DESIGN AND DEVELOPMENT OF A SOFTWARE PROCESS IMPROVEMENT MODEL

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

Master of Engineering
in
(Software Engineering)

By:

Pankaj Kumar
(80731017)

Under the supervision of:
Dr. Seema Bawa
Professor
Computer Science & Engineering Department
Thapar University
Patiala (Punjab)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

JUNE 2009

Certificate

I hereby certify that the work which is being presented in the thesis entitled, **"Design and Development of a Software Process Improvement Model"**, in partial fulfilment of the requirements for the award of degree of Master of Engineering in Software Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Seema Bawa* and refers other researcher's works which are duly listed in the reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

(Pankaj Kumar)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(**Dr. Seema Bawa**)
Professor, Computer Science and Engineering Department
Thapar University,
Patiala

Countersigned by

To
The Head
Computer Science & Engineering, Department
Thapar University
Patiala

(R.K.SHARMA)
Dean (Academic Affairs)
Thapar University,
Patiala

ACKNOWLEDGEMENT

First and foremost, I would like to express my sincere gratitude to my supervisor **Dr. (Mrs.) Seema Bawa**, Professor, Computer Science & Engineering Department for her immense help, guidance, stimulating suggestions and encouragement all the time. She always provided a motivating and enthusiastic atmosphere to work with; it was a great pleasure to do this thesis under her supervision.

I am most indebted to the P.G coordinator **Dr. Inderveer Chana** for her help, assistance and suggestions during my work. I am equally grateful to **Mrs. Anju Bala** Lecturer, Computer Science & Engineering Department, for her kind help and support in giving final shape to this work.

I would also like to express my appreciation to the TU Grid Group. The group held weekly meetings on Thursdays to address various issues and share experiences. Thus, it invoked better understanding of the work and acted as a rostrum for information sharing and problem solving.

Finally, my special thanks go to my family for their never-ending love and support. Nothing would have ever been possible without my parent's unconditional love and encouragement. The most valuable thing I acquired from the two years study in TU is to protect, survive and endure myself in this world with increased confidence and additional faith in my abilities to achieve. Last but not the least; I would like to thank the Almighty Lord for His blessings and for driving me with faith, hope and courage in the thinnest of the times.

Pankaj Kumar
(80731017)

ABSTRACT

Software Process Improvement (SPI) is a systematic approach and continuous improvement of software development organization's ability to produce and deliver quality software within time and budget constraints [1]. In the course of this project, the researcher proposed, developed, verified and validated such a model.

This thesis work presents a fit for use approach to software process improvement model for improving the overall process. This thesis work concentrates on improvement in process as well as in organisation. This model is based on experience gained on projects by software companies. This thesis work covers assessment, software process improvement, factors that influence the software process improvement.

The aim is to develop a model which would be useful in practice for software development companies. This thesis work describes the principles of software process improvement and improvement models. The proposed model is a generic model which is beneficial for a small firm as well as a large firm. This proposed model does not focus only on quality assurance but also the culture of the organization. It describes how this model can be used to guide software organizations in process improvement, and the various assessment and evaluation methods.

Key words software process improvement, software process improvement model, software quality assurance.

Organization of Thesis

The first Chapter briefly introduces software process improvement concepts. It explains concept of software process improvement and describe the steps to be carried out for software process improvement.

The second Chapter presents Review of Literature related to software process improvement models. It covers the aspects of existing software process improvement and some software process improvement models and CMM, SIX SIGMA, ISO, IDEAL, TICKIT, TQM and extra. It also describes some software process improvement methods.

The third Chapter covers Problem Formulation for this thesis work and explains briefly how the problem can be solved. This chapter also describes the limitations of some software process improvement models.

The fourth chapter covers the comparisons between the existing software process improvement models.

The fifth chapter explains the Software Process Improvement Model (SPIM). Firstly, this chapter describes the existing problems in software projects, after that it defines the d model SPIM model.

The sixth chapter describes how the SPIM model works better than the other models.

Finally, thesis work has been concluded in seventh Chapter along with the mention of scope for future work.

TABLE OF CONTENT

List of Figures

List of Tables

Chapter 1

INTRODUCTION

A crisis came in software engineering around 1965-1985. This time the failure of the project was increased. The cause of the software crises were linked to the overall complexity of the software process and the relative immaturity of software engineering as a profession. The crises manifested it in several ways:

- Project running over budget
- Project running over time
- Software was low quality
- Software often did not meet requirements
- Project was unmanageable and code difficult to maintain [4]

The crises in software are well known. In order to complete the project successfully in the international market, as software professional, need to improve the quality of software product. Then it was found that if some activities are following then good quality software could be developed. The concept behind this proposal presented in this thesis is that a generic Software Process Improvement model, can be used to help a company improve their software process in an affordable manner while also keeping business goals in focus. The ultimate aim is to develop a model which would be useful in practice for software development companies. In the course of this project, the researcher developed, verified and validated such a model. During the 1990s there has been a movement in the software development community towards the improvement of the software process as a means to improve the quality of software. Small software development companies are important to the Irish economy, and therefore, should keep up-to-date with global developments. Existing models such as the Capability Maturity Model, Boot Strap and ISO 9000, and the proposed international model SPICE, have been criticised, particularly in relation to the small company, as being cumbersome, expensive and failing to present a comprehensive improvement strategy. Another reason why it is important is that a process provides organizational stability and more control to its activity quality workforce and the latest technology. We do not want to realize that even the finest people cannot perform at their best. Many organizations are adopting the best practices in software development, such as those based on Capability Maturity Model (CMM), ISO 9001,

or Six Sigma. There are conflicting influences on software development effort, quality, and cycle time. Cycle time may be compressed at the cost of quality, experienced professionals may improve quality but at increased costs, quality may be achieved at the cost of increased testing effort. Larger team size may reduce development time while raising total costs, process maturity may improve quality but at high cost, and so forth. One of the most important consequences of improved processes is superior conformance quality. The reduction in variability is likely to be most pronounced in development organizations at CMM level 5. This is the highest level of process maturity as per the Software Engineering Institute (SEI) located at Carnegie Mellon University. As a result of previous research work, the researcher proposes a Software Process Improvement Model.

1.1 SOFTWARE PROCESSES

Software process is the task or activities used for produce a software product. It is not necessary its output is software. A process can generate the other process. It is a set of activities, methods, practices, and transformations that people use to develop and maintain software and its associated products [3] (e.g., project plans, design documents, code, test cases, and user manuals).

1.2 SOFTWARE PROCESS IMPROVEMENT

Process is a set of activity which gives an output. In the sense of software it is said to software process. Which result is software. An author defines the definition of process improvement is like that

"Software process improvement is a deliberate, planned methodology following standardized documentation practices to capture on paper (and in practice) the activities, methods, practices, and transformations that people use to develop and maintain software and the associated products. As each activity, method, practice and transformation is documented, each is analyzed against the standard of value added to the organization" [2].

Process improvement is a change in the process to get a better output rather than original output. A lot of factor are play important role in software process. Project should be with in time under budget and better quality for long term business. Software process improvement is a set of activities by which can get a maturity in our

process and fulfil all our goals and get a high quality product and maturity in organisation.

Software process improvement is a set of activity by which can get a higher level of maturity. It is important to understand the vocabulary used in describing the software process and the maturity model. These terms are used in a particular way, and it is important to know their meaning in order to comprehend the model. It is also necessary to understand that a software process model is not a mathematical formula. In this context, it is a description of how to conduct the process of software development.

Many software process techniques, such as quality assurance, configuration management, inspections, and reviews, are described in most software engineering textbooks and will not be covered here. From a general problem-solving point of view, project management may be described simply as:

- Identifying the processes
- Decide whether the improvement needed
- Analyse
- Then done the improvement

Software process improvement term is used for improving the organization. There are some activities for this;

- Identify the current process
- Identify the current status of the organization
- Identify the strength and weakness of the organization
- Analyse where the changes are needed
- What will be the effect of these changes?
- Understand these changes that how much they are affected

A simple software process improvement life cycle is presented in the Fig. 1.1.

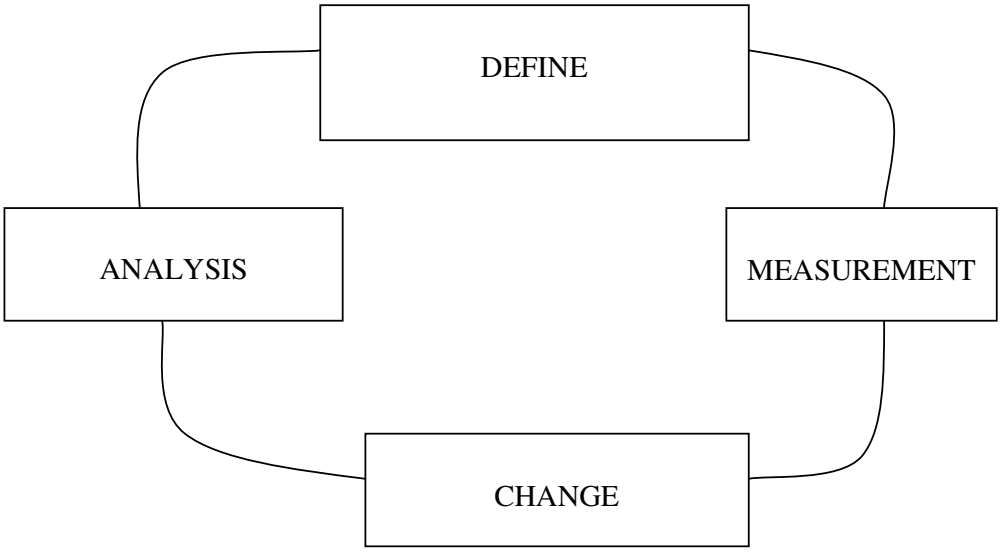


Fig 1.1: A SOFTWARE LIFE CYCLE

Process improvement is a change in the process to get a better output rather than original output. A lot of factors play important role in software process likewise:

- Project should be with in time
- Under budget
- Better quality
- For long term business
- Reuse

Software process improvement is a set of activities by which we can get a maturity in our process and fulfil all our goals and get a high quality product and maturity in organisation. Generally software process improvement works on PDCA (plan, do, check, and act). First, see whether improvement is needed, then, plan according to the problem and then, check this is right or not. If this is not suitable then improve it, otherwise, implement.

1.3 SOME STEPS FOR THE PROCESS IMPROVEMENT

a) AWARE FOR ORGANIZATION CULTURE

Always be aware to the organization culture. Know about the things which are important for the organization. What current technology is in the picture? What should come in future? Develop your self according to them.

b) CONSISTENCE SHOULD BE THERE

Consistency means discussion with all the members of the team. Before starting the work, a formal meeting should be conducted, to know whether all things are complete. No requirement is left. Divide the work according to the ability of suitable person. Commitment also plays a very important role in this, because the person who takes the responsility of a particular work should complete the work within time and respect the seniors.

c) MAKE TARGET AND KEEP IN MIND

When project comes to the company. Discuss with the customer very frankly. According to the requirements, estimates the size of the project. And make the target what we are going to do in future. All requirements should be clear otherwise some problem may be faced in the middle of project. Then work should be done according to the plan.

d) PUT A OPTION IN THE PROCESS

When people of an organization works on a project. They keep a copy of that work. If there is any problem in that project at any stage for some reason. Then work has to be done again from the baseline. For example, if a project has to be done in .NET but some problem is faced in doing that project in .NET, then, use the alternate process for completing the project.

e) COMMUNICATION BETWEEN THE TEAM MEMBER AND SENIOR MANAGEMENT

A good communication should be established among the team management and senior management. An employee can discuss with the seniors directly. They can give

the new ideas about the problems. Seniors should listen to these ideas and should give a proper response to these ideas. They should promote people not only on the basis of experience but also on the basis of their knowledge.

f) TREAT PROCESS IMPROVEMENT LIKE A PROJECT

Be serious for the process improvement. Define the problems for the process, then make a suitable document for these, make the plan according to them, then implement it in a right way to improve the process. There should be an experienced manager and a well qualified team for the process improvement.

g) MAKE CHANGES ACCORDING TO THE PRIORITY

Process improvement should not be done just once. It should be given priority. Priority should be according to the various standards. Such as, it should be done with CMM or SPICE or ISO standard.

h) MAKE A PROFILE OF THE COMPANY

First make the profile of the organization. Collect the strength and weakness of the company. It should be done by review. Every thing has a strong point and weak point. Make a profile for this. And concentrate on how to remove the weaknesses.

i) INTRODUCE A SOFTWARE PROCESS ENGINEERING GROUP (SEPG) TO YOUR ORGANIZATION

There should be a group in the organization which is well qualified and can inspect the process. Shortcomings are to be told to the members and provide the advice to improve them. This group should be well qualified and certified by the Software Engineering Institute (SEI). This group should take a lot of responsibility for the organization. So member of the team should be highly qualified.

j) HAVE PATIENCE

Some time process may be slow and costs are over budget [5]. So it is important to have patience. Sometime this process takes a long time.

k) REUSE THE PREVIOUS RESULT

Software process improvement is a methodology or a model. If this methodology becomes successful. Next project could use this result.

1.4 WHY DO WE NEED SOFTWARE PROCESS IMPROVEMENT?

Software process improvement makes an immature organization to a mature organization. Immature organization can not generate a good quality product. There are a lot of reasons for failure of good quality software. Their work methodologies make no sense, no support between the senior management and management. Project is only work at one time, no reuse, and no proper documentation of the project. Project is not according to customer demand. Customer requirement is other than project. Project is not completed at time and budget exceeds. There are a lot of shortcomings in immature organization. There is no man power. Whereas, a mature organization release provide a good quality to customer with in proper time and budget. There is commitment between the employees. If there is any problem between management then senior management handle the problem and give a good solution [21].

1.5 THE COMPARISON BETWEEN THE IMMATURE AND MATURE ORGANIZATION

Table 1.1: A Software Process Maturity Model [6]

LEVEL	CHARACTERISTIC	KEY PROBLEM AREAS
Optimising	Improvement feed back into process	Automation
Managed	(quantitive) measured Process	changing technology problem analysis problem prevention
Defined	(qualitative) process defined & institutionalised	process measurement process analysis quantities quality plans

Repeatable	(intuitive) process dependent on individuals	Training technical practices process focus
Initial	(adhoc/chaotic)	project management project planning configuration management software quality assurance

Table 1.2: A Software Process Immaturity Model [6]

LEVEL	**CHARACTERISTICS**	**KEY PROBLEM AREAS**
Foolish	(negligent) failure to allow successful development process	software reuse
Stupid	(negligent) failure to allow successful development process	Development environments Repositories
Lunatic	(contemptuous) disregard for good software engineering institutionalised	automatic programming

With the help of Software Process Improvement an immature organization can produce a good quality project.

Chapter 2

LITERATURE REVIEW

In November 1986 the software engineering institute (SEI), with assistant from the MITRE Corporation, began developing a process maturity framework that would be help organizations improve their software process [1]. In September 1987 the SEI released a brief description of process maturity framework and a maturity questionnaire. The SEI signifies the maturity questionnaire to provide a simple tool for identifying areas where an organization's software process needed improvement. After four years of experience with the software process maturity framework and the preliminary version of the maturity questionnaire the SEI evolved the software process maturity framework into the Capability Maturity Model (CMM). The CMM is based on knowledge acquired from software process assessment and extensive feedback from both industry and government. Several specific approaches to process improvement have become popular in the software industry. The include the following:

- Capability Maturity Model
- Six Sigma
- ISO standard series
- Total quality management
- Software process improvement and capability improvement
- IDEAL model
- Bootstrap model

2.1 HISTORY

2000- ISO 9000:2000 edition released
SEI CMMI V1.02 released

1998- ISO 15504 (SPICE) released to public as "type 2" Technical Reports
Tick IT V4.0 released

1997- 9000-3 re-released

1996- SEI halts SW-CMM revisions in support for CMM Integration (CMMI)
IEEE/EIA 12207 released (this is ISO 12207 with extra guidance)

1995- ISO 12207 released (initial release) ISO 15504 (SPICE) initial "draft" released

1994 ISO 9001 re-released

Trillium V3.0 released

1993 SEI SW-CMM V1.1 released

1992 EFQM/BEA: 1st Business Excellence Award (Europe) IEEE adopts the Australian AS 3563 as "IEEE 1298"

TickIT V2.0 released

IEEE 1074 released (initial release)

Improve IT V1.0 released (this is the beginning of Tick IT)

1991- ISO 9000-3 released (initial release)

SEI SW-CMM V1.0 released (initial release of model)

Trillium V1.0 released (initial release)

AS 3563 (Software Quality Management System) standard released-initial release

ISO 9001 released (initial release)

1988- NIST/MBNQA: 1st Malcolm Aldridge National Quality Award (USA)

1987- SEI-87-TR-24 (SW-CMM questionnaire) released

1983- NQI/CAE: 1st Canadian Award for Excellence (Canada) [8]

2.2 IMPROVEMENT THEMES

- **Total Quality Management:** Canada's CAE, Europe's BEA, and USA's MBNQA
- **Process Definition:** IEEE 1074, ISO 12207, and IEEE/EIA 12207
- **Quality System Definition/Improvement:** ISO 9001, ISO 9000-3, TickIT and AS 3563
- **Process Improvement:** SEI SW-CMM, SEI CMMI, Trillium, and ISO 15504 (SPICE)

2.3 APPLICATIONS/OUTCOMES

- **Award:** Canada's CAE, Europe's BEA, and USA's MBNQA
- **Compliance:** AS 3563, IEEE 1074, ISO 9001, ISO 12207, ISO 15504 (SPICE), SEI SW-CMM, SEI CMMI, Tick IT, and Trillium

- **Guidance:** all standards and models

2.4 REGIONAL FOCUS

- **International:** ISO 9000-3, ISO 9001, ISO 12207, and ISO 15504 (SPICE)
- **National:** Australia's AS 3563, Canada's CAE, Europe's BEA, and USA's MBNQA
- **Not specified:** IEEE 1074, IEEE/EIA 12207, SEI SW-CMM, SEI CMMI, TickIT and Trillium

And a lot of models are available for process improvement. A model provides organisation to more effective guidance for establishing process improvement programs. The first version of CMM reviewed in 1987. It is used by the community by 1991 and 1992. The IEEE defines “a process as a sequence of steps performs for a given purpose” [IEEE-STD-610]. A software process can be defined a set of activities, method practices and transformations that people use to develop and maintain software and the associate product (e.g., project plans, design document, code, test cases, and user manuals). A mature organisation make better plan and defined the project and more consistently implemented throughout the organization. Software process capability describe of an organisation describe the expected result that can be achieved by following a software process.

2.5 CMM (CAPABILITY MATURITY MODEL)

Capability maturity model (CMM) is a popular planned methodology for improving the software process. This is developed by software engineering institute (SEI) in 1987. CMM methodology is a continuous improvement as well as process improvement. This methodology is become so popular that it integrated and new version come is called the CMMI (Capability maturity model integration). The capability maturity model can be described as a structured collection of elements that describe certain aspects of maturity in an organization, and aids in the definition and understanding of an organization's processes.

2.5.1 HISTORY OF CMM

- 1987: SEI-87-TR-24 (SW-CMM questionnaire), released.
- 1989: Managing the Software Process, published.
- 1990: SW-CMM v0.2, released.
- 1991: SW-CMM v1.0, released.
- 1993: SW-CMM v1.1, released.
- 1997: SW-CMM revisions halted in support for CMMI [9].

CMM provides the five levels. These are called the maturity level. The CMM helps organizations to select improvement strategies based on current process maturity status and to identify critical issues in quality and process improvement.

Structure of CMM

2.5.2 MATURITY LEVEL

CMM provides five level of maturity. These are sequentially step.

2.5.3 KEY PROCESS AREA

In every maturity model some functionality are defined. The area of each has been restricted. This functionality is called the key process area.

2.5.4 GOALS

Every maturity level defines some goals. The goals signify the scope, boundaries, and intent of each key process area.

2.5.5 COMMON FEATURE

There are five type of common feature in CMM.

There are five type of common feature.

- Commitment to Perform
- Ability to Perform
- Activities Performed
- Measurement and Analysis
- Verifying Implementation.

2.5.6 KEY PRACTICES

These key practices describe the elements of infrastructure and practice that contribute most effectively to the implementation and institutionalization of the KPAs.

2.5.7 MATURITY LEVEL

CMM consists of five maturity levels. CMM became very popular. CMM has been integrated and it became capability maturity model (CMMI). CMM helps organizations to select improvement strategies based on current process maturity status and to identify critical issues in quality and process improvement [7].

Following are the five maturity levels of CMM:

- Initial
- Repeatable
- Defined
- Manage
- Optimizing

2.5.7.1 INITIAL

In the initial level the software process is on adhoc basis. The process capability is unpredictable and has no formal planning. No response of the senior management is there. No change management, no formal technique review is present. Only few processes are defined and basically, the success of the project is heroic.

2.5.7.2 REPEATABLE

At the repeatable level well planning is there. Project control has been stated by quality assurance, change control and schedule. Previous project success could be repeated. Planning and management should be based on the experience of the previous results. Software requirements should be clear.

2.5.7.3 DEFINE

In this level standard process includes both software engineering and management activities. All planning should be documented. In this level, organization establishes a group SEPG (Software Process Engineering Group) to keep the improvement and

management activities. Organization needs to deliver the software with in time and budget.

2.5.7.4 MANAGE

This level is also known as the quantitative level. All the measures of the software process and productivity are performed quantitatively. Proper databases are needed to keep the data. Software processes are instrumented with well-defined and consistent measurements. The software process capability of Level 4 organizations can be summarized as being "quantified and predictable" because the process is measured and operates within quantitative limits [10].

2.5.7.5 OPTIMIZING

In this level, organizational process focuses on the continuous improvement. The organization identifies their strength and weakness of their company. Accordingly they detect the defects in the company and try to remove them. Defect prevention is the main activity of this level.

2.5.8 ARCHITECTURE OF CMM

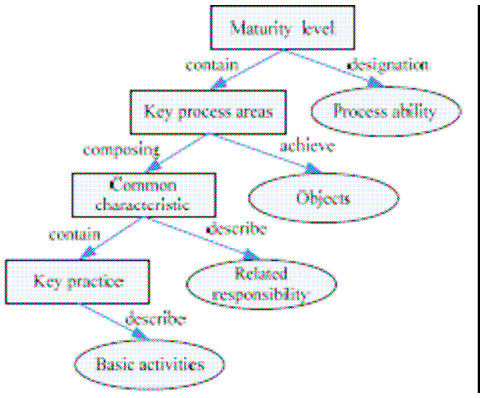


Fig 2.1 Architecture of CMM [11]

2.5.9 KEY PROCESS AREAS

Each level (Level 1, 2, 3, 4, 5) has been divided into certain KPAs. For a company to achieve a certain maturity level it must fulfil all the KPAs of the desired maturity level [13].

2.5.9.1 LEVEL 1 No key process area.

2.5.9.2 LEVEL 2 There are 6 key process areas

- Requirements Management
- Software Project Planning
- Software Project Tracking & Oversight
- Software Subcontract Management
- Software Quality Assurance
- Software Configuration Management

2.5.9.3 LEVEL 3 In this level there are 7 key process areas

- Organizational Process Focus
- Organizational Process Definition
- Training Program
- Integrated Software Management
- Software Product Engineering
- Intergroup Coordination
- Peer Reviews

2.5.9.4 LEVEL 4 In this level there are 2 key process areas

- Quantitative Process Management
- Software Quality Management

2.5.9.5 LEVEL 5 in this level there are 3 key process areas

- Defect Prevention
- Technology Change Management
- Process Change Management

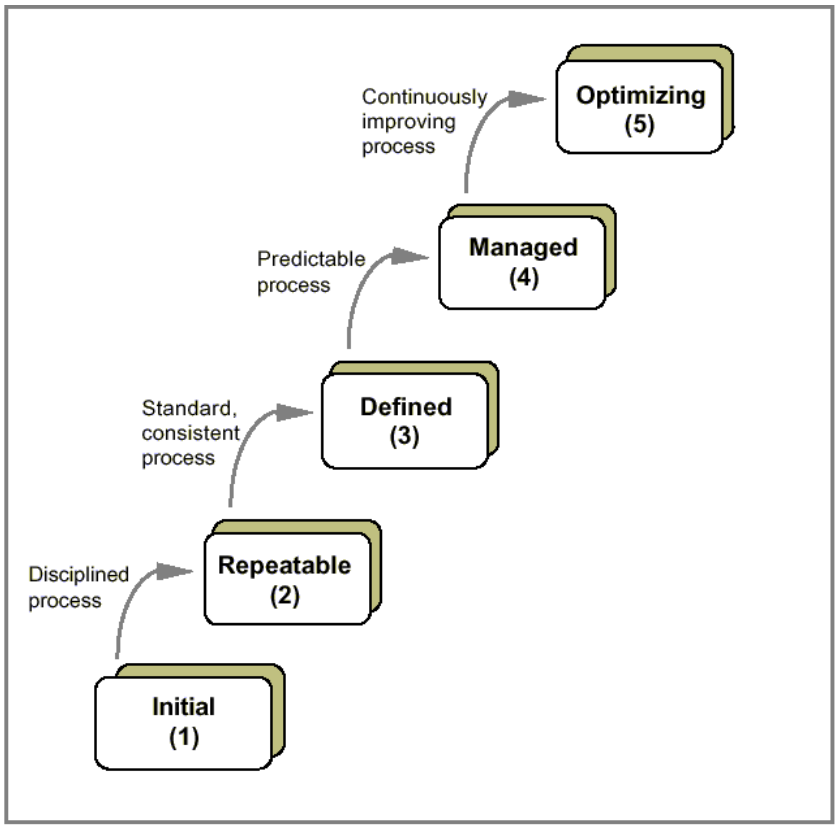


Fig 2.2 CMM Model [17]

2.6 IDEAL MODEL

IDEAL model is developed by software engineering institute. IDEAL model is an approach to continuous improvement by some steps which are necessary to establish a successful improvement plan. These activities are beneficial in software process improvement efforts. This model defines some steps according to which step by step improvement takes place. This model provides a disciplined approach for improvement program and establishes a long term improvement strategy. This model consists of five phases. Every phase has its own functionality.

These phases are following:

2.6.1 PHASE 1

2.6.1.1 INITIATING

This phase provides groundwork for all phases. Model starts with this phase in which all the estimation work is done like project estimate, team formation and sponsorship. Team formation is done according to the project.

2.6.2 PHASE 2

2.6.2.1 DIAGNOSING

This phase defines the work process which has to be performed. It first recognizes the strength and weakness of the company and then makes a feature of the future. It states what should be done with the project. Then a simple documentation is done.

2.6.3 PHASE 3

2.6.3.1 ESTABLISHING

This phase provides the planning which will help in reaching the destination. After the second activity make goal. Make the plan for that. Make the strategies at which way we can get our goals.

2.6.4 PHASE 4

2.6.4.1 ACTING PHASE

Now implement these plans. Make methods for accomplishing the plan and chose the shortest and easiest path. Then we take a pilot test to see whether it is going in a right direction or not. If not then go for second phase. Otherwise implement this method.

2.6.5 PHASE 5

2.6.5.1 LEARNING PHASE

Learning from the experience and improving ability to adopt new technology in future.

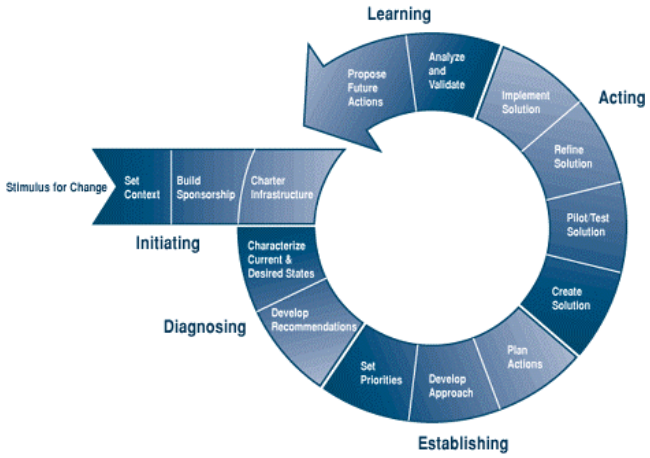


Fig 2.3: Framework of IDEAL model [12]

2.7 INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO)

ISO is a standard of quality management system. This organization gives some standard to the company. ISO is applicable for both software and hardware product. ISO certification is necessary to improve the organization. Standards ensure desirable characteristics of products and services such as quality. Customer can know the profile of the company associated with the ISO standard. Once an organization is certified with ISO, the ISO team audits this organisation next every six months. On behalf of this audit ISO, renews the certification.

The ISO 9000 series consists of three standards

- ISO 9000:2000, Quality management systems – Fundamentals and vocabulary
- ISO 9001:2000 Quality management systems – Requirements
- ISO 9004:2000 Quality management systems - Guidelines for performance improvements [22].
- ISO 10007:1995
- ISO 10006
- ISO 14000
- ISO 19011
- ISO/TS 16949

2.8 SPICE MODEL

The SPICE model provides a framework for the assessment of software process. This framework can be used by organizations involved in the planning, managing, monitoring, controlling, and improving the acquisition, supply, development, operation, evolution, and support of software [27].

SPICE identifies five levels of process capability (which are very similar to the levels of process maturity found in SEI' CMMI) [28] [25].

2.8.1 NOT PERFORMED

There are no features in this level. The work product or output of the process cannot be easily identified. There are no best practices in the process. No good knowledge of the processes is there.

2.8.2 PERFORMED-INFORMALLY

This is based on the best practices of the process. Processes are well performed. Performance depends on the individual performance. There is identifiable work product of the process.

2.8.3 PLANNED AND TRACKED

Processes are well defined and then tracked. The organisation moves towards a well defined process. Requirements should meet with the product. Performances are based on the best practices.

2.8.4 WELL DEFINED

Base practices are performed according to a well defined process using approved, tailored version of organization wide standard and documented process.

2.8.5 QUANTITY CONTROLLED

Detailed measures of performance are collected and analyzed. This leads to a quantitative understanding of process capability and ability to predict performance.

2.8.6 CONTINUOUSLY IMPROVING

Quantitative process effectiveness and efficiency goals are established. Continuous process improvement against this goal is enabled by quantitative feedback.

2.9 TOTAL QUALITY MANAGEMENT (TQM)

TQM is a management philosophy, a paradigm, a continuous improvement approach for doing business through a new management model. The TQM philosophy evolved from the continuous improvement philosophy with a focus on *quality* as the main dimension of business.

TQM is made by the combination of three alphabetical letters. These letters are the following:

TOTAL- involving the entire organization, supply chain, or product life cycle.

QUALITY- the literal definition of quality

MANAGEMENT- the system of managing with steps like plan, organization, control etc.

2.9.1 CONCEPT OF TQM

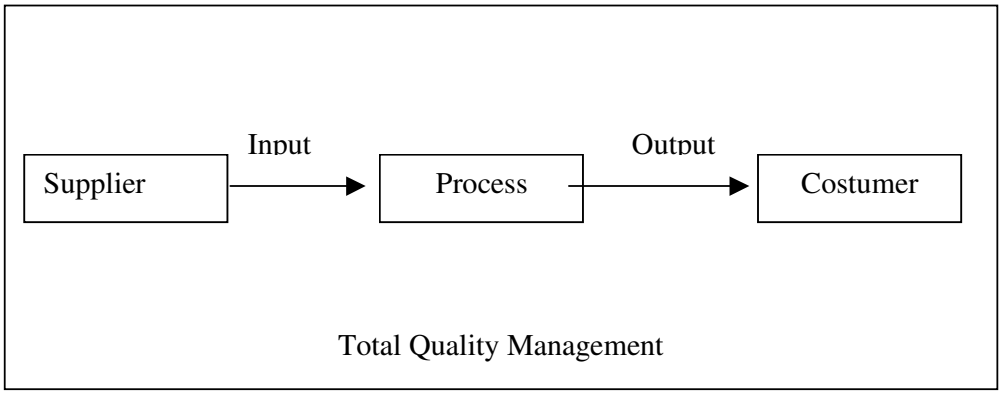


Fig 2.6 A process framework for Total Quality Management [29]

TQM is basically field of management. TQM manages the quality from a different perspective. These are following:

- Quality evolution
- Quality dimensions
- Quality management

TQM is not only work for software but it is work for hardware also. The goal of TQM is to create a system of management procedures that focuses on customer satisfaction and transforms the corporate culture so as to guarantee continual improvement. Simply stated, Total Quality Management is an interlocking arrangement of procedures and practices that ensure that all employees in every department are adequately trained and directed to continuously implement aligned improvements in quality, service, and total cost such that customer expectations are met or exceeded. TQM looks for a continuous improvement [24]. The main goal of the TQM is the total customer satisfaction. When a small project comes to the company. If the organization should not think because it is a short project and they shouldn't concentrate on this project. In this way, the project won't result in a good quality. Further, the customer can not be satisfied with the project. This is the issue of reputation of the company. Money doesn't matter in it. This customer will not come again in future to this organization. So the main goal of this methodology is to give the satisfaction to the customer. TQM expands beyond statistical process control to embrace a wider scope of the management activities of how we manage people and organization by focusing on the entire process not just simple measurement.

Now that we have discussed the difficulties of TQM implementation, and, since understanding is the first important step to success, let us examine the basic concepts of the Total Quality Management philosophy, what they mean and how they affect people in the organization. The four basic concepts which we will discuss are Total Customer Satisfaction, Teamwork, Employee Empowerment, and Continuous Improvement [46].

2.9.2 RELATION BETWEEN TQM AND CMM

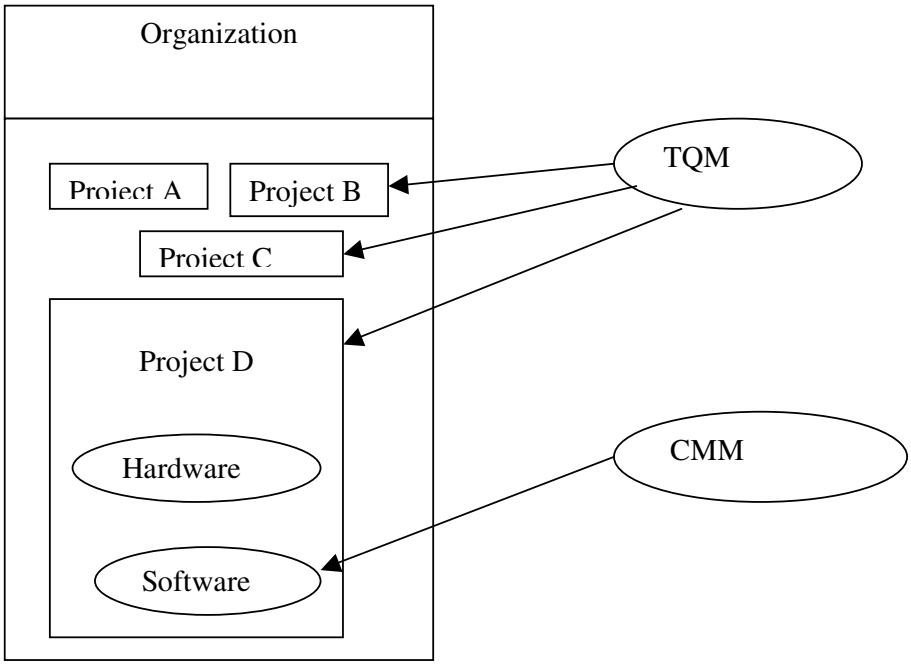


Fig 2.7: Relation between TQM and CMM [30]

2.10 TICKIT

Tick IT model is supported by the UK and Swedish software industries, has been to stimulate software system developers to think about:

- What quality really is in the context of the processes of software development,
- How quality may be achieved, and
- How quality management systems may be continuously improved.

Although certification of compliance to ISO 9001 is a contractual requirement for software suppliers in certain market areas, it should be a product of the more

fundamental aims of quality achievement and improvement, and the delivery of customer satisfaction.

With regard to certification itself, the objectives are to:

- Improve market confidence in third party quality management system certification through accredited certification bodies for the software sector,
- Improve professional practice amongst quality management system auditors in the software sector,
- Publish authoritative guidance material (the Tick IT Guide) for all stakeholders. [26]

The scope of the Tick IT Scheme, which has been approved by the United Kingdom Accreditation Service (UKAS) and the Swedish Board for Accreditation and Conformity Assessment (SWEDAC), is defined on this page.

These definitions are, with minor modifications, the result of joint activity between BSI Committee BRD/3/1 and the Association of British Certification Bodies (ABCB). The Independent International Organization for Certification (IIOC) has also participated through its representation on the Tick IT Industry Committee.

The scope definitions are used by accreditation authorities and certification bodies to determine whether a Tick IT auditor should be assigned to carry out or participate in an audit; however, only those Certification Bodies that have been accredited within the Tick IT procedures are allowed to issue Tick IT-endorsed ISO 9001 certificates [31], [32].

2.11 ISHIKAWA QUALITY CONTROL TOOLS

This is a very good methodology for the software process improvement. This is established by Karou Ishikawa. Kaoru Ishikawa, the man who pioneered the use of this chart in quality improvement in the 1960's. Kaoru Ishikawa integrated the 7 statistical tools in the 1960's. These are seven quality control tools. Their main purpose is to improve all kinds of processes, promoting the interrelation and team work of all people involved. Combination of Six Sigma and these quality control tools is an important key of success in the company for a long time. These tools are

basically based on statistical approach so it is very easy to understand and anybody can understand it. The main goal of the quality tools are:

- Increase the communication between operators and management.
- Detection of Problems and to decrease the recurrence of problems.
- Improve all kind of processes including manufacturing, educational, and business processes.

The tools are the following:

- Cause and Effect Diagram
- Run Chart
- Pareto Diagram
- Scatter Diagram
- Histogram
- Flow chart
- Control Charts

2.11.1 CAUSE AND EFFECT DIAGRAM

This is an Ishikawa tool. This is also known as fishbone diagram.

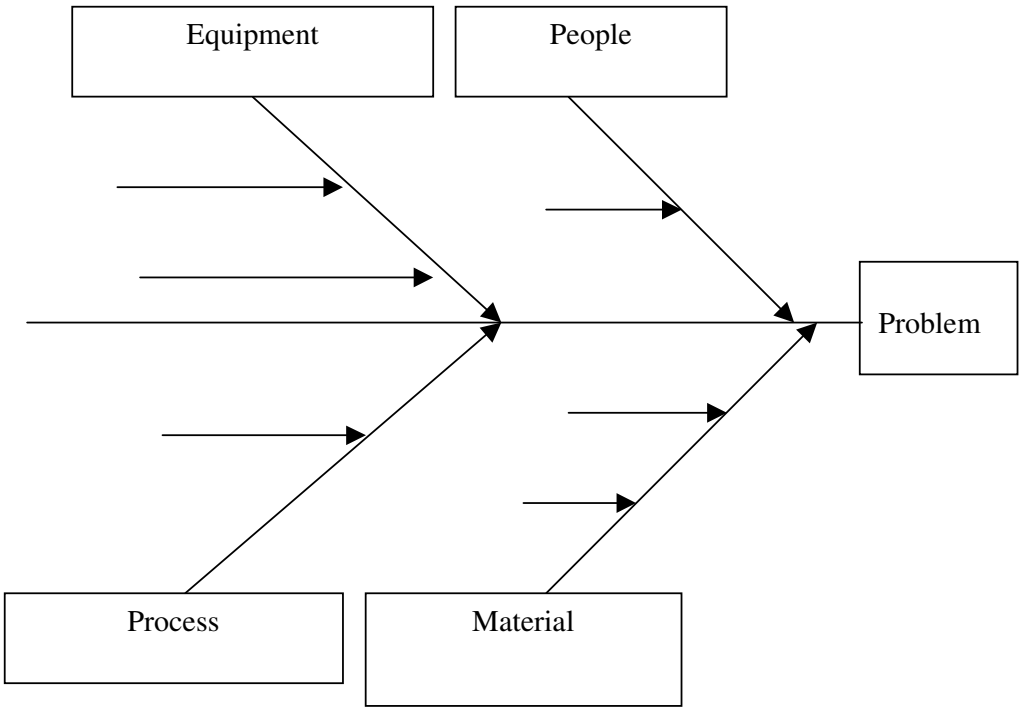


Fig 2.8 Fish bone diagram

The fishbone diagram is a tool for identifying the root causes of quality problems. This tool states a problem which is the cause of another problem and it also defines the cause.

This figure represents the causes in a typical diagram which are normally grouped into categories, the main ones of which are the following:

The 6 M's

Machine, Method, Materials, Maintenance, Man and Mother Nature (Environment) (recommended for the manufacturing industry).

The 8 P's

Price, Promotion, People, Processes, Place/Plant, Policies, Procedures, and Product (or Service) (recommended for the administration and service industries).

The 4 S's

Surroundings, Suppliers, Systems, Skills (recommended for the service industry) [33].

2.11.2 RUN CHART

This tool is dependent upon the measurement of the data. Variables are measured on the y axis corresponding to the time taken on x axis.

For each line in the run chart, the following statistics are calculated:

Table 2.2: Statistics mean in run chart [34]

1	**Mean**	The average of all the data points in the series.
2	**Maximum**	The maximum value in the series.
3	**Minimum**	The minimum value in the series.
4	**Sample Size**	The number of values in the series.
5	**Range**	The maximum value minus the minimum value.
6	**Standard Deviation**	Indicates how widely data is spread around the mean.

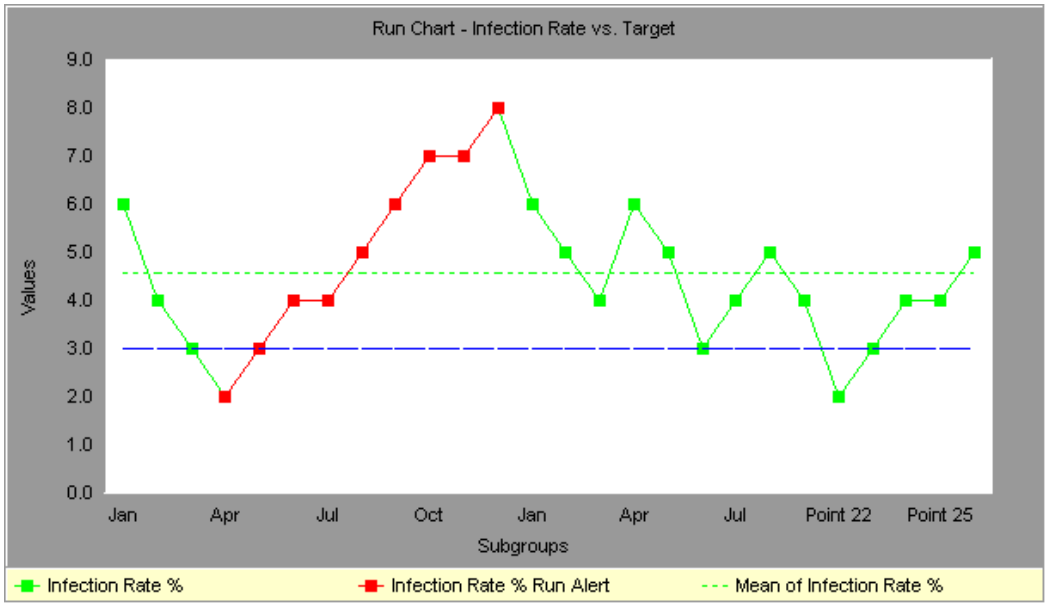


Fig 2.9: Run Chart [34]

2.11.3 PARETO DIAGRAM

A Pareto chart is used to graphically summarize and display the relative importance of the differences between groups of data. This bar graph depends on priority basis. First the priority of the events is decided. Then according to that, this graph is drawn. An example has been shown below

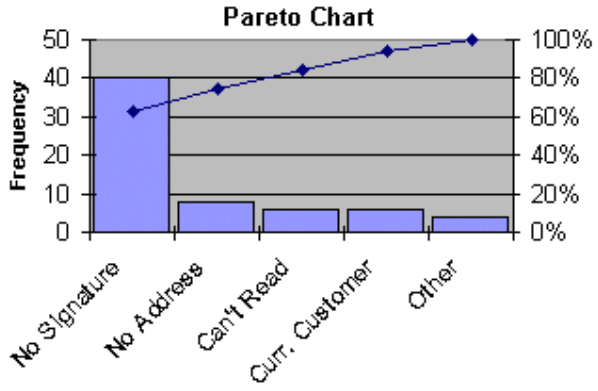


Fig 2.10: Pareto Chart [35]

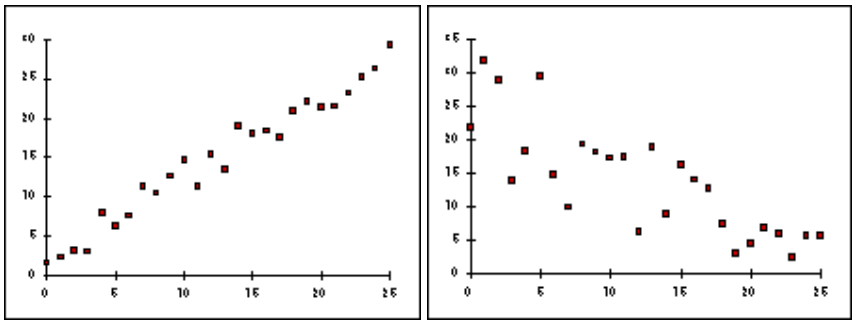
If your business was investigating the delay associated with processing credit card applications, you could group the data into the following categories:

- No signature
- Residential address not valid

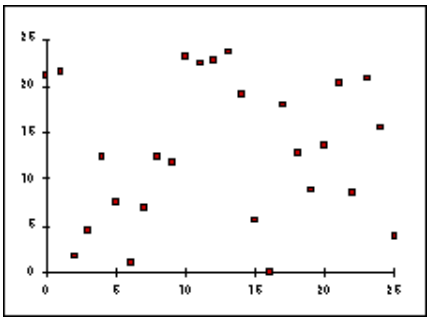
- Non-legible handwriting
- Already a customer
- Other [35]

2.11.4 SCATTER DIAGRAM

The Scatter Diagram is another Quality Tool that can be used to show the relationship between "paired data", and can provide more useful information about a production process [23]. Scatter diagram shows the association between the two variables. This shows how much a variable depends on another variable. The scatter diagram illustrates the strength of the correlation between the variables through the slope of a line.



Strong Correlation Moderate Correlation



No correlation

Fig 2.11: Scatter Diagram [36]

2.11.5 HISTOGRAM

Histogram is used to graphically summarize and display the distribution of a process data set [37]. The graphical representations depend on the frequency of the problem

within suitable class interval. Class intervals are shown on y axis and frequency is shown in x axis. The frequency is shown in the diagram as a rectangular box.

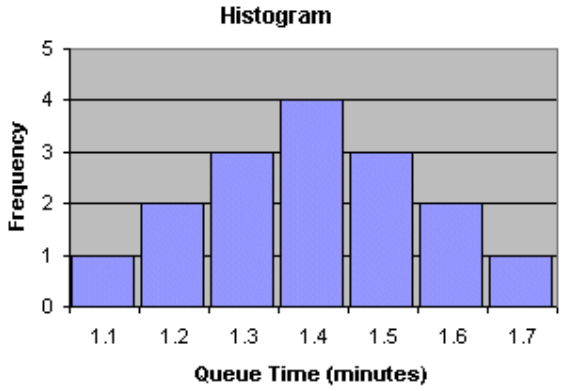


Fig 2.12: Histogram Diagram [38]

2.11.6 FLOW CHART

Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields. Flow chart is the sequential representation of activities. The four most common types of boxes in a flowchart are:

- A processing step, usually called *activity*, and denoted as a rectangular box
- A decision, usually denoted as a diamond
- Start point and ending point which is shown in circle [39]

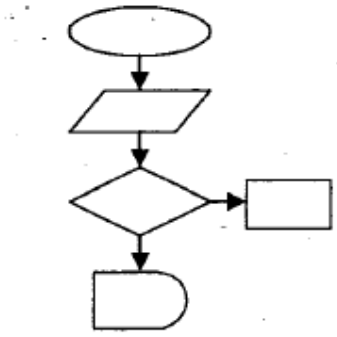


Fig 2.13: Flow chart [39]

2.11.7 CONTROL CHARTS

The control charts are used to distinguish graphically between the two sources of variability especially in manufacturing processes (common and special causes).

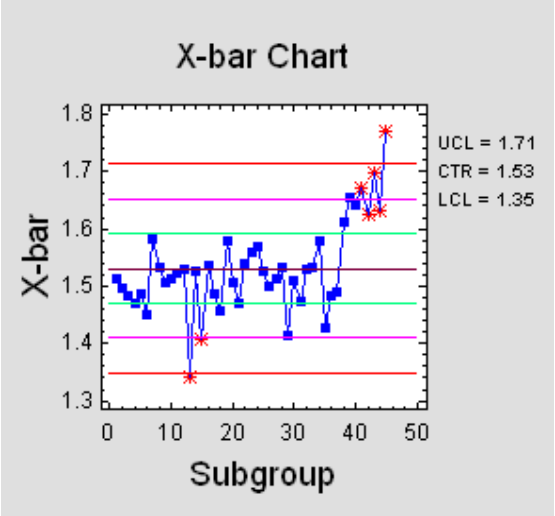


Fig 2.14: Control chart [40]

Two other horizontal lines, called the upper control limit (UCL) and the lower control limit (LCL) are also shown on the chart.

2.12 SIX SIGMA

Six sigma strategies were developed by Motorola in the early 1990s [16]. Six Sigma is based on statistical approach which does the improvement by historical data and by calculation of mathematical formulas. Six Sigma is an effective and systematic quality improvement approach to enhance the organization's performance based on the adoption of various statistical analytic techniques [14]. The goal of the six sigma is to detect the defect and reduce the defect. The word sigma is a Latin word which is used to measure how far something deviates from perfection. Six sigma means a company tries to make "error-free product 99.9997% of the time a minuscule 3.4 errors per million opportunities" [15]. SIX SIGMA has six sigmas and reduces the defect step by step. Six Sigma is usually related to the magic number of 3.4 defects per million opportunities. People often view Six Sigma as yet another rigorous statistical quality control mechanism. Today Six Sigma is delivering business excellence, higher customer satisfaction, and superior profits by dramatically improving every process in an enterprise from financial to operational to production. Six Sigma has become a darling of a wide spectrum of industries, from health care to insurance to telecommunications to software.

Table 2.1: Level of six sigma [16]

Sigma Level	Defects per Million	Defects as Percent
One sigma	690,000.0	69.0000%
Two sigma	308,000.0	30.8000%
Three sigma	66,800.0	6.6800%
Four sigma	6,210.0	0.6210%
Five sigma	230.0	0.0230%
Six sigma	3.4	0.0003%

2.12.1 ROLES IN ORGANISATION OF SIX SIGMA

In the organization a proper team is available. The members of the team are senior executive (CEO or president), executive committee, champion, process owner, master black belt, black belt, and green belt.

2.12.1.1 SENIOR EXECUTIVE (CEO OR PRESIDENT)

The senior executive provides the force, direction, and position necessary for Six Sigma's success. The most successful, highly shown Six Sigma efforts have all had committed leadership from top management.

2.12.1.2 MEMBERS OF THE EXECUTIVE TEAM

The members of the executive team are the top management of an organization. They Operate at the same level of commitment for Six Sigma management as the senior executive.

2.12.1.3 CHAMPIONS

Champions play an important role in six sigma procedure. Champions take a very active sponsorship and leadership role in conducting and implementing Six Sigma projects. They work directly with the executive team.

2.12.1.4 MASTER BLACK BELT

A master black belt takes on a leadership role as warden of the Six Sigma process. They are the advisor of senior executives or business unit managers. He or she must leverage his or her skills with projects that are led by black belts and green belts. Frequently, master black belts report directly to senior executives or business unit managers.

2.12.1.5 BLACK BELT

A black belt is a full-time change agent and improvement leader who may not be an expert in the process under study. A black belt is a quality professional who is mentored by a master black belt, but who may report to a manager for his or her tour of duty as a black belt.

2.12.1.6 GREEN BELT

A green belt is an individual who works on projects part-time (25%), either as a team member for complex projects or as a project leader for simpler projects. Most managers in a mature Six Sigma organization are green belts.

Six Sigma use two approaches for improvement

DMAIC (Define, Measure, Analysis, Improve, Control)

DMADV (Define, Measure, Analysis, design, Control)

2.12.2 DMAIC MODEL

The DMAIC is the Six Sigma model for improving an existing product, service or process. It is an alternative to the PDSA model. It has five phases:

- Define
- Measure
- Analyze
- Improve
- Control

2.12.2.1 DEFINE

This phase defines the processes regarding business and requirements. The define phase involves understanding the relationships between Suppliers-Inputs-System-Outputs-Customers (called SIPOC analysis), and analyzing Voice of the Customer data to identify the critical to quality (Cuts) characteristics important to customers, and developing a project objective.

2.12.2.2 MEASURE

The measure phase includes operational definitions for each Critical-To-Quality (CTQ) variable, determining the validity of the measurement system for each CTQ, and establishing baseline capabilities for each CTQ.

2.12.2.3 ANALYZE PHASE:

The analyze phase identifies the upstream variables (*X*s) for each CTQ using a flowchart. Upstream variables are the factors that affect the performance of a CTQ. To restate this quantitatively:

CTQ = f (X_1, X_2, X_3, K_{xoe}), where

CTQ = the critical-to-quality characteristic important to customers identified in the define phase and clarified in the measure phase of the DMAIC model.

X_i = upstream i^{tch} variable that is hypothesized to have an impact of the performance of the CTQ.

2.12.2.4 IMPROVE PHASE:

The improve phase includes the design experiment to understand the relationship between the CTQs and Xs. Developing the plan to formalize the level of Xs and optimize the CTQs is also part of this phase.

2.12.2.5 CONTROL PHASE:

The control phase includes procedures to keep away from potential problems with the *X*s with risk management and mistake proofing, standardizing successful system revisions, controlling the critical *X*s, documenting each control plan, and turning the revised system over to the system owner.

2.13 TRILLIUM

In April 1991 first version of the TRILLIUM model was elaborated by Bell Canada, Northern Telecom and Research Bell Northern [19].

Trillium has a telecommunications direction

- provides a customer focus
- provides a product perspective
- includes technological maturity
- includes additional Trillium-specific practices
- Provides a roadmap approach which sequences improvements by maturity [20].

The Trillium Model developed from a customer perspective, as perceived in an aggressive, business environment. In this context, trillium has capability levels whose main motive as follow

- Requirements meet the customer expectations
- project has no defect or minimal defects
- with in minimal cost
- In the shortest time.

A telecommunications product typically includes hardware, software, documentation, training and support services.

The Trillium model is based on the Carnegie Mellon University Software Engineering Institute's (SEI) Capability Maturity Model (CMM) v1.1 initially developed by W.

2.13.1 THE TRILLIUM SCALE

The Trillium scale has levels 1 through 5. The levels can be characterized in the following way

2.13.1.1 UNSTRUCTURED

The development process in this level is adhoc basis. Projects cannot meet quality or schedule targets. Success, while possible, is based on individuals rather than on organizational infrastructure. In this level risk is very high.

2.13.1.2 REPEATABLE AND PROJECT ORIENTED

In this level, project success is achieved a strong project management planning and control, requirements management, estimation techniques, and configuration management plays an important role. In this level risk is medium.

2.13.1.3 DEFINED AND PROCESS ORIENTED

In this level, Processes are defined and well documented utilized at the organizational level. Processes are controlled and improved in adequate way. ISO 9001 requirements are used to training and internal process auditing. In this level risk is low.

2.13.1.3.1 MANAGED AND INTEGRATED

For process improvement, Process instrumentation and analysis is used as a key mechanism. Process change management and defect prevention programs are main activities. In this level risk is lower.

2.13.1.3.2 Fully Integrated

In this level, formal methodologies are used. Organization repositories are creating for the future point of view. In this level the risk is very lowest.

2.13.2 CAPABILITY LEVEL

- Organizational Process Quality
- Human Resource Development and Management
- Process
- Management
- Quality
- System Development Practices
- Development Environment
- Customer Support [18]

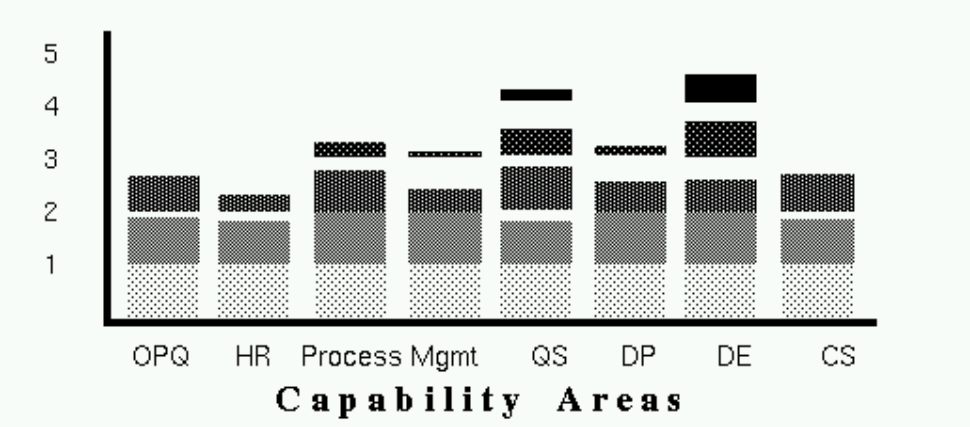


Fig 2.4: Capability area of Trillium Model

2.13.4 ARCHITECTURE OF THE TRILLIUM MODEL

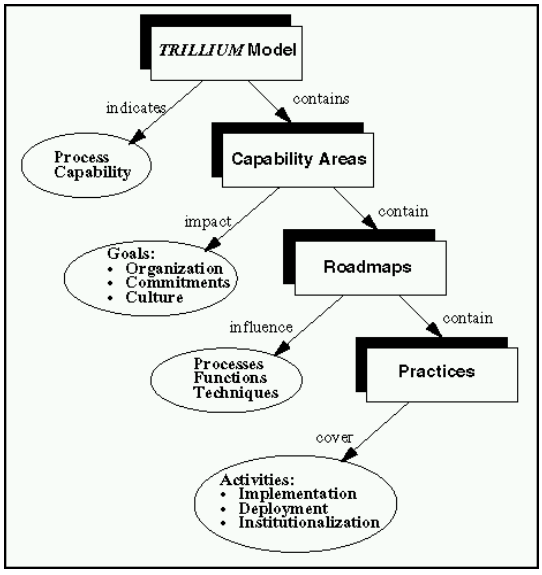


Fig 2.5: Architecture of the Trillium Model

2.14 PROCESS IMPROVEMENT METHOD

A software process improvement method is defined as an integrated collection of procedures, tools, and training for the purpose of increasing product quality or development team productivity, or reducing development time. This model is a small step for improving the processes. A lot of models exist. The purposes of the methods are:

- Fewer product defects found by customers.
- Earlier identification and correction of defects.

- Fewer defects introduced during the development process.
- Faster time to market.
- Better predictability of project schedules and resources.
    - Estimation
    - ISO 9000 certification
    - Software process assessment (SPA)
    - Process definition
    - Formal inspection
    - Software measurement and metrics
    - Computer aided software engineering (CASE)
    - Interdisciplinary group methods (IGMs)
    - Software reliability engineering (SRE)
    - Quality function deployment (QFD)
    - Total quality management (TQM)
    - Defect prevention process (DPP)
    - Clean room software development
    - Reengineering

2.14.1 ESTIMATION

Before starting the project it is important to know about size of the project and staff. A lot of methods are available like FUNCTION POINT COUNT, COCOMO model, to know about the size. Experts must be there. The reason is that if the exact size of the project at the time of dealing is not estimated then the project may suffer serious drawbacks. So this is the key of the project. This project can not get a good destiny. The project cannot be abandoned. This is the question on the reputation of the organization. In the capability maturity model estimation comes in the level 2 under the key process area of software project planning.

2.14.2 ISO 9000 CERTIFICATION

ISO standard is a quality management system established by international standards. ISO 9000 is a part of ISO series. ISO 9000 gives the guidelines for the quality management and software quality assurance. This method is categorised in the CMM level 2 under the key process area of software quality assurance. It also minimally

refers to some other KPAs such as software subcontract management, software project planning, software project tracking and oversight, peer reviews, software product engineering, configuration management, organization process definition, and training. The ISO 9000 certification audit process encourages the development of software standards and procedures documents that correspond to the ISO 9000-3 requirements. What ISO 9000 certification does not directly address is the question of whether or not the documented practices are effective.

2.14.3 SOFTWARE PROCESS ASSESSMENT (SPA)

Assessment is the technique to recognize the strength and weakness of the organization. This assessment is conducted by a group called software process engineering group (SEPG) which consists of highly qualified people. They prepare a maturity questionnaire. On the basis of the questionnaire they take an interview of all the members of the organization. Within the ideal environment, members of the software engineering process group (SEPG) would participate as members of the assessment team and then assist in implementing the improvement actions. Software process assessment can correlate with CMM level 3 under the key process areas which are organization process definition and organization process focus. Assessment should be done time to time with in two years.

2.14.4 PROCESS DEFINITION

Firstly a software development life cycle is suggested according to the customer's requirement. A lot of software process model exist. If the project is to be delivered as soon as possible then the RAD model should be used and if there are no time limits on delivering the project, then Spiral model is more suited. It is necessary to have a high-level understanding of a product development process in order to do the planning for a software project. This method is categorising in the CMM level 2 under the software project planning.

2.14.5 FORMAL INSPECTION

This is an important method for process improvement. Formal inspection refers to a structured process of trying to find defects in development documents such as programming code, specifications, designs and others during various phases of the

software development process. It is named after Michael Fagan who is credited with being the inventor of formal software inspection [41]. There is a team who performs the inspection. Every one has own role in the team.

- Author/Designer/Coder: the person who writes the low-level document.
- Reader: paraphrases the document.
- Tester: reviews the document from a testing standpoint.
- Moderator: responsible for the inspection session, functions as a coach.

Formal inspection is a sequential activity. First the planning is done and then a team is arranged. Then it is decided where the inspection should be done. After that, the training is imparted to the team members for the material according to their role. Then the participants are prepared for their roles. Then the formal inspection is done. Then defects are found and documented. Formal review should come in the CMM level 3 under the key process area peer review.

2.14.6 SOFTWARE MEASUREMENT AND METRICS

The things which can be seen and touch .We can measure the things what we have seen which we can touch. But we can not touch the software. But there are many metrics by which we can measure of software quality attributes. Software measurement enables us to estimate cost and effort devoted to project. Software measurement enables us to determine quality of software. Software measurement enables us to predict maintainability of software [42]. A lot of metrics are existing for example LOC. By LOC we can measure the size of the project. Another metrics are available for the measurement of size is COCOMO model and function point technique. This method is come in CMM level 4 under the category of key process area quantitative process management.

2.14.7 COMPUTER AIDED SOFTWARE ENGINEERING (CASE)

Computer-Aided Software Engineering (CASE) a set of tools and methods to software which results in high-quality, defect-free, and maintainable software products. The term "Computer-aided software engineering" (CASE) can refer to the software used for the automated development of systems software, i.e., computer code [43]. CASE is used to succeed in software organizations with mature, well-defined

processes. This method can be categorising in CMM level 3 of key process area software product engineering.

2.14.8 INTERDISCIPLINARY GROUP METHODS (IGMS)

This method is come under the CMM level 3 under the category of inter group coordination. This method is described about the communication between the members.

2.14.9 SOFTWARE RELIABILITY ENGINEERING (SRE)

Software reliability is a quality attributes which means is taken by probability of failure-free software operation for a specified period of time in a specified environment. This method is ensuring how much your system is reliable or failure free. There are available some mathematical formulas by which we can calculate the reliability. This method can be categorising under the level 4 of key process area quantative process management.

2.14.10 QUALITY FUNCTION DEPLOYMENT (QFD)

This method is describing the communication between the customer and company. This method sees that the entire requirement is complete before deliver the project to the customer. Quality Function Deployment (QFD) was developed to bring this personal interface to modern manufacturing and business. All the management work is come under this method customer requirement and see that they are fully meet with the project or not. Customer satisfaction is the main purpose of this method. QFD is a comprehensive quality system that systematically links the needs of the customer with various business functions and organizational processes, such as marketing, design, quality, production, manufacturing, sales, etc., aligning the entire company toward achieving a common goal [44]. This method can be categorised CMM's level of 4 under the quality process management. This method can be taken under the key process area software project requirement of CMM level 2. Because requirement also manage by this method. The main aim of this method is provide high quality software to the customer.

2.14.11 TOTAL QUALITY MANAGEMENT (TQM)

This method is improving the whole organization. This method can be taken as CMM level 4 under the qualitative process management. This method is also for suitable in level 3 under the key process area organization process focus. According to the name total this method manages the whole. Whole means quality, planning, man power and all. This method is applicable not only software hardware also. The TQM method is best applied within organizations motivated to improve their quality culture.

2.14.12 DEFECT PREVENTION PROCESS (DPP)

This method is related with the defect detection and then removes it. It is a combination of software inspection and defect prevention. Try to find defect though inspection or much method then this method removes the defect. This method categorized under the CMM level 5 under the key process area defect prevention.

2.14.13 CLEANROOM SOFTWARE DEVELOPMENT

This method is based on the statistical approach. This is a sequential approach to find the defect and then remove this. The Clean room Software Engineering process is a software development process intended to produce software with a certifiable level of reliability.

Chapter 3

PROBLEM FORMULATION

The software process improvement is an open question. This term is used for the improvement in process. During the review process it is observed that a lot of models are existing for example CMM, SIX SIGMA, IDEAL Model extra. These models have some pros and cons. These models have some limitation. The main problem in this thesis is to be proposed a software process improvement model which should be applicable for all the organization both small and large firm. So the first problem is exploring the limitations of the models. Such that SPIM Model improves these limitations.

The limitations of existing models are as follows:

3.1 LIMITATIONS OF SOFTWARE PROCESS IMPROVEMENT MODELS

On the study of the literature review some limitations of models are explore.

3.1.1 LIMITATIONS OF CMM

- The main limitation of CMM is key practice describes "what to do" but does not prescribe "how to do".
- CMM is works only a repeating task.
- CMM is a goal. Being used just as stamp of approval.

3.1.2 LIMITATIONS OF IDEAL MODEL

Ideal model is also a continuous model. But it is a full method such that there is no recovery. Means either it is success or fail.

3.1.3 LIMITATIONS OF SIX SIGMA

- Six Sigma focuses on prioritizing and solving specific problems which are selected based on the strategic priorities of the company and the problems which are causing the most defects.
- Six Sigma is a statistically-based process improvement methodology.
- Another downfall of Six Sigma is the idea behind what constitutes a defect. If a defect or mistake is not well defined, it is hard to determine if one has

occurred or not. Although many companies have had success with Six Sigma, some companies have not been so fortunate. It usually takes four to five years to see any benefits of Six Sigma once an initial commitment has been made.

- Often it is very difficult for small companies to take employees away from their regular duties in order to be trained in Six Sigma. If employees are not available to give their services, the company loses money due to a reduction in productivity.

The objective of SPIM Model is improving the quality of software product ability to work with in time and budget constrains. Remove the limitation of previous models like that in CMM Model, CMM's key practice describes "what to do" but does not prescribe "how to do". SPIM Models is a complete methodology to describe how the process improves. CMM works for the repeatable task. SPIM Model is flexible if any time between the process improvement problems creates. Because of cyclic model it can be removing at present time. In IDEAL Model, it is a full method such that there is no recovery. Means either it is success or fail. SPIM model is a cyclic model. So if any problem face then it will work until improvement has not completed. Six Sigma is a statistically-based process improvement methodology. SPIM Model does not depend on the statistical data.

The main objectives of this thesis are:

- To improve the quality of software product ability to work with in time and budget constrains.
- To study the software process improvement models
- Compare the models
- Explore the possibility of improvement in the models
- To find the limitation of existing models
- To explain the principles of software process improvement
- Develop and design software process improvement model

Chapter 4
COMPARATIVE ANALYSIS

In this chapter effort has been to compare various models including CMM, CMMI, ISO, and SIX SIGMA, IDEAL.

4.1 CMM vs. SIX SIGMA

Table 4.1: Difference between CMM and SIX SIGMA

S.NO.	CMM	SIX SIGMA
1	CMM is a goal	Six Sigma is a methodology for process improvement.
2	CMM is a framework for assessing how "mature" a software organization is	Six Sigma tries to improve profitability by reducing defects and connecting more closely with your customer.
3	Define basic process infrastructure	Does not include any process mode
4	Organization level in level 4 and level 5	Data driven/ measurement driven
5	Lack of standard metrics	Customer –centric

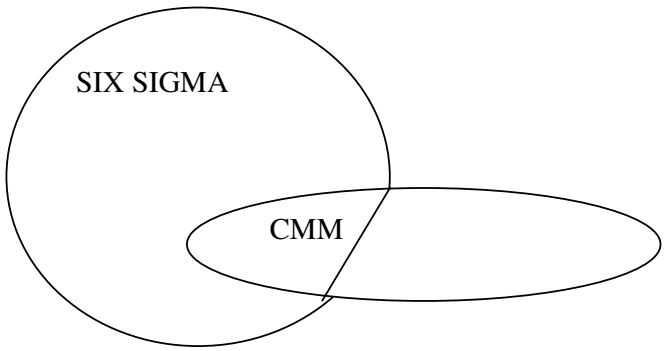


Fig 4.1: Relation between CMM and SIX SIGMA

4.2 CMM vs. CMMI vs. ISO

Table 4.2: Difference between CMM, ISO, CMMI

	CMM	**ISO**	**CMMI**
1	CMM describes about the software engineering alone	ISO describes both software and system engineering	CMM Integrated (CMMI) describes both software and system engineering. CMMI also incorporates the Integrated Procees and Product Development and the supplier sourcing.
2	CMM more about software development.	ISO works for software and hardware both.	CMMI Software development and System engineering. Means CMMI also suits the mfg industry also
3	CMM is process improvement approach. Which have 5 maturity level	ISO (International Organization for Standardization) is the world's largest developer of standards.	Capability Maturity Model Integration (CMMI) is a process improvement approach that provides organizations with the essential elements of effective processes.
4	The CMM is a way to communicate capabilities.	The ISO is a way to communicate the process.	The CMMI does not only communicate with quality but also organization.

4.3 ISO vs. SIX SIGMA

Table 4.3: Difference between ISO and SIX SIGMA

	ISO	**SIX SIGMA**
1	ISO 9001 is a Quality Management System, which includes specialized quality management standards for specific industries.	Six Sigma is not a quality management system, such as ISO-9001, or a quality certification system
2	ISO 9001, with guidelines for problem solving and decision making, requires a continuous improvement process in place but does not indicate what the process should look like	Six Sigma is a statistically-based process improvement methodology that aims to reduce defects to a rate of 3.4 defects per million defect opportunities by identifying and eliminating causes of variation in business processes.
3	ISO standard provide a template for evaluating an organization's overall quality management efforts whereas ISO 9001 does.	Six Sigma does not provide a template for evaluating an organization's overall quality management efforts

4.4 TQM vs. SIX SIGMA

	TQM	SIX SIGMA
1	Total Quality Management (TQM) is a structured system for satisfying internal and external customers and suppliers by integrating the business environment, continuous improvement, and breakthroughs with development, improvement, and maintenance cycles while changing organizational culture.	Six Sigma is a statistically-based process improvement methodology that aims to reduce defects to a rate of 3.4 defects per million defect opportunities by identifying and eliminating causes of variation in business processes.
2	TQM employs a more broad based application of quality measures to all of the company's business processes.	Six Sigma focuses on prioritizing and solving specific problems which are selected based on the strategic priorities of the company and the problems which are causing the most defects
3	TQM tends to apply quality initiatives within specific departments	Six Sigma is cross functional meaning that it penetrates every department which is involved in a particular business process that is subject to a Six Sigma project.
4	TQM provides less methodology in terms of the deployment process	Six Sigma's DMAIC framework provides a stronger platform for deployment and execution.

Table 4.4: Difference between TQM and SIX SIGMA

Chapter 5

PROPOSED SOFTWARE PROCESS IMPROVEMENT MODEL (SPIM)

5.1 PROBLEMS IN SOFTWARE PROJECTS

There are mainly three problems to be solved in the software project.

- Projects have no good quality
- Project does not deliver with in time
- Project goes over budget

Every problem has some causes, which are presented by the fishbone diagram. For example a project that does not get a good quality. Beyond these problems some causes are there. Management is one the cause of this problem but this is also a problem again beyond this problem there are a lot of reasons not only a good management. There may not be good communication between the manager and employee. Project manager are not sincere and he could not understand the problem and handle the problems. He could not present the problem in adequate way to the senior management. Senior management don't hear the problems of employees. Another cause of the problem of bad quality is misunderstanding. If organisation cannot do a good calculation of the project size at the time of deal with the customer. Then problem creates because requirement can not change after the completion of the deal. Organization wants that they should complete the project under budget. At this situation it is not possible that project gets the good quality. Another cause of the bad quality problem is manpower. If organizations do not find suitable employees in the project then project will be either late or not a good quality. Employees should be well trained. They should be aware to new technologies. If they don't know the new technologies or don't aware to the technologies. Then they can't perform well in their module.

Second problem is over budget. Project sometime goes to over budget. The main cause this problem is misunderstanding between the customer and organisation. At

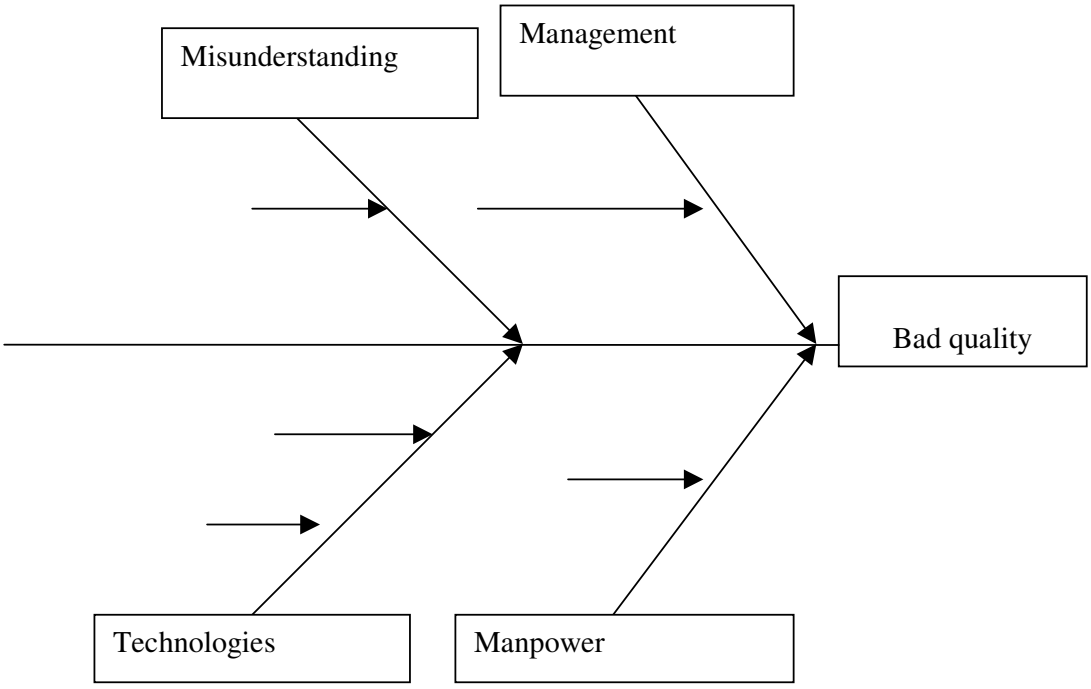


Fig 5.1: Fishbone Diagram, when problem is bad quality

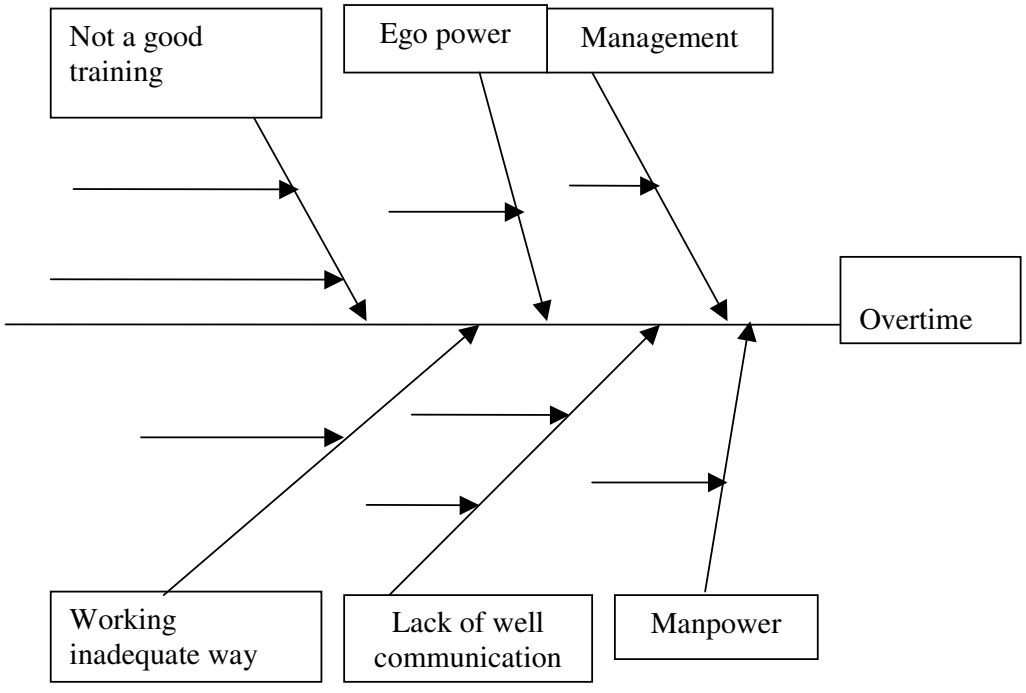


Fig 5.2: Fishbone Diagram, when problem is over time

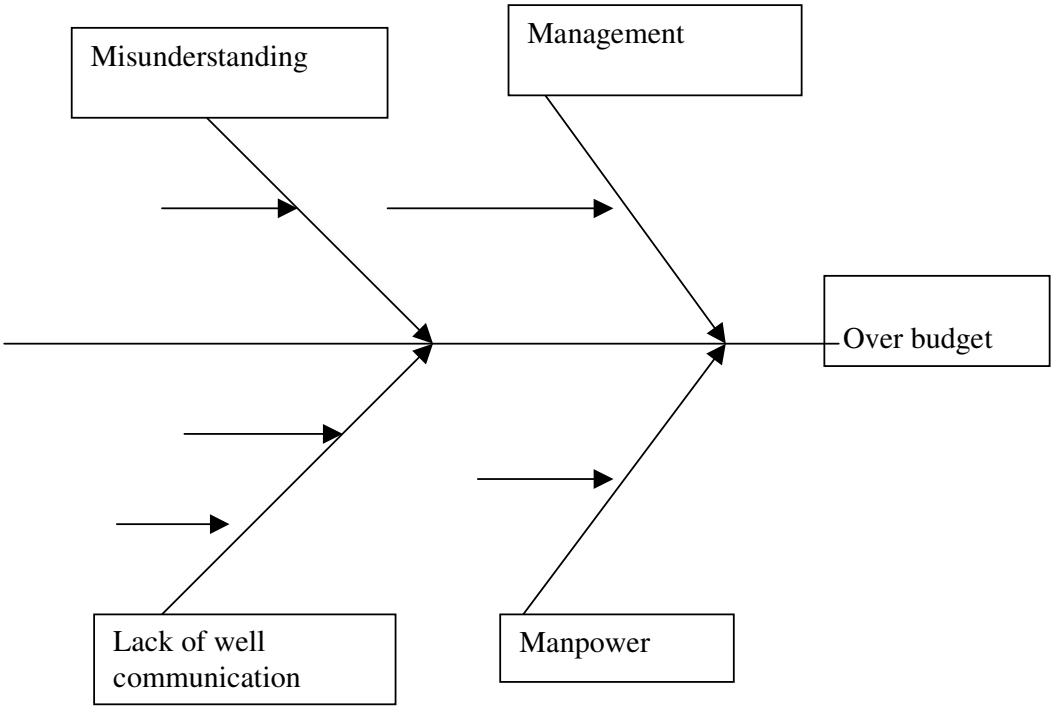


Fig 5.3: Fishbone diagram, when problem is over budget

that time they have done mistake to calculate the project. Than at this situation project goes to over budget. Third problem is over time. The project does not complete with in time which is deadline given by the customer. The main problem is not provided a good training to the employee. They don't have a good knowledge about the subject what they are going to do. So they can't perform well. And the projects cross the deadline.

Table 5.1: Decision table

CAUSE	1	2	3
Lack of management	Y	Y	Y
Misunderstanding	Y		Y
Not provide good training		Y	
Technologies	Y		
Not well Communication			Y
Inadequate work		Y	
Lack of well communication		Y	Y
Manpower	Y	Y	Y
Ego power		Y	
EFFECT			
Bad quality	Y		
Over budget			Y
Over time		Y	

5.2 PROPOSED SOFTWARE PROCESS IMPROVEMENT MODEL (SPIM)

Software process improvement model is an approach or method or both by which process improves and give better result rather than a normal process. By software process improvement a better and high quality product can be found with in budget and time.

The researcher suggests a Software process improvement model. This model is an iterative and cyclic model.

This model consist eight steps. These are following

a) Process Define
b) Problem Define
c) Recommendation and Assessment
d) Planning
e) Implement
f) Inspection
g) Configuration
h) Documentation

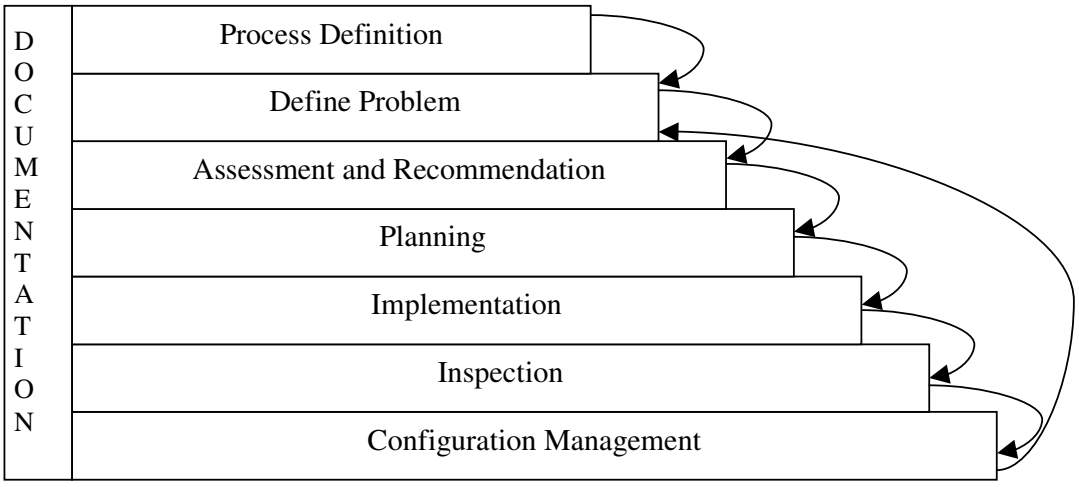


Fig 5.4: Software Process Improvement Model (SPIM)

5.2.1 PROCESS DEFINATION AND PROBLEM DEFINITION

A doctor can give the medicine when he knows about the patient diseases. Without knowing they can not do anything. Similarly any process improvement model can

work accurately if we know the exact problems in process. Therefore the main issue is to make problem clear first of all. When problem become clear then a solution can be found for the problem. What processes have taken for the development of the project? Then it can find the shortcoming in the process why the process is not giving a better quality or why project should not complete within time has given by the customer. So the first step is to understand the process. A lot of development process model are exist. For example if customer want the project to be done as soon as possible. At that time RAD model should be apply. Developer can't apply any other model. A lot of models are available for software process. Waterfall model, spiral model, iterative model extra. Study the process and try to find the shortcoming. A lot of problem may become the reason of unsuccessful project.

- Lack of management
- Lack of well communication
- Lack of well communication between customer and organization
- Technical problem and lack of resources
- Lack of man power
- Not provided a good training
- Ego power
- Working in a adequate way

5.2.1.1 LACK OF MANAGEMENT

This is a good point of view for unsuccessful project. If there are no good management in the organization. This is harmful for the project. Project manager is not well qualified or he can not hold the team and cannot inspire the team. Senior management does not hear the problem or ignore them.

5.2.1.2 LACK OF WELL KNOWN COMMUNICATION

Well known means suppose a man which is perfect in one module of a project. But management gives him other responsibility in the project then this person cannot perform well as well as he can do in his module. If every member in the team know its work very well with interest. Then output may be come in a better way and better quality.

5.2.1.3 WELL COMMUNICATION BETWEEN CUSTOMER AND ORGANIZATION

Well communication means when organization deals with the customer. They must specify the entire requirement according to the customer. Calculate every thing according to the size, budget, and time. If organization fails to right to calculate with the deal in customer. Then project can't get a good quality. Because organizations don't want that they go over budget. So they want to complete project according the customer's budget. So they can't give a good quality in the project.

5.2.1.4 LACK OF MAN POWER

Lack of man power or good manpower is also a reason for unsuccessful project. For example a project needs 100 men but due to some reason you have only 50 men. Reason may be anything like you has no such money for project, or you can not arrange so much manpower but you have taken a big project. At this condition project can not complete on time. One more reason organization have no good labour that is if one person works on a module some day but he leave after some day at this situation also a better output has not come with in time.

5.2.1.5 NOT A GOOD TRAINING

Organization should provide a good training to member of team. Without good training they can't perform so well. They can't get higher level of maturity with out training. In the training organization should give the good knowledge and tell them what the organization want.

5.2.1.6 EGO PROBLEM

A well communication have establish in member of companies. If member have ego power in himself and not cooperate with other then project will not go in a right way.

Table 5.1: Categories the problem

	Cause of problems	**Category out of 5 scale**
1	Lack of management	4
2	Lack of well communication	3
3	Lack of well communication between customer and organization	3
4	Technical problem and lack of resources	4
5	Lack of man power	3
6	Not provided a good training	3
7	Ego problem	2
8	Working inadequate way	2

5.2.2 ASSESSMENT AND RECOMMENDATION

Once the problem is defined, one can easily find about area need to be improving and then according to problem an assessment team formed with recommendation of higher authorities. Team depends upon the management and size of problem. Team, which is so formed, should make have highly qualified and experience holder members. If the size of the problem is small, a team of 2-3 researchers are good enough. If the improvement area is large enough, then team will be formed according to the size of problem. Sometime software improvement is not so much effective because no one can say or give promise that applied technique or method can improve the culture of project. So if area of improvement is small like in small project, then risk is reduced. But if area of improvement is of higher level then it may be a great risk. A project/organisation take time for improvement and have no guarantee for success, but only probability for the success of same. So at this condition it is better to take sponsorship by the third party. But it requires higher authority's permission. They can take decision about timing of sponsorship needed. They have a meeting with senior management about the improvement of process and assessment of team like who should be the member of team for improvement process. This is very important step because team should be consist of highly qualified member and as small as possible with their role defined and known to them. Member of the team must be certified by SEI (software engineering institute) as only then they can take better decision in critical situation.

5.2.3 PLANNING

After formation of assessment team, they start work on the project. They make plan and prepare strategy about how can they improve process by finding changes required. For this first of the entire problem clear and defined. Team now make a questionnaire for the members coming from different level in organization. Analysis the current situation of the company and then makes a profile for the future of the company. For that, they conduct interview of as much person as possible and make a plan. If there is more than one strategy for any process improvement, then they choose one which produces the better result. With the approach defined, a detailed implementation plan can be developed by taking consideration of all the factors like, technical and non technical. When the team complete the plan, they have a meeting with the senior management and there they give a presentation to the seniors. By doing this they reduce risk level. At this situation assessment team is not responsible for any casualty. If the senior management comfortable with your plan, then only team would be allow to move towards next step of implementation plan. If they don't satisfied with plan, then alternative plan have to make with the presentation. This plan includes schedule, tasks, milestones, decision points, resources, responsibilities, measurement, tracking mechanisms, risks and mitigation strategies, and any other elements required by the organization.

5.2.4 IMPLEMENTATION

When the plan establish according to problem, then create the solution according to plan. First make a paper solution on basis of experience and knowledge as well as skill of the assessment team. Then again gives presentation to the higher authorities. If they give some suggestion for modification in the solution, then modify the plan accordingly and if they are comfortable for solution then start the implementation. They choose a better solution. Before implementing the plan it should be discuss with the senior management and organization. When they are agree on solution, start working on the solution. All the solution does not apply in one step; it is a step by step process. Makes a baseline after every step and take a pilot test and check the result if the result is positive then next continue the solution. If there are more then one solution. In this situation don't take decision by him. Again discuss with the higher authorities. Which solution seems better then apply the solution. By doing this

risk is less because at this situation team is not responsible and if the plan be successful the all the credit goes to the assessment team.

5.2.5 INSPECTION AND CONFIGURATION MANAGEMENT

An inspection is generally an organized examination or a formal evaluation exercise. It involves the measurements, tests, and gauges applied to certain characteristics in regard to an object or activity. The results are usually compared to specified requirements and standards for determining whether the item or activity is in line with these targets. Inspections are usually non-destructive. It is not guarantee that the work plan and implementation is correct. For inspection a team is formed, in which members are moderator, reader, and inspectors. A formal meeting is conducted. Role of the moderator is to conduct the meeting and ensure the subject of the meeting. Role of the moderator is like a leader of the inspection team. So moderator should be highly qualified and skilful. Moderator tells the team how the inspection starts and leads them. Inspection team takes the interview of assessment team. They make a questionnaire for every member of team. Review of the work done by all and they analyse the result. What result will come after process improvement? Review all the factors came in the technique. Main motive of the inspection is to find the defect in the module. If there are no defects then team will give the clean chit for the process improvement. If there is any problem in the technique then they tell where the problem is or what problem can be faced. What change should be done in process? They specify where the changes are need. This work is handled by the configuration management. Before a configuration item makes a baseline of the work. A baseline is a software configuration management concept that helps us to control changes with out seriously impeding justifiable changes. Once a baseline is established, we pass though a one way door. After that changes inspection team again inspect the process. This process will continue till the defect does not end. And the improvement has done.

5.2.6 DOCUMENTATION

Documentation is an umbrella activity in this model, which is going on side by side through out process. In each phase documentation is needed. Documentation is needed for the future work so that anybody can reuse it as well as future work can be

done. The documents associated with a software project and the system being developed has a number of associated requirements:

They should act as a communication medium between members of the development team.

They should be a system information repository to be used by maintenance engineers.

They should provide information for management to help them plan, budget and schedule the software development process.

Some of the documents should tell users how to use and administer the system [45].

The work presented in this Thesis provides a software process improvement model. This model is an iterative model. The main objective of this model having done software process improvement by stepwise. First define the area of process improvement after understand the process. A lot of problems are existing. Which are the causes of unsuccessful project? Unsuccessful project means either they have no good quality or they have no completion with in time and budget. First clear the problem then next step is make an assessment team for improvement according to the size of project. Then this team is recommended by the higher authorities. This team must be well qualified and skilful. After team completion the next face come. Team start the work on the process improvement. They have done planning for process improvement at which area improvements are needed or changes are required. When they make a plan then discuss with the senior management. Then they give a presentation to the senior authorities and convince them for the improvement. After planning the next phase come. Create the solution according to the plan. If more then one solution is exist for single problem then choose the best of one then implement it. It is suggested not apply whole solution once. Make the base line and take a pilot test at every step and analysis the result at every step. After implementing the plan next phase come inspection. Inspection is basically a testing what have you doing and what have you done. Inspection is using for finding the defect. It analyse where are you wrong. It is a formal meeting in which member of inspection team have taken interview of the assessment team. If the plan is correct then give the clean chit. If there are modification are needed then modify it. This work is done by the configuration management. Identify where the changes are need then apply it.

This model is a cyclic model. After configure it will go for problem and same procedure will done again till the process improve and give a better output rather then

the normal process. One case more if something change in the problem then after complete the cycle it go again problem and done all process according to the new problem. In this model documentation is an umbrella activity. That is useful for reuse and further future work.

5.3 COMPARISON WITH SPIM TO OTHER MODELS

This model is different to other model. Its work methodology is different with other models.

5.3.1 COMPARISON BETWEEN CMM AND SPIM

Table 5.2: Comparison between CMM and SPIM

S.NO.	CMM	SPIM
1	The main problem with the CMM model is that CMM give the maturity level. That an organization across the level then it's enter the next level. But CMM does not specify the implementation.	In SPIM model implementation is defined and a proper document is prepared for every process.
2	CMM works on software.	SPIM model works on software and organization.
3	CMM is a goal. Being used just as stamp of approval	SPIM model is a method.
4	CMM is works only a repeating task.	But SPIM model is not only repeating task but also whole. If problem is change then this model works due to a cyclic model

5.3.2 COMPARISON BETWEEN IDEAL MODEL AND SPIM

Table 5.3: Comparison between IDEAL Model and SPIM

S No	IDEAL MODEL	SPIM
1	Ideal model is also a continuous model. But it is a full method such that there is no recovery. Means either it is success or fail.	SPIM model is a cyclic model. So if any problem face then it will work until improvement has not completed.

5.3.3 COMPARISION BETWEEN SIX SIGMA AND SPIM

Table 5.4 Comparison between SIX SIGMA and SPIM

S.NO	SIXSIGMA	SPIM
1	Six Sigma is a statistically-based process improvement methodology that aims to reduce defects to a rate of 3.4 defects per million defect opportunities by identifying and eliminating causes of variation in Business processes.	SPIM model is a planned methodology of continuous improvement methodology.
2	SIX SIGMA is iterative methodology reduced the defect one by one.	SPIM model is a cyclic and iterative model that improves the process one by one step. After completion of last step if further improvement is needed the cycle start again.
3	SIX SIGMA methodology works on two approaches DMAIC, DMADV.	SPIM works as a full flash model.
4	Six Sigma focuses on prioritizing and solving	SPIM model does not focus on prioritizing and solving problems.

	Specific problems which are selected based on the strategic priorities of the company and the problems which are causing the most defects.	SPIM model solve all types of problems.

Chapter 6

CONCLUSION AND FUTURE WORK

CONCLUSION

The proposed Software Process Improvement Model (SPIM) improves the process in a traditional way. This model is an iterative model. The SPIM model does the process improvement in a stepwise way. It covers user requirements, software quality assurance, and organization point of view. Many of the factors can be found in the organization from the SPIM model like management commitment and teamwork were strengthened.

SPIM model covers the some limitation of existing model (CMM, SIX SIGMA, and IDEAL). For example, the main limitation of CMM is key practice describes “what to do” but does not prescribe “how to do”. SPIM model describe the implementation and prescribe how to do.

The SPIM Model does not necessary to work for the repeatable task. When the new problems come it will work for that also.

SPIM is a flexible model. If there is a change in the process, SPIM covers all the aspect of the changing of process due to cyclic model.

FUTURE WORK

This proposed model in the future can be implemented after finding out the Key Process Areas (KPAs), then creating the templates and forms, after conducting exhaustive survey. Success of this model depends on the depth of the survey. The key process areas and capability area of SPIM model can also be defined and extended ion future.

SPIM model reduces the risk as much as possible but it has also some limitations. It takes so much time in presentation and conduct the meeting. In the further work a lot of work can be to remove the timing problem of the SPIM model.

Chapter 7
REFERENCE

[1] Jesper Arent, Jacob Nørbjerg, "Software Process Improvement as Organizational Knowledge Creation: A Multiple Case Analysis", Proceedings of the 33rd Hawaii International Conference on System Sciences – 2000.

[2] Dr. David J. Szymanski and Capt. Thomas D. Neff, "Defining Software Process Improvement"<http://www.stsc.hill.af.mil/crosstalk/1996/02/defining.asp>, February 1996.

[3] Laurie Honour Werth, "Lecture Notes on Software Process Improvement", document number CMU/SEI-93-EM-8, copyright by Carnegie Mellon University, "Introduction to Software Process Improvement", 1993.

[4] The American Corporation Today< http://www.questia.com/googleScholar.qst?docId=80352426>

[5] Mohd. Hairul Nizam, Mohd. Nasir, Rodina Ahmad, Noor Hafizah Hassan, "Resistance Factors in the Implementation of Software Process Improvement Project", Volume: 4, pp. 1-10, 26-28 Aug. 2008.

[6] Anthony Finkelstein, "Software Process Immaturity Model", Software engineering notes 17,vol. no. 4, Oct 1992 pp 22-23.

[7] Donna K. Dunaway, Steve Masters, "CMM SM -Based Appraisal for Internal Process Improvement (CBA IPI): Method Description", April 1996.

[8] http://www.tantara.ab.ca/a_isorel.htm

[9] http://cmmcmmi.wordpress.com/2007/01/16/history-of-cmm/, History of CMM, Posted by obi on January 16, 2007.

[10] Mark C. Paulk, Charles V. Weber, and Mary Beth "Chrissis Software Engineering Institute, USA, The Capability Maturity Model".

[11] Jianguo Li, Jinghui Li, and Hongbo Li, "Research on Software Process Improvement Model Based on CMM".

[12] http://www.sei.cmu.edu/ideal/ideal.bridge.html

[13] Software Testing Standards-CMM Levels, url < http://forum.onestoptesting.com/forum_posts.asp?TID=3368>

[14] Thomas Pyzdek, "The Six Sigma Project Planner : A Step-by-Step Guide to Leading a Six Sigma Project Through DMAIC", McGraw-Hill, 2003.

[15] Zhedan Pan, Hyuncheol Park, Jongmoon Baik,Hojin Choi, "A Six Sigma Framework for Software Process Improvements and its Implementation", Proceedings of the 14th Asia-Pacific Software Engineering Conference, ISBN ~ ISSN:1530-1362 , 0-7695-3057-5, pp. 446-453 2007.

[16] Bhote, K. R. "Motorola's long march to the Malcolm Baldrige National Quality Award, National Productivity Review", pp. 365-376, 1989.

[17] Watts S. Humphrey, "Introduction to Software Process Improvement ", 1992 (Revised June 1993) .

[18] Alain April, François Coallier, "Trillium: A Customer-Oriented Assessment Method for Software System Development Capability".

[19] Alain A. April, François Coallier, "Trillium V3.0 A Model for the Assessment of Telecom Software System Development Capability".

[20] <http://www2.umassd.edu/swpi/BellCanada/trillium-html/t3modc1.html#Overview.Objectives>.

[21] Martin Michlmayr, "Software Process Maturity and the Success of Free Software Projects".

[22] http://www.qualitydigest.com/oct99/html/iso.html

[23] http://www.qualityspctools.com/scatter.html

[24] Renato L. Della Volpe, Farley S.M.Nobre, Marcelo S.P. Pessoa , Mauro Spinola "The role of software process improvement into total quality management: *an* industrial experience".

[25] Timo Makinen, Timo Varkoi, Hannu Jaakkola, "Assessment of a Software Process Assessment Process".

[26] http://www.tickit.org/scheme.htm

[27] An Introduction to the Documents url< http://www.sqi.gu.edu.au/spice/suite/intro.html >

[28] D. H. Kitson, "Relating the SPICE framework and SEI approach to software process assessment", volume 5, number 3 / September, 1996, 145-156.

[29] Mitchell, K. D. Ram "Total Quality Management in Software Engineering".

[30] Paulk.M,Tqm and CMM: Software CMM Q&A #4, Software engineering Institute, ,url: <http://www.sei.cmu.edu/activities/cmm/docs/q- aanda-.4.html> April 7th 1997.

[31] http://www.tickit.org/scope.htm

[32] http://en.wikipedia.org/wiki/TickIT

[33] http://en.wikipedia.org/wiki/Ishikawa_diagram

[34] http://www.skymark.com/resources/tools/run_charts.asp

[35] http://www.isixsigma.com/library/content/c010527a.asp

[36] http://www.qualityspctools.com/scatter.html

[37] http://www.isixsigma.com/library/content/c010527c.asp

[38] http://www.isixsigma.com/library/content/c010527c.asp

[39] Pave1 Mach, Jessica Guhqueta, "Utilization of the Seven Ishikawa Tools (Old Tools) in the Six Sigma Strategy", 24th International Spring Seminar on Electronics Technology, Calimanesti-Caciulata, Romania May 9,2001.

[40] http://www.itl.nist.gov/div898/handbook/pmc/section3/pmc31.htm

[41] R. Austin, D. Paulish, "A Survey of Commonly Applied Methods for Software Process Improvement", February 1993.

[42] http://yunus.hacettepe.edu.tr/~sencer/research.html

[43] http://www.absoluteastronomy.com/topics/James_Martin_(author)

[44] http://www.qfdi.org/what_is_qfd/what_is_qfd.htm

[45] Ian Sommerville, Lancaster University, UK, "Software Documentation, Printed 7/11/01" , 1992.

[46] Pierre Juliard, Juliard Associates Brookfield, Connecticut, "Total Quality Management - making the concept work".

PAPER ACCEPTED/COMMUNICATED

1. Pankaj Kumar, Seema Bawa "A Software Process Improvement Model", International Conference on Information System and Software Engineering Icisse-09, Meenakshi Sundararajan Engineering College, Chennai, Tamilnadu India from December 28-29, 2009. [Communicated]