

Many executives will have to learn what computer executives have long known.

MANAGING IN AN AGE OF MODULARITY

by Carliss Y. Baldwin and Kim B. Clark

In the nineteenth century, railroads fundamentally altered the competitive landscape of business. By providing fast and cheap transportation, they forced previously protected regional companies into battles with distant rivals. The railroad companies also devised management practices to deal with their own complexity and high fixed costs that deeply influenced the second wave of industrialization at the turn of the century.

Today the computer industry is in a similar leading position. Not only have computer companies transformed a wide range of markets by introducing cheap and fast information processing, but they have also led the way toward a new industry structure that makes the best use of these processing abilities. At the heart of their remarkable advance is modularity – building a complex product or process from smaller subsystems that can be designed independently yet function together as a whole. Through the widespread adoption of modular designs, the computer industry has dramatically increased its rate of innovation. Indeed, it is modularity, more than speedy processing and communication or any other technology, that is responsible for the heightened pace of change that managers in the computer industry now face. And strategies based on modularity are the best way to deal with that change.

Many industries have long had a degree of modularity in their production processes. But a growing number of them are now poised to extend modularity to the design stage. Although they may have difficulty taking modularity as far as the computer industry has, managers in many industries stand to learn much about ways to employ this new approach from the experiences of their counterparts in computers.

A Solution to Growing Complexity

The popular and business presses have made much of the awesome power of computer technology. Storage capacities and processing speeds have skyrocketed while costs have remained the same or have fallen. These improvements have depended on enormous growth in the complexity of the product. The modern computer is a bewildering array of elements working in concert, evolving rapidly in precise and elaborate ways.

Carliss Y. Baldwin is the William L. White Professor of Business Administration at the Harvard Business School in Boston, Massachusetts. **Kim B. Clark** is the Harry E. Figgie, Jr., Professor of Business Administration and Dean of the Faculty of Harvard Business School. They are coauthors of *Design Rules: The Power of Modularity* (the first of two volumes), to be published by the MIT Press in 1998.

Modularity has enabled companies to handle this increasingly complex technology. By breaking up a product into subsystems, or *modules*, designers, producers, and users have gained enormous flexibility. Different companies can take responsibility for separate modules and be confident that a reliable product will arise from their collective efforts.

The first modular computer, the System/360, which IBM announced in 1964, effectively illustrates this approach. The designs of previous models from IBM and other main-frame manufacturers were unique; each had its own operating system, processor, peripherals, and application software. Every time a manufacturer introduced a new computer system to take advantage of improved technology, it had to develop software and components specifically for that system while continuing to maintain those for the previous systems. When end users switched to new machines, they had to rewrite all their existing programs, and they ran the risk of losing critical data if software conversions were botched. As a result, many customers were reluctant to lease or purchase new equipment.

The developers of the System/360 attacked that problem head-on. They conceived of a family of computers that would include machines of different sizes suitable for different applications, all of which would use the same instruction set and could share peripherals. To achieve this compatibility, they applied the principle of *modularity in design*: that is, the System/360's designers divided the designs of the processors and peripherals into *visible* and *hidden* information. IBM set up a Central Processor Control Office, which established and enforced the visible overall design rules that determined how the different modules of the machine would work together. The dozens of design teams scattered around the world had to adhere absolutely to these rules. But each team had full control over the hidden elements of design in its module—those elements that had no effect on other modules. (See the insert "A Guide to Modularity.")

When IBM employed this approach and also made the new systems compatible with existing software (by adding "emulator" modules), the result was a huge commercial and financial success for the company and its customers. Many of IBM's mainframe rivals were forced to abandon the market or seek niches focused on customers with highly specialized needs. But modularity also undermined IBM's dominance in the long run, as new

companies produced their own so-called plug-compatible modules—printers, terminals, memory, software, and eventually even the central processing units themselves—that were compatible with, and could plug right into, the IBM machines. By following IBM's design rules but specializing in a particular area, an upstart company could often produce a module that was better than the ones IBM was making internally. Ultimately, the dynamic, innovative industry that has grown up around these

A growing number of industries are poised to extend modularity from the production process to the design stage.

modules developed entirely new kinds of computer systems that have taken away most of the mainframe's market share.

The fact that different companies (and different units of IBM) were working independently on modules enormously boosted the rate of innovation. By concentrating on a single module, each unit or company could push deeper into its workings. Having many companies focus on the design of a given module fostered numerous, parallel experiments. The module designers were free to try out a wide range of approaches as long as they obeyed the *design rules* ensuring that the modules would fit together. For an industry like computers, in which technological uncertainty is high and the best way to proceed is often unknown, the more experiments and the more flexibility each designer has to develop and test the experimental modules, the faster the industry is able to arrive at improved versions.

This freedom to experiment with product design is what distinguishes modular suppliers from ordinary subcontractors. For example, a team of disk drive designers has to obey the overall requirements of a personal computer, such as data transmission protocols, specifications for the size and shape of hardware, and standards for interfaces, to be sure that the module will function within the system as a whole. But otherwise, team members can design the disk drive in the way they think works best. The decisions they make need not be communicated to designers of other modules or even to the system's architects, the creators of the visible design rules. Rival disk-drive designers, by the same token, can experiment with completely

A Guide to Modularity

Modularity is a strategy for organizing complex products and processes efficiently. A *modular system* is composed of units (or modules) that are designed independently but still function as an integrated whole. Designers achieve modularity by partitioning information into *visible design rules* and *hidden design parameters*. Modularity is beneficial only if the partition is precise, unambiguous, and complete.

The *visible design rules* (also called *visible information*) are decisions that affect subsequent design decisions. Ideally, the visible design rules are established early in a design process and communicated broadly to those involved. Visible design rules fall into three categories:

- *An architecture*, which specifies what modules will be part of the system and what their functions will be.
- *Interfaces* that describe in detail how the modules will interact, including how they will fit together, connect, and communicate.
- *Standards* for testing a module's conformity to the design rules (can module X function in the system?) and for measuring one module's performance relative to another (how good is module X versus module Y?).

Practitioners sometimes lump all three elements of the visible information together and call them all simply "the architecture," "the interfaces," or "the standards."

The *hidden design parameters* (also called *hidden information*) are decisions that do not affect the design beyond the local module. Hidden elements can be chosen late and changed often and do not have to be communicated to anyone beyond the module design team.

different engineering approaches for their versions of the module as long as they, too, obey the visible design rules.¹

Modularity Outside the Computer Industry

As a principle of production, modularity has a long history. Manufacturers have been using it for a century or more because it has always been easier to make complicated products by dividing the manufacturing process into modules or *cells*. Carmakers, for example, routinely manufacture the components of an automobile at different sites and then

bring them together for final assembly. They can do so because they have precisely and completely specified the design of each part. In this context, the engineering design of a part (its dimensions and tolerances) serves as the visible information in the manufacturing system, allowing a complicated process to be split up among many factories and even outsourced to other suppliers. Those suppliers may experiment with production processes or logistics, but, unlike in the computer industry, they have historically had little or no input into the design of the components.

Modularity is comparatively rare not only in the actual design of products but also in their use. *Modularity in use* allows consumers to mix and match elements to come up with a final product that suits their tastes and needs. For example, to make a bed, consumers often buy bed frames, mattresses, pillows, linens, and covers from different manufacturers and even different retailers. They all fit together because the different manufacturers put out these goods according to standard sizes. Modularity in use can spur innovation in design: the manufacturers can independently experiment with new products and concepts, such as futon mattresses or fabric blends, and find ready consumer acceptance as long as their modules fit the standard dimensions.

If modularity brings so many advantages, why aren't all products (and processes) fully modular? It turns out that modular systems are much more difficult to design than comparable interconnected systems. The designers of modular systems must know a great deal about the inner workings of the overall product or process in order to develop the visible design rules necessary to make the modules function as a whole. They have to specify those rules in advance. And while designs at the modular level are proceeding independently, it may seem that all is going well; problems with incomplete or imperfect modularization tend to appear only when the modules come together and work poorly as an integrated whole.

IBM discovered that problem with the System/360, which took far more resources to develop than expected. In fact, had the developers initially realized the difficulties of ensuring modular integration, they might never have pursued the approach at all because they also underestimated the System/360's market value. Customers wanted it so much that their willingness to pay amply justified IBM's increased costs.

We have now entered a period of great advances in modularity. Breakthroughs in materials science and other fields have made it easier to obtain the deep product knowledge necessary to specify the

design rules. For example, engineers now understand how metal reacts under force well enough to ensure modular coherence in body design and metal-forming processes for cars and big appliances. And improvements in computing, of course, have dramatically decreased the cost of capturing, processing, and storing that knowledge, reducing the cost of designing and testing different modules as well. Concurrent improvements in financial markets and innovative contractual arrangements are helping small companies find resources and form alliances to try out experiments and market new products or modules. In some industries, such as telecommunications and electric utilities, deregulation is freeing companies to divide the market along modular lines.

In automobile manufacturing, the big assemblers have been moving away from the tightly centralized design system that they have relied on for much of this century. Under intense pressure to reduce costs, accelerate the pace of innovation, and improve quality, automotive designers and engineers are now looking for ways to parcel out the design of their complex electro-mechanical system.

The first step has been to redefine the cells in the production processes. When managers at Mercedes-Benz planned their new sport-utility assembly plant in Alabama, for example, they realized that the complexities of the vehicle would require the plant to control a network of hundreds of suppliers according to an intricate schedule and to keep substantial inventory as a buffer against unexpected developments. Instead of trying to manage the supply system directly as a whole, they structured it into a smaller set of large production modules. The entire driver's cockpit, for example—including air bags, heating and air-conditioning systems, the instrument cluster, the steering column, and the wiring harness—is a separate module produced at a nearby plant owned by Delphi Automotive Systems, a unit of General Motors Corporation. Delphi is wholly responsible for producing the cockpit module according to certain specifications and scheduling requirements, so it can form its own network of dozens of suppliers for this module. Mercedes' specifications and the scheduling information become the visible information that module suppliers use to coordinate and control the network of parts suppliers and to build the modules required for final production.

Volkswagen has taken this approach even further in its new truck factory in Resende, Brazil. The

company provides the factory where all modules are built and the trucks are assembled, but the independent suppliers obtain their own materials and hire their own workforces to build the separate modules. Volkswagen does not "make" the car, in the sense of producing or assembling it. But it does establish the architecture of the production process and the interfaces between cells, it sets the standards for quality that each supplier must meet, and it tests the modules and the trucks as they proceed from stage to stage.

So far, this shift in supplier responsibilities differs little from the numerous changes in supply-chain management that many industries are going through. By delegating the manufacturing process to many separate suppliers, each one of which adds value, the assembler gains flexibility and cuts costs. That amounts to a refinement of the pattern of modularity already established in production. Eventually, though, strategists at Mercedes and other automakers expect the newly strengthened module makers to take on most of the design responsibility as well—and that is the point at which

The driver's cockpit for Mercedes' new sport-utility vehicle is produced by a plant owned by General Motors.

modularity will pay off the most. As modularity becomes an established way of doing business, competition among module suppliers will intensify. Assemblers will look for the best-performing or lowest cost modules, spurring these increasingly sophisticated and independent suppliers into a race for innovation similar to the one already happening with computer modules. Computer-assisted design will facilitate this new wave of experimentation.

Some automotive suppliers are already moving in that direction by consolidating their industry around particular modules. Lear Seating Corporation, Magna International, and Johnson Controls have been buying related suppliers, each attempting to become the worldwide leader in the production of entire car interiors. The big car manufacturers are indirectly encouraging this process by asking their suppliers to participate in the design of modules. Indeed, GM recently gave Magna total responsibility for overseeing development for the interior of the next-generation Cadillac Catera.

In addition to products, a wide range of services are also being modularized—most notably in the financial services industry, where the process is far along. Nothing is easier to modularize than stocks and other securities. Financial services are purely intangible, having no hard surfaces, no difficult shapes, no electrical pins or wires, and no complex computer code. Because the science of finance is sophisticated and highly developed, these services are relatively easy to define, analyze, and split apart. The design rules for financial transactions arise from centuries-old traditions of bookkeeping combined with modern legal and industry standards and the conventions of the securities exchanges.

As a result, providers need not take responsibility for all aspects of delivering their financial services. The tasks of managing a portfolio of securities, for example—selecting assets, conducting trades, keeping records, transferring ownership, reporting status and sending out statements, and performing custody services—can be readily broken apart and seamlessly performed by separate suppliers. Some major institutions have opted to specialize in one such area: Boston's State Street Bank in custody services, for example.

Other institutions, while modularizing their products, still seek to own and control those modules, as IBM tried to control the System/360. For example, Fidelity, the big, mass-market provider of money management services, has traditionally kept most aspects of its operations in-house. However, under pressure to reduce costs, it recently broke with that practice, announcing that Bankers Trust Company would manage \$11 billion worth of stock index funds on its behalf. Index funds are a low-margin business whose performance is easily measured. Under the new arrangement, Bankers Trust's index-fund management services have become a hidden module in Fidelity's overall portfolio

that can then be reconfigured into derivative financial products. Such innovations have made global financial markets more fluid so that capital now flows easily even between countries with very different financial practices.

Competing in a Modular Environment

Modularity does more than accelerate the pace of change or heighten competitive pressures. It also transforms relations among companies. Module designers rapidly move in and out of joint ventures, technology alliances, subcontracts, employment agreements, and financial arrangements as they compete in a relentless race to innovate. In such markets, revenue and profits are far more dispersed than they would be in traditional industries. Even such companies as Intel and Microsoft, which have substantial market power by virtue of their control over key subsets of visible information, account for less of the total market value of all computer companies than industry leaders typically do.

Being part of a shifting modular cluster of hundreds of companies in a constantly innovating industry is different from being one of a few dominant companies in a stable industry. No strategy or sequence of moves will always work; as in chess, a good move depends on the layout of the board, the pieces one controls, and the skill and resources of one's opponent. Nevertheless, the dual structure of a modular marketplace requires managers to choose carefully from two main strategies. A company can compete as an architect, creating the visible information, or design rules, for a product made up of modules. Or it can compete as a designer of modules that conform to the architecture, interfaces, and test protocols of others. Both strategies require companies to understand products at a deep level and be able to predict how modules will evolve, but they differ in a number of important ways.

For an architect, advantage comes from attracting module designers to its design rules by convincing them that this architecture will prevail in the marketplace. For the module maker, advantage comes from mastering the hidden information of the

design and from superior execution in bringing its module to market. As opportunities emerge, the module maker must move quickly to fill a need and then move elsewhere or reach new levels of performance as the market becomes crowded.

Following the example of Intel and Microsoft, it is tempting to say that companies should aim to

Following Intel and Microsoft, it's tempting to say companies should control the visible rules.

offerings, much as Volkswagen's suppliers operate as hidden modules in the Resende factory system.

The other result of the intrinsic modularity of financial instruments has been an enormous boost in innovation. By combining advanced scientific methods with high-speed computers, for example, designers can split up securities into smaller units

How Palm Computing Became an Architect

In 1992, Jeff Hawkins founded Palm Computing to develop and market a handheld computing device for the consumer market. Having already created the basic software for handwriting recognition, he intended to concentrate on refining that software and developing related applications for this new market. His plan was to rely on partners for the basic architecture, hardware, operating system software, and marketing. Venture capitalists funded Palm's own development. The handwriting recognition software became the key hidden module around which a consortium of companies formed to produce the complete product.

Sales of the first generation of products from both the consortium and its rivals, however, were poor, and Palm's partners had little interest in pursuing the next generation. Convinced that capitalizing on Palm's ability to connect the device directly to a PC would unlock the potential for sales, Hawkins and his chief executive, Donna Dubinsky, decided to shift course. If they couldn't get partners to develop the new concept, they would handle it themselves—at least the visible parts, which included the device's interface protocols and its operating system. Palm would have to become an architect, taking control of both the visible information and the hidden information in the handwriting recognition module. But to do so, Hawkins and Dubinsky needed a partner with deeper pockets than any venture capital firm would provide.

None of the companies in Palm's previous consortium was willing to help. Palm spread its net as far as US Robotics, the largest maker of modems. US Robot-

ics was so taken with the concept for and development of Palm's product that it bought the company. With that backing, Palm was able to take the product into full production and get the marketing muscle it needed. The result was the Pilot, or what Palm calls a Personal Connected Organizer, which has been a tremendous success in the marketplace. Palm remains in control of the operating system and the handwriting recognition software in the Pilot but relies on other designers for hardware and for links to software that runs on PCs.

Palm's strategy with the Pilot worked as Hawkins and Dubinsky had intended. In order for its architecture to be accepted by customers and outside developers, Palm had to create a compelling concept that other module makers would accept, with attractive features and pricing, and bring the device to market quickly. Hawkins's initial strategy—to be a hidden-module producer while partners delivered the architecture—might have worked with a more familiar product, but the handheld-computer market was too unformed for it to work in that context. So, when the other members of the consortium balked in the second round of the design process, Palm had to take the lead role in developing both the proof of concept and a complete set of accessible design rules for the system as a whole.

We are grateful to Myra Hart for sharing with us her ongoing research on Palm. She describes the company in detail in her cases "Palm Computing, Inc. (A)," HBS case no. 396245, and "Palm Pilot 1995," HBS case, forthcoming.

control the visible design rules by developing proprietary architectures and leave the mundane details of hidden modules to others. And it is true that the position of architect is powerful and can be very profitable. But a challenger can rely on modularity to mix and match its own capabilities with those of others and do an end-run around an architect.

That is what happened in the workstation market in the 1980s. Both of the leading companies, Apollo Computer and Sun Microsystems, relied heavily on other companies for the design and production of most of the modules that formed their workstations. But Apollo's founders, who emphasized high performance in their product, designed a proprietary architecture based on their own operating and network management systems. Although some modules, such as the microprocessor, were bought off the shelf, much of the hardware was de-

signed in-house. The various parts of the design were highly interdependent, which Apollo's designers believed was necessary to achieve high levels of performance in the final product.

Sun's founders, by contrast, emphasized low costs and rapid time to market. They relied on a simplified, nonproprietary architecture built with off-the-shelf hardware and software, including the widely available UNIX operating system. Because its module makers did not have to design special modules to fit into its system, Sun was free of the investments in software and hardware design Apollo required and could bring products to market quickly while keeping capital costs low. To make up for the performance penalty incurred by using generic modules, Sun developed two proprietary, hidden hardware modules to link the microprocessor efficiently to the workstation's internal memory.

In terms of sheer performance, observers judged Apollo's workstation to be slightly better, but Sun had the cost advantage. Sun's reliance on other module makers proved superior in other respects as well. Many end users relied on the UNIX operating system in other networks or applications and preferred a workstation that ran on UNIX rather than one that used a more proprietary operating system. Taking advantage of its edge in capital productivity, Sun opted for an aggressive strategy of rapid growth and product improvement.

Soon, Apollo found itself short of capital and its products' performance fell further and further behind Sun's. The flexibility and leanness Sun gained through its nonproprietary approach overcame the performance advantages Apollo had been enjoying through its proprietary strategy. Sun could offer customers an excellent product at an attractive price, earn superb margins, and employ much less capital in the process.

However, Sun's design gave it no enduring competitive edge. Because Sun controlled only the two hidden modules in the workstation, it could not lock its customers into its own proprietary operating system or network protocols. Sun did develop original ideas about how to combine existing modules into an effective system, but any competitor could do the same since the architecture—the visible information behind the workstation design—was easy to copy and could not be patented.

Indeed, minicomputer makers saw that workstations would threaten their business and engineering markets, and they soon offered rival products, while personal computer makers (whose designs were already extremely modular) saw an opportunity to move into a higher-margin niche. To protect itself, Sun shifted gears and sought greater control over the visible information in its own system. Sun hoped to use equity financing from AT&T, which

As a rule, managers will have to become much more attuned to all sorts of developments in the design of their products.

controlled UNIX, to gain a favored role in designing future versions of the operating system. If Sun could control the evolution of UNIX, it could bring the next generation of workstations to market faster than its rivals could. But the minicomputer

makers, which licensed UNIX for their existing systems, immediately saw the threat posed by the Sun-AT&T alliance, and they forced AT&T to back away from Sun. The workstation market remained wide open, and when Sun stumbled in bringing out a new generation of workstations, rivals gained ground with their own offerings. The race was on—and it continues.

Needed: Knowledgeable Leaders

Because modularity boosts the rate of innovation, it shrinks the time business leaders have to respond to competitors' moves. We may laugh about the concept of an "Internet year," but it's no joke. As more and more industries pursue modularity, their general managers, like those in the computer industry, will have to cope with higher rates of innovation and swifter change.

As a rule, managers will have to become much more attuned to all sorts of developments in the design of products, both inside and outside their own companies. It won't be enough to know what their direct competitors are doing—innovations in other modules and in the overall product architecture, as well as shifting alliances elsewhere in the industry, may spell trouble or present opportunities. Success in the marketplace will depend on mapping a much larger competitive terrain and linking one's own capabilities and options with those emerging elsewhere, possibly in companies very different from one's own.

Those capabilities and options involve not only product technologies but also financial resources and the skills of employees. Managers engaged with modular design efforts must be adept at forging new financial relationships and employment contracts, and they must enter into innovative technology ventures and alliances. Harvard Business School professor Howard Stevenson has described entrepreneurship as "the pursuit of opportunity beyond the resources currently controlled," and that's a good framework for thinking about modular leadership at even the biggest companies. (See the inserts "How Palm Computing Became an Architect" and "How Quantum Mines Hidden Knowledge.")

At the same time that modularity boosts the rate of innovation, it also heightens the degree of uncertainty in the design process. There is no way for managers to know which of many experimental approaches will win out in the marketplace. To prepare for sudden and dramatic changes

How Quantum Mines Hidden Knowledge

Quantum Corporation began in 1980 as a maker of 8-inch disk storage drives for the minicomputer market. After the company fell behind as the industry shifted to 5.25-inch drives, a team led by Stephen M. Berkley and Dave Brown rescued it with an aggressive strategy, applying their storage expertise to developing a 3.5-inch add-on drive for the personal computer market. The product worked, but competing in this sector required higher volumes and tighter tolerances than Quantum was used to. Instead of trying to meet those demands internally, Berkley and Brown decided to keep the company focused on technology and to form an alliance with Matsushita-Kotobuki Electronics Industries (MKE), a division of the Matsushita Group, to handle the high-volume, high-precision manufacturing. With the new alliance in place, Quantum and MKE worked to develop tightly integrated design capabilities that spanned the two companies. The products resulting from those processes allowed Quantum to compete successfully in the market for drives installed as original equipment in personal computers.

Quantum has maintained a high rate of product innovation by exploiting modularity in the design of its own products and in its own organization. Separate, small teams work on the design and the production of each submodule, and the company's leaders have developed an unusually clear operating framework within which to coordinate the efforts of the teams while still freeing them to innovate effectively.

In addition to focusing on technology, the company has survived in the intensely competitive disk-drive industry by paying close attention to the companies that assemble personal computers. Quantum has become the preferred supplier for many of the assemblers because its careful attention to developments in the visible information for disk drives has enabled its drives to fit seamlessly into the assemblers' systems. Quantum's general managers have a deep reservoir of knowledge about both storage technology and the players in the sector, which helps them map the landscape, anticipating which segments of the computer market are set to go into decline and where emerging opportunities will arise. Early on, they saw the implications of the Internet and corporate intranets, and with help from a timely purchase of Digital Equipment Corporation's stagnating storage business, they had a head start in meeting the voracious demand for storage capacity that has been created by burgeoning networks. Despite what some observers might see as a weak position (because the company must depend on the visible information that other companies give out) Quantum has prospered, recently reporting strong profits and gains in stock price.

We are grateful to Steven Wheelwright and Clayton Christensen for sharing with us their ongoing research on Quantum. They describe the company in more detail in their case "Quantum Corp.: Business and Product Teams" HBS case no. 692023.

in markets, therefore, managers need to be able to choose from an often complex array of technologies, skills, and financial options. Creating, watching, and nurturing a portfolio of such options will become more important than the pursuit of static efficiency per se.

To compete in a world of modularity, leaders must also redesign their internal organizations. In order to create superior modules, they need the flexibility to move quickly to market and make use of rapidly changing technologies, but they must also ensure that the modules conform to the architecture. The answer to this dilemma is modularity within the organization. Just as modularity in design boosts innovation in products by freeing designers to experiment, so managers can speed up development cycles for individual modules by splitting the work among independent teams, each pursuing a different submodule or different path to improvement.

Employing a modular approach to design complicates the task of managers who want to stabilize the manufacturing process or control inventories because it expands the range of possible product varieties. But the approach also allows engineers to create families of parts that share common characteristics and thus can all be made in the same way, using, for example, changes in machine settings that are well understood. Moreover, the growing power of information technology is giving managers more precise and timely information about sales and distribution channels, thus enhancing the efficiency of a modular production system.

For those organizational processes to succeed, however, the output of the various decentralized teams (including the designers at partner companies) must be tightly integrated. As with a product, the key to integration in the organization is the visible information. This is where leadership is critical. Despite what many observers of leadership are

now saying, the heads of these companies must do more than provide a vision or goals for the decentralized development teams. They must also delineate and communicate a detailed operating framework within which each of the teams must work.

Such a framework begins by articulating the strategy and plans for the product line's evolution into which the work of the development teams needs to fit over time. But the framework also has to extend into the work of the teams themselves. It must, for example, establish principles for matching appropriate types of teams to each type of project. It must specify the size of the teams and make clear what roles senior management, the core design team, and support groups should play in carrying out the project's work. Finally, the framework must define processes by which progress will be measured and products released to the market. The framework may also address values that should guide the teams in their work (such as leading by example). Like the visible information in a modular product, this organizational framework establishes an overall structure within which teams can operate, provides ways for different teams and other groups to interact, and defines standards for testing the merit of the teams' work. Without careful direction, the teams would find it easy to pursue initiatives that may have individual merit but stray from the company's defining concepts.

Just like a modular product that lacks good interfaces between modules, an organization built around decentralized teams that fail to function according to a clear and effective framework will

suffer from miscues and delays. Fast changing and dynamic markets—like those for computers—are unforgiving. The well-publicized problems of many computer companies have often been rooted in inadequate coordination of their development teams as they created new products. Less obvious, but equally important, are the problems that arise when teams fail to communicate the hidden information—the knowledge they develop about module technology—with the rest of the organization. That lack of communication, we have found, causes organizations to commit the same costly mistakes over and over again.

To take full advantage of modularity, companies need highly skilled, independent-minded employees eager to innovate. These designers and engineers do not respond to tight controls; many reject traditional forms of management and will seek employment elsewhere rather than submit to them. Such employees do, however, respond to informed leadership—to managers who can make reasoned arguments that will persuade employees to hold fast to the central operating framework. Managers must learn how to allow members of the organization the independence to probe and experiment while directing them to stay on the right overall course. The best analogy may be in biology, where complex organisms have been able to evolve into an astonishing variety of forms only by obeying immutable rules of development.

A century ago, the railroads showed managers how to control enormous organizations and masses of

Further Reading

For more information on modular product design, see Steven D. Eppinger, Daniel E. Whitney, Robert P. Smith, and David Gebala, "A Model-Based Method for Organizing Tasks in Product Development," *Research in Engineering Design* 6, 1994. For more about modular processes, see James L. Nevin and Daniel E. Whitney, *Concurrent Design of Products and Processes* (New York: McGraw-Hill, 1989). For more information on the design of financial securities and the global financial system, see Robert C. Merton and Zvi Bodie, "A Conceptual Framework for Analyzing the Financial Environment" and "Financial Infrastructure and Public Policy: A Functional Perspective," in *The Global Financial System: A Functional Perspective*, (Boston, Massachusetts: Harvard Business School Press, 1995).

For descriptions of how companies compete in industries using modular products, see Richard N. Langlois and Paul L. Robertson, "Networks and Innovation in a Modular System: Lessons from the Microcomputer and Stereo Component Industries," *Research Policy*, August 1992; Charles R. Morris and Charles H. Ferguson, "How Architecture Wins Technology Wars," *HBR* March-April 1993; Raghu Garud and Arun Kumaraswamy, "Changing Competitive Dynamics in Network Industries: An Exploration of Sun Microsystems' Open Systems Strategy," *Strategic Management Journal*, July 1993, p. 351; and Clayton M. Christensen and Richard S. Rosenbloom, "Explaining the Attacker's Advantage: Technological Paradigms, Organizational Dynamics, and the Value Network," *Research Policy*, March 1995, p. 233.

capital. In the world fashioned by computers, managers will control less and will need to know more. As modularity drives the evolution of much of the economy, general managers' greatest challenge will be to gain an intimate understanding of the knowledge behind their products. Technology can't be a black box to them because their ability to position the company, respond to market changes, and guide internal innovation depends on this knowledge. Leaders cannot manage knowledge at a distance merely by hiring knowledgeable people and giving them adequate resources. They need to be closely involved in shaping and directing the way knowledge is created and used. Details about the inner workings of products may seem to be merely technical engineering matters, but in the context of in-

tense competition and fast changing technology, the success of whole strategies may hinge on such seemingly minor details.

1. Practical knowledge of modularity has come largely from the computer industry. The term *architecture* was first used in connection with computers by the designers of the System/360: Gene M. Amdahl, Gerrit A. Blaauw, and Frederick P. Brooks, Jr., in "Architecture of the IBM System/360," *IBM Journal of Research and Development*, April 1964, p. 86. The scientific field of computer architecture was established by C. Gordon Bell and Allen Newell in *Computer Structures: Readings and Examples* (New York: McGraw-Hill, 1971). The principle of *information hiding* was first put forward in 1972 by David L. Parnas in "A Technique for Software Module Specification with Examples," *Communications of the ACM*, May 1972, p. 330. The term *design rules* was first used by Carver Mead and Lynn Conway in *Introduction to VLSI Systems* (Reading, Massachusetts: Addison-Wesley, 1980). Sun's architectural innovations, described in the text, were based on the work of John L. Hennessy and David A. Patterson, later summarized in their text *Computer Architecture: A Quantitative Approach* (San Mateo, California: Morgan Kaufman Publishers, 1990).

Reprint 97502

To order reprints, see the last page of this issue.



"Oops, doppelgängers!"

Harvard Business Review and Harvard Business School Publishing content on EBSCOhost is licensed for the individual use of authorized EBSCOhost patrons at this institution and is not intended for use as assigned course material. Harvard Business School Publishing is pleased to grant permission to make this work available through "electronic reserves" or other means of digital access or transmission to students enrolled in a course. For rates and authorization regarding such course usage, contact permissions@hbsp.harvard.edu