Motivation von Softwareentwicklern bei Anwendung der PSP (Personal Software Process) und TSP (Team Software Process) Modelle

Bachelor Thesis

University of Rostock
Department of Business, Economics and Social Science
Chair of Business Informatics

Author: André Schütz
Matrikel No.: 001200050
Course of Study: Business Informatics

Term of editing: 3 months

Adviser: Dr. Horst Günther, Chair of Business Informatics
2nd Expert: Prof. Dr.-Ing Hans Röck, Chair of Business Informatics

Rostock, 21 March 2005

Eidesstattliche Versicherung

Ich versichere eidesstattlich durch eigenhändige Unterschrift, dass ich die Arbeit selbständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen[1] genommen sind, habe ich als solche kenntlich gemacht. Ich weiß, dass bei Abgabe einer falschen Versicherung die Prüfung als nicht bestanden zu gelten hat.

Datum Unterschrift

....

[1] Zusätzlich gelesenes Material zur Informationsgewinnung, aus dem aber nicht zitiert wurde: [Bro95], [Car81], [Car95], [Cra00], [Cra01], [Hec74], [Hen01], [Kat03], [Lie01], [Ner03], [Ros00], [Rot], [Sch01], [Cos00], [Env04e], [Gre02], [Gar04], [Gla02], [Gos04], [Guc02], [Hil02], [McC96], [Nag04], [Ret00], [Rus01], [Wal02], [Wie99], [Wik04], [Uni04b], [Uni04a], [Sch03], [Uni04d], [NFP04b], [NFP04a], [Pra04], [Pri01], [Jan04], [Hum04a], [Hum04b], [Hum02], [Web04], [McA00], [Dav03a], [Hum98b], [Hum98c], [Hum98d], [Hum00b]

Contents

List of Figures

Abstract

English

Motivation is the key word of this paper. By it, people can master the biggest problems, accept a lot of pressure, find solutions for so far unsolved problems and it creates a challenging atmosphere. To motivate their people, managers have to accept them as human beings and create a relation to them instead of using them only as an expedient. Only, if companies and their managers understand that people are the most important factor, they will get the best out of them. Therefore, this paper tries to give you an introduction into the basics of motivation and the respective methods.

Deutsch

Motivation von Softwareentwicklern ist das Hauptthema dieser Arbeit. Durch Motivation können Menschen dazu angespornt werden, die schwierigsten Aufgaben zu bewältigen, großem Druck zu widerstehen, Lösungen für bislang ungelöste Probleme zu finden und eine herausfordernde Arbeitsatmosphäre zu kreiren. Um ihre Angestellten bestmöglich zu motivieren, haben Manager sie als menschliche Wesen zu akzeptieren und müssen eine Beziehung zu ihnen aufbauen, anstatt sie nur als Arbeitsmittel anzusehen. Nur wenn Unternehmen und ihre Manager verstehen, dass die Angestellten den wichtigsten Faktor im täglichen Arbeitsprozess darstellen, können sie das Beste von ihnen erwarten. Dazu versucht diese Arbeit einen Einblick in die Grundlagen der Motivation und die damit verbundenen Anwendungsmethoden zu geben.

Summary in German - Zusammenfassung in Deutsch

Motivation ist ein wichtiger Bestandteil heutiger Managementaufgaben. Manager müssen Mittel und Methoden finden um ihre Angestellten möglichst effektiv zu motivieren und sie auf die steigenden Arbeitsaufgaben vorzubereiten.

In dieser Arbeit befasse ich mich im ersten Teil mit allgemeinen Methoden zur Motivation von Menschen und insbesondere von Softwareentwicklern. Im zweiten Teil wird die Motivation in Beziehung zu drei Modellen betrachtet. Diese sind das Process Maturity Modell, der Personal Software Process und der Team Software Process.

0.1 Motivation im Software-Entwicklungsprozess

Im allgemeinen Teil werden u.a. allgemeine Gesichtspunkte der Motivation betrachtet (Teil 2.1), Auswirkungen verschiedener Methoden auf einzelne Individuen und Teams verglichen (Teile 2.2 und 2.3), theoretische Modelle eingeführt und vorgestellt (Teil 2.5), das Verhalten von Führungskräften und Managementteams besprochen (Teile 2.4 und 2.6), der Einfluss von Belohnungen erläutert (Teil 2.7) und zwei Beispiele gegeben, wie Softwarefirmen diese Methoden umsetzen (Teil 2.8).
So umfasst die Motivation (Teil 2.1) die technischen und die zwischenmenschlichen Bereiche des Arbeitslebens. Zu den technischen Bereichen zählt die Bereitstellung von Hardware, Software und den richtigen Arbeitsmaterialien, damit die Entwickler sich rein um ihre Aufgabe kümmern können und nicht durch bürokratische Hürden (z.B. Anträge zur Beschaffung von Materialien) belastet werden oder in ihrem Rhythmus gestört sind.
Die zwischenmenschlichen Bereiche handeln von der Zusammenarbeit untereinander, den einzelnen Träumen und Zielen der Arbeitnehmer und von den individuellen Bedürfnissen, die jeden Menschen einzigartig machen. Diese unterschiedlichen Gesichtspunkte machen es notwendig sich auf jeden einzelnen Mitarbeiter einzustellen und sich um ihn persönlich zu kümmern. Man muss die feinen Unterschiede herausfinden und jedem Mitarbeiter das Gefühl geben, dass er genauso wichtig ist, wie jeder andere in der Firma.

Über die Jahrzehnte hinweg hat das Management verstanden, dass man Menschen nicht wie Maschinen behandeln kann. Es sind Individuen (Teil 2.2), um die man sich im Einzelnen kümmern muss und die, jeder für sich, ganz unterschiedlich auf den Interaktionsprozess reagieren. So muss man den persönlichen Kontakt waren (Teil 2.2.1), die Unterschiede in der Altersstruktur der Mitarbeiter berücksichtigen (Teil 2.2.2), Veränderungen und Veränderungsprozesse im Betrieb behutsam einführen (Teil 2.2.3), jedem Einzelnen den ihm zustehenden Respekt entgegenbringen (Teil 2.2.4) und sich über Arbeitszeiten (Teil 2.2.5) und Arbeitsumgebungen (Teil 2.2.6) Gedanken machen. Selbstverständlich muss man sich auch darüber klar werden das Druck (Teil 2.2.9) nicht das Allheilmittel sein kann, um Entwickler zu besserer Leistung anzuspornen und welche Konsequenzen

(Teil 2.2.8) in den Beziehungen zwischen der Managementebene und den einzelnen Individuen auftreten können, wenn es zu Fehlentscheidungen im Leitungsprozess kommt.
Wenn die Aufgaben komplexer werden, dann können einzelne Individuen sie nicht mehr alleine bewältigen. Dann ist es wichtig schlagkräftige Teams (Teil 2.3) zusammenzustellen, die mit verschiedenen Fähigkeiten ausgestattet sind. Es ist nicht einfach richtige Teams aufzubauen, die ein inspirierendes und motivierendes Teamwork (Teil 2.3.1) kreieren. Man muss die Grösse (Teil 2.3.2) des Teams berücksichtigen und die einzelnen Mitglieder des Teams dazu bringen, sich den anderen Mitgliedern offen (Teil 2.3.3) zu zeigen und sie zu akzeptieren.

Wenn man die Methoden der Motivation wirklich verstehen will, muss man sich auch mit theoretischen Modellen (Teil 2.5) auseinandersetzen, welche die einzelnen Vorgehensweisen in den Individuen untersuchen. Zu diesen gehören unteranderem die Theorie X, Theorie Y, die Hawthorne Experimente, Maslow's Pyramide der Bedürfnisse, die ERG Theorie, das Job-Characteristics-Model und die Mix-Match Methode (Teile 2.5.1 bis 2.5.7).
Nicht nur theoretische Modelle sind wichtig um Auswirkungen auf die Entwickler zu verstehen. Desweiteren gibt es noch verschiedene Formen von Führungsstilen (Teil 2.6), die bewusst oder unbewusst von Führungskräften eingesetzt werden. Darunter zählen u.a. der transformale Führungsstil (Teil 2.6.1), der autokratische (Teil 2.6.4) und der demokratische (Teil 2.6.5) Stil zu den bekanntesten.

Nicht nur zwischenmenschliche Interaktionen können einen Menschen motivieren. Darüberhinaus gibt es auch noch den Weg der Belohnungen (Teil 2.7), die den einzelnen Personen oder Teams zukommen können. Sie stellen eine Art Anreiz dar, den es geht zu erlangen und der dazu führen soll, das die Leistung zum Erreichen der Belohnung erhöht wird.

0.2 PML, PSP und TSP

Das Process Maturity Modell (Teil 4), oder auch Capability Maturity Model, ist entwickelt worden, um Software Firmen in verschiedene Kategorien einzuteilen, die eine Art evolutionäre Entwicklung darstellen. Es gibt 5 Level (Teil 4.1) die unterschiedliche Entwicklungsstände der Softwarefirmen darstellen. Vom ersten bis hin zum fünften Level verbessern die Firmen ihre Methoden zur Softwareentwicklung, ihre sozialen Kompetenzen und ihre Managementmethoden.
Auf der untersten Ebene, also Level 1 (Teil 4.1.1), ist die Handhabung in der Firma noch sehr chaotisch und schlecht organisiert. Hier ist es dringend notwendig, verbesserte Arbeitsbedingungen zu schaffen und klare Strukturen aufzubauen, die den Mitarbeitern ein stärkendes Gefühl für ihre tägliche Arbeit geben. Im zweiten Level (Teil 4.1.2) herrscht noch keine klare Vorgabe, wie Softwareprojekte gehandhabt werden. So werden hier Methoden eingeführt, die helfen sollen, Softwareprojekte mit stets gleichen Qualitätsstandarts hervorzubringen.
Die dritte Stufe, Level 3 (Teil 4.1.3), hat die neuen Prozessmodelle gut umgesetzt und gefestigt.

Die Mitarbeiter werden sicherer in ihrer Arbeit und neue Methoden und Vorgehensweisen werden leichter übernommen. So eine aufstrebende erfolgshungrige Umgebung spornt an und die Entwickler fühlen sich wohl.
Die beiden abschliessenden Level 4 und 5 (Teile 4.1.4 und 4.1.5) sind vorallem durch optimisierende Prozesse gekennzeichnet. Hier haben die Unternehmen schon genug Informationen und Daten aus vorangegangenen Projekten gesammelt, um zukünftige Aufträge optimal abzuwickeln und auftretende Schwachstellen zu identifizieren und zu lösen.

Natürlich werden nicht nur die Prozesse und Methoden in den einzelnen Stufen verbessert. Auch das Zusammenspiel zwischen den Führungsschichten und den Angestellten wird optimiert. Dazu wurde das People - Capability Maturity Model entwickelt, welches die Qualität und Effizienz der Entwickler in den einzelnen Stufen verbessern soll. Darum wird in Teil 4.4 ein Blick auf die einzelnen Stufen geworfen und sich mit Methoden und Vorgehensweisen auseinandergesetzt, die helfen können, Menschen in optimaler Weise zu stimulieren und motivieren.

PSP

Der Personal Software Process (Teil 5) wurde entwickelt, um Entwicklern die Möglichkeit zu geben, ihre persönlichen Fähigkeiten zu verbessern und zu besseren Programmierern zu werden. Dabei haben sie in ein persönliches Notizbuch (Teil 5.2) zu schreiben, in dem ihre Fortschritte während eines Projektes festgehalten werden und das es ihnen ermöglicht Daten über die eigene Arbeitsweise zu sammeln. Dadurch erhält man Informationen über Probleme während des Projektes und welche Zeitaufwände man bei den unterschiedlichen Abschnitten hatte.
Diese Daten ermöglichen es dann, Schwachstellen in der eigenen Arbeit aufzudecken und die eigene Arbeitsweise besser kennen zu lernen.

Im PSP werden meist allgemeine Motivationsmethoden angewandt, wie sie schon im ersten Teil (siehe auch Abschnitt 2) dieser Arbeit angesprochen wurden.

TSP

Aufbauend auf den Personal Software Process wurde der Team Software Process (Teil 6) entwickelt. Er ermöglicht Unternehmen Software Projekte optimal in Teams zu bearbeiten und die Teamarbeit zu verbessern. Hier werden neben den allgemeinen Motivationsgesichtspunkten für Teams aus Abschnitt 2.3 auch spezielle Motivationsverfahren eingeführt, die für TSP Teams relevant sind.

Ein Anhaltspunkt ist der Aufbau eines erfolgreichen TSP Teams (Teil 6.4). Hier werden Punkte

wie der Team Zusammenhalt, herausfordernde Ziele, Rücksprachen und eine gemeinsame Arbeitsumgebung und -grundlage erläutert (Teile 6.4.1 bis 6.4.4).
Der TSP besteht aus 8 aufeinander aufbauenden Stufen (Teil 6.5). Diese sind der Einführungsprozess, die Entwicklungsstrategie, der Entwicklungsplan, der Anforderungsprozess, der Entwurfsprozess, der Durchführungsprozess, der Testplan und das Postmortem (Teile 6.5.1 bis 6.5.8).
Hier gibt es verschhiedene Methoden und Mittel die Mitarbeiter je nach Entwicklungsstand der Unternehmung zu motivieren. Die Manager müssen sich über den eigenen Stand der Firma im klaren sein (siehe auch das PML Modell aus Teil 4) und dann die richtigen Wege finden ihre Untergebenen für die bevorstehenden Aufgaben vorzubereiten.

Das TSP Modell stuft die Entwickler in den TSP Teams in verschiedene Typen ein. So gibt es Teamführer, Entwicklungsmanager, Planungsmanager, Qualitäts- und Prozess Manager und Unterstützungsmanager. Diese verschiedenen Mitarbeitergruppen leisten einen unterschiedlichen Beitrag zum Team Erfolg und können je nach ihrer Position motivierend auf die restlichen Teammitglieder wirken.

Um die Vorzüge und Stärken eines Teams besser ausnutzen zu können, muss man sich weitergehend mit Teams beschäftigen. Dazu können Manager sich die schon erwähnten Methoden aus Teil 2.3 zu Nutze machen oder zusätzliche Merkmale von TSP Teams berücksichtigen, die ab Teil 6.7 erwähnt werden. Dazu gehören eine verbesserte Kommunikation untereinander, Verpflichtungen zwischen den Teammitgliedern, die verschiedenen Mitarbeiterrollen und die Unterstützung jedes Teammitgliedes.

1 Introduction

The motivation of technical people is an important point in todays management function. Through the decades, the managers learned that people are not like machines and that it is possible to increase their morale, engagement and motivation by working in a close relation with them instead of separated from them.
Motivation is a large field that can be looked at in a psychological view which is more theoretic and constructed on models, or like in this paper in a project-management view that tries to give practical examples and describes the real working process.

This paper is subdivided into two main parts. In part one, the motivation is described in general. This part contains partially psychological models which give a theoretical background. This is important, if we want to understand the reactions and feelings which occur in human beings in different situations. Furthermore, there are descriptions of the individuals who are working for the companies and the teams they form. It is important to separate the individual and the teams. Even if many rules are working for individuals and for teams, it is necessary to handle them separately. Every good manager should not only see the whole team. It is the essential point to understand the individual in the team and to see his dreams, feelings, aims and motivational factors. If the manager can break the ice and creates a relation to his subordinates, he can lead them in a better way and in the end he can form a stronger and more successful team.
The end of part one describes the management level and the methods which are available to motivate the professionals. The managers can influence the productivity of their people in various ways. Different situations and people require different methods and approaches. The managers should be aware of their possibilities to get the best out of their subordinates.

Part two is about the motivation in PML (Process Maturity Levels), PSP (Personal Software Process) and the TSP (Team Software Process) models. It starts with the PML model that gives the basis for the following two (PSP and TSP). The PML model separates the development of a company into 5 phases. These can be seen as evolutionary steps and the higher a company is arranged in this model, the better are its capabilities.
PSP and the TSP are ongoing models which describe the individual and the teams in the software development and try to give advice how to improve their skills and project results.

To put it in a nutshell, I want to quote the following sentences.

> "Motivation, obviously, makes the place run because it makes the people run. Highly motivated people drive themselves to overachieve, while many of their brighter and more capable associates accomplish far less. The basic reason is the difference in their motivation." [Hum99b, page 63]

2 Motivation in the software development process

The following sections deal with the concept of the motivation in general. If you want to talk about motivation of engineers, you have to separate different elements and look at every single of them. These are the individuals (see 2.2) who work in a company and form the basis of the organization. Moreover, these individuals will form teams (see 2.3) which make it possible to distribute bigger projects and get the increasing amount of work done.

The other part of an organization is the management level (see 2.4) that leads the manpower and provides them with all necessary technical and nontechnical material for their work. The managers have to motivate their subordinates and support them in doing the best they can do.

To get the best out of the employees, the management level can use several theoretical methods and models, different forms of leadership and special rewards for reached aims. The theoretical approaches (see 2.5) give the background knowledge for the managers to understand the behavior of their people and to react suitable in different situations of their cooperation. The different forms of the leadership (see 2.6) depend on the type of the project, the relation of the manager to the team and the personality of the manager. Sometimes, another style of leadership can decide about success or failure of a project.
The rewards (see 2.7) which a manager can assign to his subordinates are another method of increasing the motivation and satisfaction. But like with all other aspects of this section, it can motivate people or, when the worst comes to the worst, unmotivate them. The right mixture decides about the atmosphere in an organization and how successful the management level could make a contribution to it.

Every manager should understand what motivated employees will mean for his work and for the company. Once, James MacGregor Burns hit the nail squarely on the head as he said the following about the motivation of people:

> "People need appreciation, recognition and a feeling of accomplishment, and the confidence that people who are important to them believe in them" [Hum99b, page 71]

2.1 Motivation

"It is not as important to have the right ideas as it is to have the drive and motivation to make your ideas right" [Hum99b, page 64]

Motivation is a complex matter. It is not only depending on the management to motivate their employees. It is also depending in the employee himself and how he's in relation to the task, the environment where he works and his workmates.
In motivating engineers, there are technical (like Hardware, Software, Information, etc.) and nontechnical (dreams, aims, feelings) elements which have to be taken into account. "The key point, however, is that almost all the nontechnical elements of employee motivation are directly controllable by the professional's immediate manager." [Hum99b, page 65]

> "While most people behave in reasonably predictable ways, software people are unique, both in their creative abilities and in the nature of the work they do." [Hum03]

2.1.1 The evolution of motivation in the management process

In 1911, Frederick Winslow Taylor published his book *Principles of Scientific Management* where he proposed that the work should be sub-divided into smaller parts. His theory managed human beings like machines and assumed that people would not have feelings, aims or a motivation. His model has been called 'Theory X' (see 2.5.1).
In 1924, Elton Mayo made his famous studies at the Western Electric's Hawthorne plants (see 2.5.3). His experiments were revolutionary. Now the management realized that it was possible to improve the productivity when they payed attention to their workers and maintained the communication with them. This was the first time that leaders understood that it was important to deal with their people in a fair and motivational way.
Douglas McGregor found a new approach in 1960. His famous theory was called 'Theory Y' (see 2.5.2) and said "people are psychologically motivated to work, and management should help and support them rather than coerce them." [Hum99b, page 66/67] This model is the core of many modern opinions.
In our modern time, the motivation of people is a very important and profitable matter.

2.1.2 The motivation of programmers by Watts S. Humphrey

Watts S. Humphrey developed the Capability Maturity Model[2] (see section 4), the Personal Software Process (see section 5) and the Team Software Process (see section 6). These models deal with the improvement of software processes and how engineers can improve their productivity. The following statements are discoveries of Humphrey about the motivation of software developers.

[2] also known as PML (Process Maturity Modell)

"If the programmers do not understand the job they are to do, they will not do it very well." [Hum03] The management has to provide their professionals with the required materials and knowledge which they'll need to complete their work.

"Superior software work is done by highly motivated developers." [Hum03] Not only the single individual has to be motivated. The manager has to motivate the individuals and the whole team and give them all the support that is necessary.

Some factors that motivate people are [Hum03]:

- "A personal sense of being involved and making a contribution"
- "Frequent celebrations where the team and management complemented them on their achievements and milestones"
- "Positive feedback from marketing and senior management"
- "The autonomy to do the job the way that they thought was best"

2.1.3 Pride counts more than anything

You should never underestimate the power of pride for the motivation of developers. "Because engineers and scientists have professional pride, most are anxious to do a good job even if it is not exciting." [Hum99b, page 7]
A nice example is a product manager who was not all right with his engineers. The engineers had to create a new printer for a replacement project. The developer team had problems to realize the expected features and wanted to give up. He told them that they should finish the project and create a better work than they had done up to now. Furthermore, they should convince the financial people of their work. Thereafter, the engineers went back to work and finished the project. What was the reason for this surprising change?
The product manager used a simple trick, "he did not threaten, offer reward, or asserted authority. He did not even appeal to his employees' loyalty or suggested a better way to do the job. What he did was to point out that they had not tried as hard as they might. [..] Dedicated professionals feel ashamed when it is clear they have failed to perform as they know they should, and a challenge to try again can be a great stimulus. [..] Managers should guide , support, and help, but above all, they must not let their people quit to soon." [Hum99b, page 7]

2.1.4 You have to remind them

Like in the previous section 2.1.3, it is sometimes important to remind your engineers of the importance of their work. That lies in the nature of human beings that "most people need periodic encouragement[3] to do their best. They need to be charged up and reminded that the goal is important and achievable." [Hum99b, page 7]

In reality, it is surprising that technical people mostly always find a way to solve a problem after their boss has provoked them. If you want to compare that with sports, there is a nice phrase by James J. Corbett[4]:

> "Fight one more round. When your feet are so tired that you have to shuffle back to the center of the ring, fight one more round. When your arms are so tired that you can hardly lift your hands to come on guard, fight one more round. When your nose is bleeding and your eyes are black and your are so tired you wish your opponent would crack you one in the jaw and put you to sleep, fight one more round - remembering that the man who always fights one more round is never whipped." [Hum99b, page 10]

2.1.5 Setting specific goals

Goals are very important to motivate the professionals. Goals have a very effective impact on engineers. They let people understand their work and the aims of the company. Furthermore, the professionals try to achieve the goals and work harder on them, if they accept them. Like mentioned in section 2.2.3 about changes, subgoals can help to gain the agreement of the workers. They are easier to motivate for smaller goals and steps than for one big goal that seems to be unreachable in the beginning. If professionals reach these smaller goals, they are much more motivated to reach the overall goal.

> "The art of motivation is not in giving orders, but strong objectives." [Klu04]

2.1.6 Let them work

Programmers can only be good programmers if they can work on the code and think about the problem. Therefore, the manager should let the programmer work on his tasks and should minimize the number of meetings where the programmer has to participate. Rewards should have something to do with their work. For example a programmers conference or a work shop about new techniques.

Programmers have another rhythm. They like to experiment and if the work is very difficult, they sometimes need a little break to think about it. "The real key is to let them decide when they get away from the screen." [Moo01]

[3] "Developers who give up too easily will never complete a product, and their manager must sense this and urge them to try harder." [Hum99b, page 7]

[4] Championship fighter in boxing

2.2 Individuals

Individuals are the basis of every company. If you want to get highly motivated and dedicated workers, you should not only work with them, you should rather respond to them and try to create an atmosphere of fair partnership.

2.2.1 Hold the contact

Engineers need the contact to their managers. If they don't get enough attention, they will wonder whether they are no longer important for the company or whether they got a bad name. If so, they will think about the whole situation and their work will suffer. And in the end, they will think of problems which not exist, but which will prevent them from working and destroy their creativity.
To avoid these self-made situations, the leaders should have daily contact with their subordinates. During these short get-togethers, they should on the one hand talk about the engineers work and on the other hand about their fears, dreams and personal goals.

2.2.2 See the change

Professionals learn from their mistakes and can become better during the years of their working experience. Mistakes are important to learn new methods and to improve individual skills. Furthermore, these successes create self-confidence and motivate to make new experiences.
As first, the management has to view every professional as an individual who has different aims and characteristic peculiarities. In a next step, it could be useful to see carrier stages which lead to an evolutionary process of the single professionals.

20s and early 30s: During these years, great achievements are very important for professionals. But they first have to learn all the required skills. This can be very disappointing when the first jobs are not so exciting and interesting and they have to do a lot of basic work for the other professionals. If the management supports these young professionals in a way that they gain more self-confidence, they can improve their skills and learn to manage things on their own.

the 30s: Now, the professionals have no longer doubt about their own skills and experiences. They could gain a lot of experience and make their own mistakes which were a good lesson to improve their own abilities.
The professionals see their own achievements and think about their position. If they could not achieve the position they dreamed of, they try to improve their situation and make a better career. In their situation, they need "a sympathetic and helpful manager who can assist them to reassess their own ambitions." [Hum99b, page 53]

the 40s and 50s: The professionals accept what they are and what they reached. "While they may still think fondly of what might have been, they find comfort in their position as respected elders." [Hum99b, page 53] They like to help the younger professionals and express their own thoughts freely. The managers should learn to hear what they say and accept their opinion.

2.2.3 Why do individuals fight against management changes

Normally, individuals like to improve things and see the effect of these changes. But in reality the most engineers are against these changes and will not accept them.
Mostly, the management staff is responsible for this situation. They make some important mistakes in dealing with their subordinates. Firstly, the management level should include the engineers in the beginning of the change process. They should inform them about the most important facts and rearrangements[5]. Moreover, if there are decisions to make, the management team should include the subordinates and ask them for their opinion. That avoids misunderstandings from the beginning of the process and enlightens the workers.
Another important method is to divide the change process into smaller single steps. So the subordinate is sooner willing to accept the change because every small step is easier to handle and understand than one big step.

There is a famous change formula that expresses the probability that a change will be a success.

$$C = AxBxD > X \text{ [Boy99, page 73]}$$

where:

C = the probability that a change will be a success

A = the dissatisfaction with the current situation

B = a clear statement of the expected condition after the change

D = the concrete first step

X = the expenditure of the change

"The formula says, if you want to win the organization and the employees for the change, (A) the professionals have to be convinced that the change is necessary, (B) construct a picture of the better condition after the change and (D) show that you know what you do while you get some successes in the beginning of the change process." [Boy99, page 74] The may be most important factor is A. Like already mentioned before, the management should include the professionals in the change process and motivate them by involving them in the decisions.

[5] Douglas Sherwin: "Change is great when you are its agent; it is only bad when you are its object."[Hum99b, page 261]

2.2.4 Respect the individual

You have to respect your engineers and their work if you want to get motivated and informed professionals. Such employees won't give up if there are problems or insoluble tasks. They are willing to do a thorough job and won't start back if there's a high pressure. They will try to find a solution no matter how long it takes or how much work that would mean.[6]
The management level has to respect their employees. That means to respect the employees need for personal values. Employees have own dreams and personal aims which they want to reach. Furthermore, the employee has to be sure about his position in the company and should not have personal concerns about his job or his work.
To get such professionals, the management has to create a familiar atmosphere where the employees trust them. "If managers do not demonstrably respect their people, these people will not trust their managers. [..] This means that workers should have clearly defined goals and a manager who respects their role in achieving them." [Hum99b, page 34]
This sign of respect has to be in every single decision in the company. Therefore, the company has to treat every engineer as an individual and pay attention to his personal dreams and needs (see 2.7). If the employees realize that their boss tries to get the best for them, they will be motivated to do their best for the company.

2.2.5 Working hours

Programmers have another working rhythm than the normal office employee. "Many programmers do their best work between 10 p.m. and 4 a.m. so if you force them to come in from 9 a.m. to 6 p.m. you'll ruin the art." [Moo01] Give them the opportunity to decide when to come and when to go. The only rule should be that they have to fulfill their task but how they do that should be up to them.

2.2.6 Create an inspiring environment

Because of the fact that programmers are like artists, they need an environment that inspires them and where they feel good. "Microsoft gives its software artists an office with a door." [Moo01] There are no cubes where the programmers are encircled. Like artists, programmers have to create their own environment where they like to work and where it is like home.

> "Offices work; cubicles don't. Forget the trends and provide a working environment that maximizes privacy and allows for uninterrupted time to accomplish goals. Cubicles do not inspire an atmosphere of open communication; leadership does." [Whi94]

[6] H.L. Mencken: "For every complex question there is a simple answer, and it is wrong." [Hum99b, page 33]

2.2.7 Inform them as good as possible

Information is a powerful motivator that the management can use. If the professionals are perfectly informed about the conditions of the company, the project tasks, their efficiency and the future changes, they are much more willing to accept and support changes and feel considered by the management.

> "If you really want to improve the performance of the professionals, Gilbert wrote, you should improve the flow of informations." [Boy99, page 266]

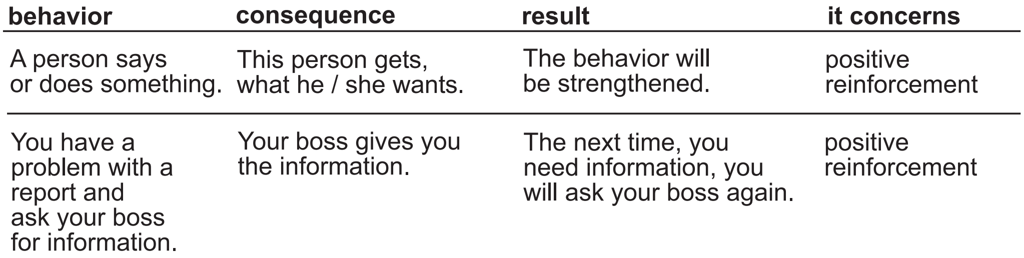
After Gilbert, better information is the best factor to improve the competence and motivation of the professionals. So, uninformed professionals are an important reason for most of the problems in a company.

2.2.8 Typification of consequences

There are four types of consequences which can occur when managers react in different situations. Two of them will rise the probability that professionals will repeat a behavior and the other two will reduce this probability.

consequence 1: The positive reinforcement

This consequence will strengthen the expected probability. (see figure 1 [Boy99, page 287])



behavior	consequence	result	it concerns
A person says or does something.	This person gets, what he / she wants.	The behavior will be strengthened.	positive reinforcement
You have a problem with a report and ask your boss for information.	Your boss gives you the information.	The next time, you need information, you will ask your boss again.	positive reinforcement

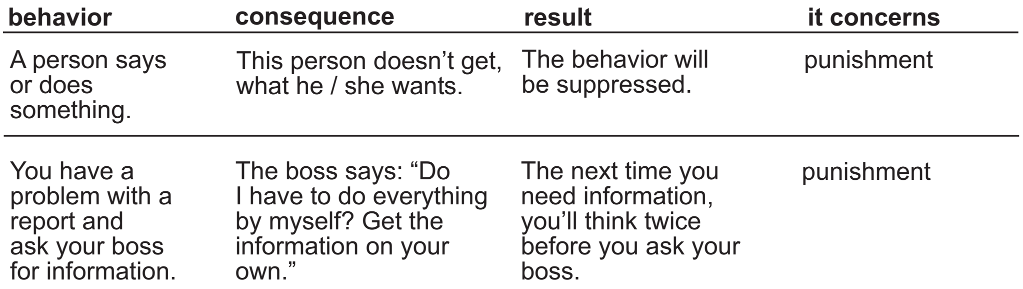
Figure 1: Positive Reinforcement

consequence 2: The punishment

This consequence will lead to the fact, that the professional won't repeat the undesirable behavior in the future. "Here the unpleasant consequences will be used." [Boy99, page 290] (see figure 2 [Boy99, page 288])

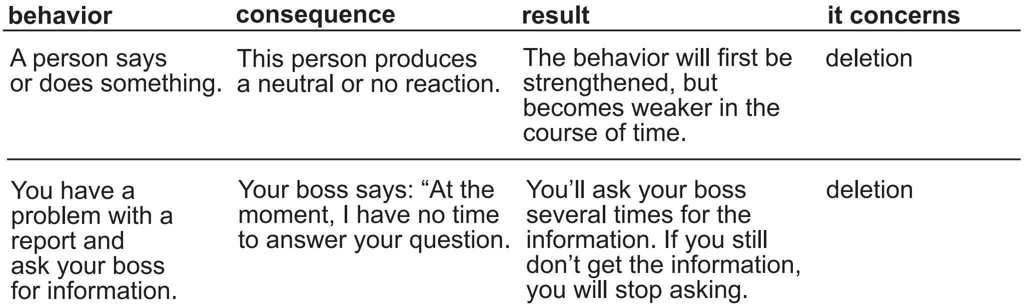
consequence 3: The deletion

This consequence will prevent a professional to repeat an undesirable behavior. This consequence is an alternative to the punishment. (see figure 3 [Boy99, page 288])



behavior	consequence	result	it concerns
A person says or does something.	This person doesn't get, what he / she wants.	The behavior will be suppressed.	punishment
You have a problem with a report and ask your boss for information.	The boss says: "Do I have to do everything by myself? Get the information on your own."	The next time you need information, you'll think twice before you ask your boss.	punishment

Figure 2: Punishment

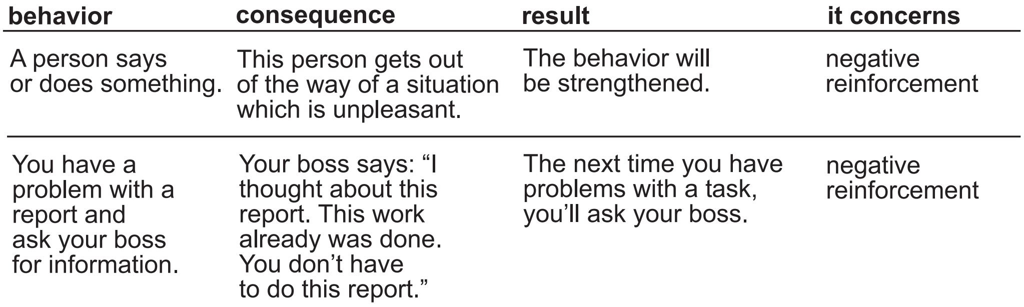


behavior	consequence	result	it concerns
A person says or does something.	This person produces a neutral or no reaction.	The behavior will first be strengthened, but becomes weaker in the course of time.	deletion
You have a problem with a report and ask your boss for information.	Your boss says: "At the moment, I have no time to answer your question.	You'll ask your boss several times for the information. If you still don't get the information, you will stop asking.	deletion

Figure 3: Deletion

consequence 4: The negative reinforcement

This fourth consequence is a little bit confusing. If a professional won't do a task, different results can happen. People who are confronted with such consequences will only do what is necessary to fulfill the task. The management will get what they want - but not more. (see figure 4 [Boy99, page 289])



behavior	consequence	result	it concerns
A person says or does something.	This person gets out of the way of a situation which is unpleasant.	The behavior will be strengthened.	negative reinforcement
You have a problem with a report and ask your boss for information.	Your boss says: "I thought about this report. This work already was done. You don't have to do this report."	The next time you have problems with a task, you'll ask your boss.	negative reinforcement

Figure 4: Negative Reinforcement

2.2.9 Pressure blocks up thinking

Very often, managers use pressure to increase the productivity of their sub-ordinates. But this can have the the opposite results. People who are under pressure are not automatically faster or dedicated.
The productivity of people can be increased, but only, in a very limited way. Like you can see in figure 5 [DeM98, page 164], DeMarco creates a curve which compares the pressure to the time. If managers increase the pressure, people will increase their productivity. But this intensification will not be more than six per cent. You can come to the conclusion that a little bit of pressure will have a little effect on the productivity. More pressure will stop any effect and too much pressure will have the opposite effect to the productivity. The quality and the productivity will decrease and the product will lose its possibility to be finished in time.

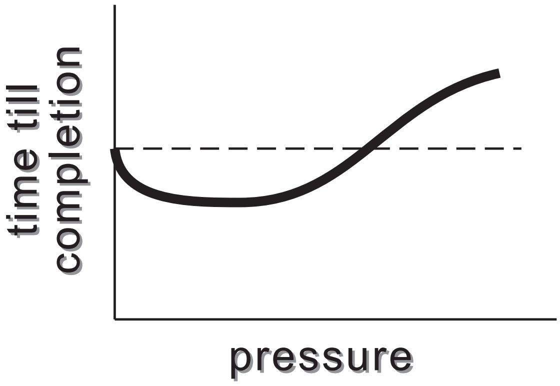


Figure 5: Relation between pressure and work

Therefore:

> "People, who are under pressure, don't think faster." [DeM98, page 170]

Managers should keep this in mind. More work and pressure for a short period of time can increase the productivity of their professionals. It concentrates the knowledge and the efforts of the professionals and creates a feeling of importance. But pressure over a longer period of time will decrease the concentration and the professionals will be unmotivated.

2.3 Teams

Teams are another important factor the management has to pay attention to. They are necessary to fulfill the projects which are much more gigantic and complex in todays software development.

The modern opinions about software teams have specific characteristics of how effective teams should act and be. These conditions were set up by professionals like Cummings, DeMarco, Dyer, Katzenbach, Mohrman, Shaw and Stevens. They are as follows [Hum00c]:

- "The team members establish common goals and defined roles."
- "The team develops an agreed-upon strategy."
- "The team members define a common process for their work."
- "All team members participate in producing the plan, and each member knows his or her personal role in that plan."
- "The team negotiates the plan with management."
- "Management reviews and accepts the negotiated plan."
- "The team members do the job in the way that they have planned to do it."
- "The team members communicate freely and often."
- "The team forms a cohesive group: the members cooperate, and they are all committed to meeting the goal."
- "The engineers know their status, get feedback on their work, and have leadership that sustains their motivation."

If projects become more and more complex, teams are the only answer to master the amount of work. The intensified amount of work needs additional skills. Therefore, "in any situation requiring the real-time combination of multiple skills, experiences, and judgments, a team inevitably gets better results than a collection of individuals operating within confined job roles and responsibilities." [Kat99, page 15]

2.3.1 Teamwork is the key

The biggest advantage of teams is the association of different skills and experiences. Therefore, the single team members can support each other and provide help if necessary. "Teamwork represents a set of values that encourages behaviors such as listening and constructively responding to points of view expressed by others, giving others the benefit of the doubt, providing support to those who need it, and recognizing the interests and achievements of others. When practiced, such values help all of us communicate and work more effectively with one another and, therefore, are good and valuable behaviors." [Kat99, page 21]
The management should support all attempts to create an atmosphere where teamwork will arise within the team. This will lead to contented team members and create a motivated and effective team.

2.3.2 Define the size of the team

The size of team is mostly preferred to be 'small'. 'Small' is a very vague value. Some authors say 5, 8 or 12 members are the best size for a team. "A larger number of people, say fifty or more, can theoretically become a team. But groups of such size more likely will break into sub-teams rather than function as a single team." [Kat99, page 45]
The management should break the project tasks into defined sub-tasks and assign them to specified teams which can master the work. It is better to create such sub-teams manually than to give all the work to overpopulated teams. Such sub-teams have an own working atmosphere and the members are much more constructive than single individuals under these circumstances. The members of such created teams will be motivated by the other individuals and the work will have a much higher quality.

2.3.3 Support the members to be open

In the beginning of the team building process, the most members are shy and adopt a wait-and-see-policy. Especially in new teams, the manager should try to create an atmosphere of trust and understanding. The individuals have to feel good. Katzenbach describes it as follows:

> "If people in the group, for example, are alert to a shy person's initial efforts to speak up and contribute, they can give him or her the positive reinforcement that encourages continued contributions. Similarly, when someone risks opening up a sensitive, conflict-ridden issue, the others on the team and especially the leader can use positive feedback to powerfully signal their openness to further such challenges." [Kat99, page 126]

2.4 Leaders and the management team

The management team of an organization has to fulfill several tasks to support the employees in the best possible way. If the management team appears motivated and engaged, it will be possible to gain the attention of the employees and to motivate them in the expected direction.

Therefore, the management has to provide the employees with the required equipment, information and assistance. Additionally, it has to determine the enterprise goals and adjust them with the personal aims of the employees to promote them in the best possible way. But not only the material part of the collaboration is important. Also the interpersonal part is of big relevance. This means that "leaders must also care about their followers. Caring is more than just practicing what the personal manual preaches; it is thinking about the people, their needs, and their aspirations." [Hum99b, page 5] The best examples were people like Napoleon who "knows the names of his gunners or a Tom Watson remembers the wives and children of his factory employees." [Hum99b, page 5] If the management level can reach the employees directly it will be possible to motivate them in various ways and get the best out of every single person.

2.4.1 Open Door policy

Like section 2.2.4 mentioned is respect a very important factor. But the respect has to start at the top of the organization and should be followed in every separate level of the hierarchy. That's why the management has to be fair to their employees and the Open Door policy can help to achieve that.
This model practices "the right to communicate with senior management through a channel that is independent of the immediate supervisor." [Hum99b, page 36] So the employees have the right to speak to every single person of the company, even a talk to the chairman of the board would be possible. As result, the employees can be sure that their personal meaning and views will 100% be heard. That creates an atmosphere of safety and acknowledgement where the employees can concentrate themselves on their work. Nevertheless, the company has to react on every single complaint to create an atmosphere of trust and consideration.
Here are some of the guidelines which IBM follows in their Open Door policy [Hum99b, page 37]:

1. "Individual employees who have a problem are encouraged to first resolve it with their (respective) immediate managers. If they cannot do so to their satisfaction or if they do not want to discuss it with their managers, they may go to higher management. [..] ... or go all the way to the chairman of the board."

2. "In the case of appeals to senior executives, an impartial investigator is promptly assigned who is relatively senior manager and is organizationally separated from the employee. [..]"

3. "The investigator contacts the employee within 24 hours of being assigned and meets with the employee before talking with anyone else. This ensures that the investigator starts with

an open mind. [..]"

4. "... the investigator explains the Open Door process, listens to the employee's concerns, and asks questions about the issues and the people involved. [..]"
5. "The investigation is kept confidential and strictly confined to the issues raised. I, however, the investigator finds other topics that should be addressed, he or she can expand the investigation at his or her discretion or initiate a separate investigation."
6. "... the investigator prepares a final report, then reviews his or her findings, including the recommendations, with the employee."
7. "The investigator reviews this report with the executive who received the original complaint. The executive makes the final decision and writes a letter to the employee thanking him or her for using the Open Door and summarizing the conclusions."
8. "Management must never take any action that could appear as retaliation for an employee's Open Door appeal, and all records of the appeal are kept in a separate file that is not available to line management. This file is retained for a maximum of three years. In no case is any mention of the Open Door or any material relating to it put in the employee's personnel file."

To create a trustful environment where the Open Door policy works, the organization should follow some of the next suggestions:

- Explain the reason for a necessary decision so that the employees know why it had to be.
- The management should be interested in their employees and show that to them.
- It should be possible for the employees to explain their concerns in several ways.
- Thank the employees that they used the Open Door policy and for their trust.

2.4.2 What leaders should do

Not every person is the born leader. Leadership can be learned and the personal skills to be a good leader can be improved. The manager has to motivate his team and provide the team with all necessary materials which are important to fulfill the work and create a pleasant atmosphere. "Some of these leaders are good motivators, but some couldn't motivate a dog to chase cats. Some are excellent organizers, but some can't find a matching pair of socks in the morning. *All* of the most consistently successful technical leaders empower people by the value they place on innovation, on doing things in a better way." [Wei86, page 21]

There are three basic characteristics which are helpful for managers to emphasize innovation. These are [Wei86, page 21]:

1. "understanding the problem"

2. "managing the flow of ideas"

3. "maintaining quality"

If a leader has to criticize an idea of his professionals, he should pay attention to the fact that people don't like to be criticized. The manager has to make clear that he only criticizes the idea and not the person.

2.4.3 Managers have to care about their people

There are several processes in a company that can directly be influenced by the management to improve the motivation and feelings of their professionals.[7]

> "Leaders who don't care about people don't have anyone to lead, unless their followers don't have a choice" [Wei86, page 123]

The managers should offer several choices to the employees. That creates an atmosphere of trust where the professional feels free to decide about his own work and the way he solves problems. Furthermore, the professional feels important and realizes the consideration of his manager.

> "When survival is concerned, there's no choice but to put people first." [Wei86, page 122]

The management should confirm the professionals in their work. The professionals shouldn't be afraid to make mistakes or to change a method. "People in fear for their lives can't do any task well, except a task devoted to their personal safety." [Wei86, page 122]

> "To be a successful problem-solving leader, you must keep everybody's humanness at the forefront." [Wei86, page 126]

The manager should always see the people and not the task as the most important factor of the company. Behind every task is a person who solved many problems to fulfill the task. "So, if you deny the human reality behind your work, you'll never be a very successful problem-solving leader." [Wei86, page 126]

And finally, the manager should always follow one simple rule if he wants to be a successful motivator.

> "If you are a leader, the people *are* your work. There's no other work worth doing." [Wei86, page 127]

[7] "Without care and affection, you cannot make people act differently as before. To achieve a behavioral change, you have to understand from where they are and why they are how they are." [DeM98, page 157]

2.5 Theoretical models

The most motivational theories are based on specific models. The following sections will give an overview of the most important models.

2.5.1 Theory X

There are two theories which are dissimilar except the fact that "each theory assumes that management's role is to organize resources, including people, to best benefit the company". [Env04a]
These theories are called 'Theory X' and 'Theory Y'.
The assumptions of Theory X are:

- People don't like to work and they will attempt to avoid work whenever possible.
- The people are not ambitious and don't want to get responsibility.
- "Motivation occurs only at the physiological and security levels of Maslow's Needs Hierarchy" [Env04a]
- The management has to control the employees to reach the aims of the company.

The theory assumes that the primary motivation of the employees is money. To motivate the employees, the managers have to create an "environment of command and control" or an environment that seeks "harmony in the hopes that, in return, employees will cooperate when asked." [Env04a]

The problem with Theory X is that it only satisfies the lower level needs (see 2.5.4 about Maslow's Hierarchy of Needs) and if they are fulfilled, the managers get problems to motivate their employees because "people use work to satisfy their lower needs, and seek to satisfy their higher needs during their leisure time. Unfortunately, employees can be most productive when their work goals align with their higher level needs." [Env04a]

2.5.2 Theory Y

In contrast to Theory X, Theory Y has the following assumptions:

- "People will be self-directed and creative to meet their work and organizational objectives if they are committed to them." [Env04a]
- People can handle responsibility.

"Under these assumptions, there is an opportunity to align personal goals with organizational goals by using the employee's own need for fulfillment as the motivator." [Env04a]
The organization can intensify the motivation by consulting the employees in the decision making process, delegate responsibility and decision making to them and help the employee to reach his own objectives in connection with the objectives of the company.

2.5.3 Hawthorne Studies

The Hawthorne Study analyzed productivity and work conditions of employees in relation to "physical and environmental influences of the workplace" [Env04d]. As result, the scientists found out that people work more effectively and engaged when they felt important. Additionally, the individuals increased their work intensity when they got less pressure.

If a manager provides additional courses in programming skills, the programmer will work harder and more effectively, because "you've given [..]" him "the feeling that" he or "she is so valuable to the organization that you'll spend time and money to develop" his or "her skills" [Env04d]
"This effect has been described as the reward you reap when you pay attention to people. The mere act of showing people that you're concerned about them usually spurs them to better job performance." [Env04d]

2.5.4 Maslow's Hierarchy of Needs

Abraham Maslow created a system of hierarchical needs which are ordered in pyramidical form. The theory says that people are motivated by their unsatisfied needs. First, people have to satisfy their lower needs before they can satisfy the higher needs.

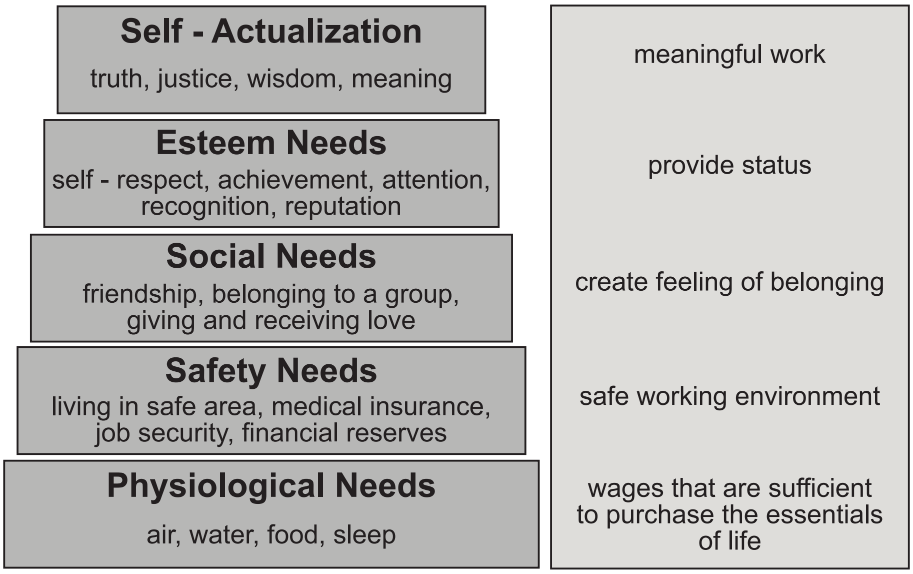

Self - Actualization
truth, justice, wisdom, meaning
Esteem Needs
self - respect, achievement, attention, recognition, reputation
Social Needs
friendship, belonging to a group, giving and receiving love
Safety Needs
living in safe area, medical insurance, job security, financial reserves
Physiological Needs
air, water, food, sleep
meaningful work
provide status
create feeling of belonging
safe working environment
wages that are sufficient to purchase the essentials of life

Figure 6: Maslow's Hierarchy of Needs

The various people in an organization have different needs and they are differently important for them. So the managers have to find out the needs of their engineers. "When a need is satisfied it no longer motivates and the next higher need takes place" [Env04c]. So it will be possible to motivate the employees if the managers pay attention to the different levels of the pyramid and the different expectations of their employees.

2.5.5 ERG Theory

The ERG theory reduces the number of levels in Maslow's Hierarchy of Needs to three (see figure 7 [Env04b]). Studies have shown that the middle levels overlap and therefore, the levels can be deleted. The letters of the theory are explained in the following way:

Existence: "Existence refers to occur with basic material existence requirements." [Env04b]

Relatedness: "Relatedness refers to the desire we have for maintaining interpersonal relationships" [Env04b]

Growth: "Growth refers to an intrinsic desire for personal development." [Env04b]

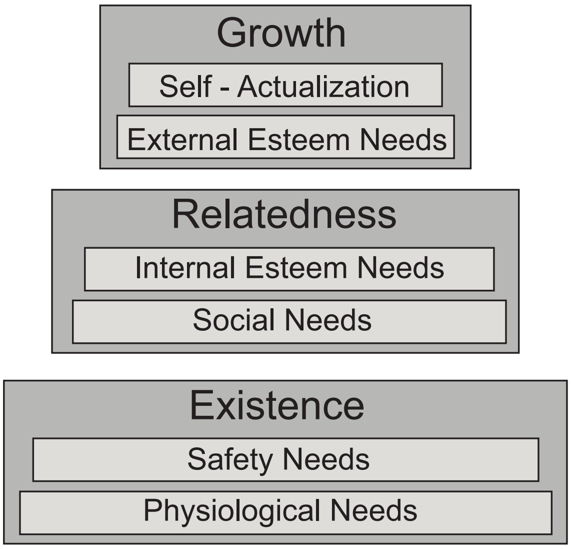


Figure 7: ERG Theory

Like in the pyramid of Maslow, the ERG model creates different levels which are one on top of the other. "Existence needs motivate at a more fundamental level than relatedness needs, which, in turn supersedes growth needs." [Env04b]

The difference to the model of Maslow is that "more than one need can be motivational at the same time. A lower motivation not be substantially satisfied before one can move onto higher motivators." [Env04b] Additionally, the "order of needs can be different for different people." [Env04b]

The model makes clear that "managers must recognize that an employee has multiple needs to satisfy simultaneously. According to the ERG theory, focusing exclusively on one need at a time will not effectively motivate." [Env04b]

2.5.6 Job - Characteristics - Model

This model combines the incentives of the work (the content of the work) with the experiences of the persons (critical psychological conditions). It was developed by Hackman and Oldman in 1980 (see figure 8 [Zim99, page 723]) and subdivides the content of the work into 5 core characteristics and determines 3 critical psychological conditions which are positive for an intrinsic motivation[8].

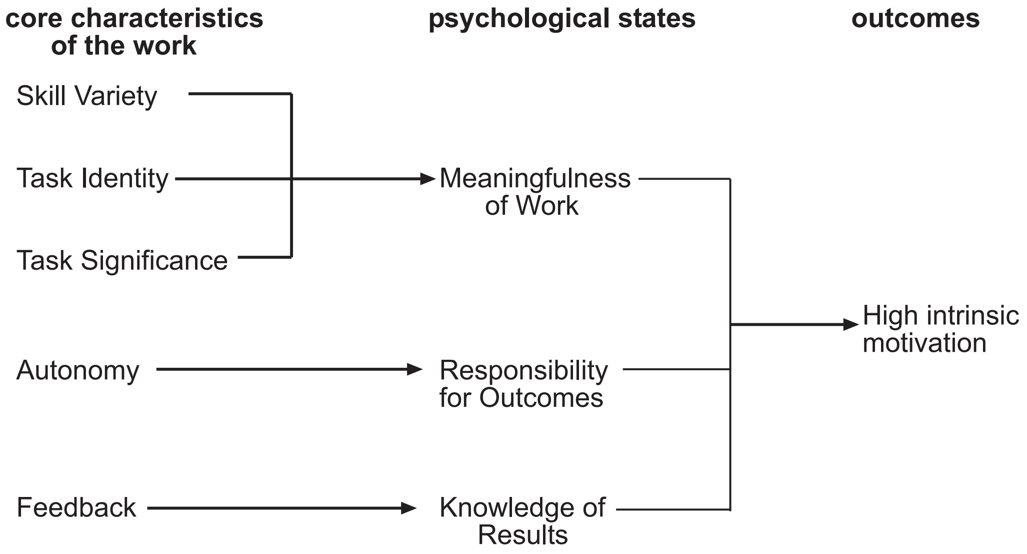


Figure 8: Job Characteristics Model

The potential of the motivation can be calculated by the following formula. (see figure 9 [Zim99, page 723])

$$\text{MPS} = \frac{\text{Skill Variety} + \text{Task Identity} + \text{Task Significance}}{3} \text{ x Autonomy x Feedback}$$

MPS ... Motivating Potential Score

Figure 9: Job Characteristics Model Formula

[8] The intrinsic motivation says that the content of the work directly motivates the employee.

2.5.7 Mix Match Method

The Mix Match Method is mostly used in matrix organizations for programming teams. Each programmer is involved in at least two projects over a year. One is an internal and the other an external project. The programmers rotate among the projects and the managers have to motivate always different groups.

> "One of the greatest benefits for programmers was the exposure and skills they would gain by working on different types of projects. [..] In general, we've found that the matrix model is great for high performers because it allows them to stay challenged without getting bored. On the other hand, it puts more pressure on poor performers, who can no longer hide under a maintenance function or a single project." [Fie02]

2.6 Leadership

There are different management forms which can influence in various ways the motivation and rate of work of people. Different companies need different models and types of guidance.

2.6.1 Transformational Leadership

If a company uses the transformational model, it tries to reach the developers ambitions and dreams. The management will signalize the engineer that he got one of the most interesting, important and exciting tasks he ever was confronted with. So the management teams will try to take advantage of the fact that "most engineers and scientists aspire to greatness and long to participate in the excitement of some grand venture." [Hum99b, page 6] As a result, the developers will gain immense personal satisfaction and are highly motivated. It creates an atmosphere of perfection who everybody wants to create the perfect product. But you have to be careful. Not every project is made to be handled in this way. For example, if you have a normal scientist, he will need months or years of preparation to make a single discovery. There will be periods during his project which are very frustrating but nevertheless, the work has to be done very exactly if you want to get the best results.

As an example you could see the team of Tom West[9] who worked for Data General in the Eagle computer project. "He portrayed his project as an effort 'to try to build the unattainable, the perfect computer. West talked for weeks to engineers about his plans and how important they were to Data General. [..] West put together a dedicated technical team that worked incredibly long hours to design the best computer they could imagine." [Hum99b, page6]

2.6.2 Transactional Leadership

The transactional way has to do with the manager himself. In this model the legitimate power of the management level can motivate or unmotivate the employees. Every member of the management level has the official power his or her position provides him. That can be the possibility to increase the salaries, provide special job promotions or assign important and interesting tasks.

The developers "know that their manager's opinion is important, and they work hard to earn and to keep their manager's favor." [Hum99b, page 6] So the extra effort to gain the managers attention can motivate to work harder and better or it can unmotivate the employee. How can that be?

For instance, "employees with worries about their salaries or the next promotion are more likely to focus on what the boss wants rather than what the job needs. They are thus unlikely to be as creative or to hotly debate a controversial technical issue." [Hum99b, page 7]

[9] Current position (2005): Senior vice-president of advanced development, Data General Corp.

2.6.3 Leading from below

Not only the top-management has to lead all others or can motivate others. Nearly every manager can agitate as a leader and create an atmosphere that motivates others.
Additionally, every level of the management team has to work very hard and motivate their subordinates. "In fact, no technical organization can do superior work unless its junior managers take charge of their jobs and energize their people." [Hum99b, page 8]
The big difference between the top management and the lower management is that the lower management has to fight against a lot of bureaucratic attitudes. The top management can react and order something to the lower management. That means, that they can act. In contrast, the lower management has to accept the orders of the top management and find out, whether there are bureaucratic barriers that first have to be overcome. These barriers can unmotivate the lower management and that will be transfered to their subordinates.

2.6.4 Autocratic style

The manager has the only power and holds distance to his subordinates. All the decisions will only be made by him and he tells everybody what he has to do.

2.6.5 Democratic style

The manager includes his subordinates in the decision making process. Here the cooperation of the employees with the management is very important.

2.6.6 Laissez-faire style

Here the employees are free in their decisions and the management will not introduce too many rules.

2.7 The meaning of rewards in motivational reasoning

Rewards are important to hold the motivation of the engineers up[10]. From time to time, they make it possible to praise some engineers for their excellent work or achievements. The engineers will be honored and notice that the company rewards good work what will motivate them to continue their efforts. But, there is one important rule that every manager has to follow. The reward has to be immediately after the success of the engineer. Otherwise, the employee can't associate the reward with the situation that has been chosen for the reward.[11]

2.7.1 What you have to take into account

Rewards should be clearly specified and independent of other rewards. Everybody should be informed about the existence of the rewards and it has to be possible for everybody to achieve them. Nothing is more unmotivating than to find out, that there are rewards but only the top management is in the position to get them.

Over and above that, it is important to realize that the subordinates do the work. If the top management honors a manager, they have to honor the subordinates of this manager, too. Otherwise, the employees think that their work will not be acknowledged and won't give all their energy and enthusiasm during the next projects for the honored manager.

2.7.2 Recognize their work

People like to be honored for their work. One of the best ways to honor the work of an engineer is the recognition that can be given to him. To recognize the work of professionals, the manager can send mails with the mention of the respective developer. The developers can be mentioned during project meetings and their work can be emphasized. Additionally, the specific developers could get special awards for the excellent work.

> "Nothing can motivate the developers to a great extent as recognizing the great piece of work done by them. When you recognize their work, they feel honoured and important. It also brings in a feeling of belongingness and great satisfaction to the developers." [Jay04]

2.7.3 Publicity is the key

If you want to honor someone, you should do it in public. Public rewards are just more powerful than private rewards. The reward is not only given by the manager, it is also the extra recognition

[10] Some experts say that a reward of 10 to 15 per cent of the basic salary will work motivational to the professionals.

[11] One of the may be most curious examples is the 'golden banana' of the Foxboro Corporation. Once, a scientist of this company solved a very complex problem and drove to the president's office to show him his solution. The president was very thankful and wanted to reward the scientist. He searched for something on his desk, but could not find something. The only thing was a banana which he handed to the scientist. Although this wasn't very special or expensive, it was enough. The engineer feeled honored and noticed. From this day on, the 'golden banana' pin is awarded to the most successful scientist.

of the other employees which honors the selected person.
Once, Charlie Beacham used the following words to express it: "If you want to give a man credit, put it in writing. If you want to give him hell, do it on the phone." [Hum99b, page 192]

2.7.4 Honor Team and Individual rewards

If there are important individual achievements to a team success, the manager should reward the whole team and the individual contributions, too. Honor the whole team during a dinner and give money awards to the individual achievements of the employees.

2.7.5 The job is the most important fact

Managers should be careful with the rewards. If the engineer only sees the reward as the aim, he will not be very creative. "As soon as the reward becomes the objective, some employees will stop striving to do a better job and will start viewing the the reward as the goal." [Hum99b, page 194]
Another important fact is the repetition of the rewards in connection with reduction of the incentive. This can lead to ignorance and make out of the reward a disincentive. A nice example of how a reward becomes uninteresting for persons illustrates the following story from the Kansas City Star:

> "An elderly man, harassed for weeks by the taunts of neighborhood children, devises a scheme. He offers to pay the children a dollar each if they'll return the next day and yell their insults again. They do so eagerly and he pays them as promised. 'If you come back tomorrow Ill give you 25 cents,' he tells them. They do and collect their quarters. 'From now on I'll pay you a penny,' the old man announces. The children are contemptuous. 'A penny? Forget it!' they reply, and they never return."[Hum99b, page 194]

This is an excellent example how rewards can go down in value.

2.7.6 Let them make their work public

One of the biggest honoring for engineers is the possibility to make their work visible. Professionals want that their work will be public. Even if no one honors their work, they hope that the visible paper will change that. But can this extra work have influence in the result?
Yes it has! Visibility motivates a professionals to deliver a really perfect work. "He thinks more logically about the alternatives and makes an extra effort to find the key references. [..] Everyone likes to be recognized, and people who have published know the thrill of seeing their names in print." [Hum99b, page 28/29]
Additionally, it generates pride (see section 2.1.3) and self-confidence[12]. Professional people who's

[12] John W. Gardner: 'Excellence is not an achievement of demoralized or hopeless individuals' [Hum99b, page 30]

work will be recognized and honored "are more willing to fight for their beliefs and to take the risks of creative work." [Hum99b, page 29]

2.7.7 Ask them what they want

The different professionals have different dreams and expectations. Sometimes it is not very useful to offer everybody the same rewards. "All of this leads to a very simple concept: communication. Simply put, ask the your shining stars what they really want." [Rus04] The answers can be very different like the persons who are standing behind these answers. "You may be surprised by the answers you hear. In fact, your employees may be surprised, as well, to learn that you are actually giving them a say in determining the reward for their efforts." [Rus04]

This way to reward the engineers has two very important effects. Firstly, it really gives the professional what he wants. This has something to do with Maslow's Hierarchy of Needs (see section 2.5.4). Second, the professionals feel important and see that the management cares about them. This type of motivation is shown in section 2.5.3 where the Hawthorne Studies are described.

2.8 Company example

2.8.1 Cisco

Cisco uses different integrated systems to inform their engineers. That are e.g.:

Cisco Newscast and Archives: "A multilayered method of filtering, publishing, and broadcasting internal and external news to employees globally." [Sys99, page 4]

Cisco My Yahoo: "A partnership with Yahoo creating a Cisco version of the popular 'My Yahoo' service that integrates key Cisco information with an employee's personalized news, stocks, sport, and weather information." [Sys99, page 4]

Specialized Dashboards: "A series of sites with Web links to existing information and applications relevant to employees globally." [Sys99, page 4]

Additionally, Cisco creates an environment of trust and understanding by giving every employee access to a platform which distributes information. Furthermore, an intranet solution was developed which is the starting point for every employee.

> "Cisco maintains an interactive intranet Web site called CEC. This information- and transaction-rich solution allows the firm's 15,000-plus employees to get information when they need it [..]" [Sys99, page 7] "This focus on developing a new employee's time to productivity not only maximizes the company's efficiency, but the quick integration also helps to build and maintain a close employee community." [Sys99, page 8] "Cisco uses the Web to build interactive, knowledge-based relationships with its employees. [..] ,all employees are connected to all aspects of the HR department [..] The result is increased productivity and higher employee satisfaction, [..]" [Sys99, page 9]

The intranet solution will also be used for training programs. It provides a simple way to inform the employees about training schedules and the management about the registered engineers.

> "Several training issues were solved by the Cisco workforce optimization solution. Cisco employees are now able to view up-to-date information on available training courses through the web. [..] can view additional information [..] individual course history [..] When employees register for a course, their manager is automatically notified [..] the system automatically reminds the manager and the enrolee [..]" [Sys99, page 12]

To gain the trust of the employees and to create an open atmosphere, Cisco placed its policy into the intranet.

> "Cisco choose to place its policy manual on its intranet, providing employees with immediate access to policy information [..] One of the best ways that a company can establish relationships with its employees is to be open with them. [..] there are very

> few doors that the company is trying to keep employees from opening." [Sys99, page 12]

2.8.2 Microsoft

All available information like revisions, problems, proposals, concepts, architectures, etc. will be send around via E-Mail. The contacted persons can react and tell their comments.
Microsoft hires only the best professionals for their teams. They were the best in school and at the university. Therefore, the competition is very strong and the professionals are forced to improve their skills.

The managers have to inform their subordinates about all important facts of the respective projects. If there are bigger changes of policy, the management has to ask the professionals about their opinion and the agreement to the change. That creates confidence between the management and the professionals and the managers can more likely expect the full dedication of the professionals.

Microsoft has three important factors which motivate their professionals [Thi01, page 99]:

Interesting job: The work is very interesting at Microsoft. People are easier to motivate for a work which is exciting and innovative.

Success is the only standard: The professionals will be measured on their success. They know that their value will grow if they have success and deliver perfect work. This strive for success is a big motivator which let the professionals work very hard for the company.

Pressure of the team: Microsoft has surely some of the best professionals in the market. Therefore, all team members expect the best from the other individuals. It is an atmosphere of pressure where all try their best for the team and the success of the project.

All professionals are only assigned to one project. Never a professional will work on two or more projects at the same time. So, the professionals can concentrate themselves completely on their project and won't be diverted.
Meetings are like an entertainment show. The managers try to amuse the professionals during endless meetings. That raises the moral and the participants don't fall asleep during boring reports. Very often, the mistakes of the higher management will be discussed and made fun of.

Fun is an important component of the work. Sometimes it can happen that the professionals play ball in the corridors. That creates an atmosphere where the people have fun during their work. They like the working place because it is a place where they have a good feeling. Here they can do things which would be impossible in other companies. "That is the cement which welds the people together." [Thi01, page 149]

The professionals have offices instead of cubicles. They can furnish the rooms and decide which equipment will be installed. "At Microsoft, the office is the property of the professionals and not of the company." [Thi01, page 170] The professionals feel happy in their own empire, and where could you work better than within *your* own four walls.

Microsoft has no dress regulations for the employees. So the professionals know that their value will be calculated by their work and not by their outfit. They can concentrate themselves on their work and not on the perfect outfit which is expected. Another free concession are free drinks and that Microsoft provides all materials which are necessary for the work without any limitations. Paper, printer or computer are free available and for everybody accessible.

3 PML, PSP and TSP

The second part of this paper deals about PML[13], the Process Maturity Levels, PSP, the Personal Software Process and TSP, the Team Software Process. These models have been developed by Watts S. Humphrey who was a senior software-development executive at IBM and is now at the Software Engineering Institute, Carnegie Mellon University.

The Process Maturity Levels form the basis for the Personal and the Team Software Process. It was developed by Watts S. Humphrey and is also known as CMM, the Capability Maturity Model. The PML divides companies into levels of improvement. The companies are differently evolved in the single levels and use different methods and processes to fulfill their contracts. Every higher level is an improvement in the development capability of the company. Therefore, the companies have to correct their mistakes on the single levels and learn new skills to rise in the hierarchy of the levels.

The two other methods are the Personal and the Team Software Process which were also developed by Watts S. Humphrey. The Personal Software Process is a guideline for students and professionals how they can improve their programming and working skills. It has several steps which make it easier for the individual programmer to plan his work, write his code, reduce his errors and get data about his own coding behavior. The produced data from current or former projects can be evaluated and give a clue where the professional can improve his skills.

On the other hand, the Team Software Process is a guideline for software development teams that helps to "improve the process management in an organization" [Uni04c]. The process describes, how managers can improve the quality and the reliability of their teams in the software development process. Furthermore, the teams learn to "plan and track their work, establish goals, and own their processes and plans." [Uni04c]

History
The PML (or CMM) model was developed to organize the software development process within organizations in the late 1980s and early 1990s. "Watts S. Humphrey decided to apply the underlying principles of the CMM to the software development practices of a single developer." [Uni04e]
As result, Humphrey developed the Personal Software Process which is "a CMM level 5 process for individual software developers." [Uni04e]
As it became clear that software companies would need additional methods for small units of teams, Humphrey developed the Team Software Process. The "TSP was designed to be a CMM level 5 process for project teams." [Uni04e]

[13] Also known as CMM (Capability Maturity Model)

4 PML

4.1 Process Maturity Levels

This model[14] is based on 5 levels (see figure 10 [Hum89, page 6]) which "reasonably represent the actual historical phases of evolutionary improvement of real software organizations.[15]" [Hum89, page 5] Additionally, they "represent a measure of improvement that is reasonable to achieve from the prior level, suggest interim improvement goals and progress measures and make obvious a set of immediate improvement priorities, once an organization's status in this framework is known" [Hum89, page 5].
The five levels are:

1. Initial
2. Repeatable
3. Defined
4. Managed
5. Optimizing

If a company is clear about its current maturity level[16], it is possible to establish the structure for the next higher level. Therefore, it is important to find the right methods for the motivation of the employees in the different levels of the model.

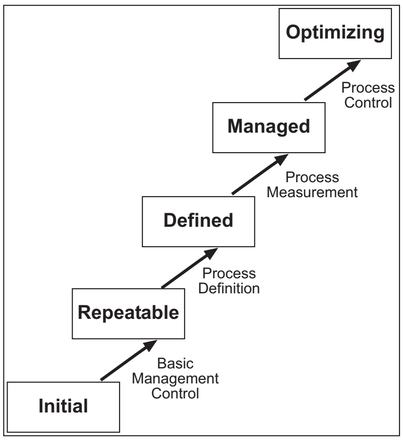


Figure 10: Process Maturity Levels

[14] Also known as CMM (Capability Maturity Model)

[15] "A CMM describes an evolutionary improvement path from an ad hoc, immature process to a disciplined, mature process with improved quality and effectiveness." [Cur02]

[16] The current maturity level can be measured with the help of additional assessment methods.

4.1.1 Initial Maturity Level

This level is marked by chaotic software development projects which are not well planed and formalized. The management behavior is rather unstructured and not very resolute. There are no specific methods to improve the efficiency of the employees or to increase the quality of the work. All these basic mistakes produce a very frustrating atmosphere. The first improvements in the motivation process would be the "effective control of commitments ." [Hum89, page 7] To achieve a better work atmosphere and quality, the company has to introduce clear responsibilities. Additionally, the company has to improve the project development in every single phase of the process.

4.1.2 Repeatable Maturity Level

In the Repeatable Maturity Level, the company could establish one big advantage in contrast to the Initial Maturity Level (see 4.1.1): "It provides control over the way the organization establishes its plans and commitments." [Hum89, page 8] This makes it easier for the employees to repeat their work and to achieve better results. They are motivated by their experience from earlier projects and the certainness to achieve the same results again. To achieve the next level, the company has to "establish a development process architecture, and introduce a family of software engineering methods and technologies." [Hum89, page 8] These steps bring additional sureness for the employees in the development process and make it easier to fulfill the required tasks.

4.1.3 Defined Maturity Level

At this moment, the company could achieve "the foundation for major and continuing progress. For example, the software teams, when faced with a crisis, will likely continue to use the process that has been defined." [Hum89, page 9] The advantage of the defined structure makes it easier for the management level to introduce new methods to the employees and gain their agreement.

4.1.4 Managed Maturity Level

Here the company uses the collected data from the former projects to get a strong basis for further projects. That makes it possible to improve the process.

4.1.5 Optimizing Maturity Level

In the former levels, the analyzed data was used to improve the product. Here "the data is available to tune the process itself." [Hum89, page 11]

4.2 The human factor within the levels

The different levels provide enough room for the management team to "enhance the talents of quality people in several ways. It helps managers understand where help is needed and how best to provide people with the support they require." [Hum89, page 12] Furthermore, the quality and the teamwork in the software development process will be improved in a manner that the working atmosphere between the different teams and individuals will be optimized. But, how to realize the single improvement steps? How to motivate the people on the different levels? How to gain the full support of your teams?
The key to realize the different innovations is information . You have to inform and involve every single member of your group, team and company. Everybody has to get the feeling to be a member of the optimizing process. "In the software process, people are the most important ingredient. It is essential to recognize their desire to do good work." [Hum89, page 20] And a second very important fact is the time. Don't overstrain your people with the whole bundle of methods in an unexpected manner. "Process changes should be made in small steps, and even then they must be tested and adjusted before wide-spread implementation. Unless process improvements have the wholehearted participation and support of the people most directly involved, and unless they are carefully planned, thoroughly tested, and introduced in phased steps, they will probably be disappointing." [Hum89, page24] In the end, the process optimization hopefully brings the following effect: "With proper leadership and support, most people can do much better work than they are currently doing." [Hum89, page 18]

4.3 Software Process Assessment

The Software Process Assessment can be used within the different levels of the Process Maturity Model 4.1. It "is not an audit but a review of a software organization to advise its management and professionals on how they can improve their operation." [Hum89, page 36] The process will be managed by software professionals which are out of the same organization or especially hired for the examination. In the end, the "assessment is to identify the highest-priority areas for improvement and to provide guidance on how to make those improvements." [Hum89, page 36]
The process assessment process can cause a lot of problems for the company between the employees. People could think that the so called experts are superficial and talk about things they only partly understand. So they get the impression that "a group of remote 'experts' reviews a large and complex organization and in a few days tells them what they are doing wrong and what they should do to improve. Generally the local people work hard, are dedicated to doing a good job, and are trying to improve." [Hum89, page 39] The employees feel misunderstood and think that their work was not enough appreciated. That can make the whole assessment unsuccessful because the employees will work against the assessment team and won't support them with the information and assistance they would need to make a good job.
To get the support of the employees and their agreement, the company has to convince them

that the whole process will be made in a cooperation. So you have to convince them to share their knowledge and that the assessment can also be a way to improve their own quality. "The professionals will be willing to share, however, only when they see the assessment as a way to get help rather than as a threat of exposure." [Hum89, page 40]
Moreover, if the assessment discovers that somebody in the company already found a solution for a problem or discovered a new idea, they should make this public. "Surprisingly, for each software problem there is often someone in the organization who has already solved it. Making this capability visible can be one of the greatest and most immediate benefits of the assessment." [Hum89, page 40] Because, if you get the recognition for your work, you will be motivated to continue your extra work and realize that the company not only recognizes the expected results.

4.4 P-CMM

The P-CMM was developed to improve the quality and effectiveness of the people of an organization. The people are the heart of the organization. If the people are motivated, skilled and experienced, they can solve nearly every problem and bring a lot of success for the company.
The model was developed by Bill Curtis and other scientists who wanted "to improve the way your organization deals with its people." [Hum99b, page 298] Mainly, the model was developed for the management of the professionals and increase their value for the organization. So, the P-CMM delivers methods and suggestions to change the management behavior with the change of the company in its PML evolution.

4.4.1 From Level 1 to Level 2

The management has to create an environment that supports the demanded goals. The professionals should get the necessary resources, to do their best to achieve the required goals. The motivation will increase, if the professionals can develop their potentials without having trouble to get the necessary materials. The communication with the individuals should be strengthened. This will create a trustful atmosphere where the creativity can shoot up. Rewards should be provided for outstanding achievements. This will lead to self-motivated professionals who know that their work has a value.
As final remark, individual or group training should be available to develop new skills. These training courses indicate the individuals that their personal improvement is important for the company and that the organization is interested in their development.

4.4.2 From Level 2 to Level 3

At this point, the organization has to extend the core competencies which are necessary for the fulfillment of the projects. The engineers have to be trained in these skills and make their own experiences. "Without a competitive workforce, it is hard for organizations to be superior, regardless of anything else they do." [Hum99b, page 299]

This in-door training has two main effects. First, you get the workforce you need to master the projects and second, your engineers realize that their development is important for the organization. They feel important and supported by their managers. This is the basis for a harmonic cooperation. Besides, the organization has to support each individual goals and career plans. This is another step to win the trust of the professionals.

4.4.3 From Level 3 to Level 4

Now, the organization has to develop team skills. Very often, a lot of scientists and engineers are necessary to manage all the required work. Therefore, the company has to support team building processes which produce an effective teamwork. (see section 2.3 about teams and section 6 about the TSP model)

In the steps before, the organization had to support the individual goals of their engineers. Now, the company should merge the individual goals with the goals of the organization itself. In the following, the engineers get the feeling that they can reach their personal dreams with the help of the company.

4.4.4 From Level 4 to Level 5

To reach the uppermost level, the focus should be on coaching the entire organization and improving all single steps to get the most effective engineers and best results. So, the managers should support their subordinates and let them improve their personal skills. This coaching process should be seen as assistance. The engineers get the required help to make their own experience and beyond it, they get attention which strengthened the self-esteem.

5 PSP

PSP[17] is a method that allows engineers to plan their work more effectively and get data during their work which they can use to improve their results. The collected data make it possible to improve their work and effectiveness in future projects.

Furthermore, the person who uses PSP is in a position to produce higher quality products. They use effective methods to remove defects in their code before they test or compile their programs. This makes it possible to work more effective and predictable what leads to more accurate time plans.

> "It is about collecting *your* data, to understand *your* performance, so you can improve what *you* are doing, and to strive to reach *your* personal best." [Dav03b]

In his book '*Introduction to the Personal Software Process*', Watts S. Humphrey describes the methods and procedures used. The following subsections contain the basic statements considering the relation to the motivation of engineers.

5.1 Introduce the new model

New models or processes are always hard to introduce in software companies. The developers have their own habit in writing code and doing their work. "Engineers are understandably skeptical about changes to their work habits; although they may be willing to make a few minor changes, they will generally stick fairly closely to what has worked for them in the past until they are convinced a new method will be more effective." [Hum98a]
This leads to a problem. How to convince and motivate the engineers that the new method is better than the old ones? Because, "engineers only believe new methods work after they use them and see the results, but they will not use the methods until they believe they work." [Hum98a] As a solution for this problem, the company could "remove them from their day-to-day environment and put them through a rigorous training course." [Hum98a] There the programmers learn in defined training levels (see figure 11) to write little exercise programs and with each new level they will be introduced into some features of the new method.

5.2 Write an Engineering Notebook

The first step in improving the quality of the programming process will be the using of an engineering notebook. This book is necessary to track the time which the professional needs for the different tasks and to get an overview of how long each task has taken. Furthermore, other things like "recording assignments, tracking commitments, [..] design ideas and calculations"

[17] "The Personal Software Process (PSP) was designed to help software engineers do good work. It shows them how to apply advanced engineering methods to their daily tasks. It provides detailed estimating and planning methods, shows engineers how to track their performance against these plans, and explains how defined processes can guide their work."[Hum99a, page 2]

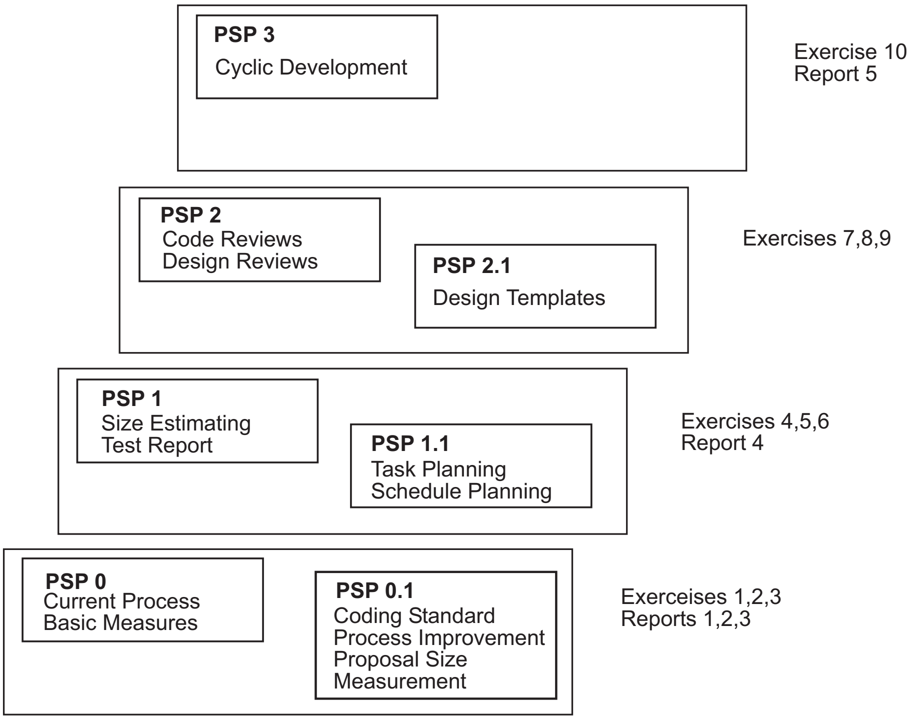


Figure 11: Training levels

[Hum99a, page 12] can be listed. The book provides the desired data to make evaluations (see figure 12 [Hum99a, page 21] as an example of a how many minutes engineers have uninterrupted worked).

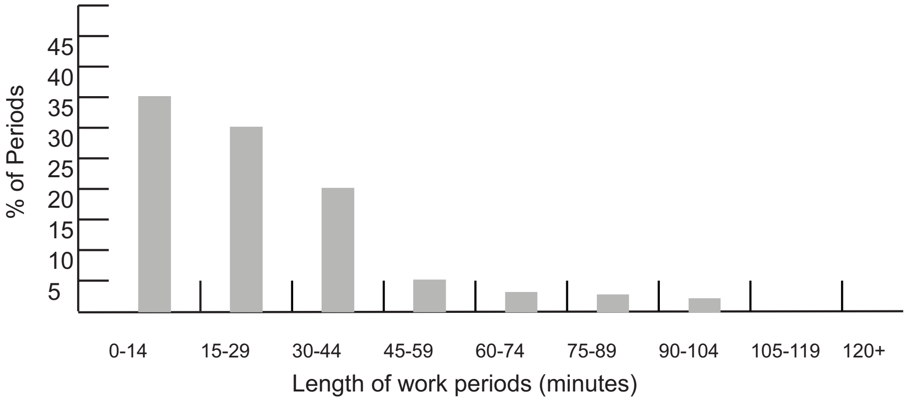


Figure 12: Work periods

Such a book can help the professional to track his work and develop his skills. The engineer can measure his working hours for specific tasks and see improvements for coming projects. Additionally, the engineer can discover the useful and the useless methods. That helps him to improve his

personal skills for future projects. If persons realize their success and the resulting improvements during their work and in coming projects, it will motivate them to go on working and to believe in the chosen methods.
To be aware of your skills and how they become better can be a big motivator.

5.3 Making product plans

Every professional should make his individual plan to get a summary about his tasks. The plan gives an overview of how much work he already has done and how much time it has taken. Besides, the engineers can track progress while he is doing his work.

Such personal plans provide a lot of clearness. The engineer can coordinate his tasks and make suitable project schedules. If the management accepts these more realistic plans, the professional is much more motivated to achieve the work. "With reasonably detailed and accurate plans they can judge where a project stands against the plan." [Hum99a, page 46] He personally knows that the schedule is possible to reach and so he will do his best to prove himself. Naturally, the engineer will may be not get a last minute crisis because he can organize his work according to his personal skills and working preferences.

5.4 How to handle commitments

There are some basic points which are very important if persons make commitments. The management should pay attention to them if they saddle their professionals with them.

"Analyze the job before agreeing to the commitment." [Hum99a, page 89] The two parties who are making the commitment should completely understand the state of affairs and have a good feeling in the commitment. For an engineer it would mean that he really wants to do the job and that "the other party intends to provide suitable compensation in return." [Hum99a, page 89] That guarantees that both parties are really interested to finish the job and will do their best to get it.

"Support the commitment with a plan." [Hum99a, page 89] A plan improves the security to reach the aim and gives an overview of the important milestones. The parties can track the progress of the commitment and intervene if help is necessary.

"Document the agreement." [Hum99a, page 89] The two parties should document their agreement to avoid misunderstandings in later meetings. This is a kind of precaution to make sure that both understood it in the right way and are on the same level. It helps to avoid disputes and gives a feeling of security for both parties.

6 TSP

The Team Software Process was designed for industrial teams who develop software systems. Teams are defined to have the following characteristics after Dyer: "A team consists of (a) at least two people, who (b) are working toward a common goal / objective / mission, where (c) each person has been assigned specific roles or functions to perform, and where (d) completion of the mission requires some form of dependency among the group members." [Hum00a, page 19]

6.1 Participation

The single members of a team have different skills, abilities and personal motivations. So the professionals will perform different contributions to the project and "the variation among the members contributions generally increases with increasing group size." [Hum00a, page 18]
If there are team members who not contribute an acceptable amount of work, the others will realize it and this is fatal for the team spirit. The management has to supervise the performance of the team and of the individuals. If there are some who obviously not work like they should, the management has to step in.

> "If everybody is suffering equally, you can move a mountain. But the first time you find someone goofing off or not carrying his share of the load, the whole thing can come unraveled." [Hum00a, page 18]

6.2 The size is important

The size of the development team can vary from two to twenty or even hundreds of engineers. The most effective teams are small and have close relationships among the team members. These teams create an atmosphere of unity where every member of the team is proud to be a part of the whole.
Normally, even bigger projects create subgroups of teams who are supervised by a manager. Watts S. Humphrey prefers the following team size.

> "... my experience has been that teams of four to eight engineers are likely to be the most effective. With fewer than four members, there are not enough people to properly handle all the team role assignments. With teams of more than eight members, it is harder for the team to develop the close relationships needed for teams to jell."[Hum00a, page 19]

If the group is successfully transformed into a team, the manager has to realize some basic conditions. These make it possible for the team to operate successfully as a team and get the right motivation for the forthcoming projects. [Hum00a, page 20]:

1. "The tasks to be done are clear and distinct; that is, the job for the team is explicitly defined, the work is meaningful to the team, and the group knows what it must do."
2. "The team is clearly identified, the members know the scope of the group, who is in it, and who is not. Everyone on the team is known to the others, everyone's work is visible, and everyone knows everyone else's team role."
3. "The team has control over its tasks; members know what to do, how to do it, when to do it, and when they are finished. The members know that they are responsible for the work, and they control the process they use. They also have the capability to do the job, and they know that no one else is charged with doing it."

6.3 How to handle the pressure

Very often, pressure is one of the main problems in big software projects. The software team can't handle the pressure and the project will fail in the end. Especially an aggressive time schedule will change their work routine. Changes can be the use of poor or easier methods, choosing new or unknown programming languages, experimenting with new tools and techniques or trying to reduce and simplify the functions of the final product.
So, pressure can have a destructive effect to the engineers. "It causes people to worry and to imagine problems and difficulties that may not be real." [Hum00a, page 16] This can lead to negative consequences for the company and reduce the chances of success of the project.
That's why managers should teach their individuals and teams how to deal with pressure. That's why managers should make detailed plans and job descriptions which they discuss with their professionals. Without this information, the professionals have to handle unknown factors. They only see the big project but can't overlook it. But if they get a plan and can see the single steps, it will be possible to talk about eventual problems. Additionally, the professionals "are now dealing with a known problem rather than an unknown worry." [Hum00a, page 16]

Pressure is something that every individual creates in himself. The individual can feel the pressure and if this person has doubts about his abilities, the pressure will become bigger and bigger. To master the pressure, the person has to find out the source of the pressure and find a way to handle it.
In software projects, the frequent source of pressure are the tight schedules. These schedules are made by the management. If a team cannot finish the product within the given schedule, the previously explained problems can occur. Therefore, the management has to make the schedules together with their teams to create realistic time schedules. If the team is involved in the planning process, they will be able to analyze the project and give professional suggestions how to achieve a quality product. And, the professionals will not be surprised by unrealistic preconditions. They can judge their time plan and don't have to be afraid of unrealistic orders.

6.4 Building an effective team

The may be most important step in working with teams is the beginning. The management has to satisfy the special needs of the teams which are necessary for the perfect environment. The additional types of support are:

- Team Cohesion
- Challenging Goals
- Feedback
- Common Working Framework

6.4.1 Team cohesion

"Cohesion refers to the tight knitting of the team members into a unified working group that physically and emotionally acts as a unit. "[Hum00a, page 21] The members of such teams freely communicate with each other and have a respectful relationship. Cohesive team members act more like friends compared with non-cohesive teams where the team members work as individuals. "Cohesive teams, however, share a common physical space, spend a lot of time together, and supportively cooperate and interact during these times together." [Hum00a, page 21]

6.4.2 Create challenging goals

The goals are another important element for effective teams. The management team has to make these goals measurable and clearly understandable. Examples are "detailed plans, performance targets, quality objectives, schedule milestones" [Hum00a, page 21]. Furthermore, the goals have to be a real challenge for the team. Such goals let the team fight together against all other trouble.

That challenging goals can have this positive effect to the team members, the success of the goals has to be tracked and made visible. Because of that, the team members can register their progress toward their goals.

6.4.3 Why feedback ?

The team has to be aware of its performance and the single team members have to distinguish between their personal performance and the performance of the whole team. If there's only one member who's performance is shrinking, the performance of the whole team can shrink and the success of the project is in danger. Therefore, the management has to supervise the performance measurement and spur on every single member. Because, in the most successful teams, every team member equally brings in its contribution for the team.

6.4.4 Creating a Common Working Framework

"Team members need to see how to achieve the goal and know what is expected of them". [Hum00a, page 22] This means that the single team members must understand their role in the team and realize that the single steps of the goals are achievable. Therefore, they should know [Hum00a, page 22]:

- "What tasks must be done?"
- "When?"
- "In what order?"
- "By whom?"

The previously mentioned elements for building successful teams are important tasks for the manager. They have to work with the teams and lead them to be successful and create their own individual team structure.

6.5 The project progress

A TSP project is subdivided into eight different steps or processes which are necessary to develop an effective team project. These processes can be seen like the phases of a software development model.

1. The launching process
2. The development strategy
3. The development plan
4. The requirement process
5. The design process
6. The implementation process
7. The test plan
8. The postmortem

6.5.1 The launching process

In the beginning of the project, the team manager has to inform the team about some important decisions. "This is also a chance for management to motivate the team. The team has the opportunity to ask any questions they might have about the product or business needs." [Dav03b] Then, the members of the TSP team have to get their roles in the team and understand the tasks which they have to handle. Because of that, the single individuals feel responsible for their tasks and see themselves as a part of the team. Professionals who know what they have to do and get a feeling of their importance for the team are much more motivated than individuals without defined responsibilities. This is the first step of building a team in the TSP model.

The second important element of the launching process is the setting of goals. There are some important points about goals for teams which the manager has to absolutely take into account.
On the one hand, "if the goal is too easy, there will likely be little motivation to strive. On the other hand, a goal that is clearly unachievable also provides little or no motivation." [Hum00a, page 30] The manager has to make realistic goals. They can be aggressive and challenging but they have to be realistic. The team has to get the feeling that they can reach the goal if they work hard enough.
Another important point is the valuation of the goals. "Teams should not be measured on whether or not they actually meet their goals. Instead, they should be evaluated on their willingness to set measurable and aggressive goals and on their efforts to meet them." [Hum00a, page 30] If the managers are only interested in the attainment of goals, it will be counterproductive for the teams because they will set goals of which they are sure that they can meet them.

6.5.2 The development strategy

In the development strategy, the team will "devise a strategy for doing the work, create a conceptual product design, and make a preliminary estimate of the product's size and development time." [Hum00a, page 49]
The team has to plan the whole strategy. The most important things are the coordination of the required materials and the personal. If the team does not get a clear overview of the project, they cannot contradict the management. But if they know that a proposed plan is not achievable, they can make a new plan which is much more realistic.

6.5.3 The development plan

A plan let the teams work more efficiently. The single members of the team will know what they have to do and when to do it. So, the team can start the project in a more composed feeling.

The best possible plan will be a balanced plan. In such a plan, the "engineers complete their planned tasks in the proper order and at roughly the same time. With a balanced plan, no one needs to wait for anyone else, and everyone is fully utilized." [Hum00a, page 66] Additionally, Envy plays no role in such plans. Everyone has the same amount of work and the team members feel equally involved.

6.5.4 The requirement process

The requirements of the software product will be described in this phase. Here it is very important to get a common team agreement about the product that they have to make. Another step to weld the team together.

6.5.5 The design process

"The principle objective of the design process is to produce a precise, complete, high-quality foundation for product implementation." [Hum00a, page 122] In a team, the manager can subdivide the entire product into little parts. That makes it possible to work faster and break a complex big project into smaller steps which are easier to handle. So the engineers are much more motivated to finish their parts very precisely to deliver a correct design.

6.5.6 The implementation process

During the implementation of the product, the team should use common standards like coding standards. This makes the reviewing of the code easier and the team members can easier understand the code of the other engineers. So communication problems will appear less often.

6.5.7 The test plan

This phase contains the testing and the documentation. If there are poor quality modules, the manager has to send them back to the responsible engineer and give him the time to improve his work. Mostly, the engineers are able to produce high quality software, but short time schedules make things difficult for them. With a little bit of extra time, even the biggest problems should be solvable and the professionals have confidence in the management team.

6.5.8 The postmortem

The postmortem[18] is the last phase of the TSP processes. The manager reviews together with the team all the required tasks to be sure that everything is completed. It is a kind of final stocktaking.
Here the manager and the team members can examine the work "to learn what went right and wrong and to see how to do the job better the next time." [Hum00a, page 185]
Not only the used methods and processes can be reviewed. Also the relations between the team members and the management can be examined. If there are differences between them, the management should try to think about them and find solutions for a better cooperation in coming projects.

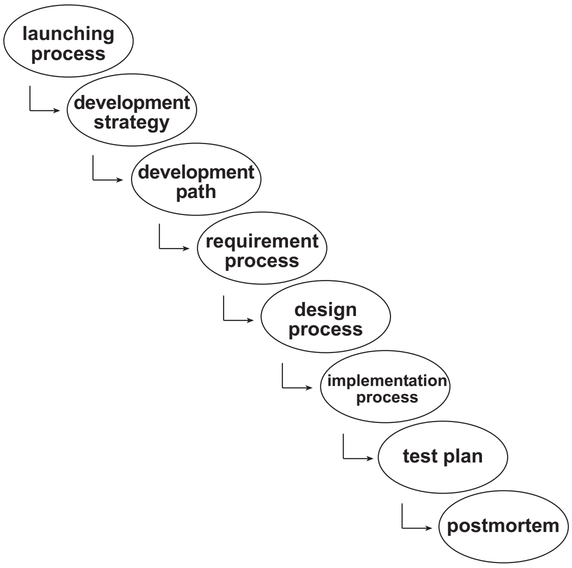


Figure 13: TSP structure

[18] "Perhaps the best way to describe the principle behind the postmortem is with a definition: 'Insanity is doing the same thing over and over and expecting a different result'. This definition captures the spirit of the postmortem process. If we do not change the way we work, we continue to perform much as we have in the past. The postmortem provides an orderly way to identify areas that need to be improved and to make the needed changes." [Hum00a, page 185]

6.6 Team roles

TSP teams consist of different roles for the single team members. Such roles can be a team leader, development manager, planning manager, quality/process manager, support manager, etc. There can be some more roles, depending on the team size and the necessity for the project.

The company should think about the engineers who are assigned with the following positions. Because not every position is suitable for the different individuals. The better the company fills the necessary positions with the perfect engineers, the better the cooperation within the group will be.

6.6.1 Team leader

The team leader has to be very cooperative and accessible. He has to combine the different individuals into a productive team. Therefore, he has to motivate the single members and make sure that everybody does his best to achieve the required aims. If there are problems within the team, the leader should try to solve them and find an agreement between the professionals. Furthermore, if there are team members who have questions or problems, the team leader should quickly try to find a way to help them.

The management should be informed by the team leader about the progress of the project and all other relevant questions. So, the other team members are undisturbed and can do their work. Professionals like to work on their parts of the project instead of being disturbed by the management.

To find the best engineer for this position, every company can pay attention to the following characteristics which are helpful for a team leader:

- He should have fun to act as a leader.
- He should be able to make objective decisions.
- He should urge on his team members and try to get the best out of them.
- The team leader has to respect the other people.

These characteristics are maybe helpful to fulfill the position as a team leader in the best possible way. On the one hand, the other team members are more willing to follow a fair leader and are more motivated to work for him and the project and on the other hand, the company has less problems to let him lead the team like he wants to lead them.

6.6.2 Development Manager

The development manager has to guide the rest of the team to produce the required software product. He should try to spur on the others to produce a high quality software product.

Useful skills for a development manager are:

- He should like to create things.
- He must be willing to lead the development work and spur on the other engineers to do their best.
- The development manager should collect the ideas of the other team members and try to find the best possible way to fulfill the required tasks.

6.6.3 Planning Manager

The planning manager has to "help and support the team in producing a complete, precise, and accurate project plan." [Hum00a, page 248] Additionally, he has to track the progress of the team and make weekly reports for the team leader.

The planning manager has to remind his team members to track their work and measure their progress. These data are necessary to create the required reports and get a realistic overview of the project status. For this, the planning manager should have "an aptitude and interest in helping people." [Hum00a, page 249]

6.6.4 Quality/Process Manager

"A quality/process manger should be interested in and concerned about quality and should have an interest in progress and process measurement." [Hum00a, page 264] Therefore, this manager has an eye on the quality of the product and that the team keeps the quality plan. If there are problems with the quality of the code or the documentation, this manager should talk with the responsible developer and find a solution. A good documented code is much easier to read for the other developers and prevents disturbing interposed questions.

6.6.5 Support Manager

The support manager has to make sure that the team gets the suitable tools and materials which are necessary to complete the required work. For that, the support manager should be able to learn new tools and development environments in a short period of time. After that, he should be able to get it across to the other team members. This care is important to hold the motivation up within the group.

6.7 Team attributes

There are several characteristics of teams which managers should take into account. Most of the following attributes already occurred in one of the former sections but will be examined under TSP viewpoints.

The jelled team: The most effective teams are those where the team members support each other without intervention of the management and see whether someone is in trouble and try to help him. These Teams are called 'Jelled Teams' by Tom DeMarco. In those teams, the single member think as a team and not as individuals. They have become a unit where the success of the whole team is much more important than the individual.
The most important point to create such a team is the "full commitment of all the team members to their team." [Hum00a, page 294] With this commitment, teams can break records and reach the toughest aims.

The communication is important: To create jelled teams, the communication between the single team members is highly desired. Communication holds the single team members informed about the progress of the others and let them know if someone needs help. Furthermore, the professionals will become familiar with their colleagues.

Commitments: Trust is one of the key attributes in jelled teams. If there are commitments between single team members or between a team member and the rest of the team, the person who made the commitments should strive to meet them. Every team member must rely on the work of the others and be sure that everybody does his best for the group.

Participate: The single team members should participate in the team activities. If there are special team activities like extra work, free time activities or celebrations, all team members should try to participate and contribute something to the event.

Roles: The team members have different roles in the team. Each member should accept his role and identify himself with his position. If so, he can do his best and fulfill his duties in the best possible way. That guarantees an effective team where the team members recognize their tasks within their scope of duties.

Support: The members of jelled teams help each other. That means, that in addition to their official work, the single members have to find time and leisure to support the other team members if they have any problems. So they have to "participate in inspections, give advice, and contribute to design sessions." [Hum00a, page 306]

7 Conclusion

Motivation is an important aspect of todays management activities. The managers have to consider inner and outer factors which appear in their scope of duties.

The Personal Software Process helps every individual software developer to increase its productivity and to find out something about his personal strengths and weaknesses. It helps to understand the personal working behaviors and delivers data which can be used to get statistical evaluations.
The PSP can increase the motivation of the professional by providing helpful methods and procedures which increase the quality of the developer. The professional sees his personal development and will be motivated by his own evolution.
PSP is especially for the individual development and provides only a few clues for the managers to increase the motivation of their subordinates. PSP alone is more a productivity improvement program than a motivation increase process. But it can be supported by general motivation methods like mentioned in section 2 about the general motivation of people.

The Team Software Process was developed for software development teams to increase their productivity and internal cooperation. There are several ways to better the motivation of professionals by the TSP.
The management can use general methods to build up the TSP teams and different procedures in the 8 TSP levels to motivate the team and the single team members.
Although the TSP, like the PSP, is more a productivity improvement program, it does combine methods to motivate whole teams or individual members of a team. The managers can directly take influence and support their subordinates during their daily work.
It is a perfect teamwork between motivation and productivity, because motivated people are more productive and productive people are mostly motivated by experienced and engaged managers.

All in all, motivational methods can increase the productivity and working satisfaction of professionals. Experienced managers are able to create a productive atmosphere where each individual can develop the necessary skills for his/her work. They are much more engaged to bring in their abilities and to follow the management in stressful situations.

References

[Boy99] Joseph H. Boyett. *Managementguide.* Verlagshaus Goethestraße GmbH & Co. KG, München, 1999.

[Bro95] Frederick P. Brooks. *The Mythical Man-Month.* Reprinting Edition published by Pearson Education North Asiat Ltd. and China Electric Power Press, 1995.

[Car81] Dale Carnegie. *How to Win Friends & Influence People.* Revised Edition Copyright ©Donna Dale Carnegie; Simon & Schuster Inc., 1981.

[Car95] Dale Carnegie. *Durch Menschenführung zum Erfolg.* Metropolitan Verlag Düsseldorf München, 1995.

[Cos00] Russ Cosentino. *Assessment and Certification: Key to Leveraging and Optimizing Human Resources.* (2004-11-08), 2000.
`www: http://www.brainbench.com/xml/bb/mybrainbench/community/whitepaper.xml?contentId=280`.

[Cra00] Stuart Crainer. *Die ultimative Managementbibliothek.* Campus Verlag, Frankfurt/New York, 2000.

[Cra01] Stuart Crainer. *Die Jack-Welch-Methode.* Wilhelm Heyne Verlag, München, 2001.

[Cur02] Bill Curtis. *Overview of the People Capability Maturity Model.* (2004-11-07), 2002.
`www: http://www.awprofessional.com/articles/article.asp?p=25349 .`

[Dav03a] Noopur Davis. *Relating The Team Software Process to the Capability Maturity Model for Software (SW - CMM).* (2005-01-10), 2003.
`www: http://www.sei.cmu.edu/pub/documents/02.reports/pdf/ 02tr008.pdf`.

[Dav03b] Noopur Davis. *The Team Software Process in Practice: A Summary of Recent Results.* (2005-01-10), 2003.
`www: http://www.sei.cmu.edu/pub/documents/03.reports/pdf/ 03tr014.pdf`.

[DeM98] Tom DeMarco. *Der Termin.* Carl Hanser Verlag München Wien, 1998.

[Env04a] Envision. *Douglas McGregor's Theory X and Theory Y.* (2004-11-21), 2004.
`www: http://www.envisionsoftware.com/articles/Theory_X.html`.

[Env04b] Envision. *ERG Theory.* (2004-11-21), 2004.
`www: http://www.envisionsoftware.com/articles/ERG_Theory.html`.

[Env04c] Envision. *Maslow's Hierarchy of Needs.* (2004-11-21), 2004.
`www: http://www.envisionsoftware.com/Articles/Maslows_Needs_Hierarchy.html#Esteem_Needs`.

[Env04d] Envision. *The Hawthorne Effect.* (2004-11-21), 2004.
www: http://www.envisionsoftware.com/articles/Hawthorne_Effect.html .

[Env04e] Envision. *The Pareto Principle - 80/20 rule.* (2004-11-21), 2004.
www: http://www.envisionsoftware.com/Articles/Pareto_Principle.html .

[Fie02] Jeff Fields. *The Mix and Match Method.* (2004-11-09), 2002.
www: http://www.cio.com/archive/071502/peer.html.

[Gar04] Piyush Shankar Garg. *Motivation & Competitiveness Framework for Application Support Teams.* (2004-11-09), 2004.
www: http://pm.ittoolbox.com/browse.asp?c=PMPeerPublishing&r=http
%3A%2F%2Fhosteddocs%2Eittoolbox%2Ecom%2FPG071604%2Epdf.

[Gla02] Robert L. Glass. *Facts of Software Management.* (2004-11-07), 2002.
www: http://www.awprofessional.com/articles/article.asp?p=30091 .

[Gos04] Martin Gosling. *Employee Motivation.* (2004-11-07), 2004.
www: http://www.d-q-e.com/motivation.pdf .

[Gre02] Fog Greek. *Motivation during bugfixing phase ...* (2004-11-08), 2002.
www: http://discuss.fogcreek.com/joelonsoftware/default.asp?cmd=show
&ixPost=17798.

[Guc02] Randy Guck. *Managing Distributed Software Development.* (2004-11-20), 2002.
www: http://www.stickyminds.com/sitewide.asp?ObjectId=6002&Function=
DETAILBROWSE&ObjectType=ART .

[Hec74] Heinz Heckhausen. *Motivationsforschung.* Verlag für Psychologie, Dr. C. J. Hogrefe, Göttingen, 1974.

[Hen01] Joachim Hentze. *Personalwirtschaft I.* Verlag Paul Haupt Bern, Stuttgart, Wien, 7th edition edition, 2001.

[Hil02] Benjamin Hill. *Free Software Project Management HOWTO.* (2004-11-08), 2002.
www: http://www.tldp.org/HOWTO/Software-Proj-Mgmt-HOWTO/developers.html .

[Hum89] Watts S. Humphrey. *Managing the Software Process.* Addison-Wesley, 28th edition edition, 1989.

[Hum98a] Watts S. Humphrey. *Pathways to Process Maturity: The Personal Software Process and Team Software Process.* (2004-11-14), 1998.
www: http://www.sei.cmu.edu/news-at-sei/features/1999/jun/Background.
jun99.pdf.

[Hum98b] Watts S. Humphrey. *Three Dimensions of Process Improvement - Part I : Process Maturity.* (2005-01-10), 1998.
www: http://www.stsc.hill.af.mil/crosstalk/frames.asp?uri=1998/02/processimp.asp.

[Hum98c] Watts S. Humphrey. *Three Dimensions of Process Improvement - Part II : The Personal Process.* (2005-01-10), 1998.
www: http://www.stsc.hill.af.mil/crosstalk/frames.asp?uri=1998/03/dimensions.asp.

[Hum98d] Watts S. Humphrey. *Three Dimensions of Process Improvement - Part III : The Team Process.* (2005-01-10), 1998.
www: http://www.stsc.hill.af.mil/crosstalk/frames.asp?uri=1998/04/dimensions.asp.

[Hum99a] Watts S. Humphrey. *Introduction to the Personal Software Process.* Addison-Wesley, 6th edition edition, 1999.

[Hum99b] Watts S. Humphrey. *Managing Technical People.* Addison-Wesley, 5th edition edition, 1999.

[Hum00a] Watts S. Humphrey. *Introduction to the Team Software Process.* Addison-Wesley, 2000.

[Hum00b] Watts S. Humphrey. *The Personal Software Process.* (2005-01-10), 2000.
www: http://www.sei.cmu.edu/pub/documents/00.reports/pdf/ 00tr022.pdf.

[Hum00c] Watts S. Humphrey. *The Team Software Process.* (2005-01-10), 2000.
www: http://www.sei.cmu.edu/pub/documents/00.reports/pdf/ 00tr023.pdf.

[Hum02] Watts S. Humphrey. *Three Process Perspectives: Organizations, Teams and People.* (2005-01-10), 2002.
www: http://www.sei.cmu.edu/tsp/perspectives.pdf.

[Hum03] Watts S. Humphrey. *Some Programming Principles: People.* (2004-11-08), 2003.
www: http://www.sei.cmu.edu/news-at-sei/columns/watts_new/2003/4q03/watts-new-4q03.htm.

[Hum04a] Watts S. Humphrey. *The Software Quality Profile.* (2005-01-10), 2004.
www: http://www.sei.cmu.edu/publications/articles/quality-profile/ index.html.

[Hum04b] Watts S. Humphrey. *Why don't they practice what we preach?* (2005-01-10), 2004.
www: http://www.sei.cmu.edu/publications/articles/practice-preach/practice-preach.html.

[Jan04] Steve Janiszewski. *Integrating PSP, TSP and Six Sigma.* (2004-11-14), 2004.
www: http://www.google.de/search?q=cache:sW2xs2qNhUYJ:www.asq.org/pub/sqp/past/vol6_issue4/sqpv6i4janiszewski.pdf+PSP+TSP+Watts+Humphrey+motivation+OR+motivate&hl=de.

[Jay04] Arun Jayaraman. *Twenty Points for Effective Software Project Management.* (2004-11-08), 2004.
www: http://www.dmreview.com/whitepaper/WID1008853.pdf.

[Kat99] Jon R. Katzenbach. *The wisdom of teams.* HarperCollins, 1999.

[Kat03] Jon R. Katzenbach. *Why pride matters more than money.* Crown Business, New York, 2003.

[Klu04] Bruno Klumpp. *Mitarbeiter motivieren.* (2004-11-20), 2004.
www: http://www.methode.de/pm/um/pmum1.htm.

[Lie01] David J. Liebermann. *Macht macht Spaß.* Wilhelm Goldmann Verlag München; Verlagsgruppe Random House GmbH, 2001.

[McA00] Donald McAndrews. *The Team Software Process: An Overview and Preliminary Results of using disciplined practices.* (2005-01-10), 2000.
www: http://www.sei.cmu.edu/pub/documents/00.reports/pdf/ 00tr015.pdf.

[McC96] Steve C. McConnell. *Classic Mistakes: Enumerated.* (2004-11-09), 1996.
www: http://www.pslashg.org/ vool/excerpts/McConnell.html.

[Moo01] Talking Moose. *MoosePoop: Motivating Software Artists.* (2004-11-08), 2001.
www: http://talkingmoose.manilasites.com/discuss/msgReader$53.

[Nag04] Randall A. Nagy. *The Tao of the Wolves.* (2004-11-07), 2004.
www: http://www.soft9000.com/motivation1.html.

[Ner03] Friedemann W. Nerdinger. *Grundlagen des Verhaltens in Organisationen.* In Dietrich von der Oelsnitz & Jürgen Weibler (Hrsg.); W. Kohlhammer GmbH Stuttgart, 2003.

[NFP04a] The Software Productivity Consortium NFP. *PSP.* (2004-11-14), 2004.
www: http://www.software.org/quagmire/descriptions/psp.asp.

[NFP04b] The Software Productivity Consortium NFP. *TSP.* (2004-11-14), 2004.
www: http://www.software.org/quagmire/descriptions/tsp.asp.

[Pra04] Lisa Pracchia. *The AV-8B Team Learns Synergy of EVM and TSP Accelerates Software Process Improvement.* (2004-11-14), 2004.
www: http://www.stsc.hill.af.mil/crosstalk/2004/01/ 0401Pracchia.html.

[Pri01] Rukmini Priyadarshini. *Codes of Calibre.* (2004-11-14), 2001.
`www: http://www.hinduonnet.com/businessline/ew/2001/11/21/stories/0321c101.htm`.

[Ret00] Fabrice Retkowsky. *Software Project Management.* (2004-11-08), 2000.
`www: http://www.runtime-collective.com/SoftwareProjectManagement.html#6%normalsize`.

[Ros00] Lutz von Rosenstiel. *Grundlagen der Organisationspsychologie.* Schäffer - Poeschel Verlag Stuttgart, 4th edition edition, 2000.

[Rot] Rothhardt. *Praxis der Softwareentwicklung.* VEB Verlag Technik Berlin, —-.

[Rus01] Mike Russiello. *Skill Standards Unleash the Power oh Human Capital.* (2004-11-08), 2001.
`www:http://www.brainbench.com/xml/bb/mybrainbench/community/whitepaper.xml?contentId=627`.

[Rus04] Daiv Russel. *Break the Golden Rule - Rewarding Top Performers.* (2004-11-21), 2004.
`www: http://www.envisionsoftware.com/articles/Break_the_Golden_Rule_techy.html`.

[Sch01] Wolfgang Schur. *Wahnsinnskarriere.* Eichhorn GmbH, Frankfurt am Main; Wilhelm Heyne Verlag GmbH, München, 2001.

[Sch03] Kent Schneider. *Northrop Grumman Information Technology Results.* (2004-11-14), 2003.
`www: http://www.dtic.mil/ndia/2003CMMI/kent.ppt`.

[Sys99] Cisco Systems. *Optimizing Organization and Employee Capabilities.* (2004-11-08), 1999.
`www: http://www.cisco.com/warp/public/779/ibs/solutions/optimization/whitepapers/opec_st.pdf`.

[Thi01] David Thielen. *Die Microsoft Fibel.* Econ Ullstein List Verlag GmbH, München, 2001.

[Uni04a] Carnegie Mellon University. *Compilation Data for Projects Using TSP and PSP.* (2004-11-14), 2004.
`www: http://www.sei.cmu.edu/tsp/compilation.html`.

[Uni04b] Carnegie Mellon University. *Personal Software Process (PSP).* (2004-11-14), 2004.
`www: http://www.sei.cmu.edu/tsp/psp.html`.

[Uni04c] Carnegie Mellon University. *Team Software Process (TSP).* (2004-11-14), 2004.
`www: http://www.sei.cmu.edu/tsp/tsp.html`.

[Uni04d] Carnegie Mellon University. *Teradyne Results Using PSP and TSP.* (2004-11-14), 2004. www: http://www.sei.cmu.edu/tsp/teradyne.html.

[Uni04e] Carnegie Mellon University. *TSP and CMMI: A Brief History.* (2004-11-14), 2004. www: http://www.sei.cmu.edu/tsp/history.html.

[Wal02] Kenneth Walsh. *The Role of motivation and risk behaviour in software development success.* (2004-11-07), 2002. www: http://informationr.net/ir/7-3/paper129.html.

[Web04] David Webb. *Using the TSP on the TaskView Project.* (2005-01-10), 2004. www: http://www.stsc.hill.af.mil/crosstalk/1999/02/webb.asp.

[Wei86] Gerald M. Weinberg. *Becoming a technical leader.* Dorset House Publishing, New York, 1986.

[Whi94] Ken Whitaker. *Managing Software Maniacs.* (2004-11-08), 1994. www: http://www.greatideagear.com/managing.sw.maniacs.html.

[Wie99] Karl E. Wiegers. *Secrets of Successful Project Management.* (2004-11-09), 1999. www: http://www.processimpact.com/articles/proj_mgmt_ tips.html.

[Wik04] Wikipedia. *Motivation.* (2004-11-21), 2004. www: http://en.wikipedia.org/wiki/Motivation.

[Zim99] Philip G. Zimbardo. *Psychologie.* Springer Verlag Berlin, Heidelberg, New York, 7th edition edition, 1999.

Abbreviations

CMM	⇒	Capability Maturity Model
HR	⇒	Human Relationship
P-CMM	⇒	People Capability Maturity Model
PML	⇒	Process Maturity Levels
PSP	⇒	Personal Software Process
TSP	⇒	Team Software Process

Index