

# RELATIVE METRICS EFFECTING SOFTWARE DEVELOPMENT PROCESS

A Thesis submitted  
in partial fulfilment of the requirements for the  
award of the degree of

MASTER OF ENGINEERING  
IN  
SOFTWARE ENGINEERING

*Submitted By:*  
Ms. AVNEET DHINGRA

*Under Supervision of:*  
Mr. R.S. SALARIA



COMPUTER SCIENCE AND ENGINEERING DEPARTMENT  
THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY  
(Deemed University)  
PATIALA - 147 004 (INDIA)

## ABSTRACT

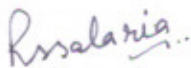
These days software organisations run like businesses, trying to deliver what they promise- on schedule and within budget. The potential customers are from varied fields like aerospace, process control, medical, petrochemical and telecommunication industries. These varied customers have varied demands that make the development process very complex. The software development organisations have to manage these complex systems in a competitive market, which cannot be understood at intuitive level. Further, a system is dynamic; that is, it keeps on changing rapidly. Failure to confront this change with adequate tools leads to a failure of management control.

For maintaining and improving the development process, process needs to be evaluated. This evaluation can be either internal or external. Both these factors are taken up individually and do not consider others viewpoint. Factors such as market demands, strategic issues and competitors are neglected when evaluating internally. On the other hand, the internal capabilities and possibilities to fulfill new market demands are not considered when evaluating external factors.

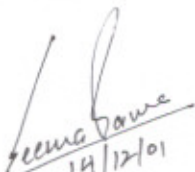
The main aim of thesis is to develop metrics that effect the software process taking operational and strategic issues in one evaluation. It also visualizes the evolution of these metrics in the software development process of different companies.

# CERTIFICATE

This to certify that the dissertation entitled "Relative metrics effecting Software Development Process", by Avneet Dhingra, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.



**(R.S. Salaria)**  
Assistant Professor,  
Computer Science and Engineering Deptt.  
Thapar Institute of Engineering and Technology  
(Deemed university)  
Patiala

  
14/12/01

**(Seema Bawa)**  
Head  
Computer Science & Engg. Deptt.  
Thapar Institute of Engg. & Tech.  
(Deemed University)  
PATIALA - 147 004.

  
14.12.2001

**(Dr. D.S. Bawa)**  
Dean, Academic Affairs  
Thapar Institute of Engg. & Tech.,  
(Deemed University)  
PATIALA - 147 004.

## ACKNOWLEDGEMENT

I would like to thank my thesis supervisor, Mr. R.S. Salaria for his constant help, encouragement and support throughout the course of this work. In spite of his hectic schedule he was always approachable and took his time off to attend to my problems and give the appropriate advice.

I express my heart felt thanks to Mrs. Rita Raina, lecturer, School of Management, Thapar Institute of Engineering and Technology, Patiala, for helping me throughout and guiding me wherever I got stuck. I am also grateful to all my colleagues for their support during this tenure.

I am thankful to my brother, Rishipal Singh Gulati who arranged for the meetings at HSS and encouraged me throughout this work. My thanks are also to Mr. Shiv Garg, Asstt. Manager, Quality, Hughes Software Systems, for extending his help and cooperation, whenever requested.

I am also grateful to my husband and my family for being patient and supportive and lending all possible help to complete my thesis work.

Avneet Dhingra

# CONTENTS

		PAGE NO.
ABSTRACT		(i)
CERTIFICATE		(ii)
ACKNOWLEDGEMENT		(iii)
<b>CHAPTER – 1</b>	<b>INTRODUCTION</b>	1-5
	1.1 Background	1
	1.2 Defining the Problem	3
	1.3 Limitations	4
<b>CHAPTER – 2</b>	<b>METHODOLOGY</b>	6-10
	2.1 Method	6
	2.2 Secondary Data Gathering	7
	2.3 Primary Data Gathering	8
	2.4 From Thorough to Basic	10
	2.5 Problems Faced During Fathering Data	10
<b>CHAPTER – 3</b>	<b>THEORETICAL BACKGROUND</b>	11-25
	3.1 Process and Its Density	11
	3.2 Management of Software Processes	13
	3.2.1 Project Time Management	15
	3.2.2 Project Cost Management	16
	3.2.3 Project Quality Management	17
	3.2.4 Umbrella Activities	18
	3.3 Organizational Culture	20
	3.4 Organizational Structure	21
	3.4.1 Decision Making Process	21
	3.4.2 Information Flow	22
	3.5 Performance Measurement Method	24

<b>CHAPTER – 4</b>	<b>PROPOSED MODEL</b>	<b>25-34</b>
	4.1 Software Processes and Perspectives	25
	4.2 Perspective at Strategic Level	27
	4.3 Perspectives at Operational Level	29
	4.4 Interconnecting the Strategic and Operational Levels	33
<b>CHAPTER – 5</b>	<b>DEVELOPMENT OF METRICS</b>	<b>35-59</b>
	5.1 Measurement	35
	5.1.1 Possible Measures	36
	5.2 Extraction of Metrics	38
	5.2.1 Supporting Metrics	39
	5.3 Definition of Extracted Metrics	41
	5.3.1 Definition of Strategic Metrics	41
	5.3.2 Definition of Operational Metrics	49
	5.3.3 Definition of Interconnecting Metrics	56
<b>CHAPTER – 6</b>	<b>ANALYSIS</b>	<b>60-71</b>
	6.1 Empirical Study	60
	6.1.1 Development Process	60
	6.2 Interpretation of Extracted Metrics	63
	6.2.1 Strategic Level Metrics	63
	6.2.2 Operational Metrics	67
	6.3 Concluding Remarks on the Analysis	70
<b>CHAPTER – 7</b>	<b>CONCLUSIONS</b>	<b>72-75</b>
	7.1 Need for the Metrics	72
	7.2 Development Processes Differ	72

	7.3 The Analysis of the Indicators is Manageable	73
	7.4 Theoretical Achievements	73
	7.5 Unbalanced Theoretical Support	74
	7.6 Preferred Refinement of the Metrics	74
	7.7 Future Scope	74
	7.8 Final Remarks	75
<b>APPENDIX – A</b>	<b>QUESTIONNAIRE AND RESULTS</b>	<b>76-85</b>
<b>REFERENCES</b>		<b>86-88</b>

# CHAPTER 1

## INTRODUCTION

The present work represents a fairly new emphasis on software engineering and business administration. It is based on the premise that software engineering has to consider strategic issues and concerns, as well as business administration has to consider operational aspects of software development. This chapter gives the basic idea and understanding of the problem.

### 1.1 BACKGROUND

In a move from the industrial era into the era of information technology, many companies have moved their focal points from selling physical products to selling knowledge and services. The company's value has therefore shifted from their tangible assets, including machines, buildings and stocks, towards their intangible assets, such as competence of their employees, processes and brand name. Traditional methods have been developed to estimate a company's tangible assets, which hence can be accounted for in the balance sheet.

Nowadays, companies in the software industry have to compete in the fast changing environment. In order to do this; smart processes need to be developed that are easily adapted to the changing environment. These processes must have an ability to learn and to apply this new knowledge when structuring the processes. There is no general way to evaluate the productivity and the quality of a certain process or how well performing a company is when it comes to recognising change and adapting to it.

According to Jim Highsmith,

*“Organizations today are faced with delivering quality software in a chaotic environment. Facing the challenge requires high levels of technical skill and ability to change, learn and thrive in the face of that chaos.”*

One of the managers of software organisation describes the situation as:

*“In this time of rapid technological change, vigorous global and domestic competition, and enormously expanding information processing capabilities, management accounting systems are not providing useful, timely information for the process control, product costing, and performance evaluation activities of managers.”*

Today, many firms have placed increased attention on planning, because they are trying to shorten the time-to-market interval by scheduling tasks in parallel using iterative and spiral techniques instead of sequential development approaches. They do so because there isn't enough time to plan all the options. Today, software managers can say with conviction whether they can do the job with the budget authorized, but they still need better requirements and detailed plans to estimate more precisely.

Despite the challenges of increased productivity and quality, and the introduction of new software development processes, most software projects continue to run into cost overruns, schedule delays and quality problems. In an environment like this, management models need to be calibrated to experience so they can achieve the desired accuracy. Firms must bring in specialized management tools and calibrate them to their experience to get improved software productivity and desired quality.

The field of software engineering focuses on different processes and methods for the attainment of the goals of the software developing companies. Business administration tends to evaluate these companies on what is visible to the customers. These two objectives are rarely united, even though they are actually two sides of the same coin. The present work is intended to bridge the gap between business administration and engineering practices.

The present work was carried out at Computer Science and Engineering Department, Thapar Institute of Engineering and Technology (Deemed University), Patiala, with the help of data obtained from Hughes Software Systems (HSS), Gurgaon.

## **1.2 DEFINING THE PROBLEM**

New times demand new solutions and the analysts need new models to value software developing companies. The fundamental question in the field of evaluation of these companies is how the companies manage their development processes. In past, two different ways of evaluation of the development process have been used, either an external or an internal evaluation. The problem with both the internal and external viewpoints is that they do not take the other viewpoint into consideration.

According to a recent study, there are seven areas to examine in order to value fast growing companies. These areas are; level of consolidation, growth, barriers of entry, trademark, management, innovation and dependence on key-personnel. This approach of performance evaluation can be compared with scoreboard in a cricket game. The scoreboard might tell whether a team is winning or losing a single match, but tells little of why the team is winning or losing. Furthermore, the scoreboard does not tell what the team is doing right or wrong in carrying out the tactics or the strategies. If the software developing company only relies on the scoreboard, then the company will not be successful in the future market.

To capture the internal key processes in organisations, different models have been developed. They often concern the gathering of different metrics to improve their software engineering management practices. The most widely known of these models is the Capability Maturity Model (CMM). CMM does not consider what to develop only how it is achieved and thus disregard the strategy of the company. Furthermore, mighty models like CMM are too bureaucratic to be effective for smaller companies.

Two areas appear to be central to be focussed upon. The first is how a company can manage the development process, and the second is how the strategic issues are connected with the development effort in order to improve development process. The purpose is to extract relative attributes for measurement of software process where strategic issues and operational issues are considered in one evaluation model. The purpose of this evaluation model is to benchmark different software developing companies with one another and to visualize the evolution of those factors.

### **1.3 LIMITATIONS**

Following areas have not been examined in this work -

- It does not address how companies choose their development process. Its basic concern is development processes in the software industry. This is considered as determined and not questionable.
- The approach is to focus on the supporting activities of the development process. A discussion on the different steps in a generic software development process is not included. However, the connections between the supporting activities and a generic development process have been discussed wherever it is appropriate.
- It does not address the motives to adopt different goals and systems for their evaluation. These are also not questionable.
- It specifically excludes the initial phases of software development including the development of basic technologies and market strategy.

- It does not cover important aspects of software development, such as specific programming languages and supporting software to the development process. This is a consequence of the fact that specific implementing technologies and methods are replaced much more rapidly than the employees are.

It is assumed that the company begins with a mission statement, identifying a target market and the core benefit the product will provide to that market.

## CHAPTER - 2

### METHODOLOGY

As the definition of the task evolved, two problems were to be tackled. Internal software development process was to be examined and a new method thereby created for an external evaluation of this process. This chapter gives an understanding of the methods and problems faced during the development of present work.

#### 2.1 METHOD

If the target of a study is to gain knowledge and understanding of a real event, a qualitative method is the best way to approach the problem, that is, study of software development process. This method was chosen to gain knowledge and understanding of how the development process is shaped at software companies today.

The quantitative research method, though is to prefer when it comes to statistical research of “non-living” instruments. You can measure feelings, conceptions and values with a quantitative method, but then you have to translate these factors into numbers and there is a risk that you loose the essential content of a feeling or a thought of the source.

To be able to create results that correspond with the purpose of this work, both a qualitative and a quantitative case study has been chosen. There are two approaches by which research can be carried on. These are:

1. **Deductive Research:** This means that the researcher outlines new hypotheses from a number of sayings available. As the hypotheses are created they are tested in the empirical context. A theory is never complete as long as the researcher can ask oneself the question "why?"
2. **Inductive Research** on the other hand means that the researcher draws general conclusions based on empirical facts. There is much material that affects the empirical study and that is why the researcher never can be a hundred percent sure that his/her conclusions are correct. In a researchers daily work the boundaries are not as distinct as stated above.

During the tenure of present work, these issues have been recognized throughout the entire process from purpose statement to conclusion. Further, conclusions have been drawn from both empirical and theoretical studies and hence both a deductive and inductive approaches have been used.

## **2.2 SECONDARY DATA GATHERING**

Internet has been an important source of information for the present work. I have surfed for papers and working essays from universities and organizations around the world on the Internet.

Books from courses in Management have also played an important role in this work. The work has been influenced by the past curriculum also.

Large amount of literature in the field of software development was obtained from HSS.

Above all access to TIET Library was of immense help.

### 2.3 PRIMARY DATA GATHERING

Of the various information-tools available, interview and questionnaire are effective techniques for eliciting information about complex subjects and for probing the sentiments underlying expressed opinions.

Interviews and questionnaires are of two types:

**Structured:** In this approach, questions are presented with exactly the same wording and in the same order to all subjects.

**Unstructured:** This is a relatively non-directive information-gathering technique. It allows respondents to answer questions freely in their own words. The responses are spontaneous rather than forced.

Structured Approach is again of two types:

**Qualitative:** Here the questions are open-ended, i.e. the responses to the questions can vary from person to person. It is an attempt to understand the world from the subjects' point of view (Kvale, 1996).

**Quantitative:** Here the questions are closed-ended, i.e. the response is to be chosen from the given choices.

Alternative used for preparing questionnaire and taking interview was structured and both qualitative and quantitative.

The qualitative interview gives the same instructions to everyone interviewed but the result among the persons interviewed differs. It allows the participant to describe what is meaningful or important to him or her using his or her own words rather than being restricted to predetermined categories; thus participants may feel more relaxed and

candid. It allows evaluator to probe for more details and ensure that participants are interpreting questions the way they were intended.

Drawback of Qualitative interviews is that the interview can become too flexible and analyzing and interpreting qualitative interviews is much more time-consuming than analyzing and interpreting quantitative interviews.

Qualitative interview conducted as first step was *Standardized open-ended Interview*. In this format, the interviewers adhere to a strict script, and there is no flexibility in the wording or order of questions. It is still considered a qualitative interview rather than a quantitative interview, because the responses are open-ended.

Thus a structure for the questions to be asked was created and from that structure there was a discussion with the person interviewed rather than a direct interview.

The quality manager at Hughes Software Systems was approached and four persons were selected for interview from middle management and up. To get a better overall view of the software development process, people were chosen from different parts of the organization. Interviews were conducted at HSS. Later on, the processing of the results from the interviews was undertaken. The interviews were prepared by sending a questionnaire to the interviewees prior to the interviews, so that they knew the sorts of topics that the interview would cover. The questions were quite open and designed to let the interviewees start thinking of the different activities performed and which routines that were followed. The questions covered a number of topics including organizational policies, product and software development.

Quantitative survey was conducted among the software developers at HSS, Gurgaon. The survey was conducted as a questionnaire that consisted of 48 questions that concerned the development process.

The questionnaire was distributed among 30 software developers working for different projects at Gurgaon office. The questionnaire has been attached as Appendix-A.

## **2.4 FROM THOROUGH TO BASIC**

People tend to have different opinions regarding issues that are included in the organization. These different opinions introduced problems at an initial stage and made it hard to focus on the real problems in the development process. Investigation was carried out for different development models and development processes.

The problems of the earlier models are that they are either focused on process technology or management. Actually both are opposite sides of the same coin. While the developed models were connecting the strategic and operational levels of the organization, a need for a basic focus on the development process was recognized.

The basic approach and the creation of the reference models gave invaluable insights into a software development process. It gave the opportunity to reflect on the interconnections between strategic and operational levels including different activities in each level.

## **2.5 PROBLEMS FACED DURING GATHERING DATA**

Secondary data was easily available as the internet access was available in the institute. Primary data gathering was a bit difficult, as the special day was to be assigned to visit the company at Gurgaon. The meeting day was to be arranged in the beginning. Once contacts were made, questionnaire was sent via e-mail to all concerned and data collected thereof. One visit was given in the last to clear off the doubts.

Unfortunately, some people were unavailable at the pre-arranged interview times, and the period for this activity had to be extended.

## CHAPTER 3

# THEORETICAL BACKGROUND

The aim of this chapter is to create the theoretical background to the metrics developed in the succeeding chapter. It reflects all levels of software development from the viewpoints of managers and middle management down to the individual level. It is the combination of the different levels that is of importance and not the specific theories in isolation.

### 3.1 PROCESS AND ITS DIVERSITY

A process can be defined as a set of operations occurring in a definite sequence that operate on a given input and converts it into desired output. In software organisations, many of these steps and activities are intellectual and organizational rather than physical. This is particularly true when discussing the development process in a software development company. These days software organizations run like businesses, delivering what they promise-on schedule and within budget. Like well-lubricated machines, they crank out the products for their customers- in the aerospace, process control, medical, petrochemical and telecommunications industries, etc.

According to Fagan,

*“ In practical application, however, it is necessary to describe the input, output, internal processing, and processing times of a process in very specific terms if the process is to be executed and practical output is to be obtained.”*

Maddox explains the importance of process in the competitive market, in his article “*Enterprise-wide Processes Boost Productivity, Competitiveness*”, as

*“ The role of ‘process’ is very significant one in maintaining productivity in the face of an aging workforce and tight labor market. ”*

The specified processes should be so as to maintain the productivity in the dynamic environment. The used development process affects the research and development of new products and reflects the undertakings of an organization.

A particular process can characterize the development processes used in each software development project. Some of the processes are predefined, but some might not be revealed until the end of the project and others will not ever be revealed.

Furthermore, the process used for each of several different types of development projects in a company might differ and could be changed between similar projects.

According to recent research in the software development, the reasons for the process diversity is manifold. Firstly, the process varies in accordance with the project’s goals and available resources. Secondly, time-to-market, cost restraints and quality are all factors that affect the actual work procedure. Lindvall and Rus argue that some of the most significant factors in the decisions of the process are the company’s size, knowledge and experience of the workforce. The application domain and the corresponding software and system requirements together with other constraints are other possible reasons.

Because all the process components, such as different activities, products and tools and all interactions between them can vary, processes will differ. The complexity of the process is evident and one general process is therefore not apparent in the software development industry. To solve actual problems in an industry setting, an organization must incorporate a development strategy that encompasses the process, methods and tools. This strategy is often referred to as Process Model. The choice of process model is

based on the nature of the project and application, the methods and tools to be used, and the controls and deliverables that are required.

### 3.2 MANAGEMENT OF SOFTWARE PROCESSES

Management can be defined informally as the art of getting work done by other people. But a more classic definition that allows a systematic study of the subject is given by Koontz and quoted below:

*“The creation and maintenance of an internal environment in an enterprise where individuals, working together in groups, can perform efficiently and effectively towards the attainment of Group goals.”*

Management functions can be classified as following:

- **Planning**, includes defining goals, establishing strategy and developing plans to coordinate activities.
- **Organizing**, determines what tasks are to be done who is to do them, how the tasks are to be grouped, who reports to whom, and where decisions are to be made. Organizational structure is to be decided only when the goals of the enterprise are clear, and this all depends on effective planning.
- **Staffing**, deals with hiring personnel for the positions that are identified by the organizational structure. It involves defining requirements for personnel, recruiting, compensating, developing and promoting employees.
- **Directing**, involves leading subordinates. The goal of directing is to guide the subordinates to understand and identify with the organizational structure and goals of organization.

- **Controlling**, involves monitoring that is, measuring and correcting activities to ensure that goals are achieved. It requires the measurement of performance against plans and taking corrective action when deviations occur.

These general functions apply to all activities going on in an organization and at all levels of organization, from the first level manager directing a few engineers to the CEO of a company directing several general managers under him.

Management has to work to achieve organization objectives. Flippo classifies these objectives as primary and secondary. Primary objectives include, creating and distributing a product or service, satisfying personal objectives of members of organization (profits for owners, salaries for employees) and meeting community and social obligations. Secondary objectives include economy of operation in meeting primary objectives and effectiveness of operation in meeting primary objectives.

Management has to look into the processes and activities of the organization and work for improving the processes. It should be clear of the fact that if processes are improved, products also get improved.

For the development process to achieve its final objective, a holistic approach must be used taking into account the elements that would help to comply with customer's expectations. Management is expected to know the competition in market, know the customer's needs and align the product with the corporate mission of organizations. All this requires time, effort and commitment. A business plan is built around the corporate mission and effectively passed on to all those working for the organization for appropriate step. Feedback is taken from various levels of organization and it is ensured that all are working towards a common goal.

Management has to assure that software development process is aligned with project portfolio established at strategic level of an organization. To carry out the content of the aggregated project plan efficiently, a good project management is needed. Management

decisions have a strong impact on the technical aspects of software engineering. Project manager's task can be summarized as "plan the work and work the plan".

Project management is concerned with three major tasks as shown in the figure 3.1. The triangle consists of two resources, time and cost. The third factor, quality, represents what the project delivers. Another important factor that is inter-related to time and cost and exactly determines quality is requirement. The interactions between time, cost, quality and requirement are important because they are the linkages between the internal and the external factors. The triangle implies that there is trade-off between the different dimensions. The center of the gravity of any project will always be somewhere within the triangle. All these are somewhere or the other related to requirements.

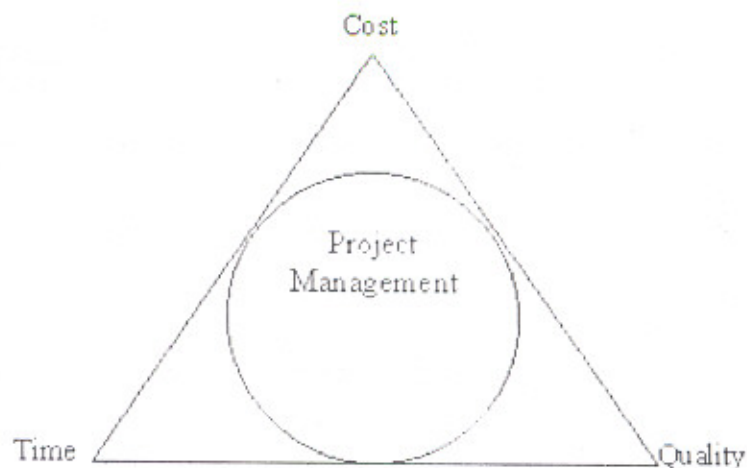


Figure 3.1 : The project Triangle

### 3.2.1 Project time management

Project time management includes the process required to ensure timely completion of a project. Project managers have to define and estimate the time resources needed for the different activities to produce the project deliverables. Through acknowledging their interdependencies an activity sequence can be established. With the help of this work break down structure and expert opinions or analogies managers can achieve better time estimates. When the activity sequence and its time duration is established, project

managers have to assure that the right resources are available at the right time. The outcome of this process is a time schedule with appropriate milestones attached to it. Time has become an important competitive factor. Especially as product life cycles have shrunk and missing the window of opportunity could be devastating for a project and the company as a whole. If deadlines are not kept, "bad will" can be incurred externally and due to project interdependencies the aggregated project plan could fall apart. To reach the window of opportunity, the development time can be decreased by, for example, parallel development, prototyping or reuse of already existing software.

### **3.2.2 Project cost management**

Project cost management aims to assure that the project is completed within the approved budget. If the budget is not kept this can undermine the existence of the project and decrease the company's overall profitability. Project management is primarily concerned with the cost of the resources necessary to complete the different activities within the project schedule. However, managers should also consider effects of project decisions on the cost of using the product. This broader view is commonly known as "life-cycle costing".

With the resources needed for carrying out the project activities identified and the time estimates for the different activities, managers can estimate the project cost. This estimation will serve as a foundation for the project budget, which allocates the overall cost estimates to individual work items. The budget serves as a baseline for measuring project performance and if any changes are made, either positive or negative, the reasons why they have occurred must be identified in order to assure organisational learning.

Decreasing the costs of software development is primarily realized by decreasing the software development labor, and therefore cost reduction will mostly aim at executing the software development process more efficiently. A first step to this is identifying current effort and expenditure. Examples from practice in which cost is expressed as cost per source of line, cost per function point, cost per life-cycle phase, cost per product or cost

per subsystem. Cost estimates in the software industry tend to be less accurate than for those in manufactured goods. One reason is that projects are less standardized, and cost information that has been accumulated for similar work is therefore not a valid basis for comparison.

### 3.2.3 Project Quality management

Project quality management includes the processes required to ensure that the project will satisfy the needs for which it was undertaken. According to PMBOK, it includes all activities of the overall management function that determine the quality policy, objectives and responsibilities and implements them by means such as quality planning, quality control, quality assurance and quality improvement, within the quality system.

Determining what truly represents software quality in the customer's view can be elusive, it is equally clear that the number and frequency of problems and defects associated with a software product are inversely proportional to quality of software. Software problems and defects are among the few direct measurements of software processes and products.

Project quality management aims to assure the quality of the product being supplied to the customer.

Ghezzi classifies quality as *external* quality and *internal* quality. External qualities are visible to the users of the system, while internal qualities are concerned with the developers of the system. In general, users of the software only care about the external qualities, but it is the internal quality which deals largely with the structure of the software, that help developers achieve the external qualities. Quality management group has to look into both.

Quality control activities are conducted throughout the project development cycle to ensure that work product meets the standards. These activities include reviews, tests, walkthroughs and inspections. Necessary feedback is taken up from the user and the

developer. The combination of measurement and feedback allows tuning the process when the work products created fail to meet their specifications.

Quality assurance activities provide management with the data necessary to be informed about product quality, thereby gaining insight and confidence that product quality is meeting its goals. Of course, if the data provided through quality assurance identify problems, it is management's responsibility to address the problems and apply the necessary resources to resolve quality issues.

### **3.2.4 Umbrella activities**

Other umbrella activities undertaken by management are:

- Risk Management
- Configuration Management

#### **Risk Management**

Risk management includes the processes concerned with identifying, analyzing, and responding to project risk. It includes maximizing the results of positive events and minimizing the consequences of adverse events.

Project risks threaten the project plan, that is, if project risks become real, it is likely that the project schedule will slip and that costs will increase. Project risks identify potential budgetary, schedule, personnel, resource, customer and requirement problems and their impact on a software project. Technical risks threaten the quality and timeliness of the software to be produced. If the technical risk becomes a reality, implementation may become difficult or impossible. Technical risk factors can be specification ambiguity, technical uncertainty, technical obsolescence, and "leading-edge" technology. Business risks threaten the viability of the software to be built. These often jeopardize the project or the product. Business risks include market risk, strategic risk, management risk, and budget risk.

Risks can be coped up using either reactive or proactive strategies. Most commonly, software team does nothing about risks until something goes wrong. At the most, resources are set aside to deal with them, should they become actual problems. This forms the reactive strategy for fighting risk.

A proactive strategy begins long before technical work is initiated. Potential risks are identified, their probability and impact assessed and prioritized by the impact, if they occur.

Risk management analysis can absorb a significant amount of project planning effort. Identification, projection, assessment, management and monitoring, all take time. But the effort is worth it. Quote SunTzu, a Chinese general who lived 2500 years ago, *"If you know the enemy and know yourself, you need not fear the result of a hundred battles"*. For the software project manager, the enemy is risk, which needs to be identified well in advance and mitigated.

## **Configuration Management**

Software configuration management is a set of activities that have been developed to manage change throughout the life cycle of computer software. Change is inevitable when computer software is built. The change occurs because of varied reasons. First, new business or market conditions dictate changes in product requirements or business rules. Second, new customer needs that demand modification of data produced by information system, functionality delivered by products, or services provided by computer based systems. Third, reorganization and/or business downsizing that cause change in project priorities or software engineering team structure. Fourth, budgetary or scheduling constraints that causes a redefinition of system or product.

Because change can occur anytime, configuration management activities are developed to identify change, control change, ensure that change is being properly implemented and report change to others who may have interest.

As the software development progresses, the numbers of software configuration items grow rapidly. A system specification spawns a software project plan and software requirement specification. These in turn spawn other documents to create a hierarchy of information. Configuration management is the discipline of coordinating software development and controlling the change and evolution of software products and components.

According to PMBOK, configuration management is a documented procedure used to apply technical and administrative direction and surveillance to:

- Identify and document the functional and physical characteristics of an item or system.
- Control any changes to such characteristics.
- Audit the items and system to verify conformance to requirements.

In many application areas, configuration management is a subset of the change control system and is used to ensure that the description of the project product is correct and complete. However, in some areas, it is used to describe any rigorous change control system.

### **3.3 ORGANIZATIONAL CULTURE**

Organizational culture is a common perception held by the organization's members. It is a system of shared meaning held by members that distinguishes the organization from other organizations. According to Robbins, it is a set of key characteristics that the organization values. Organizational culture has a strong impact on the employee's behavior.

Culture performs number of functions within an organization. First, it has a boundary-defining role; that is, it creates distinctions between one organization and others. Second, it conveys a sense of identity for organization members. Third, it facilitates the generation of commitment to something larger than one's individual self-interest. Fourth, it enhances social system stability. It helps hold the organization together by providing

appropriate standards for what employees should say or do. It serves as sense-making and control mechanism that guide and shapes the attitudes and behavior of employees. From the employee's viewpoint, culture is valuable because it reduces ambiguity.

The ideas, values, norms and rules of the company are passed on through informal channels of communication. In a strong culture, the organization's core values are both intensely held and widely shared. The more the members, who accept the core values and the greater their commitment to those values is, the stronger the culture is. Such type of culture has a high degree of sharedness and thus high behavioral control.

Computer oriented companies have "work hard/play hard" culture. This culture has low risk but fast feedback. The organizations are market oriented and the customers' needs are in centre. Organizational culture is a firm resource that is potentially very valuable and of great strategic importance.

### **3.4 ORGANIZATIONAL STRUCTURE**

Organizational structure defines how job tasks are formally divided, grouped, and coordinated. The knowledge-oriented company must develop an organization that can cope with an ever-changing environment. This creates difficulties in transmitting proper and required information, both for leaders and employees in the organization. One of the most vital processes in the organization is the decision making process. Of course, the leadership plays an important part in this process and it is vital that all aspects of the organization are considered when making decisions. The organizational structure also effects the communication process within the organization.

#### **3.4.1 Decision making process**

Making the right decisions on managerial and non-managerial level, in the company will always be important and is vital for the company to stay alive in a competitive market. How these decisions are taken is necessary to understand and there are many reasons why it is so important to consider them. First, there can be severe consequences for the

company if decision is not considered in a reasonable way. Second, many decisions are made almost automatically. If there is a process of decision-making, then groups and individuals have a greater possibility to affect the decisions made.

Authority decisions are made by a controlling part of a group or the leader. The group follows the leader's decision and keeps quiet about what they really think. Supporting decisions occur when one person makes a suggestion and one or two other persons support this decision but not the whole group. Majority decisions are very common in today's companies. But majority decisions have a difficulty in the sense that the minority in the group often has very hard to accept the decision of the majority. There are also different kinds of decisions where a group is united. Either you discuss a suggestion until all members of the group are united or you could change the decision so that it satisfies all the members. Many of the decisions in an organisation are made at the operational level. Regardless of the dimension of the decisions, they all play an important part in the development of the organisational structure. Some authors state that one of the most important prerequisites for an organisation survival is the ability to make rational decisions. The rational decision process is characterised by the following steps

- Define the problem
- Establish the goals that needs to be obtained
- Seek different alternatives that can help you to achieve the goals
- Value the alternatives by analysing the consequences if they are executed
- Chose the alternative that best fulfils the goals
- Follow up and control the actual result of the decision

### 3.4.2 Information Flow

*“ An idea, no matter how great, is useless until it is transmitted and understood by others.”*

The above statement by Robbins states the importance of good communication (information flow) within an organization. Communication is the fundament for accomplishing co-ordinated activities.

Communication serves *four major functions* within an organization: control, motivation, emotional expression, and information. It acts to *control* member behavior in several ways. Organizations have authority hierarchies and formal guidelines that employee is required to follow. It helps in *motivation* by clarifying to employees what is to be done, how well it is to be done, how well they are doing and what can be done to improve performance. The communication that takes place within the group is a fundamental way of showing frustrations and feelings of satisfaction. Thus, communication provides a release of *emotional expression* of feelings and for fulfillment of social needs.

It provides the *information* that individuals need to make decisions by transmitting the data to identify and evaluate alternative choices.

Organization structure is often tightly linked with the communication structure of the organization. The information needs of the stakeholders should be analyzed to develop a methodical and logical view of their information needs and sources to meet those needs. The analysis should consider methods and technologies suited to the project that will provide the information needed.

The technologies or methods used to transfer information back and forth among the project elements can vary significantly from brief conversations to extended meetings, from simple written documents to immediately accessible on-line schedules and databases.

Information can be shared by team members through a variety of methods including manual filing systems, electronic text databases, project management software, and systems which allow access to technical documentation. Information may be distributed through project meetings, hard copy document distribution, shared access to networked electronic databases, fax, electronic mail, voice mail and video conferencing.

### 3.5 PERFORMANCE MEASUREMENT METHODS

The competitive environment requires that managers have timely and accurate information to help them make processes more efficient and more customer-focused. According to Sage and Jensen this information can only be obtained through systematic measurement and the development of models that support the processing of the information.

Currently many software development practitioners and researchers are involved in software process improvement. Several improvement models, methods and techniques are available, and they are divided into two major approaches, either top-down or bottom-up.

The top-down approach is mainly based on assessments and benchmarking. CMM is example of this approach. In the other case, the bottom-up approach, measurement is mainly applied as the basic guide for improvement. The Goal-Question-Metric (GQM), represents this approach. Although both approaches complement each other to a great extent, they are rarely applied together.

It is recommended to use a top-down approach to create a first overview on the status of a software organization. Based on the result of this assessment, improvement goals should be identified that suit the organization in the best possible way. Applying some form of activity measurement should create guidance of the improvement activities towards these goals. Measurement provides an overview and the opportunity to evaluate whether implemented improvement activities toward clearly defined goals, a so-called "goal-oriented" measurement method should be selected to implement measurement. According to the corresponding literature, the CMM helps organizations to improve the maturity of their software processes through an evolutionary path from ad hoc and chaotic to mature and disciplined. A low level of maturity incorporates a high level of risk in applying a process. As organizations become more capable, risks decrease and productivity and quality are expected to increase. Each maturity level is intended to add further

enhancement that software organizations typically master as they improve. The expected usage of CMM is to determine the most important areas for immediate improvement. For this purpose the CMM literature argues that it provides an assessment method to objectively and consistently assess the capability of software organizations and place them on one of CMM's five maturity levels. After executing these actions organizations can then return to re-assessing the established process, as process improvement is a continuous cycle.

Many measurement programs fail due to that they measure what is convenient to measure without assuring that the data extracted is useful to the developers of the software. To avoid this, GQM is designed with the goals of the project in mind. The GQM approach provides a framework involving three steps:

- List the major goals of the development or maintenance project.
- Derive from each goal the questions that must be answered to determine if the goals are being met.
- Decide what must be measured in order to be able to answer the questions adequately.

## CHAPTER - 4

### PROPOSED MODEL

This chapter gives in detail the model referred for extraction and development of metrics. This model has been proposed after extensive theoretical study undertaken.

#### 4.1 SOFTWARE PROCESSES AND PERSPECTIVES

The software development process in reality is too complex to be characterized in exact terms. In order to control the process performance, it needs to be managed. Certain aspects are more complex than others are. These can be divided into four different perspectives- *resource planning, organizational structure, specification process and information flow*. All these four perspectives can be studied at two levels- Strategic level and Operational level.

Both levels work for the same goal, that is, developing software and follow different processes to achieve it. In software organization, management interacts with clients, understands their requirements, develops strategy to meet the desired results, allocates resources to different projects being worked upon in the organization and monitors the development process.

Operational level is busy implementing the tasks assigned, plan how to meet the end desired results taking into consideration the standards and procedures defined by top-level managers. It requires good project management and implementation skills.

All the above mentioned four perspectives form the basis of the product development process and are dependent on each other. Figure 4.1 shows different interactions that have to be considered in order to create the desired product performance.

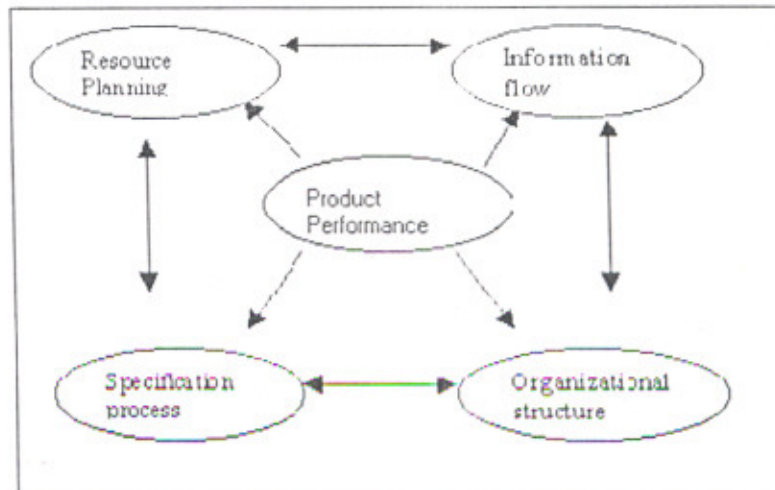


Figure 4.1: Potential interactions between process driven factors

## 4.2 PERSPECTIVES AT STRATEGIC LEVEL

At strategic level, all four aspects need to be controlled to achieve the best results in software development process. As the tools and platforms are continuously changing in the software industries, the implementation techniques play a less significant role to the overall picture.

### Resource planning

It involves determining what physical resources (people, equipment, and materials) and what quantities of each should be used to perform project activities. The resources within a company need to be matched to the opportunities that arise in the market. The policies of the performing organization regarding staffing and rental or purchase of supplies and equipment must be considered during resource planning. As these resources often are shared between different functions and projects in an organization, management of these resources is essential. According to Grantt, the greater the rate of change in the company's external environment, the more likely the internal resources is to provide a secure foundation for long-term strategy.

## Organisational structure

One might argue that an organisation, where the roles are specialized and thus has a clearly defined division of tasks, is to be preferred. The consequences of specialization are rather that the company gets bureaucratic and stiffening than that the company is nurturing the flexibility of the organisation. The bureaucracy is characterized by highly routine operating tasks achieved through specialization, very formalized rules and regulations, tasks that are grouped into functional departments, centralized authority and decision making that follows the chain of command. The disadvantage of bureaucracy is that there is no scope of modification of rules and regulations.

This aspect has to be considered to be able to fit the software to customer demands. Most organisations now recognise that various functions of the company are not independent. Instead they are closely interconnected. This interconnectedness implicates that employees from various functions must work towards a common goal to achieve a prosperous company.

## Information Flow

For launching a successful product in a market, it is essential that there is overall understanding throughout the company of both customer demands and the resource situation. To be able to compete in the market it will not suffice that only the market department knows what the customer demands are. How management provides information on the actual standings on the market and the future direction of the company reflects the efficiency of the information flow within the company. To create and implement a business strategy is not an issue for the management alone. The realization of the strategy has to be supported by a certain individual and departmental educational level inside the company by those who have to execute it. Otherwise, the strategy implementation is futile and thus are visions and strategy statements not translated into actions at the operational level. A broad communication from top to bottom and vice versa is essential to share the strategy and critical objectives with all employees if the strategy of the company is to succeed.

## Specification Process

The principle challenge in any industry is to develop a system that meets the need of the customers. According to recent research, incomplete specifications are still a considerable source of misunderstandings between the customer and the individual developer. To solve this IEEE tried to define completeness of the specification process in 1984 in the *Guide to Software Requirements Specifications* as:

*"A specification is complete when all the requirements relative to functionality, performance, constraints on system structure, attributes and external interfaces are written and if all the terms used in these requirements are defined."*

The initial effort of the specification process at strategic level is thus to identify a set of representative users, who will provide basic software requirements proposals. After identifying users, the strategy is decided as to how to translate requirements at the local level, how to write these requirements on what to develop and how to achieve these requirements.

Specification process also handles the issue of controlling that the requirements are fulfilled and thus includes the areas of verification and validation. The customers continuously demand new and extended requirements on the product. It is therefore crucial to the specification process that the requirements held by the software developing company itself, are primarily the same as those held by the customers. The management of the company is significant to the specification process while these set the direction and articulate the goals of the product.

### 4.3 PERSPECTIVES AT OPERATIONAL LEVEL

An awareness of corporate goals at management level is not enough to manage the employees' behavior. Somehow, the company's overall strategic objectives and measures must be translated into objectives and measures for operating units and individuals. The development processes differ between companies. The reason for this is that there are no

development process or standards of behavior that can be reasonably applied at all times and in all places. Instead, the question of whether or not a development process is adequate must be answered relative to the market structure and organisational size in which it takes place.

The approach at this level will focus on the supporting activities of the software development process. By supporting activities we classify those activities that are vital to manage in order to create a software product in accordance to actual customer demands. The supporting activities are resource planning, organisational structure, specification process and information flow. These supporting activities assist the whole software development process from concept development to integration- and beta -testing. The ideas are presented in figure 4.2 as the proposed value-chain in a software developing company. The relationships between the steps are merely linearly dependent.

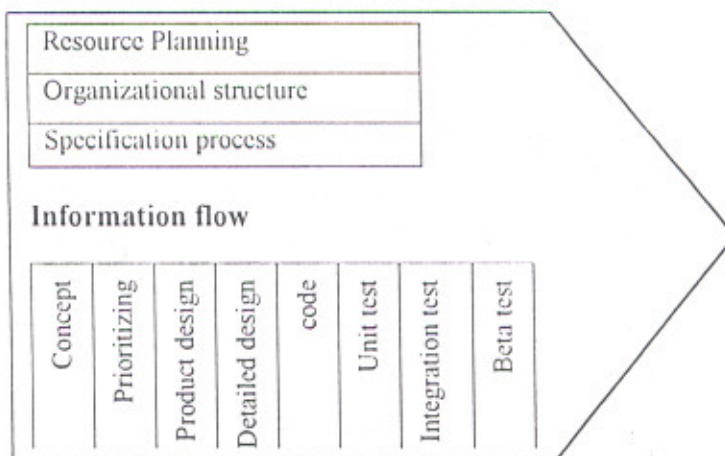


Figure 4.2: A proposed software development value-chain

### Resource planning

Resource planning at this level is the stage where the strategy and policies are translated into activities to be undertaken over a predefined amount of time. The projects that have been derived from the strategic level can vary in size from small projects as building special editions of the software product to new product development.

Making optimal use of available resources, by planning and tracking, is vital, as software development is an expensive task. To control the resources a couple of obvious obstacles must be handled. First, specialised skills are often required to complete a project and the aim of the company is to make sure that essential skills are available. Additionally, one or more key team members may have other unavoidable responsibilities. It is thus important to carefully consider how and when these resources have to be available. Another obstacle to consider is that project work tasks are not constant over time. This is especially important to have in mind if several projects are to be concluded on a common release date and if they use shared resources.

### **Organizational structure**

The organisational structure includes a company's formal reporting structure, the controlling and co-ordinating systems, as well as informal relations among projects within the company and between the projects and its customers. The organisational structure and the management style that operates within the company are one of the most obvious ways of discovering the company culture. The culture is additionally dependent on a number of factors including the size, history and tradition, technology, environment, leadership and management style.

The culture in a software development company distinguishes the effort of the employees to use developed standards and methods. This is, however, not specific to a software development company and one conclusion is that it is more or less a matter of attitudes or a question of mental resistance. This implicates that spending time and effort on human resource management and employee motivation amplifies the productivity of the company and its projects.

### **Information Flow**

Once strategic goals have been set and communicated, managers at the project level are responsible for producing plans to achieve them. In addition, the process by which the projects are conducted should then be communicated throughout the project.

Communication of plans and processes are critical to the effective implementation of projects. An environment of open communication and sharing of information is seen as essential to achieve project goals. This requires high levels of trust, respect and confidence.

Successful innovation and product development have been linked repeatedly to a creative corporate climate. In order to form this creative climate, communication between different functions throughout the entire company and informal and lateral interactions between them are essential. By division of the company functions into different areas of the company, building the communication is less effective and will not support informal communication. However, by gathering members in the project team from different functional areas this can be avoided.

### **Specification process**

A critical step in the software development occurs when the strategic goals specified in the requirements are transformed into how the requirements are to be achieved. Not until this has occurred can the feasibility of the software product be analyzed and the requirements be verified. The requirements are then the initial statements of all feature deliverables.

In current practices, the verification and validation of the developed software products is an inconsistent activity, generally not supported by refined software tools. At the best a set of test cases are defined at the beginning of the project, before the initial design has taken place. Typically the test cases are developed past implementation.

This is a crucial area to manage and an early definition of test cases limits the question on what to develop and the effort is thus on implementation. To document the requirements and tracking the changes due to conditions during the project are important issues if the software product is to be used for further development and enhancements.

By documenting these aspects the efforts in succeeding development projects are optimized and the management of the software product is simplified.

#### 4.4 INTERCONNECTING THE STRATEGIC AND OPERATIONAL LEVELS

The development processes of today generally do not provide adequate information to support organisational learning and improvement. In a turbulent competitive environment the managers and operators need to have timely and accurate information in order to make the performance of the processes more efficient and customer focused. The question of whether the performance of the software development department is determined by the operational level or the strategic level might be debatable.

A twofold link between the strategic level and the operational level is presented and the approach is to describe the interconnections between them. In an overall context, the visions and strategies of the top management together with the executioners are important to the success of the organisation.

The feedback process within a company creates interconnection between managerial and operational level and probes into the future. Interconnecting these levels concerns learning from past experiences. Feedback can be controlled through evaluation of the software development process and product. Unfortunately, few companies institutionalize a process for learning from their mistakes. The reasons appear to be made up of deeply rooted cultural and institutional factors that obstruct organizational change. At the same time, there is the need to continually update software competencies.

One important issue for the software development department to handle is how to link current and future projects in the product portfolio. As previously described, the strategic and operational levels are interconnected and no cause and effect can be distinguished. Additionally, there is a risk of sub-optimization in the software development if the overall picture of the company can not be viewed. In figure 4.3 the links between current and future projects and strategic and operational levels are described. The model shows that both strategic and operational levels must be managed in order to create the required

software. That is, for the possibility to value the software development process, it is important to investigate how current projects affect the possibility to decide on what tasks to undertake in the future.

The feedback loop starts with the specification of the strategic goals of the company and an overall plan on which projects to undertake, the project portfolio plan (1). The second step is to plan the individual projects according to the resources available and the objectives from the top management (2). The project is then divided into different phases, which are individual to each software development company. However, both requirement elicitation and verification and validation are vital to the development process and are represented by step 3 and 4. As the testing and initial customer feedback is gathered, the overall project portfolio plan is affected in step 5. If the steps from project level to verification and validation are not handled correctly, failures are rectified at strategic and operational levels.

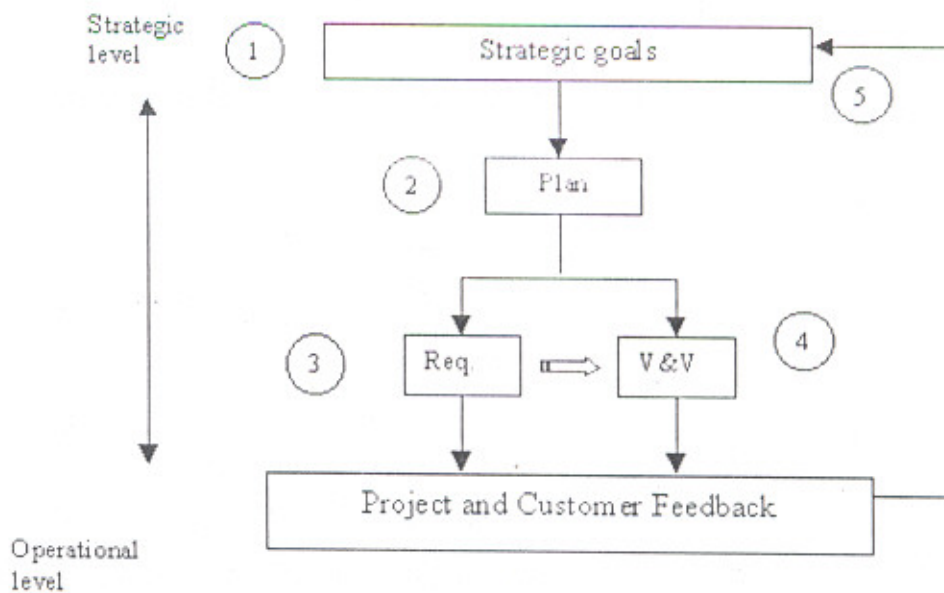


Figure 4.3: Feedback loop of a software developing company

# DEVELOPMENT OF METRICS

The ability to express things meaningfully in numbers is an essential component of any engineering discipline. Metrics and measurement have the potential to revolutionize Software Engineering. This chapter discusses the metrics used to evaluate the software development process.

### 5.1 MEASUREMENT

Lord Kelvin once said:

*"When you can measure what you are speaking about and express it in numbers, you know something about it; but when you cannot measure, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind."*

An entity is an object or an event in the real world and there is often a need to describe the entity by identifying characteristics that are important to us in distinguishing one entity from another. The area of measurements is intended to capture information about the attributes of these entities. An attribute is a feature or property of an entity and typical attributes include cost, time or quality. One possible measurement definition is:

*"The process by which numbers or symbols are assigned to attributes of entities in the real world in such way as to describe them according to clearly defined rules."*

To measure with explicitness and exactness, it is of utmost importance to clearly define the entities being measured.

Measurement is needed for assessing the status of the company's projects, products, processes, and resources. Companies need to document trends, the magnitude of corrective action, and the resulting changes. In other words, they must control their projects, not just run them. Without measurement, methods such as CMM and GQM would lose their meaning. It is rightly stated that:

*"You cannot control what you cannot measure"*

Furthermore, every measurement action must be motivated by a particular goal or need that is clearly defined and easily understandable. That is, it is not enough to assert the need to measure to gain control. The measurement objectives must be specific and tied to the needed knowledge of managers, developers and users. Thus, these objectives may differ according to the kind of employees involved and at which level of the organisation they are generated. But it is the goals that tell us how the measurement information will be used once it has been collected.

There are many different arguments why an organisation should make the effort of measurement and the attached investments in capital and resources. One implication is that measurement can aim to establish the company position and its development. Without this knowledge, strategic and operational decisions get insecure and hard to implement in the organisation. Equally important, it is to internally communicate the company's position and development in order to create a shared motivation to achieve goals and improve the product, the process and the company culture.

### 5.1.2 Possible measures

Time, cost and quality are the most common aspects that are measured in an organisation and these aspects are also reflected in the project triangle (Figure 3.1).

The reason why time is a common attribute is because it can be used in order to assess internal efficiency as well as external efficiency. The implication of measuring time is the

possibility to shortening cycle-times and to make cost reductions and thus increase customer satisfaction.

Productivity of a process can be assessed by the relationship between its input and its output. Typical outputs are physical volume and accrued profit. Inputs are investments, human resources or the process total cost. Since cost can be hard to trace to specific processes or activities methods such as Cosmo models can be of great help.

The quality is an attribute of a process's external effectiveness. There are several dimensions of what characterises quality. They can only be assessed indirectly and they all ultimately aim to capture customer satisfaction. Some common metrics is the number of customer complaints, time to solve a customer problem, human effort expended, schedule conformance or mean time between failures. Figure 5.1, gives the determinants for software quality and organisational performance.

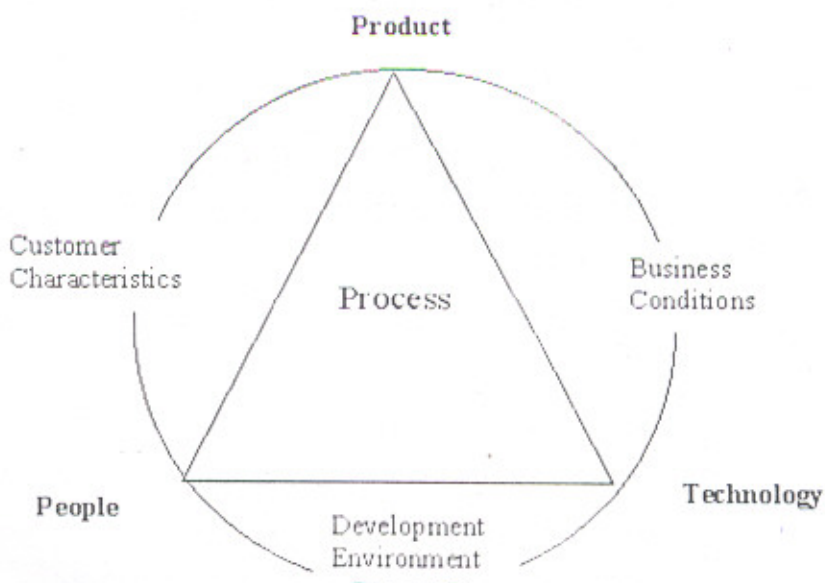


Figure 5.1: Determinants for software quality and organizational effectiveness

The complexity of the product and the technology can have a substantial impact on quality and team performance. The process of development is dependent on type of

communication with the customer, business rules, and development environment, which comprises of tools used.

## 5.2 EXTRACTION OF METRICS

A software metric is any type of measurement, which relates to software system, process or related documentation. Another definition says that metric is a quantitative measure of the degree to which a system, component or process possesses a given attribute. An attribute is a quality concept or resource concept, which describes a system quantitatively. Attributes are characteristics of the system, which represent constraints.

Metrics fall into two categories namely control metrics and predictive metrics. Control metrics are those used by management to control the software process. Estimates and measurements of these metrics can be used in the refinement of the project planning process. Predictive metrics are measurements of a product attribute that can be used to predict an associated product quality. A set of metrics can be collected so as to get an insight into the software process, software project, or product itself. This helps managers to adjust the process, the project or the product to make the things better. According to Mossberg there are some fundamental characteristics that metrics should fulfil. First, they should comprise information and second, what is measured should be of high value for the total picture.

Behind every metric there ought to be a well-formulated hypothesis. This hypothesis should contain information on what the user expects the metric to show. It can be compared to a model for how to perceive the metric. The hypothesis should be continuously evaluated and possibly changed in line with the experience of it. In the beginning, a metric should be dealt with in isolation, and only in later stages can several metrics be merged together with one another. Naturally, the more the metrics those aim to highlight a specific entity the better, since this will increase the richness of the measurement.

To increase the usefulness of the metric, the metrics should live up to the following criteria:

1. They should be *significant* compared to what they want to establish. That is, there should be a clear connection between the metric and the entity.
2. They ought to be *quantifiable*. However, these criteria can be modified to a graded scale, relative or ordinal. Even subjective evaluations can be quantified by, for example, points or grading system.
3. The metrics should be *independent* from one other and if there is a strong dependence between different metrics these should be presented together and their dependence clearly described.
4. The metrics should be *useful over a longer time period*. This means that they ought to be available and constant in their definition over time.
5. Metrics should be “*objective*” in the sense that everybody involved should agree on the meaning of the metric even though the opinions of the result may differ.
6. The metrics should be *easily understood* and coupled to the original data even though this aspect is relative due to the competence of the individual.

### 5.2.1 Supporting Metrics

As discussed in the previous chapter, organisation has to give attention at four perspectives: resource planning, organisational structure, information flow and specification process. These perspectives are further divided into three levels: strategic level, operational level and interconnecting the strategic and operational levels.

The metric is different measure of certain aspects of the perspectives and several metrics are interconnected, which is indicated in the definitions.

Thirty-five metrics have been developed and presented by name, perspective and level in next section and henceforth.

To evaluate the development process according to the definitions of the metrics stated, the questionnaire in appendix A was created. The questionnaire reflects different aspects of the development process and each indicator analyzed is supported by one or several

questions in the questionnaire. When the questionnaire are used in other organisations the questions have to be changed in order to align to organisational specific vocabulary and business environment. The basis of the questions, however, remains the same.

### Strategic Level

<b>Resource planning</b>	<b>Organisational structure</b>	<b>Information flow</b>	<b>Specification process</b>
<i>Overall project Status</i>	<i>Company reactiveness</i>	<i>Strategic information</i>	<i>Development goal</i>
<i>Reactive or proactive action</i>	<i>Department Representation</i>	<i>Priority of Information</i>	<i>Priority of requirement</i>
<i>Estimating resources</i>	<i>Project priority</i>	<i>Formal meetings regarding company strategy</i>	<i>Requirement change window</i>
	<i>Customer specific dependant</i>	<i>Departments alignment to overall strategy</i>	<i>Requirement cost</i>
			<i>Documentation standard</i>

### Operational Level

<b>Resource planning</b>	<b>Organisational structure</b>	<b>Information flow</b>	<b>Specification process</b>
<i>Release date Accuracy</i>	<i>Support of decisions</i>	<i>Project co-ordination Requirements</i>	<i>motivation</i>
<i>Release workload</i>		<i>Individual dependence</i>	<i>Requirements testability</i>
<i>Individual burden</i>		<i>Supportive tools alignment</i>	<i>Building software</i>
<i>Clarity of project Interdependence</i>			<i>Degree of automated test</i>

## Interconnection of two levels

Resource planning	Organisational structure	Information flow	Specification process
<i>Resource evaluation</i>	<i>Documentation alignment to actual work procedure</i>	<i>Organisational learning</i>	<i>Feedback on quality at department level</i>
<i>Time report system</i>			<i>Feedback on quality at individual level</i>
			<i>Overall system knowledge</i>

### 5.3 DEFINITION OF EXTRACTED METRICS

The extracted theories and the developed reference models have influenced the definitions defined in this chapter. These indicators can be used when analysing different development processes and are analysed in the following chapter.

#### 5.3.1 Definition of Strategic metrics

##### Resource planning

##### Overall project status metric

*Relevant Questions: 13,14,30*

Overall project status is concerned with how the software department handles its current aggregated project plan, not what types of projects it should contain. Therefore it will be of interest to map what kind of information that this plan has attached to it and how often this information is formally updated.

Information that could be of interest is related to resource allocation amongst the projects like the projects' budgeted time and cost, what human resources and during what time period they are assigned to a specific project, who is the project manager, what project dependencies are there and what is the current status of each project. The status of a

project includes how much time that has been consumed by the project so far and thus the amount of sunken costs and finally how much that is estimated to be left of the project.

The relevance of this indicator increases with the size of the development department. The more people that have to be updated on the progress of the projects the more formal the process of controlling the aggregated project plan becomes. In smaller departments it might be enough with an informal approach towards management since the complexity is more manageable than in larger departments.

### **Reactive or proactive development metric**

*Relevant Questions: 25,26,30*

This indicates whether the software development department is reactive or proactive. It focuses on the amount of resources that the company spends on customer support reports, development related to that the product has been beta tested and development occurring before beta-testing. The underlying idea is that one can get an approximate evaluation of these three major activities and how reactive/proactive the development process is. By taking the relationship between software development and the two remaining areas one can get a first impression on how well this development works. Similarly, one could get a first impression on how effective the beta-testing is if one looked at the amount of resources spent on customer support reports. It is important to note that the primary goal is to remove the need to spend resources on customer support reports since errors found at this time will affect the customers directly and will cost more to correct.

The relevance of this indicator gets higher, the longer a company's software product has been around as it is believed that a product's complexity will increase with its age and that this indicator might show how well the company has dealt with this complexity.

### **Resource estimation metric**

*Relevant Questions: 19,27,28*

Time is the most important factor for decisions, which is looked into by this metric. Time, in the form of salaries to employees, is by far the largest cost for software development. Computers, office material etc. just cost a small sum compared to salaries to employees.

Resource estimation is an important part in the decision process and managers have to have great knowledge and experience to make a correct estimate of time. The ability to estimate resources will always be of relevance for resource planning and the strategic decision process in a company. Any business decision weighs costs against revenues, in the long run or the short run. The resource planning decisions has to be well substantiated so they support the overall strategy plan.

The balance is to be found when the actual time spent coheres with estimated time. If this indicator shows a negative trend then it might be of interest to take a look at the individual deadline metric, the individual burden metric, the time report system metric and the project dependence metric in order to figure out if there are any correlation between them. For example, if the individual burden metric is too high then the time estimates will suffer as a consequence.

## **Organisational structure**

### **Company reactiveness metric**

*Relevant Questions: 22,25,26,47*

This metric indirectly measures the decision process function of management. A small strategic group should be able to take fast and solid reactive and proactive decision in the constantly changing competitive environment. The balance between the two is very important. If you give in too much to the customer there is a big risk that you will spend too much resources on solving the problems on the market today when you can be proactive and solve the problems of tomorrow.

### **Department representation metric**

*Relevant Questions: 11,20*

This metric is one of the most important for the decision process. It shows whether the employees support the decisions or not. If they are dissatisfied with the decisions then they do not think that they are well represented either. The employees will support a decision if it is well founded and motivating.

It is also a question of making the right decision. If everybody in the company feels well represented they can at an early stage in the decision process signal threats and possibilities that can effect the decision. On the other hand if they do not feel represented, they do not feel any involvement and keep the information to themselves.

### **Project priority metric**

*Relevant Questions: 21,38*

This indicates the existence of a prioritising system of projects and requirements. The existent of such a system is the first step towards new ways of thinking. If the company is going to start prioritising they have to motivate why one project or requirement is more important than another. To do this you need better decision material. It is believed that by starting to prioritise, the company builds up a better decision process.

### **Customer specific dependant metric**

*Relevant Questions: none*

This captures the spectrum of customers' needs that the company satisfies. In other words, how much of the company's resources are dedicated to satisfy it's total customer portfolio compared to how much resources they spend on their main customers' needs.

The relevance of this metric will always be of interest since it gives an idea of what the actual company strategy is. The reason behind this is that this metric reveals how the company's resources are truly used and can be compared with the estimated values from overall project planning.

When analysing this metric it is of interest to take into account the departments alignment to overall strategy metric in order to get a better picture of the company strategy.

## **Information flow**

### **Strategic information metric**

*Relevant Questions: 5,12,23,24*

This metric detects the ability of an organization to share information related to overall strategy within the organisation. It is important that information of the strategy is spread throughout the organisation and that the employees have possibility to reflect and give

feedback on the strategy. There has to be a good balance between the employees' interest in strategy and the willingness of the management to give out information about the strategy.

This metric is of most concern at the strategic level of the company, but it is also important at the operational level. Without the possibility of the management to share the strategy and visions throughout the company, the strategic decisions are futile. The management must create an overall interest in strategy so that all members of the company at the operational level can have a clear view of the business. By introducing the employees in the strategy development they become more confident in their daily activities. The members of the organisation have to know their role in the strategy development.

Its focus is on the fundamental reasons why the organisation exists. The metric affects the goal of development and the reactive or proactive development.

#### **Priority of information metric**

*Relevant Questions: none*

This captures whether the company prioritises information or not. Prioritising information within the company is very important. This is due to the simplicity by which information gets spread at present, through for example e-mail. The use of e-mail and other technological advances has imposed a huge amount of information on employees that must be processed and thus there is a great need for prioritising this information.

The relevance of this metric is at present of high interest due to the technological advances that aims to facilitate the communication between individuals. However, people will learn how to use the simplicity by which information gets spread the moment when new technologies get developed to aid the selection of information.

#### **Formal meetings regarding company strategy metric**

*Relevant Questions: 12,13*

The number of formal meetings within the company regarding the strategy of the company per a quarter of a year assure that the employees receive strategic information

and are given an opportunity to actively participate in the creation of the company strategy. At formal meetings all the different departments within the organisation should be present in order to make sure that all aspects regarding company strategy are reflected. The number of attendants of the meeting should also be as high as possible.

The relevance of this metric increases with the size of the company. In larger organisations formal meetings are necessary to achieve the strategy alignment. In smaller firms' informal discussions regarding company strategy might be adequate to confirm that all employees participates and thus all aspects are covered.

When analysing this metric, it is of interest to investigate how well the employees' personal views of the company strategy are aligned to the decisions made at the strategic level. The more actively involved employees are in the creation of the company strategy, the easier it is to get them to strive towards a common strategic goal and thus implement the strategy of the company.

### **Departments' alignment to overall strategy metric**

*Relevant Questions: 20,24*

This metric describes more thoroughly the interaction between the overall strategy in the company and different departments/individuals activities. It is important to know how well the organisation is following the strategy that the management has outlined. In other words, it examines how good the company is at implementing the decided strategy of the company.

### **Specification process**

#### **Development goal metric**

*Relevant Questions: 31*

This metric focuses on the ability to forecast the outcome of the development process. If it is satisfactory then the developers are able to better fulfil the commitments of the company. Another aspect is the possibility to affect the content of the requirements and thus produce a clear goal of the development effort which both the organisation and the individual developers accept.

It is of strategic concern to the company that there are clear goals of what to be developed in the development process. This is because of ever-changing market and emerging competition that the importance of developing the demanded products is increasing. In order to manage this, it is crucial that the entire organisation is able to comment and influence the requirements and thus consider all aspects of the development.

The metric is closely connected to resource planning and prioritization and thus the outcome of the ability to estimate resource requirements indicator, project priority metric and requirement motivation metric affects this metric. As this metric improves, the development result will be better aligned with the demands of the customers.

### **Priority of requirement metric**

#### *Relevant Questions: 38*

This metric points to the amount of prioritisation between the requirements undertaken by the development department. Without clear prioritisation between the tasks at hand, relevant functionality might not be implemented within the time for the next release. When the prioritisation is well defined, the developers are able to distinguish between requirements that have to / ought to and might be implemented.

It is utterly important at the strategic level that the aspect of prioritisation is well established in the organisation. This could sometimes be achieved through informal communication in small companies, but as the company is growing, prioritisation has to be made formally in the requirement definition. The insufficiency in the information flow from the customers to the development department is one vital obstacle for dividing the requirements into different priority levels.

The prioritisation is important to the overall performance of the company. The most affected metrics is the ability to estimate resource requirements, project priority and department alignment to overall strategy. The requirement motivation metric is on the other hand affecting the result of the metric.

**Requirement change window metric**

*Relevant Questions: 22,27,28*

It is important that no new requirements are introduced close to or after the initial deadline of the projects in order to have a secure release management. The metric is a reflection of this matter and if it receives a bad result there is an indication of problems in the specification process. How the requirements change in the development process is also a reflection of how reactive the company is.

Several different causes affect the outcome of it. Most of them is concerning metrics at the strategic level of the company. As stated before, if the company is too reactive this could be problematic to both the specification and the development processes. Time estimations and management of release dates are other aspects that are of concern when investigating this metric. If the result of this metric is unsatisfactory the long-term strategy might get damaged.

This metric and the long-term strategy are of concern to many different metrics and is mostly affecting other strategic metrics such as the reactive and proactive development and customer response. The release content metric and the project co-ordination metric are other influenced metrics.

**Requirement cost metric**

*Relevant Questions: 22*

The performance of the product is connected to the total cost of developing it. This metric reveals the understanding that the employees have on this matter in the development department. If not, there is a risk that new requirements are implemented without consideration on the costs of including the requirement into the project or release.

This metric is important at the strategic level of the organisation as it is of concern to the projects to be implemented and to the content of requirements included into them. The awareness in the development department as well as in the marketing department of the costs to introduce new requirements is therefore essential. If the costs to implement the

requirements are revealed then it becomes more understandable to implement according to priority levels.

The cost awareness and particularly the metrics at strategic level such as ability to estimate resource requirements, the departments' alignment to overall strategy and the project priority are influencing and are influenced by this level of the development process.

#### **Documentation standard metric**

*Relevant Questions: 42*

This metric focuses on the ability to document the activities during the development process. The documentation is of concern in both the specification process and the overall information flow in the organisation. Without a proper documentation in the specification process, the organisation is not able to decide when the requirements are fulfilled. However, if too much structure is used, there is a risk in smaller organisations to lose the vitality and creativity.

It is of concern to the strategic level of the company that documentation is satisfactory. If the company is not able to document the decisions made and the product requirements in an orderly manner, the specification process might get volatile. A number of different metrics is connected to this metric, such as the strategic information, the release content and the development goal metric.

#### **5.3.2 Definition of operational metrics**

In this section the indicators at the operational level are defined.

##### **Resource Planning**

###### **Release date accuracy metric**

*Relevant Questions: none*

The purpose of this metric is to give an idea of how good the company is at keeping release dates. How good the company is at it will affect the company's ability to gain new customers and to keep old ones.

There are naturally other metrics that will affect the outcome of this metric. For example, the aggregated project status and the ability to estimate resource requirement metric. As with project there might be a trade off between the time factor and the content of the product. Therefore it might be of interest to take a closer look at the content metric.

### **Release workload metric**

*Relevant Questions: none*

This metric captures how the upcoming of a release date affects the organisations overall work load. By examine the time report system one can map accumulated overtime against time and thus illustrate how the total overtime is allocated over time.

If the amount of accumulated overtime is too large in conjunction with a release this can be a signal that the company's resource planning process is not functioning well. If the resource planning process was well functioning then there would be no need for making the employees work overtime. Furthermore, if the accumulated overtime per employee is too large this can affect the time that the employees stay within the company negatively. Which means that the company is at risk of loosing valuable tacit knowledge.

A number of metrics are of interest when analysing this metric. For example, the resource estimation, the release dates accuracy and the release content in order to create a better understanding of the outcomes of this metrics.

### **Individual burden metric**

*Relevant Questions: none*

The underlying assumption, which this metric relies on, is that the work efficiency per employee increases in the range from zero to three projects carried out in parallel. This is because; if one project gets stuck another one can be continued with while the solution to the former project takes shape. However, if the number of project rises above 3-4 projects then the efficiency is believed to drop. The reason is that the switching costs gets too high.

In the introduction of technologically-enhanced process elements, once again the human element is the recipient of change, and improvements introduced in the technological

component of development process have a great impact on their human component, such that the incorporation of technology such as CASE Tools, requires a shift in the organization culture of persons participating in the process.

As stated in the ability to estimate resource requirement metric this metric should be of special interest to take a look at in order to see if there is a correlation between constant project overruns and this metric.

### **Clarity of project interdependence metric**

*Relevant Questions: 37,39,48*

This metric is assembled through the information on the project interdependencies. It points out the relevance to ensure the scope of the project in the specifications. Without clear information on this matter, the developers might negatively influence the product, as different functionality is demanding change in the same code. A specification on the modules and files that are affected is therefore important.

This metric is positioned at the operational level in the development process. Although it is at the strategic level that the projects are decided upon, the interdependence is normally managed at the operational level. This interdependence affects several aspects of the accomplishment of the projects as some projects might be sequentially implemented and others might be implemented in parallel.

This metric is dependent on the strategic decisions and thus has the ability to estimate resource requirements metric connected to this metric. Other metrics influenced are building software metric, individual deadline metric and release content metric at the operational level.

### **Organisational structure**

#### **Support of decisions metric**

*Relevant Questions: 23,24*

This metric indicates how well the operational decisions within the software department are supported by the employees. A prerequisite to get the support of the employees is to back up the decisions with well-founded arguments and to have a clear purpose. If this

turns out well then the employees' motivation to carry out the assignments is believed to be high. Well-motivated employees increase the productivity. On the other hand if the employees feel insecure in the development of the working environment they also feel insecure in the leadership.

The organisational culture also plays an important part in the support of a decision. It is harder to get a hundred percent support from the organisation if the members are used to a total democracy when making decision about the company's future.

This metric is ultimately connected to the Department representation metric and the Formal meetings regarding company strategy metric since the operational decisions originate from the company strategy.

## **Information flow**

### **Project co-ordination metric**

*Relevant Questions: 14, 29*

This metric emphasises the importance of co-ordination between different activities and projects. It is thus closely linked to project time management and project cost management. Specifically, the indicator captures the number of project co-ordination meetings per quarter of a year, since this gives a good estimate of how seriously the software department takes this co-ordination of different activities and projects. Middle management and top management should be able to co-ordinate different projects and activities with the help of a functioning report system.

### **Supportive tools alignment metric**

*Relevant Questions: 31, 42*

The implication for this metric is that the specification process needs to have supportive tools in the development process, which reflect the actual work performed. If the metric is satisfactory then the supportive tools are able to handle division of requirements, aspects of different versions and the total requirement documentation effort.

The requirements have to be supported during the initial development as well as in the documentation and feedback phase of the specification process. The outcome of this

concerns the operational level of the company. The metric is based on the project repetitiveness and the ability to learn in the organisation. The ability to manage different versions of a product is important if different customers have considerable different versions of the product.

As the metric is based on the ability to learn, it involves the feedback process in the development department. The indicators in the specification process and in the other perspectives are therefore influencing the result. Due to the learning ability, an metric that affects the result is the aggregated project status indicator. The project co-ordination indicator also affects the outcome and this is based on how to co-ordinate the different projects into different versions.

### **Specification process**

#### **Requirements motivation metric**

*Relevant Questions: 37,40,47*

This metric is intended to show if the company knows the origin of the requirements. Different customers are demanding different aspects of the implemented requirements. The outcome of this metric shows how well understood the demands of the requirement are.

Requirement motivation metric is important at operational level as this gives an indication of the importance to implement the requirement. This metric is especially important if the company is mostly relying on externally funded projects.

The indication is also important if the company is mainly reactive. If the requirements are of interest of several customers, then the customer-segments should instead be indicated. The customer motivation gives the developer an idea of how the requirements should be used in a broader context.

The metric depends on several different metrics, such as the customer specific metric, customer reactive metric, and requirement prioritisation metric and project co-ordination metric. The dependency concerns aspects such as co-ordination and prioritisation of demands from different customers and different market-segments. This shows how

broadly the customer segments are defined in the requirements and is usable in order to develop from a reactive to a proactive organisation.

### **Requirements testability metric**

*Relevant Questions: none*

This metric describes the ability to test the developed requirements for completion. It is intended to show the ability to forecast the actual result of the development. If the implemented requirements are tested and confirmed to a great extent, then the product performance is better aligned with the initial requirements. The maintainability of the product is another aspect of this metric. If this metric is satisfactory then the maintenance and regression testing is improved.

This metric is of greatest concern at the operational level. The ability to test the requirements is also crucial to the strategic level, as it is at the operational level that strategic visions are implemented. This metric is intended to verify if the company is able to test and confirm the requirements. It is also intended to verify how well the company is using the possibility to test the requirements at present and in future regression tests.

This metric is focusing on the last phase of requirement handling, when the requirement is actually implemented. The metric is therefore dependent on the development goal metric. Other aspects are the resource planning and thus the status of the testing in the company. As this indicator improves, the requirements definitions are improved and then the result of the implementation.

### **Software building metric**

*Relevant Questions: 46*

This metric reflects the work performed at the preceding steps within the development process. If this gives a positive result then the process developing the product is more likely to be thoroughly completed than otherwise. It is a receipt on how the requirements are handled among interconnected projects and how different interfaces are specified. There is a great risk to the development process if this metric is negative.

The operational level is most affected by this metric. The metric is a reflection of how the operational level has accomplished its undertakings in creating a product from the initial requirements. Furthermore, this indicator is important at the strategic level while it is of concern when estimating the release date of the product and many different projects are relying on this result.

As stated before, the metric is important both at the operational and strategic level. The different perspectives that are interconnected to this indicator are the estimate resource requirements metric, project co-ordination metric and organisational learning metric.

### **Degree of automated test metric**

*Relevant Questions: 36,44*

In order to ensure the performance of the product, it is important that the product is tested thoroughly and in a controlled manner. This metric reveals the amount of testing that has been automated and if the tests are easily completed. A satisfactory result would be a testing of the requirements in more than fifty percent of the total amount of requirements that can be tested.

The specification process is considered in this metric and the result is relevant to the whole operational level in present and in future releases of the product. It is relevant to the work performed in all of the different perspectives at the operational level as well as at the strategic level. As the product matures, more and more of the tests executed to ensure the performance of the product is more or less automated.

Several metrics influence this one and the most obvious of them are of concern to the specification process, such as the requirements testability metric and feedback on quality at department level metric. Other metrics are the release content metric and aggregated project status metric.

### 5.3.3 Definition of interconnecting metrics

#### Resource planning

##### Resource evaluation metric

*Relevant Questions: none*

This metric captures how much time that the employees within the software development department spend on project reflection and feedback. The time spent on this activity should not be too small in comparison to the total project time in order to facilitate organisational learning and thus improve the overall development process.

As stated above project reflection and feedback is important in order to promote organisational learning and thus this indicator will always be of significance.

When analysing this metric it is of relevance to regard the outcome of the feedback on quality at individual level metric and the total amount of time that each employee spend in training per year since those also concern organisational learning.

##### Time report system metric

*Relevant Questions: 29*

The metric tries to capture how accurate the time report system is and thus how useful this system is as basis for making new decisions and evaluating past decisions/projects. If the information contained in this system is of high accuracy then managers can use this information in combination with their own and their employees accumulated experience in order to make better estimates in the future.

The relevance of this metric is very high in larger organisations since it gives a hint of how much control the company has of its development process. To be more precise, what knowledge the company possesses on how the developers spend their time and thus the costs to carry out different tasks.

The metric can be linked to the status report metric in order to tell how valuable the time information given at these occasions might be.

## Organisational structure

### Documentation alignment to actual work procedure metric

*Relevant Questions: 11,29*

To be able to introduce new employees into the organisation and to align the individual work procedure to the common standard in the organisation, the use of organisational control can be interpreted in different ways. One way is to have thorough documentation of the different work procedures gathered by a standardisation department. Another way is to leave it up to the individuals to align their common work procedure to each other in an organic manner.

This metric is intended to show how the usage of the formal reporting system in the company is constructed and how the resources spent coheres with what is actually reported. As the company grows there is a need for the organisational control mechanism used within the company to be more standardised and specified. In a smaller company the specified work procedures will be more or less minor guidelines.

This metric affects several other metrics as it resides within the feedback loop. The work procedures will be enhanced in each iteration of the development process from strategic decisions to the release of the product. The most immediate metrics affected is the feedback on quality at both individual and departmental levels. Another metric is the individual dependence indicator.

## Information flow

### Organisational learning metric

*Relevant Questions: none*

This metric reflects the ability of the organisation to create organisational knowledge and improvements by transformation of knowledge into explicit knowledge. Organisational knowledge creation is a process started at the individual level and expanded through the organisation by interactions that crosses departmental, divisional, and organisational boundaries.

The products created by the organisational knowledge process shall be reviewed for its coherence with both operational and strategic goals. Even if the newly created product

has superior quality, it may conflict with the strategic or operational goals that are stated at the different levels of the organisation. The indicator is thus relevant to the feedback of the development process.

This metric is closely linked to a variety of indicators as it concerns how the organisation learns and creates knowledge. Metrics that are affected are the development goal metric, the ability to estimate resource requirements metric, and department alignment to overall strategy metric. Furthermore, the reactive or proactive development indicator is affecting this metric.

## **Specification process**

### **Feedback on quality at department level metric**

*Relevant Questions: 14,34*

To improve the performance of the product and align it to the expectations of the customers, there is a need for feedback on the perceived quality at departmental level. The product performance is connected to the fulfillment of requirements and it is the feedback on the requirements that is the important aspect of this metric.

The metric concerns the feedback into the development process and is thus relevant to the feedback perspective of the development process. This is the link between the operational and strategic level concerning the requirements. If the feedback at departmental level is inadequate the strategic feedback might be neglected and constant improvements of the development process are restrained.

As the metric connects the strategic and operational levels, several other metrics influence this metric. The influencing metrics reside in all perspectives, such as aggregated project plan metric, departments' alignment to strategy metric and project priority metric.

### **Feedback on quality at individual level metric**

*Relevant Questions: 33,34*

This metric reflects the ability to receive feedback on individual performance in the development process. Without feedback at the individual level there is no possibility to

evaluate the performance and make corrections and adaptations. Another aspect is the search for responsibility in the organisation, if the individuals are striving to gain knowledge and learn from the process.

The metric is mostly concerned with the connection between past, present and future performance at the individual level. It is thus the link between the operational and strategic levels in the organisation. There must be a support at strategic level for progress by the individuals and hence the performance at operational level.

As previously mentioned this metric is the link between strategic and operational levels and therefore are several indicators affected and influenced by this metric. The most immediate metric is the feedback on quality at departmental level metric. Other metrics are individual deadline indicator and individual dependence metric.

#### **Overall system knowledge metric**

*Relevant Questions: 6,7,10,15,16,17,18,19,20,35,43,48*

This metric evaluates if the employees know how development of new functionality affects other parts of the software. With a thorough knowledge of the overall system, it is more probable that implementation of new requirements are made without sub-optimisation of the total system. An overall system knowledge indicates that the employees know how their own effort affects the entire system.

The feedback system within the software development department affects this indicator. Through organisational learning, the individuals' knowledge of the overall system will increase. There must be a support at the strategic level for individual progress in this area hence an improvement at both strategic and operational level.

A few indicators influence this indicator and the most obvious concerns the specification process, such as the requirement motivation metric and development goal metric. Other metrics are the individual dependence metric, strategic information metric and the resource evaluation metric.

## CHAPTER 6

### ANALYSIS

The purpose of developing metrics is to actually use them. This section is a presentation of an attempt to use the metrics, as intended, as an evaluation tool of the development process in a software developing company.

The analysis has been divided into four parts; strategic level, operational level, an interconnection between strategic and operational levels and concluding remarks. Each level is then divided into the different perspectives presented in the reference model.

The metrics are in most cases analysed through several questions stated in the questionnaire, found in appendix A, and in some cases the empirical study is also included in the analysis. An inclusion of the results from the questionnaire at HSS is as well presented in the appendix A.

#### 6.1 EMPIRICAL STUDY

As stated earlier, 30 employees of HSS were distributed the questionnaire. Out of these, 28 employees answered the questionnaire on basis of the answers received and the interviews conducted, information gathered regarding software development process is presented in following sections.

##### 6.1.1 Development process

Hughes Software Systems Limited (HSS), specialists in convergent network software is the number one communications software company in India. Leading Communication

OEMs and Application Developers around the world source a broad range of communications-related software services, products, and solutions from HSS.

Its assessed at SEI-CMM level 4 for all its development centers. HSS's focus is in the Convergence market. It enables OEMs deliver the Next Generation of Communication Networks through Voice over Packet solutions, which include Media Gateway, Softswitch, Media Gateway Controller and Gatekeeper frameworks along with a wide variety of protocol stacks.

A leading provider of professional services to leading communication OEMs, HSS provides full life-cycle development for communication software, consisting of problem definition, system design, implementation, testing, delivery and post sales support to enable development of communication technologies.

The development process being followed is the modified waterfall model presented in the figure 6.1. Umbrella activities are taken out throughout the project life cycle, which assure the correspondence between the management level and operational level. These activities include project tracking and monitoring, Quality Assurance, Configuration management and Training.

Project Tracking and monitoring is done at regular intervals to ensure that the project is moving on the right track with the right pace. Time sheets and status reports are used for evaluating the process.

The QA department assures quality by performing inspections, walkthroughs and testing the product. There are 100 projects being undertaken by HSS and 10 inspections are done for various artifacts of all these.

Employees are recruited as trainees. HR department undertakes this activity. Trainees are allocated projects according to requirements of the projects. Training is given before any responsible task is assigned to the employee.

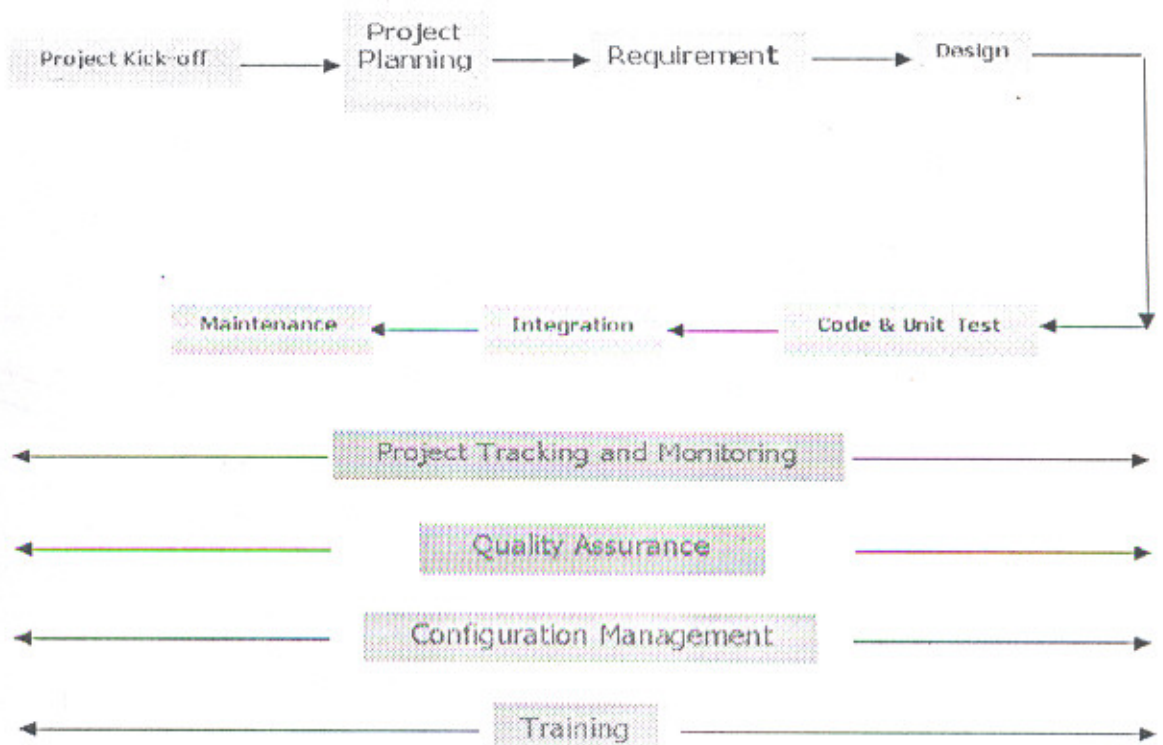


Figure 6.1: Modified Waterfall model

Configuration management is undertaken to keep a record of various changes made while working on the project and to control the versions of product.

Finance department takes care of all the finances of company. Vendors take up some part of product development. While subcontracting, vendor evaluation is carried out as per process defined in the QMS.

Products are developed with respect to some generic requirements. Once the SRS is finalized, the *business group* is considered as the customer and full SDLC is followed as per the development process model, apart from different phases, the product has to pass through an independent product qualification test, which decides whether product can be supplied or not. Maintenance of products is taken care by product support team.

## 6.2 INTERPRETATION OF EXTRACTED METRICS:

### 6.2.1 Strategic level metrics

#### **Resource planning**

##### **Overall project status metric**

Report on status of project is given to the immediate superior once in a week. Meetings are held at regular intervals to discuss the status of work and any modifications to be made.

Resources are allocated to the projects based on the criticality of project. The schedule and cost are estimated using COCOMO-II, FPA and past productivity figures.

All the responses show that management has complete knowledge of what is going on in the organization and gets the updations regularly.

##### **Reactive or proactive development metric**

The software development department is mainly proactive in making decisions. This is evident from the resource allocation chart, which shows that only 5% resources are allocated for customer support reports. This indicates that beta testing is quite effective and 10% resources are allocated for it.

Errors found after beta testing is less and hence cost involved in this activity is also less. This helps lowering the cost of the project on the whole.

##### **Resource estimation metric**

Three major resources are time, cost and human resource. These need to be managed and requirement of each needs to be estimated before the project begins. Time estimation has a very important part to be played in decision making.

The trend in an organization shows that time and cost estimates vary from actual time and cost by 8-10%.

Less than 10% projects finish before given deadline, 60% take longer than estimated. This matter needs to be looked into deeply as the customer satisfaction depends largely on adherence to the deadlines.

## **Organisational Structure**

### **Company Reactiveness metric**

From the questionnaire it's clear that mostly company's decisions are proactive. In some situations, however, management needs to react to customers' response or change in requirements due to competitive environment. For this new functionality requirements are added after the original freezing date. In most cases the company's decisions are proactive. The business group surveys the market and develops the software requirement specification as per the demands of the market. The SRS is finalized after a discussion with the intended customers, business group representatives and project developers.

Each and every department is involved in making decisions related to the project. Majority response is in favour of way the company responds to its customers.

### **Department Representation metric**

100% people in the survey say that their department is very well represented in decision-making. This involves employees in the decision making process. They can express their views confidently. They feel motivated and this increases the productivity of individual.

### **Project priority metric**

Setting priorities indicates the good strategy making capacity of management. It suggests that management has clear idea of what each project is going to get them and what can be expected of it.

In HSS, the work is going on for about 100 projects. These projects are prioritized depending on the contract and nature of the project. Within the project, three different priority classes for requirements are set and implemented in the order of priorities.

### **Customer Specific dependant metric**

As specified earlier, the business group looks into the needs of the customers. From the information obtained, the management allocates the resources to different projects.

This helps in overall planning of projects in the organization.

### **Information Flow**

#### **Strategic Information metric**

Strategic information is available online for ready reference by the employees. Apart from this, information is sent formally to all every quarter. There is a balance between the employees interest in the strategy and willingness of the management to share the information about the strategy.

Mostly the ongoing projects within the organisation are aligned with the decisions made on the company's development.

#### **Formal meetings regarding company strategy**

Formal meetings are held quarterly and help in achieving strategy alignment with the ongoing projects.

#### **Departments' alignment to overall strategy**

90% of people surveyed say that there is good alignment between ongoing projects and the decisions made on the company's development. The departments are very well represented in decision making process. This motivates the employees as they feel that what ever they suggest is take into consideration and does not go futile.

## **Specification Process**

### **Development Goal metric**

Project management tools are used mostly (90% of times) to decide the schedules, estimate the cost and decide the deadlines of various phases and final delivery. Tools used are COCOMO-II, FPA and past productivity figures.

This helps give approximate (sometimes-accurate) figures to the customer and help company make commitments.

### **Priority of Requirement metric**

Prioritizing requirements helps implement relevant functionality within the time for next release. Here there is clear prioritization between tasks. Requirements are divided into three priorities and implemented in that order.

### **Requirement change window metric and requirement cost metric**

Sometimes new functionality requirements are added after the original freezing date. This view is supported by 40% of employees surveyed. This gives the reactivity of the company in fulfilling the demands of the customers. As 60% of projects take longer than scheduled deadline, the reason could be addition of new functionality requirement at later stage of the project.

### **Documentation standard metric**

Being a SEI-CMM level-4 organization, all the activities are documented during the development process. There is a clear specification process and also the information flow is organized.

## 6.2.2 Operational metrics

### Resource planning

#### Release date accuracy metric

80% of the surveyed employees state that the schedule deadlines are skipped by 8-10%. This variance could be because of addition of new requirements at the later stage or the change of demand in the market.

#### Release workload metric and individual burden metric

As the release date approaches, the workload on individuals increases. This is clear from the time report system. Amount of accumulated overtime is large as the release date nears.

#### Clarity of project interdependence metric

The information that one project is related to another is given to the developers of related projects. All related specifications of modules and files are also made available to all. This avoids any negative influence on the product developed for the customer.

### Organisational Structure

#### Support of decision metric

The employees support the operational decisions within the software department. 85% of employees surveyed say that the decisions are properly aligned with the strategy. The information on strategy is made available online and all have a right to give suggestions.

Developers have a right in finalising the requirement specification. This makes them clear as to what they are to develop and motivates them to work for better.

## **Information flow**

### **Project co-ordination metric**

Weekly meetings are held to discuss the status of on going work. Timesheet submitted weekly give a fairly good idea of the status of work. It indicates the development stage where the project presently is.

### **Supportive tools alignment metric**

Supportive tools are used for preparing requirement specifications, documenting the work done and taking the feedback from various sources.

## **Specification process**

### **Requirement motivation metric**

The answers received to the questions indicate that the developer has the right to affect the composition of the requirements. Customers affected by the specification are clearly mentioned alongwith.

This indicates that the company knows the origin of the requirements and that each developer knows the importance of implementing the specified requirement.

### **Software Building metric**

The analysis and design phase is given lot of importance. Considerable time is spent on specifying the requirements. This loads to thoroughly completed product, which has high value in the market.

## **Interconnecting metrics**

### **Resource planning**

### **Resource evaluation**

### **Time report system metric**

Time report system reflects the use of time within the company. Here it reflects the development process quite well. Managers to make better estimates in the future use the information. This indicates a strong control of company on its development process.

## **Organisational Structure**

### **Documentation alignment to actual work procedure metric**

Reports used are time sheet and status sheet. These indicate the amount of work completed and effort put in by an individual. Documentation of each phase is done efficiently. Detailed artifact is produced after every phase.

## **Specification Process**

### **Feedback on quality at department level**

Weekly status of work is reported to the immediate supervisor. The failure reports are investigated and traced at project level. This provides necessary feedback on the perceived quality at department level. This metric indicates the interconnection between the strategic and operational level.

### **Feedback on quality at individual level**

Individuals report to the supervisors during and after project closure. The failure reports are investigated at project level. This metric indicates the performance of individual in

the development process. This helps the evaluation of performance and make corrections and adaptations. It helps learn from past experiences and also affects future projects.

### **Overall system knowledge metric**

The employees have a considerable knowledge of development process. They are aware of what is expected of them and what requirements are to be implemented. The feedback provided helps optimize the process, and not meet the requirements of customer and more confidently.

### **6.3 Concluding remarks on the analysis**

Before concluding, it is worth mentioning that this case study is not to give grade or judgement to the process undertaken by HSS. It is to understand how a software development process is taken up by software industries, what all factors affect it and various metrics which measure these factors.

At the strategic level, two similar metrics; reactive or proactive development metric and the company reactive, indicate the strategy of the company on different levels. Former metric indicates that the company is proactive in development of the project while latter indicates the company's strategy towards customer demands. At company level HSS is reactive towards customer while in case of software development, it is proactive in predicting the future needs.

An organizational focal point is established for quality planning and tracking. Resources are provided to support technology introduction. Periodic management reviews are held of performance against quality plans and quality improvement actions.

Organization has a strong hold on its information flow. Means are established to ensure that software process and environment support issues are identified and promptly resolved. Strategic information metric and formal meetings regarding company strategy metric indicate this.

Project co-ordination metric, resource planning metric and clarity of project interdependence metric measure how well the resources planned at the operational level. Supportive tools alignment metric indicates the feedback received from employees and ability to learn and improve from this feedback. Direct concrete feedback is a way to improve individual work.

Use of automated tests ensure that the product performance is upto the required level of customer satisfaction. Resource planning is one of the main aspects to be considered in an organization. Resource estimation metric and resource evaluation metric indicate how the resources are allocated or distributed to different projects.

At an overall level, the metrics describe the software development process at HSS. Being a SEI-CMM level 4 company, it has good customer industry relationship. The organization has initiated comprehensive process measurement and analysis. The resulting precise knowledge of process provides the foundation for significant and continuing product quality and productivity improvement. All projects have established plans, resources are provided to support technology introduction.

Performance is tracked against quality plans and quality improvement actions. Documented standards are produced for inspections, tools and methods, quality plans and quality tracking. A process database is established for reference of ongoing and future projects. Organizational culture preserves the positive things about the organization al structure and the information flow.

However, the concept of reusability can be introduced into the development process. With all the requirements of SEI-CMM level 4 being fulfilled, organization can strive for SEI-CMM level 5 and work for optimizing its software development process.

## CHAPTER - 7

### CONCLUSIONS

In the beginning of the thesis, the purpose was set and the efforts were directed towards this goal. This chapter aims at discussing the findings made during the research and the theoretical contribution. The focus is on a discussion, if the developed metrics are able to reflect the actual capabilities of the development process. In other words, if metrics capable to indicate the organization's ability to develop products that outperform the competitors.

#### 7.1 NEED FOR THE METRICS

By developing and using the metrics supporting the reference models that are suggested in the master thesis, the organization is made to learn about product development and its environment. At first glance, the reference models and the corresponding metrics could be yet another capability measurement technique. However, it is actually a new form of evaluation of the management system that links strategic and operational levels of the organization to the needs of the customers.

#### 7.2 DEVELOPMENT PROCESSES DIFFER

As mentioned in the previous chapters that no one-development process is applicable to every situation, then aspects of resource planning, organisational structure, information flow and specification process play a vital role in the product development. To manage the differences, the metrics are based on basic questions and assumptions and are thus an applicable tool throughout the organization. It requires both the strategic and operational

levels of the organization to reflect its position from four different perspectives and to answer basic questions regarding development process.

### **7.3 THE ANALYSIS OF THE INDICATORS IS MANAGEABLE**

Having understood the basic assumptions behind the four perspectives, the company is able to create a method for collecting the required data. In the analysis of HSS an elementary questionnaire was developed and by slight adaptations it could be used in other organisations. If an appropriate data requirement method is at hand, the management is able to set targets on the four perspectives and then measure the achievements and improvements.

These targets are then communicated to management and staff within the company in order to create an understanding throughout the organisation how individual efforts contribute to company development and vice versa. This helps in establishing strong relationship between management and developers.

### **7.4 THEORETICAL ACHIEVEMENTS**

In a survey on the matter of evaluating the software development process several research areas are approached. It was found unreasonable to confront each and every one of them thoroughly. However, the four perspectives stated here, are sufficiently described to support the current definitions of the metrics.

By the development of the reference models, new applications of available theories are concluded. It is foremost in the bond between strategic issues in the software development department and operational aspects of the software development process that the main contributions are laid. It is made clear that there is generally no direct connection between cause and effect in this system.

## 7.5 UNBALANCED THEORETICAL SUPPORT

In the development of the theoretical background, question that came to mind was whether different theoretical aspects should be contained in one perspective or in another. After brainstorming, the conclusion that can be drawn is that the four perspectives are closely interconnected. As a consequence one can, in some cases, argue that too much attention is laid in one area while other areas seem to be incomplete. The aspects defined for one perspective are somewhat related to other perspectives also.

## 7.6 PREFERRED REFINEMENT OF THE METRICS

The current status of the metrics is that they need refinement and feedback by application on a wider population. These metrics cannot be regarded as static. Instead by using them as tools for both benchmarking and internal evaluation, they will appear to be slightly dynamic. Although, the definitions might change to some extent and the substance might, in some cases get questionable, it can be stated that the specific metrics are meaningful to the overall understanding of software development process.

## 7.7 FUTURE SCOPE

In future work there is a need to refine the metrics through a systematic research in a broad range of development processes in different software organisations. By this there will be an ability to refine the scope of the metrics, so that they can be expressed in a more complete and systematic manner.

In the end, there is a possibility to propose evaluation methods or techniques that are more general. A method of evaluation would include appropriate metrics, forms of data, forms of analysis and techniques of interpretation to produce the evaluation of the development process.

## 7.8 FINAL REMARKS

These metrics cannot be used too simply in the development process. Each software development company is unique and each has different obligations towards its customers and employees. The developed metrics are solely intended as relative measure, and not as absolute measures. Correctly used they can provide a guideline to both the management of the development process and its employees.

# QUESTIONNAIRE AND RESULTS

## A.1 QUESTIONNAIRE

### *Important note!*

**The questionnaire is anonymous and is meant to be used for research only. Under no circumstances will it be traced to individuals.**

1. How do you relate your organization to...?  
Strategy  
Market  
Customers  
Competitors
2. What is the development process followed by your organization? Give a flowchart of the development process?
3. Does your organization investigate the development process of other companies? If Yes, to what extent?
4. How would you define product development?
5. In what way does the company take care of risks in product development?
6. Does financial requirements affect the product development? Has any product development curbed due to financial crisis? If Yes, how often?
7. Which departments are collaborating in and affecting the product development?
8. Is there a development plan to each part of the product?
9. To what content does the company locate resources on maintenance?
10. How are the employees evaluated?(From Management's point of view)

11. What report systems are used in the company (for example the time report system, the status report system etc)?  
How often is each report used and by whom.
  - (a) Once a month
  - (b) 2-3 times per month
  - (c) 4-5 times per month
  - (d) More than 5 times per month
12. How often does the managerial body send out strategic information to the staff?
  - (a) Monthly
  - (b) Quarterly
  - (c) Twice a year
13. At department level, how many project co-ordinating meetings do you have?
  - (a) Weekly
  - (b) Twice a month
  - (c) Monthly
  - (d) Depends on project
14. How often and at what development stage is the status of works reported to nearest superior?
15. How is quality measured? (Number of defects, defect density etc.)
16. Define defect.
17. How is cost estimated and schedule set-up?
18. What is the frequency of structured walkthroughs, reviews and inspections?
  - (a) Once in a project
  - (b) Once in each phase
  - (c) Atleast once for each artifact
  - (d) Not necessary
19. How good are the project time estimates, budgeted compared with final outcome?  
Give the variance of final time taken to the estimates.
20. Are product departments well represented when decisions concerning different projects within the company are made?
  - (a) Very well represented
  - (b) Well represented
  - (c) Less well represented
  - (d) Badly represented

21. Is there any priorities set between different projects within the company( Customer Support Reports excluded)?
  - (a) Yes
  - (b) NoIf yes, on what basis is prioritization done?
22. How often are new functionality requirements added after the original freezing date?
  - (a) Never
  - (b) Sometimes
  - (c) Often
  - (d) AlwaysIf answer is not (a), at what phase functionality requirement is added.  
How does it effect cost and schedule.
23. How often do you receive information from your nearest superior concerning the company's development taken at a strategic level?
  - (a) Weekly
  - (b) Monthly
  - (c) Quarterly
  - (d) As and when required
  - (e) Varies from project to project
24. How well do you think that the ongoing projects within the organization are aligned with the decisions made on the company's development?
  - (a) Very well
  - (b) well
  - (c) Less well
  - (d) Bad
  - (e) Don't know
25. Are company's decisions proactive or reactive, that is, are the decisions a product of having investigated customer needs or a reaction to urgent customer demands?
  - (a) Mainly proactive
  - (b) Mainly reactive
  - (c) Depends on project
  - (d) Don't know
26. Has the share of reactive projects increased or is it indifferent comparing to last year?
  - (a) Increased
  - (b) Indifferent
  - (c) Decreased
  - (d) Don't Know

27. How often are projects finished earlier then estimated? ( Number of project finished earlier then estimated/ total amount of projects)  
State the average percentage of the projects
28. How often do projects take longer to complete then estimated (Number of projects that take estimated? Longer to complete/total amount of projects)
29. How well, at the project level, does the time reports reflect how the time is actually used?  
Do time reports give an idea of the stage of the project?
30. How are resources distributed in
  - a) Software Development, Beta –Testing, Customer Support Report (CSR).
  - b) Different phases within Software development.  
Software Development includes: Work performed before the product is sent for beta-testing.  
CSR –resources required to fulfill customer needs after the product has been given to him.
31. To what extent of the requirements being implemented are there written/formal project management tools stated to be used?  
State in percent/project
32. Who develops functionality test cases?
33. How often is the formal feedback on the employee's work taken?
  - (a) After every project closure
  - (b) During and after project closure
  - (c) once a month
  - (d) once every half a year
  - (e) once a year
34. At what level within the company are the verified failure reports investigate/traced?  
Department
  - (a) Project
  - (b) Individual
  - (c) Not at all
35. At project start, what knowledge is given to the developers/managers?
36. Are alpha tests performed with the help of automated methods?
  - (a) Yes
  - (b) No
  - (c) Don't know

37. Does the developer have a possibility to affect the composition of a requirement?  
If yes, then state whether the changes can be made at Customer specific requirements or internal requirements.
  - (a) Yes
  - (b) No
  - (c) Don't know
  
38. Are requirements subdivided into different priority classes?
  - (a) Yes
  - (b) No

If yes, how many different priority classes are there?  
Are requirements implemented in this priority order?
  
39. Is it specified to what extent programs will be affected by each requirement?
  - (a) Yes
  - (b) No
  
40. Is it specified how many customers that are affected by each requirement?
  - (a) Yes
  - (b) No
  
41. How much of a project has been completed when the last requirements are being added? State the average per cent!
  
42. What are the project documentation produced to document events and decisions?
  
43. Time devoted to each
  - a) Requirement specification
  - b) Analysis and design
  - c) Implementation
  - d) Testing
  - e) Documentation
  - f) Reflection and Feedback
  
44. Frequency formal test developed in order to control that functionality goals are met?*(that the functionality is fulfilled)*  
State the average per cent Average
  
45. To what extent of the new functionality are test cases developed in order to guarantee that the demands placed upon the new functionality are fulfilled?

46. How often is a linking disrupted due to that a none buildable module has been checked in on the branch?
- (a) Never
  - (b) Sometimes
  - (c) Often
  - (d) Always

How long time does it take to fix this error?

47. How is the origin of your work tasks spread?
- (a) Financed by customer
  - (b) Financed by customer and company
  - (c) Financed by company
  - (d) Depends on the project
48. At the project start, how often do you possess knowledge of other projects that depend on your own project?
- (a) Never
  - (b) Sometimes
  - (c) Often
  - (d) Always

## **A.2 RESULTS**

The results presented in this section are from the completed questionnaires by the employees at HSS in the year 2001. The questionnaire was handed out to 30 employees in the software development department at various levels and 28 were received more or less completed. The questions referring to each answer is not presented in this section while they have been presented in section A.1.

In the case of questions requesting a choice, percentage of employees answering each choice has been indicated. To the questions requesting numerical answers the range in which the answers belong, has been specified.

1. Strategy: follows strategy of customer's satisfaction  
Market: foresee the scenario and study the market thoroughly  
Customers: most concerned of their needs and requirements
2. Modified waterfall model
3. 90% say that it investigates the development process of the vendors before assigning any task to them.

It is a job of top management.

6. Yes -75%  
No-20%  
Can't say-5%
7. Collaborating departments:  
Business group, PQT, Finance, Quality, IT, Training, Recruitment,  
Engineering
8. Yes-85%  
No-13%  
Don't know-2%
11. Once a month-0%  
2-3 times per month-0%  
4-5 times per month-100%  
more than 5 times per month-0%  
time sheet and status sheet is evaluated by manager.
12. Monthly-100%  
Quarterly-0%  
Twice a year-0%
13. Weekly- 90%  
Twice a month-0%  
Monthly-0%  
Depends on project-10%
14. Weekly
15. Cost: Effort(man months)  
Time: Schedule(calendar months)  
Quality: Defect density
16. 1. Anything which is not acceptable to user.  
2. Anything that effects the proper functioning of the system as per  
the user requirements.
17. COCOMO II  
FPA  
Past productivity figures
18. Once in a project – 4%  
Once in each phase – 6%  
Atleast once for each artifact- 90%

- Not necessary - 0%
19. Variance at the company level is 8-10%
20. Very well represented -100%  
Well represented -0%  
Less well represented -0%  
Badly represented -0%
21. Yes, know the basis -20%  
Yes, don't know the basis -80%  
No -0%
22. Never -3%  
Sometimes -92%  
Often -5%  
Always -0%
23. Weekly -3%  
Monthly -0%  
Quarterly -0%  
As and when required -80%  
Varies from project to project -17%
24. Very well -5%  
Well -85%  
Less well -5%  
Bad -0%  
Don't know -5%
25. Mainly proactive -70%  
Mainly reactive -0%  
Depends on project -20%  
Don't know -10%
26. Increased -1  
Indifferent -3  
Decreased -1  
Don't Know -95
27. Top management says- less than 10%  
Developers say they don't know
28. Answers vary in the range of 60 %- 70%.

- 29. At the project level, time reports reflect the time actually used quite well.  
Yes -97%  
No -0%  
Don't Know -3%
- 30. Software Development -85%  
Beta -Testing -10%  
Customer Support Report (CSR) -5%
- 31. Answer varies in the range of 80% - 95%.
- 32. Testing Team
- 33. After every project closure -2%  
During and after project closure -85%  
once a month -10%  
once every half a year -0%  
once a year -0%  
can't say -3%
- 34. Department -5%  
Project -95%  
Individual -0%  
Not at all -0%
- 36. Yes -96%  
No -1%  
Don't know -3%
- 37. Yes -100%  
No -0%  
Don't know -0%
- 38. Yes, know the priority classes -40%  
Yes, don't know the priority classes -60%  
No -0%
- 39. Yes -100%  
No -0%
- 40. Yes -97%  
No -4%
- 41. Answers vary in the range 60%-70%

- 42. Each phase has Output controlled documents and quality records.
- 43. Requirement specification –10-15%  
Analysis and design –30-35%  
Implementation –25-30%  
Testing –20-25%  
Documentation –included above  
Reflection and Feedback –5-10%
- 44. Answers vary in the range 95%-100%
- 45. 100%
- 46. Never –0%  
Sometimes –75%  
Often –0%  
Always –0%  
Can't say –25%
- 47. Financed by customer –15%  
Financed by customer and company –10%  
Financed by company –10%  
Depends on the project –65%
- 48. Never –0%  
Sometimes –0%  
Often –10%  
Always –90%

## References

1. Armitage, James W., Kellner, Marc I., and Phillips, Richard W., "Software Process Definition Guide—Content of Enactable Software Process Definitions (CMU/SEI-93-SR-18). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, August 1993.
2. Bill Munro, "Measuring IT Project Performance", <http://www.tsepm.com / spring01 / article3.html>
3. Daniel Galorath, Lee Jischman, K. Mckitchie, "During quality through parametrics," [www.galorath.com](http://www.galorath.com)
4. Donald J. Reifer, "Software Management's Seven Deadly Sins", IEEE Software March/April 2001.
5. Eldon Wig, "Decisions to Focus By", Feature Article <http://www.spc.ca/download/resources/essentials/issue18.pdf>
6. Eldon Wig, "The Dynamic Duel", Feature Article <http://www.spc.ca/download/resources/essentials/issue14.pdf>
7. Eldon Wig, "We Built it, But they don't Buy", <http://www.spc.ca/download/resources/essentials/issue8.pdf>
8. Elias M. Awad, "System Analysis and Design", Second Edition, Galgotia Publishers.
9. Fagan M.E., "Design and Code inspections to reduce errors in program development", IBM Systems Journal, Vol 38, Nos 2&3, 1999
10. Fairley R, "Software Engineering Concepts", Tata McGraw Hill Publications.
11. Flippo E.B. "Personnel Management", 6<sup>th</sup> Edition, McGraw Hill International Editions.
12. Ghezzi C, Jazayeri M, Madrioli D, "Fundamentals of Software Engineering", Prentice Hall India.
13. Gibson R., "Characteristics of organizations that achieve high process maturity levels," [aisel.isworld.org/pdf.asp?vpath=AMCIS/2000&PDFpath=092.pdf](http://aisel.isworld.org/pdf.asp?vpath=AMCIS/2000&PDFpath=092.pdf).
14. Gilb T, "Principles of Software Engineering Management", Addison-Wesley.
15. Gilb T, Graham Dorothy, "Software Inspection", Addison-Wesley
16. <http://www.enteract.com/~bradapp/links/sw-proj-mgmt-links.html>

17. <http://www.iitk.ernet.in>
18. <http://www.literateprogramming.com/mfound.pdf>
19. <http://www.processimpact.com/articles/metrics-primer.html>
20. <http://www.qai.usa.com>
21. <http://www.result-planning.com>
22. <http://www.sei.cmu.edu/managing>
23. <http://www.spc.ca/resources/prodmgmt>
24. <http://www.sqi.gu.edu.au/SPICE>
25. <http://www2.umassd.edu/SWPI/measurement/measurement.html>
26. Hudson F.J, Behforooz A, "Software Engineering Fundamentals".
27. IEEE Standard collection on S/W Engg., (1994), IEEE Press.
28. James D. Herbsleb, D Moitra, Lucent Technologies "Global Software Development", IEEE Software March/April 2001.
29. Jeffrey Voas, "Faster, Better, Cheaper", IEEE software, " [www.computer.org/software/homepage/2001/03voas](http://www.computer.org/software/homepage/2001/03voas).
30. John J. Machiniek, "Encyclopedia of S/W Engg.," Vol. 2, Wiley International Science Publishers.
31. Kothari C R, "Research Methodology: Methods and Techniques", Second Edition, Wishwa Prakashan.
32. M.Barbacci, M.H.Klein, T.A.Longstaff, "Quality Attributes", CMU/SEI-95-TR-021, Software Engineering Institute, Carnegie Mellon University, December 1995.
33. Maddox, "Enterprise-wide Processes Boost Productivity, Competitiveness" [www.Software.org/quarterly/1297/tracos.GDE.html](http://www.Software.org/quarterly/1297/tracos.GDE.html)
34. P. Jalote, "An Integrated Approach to Software Engineering", Springer-Verley. 2<sup>nd</sup> Edition, 1994.
35. P.Hollington, "Where's the PM Beef?", Feature Article at [www.spc.ca / essentials/resources](http://www.spc.ca/essentials/resources).
36. PMBOK, Project Management book of Knowledge, 1996;
37. Pressman R.S., "Software Engineering: A Practitioner's Approach", Fourth Edition, McGraw Hill Publications.

38. Prez, M., Rojas T., Perras L., "Systems quality and the organisations culture," [aisel.isworld.org/pdf/asp?vpath=AMCIS/2000&PDFpath=093pdf](http://aisel.isworld.org/pdf/asp?vpath=AMCIS/2000&PDFpath=093pdf).
39. R.V. Solingen, E. Berghout, "The Goal/Question/Metric method: A Practical guide for Quality Improvement of Software Development", McGraw Hill Publishing, <http://is.twi.tudelft.nl/gqm>
40. Robbins S P, "Organizational Behavior", Eighth Edition, Prentice Hall India.
41. Robert E.Park,W.B.Goethert,J.T.Webb, "Software Cost and Schedule Estimating: A Process Improvement Initiative", CMU/SEI-94-SR-3 , Software Engineering Institute, Carnegie Mellon University, May1994.
42. Saxena A.K., "Statistical Process Control and Quantitative Quality Management in Software Processes", IIT Kanpur, May2001.
43. Soumerville I, *Software Engineering* , Prentice Hall India
44. Stan Rifkin, Master Systems, "What makes measuring software so hard ? IEEE Software, [www.computer.org/software/homepage/2001/03v.rifkin](http://www.computer.org/software/homepage/2001/03v.rifkin).
45. Steve McConnell, "The Best Influences on Software Engineering", IEEE Software January/February 2000.
46. SWEBOK, Software Engineering Book of Knowledge
47. Warren Keuffer, "No Silver bullets", Software Development Magazine, April 2001
48. Warren Keuffer, "Best practices Actually Applied", Software Development Magazine, [www.sdmagazine.com/documents](http://www.sdmagazine.com/documents)
49. William A. Florac, "Software Quality Measurement: A Framework for Counting Problems and Defects", Technical Report, CMU/SEI-92-TR-22, Software Engineering Institute, Carnegie Mellon University.