IN SEARCH OF STUPIDITY

Over 20 Years of High-Tech Marketing Disasters

SECOND EDITION

Merrill R. Chapman

Apress®

In Search of Stupidity: Over 20 Years of High-Tech Marketing Disasters, Second Edition

ISBN-13: 978-1-59059-721-7

Lead Editor: Jim Sumser
Editorial Board: Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick, Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Dominic Shakeshaft, Jim Sumser, Keir Thomas, Matt Wade
Project Manager: Elizabeth Seymour
Copy Edit Manager: Nicole LeClerc
Copy Editor: Kim Wimpsett
Assistant Production Director: Kari Brooks-Copony
Production Editor: Katie Stence
Compositor: Dina Quan
Proofreader: Elizabeth Berry
Cartoon Artists: Marc F. Richard (http://www.lartiste.com), Wolfgang Niesielksi
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Library of Congress Cataloging-in-Publication Data

Chapman, Merrill R., 1953-
In search of stupidity : over 20 years of high-tech marketing disasters / Merrill R. Chapman. -- 2nd ed.
p. cm.
Includes bibliographical references and index.
ISBN 1-59059-721-4
1. Computer software industry--Management--Case studies. 2. Computer industry--Management--Case studies. 3. Business failures--Case studies. I. Title.

HD9696.63.A2C53 2006
338.4'7004--dc22

2006029296

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit http://www.springeronline.com.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

thirteen

ON AVOIDING STUPIDITY

I periodically give lectures at such venues as technology trade shows, product managers' groups, and high-tech councils, and before beginning my talk, I always ask this question: "How many people have read the following books?" I then list some of the seminal publications on the history of high tech. These include such books as *Apple* by Jim Carlton, *Gates* by Steve Manes, and *Hackers* by Steven Levy. Invariably, only one or two hands go up; often, none do.

Then, I ask how many people have read the newest, hottest, the-promised-land-is-within-your-reach-if-you-just-follow-the-diktats-of-this-newest-business-guru wonderbook. The latest wonderbook differs from year to year and from decade to decade. In the 1980s it was, of course, *In Search of Excellence* by Thomas J. Peters and Robert H. Waterman and its myriad of profitable spin-offs (all based on an original foundation of bogus data). In the early-to-mid 1990s it was often *Crossing the Chasm* by Geoffrey Moore and its myriad of spin-offs (all based on product life cycle models that were first introduced in the 1950s). By the late 1990s and early 2000s it was often *The Innovator's Dilemma* by Clay Christensen, which discusses how established companies have a hard time dealing with new ideas and the very successful follow-on, *The Innovator's Solution*, which proposes a solution to the problem the author of the book admits no one has ever actually used (which is a rather innovative way to end a business book series, when you come to think about it).

Now, in all fairness, many of these business books offer practical, if often generic, advice about how to run a business and the best things to do while you're doing it. Like most exercise machines, many of these books "work" if you rigorously follow their commonsense advice. In most cases, thinking up new or improved products or services to sell to people (this process is currently being lionized as "innovation," and businesses have been doing it since the pyramids, but apparently the new label makes everyone feel even better about the process), being open to new ideas, treating customers well, organizing your data, hiring good employees, not committing accounting fraud, and so on, and so on, will certainly improve your chances of success. But this is rather like saying that breathing increases your chance of competing in the 100-yard dash. It will, but mere respiration is not what separates winners from losers in a race.

The danger comes when you dig into the specifics and try to apply the generic to your specific business and its challenges. Most writers of "theory" books can't overcome the tendency to fit the facts into their grand frameworks, leading to a lot of misleading and contradictory advice. You already know the problem with *Excellence*. The *Chasm* books sometimes work well when talking about enterprise markets with fairly well-defined buying processes, but if you had relied on their advice during the microcomputer market's early growth spurt in the 1980s, you would have been caught utterly flat footed by the rapid pace of events (Moore tries to deal with this problem with a later book that acts as a retrofit to the original theory, but it's unconvincing). The *Innovator* books, written by an academic who has never worked in business, proffers a solution that has never been demonstrated to solve anything.

It's not just high-tech firms that get themselves into hot water in this regard. Super-duper consulting firm McKinsey first wrote about and then introduced the concept of "eagles flying high" at Enron, a theory based on the belief that by hiring lots of smart people and letting the wind beneath their super-intelligent wings push them into the stratosphere, Enron profits would soar to ever loftier heights, clutched safely in the talons of all these Einstein flyers. Unfortunately, the theory didn't take into account that really, really smart people might, in the interests of self-enrichment, create myriads of business deals and projects that objectively evaluated had little or no chance of turning a profit and then create a dizzying array of interlocking shell companies where accumulating debt could be buried, all at the expense of stockholders and company employees' retirement funds.

Another problem with all business books that focus on grand theories of business success is that, in a very real sense, no such theory can ever exist. To help understand this concept further, take a quick look at a popular Hollywood fantasy, that of the young go-getter who develops a surefire way to "beat" the stock market. Now, suppose this fantasy could be translated into reality. Imagine that through the use of supercomputers and sheer genius programming, you create a stock-picking system that infallibly predicts which stocks will go up and down and then write a book releasing this information to the world.

What would happen?

What would happen would be that the stock market would immediately congeal into immobility and would have to be rejiggered to work in such a way that all your good advice and smart programming would be rendered useless. This goes back to the fundamental reality underlying all market-driven systems: there must be a winner and a loser in every transaction for the system to work. (It's a grim fact, but before you run shrieking into the comforting arms of Marx and Lenin, the empirical evidence suggests that communism simply creates losers all around.)

This carries over to the competitive environment all companies must endure in market-driven economies. Competition must winnow the myriad of firms over time to ensure the market can function. Failure must happen. But failure must also always have a cause.

The Main Causes of Failure

The main reasons for company failure can be broken down into four basic types:

- Your company is based on fraud and/or the sale of illegal products and services.
- Your company is built around an unrealistic or ridiculous business assumption.
- Your company does not have a strategic vision and plan for success.
- Your company has failed to execute business basics in the course of selling its products and services.

In regards to the first two types of failure, I don't have much advice to give. If, like Enron, ZZZBest, and thousands of other companies over the course of the 20th century and continuing into the 21st, your underlying business model is a Ponzi scheme, your business will fail, and maybe you will go to jail. If your company plans to market heroin or cocaine in the United States, you will fail and probably go to jail if someone doesn't shoot or decide to dismember you with a chainsaw first during a dispute about optimal distribution strategies and reseller

margins. If, like a very bright gentleman I spoke to during the course of writing the second edition of *Stupidity*, you intend to bring to market a new word processor for Windows, you will fail, and no one will even pay attention to you. If you're the new SoftRam, please send me a shrink-wrapped version of your product for my collection.

The third class of failure, lack of strategic vision and planning, is, as you've seen, the one most business writers like to write about primarily because books about this topic tend to make the most money. Of course, all successful companies have to develop and sell products and/or services people will pay for; this is the essence of modern commerce, and it is here that a "strategic" plan is both useful and necessary. But few companies and theorists are content to leave it at just that. For the past several decades, American business has been obsessed with the idea that somewhere out there exists a grand unified theory of business that explains once and for all how success can be guaranteed if only the theory can be uncovered and explained. For a time "excellence" seemed to provide the Great Answer. Then the path to proven success appeared to encompass leaping like gazelles over chasms. Now, many believe institutionalizing innovation (something of an oxymoron) into the corporate genome is the first step on the path to enlightenment.

American companies are obsessed with this concept of strategy and are always recasting and reorganizing themselves so as to realign with the latest, greatest strategic vision as brought to them by a newly minted business guru or a shiny new CEO. Caught in a tautological loop, far-reaching business plans are developed that are excellent or that leap far enough or that are innovative because excellent or leaping or innovating is what we do. Things usually go well at first. Americans, despite romantic self-images of rebel cowboys and sturdy nonconformists, are actually pretty good at organizing themselves and taking orders. We're not as antlike as the Japanese, of course, but we do stand patiently in lines, stop at red lights, wait our turn, and take orders from authority with a fair degree of alacrity. So when plans and dictates come down from on high, companies can usually be whipped into fighting trim in fairly short order. A laser-like focus is brought to the creation of new marketing and sales campaigns. The competitive terrain is analyzed thoroughly via market research and focus groups. The distribution channel is primed with promotional money, advertising, and collateral. New products and

services are manufactured in record time with maximum efficiency. The company maneuvers with stereotaxic directness toward its launch point and pauses in readiness, waiting for the command to move out. When the clarion call comes, the entire firm surges forward in lockstep unison, eyes set straight ahead on the prize, moving in a determined sweep to clear from the field of battle any obstacle that stands in the way of the ultimate victory and triumph.

But then things start to go badly wrong. You unleash a new word processor, and your entire company falls into the mud of massive confusion and market resistance because you've made a fundamental positioning mistake (MicroPro). You launch a hot new microprocessor, and because you've failed to realize that now that you're a consumer brand, the PR rules have changed, and the charge forward has been halted by incoming fire from the press (Intel). You become the Internet's most strategic site by dint of a marketing campaign that secures every corner of the globe with platoons of floppies and CDs, and then find yourself in full retreat as everyone stops using phones to connect to the Internet and starts using high-speed connections (AOL). You attack the market for digital content in 1998 with a can't-miss device, an MP3 player the size of today's iPod…and hardly anyone pays attention (Saehan/Eiger Labs).

And then an awful realization bursts upon you. Business is not…war, at least not conventional war. Innovation doesn't always lead to success; failure to innovate sometimes leads to disaster. Markets are not terrains that can be swept clear of enemies in all conquering waves. What is excellence in the context of one industry is a waste of money in another. If business is war, then it's an odd sort of ongoing guerilla conflict in which the enemy can be an opposing company one day and a division or business unit at your own firm the next. Markets are swampy, Escheresque lumps of chaos studded with redoubts and obstacles that disappear and reappear from any direction, studded with over and under ramparts onto which confused invaders stumble and then stagger off from view. And even when you succeed in your objectives and take the field, sometimes the field disappears beneath you, and you find yourself slogging about in a pale foam that obscures your vision and leaves you wandering directionless in a vast wilderness.

And sometimes you get amazingly lucky.

For instance, let's take the success of Microsoft Windows, to date high tech's most dizzying product triumph. Overcoming its humble roots as a clumsy imitation of the far more sophisticated Macintosh operating system, Windows's success from 1990 onward drove Microsoft by 2005 to more than $40 billion in revenue and 60,000 employees, with 2005 profits exceeding $3 billion. Windows was first announced in 1983 when the GUI wars were first taking shape in the wake of Xerox's pioneering work in the field and the first version was released in 1985. Over the years Windows bested GEM, VisiOn, GeoWorks, the Mac OS, and, most notably, OS/2 in the war for supremacy. What clearer example could exist of a company having a strategic vision for a product and then pursuing that vision to ultimate success?

But for Windows to achieve its current monopoly position, the following events had to occur:

- Xerox, the original inventor of what we now call the *graphic user interface*, had to never develop a clue about how to commercialize most of the groundbreaking developments that came out of its PARC labs.
- Digital Research had to blow off IBM when it came calling for an operating system for the original IBM PC.
- IBM, which during the early years of its relationship with Microsoft could have crushed the company like a bug, had to behave as if prefrontally lobotomized from 1985 to 1995 as the gruesome OS/2 saga ground on.
- Apple had to decide to not license the Macintosh operating system, a decision that led to the company going from approximately 30 percent market share in the early 1980s to 4 percent market share by 2006.

Other events that contributed to the eventual success of Windows also encompassed the following:

- The failure of industry pioneer VisiCorp to release a successful version of VisiOn, an early graphical OS for the PC that scared Bill Gates into almost shampooing his hair.

- Apple suing Digital Research over the release of its DOS shell, GEM, shortly after the product's release. GEM was a direct Windows competitor and far more sophisticated than early releases of Windows in its look and feel (it looked and felt like a Mac). Before the Apple suit crippled the product, GEM was on the verge of achieving widespread adoption in the PC market.
- An unexpected run-up in the cost of memory chips (and temporary violation of Moore's law), which helped cripple the release of OS/2 1.0.

Now, how does one fashion a credible strategic plan that assumes your competition will agree to collectively shoot itself in the forebrain while unpredictable market forces break in such a way as to help ensure your eventual success?

The answer is that you can't. Microsoft's success with Windows, which, depending on how you count these things, ranges from $60 to $100 billion (and still counting!) is as much a result of good luck and stupidity on the part of its competition as much as any vision on the part of Microsoft. No strategic plan that anyone would take seriously could include the actual events as they unfolded over the decades. And whatever strategic plans Microsoft had for Windows in 1983 were obsolete by the product's release in 1985. And whatever plans Microsoft had in 1985 were obsolete by 1987, the year of OS/2's release. And certainly by 1990 everyone's plans for Windows were obsolete as a technically inferior but useful DOS shell swept to market supremacy over far more sophisticated and feature-rich rivals that couldn't do much.

But in the meantime, as I've already pointed out, while Microsoft's competition was engaged in various sorts of self-immolation, the company was continually executing business basics effectively. From the early 1980s through the 1990s the company entered the word processing, spreadsheet, and business presentation markets with good products that sold well and received generally favorable reviews. During this same period, Microsoft was creating a PR campaign that effectively developed a pleasing persona around Bill Gates that supported Microsoft's marketing and sales efforts. The company also continuously improved and refined their development products, releasing new IDEs, languages, and tools that were well received by developers. In 1993 the company fortuitously stumbled onto the Office concept and rode its success to even larger profits. It also figured out how to make profits during the Internet

bubble by selling products such as FrontPage. In the aggregate, all these events have contributed to Microsoft's success, and little strategic planning was involved. Microsoft simply gravitated to good opportunities, executed well (or at least better than its competitors), and reaped the rewards.

You're not convinced? OK, let's look at another seminal company in the industry, one undergoing a seemingly miraculous rebirth in high tech. Let's look at Apple, a company I had quite a bit of fun with in the first edition of *Stupidity*.

Now, before we go further, I'm going to give you a test. Let's imagine, for a few minutes, that you have gone down to the mall to visit your local Apple store in order to peruse its wares and decide whether you're going to buy a sleek, dazzling new Apple Intel-based Powerbook or save a few hundred bucks and buy a boring but decent Dell laptop. As you fight your way into the place past hordes of crazed shoppers battling to scarf up the latest iPod, a dazzling light suddenly appears from nowhere in the middle of the store's ceiling. The light grows brighter and more intense, and everyone in the place, except you, falls into a deep sleep and slumps gently to the store's floor, still clutching their iPod boxes. As you watch in amazement, the light contracts into a glowing orb that descends to the floor and coalesces into a beautiful girl. (I feel these Disney trappings most appropriate in light of Steve Job's ascension to the Disney board of directors as a result of the Pixar buyout.) This dazzling apparition is dressed in a gown of diaphanous gold filigree and wafts a wand so white it almost hurts to look at it. As you gape in amazement, the wand glows and shimmers while emitting magical sparks that seem to distort reality itself! You reach out in delight to touch this marvelous instrument, but the vision in front of you quickly yanks it away with a warning that the thing scratches like heck. Tucking the wand safely away in a silicon rubber holster, the magical lady explains that she is your Apple Fairy Godmother and that she has come to ask you to develop an enchanted strategic business plan.

You are, she explains as you listen with rapt attention, to help Good King Steve Jobs come up with a wondrous way to help Apple return to the Glory Days of the late 1970s and early 1980s, when Apple was the predominant player in the nascent microcomputer industry. It shouldn't be too difficult, she says, for someone as brave and handsome as you. And, after all, she says with a lustrous smile on her face, Apple has

exquisitely designed and colored computers on which reside the industry's slickest and most intuitive GUI, Mac OS X, version Panther, or Tiger, or KittyKat, or something. This is all running on top of a rock-solid, open source foundation called Darwin, a derivative of the widely praised FreeBSD. OS X Server, OS X's bigger, brawnier brother, is a snap to set up and maintain. And the incredible success of the iPod has put Apple's name on every consumer's tongue and in just about every music lover's pocket.

Now, what's your plan? How do you plan to succor Good King Jobs? We'll stop the book for a bit and give you some time to think through what you're going to do.

OK, time is up.

What you do, of course, is smile regretfully and explain to the hallucination in front of you that you intend to quickly recover from the slight concussion you suffered when a shopping-hardened yuppie sprinting up the aisle in pursuit of the last white 6 gig Nano accidentally hit you upside your head with a purse loaded with a PDA, cell phone, and her current fourth-generation 60 gig iPod. Shaking your head vigorously, the fairy disappears with a *POOF* and the shoppers resume their mad scrambles. Then, after browsing quickly through the software displayed on the shelves and spending some time on the store's web kiosk, you bail out of the place. You see, you're a finance guy with an accounting degree working on your CPA, and one day you plan to be a CFO somewhere. You're looking for a specialized package that can roll up budgets across different company divisions and business units and create a unified financial model of the entire company, something you really can't do with plain old Microsoft Excel. No one offers such a program for the Mac, so it will have to be the Dell.

Now, why didn't you let the magic linger a little longer? Why not take a stab at planning to put Apple back on the throne from which it once reigned microcomputing 25 years ago? After all, everyone is bored with Windows and hates its copy protection. Linux, the only possible other competitor, has all the computing charm of a diesel truck and requires a degree in computer science to install. And everything the Apple Fairy Godmother said is true, and she left out some hard revenue facts besides. In 2003, Apple's annual revenue hovered around $6 billion. In 2005, Apple sold more than 32 million iPods, and more than one billion songs were downloaded from its iTunes service by the winter of

2006. Yearly revenues from 2005 were almost $14 billion with more than a billion of that being profit.

Because such a plan is as impossible to write as was a 1983 strategic plan for Windows that possessed any credibility. In 2003, when writing the first edition of *In Search of Stupidity*, I noted that Apple had about 3 percent to 4 percent market share of new computers sold worldwide (an observation that carries over to the Mac OS, which still runs only—officially—on Apple boxes). Actually, I was generous; by the time the book went to print, Apple's share had slipped to less than 3 percent in some analyses. And today, after the iPod's stunning success, Apple's worldwide market share of PCs/operating systems worldwide is now about…3 percent to 4 percent.

It isn't as if Apple hasn't tried to change this. Since Steve Jobs returned to Apple, the company has launched several "switch to the Mac" campaigns, all of which have had little impact on the market. (Apple doesn't even pretend to try hard in the server market, despite its product's excellent performance). Apple has been able to hold onto its installed base, but little more. People seem quite content to connect their Apple iPods to their Wintel machines. Teenagers, always harbingers of new trends and fads, seem happy to rely primarily on Windows-based peer-to-peer networks to "liberate" music via the Internet and break the RIAA's heart. And many I speak to seem quite put out by iTunes's digital rights management (DRM) schemes. Apple's growth is coming from consumer electronics, not computers, and no one on this planet has ever figured out how to take a company from 4 percent market share to industry dominance in the face of an entrenched competitor determined to defend its turf. Apple came close to industry dominance in the early 1970s and 1980s, but this was before IBM woke up. And despite Microsoft's creeping development of the senescence that inevitably afflicts all megasized corporations, unless a big meteor hits Redmond and Bellevue, Apple cannot hope Steve Ballmer and Bill Gates are going to stand idly by while Apple lops off significant amounts of market share and money from Microsoft.

Does this mean Apple will eventually leave the PC business? Maybe. One possible scenario is that the company focuses on building more consumer devices, using the Mac OS as an embedded operating system to run ever more sleek and scratch-prone proprietary gadgets. Perhaps Apple eventually merges with Sony or another major consumer electronics giant

and merges their technology with the new company. Apple has already provided their Intel-based computers with an easy way to run Windows, and the company gracefully exits the market with a solution that doesn't leave its customers with the option of running only soon-to-be obsolete software. Given the pace of hardware advancement and evolution, the entire affair would take only two to three years.

Or maybe the market is changing under Microsoft, and Apple is in position to take advantage of the chaos that will ensue. The iPod's success is ushering in a new era of content where music, film, and, eventually, literature is casting off its ties to the physical. Say a permanent good-bye to liner notes and beautiful album covers (two institutions already wounded by the move to CDs). Today's new music consumer expects to take their music with them, be it on an airplane, in a car, or even from their hotel room. iPods are just way stations, disposable transmitters that facilitate the job of providing personalized content 24/7/365 to consumers. And if you want cover art with that music, well, that's what websites and screen savers are for. And isn't it nice those pretty images are also available anytime from anywhere?

In this milieu, what's needed is a beautifully designed and easy-to-use system that seamlessly manages the task of providing, creating, and managing content for both professionals and the masses, a plan that calls for a hardware platform with plenty of oomph. It's called *convergence*, and high tech has been waiting years for it to occur. For Microsoft, the problem is Windows doesn't seem suited to the task; the system is feature laden but hard to use, loaded with extrusions and encrustations that make the heads of people already defeated by the remote control ache. But anyone who has used an iPod knows Apple can build lean, elegant, easy-to-learn interfaces people like. And its computers are certainly powerful enough to handle content management and transmission. So perhaps it's Apple that dominates this new world, leaving Windows to its fate as a backroom grease monkey that does the grimy, dirty work of chugging through spreadsheets and grinding out yet more business memos. The consumer market is now where it's at, after all, with COMDEX replaced by CES as high tech's major show. And now that Steve Jobs is on the board of Disney, where obviously he plans to sit quietly in the background and provide some helpful advice to the new CEO, we can hope the video iPod and its successors will at least provide us with a steady diet of nice cartoons and the latest Pixar/Disney movies.

There are many other possible scenarios. Perhaps Microsoft buys into several key markets and stitches together a convergence solution that, although not as elegant as Apple's, has enough functionality, price advantage, and nonproprietary advantages to succeed in extending Windows into the living room. After all, who wants to bet against Microsoft and all those billions? And Microsoft has already executed such a strategy, with considerable success.

Of course, if you write enough business plans, I suppose one of them will be the right one. But this smacks of hiring a room full of chimps to sit in front of a group of terminals and hack randomly at a business plan software package in the hopes they'll crank out the next Netscape IPO. The last time this worked was during the Internet bubble, and I think you'll have to wait a few more years before you can get away with this.

Another paradox that awaits strategic plans and planners is that, paradoxically, as a company grows larger, its ability to plan strategically withers away. IBM and Microsoft are both excellent exemplars of this principal. In the early 1980s IBM ruled the mainframe world, it was equal with rivals DEC and Data General in midsized systems, and the story of the PC's success doesn't need repeating. IBM was also the largest software company in the world, with its business products in use in practically every industry on the globe. The company even introduced several desktop software titles, such as an editor, that were initially well received. IBM was in a position to buy any company it needed to help ensure its continued supremacy and indeed was at one time or another rumored or actively interested in buying Intel (in which it held a significant minority stake), MicroPro, Microsoft, Novell, Apple, and many others. Yet today IBM is out of the PC business. Microsoft dominates software. The mainframe market is still profitable, but static. Minicomputers are gone. IBM's most successful business is now in consulting, telling other businesses how to use technology that in many cases IBM no longer produces.

The reason for this is that by the late 1980s, as I point out in Chapter 6 of *Stupidity* was that IBM had become too large for anyone to coordinate its various components into a strategic "whole"; the company was simply too big to coordinate the differing agendas of its myriad numbers of divisions, business units, initiatives, alliances, channel, and so on, and so on, into anything resembling a coherent plan. At the end, IBM's strategic plan had devolved to "grow by 10 percent per year." Or 5 percent.

Or **something**. But to achieve even minimal growth, IBM was forced to turn to selling its consulting services and begin shedding different businesses and products (PCs, disk drives, printers, and so on) it could no longer manage effectively and profitably, even though other firms have been able to do just that. Although then-CEO John Akers's 1992 plan for breaking the company into smaller pieces ended when he lost his job, the effective result over time has been exactly that.

The same conundrum now faces Microsoft. As of 2006, Microsoft was launching or continuing initiatives in the following areas:

- A renewed push into the small business and personal finance market, an effort that was thwarted by the feds putting the kibosh on an earlier attempt by Microsoft to merge with Intuit
- Launch of a renewed assault on Sony and Nintendo with its Xbox system
- Announcement of an attempt to create a Microsoft MP3 player to compete with the iPod
- Further attempts by Microsoft to make Larry Ellison's life miserable with a renewed push into the enterprise database with its SQL product line
- New initiatives to displace Lotus Notes and Novell's GroupWise with Exchange
- The creation of a new document format that competes with Adobe's ubiquitous PDF
- The development of a new image package aimed at PhotoShop as well as Sparkle, a Flash competitor
- The launch of a new antivirus and spyware product that takes direct aim at market leaders Symantec and McAfee
- Initiatives into the mobile e-mail market now dominated by RIM, as well as an early stab at VOIP

The "strategic" goal behind all these initiatives? Grow by 10 percent per year. The strategic plan? Act like a giant maw and attempt to slurp up every $1 billion dollar market in reach in order to continue to fuel growth. The final result of this strategy?

See IBM.

So, if thinking "strategically" is a) often impractical and b) impossible, what's left?

Yes, the boring stuff. Executing business basics in the context of your industry's technological, financial, and competitive factors. Checklists. Spreadsheets. God help you, sometimes even meetings. There's no getting away from it.

Of course, companies and far-sighted CEOs will continue to read the latest business wonderbook and develop strategic plans based on them. Just be aware that your plan will almost certainly be obsolete immediately upon completion; that if you're spending more than 5 percent of your time on thinking deep strategic thoughts, you are almost certainly neglecting the important business of running your company; and that payroll is Thursday.

So, in the end, it's all about coming up with a good idea and then figuring out how to go out and do the basic marketing and selling blocking and tackling that leads to success. Now, I fully realize that's a truism, and it's easy for me to say. To make things worse, I'm not even prepared to tell you exactly what blocking and tackling you need to carry out to successfully market and sell your products and services. Why? For two reasons. One is that I'd have to write at least one book within a book, a detailed field manual that breaks down by categories the various components of an effective sales and marketing effort within a specific industry. I have written one such book, *The Product Marketing Handbook for Software*. At almost 700 pages with more than 2,600 to-do items listed in a suggested order of execution, it's a wonderful book for software types but not as useful for builders of PDAs.

The second reason is that over the years I've noted that the ability of companies to execute successful business programs rises and falls in correlation with the certain key characteristics:

- A corporate management structure that intently studies the history of its industry
- Company managers who have a contextual understanding of their industry's business requirements based on their analyses of the preceding
- An understanding of the company's basic "type" (in other words, sales driven, market driven, technology driven, and so on)

- A lack of age discrimination within the firm
- A temperamentally balanced management group
- An interest in providing cross-functional training to company managers

Let's analyze the value of these characteristics one by one.

You Shall Study the Past, and the Past Will Make You Less Stupid

The first and most valuable thing most companies can do to avoid acting stupidly is to encourage all employees to learn about the history of the industry in which they compete. The great thing about history (hindsight) is it is full of facts from which you can learn things, such as how to avoid positioning disasters and what to do if a PR roof falls in on you, while many strategic business books are often full of suppositions and untested conjectures. Now please, don't waste everyone's time with an attempt to wiggle out of your required reading by telling us about the "subjectivity" of history; we're all aware that people can differ about the significance of different events. If different writers and historians have different opinions about the facts, read them all, and make up your own mind from an informed viewpoint.

In the spirit of the advice just given, the following sections include my particular lists of "must" and "recommended" reading. Most of these books focus on high tech, but I've thrown in a couple of tomes from other industries to stretch your brain and provide you with some cross-cultural diversity. Feel free to criticize this lineup and add and subtract to it as you see fit. These lists are not that long, and when you are done reading these or similar books, you will have a well-rounded understanding of the forces that shape the high-tech industry, a truly invaluable asset. Both lists are in alphabetical order.

Must-Reads

These are the must-reads:

- *Apple: The Inside Story of Intrigue, Egomania, and Business Blunders* by Jim Carlton. This is a seminal history of how, where, and why Apple lost its bid for market dominance in desktop computing.
- *Big Blues: The Unmaking Of IBM* by Paul Carroll. This book is a well-written account of the critical period in the late 1980s and early 1990s when IBM lost its luster and market leadership.
- *The Dream Machine: J.C.R. Licklidder and the Revolutions that Made Computing Personal* by M. Mitchell Waldrop. This is a fascinating look at developments in the 1950s, 1960s, and 1970s that led to the rise of personal computing. Read the section on the events at Xerox's legendary PARC laboratories carefully.
- *Gates: How Microsoft's Mogul Reinvented an Industry and Made Himself the Richest Man in America* by Steve Manes and Paul Andrews. To date, this is the most comprehensive history of the early and middle years of Microsoft and Bill Gates you can read.
- *Hackers, Heroes of the Computer Revolution* by Steven Levy. This interesting book spans the world of MIT hackers to the founders of some of the first PC games companies. Of particular note is its profile of Richard Stallman, father of the Free Software and open source movements.
- *Joel on Software* by Joel Spolsky. This is a compendium of fascinating ruminations and rants on running a software business and development trends by Joel Spolsky.
- *Marketing High Technology: An Insider's View* by William H. Davidow. A bit rambling and general at times, this is nonetheless a high-tech marketing classic. Davidow was one of the authors of the Intel "Crush" campaign, a marketing program that cemented Intel's lead in the microprocessor market and relegated arch rival Motorola to also-ran status.

- *The Reckoning* by David Halberstam. This big, long book describes how the Japanese kicked the stuffing out of the American auto industry. For a classic example of how companies learn lessons only to forget them, turn to page 558 to read about how Lee Iacocca revived a Chrysler promotional program that offered purchasers of Chryslers a five-year and 50,000-mile warranty to help turn the company around. Decades later, Hyundai would relearn a lesson Chrysler forgot.
- *Selling Air* by Dan Herchenroether. This is the only book ever written that describes accurately the process of selling software in the enterprise. Both highly educational and a fun read.

Recommended Reading

These are recommended reading:

- *Beer Blast: The Inside Story of the Brewing Industry's Bizarre Battles for Your Money* by Philip Van Munching. This is an excellent look at product marketing in an industry dominated by distribution and image advertising. If you're working in a market segment dominated by many products, or which has undergone commodization, this is an invaluable guide to tactical infighting.
- *On the Firing Line: My 500 Days at Apple* by Gil Amelio. This is a very odd book by a very interesting man. Amelio's tenure at Apple was notable for the lack of progress made in halting Apple's sales and marketing slide, and reading his book it is not hard to figure out why. By his own admission, he failed to hire the right people, failed to enforce edicts against channel stuffing and stupid discounting, and eschewed any serious look at the product marketing dilemma facing the then struggling computer company. In fact, one of the striking things about this book is the lack of focus given to Apple's products and marketing. We do read a great deal about Amelio's salary negotiations, which helps explain his failure at Apple and is an instructive point to ponder in an era of silly CEO salaries.

- *Open Source: The Unauthorized White Papers* by Donald K. Rosenberg, Ph.D. This book is an excellent look at the issues and challenges surrounding the development of open source software. The book covers the history of the current general public license (GPL) under which Linux and related products are released, and it covers the different variants that have sprung up over the years. Rosenberg also discusses Microsoft's reaction to the development of Linux and the efforts the company has made to strangle this annoying infant in its crib.
- *Odyssey* by John Sculley. Long out of print, this book should be read after *Apple*. Apple's most significant CEO with the exception of Steve Jobs, John Sculley made some of the worst marketing, technical, and sales decisions ever seen in the industry. This book provides insights (many unintentional) into how he did it.
- *The Product Marketing Handbook for Software* by Merrill R. (Rick) Chapman. One of my other books, this is an extensive field manual for software marketing and sales. At almost 700 pages with more than 2,600 checklist items, it's the most comprehensive book of its kind.
- *The Second Coming of Steve Jobs* by Alan Deutschman and *iCon Steve Jobs: The Greatest Second Act in the History of Business* by Jeffrey S. Young and William L. Simon. Read together, both of these books will help you understand why in the not too distant future, many people at Disney are going to be living interesting times.
- *Once upon a Time in Computerland: The Amazing, Billion-Dollar Tale of Bill Millard* by Jonathan Littman. California, est, and ethics meet in the high-tech distribution channel. Ethics lost, but sometimes there is justice in this world. This buried classic explains how the first and greatest computer chain went to its eventual demise and the rise of the California approach to high-tech business.

~

Now That You Know, Do You Know When to Know It?

Now that you're done reading and your brains are stuffed with knowledge and insight, you're ready to tackle the problem of context, something you can't do until you're well read. In Chapter 4 of *Stupidity*, I describe the fundamental positioning error MicroPro made in the release of WordStar 2000 and its ultimate impact on the company. Over the years, other software companies, including Microsoft, Borland, Novell, Sun, and many, many others in the software business, have repeated the same mistake, with very much the same consequences. But what is a mistake in one milieu does not necessarily carry over to another industry. Context can change everything, which is why you need to study and learn before you are fit to make important decisions.

Shortly after *Stupidity*'s release, an interesting question was raised on the Joel on Software forum (www.joelonsoftware.com). A reader of ISOS asked if the iPod mini, on the verge of its release, ran the risk of running the same positioning conflict as MicroPro did with its WordStars. The astute reader pointed out that the two devices were named the same thing (iPod), did much the same thing, had disturbingly close pricing, and the new iPod mini offered considerably less storage functionality than a full-blown iPod.

The answer was…no. Why? Because in the world of high-tech hardware, small and sleek has huge appeal. Smaller items frequently sell better because they're easier to carry around, and good design reflects directly on the persona of the person buying a gizmo. The fact that the iPod mini offered the functionality it did in the size it came in was enough to separate it from its older brother in the mind of the buying public.

Failure to study and thus understand the context of the market has led software companies to frequently take the experience of hardware and misapply it to software, almost always to bad effect. Over the years, the "lite" word processors, spreadsheets, databases, suites, and graphics packages have not taken the market by storm. Why? Well, software is not carried around (at least not where it can be seen) and applications

are used in a wide variety of situations. And a product that's 20 megs in size as opposed to 60 doesn't make you look cooler or help you get dates. And finally, the cost of computer hardware continually drops relentlessly, making the gains realized by tight code irrelevant from the buyer's standpoint.

But for literally over a quarter of a century, the press has bemoaned product bloat in software. For more than a quarter of a century, different software companies have attempted to build "smaller" software products. And for more than a quarter of a century, customers have voted with their dollars almost every time for more powerful software products stuffed with every extra strip of "chrome" and every last power option available. And since software is very cheap (if you doubt this, compare the cost of buying a word processor, spreadsheet, presentation package, and database in 1986 and compare it with the cost today, and then factor in inflation), why not buy the extra dagmars[1] for your PC? Who knows when you might not need to create a slide or two? Or maybe crunch some numbers? The current market for application suite software such as Microsoft Office bears this out; although it is possible to buy just Microsoft Word or Excel, today almost no one does.

It's a factor to keep in mind as "component" computing, now being referred to as "mashups," begins to reappear on the horizon. In the mid-1990s, as object-oriented programming began to sweep through the industry, the concept of "components" appeared in software. Under this new paradigm, users would pick and choose from bundled packages of components and assemble their own spreadsheets, word processors, databases, and so on, thus building their very own personalized applications that had just the right amount of features and avoided the dreaded "bloat." As mashups (mixes of components that are accessed via the Internet) have become more popular, the press (which, because of layoffs and turnover, has as short an institutional memory as the companies they cover) has begun to beat the "bloatware" drum again. And, inevitably, some poor schnook of a software company is going to create a web-based "lite" word processor or spreadsheet or application suite that will do just as well as its floppy and CD-based predecessors.

---

[1] Dagmars were large, black, rubber protuberances found on the bumpers of 1950s cars, most notably Cadillacs. The name was created in honor of a well-endowed actress of the same era.

~

Are We…Are We Ourselves?

One of the most valuable exercises a company's managers can undertake is to arrive at a candid assessment of your company type. Doing this helps you understand many of the underlying reasons and motivations for what happens inside your company and others. This in turn will help you successfully maneuver through the shoals of corporate politics and safely pass the rocks of competitive pressure.

The problem with this exercise is 95 percent+ companies that undertake it lie to themselves. When asked, just about every CEO or member of upper management will blurt out they are a "market-driven" organization. Sometimes they are, and sometimes they're lying (usually to themselves).

There are four basic classes of firms:

The technology-driven company: The technology-driven company is controlled by the desires and direction of its development staff and is (naturally) the most common type of high-tech firm. MicroPro, of whom we've read, was a classic example of this company type, and its inherent tendencies explain why the company engaged in the final act of technical immolation I discuss in Chapter 4. The most common problem with the technology-driven company is that it builds products that satisfy its development staff, rather than providing the features and benefits the market wants. A classic example occurred early in WordStar's evolution. Over time, many users requested the ability to format text in side-by-side columns, useful for many types of writing, particularly resumes and newsletters. The development group refused to add this feature to the product and noted that over time requests by users for side-by-side columns diminished. This was absolutely true! Users wanting this feature purchased WordPerfect or Microsoft Word. (To the day the product died, it never had this feature). Novell was another example of this syndrome in action. Its key programmers successfully fought the incorporation of a GUI into NetWare long after it was clear that this was what the market wanted.

The sales-driven company: Ashton-Tate, publisher of dBase, was a classic example of a sales-driven company, myopically focused on fulfilling quarterly sales quotas. When demand for certain products weakened, it would offer product at special prices, bundle slow sellers with quick movers, offer special returns, offer stock swaps—anything in an effort to meet unrealistic quotas. At one point, the distribution system had backlogs of more than 24 months for certain products, but Ashton-Tate had satisfied its quotas, for the time being. Of course, much of this product eventually came back, and the revenue piper had to be paid, with much wailing, gnashing of teeth, and layoffs. Siebel was a more modern example of a sales-driven company; its desire to close business, making it oblivious to the desire of some of its key customers to give the CRM publisher some business back.

The market-driven company: This company is motivated by the needs and desires of its customer base. This is often the most successful of company classes because all functional groups sublimate their egos to their customers' needs. Although this is an easy philosophy to preach, it is a hard one to put into practice. But market-driven companies have their own problems. Often, they lose the will to lead. The company becomes too reactive and fearful of change, waiting for the safe road to appear while missing opportunities stemming from an aggressive but intelligent, proactive development strategy. An excellent example of this phenomenon occurred with WordPerfect. For years, WordPerfect offered toll-free phone support to buyers of its products, a practice that made customers love the company and the company love them back. WordPerfect's legion of loyal DOS users lured the company into thinking that its Windows development effort did not have to be a top company priority. As history has demonstrated, all those DOS fans were wrong.

The finance-driven company: Not many pure finance-driven companies exist in high tech, though some will argue that Dell comes close. Probably the best examples can be found outside of high tech. For instance, during the 1980s, auto-industry observers noted that General Motors was a finance-driven company. The road to upper management usually led through GM's accounting

department. Over time, this led to GM implementing many cost-saving programs that made it more economical to create cars that nobody wanted to drive, a practice that helps account for the company's continued shrinkage over the years.

Few companies are perfect examples of any one type. Most are a mix, with one element predominating, but it is important to understand which type(s) best describes your firm. This in turn will provide you with insights into the potential problems your company will likely face and its strengths and weaknesses when dealing with different sets of problems.

Never Trust (or Hire) Anyone Over 30

High tech is awash with barely disguised age discrimination. High-tech companies vigorously deny this because U.S. law forbids age discrimination; the companies, of course, are lying. In most cases, if you decide to make a career in high technology, you will be fawned over whilst in your 20s and respected until your late 30s. At age 40+, if you have not escaped into upper management, it is assumed you will be either a) rich from the money you made working for a hot start-up or b) preparing for a second career, perhaps as a fries preparation specialist at the food court of your local mall. At 50+ a perk of your job will include a shiny new shovel, with which you are expected to dig your own grave, jump in, and then drag the dirt over on top of you. If you are 60+ and are spotted in the halls of a high-technology company, it is assumed you are either a) the grandparent of an employee or b) a ghost.

The impact of foolish youth on your company's operations can range from damaging to catastrophic, depending on how unlucky you are. This was brought home to me shortly before the first release of *Stupidity* in 2003. I was contacted to possibly consult with a software company that had recently made a series of missteps in dealing with the press upon receiving less than stellar reviews. Reviews can be very important in a firm's marketing efforts, and bad notices can put a serious crimp in your product's sales. When a reviewer has slammed your product, you can go down two paths. The first path consists of the following:

1. Gritting your teeth and carefully reading the review.
2. Analyzing any mistakes the reviewer has made about your product's capabilities or misapprehensions.
3. Writing a letter of corrections to the editor of the publication and hoping it will be printed in an upcoming issue (your leverage in this regard will be greatly enhanced if you're an advertiser).
4. Contacting your customers to inform them of your efforts. Contact vehicles can consist of letters, white papers, PR releases, blogs, podcasts, TV appearances, and so on.
5. Putting in place a review management program that, you hope, educates the market and future reviewers on your product or service's abilities.
6. And, finally, fixing legitimate complaints and shortcomings in your product.

If executed properly, this type of program can have a positive impact on your future reviews and marketing, though none of this will make your current suffering go away.

The other path usually incorporates the following:

1. Screaming loudly at your employees whilst simultaneously tearing at your hair in agony and disbelief that the cretin putting this bilge to paper has achieved the miracle of somehow succeeding in putting pen to paper when it is clearly evident they don't possess the required gray matter to sustain basic autonomic functions, such as breathing.
2. Calling the editor of the offending rag and hurling threats of defenestration, physical violence, and never, ever advertising in their Codex of Evil ever again (this final threat will be taken seriously by some publications but never works when accompanied by shrieks of fury at the unfairness of it all; rather, this approach is sometimes effective when delivered by a soft velvet touch and will only have an effect on subsequent reviews).
3. Calling the reviewer at his or her home and yelling at them.

The second path, which is the one this prospective consulting customer had chosen, is never effective and usually leads to a company developing a toxic reputation amongst press, an outcome that can have

a long-term and devastating impact on your sales and marketing efforts, as I describe Ashton-Tate discovering in Chapter 5.

As part of the "getting to know you" evaluation I conduct before taking on a new client, I first reviewed the company's PR program with the firm's director of marketing, a bright, personable woman with an MBA and two years of post-college experience under her belt. As I had suspected, neither she nor anyone at the company had any experience with the software review process. In the course of the discussion, I was taken on a tour of the company's website. As she showed me one of the site's promotional pages, I noticed an image of a professional golfer—not a first-tier player but a well-known second-tier star with something of a "bad boy" reputation (at least as bad boy as a guy playing golf can get). "That's an interesting choice of an endorser," I told her. "Can you tell me how much he cost?"

"Oh, we're not paying him anything," she said in a sunny tone. "One of our web designers saw the image, liked the way it fit with our campaign, and put it up."

"Uh, you do realize you're running a liability here?" I told her. "You can't just slap up an image of a personality and not pay them or at least get their permission to use their likeness."

"I'm not worried. We're covered under fair use law."

"No, you're not," I said. "You're not a news organization. You're not reporting on anything. You're not conducting scholarly research. You're not writing books on bad golf swings. These are the criteria for using content for fair-use purposes. Sticking this fellow's picture up here doesn't fall under any of these."

She seemed unconvinced, and I let the matter drop. Later I mentioned to the company's CEO that he ought to check with his lawyer about the potential lawsuit issues. (The web page vanished a few days later.) I also mentioned that his marketing department seemed to consist entirely of 20-somethings with very little industry experience.

"I like the energy they bring to the company," he said. "And they work cheap."

"How much do you think those bad reviews you've been getting are costing you in sales and reputation?" I asked.

After this encounter (I wasn't hired by the way; the company founder thought my fees were too expensive), I decided to conduct an impromptu poll of ten small- to medium-sized high-tech companies and asked them

to estimate over the last six months how many new hires were 20+, 40+, and 50+ years old, respectively. (This was something of a guerilla poll, since the HR departments at these companies would never have cooperated with me.) The smallest company had $1 million in revenue, the largest more than $70 million. The numbers broke down as shown in Table 12-1.

Table 12-1. *Ages of New Hires*

Age Range	Sales	Marketing	Development
20+	80%	95%	80%
40+	20%	5%	10%
50+	10%	0%	0%

These numbers were striking. Based on them, if you're older than 50, the chances of being hired by a high-tech firm are almost nil (there's some hope in sales, perhaps a reflection of the value of a salesperson's track record and contact list). And even if you're just in your 40s, the numbers are almost as grim, especially in marketing. The reason for amateur behavior on the part of many software companies may be quite simple: Amateurs are running the show.

Shortly after my visit with the software company, the impact of amateurism in regards to fair-use laws came into sharp focus, courtesy of Salesforce.com. Salesforce.com was an early leader in the SaaS (formerly ASP) market, providing customer relationship management (CRM) software you accessed via your web browser. Salesforce.com was a happy green island of profitability in the sea of red ink that subsumed the early ASP market and bluff and very quotable founder and president Marc Benioff became a much-reported-on celebrity in the technology press. As Salesforce.com went public and continued its profitable ways, Benioff and the company marched off to do battle with the likes of Siebel and Oracle under the banner of "The End of Software," a direct challenge to the licensing model that has dominated the industry since its development.

Along the road to sales and IPO success, Salesforce.com also became known as a contributor to Tibetan human rights organizations and causes. Tibet and its exiled leader, the Dalai Lama, are currently the

darling of Hollywood, Richard Gere, and the left coast of the United States in general. Tibet is the type of oppressed nation Silicon Valley can get behind; the Dalai Lama is Disney cute, and Tibetan Buddhism's belief in reincarnation is magical and mystical in a way that generates a delicious shiver in the souls of California chardonnay drinkers and Shirley MacLaine acolytes everywhere. Even better, the country's remoteness, mainland China's possession of nukes, and the geological fact that Tibet suffers from a complete lack of oil, means there's virtually no chance the United States will actually **do** anything useful to relieve the sufferings of the oppressed Tibetan people (such as send in the Marines to shoot the oppressors à la Iraq).

Flush from success and a belief in the ultimate holiness of its cause, in 2003 the company was one of several major donators to a September 5th event in San Francisco hosted by the American Himalayan Foundation at which no less an august presence than his holiness, the Dalai Lama, was scheduled to speak. Plugging its role of event donor, in August the company created 500 posters that it sent to local companies and members of the press. The poster featured a photograph of a meditative Dalai Lama (the Pope and the Archbishop of Canterbury presumably being busy that day) beneath a caption that read "There is no software on the path to enlightenment." Beneath the photograph was emblazoned Salesforce.com's "No Software" logo and the subhead, "Salesforce.com celebrates 100,000 enlightened Salesforce.com subscribers."

The Dalai Lama, who to the best of everyone's knowledge, does not currently use hosted sales automation software, was reportedly not amused by a campaign that used his likeness to give an implied commercial endorsement and made his displeasure known. At first, Salesforce.com attempted to blow off the criticism by pointing out its support for all things Tibetan; however, the outcry against the company's commercial crassness refused to die down. Then rumors reached Salesforce.com that the Dalai Lama had retired to a mountain retreat and was praying that the company's marketing and PR group be reincarnated as earthworms. Upon this, Salesforce.com promptly capitulated, the posters were recalled and destroyed (the few that survive are now prized collector items), company president Marc Benioff publicly apologized, and presumably Salesforce.com's marketing personnel will enter the next cycle of existence in the form of creatures whose principal diet does not consist of forest refuse.

~

The Best Generals Hire the Best Generals

In the foreword to *Stupidity*, Joel Spolsky states his belief that high-tech companies can't succeed unless there's a programmer (and we'll assume he's also partial to hardware engineers) at the head of a company. It's a natural assumption; after all, Joel is a programmer, and coders and engineers do tend to be the loci of new ideas and products for high tech. But it's a supposition that's easy to argue with. Ray Noorda was the man most directly responsible for the early success of Novell, and he wasn't a programmer. Steve Jobs was the man most responsible for the rise of Apple and the creation of the Macintosh OS. Jobs was never a programmer and only nominally an engineer. It was Steve Wozniak who did all the significant work in this respect during Apple's early days. Charles Wang, founder of Computer Associates (now CA), may have done some nominal coding, but his main interest was always business. Charles Tate of Ashton-Tate never programmed for a living nor did Scott Cook, founder of Intuit.

Of course, Joel advocates can point to some significant examples that prove his case. Bill Gates was certainly a programmer. Mitch Kapor, founder of Lotus, was a developer as well. John Warnock of Adobe was an engineer and onetime chief scientist at the legendary PARC lab.

But then you look at the history of developers in relation to the companies they either founded or worked for, and the picture again becomes decidedly mixed. The SuperSet helped damage Novell. The WordStar programming team drove the final stake through MicroPro's heart. Dan Bricklin and Bob Frankston, developers of the first blockbuster desktop application, VisiCalc, engaged in a foolish and ruinous fight with the publisher of the product, VisiCorp, that destroyed the spreadsheet cash cow for both companies. Apple's developers made a 20-year career of running amok from time to time. It's probably fair to state that programmers and developers help kill and hurt as many companies as they start.

Many historians believe the principal reason that Napoleon was beaten at Waterloo had little to do with the specific tactics used on the

field of battle but by the absence of three men from the campaign: Marshalls Berthier, Lannes, and Davout. Berthier was Napoleon's chief of staff, a master of interpreting Napoleon's wishes and transmitting complex orders in clear, simple terms. After Napoleon's first exile, he switched sides and never switched back. During the battle, his replacement, Marshall Soult, proved unable to provide the same clarity of communications achieved by Berthier, and the French suffered mightily because of it. Lannes, a brilliant fighter who was willing to talk back to the Corsican when he felt he was wrong, couldn't make the big event; he was dead, killed at the battle of Aspern-Essling. Davout, Napoleon's strategic equal and a man who would have crushed the Prussians at Wavre where Marshall Grouchy failed and thus made it likely Napoleon would win at Waterloo, declined to show up for the battle, tired of Napoleon and his endless wars. The marshals who did show up to support the emperor were for the most part brave men and competent, but they were also his managerial second tier.

A well-run company follows the example of Napoleon at his best (I'll pause a moment for all you current and nascent CEOs and future members of upper management to enjoy the frisson this analogy is generating, but please remember we're simply resorting to an analogy here; business is not war) and develops a well-rounded, high-quality management team. An interesting aspect of many of the most successful high-tech companies is that they seem, at least for a time, to follow a "binary star" system, with two people in essence sharing the CEO's job—one person focusing on the technical side of the company and the other on key business issues. Notable examples of this approach include Gates/Ballmer, Warnock/Geschke, Jobs/Wozniak, and Cook/Proulx.

Regardless of how or whether the top job is split up, you're also going to need to establish and manage a management group with a diverse psychological profile. Please note I'm not discussing specific skill sets, such as the ability to write Java code or do basic bookkeeping; it's a given that if you are to build and sell products and services, that you have at least some initial competence in doing so.

What I mean by "diverse" is the mental landscape of your team and the way they choose to use their abilities and ambition. In many companies, the founder and CEO tend to create an upper management team

that is a clone of themselves. Each member of the team, when they look around, tends to see a somewhat distorted image of themselves that smiles back in approval and affirmation. In such an environment, a company's upper management functions in almost cult-like fashion—self-referential, self-absorbed, and increasingly cut off from both the rest of the company and the market.

Another extreme is the management theory that the executive suite functions as an analog to the Roman Colosseum in its heyday. Periodically, members of the management team are expected to check in for fights to the death with each other held under the watchful eye of the company founder or CEO. The rationale normally offered for this practice is Darwinian in origin; by encouraging this type of ongoing fratricide, you are supposedly building a tougher, better business executive. Apparently, no one has ever considered the possibility that what you are evolving toward is a manager better adapted toward killing his peers than the competition.

The best management systems I've worked with or observed avoid both of the extremes described. Instead, they seek to blend a psychologically diverse group into an effective group. The best teams at minimum always seem to possess the following:

- Someone with the ability to successfully communicate the CEO's idea and business goals to the rest of the company and its managers. The "Berthier."
- Someone equal in business skills and abilities to the CEO, an individual who can step in and run the company in the event something renders the company's leader hors de combat; this person is also willing to step back into their assigned role and take orders. The "Davout."
- Someone unafraid to challenge upper management's assumptions and beliefs when warranted. The "Lannes."
- Someone with a strong understanding of the company's logistical needs and capabilities (this can include the firm's finances). The "Wellington." (Napoleon and his managerial group never fully mastered the importance of logistics, as the 1812 debacle in Russia demonstrated.)

~

Now That You Know, What Do the Rest of You Know As Well?

My last piece of good advice focuses around the issue of "siloization," the tendency of key functional groups in high-tech companies (development, marketing, sales, Q&A, and so on) to be ignorant of the value and contribution of other groups. Of course, a great deal of lip service is always being served in this regard. For example, I've actually never had a technologically driven company tell me they think their sales group is worthless; they simply act like it.

Of course, if you've done your reading and learned your history lessons, your managers and employees should be intellectually inoculated against this type of foolishness, but it's one thing to know something and another to feel it. My solution? Encourage members of your company to compete in business simulation games. Simulators from companies such as Forio and the famous Marketplace simulator that is sold by several companies allow your managers to game price wars, brand management, sales and marketing campaigns, distribution strategies, and more. Simulators are a marvelous opportunity to provide your employees with a chance opportunity to test ideas and concepts in cyberspace before they approach your bottom line.

Simulators are also excellent tools for encouraging teaching teamwork and collaboration if members of different groups from your company play in teams. Just about every person who plays in a simulator ends up learning the true value of cash flow and finances in a company's operations. I've played Marketplace; the game has a loan shark if you run out of cash, and you really don't want to meet him.