9]

The Third Generation: The zSeries and E-Business Applications

Having regained its footing in the core business of online transactions processing, IBM was still losing share in the larger server market because it was not adequately addressing client server and more distributed Internet computing. The price of high-end hardware also continued to be a pressing issue. By 1998, for example, the price of IBM's large servers ranged from $250,000 to $3 million per machine. Sun's, by comparison, ranged from $50,000 to $1 million per machine. Also, growth rates for machines over $1 million were in the single digits; growth rates for machines under $50,000 were in the double digits. Additionally, Sun's most powerful computer, the UE-10,000 was fast, reliable, and open to applications and software development tools from the Unix world.

In 1995, the year of the G2, Linda Sanford—a ranking executive in the S/390 division—chartered a team to put together a business plan to compete in the e-business space. IBM would continue to develop successive G series models for the next five years, but in parallel, it would develop a much more powerful architecture: the zSeries.

Parallel development of next-generation architectures is essential if a company wants to have powerful new solutions available before its current ones "run out of gas." Having been caught at great disadvantage once, IBM's senior management was determined not to let this happen again.

Sanford assembled a multifunctional task force of mid-level managers from across IBM, including Europe, and asked them to create the growth plan for IBM's server business. These individuals were thought leaders in their respective fields. None had yet achieved executive rank; an executive on that task force might have clung too closely to dated approaches. The team was named ES2000 (for enterprise systems in the year 2000).

In just three months (spring 1995), the team created a new vision for IBM's future that included a new product line of hardware and software, as well as a new business model for the company. The plan synthesized user needs and competitive changes, new hardware and software architectures, channels, branding, and organizational strategies into a unified whole—and then set forth the investment required to achieve a set of projected financial results. In many ways, IBM has been executing the ES2000 plan ever since.

Carlucci and Sanford did more than champion the next-generation solution for e-business computing; they nurtured and empowered the next generation of IBM's executive leaders. Most these mid-level managers went on to executive positions throughout the company.

The ES2000 team came up with the letter "z" to suggest a new dimension of computing. The x and y dimensions in the old world were performance and price; this new dimension symbolized dynamic management and coordination of all subsystems in the computing architecture. The subsystem that would deliver dynamic management within the "z" architecture was called the intelligent resource director. This was a reincarnation of a capability that had been left behind with the H Series on account of time pressures but that the team wanted to bring back into the zSeries. This feature, *multipoint switching*, is an important part of the secret sauce of today's machines.

As successive G series machines were being developed, IBM engineers fondly recalled earlier approaches to achieving scalability. "Let's bring back the multipoint switching of the H," they concluded. This was initially done for the G5 and extended to the G6, the last of the G series line. However, in the new zSeries, the data flows were substantially larger, making the microcode implementation of multipoint switching much more complex. That implementation became known as the Intelligent Resource Director.

Development of the Intelligent Resource Director ranks as one of the computer industry's largest CMOS design efforts. It dynamically moves capacity between partitions in microprocessor-addressable memory and channels (for I/0 to storage devices and networks). It allows, for example, a brokerage firm to achieve that 15X peak performance over standard utilization without interruption.

The intelligent resource director and many other highly innovative subsystem innovations came to market in the year 2000 as the z900. All microprocessors used in the zSeries were newly designed 64-bit, RISC-based CMOS chips—called Blueflame. The 31-bit microprocessors of the G series could access only two gigabytes of random access memory. Blueflames could access as much memory as IBM chose to place in the box. The combination of the Blue-flame and Intelligent Resource Director innovations created a powerhouse machine. In striking contrast to the earlier H to G transition, a standard z900 systems configuration delivered 2,600 MIPS, which was a *1,000 MIPS faster* than the last machine in the G series.

IBM also sought to make its mark in the software domain by leveraging its mainframe technologies to new users and uses. IBM had been working its way toward "open systems" by providing a Unix emulator within OS/390. However, the new Web commerce applications needed more than just the Unix operating system: The language of these applications was Java components, TCP/IP, and Web programming languages such as HTML and XML. These were foreign to the traditional mainframe world but became the rage in the second half of the 1990s. Fortune 400 accounts were spending heavily on e-business and developing these applications on *other manufacturers*' machines, networks, and programming environments.

To combat this, IBM executives took the bold step of introducing Linux as a native operating system on the G6 in 1999. Even though IBM had tremendous Unix expertise in its Austin, Texas, workstation division, there were simply too many versions of Unix available during the late 1990s to allow true cross-machine software portability. Fortune 400 IT managers worried about version control, applications compatibility, and security.

Linux was new. It also offered an opportunity to achieve excellence in an arena not yet dominated by competitors.[10] At the same time, IBM wanted to prevent the fracturing of Linux that had hurt Unix; it did this by supporting the open standard/open source status of Linux in every way possible.

IBM's German lab had been working in stealth mode to make Linux run on the new Blueflame microprocessor. When managers of that lab made their work known, IBM executives on both sides of the ocean were excited. IBM had a working version of Linux available for its new Blueflame microprocessor. The timing could not have been better.

Dave Carlucci insisted that the entire company—not just the mainframe division—commit itself to supporting and featuring Linux across all its servers. The zSeries was launched with both native Linux and IBM's proprietary operating system called zOS to run traditional applications, such as decade-old COBOL programs.

Once the first zSeries machine—the z900—was launched, development continued to accelerate. IBM introduced the z900 Turbo in 2001 and, with it, broke the 3,000 MIPS barrier. In spring 2001, within only a year of the first zSeries introduction, the z900 was given the "best hardware platform" award at Linux World. Who would have guessed in 1995 that a mainframe would walk away with a major open systems prize?

zSeries has since been renamed System z, and it has been an incredible success. By 2004, it achieved 9,000 MIPS with a 32 co-processor configuration in the zSeries 990. Reflecting a "better, best" product strategy, IBM also introduced at the same time the zSeries 890, a 1,360 MIPS four-processor machine for less computing-intensive applications. To more than a few observers, the 890 was a "Sun killer." zSeries machines were also equipped with special application assist processors, specifically designed to efficiently run the Linux operating system and Java applications while still running core zOS applications on general processing engines.

By 2005, a number of important trends were in full force. Corporations wanted to migrate their financial, manufacturing, customer management, and internal communications systems to an intranet/Internet foundation. IT managers were also consolidating data, systems, and processes on “mainframe servers.” Mining dispersed databases for better customer and product insights became another top priority. And secure access and data protection were as important as anything else. These all played to IBM’s strengths.

Business Model Innovation

In addition to market and technology innovation, IBM made important changes in its business model. As we shall see in later chapters, a business model is more than financial numbers. A business model is not what you do; it’s how you make money doing what you do.

The ES2000 team proposed a radically different approach to IBM’s business model for software. Up to that point, IBM had garnered substantial revenue from licensing its proprietary operating systems to customers. With Linux, IBM made a 180-degree turn. It made the unconventional yet essential decision to use third parties to sell and support Linux, the cornerstone of its foray into open systems computing. IBM would no longer make money on the sale of the operating system. Rather, it would make money on the development and licensing of software created to run on that operating system. Further, it would foster the development of as many third-party Linux tools and applications as possible. To do this, the company had to help software vendors rather than compete with them.

IBM selected independent companies to distribute and support Linux for its new zSeries machines. These included RedHat, Suse, and TurboLinux. It then handed its software development tools over to Linux and either acquired or developed many more. It also applauded when Sybase, SAP, and Oracle did Linux ports for their own products. By the end of 2002, there were some 2,000 off-the-shelf commercial software products available for Linux.

The ES2000 team also saw that IBM would have to invest heavily in improving software development and integration tools for its new mainframes. Today, this is called “middleware.” IBM became one of the market leaders in Java-based middleware (with its Websphere, Information Management, and Tivoli storage management offerings). Getting to this point required a herculean effort.

New business models need to be supported with forceful marketing and branding. Gerstner, coming from Nabisco, understood this and asked the company's Internet Division to form a team similar to ES2000 to assess IBM's product positioning and branding. Three very powerful ideas emerged from the marketing team's work. First, e-business would help IBM's customers increase efficiencies, shorten cycle times, and lower costs in their operations and supply chains. Second, the Web would build a closer connection between companies and their customers, strengthening customer loyalty, and extend market reach to new customers. Third, intranets developed within companies would help them make their own employees more effective and productive. This, in turn, would improve new product development, service deployment, and sales. The marketing team took these core benefits to the next level by translating each benefit into tangible systems and services requirements for all other divisions across IBM. It also created a single, unified promotional message around e-business. Gerstner was prepared to place $1 billion on the table to communicate the new e-business brand.

IBM then made the very unconventional decision to *not* copyright the term "e-business." IBM hoped that other companies would use the term "e-business" in their own marketing, lending credibility to a new concept in which IBM would strive to be a leader in both product and in image. Within five years, Hewlett Packard's brand became "e-services," Compaq's "non-stop e-business," and Oracle's "the engine of e-business." EDS created its new e-business solutions unit, and even Microsoft started touting "the Business Internet." The snowball effect was far greater than IBM could have predicted. Within three years of the e-business brand launch, the company found itself having to differentiate. In 2000, IBM launched "e-business infrastructure™" and allocated another $300 million to this campaign. That brand campaign has continued to evolve and was recently enhanced to focus on "e-business on demand." Then, IBM extended its on-demand brand to "hosting on demand" for application servers, "capacity on demand" (blade servers), and "information on demand" (database access), all reflecting IBM's Web services architecture.

This business model innovation transformed IBM from a company that made most of its money selling "big iron" to one that would provide complete solutions for customers: hardware, middleware, and services that would wrap around the operating systems and end-user applications made by independent companies. Approximately half of the company's revenue during the zSeries epoch came from professional services.

The combined effects of market innovation, technology innovation, and fundamental changes to IBM's business model transformed a company that was on its deathbed into one of the fastest growing and most profitable technology companies in the world. E-business continues to reveal new users and new uses. From a low point in 1993 of $8.1 billion in losses on about $63 billion in revenues, IBM closed out 2005 with more than $91 billion in revenues and $8 billion *in profits*. Like the phoenix, IBM emerged from the ashes.

Take Courage from IBM's Example

What can we learn from IBM's successful turnaround and growth? Perhaps the most important lesson is that a company with good people and good technology doesn't have to go on an acquisition binge in order to grow. It can grow from within—by leveraging its capabilities to new users and uses. Another lesson is that good people and technology are insufficient. A growth-seeking enterprise needs a strategy for leveraging its technologies into new market applications. It also must invest in R&D. IBM would not have pulled back from the brink without the core technology work done in Germany and the systems developed in Poughkeepsie.

A powerful marketing strategy that creates excitement for new product lines and services is also a must. Gerstner understood the power of marketing and branding; e-business and On Demand have helped fuel IBM's resurgence.

Finally, senior management must be committed to growth, willing to structure the organization to better pursue it, and not be afraid to empower teams and *share their risks*. Executives must have "skin in the game."

It would be naive to generalize this much from a single case. But IBM is not unique. Other companies have found the path to internally generate growth through similar marketing, technology, and business model innovations. You'll meet some of them in the chapters that follow. But first, let's turn to a framework you can use—a practical roadmap—for getting on the fast path to growth.

Notes

1. I wrote this chapter with the valued help of Mark Anzani, an executive of

IBM. It draws on the work we did (with George Walsh, another executive with IBM) for an article in *Research Technology Management*, which is the journal of the Industrial Research Institute. Peter Tarrant, an executive leading IBM's branding initiatives during the turnaround, was another invaluable source of information. See: Marc H. Meyer, Mark Anzani, and George Walsh, "Innovation and Enterprise Growth: How IBM Develops Next Generation Product Lines," *Research Technology Management*, July–August 2005, 34–44.

2. IBM, MVS, VTAM, DB2, S/390, AS/400, RS6000, zSeries, and System z are registered trademarks of IBM. Solaris and Java are trademarks of Sun Microsystems.

3. Frederick Brooks, *The Mythical Man Month, Anniversary Edition: Essays on Software Development* (Reading, MA: Addison-Wesley, 1995). Also see Carliss Y. Baldwin and Kim B. Clark, *Design Rules*, Volume 1 (Cambridge: MIT Press, 2000).

4. John Verity and Stephanie Forest, "Does IBM Get It Now?" *BusinessWeek*, December 28, 1992.

5. John Verity, "Guess What: IBM Is Losing Out in Mainframes, Too," *Business-Week*, February 8, 1993.

6. By the turn of the millennium, IBM would be competing head to head with Sun, supplying drives to EMC, and competing with it on storage systems. IBM had also sold portions of its networking division to Cisco and was working in partnership with Cisco for the mainframe business.

7. Marc H. Meyer and Paul Mugge, "Make Platform Innovation Drive Business Growth," *Research Technology Management*, January–February 2001, 25–39.

8. Customers would know products based on that architecture as the 9021 family.

9. Louis Gerstner, *Who Says Elephants Can't Dance*? (New York: Harper Collins, 2003); Marc H. Meyer, Mark Anzani, and George Walsh, "Organizational Change for Enterprise Growth," *Research Technology Management*, November–December 2005, 48–56.

10. Linux was initially created by Linus Torvalds, then a student at the University of Helsinki in Finland. He had an interest in Minix, a small UNIX system, and decided to develop a system that exceeded the Minix standards. He began his work in 1991, when he released version 0.02. He kept improving

the system and released 1.0 in 1994, just as IBM was working on its first CMOS mainframe. By the end of 2000, Linux had had almost several dozen major kernel releases.