The

By DAVID L. BRADFORD and ALLAN R. COHEN

The enormous popularity of books on how to achieve excellence in organizations points to the unprecedented hunger that business people feel for a way to recapture the momentum of quality and innovation that was once the hallmark of U.S. business.

As valuable as these books might be, they focus almost exclusively on the total organization and speak to the chief executive officer, who makes large-scale decisions, even though the implementation of these plans falls to managers in the next levels down. The head of data processing, for example, cannot alter the organization's culture, create lifetime employment practices or invent a new corporate strategy.

Thus there lies a great gap between what is known about what organizations ought to be like and what individual managers need to do to achieve excellence. Yet it is the middle and upper-middle managers who hold the key to high performance. If they can produce, their leadership has a cascading effect on several levels of important subordinates.

Bradford and Cohen have developed an explicit leadership model to bridge the gap in attaining organizational excellence, one that suggests ways for managers to behave and influence every day. It asserts that the model of leadership used by most managers—that of the "heroic" leader who can know all, do all, solve every problem—is severely outdated and inadequate in eliciting the best performance in complex, contemporary organizations.

What do Bradford and Cohen offer in place of the John Wayne-style managerial hero? The postheroic leader: the Manager-as-Developer.

David L. Bradford is a lecturer at the Stanford University Graduate School of Business and is executive director of the Organizational Behavior Teaching Society. Allan R. Cohen is a professor of management at Babson College and senior vice-president of Goodmeasure, Inc., Boston, Mass.
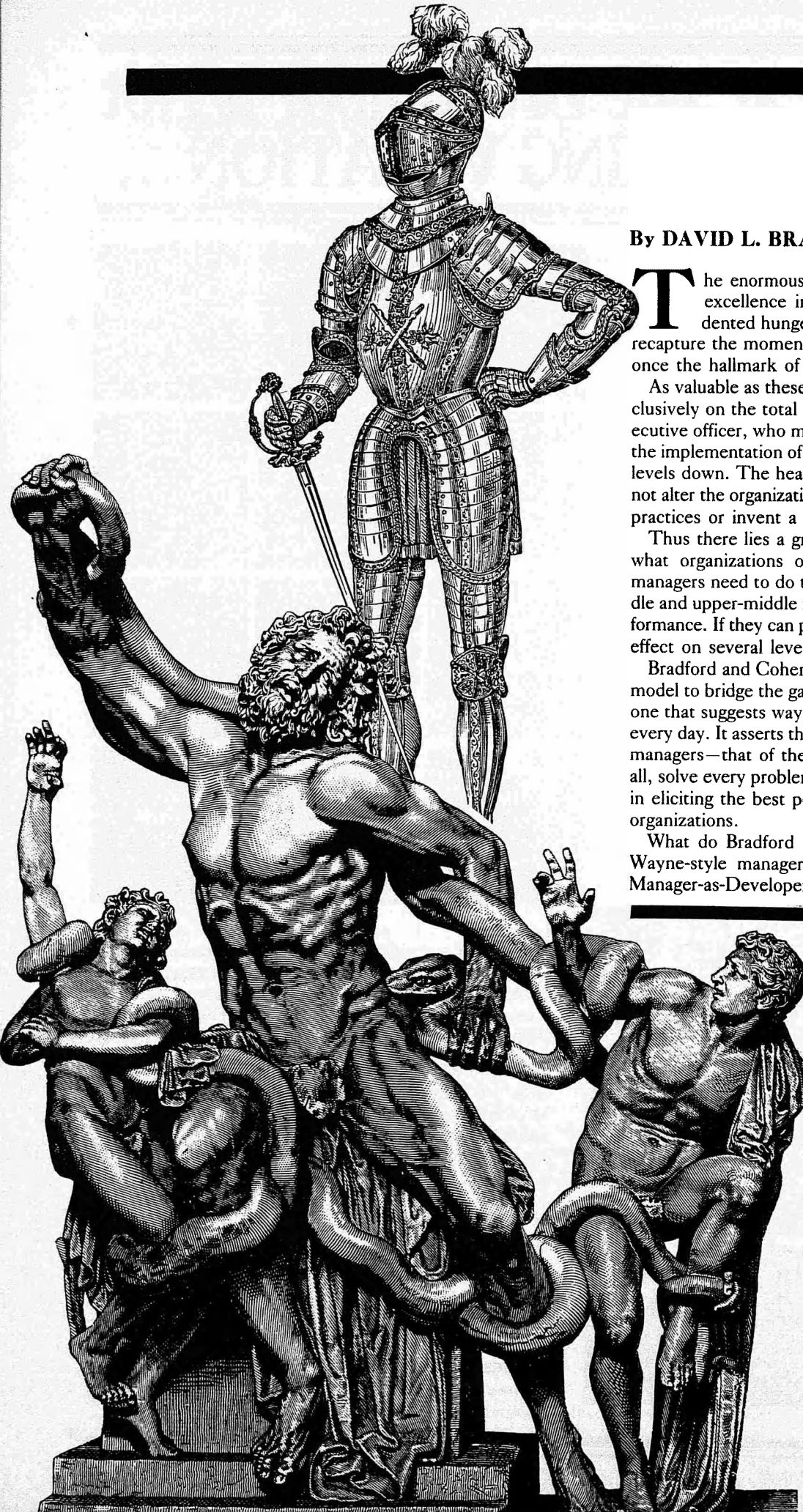
Postheroic Leader

How can managers act in ways that will achieve excellence rather than block it? How can leaders act so that heroic over-responsibility doesn't prevent full use of subordinates' abilities, doesn't dampen their commitment to high performance and doesn't cause them to avoid taking initiative when problems arise?

The answer is not for the manager to renounce responsibility and abandon all control. Although some subordinates respond well to being left alone, there are too many important managerial tasks to guarantee that a no-boss-is-the-best-boss approach will work.

It is also futile to seek a magical mid-point between the extremes of too much and too little leader responsibility and control. Rather, an entirely new definition of leadership is required if a department is to be led into new and unanticipated areas.

This new definition is a fundamental reorientation away from the heroic model.
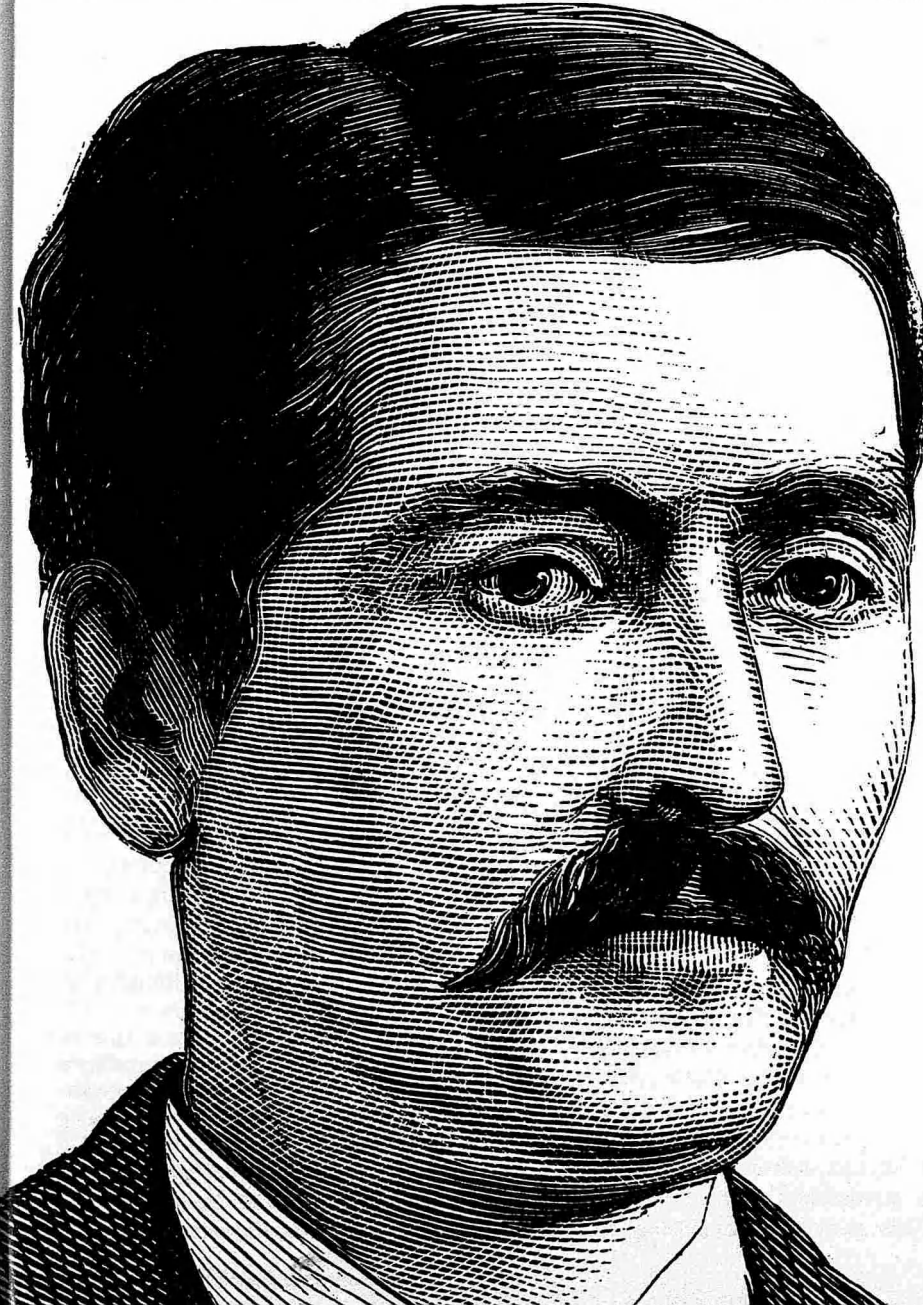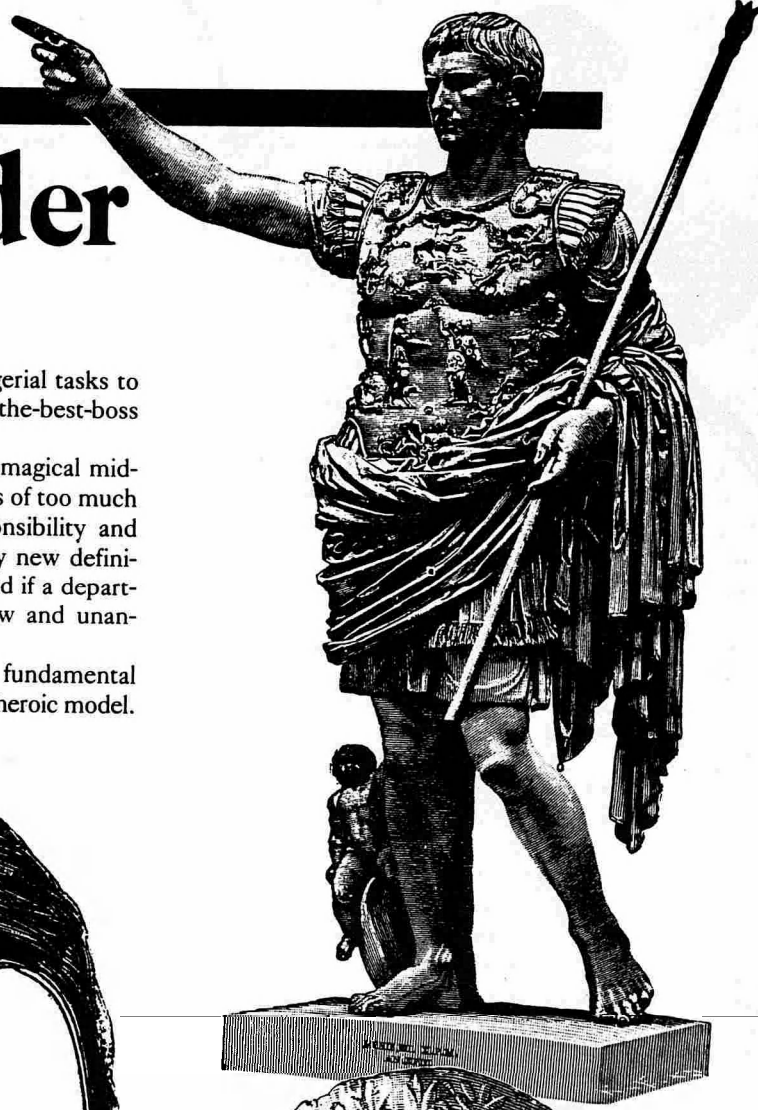
Shared responsibility and control take the place of the individual hero carrying the burdens alone. For those who have staggered around trying to do it all—and have ended up forcing more burden onto their own shoulders by causing the very problems with subordinates that they wanted to avoid—the postheroic view of managing is a profound shift.

At the same time, no part of the new model is all that unfamiliar to any contemporary manager. Everyone knows about tapping subordinate talents, exciting them about the mission and building effective teams, just as astronomers in the time of Copernicus knew about the earth, sun and stars whirling in the heavens. Yet the "simple" reversal of putting the sun, rather than the earth, at the center of relationships caused a revolution in philosophy as well as science.

To achieve excellence in earthly organizations, a manager must first believe in the concept and then act in the creation of a team of key subordinates who are jointly responsible with the manager for the department's success. *At the same time that the manager works to develop management responsibility in subordinates, he or she must help develop the subordinates' abilities to share management of the unit's performance.* Only when all directly reporting subordinates are committed to joint responsibility for overall excellence—when pieces of the task are no longer conditionally delegated, but become parts of decisions—will control cease to be the sole province of the boss.

At the same time, only when subordinates become skilled in the managerial tasks required for total departmental success can the sharing of responsibility lead to excellence. Since neither willingness to accept overall responsibility nor ability to do so are automatic and instant, we have called this model the Manager-as-Developer. Skills have to be learned, common goals accepted, expectations changed and norms modified. Over time a team can be built.

This management model was created by examining leaders who achieve excellence and the images they seem to hold of managing. Rather than depending on heroic rides to the rescue—with the answers and the total responsibility—they have sought the far greater power and potential for excellence available in the commitment and abilities of their whole group. These managers have in mind a developmental, collaborative, galvanizing, but subordinate-centered image.

Unfortunately, a good, easily recognized image of this kind of leader does not exist. Flashes of description occasionally appear when leaders refer to times when "we were all in it together and we knew we could conquer the world." Certain inspirational moments display the quality of shared responsibility; the leader does not command yet is willingly joined in a pursuit where all are at risk.

Perhaps the image is most like a very demanding but supportive and inspirational coach, who works hard to bring the team along, insists on high standards and rigorous effort, but passes on all the knowledge that will help the athletes grow. This coach often works alongside the team, but delegates increasing responsibility for the game plan and especially for on-the-spot adjustments. From the sidelines the coach takes great pleasure in the centrality and achievements of the athletes.

But this coaching analogy imperfectly transfers to managing: The manager is a more involved participant in the action than a coach can ever be. Nevertheless, leaders who achieve excellence are less likely to be guided by images of the central, overresponsible and overcontrolling hero—an image that ultimately dilutes the effectiveness of technicians and conductors when they have expert subordinates doing complex and interdependent work in changing circumstances.

This new model calls for no less effort, energy, investment or imagination than does the Lone Ranger style. Since active engagement is necessary to undertake and sustain increased subordinate learning and overall responsibility, we think of the developer as postheroic rather than nonheroic. Indeed, for some managers who are used to rushing in with the answers, it takes heroic efforts not to be so heroic. Developer-managers learn to have impact without exerting total control, to be helpful without having all the answers, to get involved without demanding centrality, to be powerful without needing to dominate and to act responsible without squeezing others out.

The central orientation

What does it take to fulfill this tall order in management? First and foremost is a change in orientation. A whole new array of options open up when the leader's orientation becomes: "How can each problem be solved in a way that further develops my subordinates' commitment and capabilities?"

This new orientation may lead the manager to throw some problems back to subordinates ("I think you have a good handle on the difficulty, so why don't you do a first cut and come back in two days"). The manager may ask questions that help subordinates focus on the key issues, while at other times the leader can best aid development by exploring the situation jointly with a subordinate. There may also be times when it is most "developmental" for the manager to provide the answer. Whichever alternative is chosen, the underlying developmental orientation of the leader remains consistent.

What these responses have in common is achieving the dual goal of getting the job done while engaging subordinates in a way that helps them stretch. This orientation does not sacrifice task accomplishment for development (or vice versa). The leader is not saying, "I will develop today and put off solving the problem until tomorrow." Both goals are kept firmly in the forefront.

This postheroic orientation also requires that subordinate development not be restricted to off-site training functions. It occurs on the job—in real time with real issues. Fuller use is made of the already existing abilities of the subordinate while potential ones are developed.

Let's look closely at how a Manager-as-Developer would use this orientation to deal with a problem. Bob Young is faced with a problem: Increasingly, customers have been complaining about defective gaskets, a crucial component in the company's key product.[1] Concerned about this dangerous situation, Bob has called a special meeting of the operations committee. The four subordinate members of the committee have strong feelings—positive and negative—about one another and about Bob Young that make them reluctant to talk openly about the actual source of the problem: a change in supplies and inspection procedures.

This is just the sort of issue that is likely to elicit the heroic response; the organization is under severe time pressure to solve a crucial problem that subordinates are colluding to bury. Bob opened the meeting with his four subordinates as follows:

"As you know, we are having major problems meeting our deliveries on time. Furthermore, customers are reporting that quality is not up to standard. I have heard this from several of our very important customers, and I am quite bothered, as I am sure you are. You know that we are in a highly competitive market and they

Rather than depending on heroic rides to the rescue, the Manager-as-Developer seeks the far greater power and potential for excellence available in the commitment and abilities of the whole group.

can go elsewhere. We have always prided ourselves on our ability to produce gaskets that have no defects, that our customers can use without worry. Our financial future is in jeopardy unless we quickly find out what is going wrong and correct it. You are the people who best know the situation; you know what causes it, and you know what the best solution looks like. Therefore, I want us in this meeting to work together and come up with the best answer."

This introduction signaled the importance of the problem, reminded subordinates of the company's identification with quality and gave an initial indication of how Bob wanted this situation to be tackled.

Yet subordinates know that words are cheap. What did Bob really mean? Was he out to find someone to blame? Did he really have an idea of how the problem should be solved and merely used a pseudoparticipative style to get them to buy into his solution?

In addition, Bob Young's statement indicated a change in the rules of the game. In trying to develop a management team with members who feel responsible for the operations of the plan and not just for individual areas, Bob Young was demanding that members change their ways of operating. No matter what objections they might have had to Bob's previous style, the subordinates had learned to live with (and around) it. Thus, they tested the waters very carefully before they jumped in and accepted at face value his statement about their enlarged responsibility.

First, they tried to revert to the previous way of operating by throwing the responsibility back to him.

"I don't know, Bob. You know the operations inside and out. What do you think the best solution is?"

Bob replied,

"This is the kind of issue we need to tackle together, because then we'll be sure not only of getting this problem solved, but we'll be able to prevent similar dilemmas in the future."

This exchange was followed by long silence; subordinates hoped they could outlast Bob and force him to take over. When it was clear that he wouldn't move

in, there was another attempt to resurrect the old way of operating, in which information that might appear detrimental to any one of the four subordinates was kept from Bob while they worked out their problems on the side. The head of production, Don Blue, glanced over to the quality control manager, Roy Gray, and turned back to Bob:

"Bob, you are busy getting us major contracts. We don't have to take up valuable meeting time on this issue. Roy Gray and I will meet and come up with the solution, and I'll let you know tomorrow."

This suggestion gives the appearance of subordinates assuming responsibility, but it was actually an attempt to hide any dirty linen from Bob Young. Also, the response did nothing to build the operations committee into a mature team that could handle the major operating problems.

As a Developer, Bob Young recognized the presence of two problems: a technical one—quality products were not being delivered on time—and a managerial one—for some reason the operations group had been unable over the past several weeks to resolve the task problem. The defective gaskets problem wasn't news to any of them. Thus, Bob realized that if he delegated responsibility for solving the technical problem to two of his subordinates (or if Bob, being heroic, came up with an answer himself), he wouldn't solve the team/managerial problem even if a good technical solution was produced.

Rather than wading in to solve the problem personally, Bob decided that he had to hold his subordinates' feet to the fire while they solved the problem. He had to be sure they knew he expected them to work jointly on the solution. Bob thus responded to Don Blue's suggestion by saying:

"Don, I'm sure you and Roy could come up with something, but I also want all of us to improve our collective ability to solve problems. To do that we need to work on it together, since everyone's involved."

After a long silence Bob turned to Fran, the most junior member of the group, and asked for possible causes of the problem.
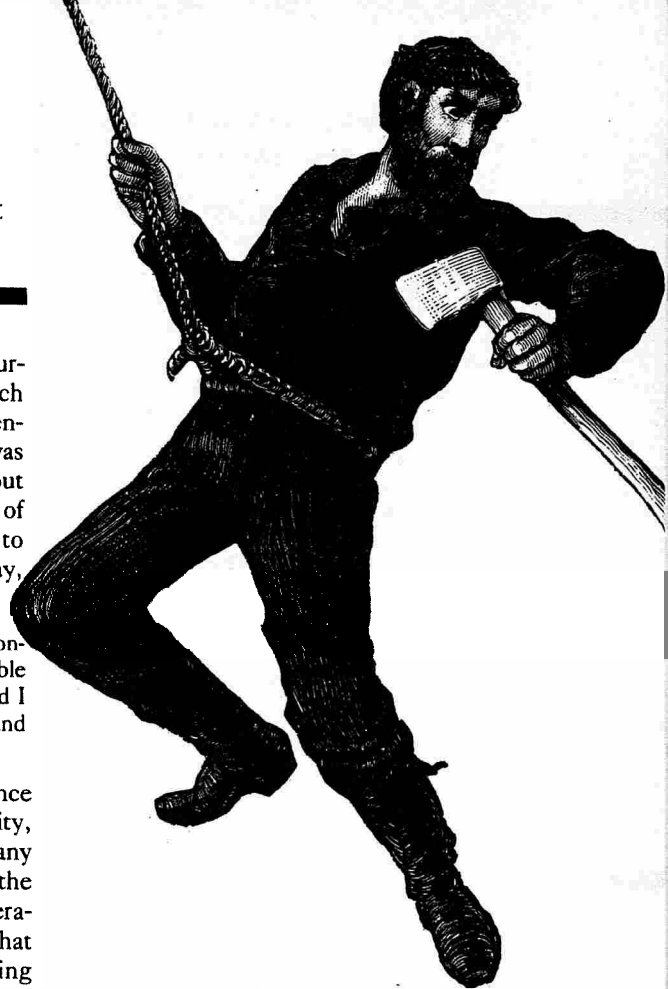
Bob wanted to be certain that Don's seniority and dominant style would not keep Fran from contributing. After Fran made a few comments suggesting that the old-timers' resistance to new methods was an issue, the others leaped in to start blaming one another. A hail of "you didn'ts" and "you should haves" filled the room. Bob showed no signs of being upset about this arguing, and after several minutes, when the accusations had died down, he said,

"Let's see if we can keep our eyes on what happened and how to fix it. Pointing fingers only makes everyone defensive, which doesn't help quality control. What advantages did the old system have, and what were the goals in changing the system? Once we have these answers, we can determine the kind of changes we should make in purchasing practice so we get the advantages without the problems."

After Bob redirected the discussion into a problem-solving mode, the members collectively dug in to get at the basic issues. They acknowledged that they should stop playing at "It's not my fault," and instead they set up a series of procedures for solving the immediate consequences of the procurement change. They also agreed that a more thorough discussion of inventory policy was needed

43

and set a time the next day to work out the specifics.

Bob Young worked hard to increase the team's willingness to share the responsibility for managing the plant. He recognized that one general statement would not be sufficient. He wanted the team to own not only the problem, but the solution as well.

Instead of trying to get them to disclose the information that would give him determination of the solution or manipulating them to come up with his solution, he worked at how to get the problem solved while increasing the team's commitment and capacity to solve such problems. He kept standards high by using their tradition of quality as the measure against which they could all judge what was being proposed. And by sticking to his determination to increase shared responsibility, he helped develop the team's ability to work together on future problems.

Benefits of the Developer approach

The Developer approach to management has several distinct biases toward excellence. First, the chance is increased that tasks will be accomplished at a high level of quality. Task excellence is more likely when members seize new opportunities as they develop; uncover problems and difficulties early, before they become major crises; share their knowledge and expertise; and feel committed to carrying out decisions.

These four determinants of excellence are especially important in contemporary organizations. High rates of change are accompanied by increased opportunities for innovation. When members feel responsible for departmental excellence, their responses to new conditions are quicker and more appropriate.

Rapid change also brings with it greater likelihood that familiar solutions, practices and procedures will soon become outdated. Often, members are aware of developing problems before the leader. If the working climate encourages such difficulties to be raised quickly, not hidden, problems can be tackled as they are beginning to smolder rather than when they have grown into major conflagrations.

When tasks are complex, it is likely that different people know different parts of the problem. When subordinates are competent, they are likely to bring diverse knowledge and expertise. If the problem is multifaceted and members have different but complementary abilities, groups can make superior decisions.

Finally, high member commitment to departmental excellence increases the likelihood that solutions will be fully implemented. Part of the quality of a solution rests with how well it is carried out. Workable solutions take into account political factors in the organization, the skills and styles of those having to implement the decision and unique aspects of each situation. Thus, even a leader's technically correct answer may be inferior in practice to collaborative solutions of members who know the specific conditions that must be met to put the solution into practice.

Actual implementation, in turn, has a greater chance for success if multiple resources stand behind it. A task is too often assumed to be the sole province of the individual to whom it is formally assigned; the heroic style prevents an individual from asking for help. A team that is committed to excellence will also recognize what assistance is needed.

When the perspective of subordinates is broadened beyond a narrow concern for the specialized area of each and includes responsibility for the unit as a whole, chances are increased that the four components of task quality will be achieved. Quality will naturally increase as member competencies to solve future difficulties grow from success in solving present problems.

The Developer approach has the added benefit that increased feelings of responsibility by subordinates are not limited to task issues but extend to managerial ones as well. Thus far, the various forms of participative management that have been advocated have done a better job of involving subordinates in specific task problems than they have in making subordinates feel responsible for the successful management of the unit.

A third benefit of the Developer approach is increased subordinate motivation. If subordinates seek tasks that develop their abilities, they will find the increased responsibility of this management model highly satisfying. If they want to advance into managerial positions, they will be attracted to a job that involves learning how to work with others and sharing in the management of the unit. The opportunities for personal learning and development help build useful career skills. Full participation in departmental

People do not want to work for money alone.

decision making means fewer frustrating bottlenecks at the level of the boss.

Components of the model in action

We have emphasized the importance of a new conceptual orientation; but the model, by itself, is not usually sufficient. In action, the Developer model consists of three components that must be nurtured to ensure that excellence is achieved.

• *Component 1—Building a shared-responsibility team.* The kinds of organizational settings we have been discussing are faced with complex problems with solutions that require complicated coordination before they can be implemented. Only rarely do members have self-contained tasks; assistance from others is often required to implement decisions and programs. Difficulties with adequate coordination are compounded by a never-ending stream of externally induced changes. Policies, procedures and regulations must be frequently modified in response to changes in tasks, technologies and clients' needs. Thus, previously arranged modes of cooperation and past practices rapidly become obsolete.

Departments that strive for excellence have to develop methods to handle this high degree of interdependence and increasing rate of change. Successful departments must be quickly responsive to new conditions and be able to untangle swiftly the snarls that inevitably develop when strong, capable individuals have to integrate their efforts.

Problems of coordination and responsiveness increase as members strive for excellence. People may be willing to put up with frustration and bureaucratic delays when they are not personally invested in their work or when they seek merely to perform at a minimum level of competence. But as they become more committed to quality work, their tolerance decreases for what they see as unnecessary delays and inadequate solutions on the part of their peers.

These problems of bringing together the necessary information and producing the needed coordination are growing each year at increasing speed. As a result, leaders who feel totally responsible for resolving these issues take on an impossible burden. Seeing that work is fairly allocated, tasks are properly coordinated, problems raised and resolved and work performed at high standards turns the manager into the white rabbit of *Alice in Wonderland*, always hurrying but forever late. There is just too much for one person to know and do.

The solution, then, is not for the leader to work harder and run faster, but instead to build a team that shares in the responsibility of managing the department. A departmental team is not a group that functions to advise and counsel the leader; it is a joint-responsibility group that shares in making the core decisions and in influencing each other to insure high-level performance.

Since the subordinates are the ones who have to be coordinated and thus have a vested interest in their peers performing well, adequate coordination and control is more likely to occur if members can influence each other. There are more points of contact; all-controlling-all allows more influence than one-controlling-all. Members of the team must function not only as heads of their own specialized areas but as part of the overall management function for the department. Subordinates must wear both hats—fighting for the interests of their subunit but equally on the line for the department's overall interest.

An example of these dynamics can be seen in a staff discussion about Blue Cross benefits. Hourly employees objected to the existing plan and suggested some changes; it was clear that the company's response would be seen as a measure of its concern for dissatisfied workers. The production manager, personnel head and marketing manager all agreed with the general manager that they "had no choice and had to be responsive." The finance manager, however, insisted on raising the issue of costs. Frustrated by her fellow managers' apparent fear of worker reactions, the finance manager finally turned to the marketing manager and said, "Do you realize this new plan will add 1.5 cents per unit to our product cost? Are you ready to tell that to your customers?"
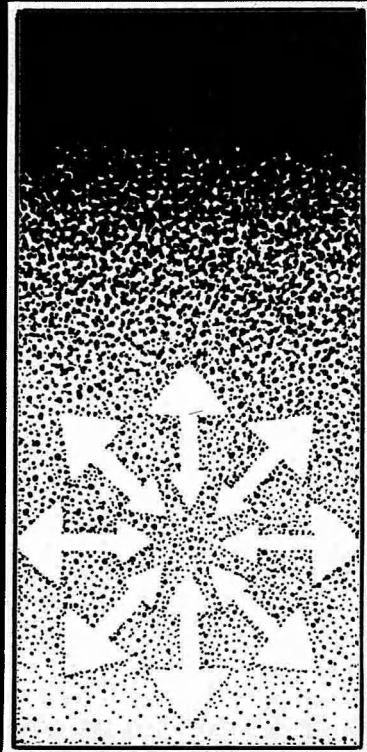
"You really know how to hurt a guy," was the reply, and an animated discussion followed. The finance manager eventually acknowledged the need to respond to worker concerns and agreed to the health plan changes, but she succeeded in seeing that financial issues were not overlooked in the stampede to a decision. Bringing a team to where subunit viewpoints are fully considered, yet the total context receives every member's concern, is difficult but essential.

Having to deal with the dual loyalties of the members teaches important managerial skills: balancing competing interests; seeing issues in a wider context; working collaboratively; sustaining good working relationships with those who do not agree and who are prepared to fight for their views; and making sacrifices for the greater good without being a pushover. These skills are useful even for subordinates who do not directly supervise anyone else.

Building such a team, one with members involved in key decisions and genuinely sharing responsibility for the working of all the parts, should lead to greater subordinate commitment and motivation as well as better performance. After all, being responsible for making the decision is more involving for a subordinate than trying to influence the leader who is making the decision.

A cohesive, mature team increases the range of skills and knowledge available to the department. In a rapidly changing environment that constantly demands new knowledge and skills, leaders can no longer be expected always to have all the necessary experience. If a team can be built of members who feel free to share knowledge, a much wider range of skills becomes accessible.

A more diverse range of managerial behavior also becomes available. As others have pointed out, contradictory

45

demands are placed on a leader.[2] Sometimes the manager needs to be risk-taking and entrepreneurial, other times more prudent. Sometimes the manager should hold fast to a position, sometimes should be more conciliatory. Rather than expecting one person to be able to perform perfectly at all times, it is more realistic to expect that a richer variety of leadership skills will exist among the total membership of the management team. Drawing upon members' skills can supplement the leader's capabilities.

Another advantage of such a team is that the amount of control in the unit is increased since its sources are multiplied. Managers therefore seldom have to deal with the natural limits to their personal control capacities, to how many projects and subordinates they can personally oversee. When a heroic manager is able to come up with a good solution, there is still no indication that subordinates will carry it out with much enthusiasm. The heroic manager still must chase around trying to gain compliance.

On the other hand, members who feel ownership of a solution also feel that it is their right and responsibility to influence others who are implementing the solution. Members' adoption of this managerial perspective serves as an excellent control mechanism to see that tasks are completed well and on time. Groups can exert greater pressure than one individual can, even when that individual is the boss. Most people are even less willing to face their peers with an unfinished or inadequate performance than they are the boss.

Developer managing makes fuller use of the resources of the members and increases their competencies. It forces subordinates to take account of a wider array of issues, and it requires more sophisticated abilities—the same managerial ones that the leader uses. Thus, this is a system that starts to build on itself. Members more rapidly learn management skills when working collectively to make the decisions with which they all have to live.

Unfortunately, too few management teams come anywhere close to realizing the potential benefits from teamwork. It is possible to achieve full performance and learning benefits only if managers are willing to give up their heroic styles and work in a more developmental way. It is necessary to transform a collection of individuals who push their individual interests and hide or avoid the real issues into a mature, productive team. This process is neither easy nor instantaneous, but it can be done.

• *Component 2—Continuous development of individual skills.* A shared-responsibility team can work well only when all members have the skills to master these additional responsibilities. As one concerned manager put it, "All this stuff about real teamwork and sharing of the overall problems is well and good, but you wouldn't be so positive if you saw some of the turkeys who work for me." Ideally, everyone who reports to a would-be Developer is already fully competent, but the usual situation finds most subordinates with reasonable technical competence, some with managerial skills, but few who are strong in interpersonal and group areas.

A team cannot mature if some of its members can't be trusted to take responsibility. Although even the most developed group will include variations in member competence, each person has to have a minimal level of managerial ability to accompany the requisite technical knowledge. For members to engage in consensual decision making and mutual influence, they must have managerial and interpersonal skills. And if members are to comment on problems that involve their colleagues, they must have enough technical knowledge of these areas to help form judicious decisions.

Expanded technical knowledge can be gained through work assignments and discussions in team meetings, where members can learn from each other, particularly when everybody feels free to share the problems and issues of sub-areas. Managerial and interpersonal skills pose the greatest obstacle in this system.

Lack of technical knowledge is seldom the reason that a person above the rank of supervisor fails to perform. Performance difficulties usually arise from the "softer" issues of how to motivate others, handle conflict, make and meet commitments, conduct meetings and use influence. The most frequently mentioned reason (except for economic staff reduction) in numerous surveys conducted by recruiting firms for why managerial-rank personnel lose their jobs is that the person in some way "didn't get along well," not because particular knowledge was lacking.

It is in these cooperative, managerial areas that Developers need to pay greatest attention to subordinate growth and learning. These skills are the most vital toward subordinates' becoming fully contributing participants in the

management of the unit. Too often managers tend to share responsibility with those few subordinates they regard as already fully developed, and leave the remainder of the team unengaged until everything else is out of the way, risk is low and time is ample—a state that never arrives. Such a manager might alternatively choose to ship the difficult ones off to a management training program, where they can be "shaped up." But, the second option is problematic and out of the manager's control.

The Manager-as-Developer must use daily interaction with the subordinate as the setting for growth. Day-to-day interactions of various members of the team contain the greatest potential for teaching and learning crucial interactive managerial skills. Tasks can be assigned that broaden the subordinate's knowledge and skills; coaching can occur that builds competencies (as well as increases the probability of successful task accomplishment); and feedback can be given on a continual basis. In addition, the subordinate learns from how bosses go about developing—how they ask questions, support and confront, make suggestions.

Acquiring technical, managerial and interpersonal knowledge becomes even more salient when the subordinate is learning to help manage the whole department. The subordinate is not being groomed for nebulous future responsibilities, but for crucial present ones. Thus, continuous development is a critically important aspect of the postheroic model.

This approach offers several important benefits for the manager and subordinate. First, the material being worked on is fresh and first-hand. The clumsy subordinate, for example, may not need an external workshop on influence skills if the manager gets together with this subordinate immediately after the meeting and says, "Chris, I think you gave up your position too quickly—as I've seen you do before. If you had instead tried recasting your points to show the person you were arguing with how she would benefit, your view might have been understood without your appearing bullheaded." This kind of interaction, close to the event and reinforced over time, increases the chance that new learning will take hold.

The second benefit of this developmental orientation is that it can turn a behavioral problem into an opportunity for growth. Most managers resist giving negative feedback for fear that it will alienate and discourage subordinates. But

if the underlying message is, "There is more you can do, and I have confidence that you can do it," then problems can be raised directly.

The third benefit of a continual developmental orientation is its extremely motivating and energizing effect on the subordinate. Since we are talking about the kind of subordinates who are turned on by challenge and opportunity, the developer, in keeping the growth of individuals central and in the service of task accomplishment, keeps motivation high. Thus, even difficult, nasty tasks may become occasions for learning needed managerial skills.
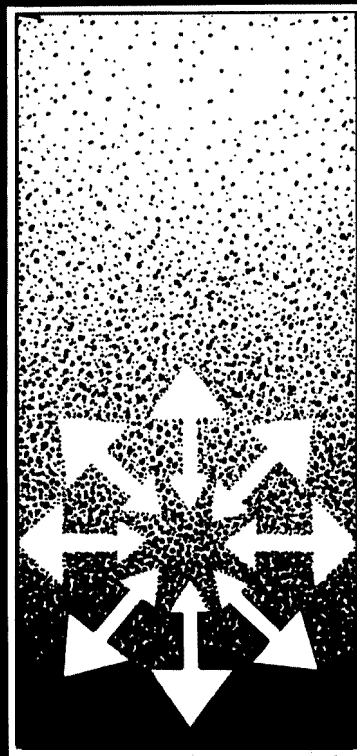
The chance to grow by being helped to learn from even negative experiences is a powerful attraction to ambitious subordinates—they can tolerate almost anything, including the intense dedication needed to achieve excellence, if their efforts are perceived as leading somewhere. Young professionals such as lawyers, medical interns and new business graduates will knock themselves out for several years because they accept their hard labor as necessary training before they can handle even larger responsibilities in the future. But rather than requiring extra effort as dues for future rewards, the Developer links today's learning with carrying out today's important technical and managerial responsibilities.

This is not to suggest that all subordinates can be developed. Some do not have the interest, motivation or ability. But before any subordinate is written off or relegated to a dead-end job, the manager should strive to know his or her full potential. A developmental stance toward both the performer and the underperformer has the dual advantage of making fuller use of abilities while quickly revealing those cases where a subordinate can't or won't perform adequately. Furthermore, since this orientation is not punitive, an incompatibility between what the department needs and what the subordinate can deliver generally leads to a mutual agreement to separate rather than a disputed termination.

To illustrate how subordinate development can occur through day-to-day tasks, let's see how a Manager-as-Developer handles a problem in widget production and distribution.

Subordinate: We're having trouble delivering disposable widgets to Techcorp.

Manager D: Oh-oh, that doesn't sound good. What's going on? Can I be of any assistance?

47

Subordinate: Well, I thought I'd let you know about it and run through the actions I've taken already. I think I know what to do, but I don't want to miss any bases.

Manager D: Fine. Go ahead.

Subordinate: I got a call from Dan saying we couldn't make schedule. When I asked why, he gave me a song and dance about poor materials. Since I know he likes to use that as an excuse, I called Ellen in purchasing, who told me the stuff was exposed to excess moisture while Dan was storing it. Dan, of course, denies it.

Manager D: So what did you do?

Subordinate: I haven't done anything about that part of it yet. I don't think we should get directly into a confrontation with Dan, do you?

Manager D: That depends on what our options are. In general, it's not a good idea to make anyone lose face, but let's see if there are any other choices. We've got to find a way to meet our commitments, or we'll lose our edge as the most reliable manufacturing unit in the business. What else have you done?

Subordinate: I checked with Techcorp's purchasing agent about why the pressure. Turns out that their people made some promises to one of their customers without checking on actual availability. She's going to see whether she can get the customer to allow some slippage.

Manager D: Will they accept a looser tolerance on the widgets? If they would, would that help us?

Subordinate She's going to check that, too. Then we'd have to fight with our quality control people. If I need you, would you talk to Ted's boss to pave the way?

Manager D: You expect trouble from Ted?

Subordinate: He's done it before.

Manager D: How would you approach him?

Subordinate: I've always tried to play it straight and tell him what our needs are. But he's a stubborn old coot.

Manager D: What do you suppose his interests are?

Subordinate: I don't know; I've never thought about that. I suppose he's very proud of our quality and doesn't want his reputation to be hurt.

Manager D: Besides, he's been at that job a long time and is likely to be there forever.

Subordinate: If I involve him in deciding how much tolerance the product can take, he'll respond better, and we'll come up with something reasonable.

Manager D: That sounds on target. Do you suppose it would help to reinforce with him the organization's theme of dependable and reliable service?

Subordinate: I think so, especially if I put it in that light. In fact, I've got another idea. I'll call him and get him together with the Techcorp buyer. That would be exciting for him, I'll bet.

Manager D: Sounds good. When you put yourself in the shoes of someone who gives you a hard time, it's amazing how different things look. What else?

Subordinate: It just hit me; there may be some ways to reschedule some other production so we can concentrate on widgets. I know how to proceed now. I'll touch base this afternoon. Thanks.

In this exchange, the Developer is constantly working to help the subordinate reach an excellent solution while acquiring important managerial skills. Working out problems around an issue that involves many parties, which in turn requires the ability to diagnose other people's needs and develop an appropriate strategy for gaining cooperation, is a valuable lesson for this subordinate. The Developer uses the conversation to reinforce the unit's reliability goal and, with that as a base, expands on what the subordinate must take into account when trying to solve a potentially difficult organizational problem.

The Developer perspective not only allows focus on the crucial managerial skills that help subordinates prepare for advancement; it helps the manager balance the organization's needs with the individual's interests. The potential for tension always exists between individual and organizational needs; heroic styles are almost inevitably overconcerned with short-term task accomplishments at the expense of member interests. "Get the job done and we'll worry about your personal development when we have time" is an all-too-familiar refrain. Although the Developer must also be working toward outstanding task accomplishment, the skills the individual will need for advancement are linked with responsibility for total unit task accomplishment.

Of course, some subordinates are interested only in learning further technical skills or working on technical problems, even after they understand how this view may limit their career opportunities. When being solely technical is what the person wants, he or she should be supported with appropriate assignments.

Nevertheless, for the type of work setting we have been describing, the interest of most subordinates in learning to be more effective managers coincides with the organization's needs to give managerial talent a chance to unfold. Working in a department that strives for excellence by sharing overall responsibility and teaching managerial skills through daily interaction is an excellent way to enhance one's career opportunities; the accomplishment of tasks at an excellent level of performance in turn enhances the department's objectives.

This approach to managing also encourages the best use of the maximum number of people. If all reporting subordinates (except perhaps the "closet technicians" who just want to be left alone) are to share in unit responsibility, the Developer has to find ways to work well with each one. This imperative applies even to those individuals who are, from the manager's point of view, difficult employees.

■ *Component 3—Determining and building a common department vision.* Even though the creation of a shared-responsibility team and developmental relationships may move a department far beyond the heroic model in building member commitment, it may still not be sufficient to produce the level of excellence of which the department is capable. What is needed is the third component of the Manager-as-Developer model: a larger purpose that gives meaning to work, whether routine or nonroutine. Unexciting but necessary tasks can still have importance if they are in the service of larger, clearly important goals.

A central vision can also improve the coordination of effort. Even if the key management team assumes some of the responsibility for coordination of individual efforts, problems of integration may persist. After all, reasonable people differ. Coordination problems can be especially acute if there is no agreement on the department's central purpose. Decisions about priorities and allocation of scarce resources and agreement on new products and services may be difficult or even impossible to achieve.

The leader must articulate and gain member commitment to an exciting departmental overarching goal. Although all departments have a purpose and some even have perfunctory mission statements, more is needed: an overarching goal that captures the unique thrust of the specific unit.

Such a goal fulfills several functions. First, it unites and inspires members with a vision that justifies extra effort. People do not want to work for money alone. If subordinates are to move beyond a minimum level of performance and above the mundane, there has to be a purpose—a reason that makes the extra effort worthwhile.

Evidence indicates that no successful overarching goal can be expressed in exclusively financial terms; that is, aiming for "the best return on investment" or "the greatest financial growth" by itself won't

excite organizational members to work to potential.[3] The real value of financial measures in a productive organization is to mark the achievement of other goals, such as overcoming a challenge or being a cutting-edge operation.

In some ways it is easier to frame exciting goals for whole organizations—"progress is our most important product" or "we cure diseases by new medicines." But such statements are often abstract and vague for members not at the top. Regardless of whether the organization as a whole has a clear, inspiring mission, excellent departments need to have their own exciting goals. An engineering department that designs machine tools might focus on its aim "to fight inflation by designing equipment that reduces manufacturing costs." A computer software design group might define itself by the motto: "We know our clients' needs so well that our programs produce far more benefits than any of our clients anticipate."

Common themes in these examples are innovation and quality in important human areas. At some very personal level, goals of this kind speak to the individual's concern about doing something of value. There is the potential for the work to speak to a core need within each person—the need to be a part of an organization that makes a difference.

The second important function of an overarching goal is to serve as a standard by which to make decisions. If the goal is specific enough, it can be a guideline for choosing among options. For example, the sales departments of two separate companies operating in the same field might have very different overarching goals. One department might say, "We are the best in providing low-cost yet quality goods to help our clients compete in the marketplace." The other department might define itself as "the best in the business at anticipating changes in consumer desires," a statement that reflects the firm's lead in product development. In order to be consistent with these quite different goals, each of these sales departments must make strikingly different decisions about the way they are organized, who is hired, the training to

This article is adapted from Managing for Excellence: The Guide to Developing High Performance in Contemporary Organizations, *copyright John Wiley & Sons, Inc., 1984. For further information, contact Kenneth MacLeod, John Wiley & Sons, Inc., 605 Third Avenue, New York, NY 10158. (212) 850-6592.*

be provided, the nature of intracompany relationships and the types of reward systems offered.

The third function of an overarching goal is to make clear the departmental direction; the goal defines the future. Part of the responsibility of leadership is to move beyond status quo in a number of important areas—quality of performance, products or services, work methods or organizational structure. By specifying the area in which the department seeks to be excellent, the goal points toward tomorrow and can carry departmental members beyond today's irritations.

Interactions among the three components

The impact of the three parts of this leadership model is further increased because each one builds on the others. For example, the leader may help the group function better by taking individuals aside after meetings and showing them how they can be more productive in the team. Likewise, as the group develops in trust and maturity, members will be willing to examine openly how they are working together. That examination, in turn, is a further source of individual learning as members comment on each other's behavior. Part of how a leader can get members to adopt new behavior is in the service of a larger goal. Groups are more willing to disagree over the means if there is consensus that all are striving to achieve the same goal. And members pulling together further strengthens that vision.

The unusual thing about the Manager-as-Developer model is that it is both a description of an end state—where subordinates have formed a team that achieves excellence by sharing responsibility for achieving an exciting goal—and a description of how to achieve that state. With the developer orientation, it is possible to use the concept of the team from the beginning and start to share responsibility immediately. The leader can immediately start to improve working relationships with individual subordinates and even begin to explore a viable overarching goal. Indeed, a good way to figure out how to move a group of recalcitrant individuals into a shared-responsibility team is to share that responsibility with them.

When the individual subordinates are technically competent and working at complex interdependent tasks, development activities can move quickly. However, when these basic conditions

are not in place, they must be addressed, which sometimes means that the leader must act autonomously to build technical competencies, find new people or reorganize tasks. At other times, a developmental approach can be used in tackling these deficiencies.

For a leader who is just starting to manage as a Developer, it does not make sense to renounce control totally when some subordinates are resistant, dependent, abrasive, uncooperative or unable to work together. However, the fear a manager has that giving up heroic control and responsibility will inevitably lead to chaos is exactly what prevents traditional managers from changing.

Thinking developmentally can free the manager from this paralysis. Seeing change as a step at a time moves the issue beyond "giving up control" to "expanding control." As noted earlier, an increase in shared responsibility increases the total amount of control available. Subordinates can help deal with problems of motivation, commitment or collaboration.

Another way to keep problems from blocking change is to see those very difficulties as opportunities for change. Just as Bob Young used a severe quality problem to build his team, and the widget manager used a typical production problem to teach problem-solving skills, managers who start interacting daily in developmental ways can rapidly create tangible progress.

Development is a never-ending process. Development never arrives at an end place, where all is in order and the machinery moves so smoothly that maintenance is no longer needed. Once a unit reaches high gear, it is possible to manage with less friction and uphill pushing, but development will always be needed. Even the most advanced departments face changes in external conditions, shifts in priorities and assignments, turnover in personnel, expansion and contraction in size of the unit and just normal running down.

References

1. Adapted from Bowen, D.D. Young Manufacturing. In D.T. Hall et. al., *Experiences in management and organizational behavior.* New York: Wiley, 1982.
2. Adizes, I. *How to solve the mismanagement crisis.* Homewood, Ill.: Dow-Jones, 1979.
3. Peters, T.J. Symbols, patterns and settings: An optimistic case for getting things done. *Organizational Dynamics,* Autumn 1979.

49