

APPLIED INTELLIGENCE

Innovative Management Techniques Increase Productivity

Managers can use new development methodologies such as rapid applications development (RAD) to attract and motivate key development personnel.

Among the benefits of innovative applications-development methodologies

such as RAD is the ability to increase productivity and reduce turnover of development personnel.

When key developers leave, productivity is harmed. In a typical organization with about 50 professionals, the recruitment of a new, skilled developer is likely to cost two months' salary, and replicating the previous developer's skill level takes about three months. Training costs and moving costs must also be taken into account.

In all, a turnover of 20 percent in the organization costs about \$150,000 annually. This is just the cost reported by accountants; the real cost is much higher because development productivity suffers.

The figure shows the impact of turnover in a well-organized development shop that uses a traditional development life cycle with productivity tools and fourth-generation languages. The organization spent about \$3 million per year on this development; the difference between the development costs with high turnover (11.5 function points per person-month) and those with low turnover (19.3 function points per person-month) was about \$1 million annually.

The hidden costs were even higher because slow development caused by high turnover impeded the introduction of new efficiency measures and competitive business thrusts. This may have affected earnings by another \$1 million per year.

Turnover tends to be lower among developers who are excited about their jobs. Developers enjoy using powerful tools and creating results fast enough to see their effect, as is the case in RAD projects. Morale is often lowest on long, drawn-out traditional life cycles where a target delivery date is not achievable. High turnover on such projects worsens a bad situation.

There is less of a chance that developers with a three-month life cycle will leave in mid-project than will those with a two-year life cycle. Members of a small, highly motivated construction team are likely to avoid leaving in mid-project so that they do not disappoint their fellow team members.

Some IS organizations, however, have a very high turnover. Tom DeMarco and Timothy Lister, in their book "Peopleware, Productive Projects and Teams," say the typical turnover figures they encounter range from 33 percent to 80 percent per year, with average employee longevity from 15 to 36 months.

Some IS organizations treat analysts and programmers as disposable parts. As a result, these employees lack a sense of loyalty and have no feeling of long-term involvement in the job. They believe that the route to high salaries is through job-hopping.

stay and should make them feel that they are a special part of the organization.

A good programmer can create three times as much bug-free code as the average programmer—and a "genius" programmer creates 10 times as much.

The amount of code created by the least skilled programmer is much less than the amount created by an average programmer. The good programmer also often creates far more elegant program designs than the average programmer; the worst programmers make an unholy mess.

I-CASE tools change this situation, however, because they enforce well-

If programming is an art, I-CASE development is an engineering discipline. Programming requires the handcrafting of code one line at a time. I-CASE tools, however, produce code automatically from graphical design specifications.

The art and creativity in I-CASE development is not in devising code structures but in thinking about how to improve the world of the user, sometimes fundamentally restructuring how information is handled. This rethinking of procedures requires intelligent designers.

The RAD methodology, however, does not require genius-level programmers. Instead, it relies on high-end, integrated tool sets coupled with management techniques designed to motivate members of the development team. Key development personnel should be intensively trained to use the best available tools and organized into small teams that can move with speed.

People who are resistant to change should be avoided. Often the most productive programmers using older technologies are the most resistant to the changes introduced by new life-cycle processes such as RAD.

These people may be skilled in coding with COBOL or PL/I, or they may have become acrobats using older productivity-enhancement tools, such as ones that facilitate the generation of Customer Information Control System (CICS) transactions.

But they often feel threatened by new, unfamiliar technology. They do not see how new techniques can enhance their job security and do not want to take the risk of learning new technology.

The Right Choice

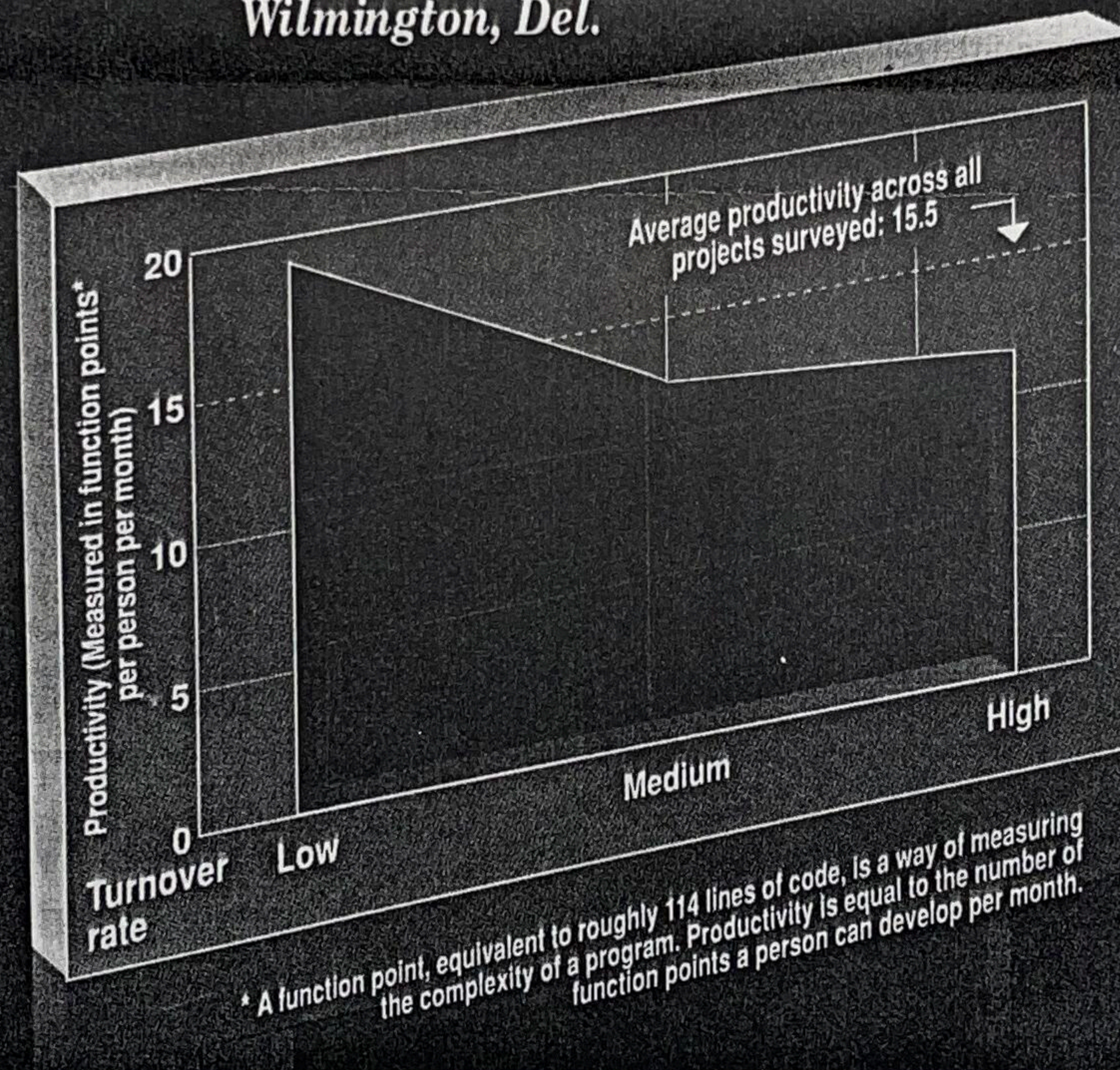
Individuals who adapt readily to the RAD life-cycle process are excited by the technology and strive for accomplishment. They intuitively recognize the advantages—both to themselves and to the business—of efficient development tools and small, highly motivated development teams. They are eager to learn the new techniques and are willing to work hard in return for recognition and a sense of accomplishment.

The RAD life cycle is highly dependent on the quality of the joint applications design leader and the members of the construction team. These individuals must be chosen with care. While training is a very important part of developing the requisite skills for a job, as with a sports team, you can't train just anyone to be a winner. You must select the right people.

Next week I will discuss the management of cultural change within an IS organization. ■

Personnel Turnover Jeopardizes Productivity

Development Statistics for 41 Projects
Over a Five-Year Period at Du Pont Fibers Department,
Wilmington, Del.



John Avakian

Rapid-applications development relies on high-end, integrated tool sets coupled with management techniques designed to motivate members of the development team.

In such an environment, turnover engenders turnover. Because people leave quickly, management believes it is not worth spending money on training. As a result, such organizations have had difficulty introducing integrated computer-aided software engineering (I-CASE) tools and establishing an environment that promotes high-speed development.

A RAD organization makes investments to build a high level of skill in its development teams and keeps the teams together. It has a professional career path that allows the most proficient team members to rise to high salaries. It should recruit developers who expect to

structured design. A new developer can easily understand and modify someone else's design, since the design is highly graphical, easy to understand and easy to modify.

There is less scope for a superb programmer to do something highly complex and difficult to understand; some programmers, therefore, do not want to work with I-CASE tools. I-CASE tends to have a leveling effect: The brilliant developer does better than the average developer, but not 10 times better. Most intelligent analysts can become highly skilled with I-CASE tools and usually find them fun to use.

The concepts embodied in RAD are described in a new volume in the James Martin Report Series. For more information on this volume, call (800) 242-1240. For information on seminars, contact (in the United States and Canada) Technology Transfer Institute, 741 10th St., Santa Monica, Calif. 90402 (213) 394-8305. In Europe, contact Savant, 2 New St., Carnforth, Lancs., LA5 9BX United Kingdom (0524) 734 505.