

# Best Practices to Increase Productivity and Decrease Turnover

## Research on People and Teams

By Janet Amirault, CEO, Software Consortium

Turnover, productivity, and meeting client objectives are large challenges in software environments. According to the Standish Group, 74% of projects fail to meet one or more of the following success criteria: deliver on-time, deliver on-budget, or deliver with the requested features. For software development managers and leaders, what are the options in managing in this difficult environment? From Software Consortium's experience and research, the areas of largest leverage are people and creating environments where people can thrive. This article explains the research and people levers that can be used to increase success for information technology clients, while at the same time increasing the performance and satisfaction of the software development team. Below is a summary of some research and recommendations from Peter Scholtes, Steve McConnell, Gerald Weinberg and other leaderships and software management experts.

According to Steve McConnell, there are four areas of leverage in software development projects: people, process, product and technology. **Of these, he states that "people has the greatest potential".**

From the research world, there are a number of studies that lend credibility to Steve McConnell's conclusions:

- Performance differences on the order of 10-to-1 or more between different developments with the same levels of experience (Sackmann, Erikson and Grant, 1968; Curtis, 1981; Mills, 1983; DeMarco and Lister, 1985; Curtis, et. al. 1986; Card, 1987; Calett and McGarry, 1989).
- Barry Boehm, in a study of 69 projects at TRW, identified that the best teams were at least 4 times as productive as the worst teams (Boehm, 1981).
- DeMarco and Lister in a study of 166 programmers from 18 companies identified programmer productivity differences of 5.6 to 1 (DeMarco and Lister 1985).
- In one study of 7 identical projects, the developers were all professional programmers with several years of experience who were enrolled in a

computer science graduate program. The products still ranged in effort by a factor of 3.4 to 1 (Boehm, Gray and Seewaldt, 1984).

### *What are the factors that lead to the variations?*

#### **Team Cohesiveness**

According to B. Lakhanpal in studying group cohesiveness, individual capabilities and experience across 31 projects, group cohesiveness was number one, followed closely by individual performance (Lakhanpal, 1983).

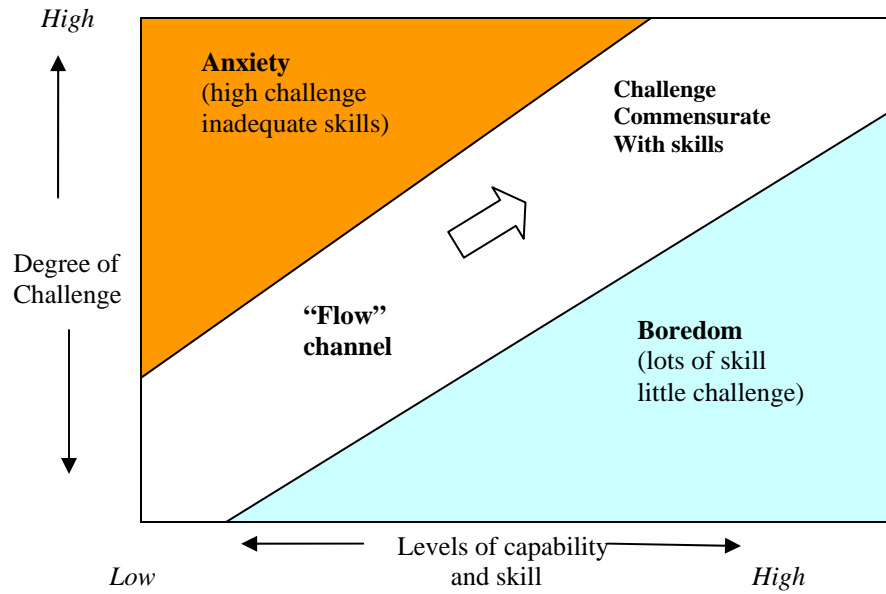
#### **Individual Performance**

##### *How does motivation relate to all this?*

Motivation is undoubtedly the single greatest influence on how well people perform. The research showed:

According to Barry Boehm, most productivity studies have found that motivation has a stronger influence on productivity than any other factor (Barry Boehm, 1981).

When environmental, skill and individual motivators are in place, "flow" (according to Csikszentmihalyi (1990) or the highest productivity possible is reached. Reference the following diagram.



If motivation is the single greatest influence on how well people perform, what are the top motivators?

	Programmer/Analysts	Managers of Programmers	General Population
1	Achievement	Responsibility	Achievement
2	Possibility for Growth	Achievement	Recognition
3	Work Itself	Work Itself	Work Itself
4	Personal life	Recognition	Responsibility
5	Technical supervision opportunity	Possibility for Growth	Advancement
6	Advancement	Interpersonal relationships, subordinate	Salary
7	Interpersonal relations, peers	Interpersonal relations, peers	Possibility for Growth
8	Recognition	Advancement	Interpersonal relations, subordinate
9	Salary	Salary	Status
10	Responsibility	Interpersonal relations, superior	Interpersonal relationship, superior

Adapted from *Software Engineering Economics* (Boehm, 1981) and "Who is the DP Professional?" (Fitz-enz, 1978)

**And the corollary, what are demotivators?**

The classic study on motivation was Frederick Herzberg, 1987. Herzberg differentiates between types of motivation – motivating factors ("satisfiers"), which stimulate performance when they are present and hygiene factors ("dissatisfiers"), which

degrade performance when they are not present.

**Demotivating hygiene factors for developers**

According to Steve McConnell's research, the demotivating factors for developers are:

- Management Manipulation
- Excessive Schedule pressure
- Lack of appreciation for developer's efforts
- Inappropriate involvement of technically inept management
- Not involving developers in decisions that affect them
- Productivity barriers
- Low quality environment
- Heavy handed motivation campaigns

### ***How can the Top 5 Developer Motivation Factors be Maximized?***

#### **Achievement**

The best way for managers to motivate developers is to provide an environment that makes it easy for them to focus on what they like doing, which is developing software.

#### **Ownership**

"Buy-in" is one key to achievement motivation. People will try harder to achieve their own goals than to achieve someone else's.

#### ***Goal setting***

Explicitly setting development objectives (goals) is a simple, obvious step in achieving accelerated software development and is often overlooked. Weinberg and Shulman (1974) found remarkable results in setting objectives and getting development teams to meet them.

Successful projects use goal setting to their advantage. Boeing applied this concept in their design of the 747.

Setting too many goals is a common problem and should be avoided. For best results (Larson and LeFasto, 1989), select one objective and make it clear that it is the most important one.

#### ***Possibility for Growth***

For software developers, one of the most exciting aspects of software is working in

a field that is constantly changing. An organization can harness this motivation by providing developers opportunities to grow on their assigned projects. Alignment of the growth goals of the individual with the growth goals of the organization is the key to meeting the goals of both the organization and the individual.

Software Consortium provides training, mentoring, individual goal setting and mentoring as options for their employees, as well as establishing an overall culture of learning from the executives down to the receptionist and time is set aside each week for learning.

According to Naisbett and Arbonne, (1985), "the best and brightest people will gravitate toward those corporations that foster personal growth."

#### **Work Itself**

Richard Hackmann and Greg Oldham (1980) believe that people's internal motivation comes from three sources:

- They must experience meaning in their work
- They must experience responsibility for the outcome of their work
- They must know the actual results of their work activities

They went on to identify 5 dimensions of the work itself that contribute to these sources of motivation:

- **Skill variety**, the degree to which your work requires you to exercise a variety of skills.
- **Task identity**, the degree to which your job requires you to complete a whole, identifiable piece of work. People enjoy completing a "whole" job.
- **Task significance**, the degree to which your work affects other people and contributes to the social welfare.

People need to feel that the final product has value.

- **Autonomy**, the degree to which you have control over the means and methods you use to perform your work – the sense of being your own boss and the amount of “elbow room” you have.
- **Job feedback**, the degree to which carrying out the job itself provides you with direct and clear information about how effective you are.

One key to motivation to control these 5 dimensions is to match up work with the people’s motivations. Robert Zawacki (1993) reported that his 15 years of research shows that about 60 per cent of a developer’s motivation comes from the match-up of the job and the developer (also reference Czikszentmihalyi (1990) earlier in this document).

### **Opportunity to Focus on the Work Itself**

Another motivational aspect of the work itself is the degree to which one can focus on it versus focusing on related concerns.

Procurement of supplies, set-up of PCs, administrative policies, etc., can all be significant demotivators since they take away from being able to focus on the work itself. All policies, procedures, administrative meetings, etc., should be reviewed from a perspective of how can they be eliminated or minimized as much as possible to focus on the work itself.

### **Personal Life**

Achievement, possibility for growth and work itself are in the top 5 for both managers and developers. The greatest deviation is that personal life is fourth for developers and fifteenth for managers. A close second in deviation is responsibility, which is ranked first for managers and tenth for developers. These deviations represent the greatest area of conflict and misunderstanding between developers and their managers.

The way this deviation can be manifested is that a manager may assign a developer to a high profile project which requires long hours, thinking that the extra responsibility is something that the developer desires. To the developer, the extra responsibility is not a top motivator and the impact on the personal life is a punishment.

Policies to honor personal life can be in allowing time off during the day to attend children’s events, giving extra vacation days as bonuses, minimizing after work meetings and social events, etc.

### **Technical Supervision Opportunity**

Managers are less likely than developers to enjoy the technical supervision responsibility. For developers, this is ranked as number five and for managers, the ranking is thirteenth.

This motivator can be enabled via:

- Assigning each person on a project to be the technical lead for a particular part of the project – UI design, database, reporting, etc.
- Assigning responsibility for a particular process area – technical reviews, reuse, system testing, etc.
- Assign all but juniors to be mentors in particular, defined areas. There can also be levels of mentors.

### **Team Performance**

There can be a variation of up to 5 times in team performance, according to the research studies detailed above. Since cohesiveness is viewed as the number one factor in producing this kind of productivity boost, what are the factors that will create a high performance team?

- A shared elevating vision or goal
- A sense of team identity
- A results-driven structure
- Competent team members

- A commitment to the team
- Mutual trust
- Interdependence among the team members
- Effective communication
- A sense of autonomy
- A sense of empowerment
- Small team size
- A high level of enjoyment

Research from Larson and LaFasto (1989) found unusual consistency among the attributes of highly effective teams, regardless of industry and levels of education.

### **How to Manage a High-Performance Team**

A cohesive team forms an “us” and the manager is in the position of not quite being a part of the team.

Keys to establishing a high performing team:

- Establish a shared vision
- Create change - realize that the vision might require change and bring it about positively.
- Manage the team as a team. Make the team responsible for its actions rather than focusing on individuals on the team.
- Delegate tasks in ways that are challenging, clear and supportive.
- Leave details of how to do the task to the team.
- When the team isn't functioning well, think about the motivation, organization and information model Try to remove roadblocks from these three factors.

### **Shared, Elevating Vision or Goal**

A team must “buy in” to a common vision or goal. A common vision builds trust among the team members because they know that they are all working to the same objective. There are a number of organizational development approaches to develop a truly “shared” vision and

Software Consortium can coach clients on the best approach for their environment or team. How this vision is determined is as important as ensuring that there is a shared one.

When team cohesiveness is not present, it is typically due to different visions or lack of a clarifying vision or goal.

It is critical that the vision that the team determines is in alignment with the overall organization.

### ***Challenging Work***

To have a motivating effect, the vision must also be inspiring or elevating. The team must be presented with a challenge, a mission.

### **Sense of Team Identity**

As team members work toward their common vision, they begin to develop a sense of team identity. Team members can name their teams; develop a sense of elite-ness...all things that bond the team to each other and to their inspiring mission.

### **Results-Driven Structure**

Software development teams where date is the overriding goal should be designed to be structured for the maximum development speed.

Four essential characteristics of a results-driven structure:

- Roles must be clear and everyone must be accountable for their work at all times.
- Team must have an effective communication system that supports free flow of information.
- Individual performance must be monitored and there must be a way to provide feedback.
- Decisions must be made based on facts instead of subjective opinions whenever possible.

### Competent Team Members

The three competencies that are important in selecting team members: 1) specific technical skills, 2) a strong desire to contribute and 3) specific collaboration skills to work effectively with others.

Effective teams also have a mix of skills and they play different leadership roles. According to Dr. Meredith Belbin, the following leadership roles are important:

**Driver** – Controls team direction at a detailed, tactical level.

**Coordinator** – controls team direction at the highest strategic level.

**Originator** – provides innovation leadership, especially on major issues.

**Monitor** – Sees problems from a practical view and helps team make right decisions.

**Implementer** – takes ideas and plans and makes them happen efficiently and as agreed.

**Supporter** – Builds on team members strengths and provides emotional leadership that fosters team spirit. Aids in communication among the team.

**Investigator** – explores and reports on ideas outside of the group and develops external relationships important to the group.

**Finisher** – Focuses on the details for the group and creates the sense of urgency necessary to complete the work.

Different people will play different roles at different times. An underperforming group can be rigid about roles and to become high performing, the team must be willing to be focused on its needs.

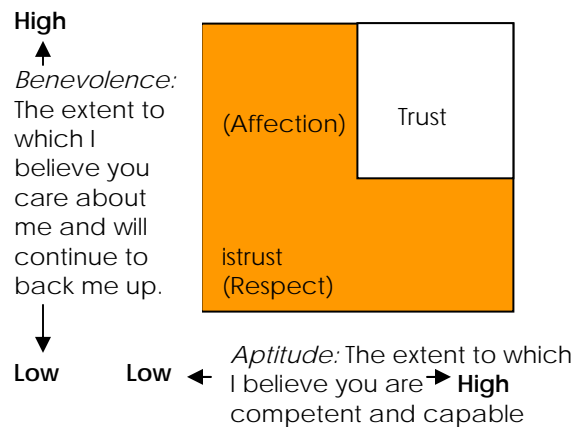
### Commitment to the Team

With vision, challenge and team identity, commitment of the team will follow. The team must commit to the team and put the team first.

If goals are lacking, the team cannot commit to anything.

### Mutual Trust

Larson and Lasto found that trust consists of honesty, openness, consistency and respect. If any of these are breached, trust is broken. Trust is an effect of an effective team – it cannot be mandated. When the starting point between individuals or groups is distrust, there needs to be a thoughtful series of activities that will allow each party to test its willingness to believe in the others competence and caring. When starting to work from an adversarial relationship, begin by doing simple, low-risk activities to build trust. Competency and caring are both required to build trust. Peter Scholtes, author of the classic The Team Handbook, represents this two-way dynamic of trust pictorially below.



### Interdependence Among the Team

Effective teams rely on each other's strengths and will ask others to do things that they can do more effectively. Everybody feels that they have a chance to contribute and their contributions matter.

### Effective Communication

Members of effective, high performing teams stay in touch with each other constantly. The common vision and sense of team identity helps their communication since they have a mutual understanding of what they are

trying to accomplish. Individual agendas fall by the wayside as the team's goals become the first objective.

Team members also express what they are truly feeling, even bad news. Without this openness, bad news will not be shared, leading to the team's inability to meet its goals.

#### **Sense of Autonomy**

Effective teams have a sense that they are free to do whatever is necessary to make the project successful. This sense of autonomy is related to the level of trust that they feel from their manager.

#### **Sense of Empowerment**

An effective team needs support and resources to be effective. It also needs to manage up when there are things that they cannot accomplish.

#### **Small Team Size**

Research from Emery and Emery (1975) and Bayer and Highsmith (1994) says team size should be a maximum of 10 and if the size of the project requires more, teams should be sub-divided into appropriate groups.

#### **High Level of Enjoyment**

Productive teams enjoy a high sense of personal enjoyment and humor.

© 2007, Software Consortium