THE JOURNAL OF
**PRODUCT
INNOVATION
MANAGEMENT**

Book Reviews

Book Review Editor: Preston G. Smith, CMC

Books reviewed in this issue:

- *The Innovator's Solution: Creating and Sustaining Successful Growth*
- *Seeing What's Next: Using the Theories of Innovation to Predict Industry Change*
- *The Second Century: Reconnecting Customer and Value Chain through Build-to-Order: Moving beyond Mass and Lean Production in the Auto Industry*
- *Balancing Agility and Discipline: A Guide for the Perplexed*
- *Organizing Business Knowledge: The MIT Process Handbook*
- *Leading Pharmaceutical Innovation: Trends and Drivers for Growth in the Pharmaceutical Industry*

The Innovator's Solution: Creating and Sustaining Successful Growth, by Clayton M. Christensen and Michael E. Raynor. Boston: Harvard Business School Press, 2003. 304 + x pages. US$29.95.

Seeing What's Next: Using the Theories of Innovation to Predict Industry Change, by Clayton M. Christensen, Scott D. Anthony, and Erik A. Roth. Boston: Harvard Business School Press, 2004. 312 + xi pages. US$29.95.

For those who have read *The Innovator's Dilemma* (Christensen, 1997) or those who have heard Clayton Christensen explain its theories, these two books serve to extend and to apply that thinking, making it far more useful to those involved in product and business innovation. For those new to this thinking, *The Innovator's Solution* provides a great overview of a well-researched and compelling theory that explains why launching new growth businesses is so hard while showing would-be innovators how to make the process of creating new growth businesses more predict-

able. *Seeing What's Next* takes the theory a step further and shows how to use it to analyze how innovation will change an industry.

If you are looking for a book on how to generate new ideas or how to manage the "fuzzy front end" at the project level, you will not find these very useful. These two books will be most useful to those involved in business innovation and strategy, although there is plenty that can be applied to innovation management as well as strategy for new products, product platforms, and product lines. Those involved in portfolio management or portfolio decision-making also should find the concepts useful. It might be helpful to think of these as addressing macroinnovation theory instead of microinnovation theory (like macro- and microeconomics). If you are hoping that reading these books will help you identify new opportunities that can be realized in 6 to 18 months, you also are likely to be disappointed. Many of the examples show that it can take a long time to realize fully the power of a disruption.

As indicated in Karen Graziano's January 1998 *JPIM* review of *The Innovator's Dilemma*, Christensen writes from the basis of a great deal of research, but the original theory relied perhaps too heavily on a lengthy example from the disk drive industry. In both of these new books, the examples are far more diverse, making the theory more meaningful and robust. Readers will find these two books to be well organized, with a deliberate "tell them what you're going to tell them, then tell them, then tell them what you told them" kind of style. The writing includes excellent comments and references, important because the authors synthesize contributions from many others along the way (such as Dorothy Leonard, Charlie Fine, and Geoffrey Moore). Noteworthy of both books are the various "litmus tests" offered up to help guide analysis and validation of disruption. I also

liked the way these books try to establish clear definitions of terms. In particular, new product development (NPD) professionals would be served well to adopt the definitions of sustaining and disruptive and to avoid the mistake of calling all radical innovation disruptive.

If you are in a dilemma about which of the trilogy to buy, I found *The Innovator's Solution* to be the most valuable of the three and would recommend it as the starting point for NPD professionals new to this thinking or wanting to understand it better. Nonetheless, I found the additional concepts, summaries, and extensions of the thinking in *Seeing What's Next* to be useful. I have highlighted these concepts in the comments following, where I outline each book (beginning with *The Innovator's Solution*) with brief comments on chapters and themes I found most valuable.

Rather than begin *The Innovator's Solution* with repetitive rehashing of the original theory, it starts with a provocative look at how business theory should be developed if it is going to lead to predictability. I found those insights the most valuable part of chapter 1. Chapter 2 quickly reintroduces the original theory using a different industry (minimills in the steel industry) and goes on to extend that theory in an important way. My biggest problem with the original book was the way market disruption examples seemed to be portrayed as technology disruptions. For me, it made it more difficult to apply the thinking. I was pleased to see that Christensen had addressed this problem head on, adding a new dimension to his core theory, now differentiating between low-end disruption and new-market disruption.

From there on, the book is organized to answer a series of questions about how to beat competitors, what products to develop, what customers to target, what to insource versus outsource, how to maintain competitive advantage, and how to organize. Chapter 3 on products highlighted segmentation, and I valued the insights gained by taking a circumstance-based view that requires looking at the job the customer hired the product to do. This is essential thinking to establish a disruptive foothold in a marketplace. In the next chapter the key learning is that most successful disruptions are accompanied by business model innovation that includes all aspects of the value network, an important thread developed in later chapters. Chapter 5 went on to explain that product platforms and modularity are not always a good idea. Knowing when to take an integrated approach

and when to go modular is an important addition to the current platform theory offered by McGrath (2001) and Meyer and Lehnerd (1997). Chapter 8 builds on this thinking by laying out a view of the strategy process that includes both deliberate and emergent strategy. I particularly liked how portfolio management was characterized as strategy rationalization.

I found this book to have very few drawbacks. The chapter on organization seemed weak relative to the others. For example, I found that the definitions of lightweight and heavyweight teams seemed to confound the issue of functional versus cross-functional. Still, the insights provided on leadership and the chief executive officer's (CEO) role were quite good. Finally, I would have preferred more insight into the linkages between this theory and that on technology, market, and product lifecycles. There seemed to be little mention of it, though those lifecycle forces would certainly be at work.

Seeing What's Next is clearly the more macro of the two, focusing much more on predicting industry change. That focus, while valuable from a strategy point of view, makes it less valuable to those focused on new product innovation. It is organized into roughly two big chunks with helpful "bookends" around them. One chunk is chapters 1 through 4, which present a theory to predict industry change by identifying signals, evaluating competitive battles, looking at strategic choices, and understanding regulatory drivers. The second chunk comprises "future scoping" several industries, including telecom, health care, semiconductors, aviation, and education. These two chunks are preceded by a simple recap of the key aspects of the previous book and are followed by a handy glossary of key concepts.

For me, the two biggest contributions of *Seeing What's Next* were the innovation motivation/ability framework and the chapter on competitor analysis. The motivation/ability framework shows how different combinations of motivation and ability have a great bearing on innovation. I felt that in addition to explaining innovation at a macro level, they were meaningful at the micro level (for instance, explaining perhaps why companies like 3M can drive more innovation by doing things to increase both motivation and ability.) The competitor analysis chapter was highly insightful because it seemed to synthesize many of the key theories of the two books and to apply them to analyzing competitors. As a learning tool for the reader, this helped me to internalize the concepts—precisely what the authors intended by writing

this last book from the perspective of applying the thinking looking forward.

The drawbacks of this book are twofold. Its macro focus makes it less useful for product development, and it does not really add that much more to the thinking provided in *The Innovator's Solution*. Still, the concepts it did add, along with the thought-jogging application from a forward-looking perspective and the simple recap of all the critical concepts, have great value if it means that the concepts will be understood and applied.

From my perspective, *The Innovator's Solution* should be required reading for anyone responsible for business or product innovation and/or strategy. It is well worth the investment and should find itself well used over time.

References

Christensen, Clayton M. (1997). *The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail*. Boston: Harvard Business School Press.

McGrath, Michael E. (2001). *Product Strategy for High Technology Companies: Accelerating Your Business to Web Speed*. New York: McGraw-Hill.

Meyer, Marc M. and Lehnerd, Alvin P. (1997). *The Power of Product Platforms: Creating and Sustaining Robust Corporations*. New York: The Free Press.

Mark J. Deck, NPDP
PRTM Management Consultants

The Second Century: Reconnecting Customer and Value Chain through Build-to-Order: Moving beyond Mass and Lean Production in the Auto Industry, by Matthias Holweg and Frits K. Pil. Cambridge, MA: MIT Press, 2004. 232 pages. US$ 35.00.

To the casual observer, *The Second Century* appears to be a scholarly study confined to cyclical supply chain issues of the auto industry. Further examination reveals complex lessons in system dynamics. *The Second Century* was not written for individuals who are looking for a quick fix; rather, it is intended for strategic planners entrusted with optimizing interdependent, dynamic systems. "Those who believe that volume and efficiency drive profitability have only to look at the industry's current state. Factory efficiency is at an all-time high while industry-wide profitability is poor at best" (p. 67). This summary is typical of the conclusions presented in *The Second Century*.

The Second Century does not address automotive design, portfolio, or platform issues. It does not suggest marketing strategies or tactics. It is not written for project management professionals. It is not a sequel to the 1990s book, *The Machine that Changed the World* (Womack, 1990). Instead, it addresses systemic views that "may require sacrificing some local optimization to enhance system-wide performance" (p. 3). Thus, it addresses topics such as outsourcing, colocation, and "new entrants" in the context of a system having legacy issues that span a century.

The Second Century contrasts two automotive industry supply chain strategies: forecast-based business models and build-to-order business models. A *forecast-based environment* uses annualized historic data and projections to create a production plan to build cars with a particular configuration such as a dark green, four-door car with air conditioning and certain other options. It passes thousands of parameters to multiple first-tier and second-tier suppliers to have the required parts available for manufacturing at a specific factory at a specific time. Such a system is rather rigid, but it does produce these dark green cars with great efficiency. A forecast-based system can be characterized as a push system. If the weekly inventory of dark green cars is too great, manufacturers use a combination of incentives to dispose of the excess inventory. To offset the reduced profit, suppliers receive requests to produce future components at lower prices.

In a *build-to-order business model*, an order triggers all of the parts for the dark green car and schedules a build date when an actual customer requests that exact configuration. This pull approach demands a supply chain that is very flexible. To deliver a specific car to a specific customer in an acceptable amount of time at an acceptable cost, the system (including ordering, logistics, and transportation) must work synergistically. Influences of both strategies are evident at all major car companies. In general, higher priced cars have a greater percentage of deliveries from build-to-order processes.

Those familiar with the "Beer Game" [a logistics game and management simulator developed at MIT in the early 1960s and popularized in business books (Senge, 1990)] or the "bull whip effect" will predict that disparities will arise between supply and demand when there are multiple feedback loops, nonlinearities, and lags in the system. The dealer will not be too surprised when a potential customer arrives requesting a dark green, four-door car with air conditioning

that this configuration is not available immediately. This configuration either is sold out (because it is a very popular configuration this month) or never was ordered (because it was expected to be an unpopular configuration this year).

Although it is a common perception that the disparate systems of dealers, manufacturers, and first- and second-tier suppliers can be synchronized using e-commerce exchanges, the authors state that "although we believe that [information technology] IT in theory can certainly enable a build-to-order system at some point, it is more of an inhibitor right now" (p. 106). Legacy issues that have cultural and technological origins prohibit quick-fix solutions.

Matthias Holweg and Fritz Pil strongly imply that forecast-based business models have produced a large inventory (approximately 60 days) that is a suboptimal mix (p. 85). To reduce inventory, General Motors provided an average financial incentive of US$3855 per car in October 2002 (p. 90). This reduced the profit from vehicles sold and introduced skewed data into the production forecast, further exacerbating the problem.

Once they specify a configuration and the subassemblies are available, most manufacturers can build a car in about one day. "Because factory efficiency has been the target of so many improvement efforts, it would seem that very little change is required to provide the needed flexibility. But the value grid is complex, and the secret of flexible factories, as it turns out, rests not in robots and fancy computer systems but in how the factory manages and organizes its work" (p. 127). The authors prefer the phrase "value grid" instead of the more common phrase "value chain," which implies sequential relationships.

In chapters 9–15, Holweg and Pil explore the three dimensions of responsiveness: process, product, and volume. These are "prerequisites to implementing any build-to-order system" (p. 211). They cite Dell Computer and Alcoa as companies that are paragons of responsiveness. These chapters suggest exploring specific topics to improve responsiveness. These include ordering patterns, slotting demand, visibility, online sales, workforce organization, compensation, capacity use, and automation.

Because corporate success requires a customer focus, value migrates to companies that satisfy changing customer priorities quickly. A key insight for innovators and developers is that "companies seem to have forgotten that profitability comes, not from optimizing cost, but from building the right product at the right time" (p. 2).

References

Senge, Peter M. (1990). *The Fifth Discipline: The Art and Practice of the Learning Organization*. New York: Doubleday Books.

Womack, James P., Jones, Daniel T. and Roos, Daniel (1990). *The Machine that Changed the World*. New York: Rawson Associates.

Mark A. Hart
OpLaunch

Balancing Agility and Discipline: A Guide for the Perplexed, by Barry Boehm and Richard Turner. Boston: Addison-Wesley, 2004. 266 + xxvii pages. US$29.99.

A subtle conflict is smoldering in the product development community. On one side are those who favor relatively formal phased methods, such as Stage-Gate® or PACE®. On the other are those who prefer dedicated colocated teams, just-in-time planning, minimal documentation, and similar techniques to operate faster and more flexibly but without a safety net.

This conflict has broken into open flames in the software development community, and I believe that we can learn from their experience. *Balancing Agility and Discipline* is an excellent introduction to the conflict. More important, it provides middle-ground solutions that tap the strengths of both approaches. Consequently, this is a book for those who are interested in product development methodologies, who want to align their process closer to their objectives, who wish to improve predictability or consistency, or who need to shed excess baggage. In case you are concerned, you do not have to be a computer geek to understand it.

Barry Boehm is a very senior and highly respected member of the software development community, and Richard Turner is a professor of engineering management and an author of a leading software methodology. The cover lists three forewords: one by a charismatic leader of the agile movement, one by the U.S. government official responsible for the life-critical software running the air traffic control system, and one by a self-proclaimed veteran of three software methodology wars. This sets the tone of the book: viewpoints from the agile perspective, the disciplined perspective, and the middle ground, respectively.

First, let me offer some background. Prior to about 1990, most software development applied a sequential phased model, called waterfall, similar to the phased methods usually employed in product development today. Since then, the advent of object-oriented technology has prompted a shift from a phased to an iterative style in which each of several iterations produces a *working* product (although normally not a salable one in early iterations). Iterative development, in turn, has spawned the possibility of agile development, although the mainline in software development today is what the authors call plan-driven development. The goal of agile development is achieving rapid customer value and responding to change, whereas the goal of plan-driven methods is predictability, stability, and high assurance. Although most of the warriors have taken sides, Boehm and Turner show us that each approach has its strengths and limitations under certain circumstances.

The Agile Manifesto (www.agilemanifesto.org) captures the distinction crisply: "... We [agilists] have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan.

That is, while there is value in the items on the right, we value the items on the left more" (p. 195).

Chapter 1 introduces the plan-driven and agile approaches, providing some key characteristics and examples of each. For instance, plan-driven approaches emphasize verification (work products reflect the original requirements) and validation (final product satisfies its intended mission), while agile ones emphasize simple designs based on YAGNI (you aren't going to need it), a concept of not designing anything that is not needed currently—because, with change, it may never be needed. Because product developers following a stages-and-gates approach would be familiar with the plan-driven style, I will provide examples of the perhaps more foreign agile approach.

Arguably, the most-discussed agile methodology is eXtreme Programming (XP) (Beck, 2000). Its basis is four fundamental values: communication, simplicity, feedback, and courage, and it encompasses 12 carefully selected and mutually reinforcing practices. Thus, the purists would say that you must employ them as a whole and not tamper with them, but Boehm and Turner's objective is to adapt their essence

to other situations. I have already mentioned one practice—simple design and its associated YAGNI. Another is pair programming, in which two programmers sit at one computer with one keyboard and mouse, working together. This practice arose by observing the benefits of colocation and early code inspection, combining them, and, as the XPers would say, "turning the knobs up to 10." Similarly, they noticed the feedback value of early testing, so they turned the knob up to 10 and established the practice of writing a module's test before writing its associated code.

Chapter 2 looks at the two approaches from another angle by selecting key project characteristics, such as project goals, customer relations, and product requirements (specifications). The authors discuss the strengths and weaknesses of agile and plan-driven approaches with respect to each of these characteristics. For example, close customer contact is an advantage of agile, but it can hurt if your selected customer is not representative or is flighty. Success depends on having CRACK (collaborative, representative, authorized, committed, and knowledgeable) customers to guide development. The caliber of your personnel is critical to many methodologies, both plan-driven and agile. As the authors put it, "... The unavoidable statistic [is] that 49.999 percent of the world's software developers are below average" (p. 47). Here, the book draws upon the work of agilest Alistair Cockburn (2002) to establish five levels between −1 (may have the technical skills, but unwilling to collaborate or follow shared methods) and 3 (able to revise a method—break its rules—to fit an unprecedented situation). In summary, the authors establish the "home grounds" of both agile and plan-driven approaches and formulate five critical factors that distinguish them: project size, product criticality, dynamism of the environment, personnel caliber required, and cultural adaptability. A very useful graph of these five factors on calibrated scales helps to map methodologies to the needs of a given project.

Chapter 3 captures the feeling of each approach by watching a day in the life of a project pursued by a plan-driven methodology and by an agile one. The authors also show variations for a crisis day. Their conclusion is this: "The bottom line, however, is that getting the people factors right is much more important in the end than technology or method considerations" (p. 81). Chapter 4 then moves into hybrid approaches with a case study of scaling up an agile methodology by adding some planning and another

case study of applying agility to streamline a plan-driven project. Conclusions: scaling up requires a relatively scarce Cockburn Level 3 process master, but "tailoring down" is unpalatable inherently.

This brings us to a resolution of this dilemma and the creative heart of the book, chapter 5, which proposes a means of balancing agility and plan-driven approaches to suit the needs of a given project. The balancing depends on looking at the risks of swinging too far in either the plan-driven or the agile directions. For instance, consider YAGNI. In a predictable environment, YAGNI leads to rework to add the ignored feature to the design later, but in a highly fluid environment, incorporating features on speculation means being wrong much of the time, which leads not only to wasted design time but also to unnecessarily complex code that is more difficult to maintain. Thus, the ideal amount of YAGNI to apply depends on the dynamism in the environment (recall this critical factor aforementioned). Boehm and Turner make this balancing easier by applying a standard list of about a dozen generic risks divided into the categories of environment, agile (one of which is YAGNI), and plan-driven risks. To illustrate the technique, they apply it to three projects: an agile-biased one, one in between, and a plan-driven-biased one.

The last third of the book comprises five useful appendices, especially for "outsiders," such as product developers. For instance, Appendix A offers short summaries of 13 contemporary software development methodologies, including XP, spanning a range from agile to plan-driven, and Appendix B covers the Agile Manifesto. Appendix E provides the empirical evidence and business effectiveness connected with many of the topics addressed. I suggest that you familiarize yourself with the appendices first so that you can refer to them while reading the book.

A nice feature of the book is thumbnails of each paragraph printed in the margin, which facilitates shifting between reading and scanning. The book's main shortcoming for product developers is that it uses many programming terms that would be obvious to the core audience but are undefined or do not appear in the index. Examples are COTS, UML, timeboxing, GUI, spiral model, and KSLOC. Fortunately, few of these are necessary to understanding the concepts, but a glossary would overcome an unsettling feeling for us product developers.

I highly recommend studying this book to assess whether your development process is where it should be in our dynamic world.

References

Beck, Kent (2000). *Extreme Programming Explained*. Boston: Addison-Wesley.

Cockburn, Alistair (2002). *Agile Software Development*. Boston: Addison-Wesley.

Preston G. Smith, CMC
New Product Dynamics

Organizing Business Knowledge: The MIT Process Handbook, edited by Thomas W. Malone, Kevin Crowston, and George A. Herman. Cambridge, MA: MIT Press, 2003. 619 + xiii pages. US$55.00.

The *MIT Process Handbook* presents a systematic method of organizing and sharing business knowledge, in the broadest sense, proposes a set of fairly fundamental concepts to direct the analytical methodology, and provides a classification framework for organizing knowledge.

This book documents the considerable work of 40 researchers and industrial sponsors and includes consolidated references and a detailed index. The *MIT Process Handbook* project started in 1991 and has resulted in the development of an extensive, publicly available online knowledge base that includes a set of software tools to maintain and access this knowledge base.

The book is primarily of interest to academic researchers interested in knowledge creation and maintenance, but business educators, software developers, computer scientists, and consultants interested in best practices also will find the work useful. However, for the NPD professional, it is somewhat intimidating in its size and scope

Essentially, the work demonstrates an approach for developing a "... comprehensive framework for organizing large amounts of useful knowledge about business" (p. 221). The authors' focus was on creating three primary kinds of "entries:"

- generic models of typical business activities (e.g., buying and selling),
- specific case examples of interesting things particular companies have done, and
- frameworks for classifying business knowledge.

The totality of the entries (5,928 as of July 2002) represents a repository of knowledge. Such a repository clearly must be organized properly to be accessed by normal search methods; a sample screen image of

an entry is included to illustrate how they accomplished this (p. 223).

A very useful section (section 8.7) examines specific business processes, which are

- Process Classification Framework, by the International Benchmark Clearinghouse;
- Supply Chain Operations Reference, by The Supply Chain Council;
- Lean Enterprise Manufacturing Model, by the Lean Aircraft Initiative consortium, led by MIT;
- European Foundation for Quality Management EFQM model; and
- Xerox Management Model.

In addition, they extract several models from marketing and product design books.

Readers will find many of the process diagrams (innovation and resource flow, for instance) and dependency and simplification diagrams in chapter 9, among others in the chapter, to be useful for comparing against one's own processes. For example, the "Reengineer business processes" chart (p. 276) identifies activities and resources required to perform this business process, in such an easily understood way that one could determine quickly if their reengineering methodology is comparable.

Later chapters are devoted to integrating software components, describing and redesigning processes, inventing new business processes using a process repository, generating new business process ideas, designing robust business processes, and managing process knowledge. However, most examples are related to service, purchasing, and human resource process areas. Chapter 19 presents the Product Workbench, an interesting means of mass customization of production processes, and the example used involves a hypothetical new financial product created by an account manager at a bank.

The Appendix discusses enabling technology, specifically the process interchange format (PIF), which is an outgrowth of *The MIT Process Handbook* project. This technology is a computer methodology to allow the sharing of process descriptions through a common descriptive format. The goal is to support maximum sharing of process descriptions across heterogeneous process representations.

Although it is limited in specific examples of product development and management, this book is intellectually challenging and advances several areas of process examination. *The MIT Process Handbook* should be viewed as a resource showing that a useful business knowledge base is feasible and can be expanded in terms of sheer volume, examples, and scope.

Finally, readers have the opportunity to contribute to the ongoing project through the Open Process Handbook Initiative on the project website (http://ccs.mit.edu/ph/). In particular, NPD professionals could suggest a content-user group for NPD practices and knowledge, which can be requested through the user group administrator.

Steve Guerin
Axiomatic Research Corporation

Leading Pharmaceutical Innovation: Trends and Drivers for Growth in the Pharmaceutical Industry, by Oliver Gassman, Gerrit Reepmeyer, and Maximilian von Zedtwitz. Heidelberg: Springer-Verlag, 2004. 178 + xi pages. US$79.95.

This book examines the international pharmaceutical industry with special emphasis on the current and emerging research methodologies now being employed for drug discovery. While the geographic frame of reference is implicitly international (i.e., North America, Europe, and Japan), the case studies and extended illustrations are drawn from the experience of Novartis and Roche, two Swiss-based firms. This should not be too surprising, however, given the Swiss affiliations and/or education of all three authors. On the other hand, given the important role played by these firms in what is becoming a truly global industry, this perspective is both refreshingly different and highly relevant to any study of the pharmaceutical business.

Leading Pharmaceutical Innovation is essentially a short summation of industry views on research and development and technological possibilities in this regard. The major chapters include a pair on innovation as the key to pharmaceutical success and the special case of Switzerland; three on pharmaceutical challenges (science and technology, pipeline management, and outsourcing and internationalization); and a pair of concluding chapters on "management answers" and the usual "future directions and trends."

What distinguishes this book from others on the same subject is its emphasis on empirical findings (53 figures, 20 tables) and an exceptionally good, if short, discussion of various research technologies such as bioinformatics, cloning, and combinatorial chemistry

to name but three. The six-page glossary of industry terms, when used with its decent index, is a particular joy.

However, this short book does not address several important topics, including most if not all market-centered forces for change and innovation in this industry. For example, the authors do not address the pressures faced by the basic price-discrimination model of this industry in a world of government-specified formularies and centralized price negotiations. Nor do they mention servicing needs in Africa and other hard-pressed regions or servicing the needs of the poor and aged in wealthier nations. There is no mention of the relative absence of comparative studies of product efficacy or the rise of government research interest in this subject. Likewise, this book treats generics as an industry problem and ignores their implications for health care management and economics. In short, this is a book about technology, not about marketing or economics.

Despite these criticisms, inevitable in a book of this length with such a tight focus, I still recommend this offering to prospective managers and researchers in this industry and to others who might be interested in the future directions pharmaceuticals might take.

Robert R. Rothberg
Rutgers Business School