PROCESS CAPABILITY PROFILE BASED ON CAPABILITY MATURITY MODEL INTEGRATION (CMMI) FOR SMALL AND MEDIUM ENTERPRISES

Dr. S. Vince Raicheal

Dr. M. V. Srinath

JUPITER PUBLICATIONS CONSORTIUM

www.jpc.in.net

PROCESS CAPABILITY PROFILE BASED ON CAPABILITY MATURITY MODEL INTEGRATION (CMMI) FOR SMALL AND MEDIUM ENTERPRISES

Authors:
Dr. S. Vince Raicheal
Dr. M. V. Srinath

ISBN: 978-93-91303-70-9

First Published: July 2023
DOI: www.doi.org/10.47715/JPC.B.978-93-91303-70-9
Price: 400/-
No. of. Pages: 160

Published By:

Jupiter Publications Consortium
22/102, Second Street, Venkatesa Nagar
Virugambakkam, Chennai 600 092.
Tamil Nadu, India.
www.jpc.in.net

Printed by: Magestic Technology solutions (P) Ltd
Www.magesticts.com

ACKNOWLEDGEMENT

In quiet humility, standing before Lord Almighty, a heart swells with profound gratitude. His mighty name echoes in hushed whispers, homage paid for divine intervention that breathed life into this tome of knowledge. In every stage of evolution, His benevolent grace remained omnipresent, acting as a lighthouse in an otherwise overwhelming sea of complexity, guiding each pen stroke in the process of crafting this monumental work. His blessings, abundant and ceaselessly generous, became instrumental forces, steering me toward this landmark achievement and helping to realize this labor of love and devotion. Amidst this endeavor, the fortress of familial bonds emerged as my bulwark. Their unwavering support, a ceaseless source of inspiration, created a momentum that drove me onward. Foremost amongst them, cherished parents. Their love, deep as the ocean, and wisdom, bright as a star, served as my guiding constellations on this introspective journey of self-expression and discovery. In each heartfelt pulse of gratitude,

I honor my siblings; their encouragement served as a strong wind under the wings of my spirit, uplifting even during the darkest of hours. Reserved for my esteemed partner in life, Mr. S. Sivaraj, are my deepest feelings of love and respect. His faith in my potential, unyielding and steady, became a sanctuary where I drew the strength to test and expand the boundaries of my capabilities. His unwavering support, as soothing as a balm, brought solace to my soul in its most trying times.In the shining faces of my daughters, S. Vijidha and S. Abaya Pradha, I found the greatest motivators.

Their innocent joy and unwavering belief in my abilities infused my work with a unique vibrancy and sense of purpose. Their youthful zest for life was a constant reminder of the world's beauty and potential, encouraging me to embed a sense of awe, wonder, and optimism within the book's pages.

With all humility and gratitude, I, Dr. S. Vince Raicheal, find myself at the edge of this significant achievement. A deep sense of accomplishment courses through me, bolstered by the affection and support from those dear to me, and forever indebted to the divine guidance that lighted my path.Reflecting upon this journey, the fulfillment derived from the process of completing this book matches the accomplishment itself. My life has been enriched beyond measure, filled with blessings too numerous to count and relationships treasured beyond words. It stands as a testament to the formidable power of perseverance, faith, and the warm embrace of family support.

- Dr. S. Vince Raicheal

Name of the Book:
PROCESS CAPABILITY PROFILE BASED ON CAPABILITY MATURITY MODEL INTEGRATION (CMMI) FOR SMALL AND MEDIUM ENTERPRISES

Authors:
Dr. S. Vince Raicheal
Dr. M. V. Srinath

ISBN: 978-93-91303-70-9
Volume: I
Edition: First
Published by: Jupiter Publications Consortium.
Printed by: Magestic Technology Solutions (P) Ltd.
info@magesticts.com | www.magesticts.com

www.jpc.in.net

Address: 22/102, Venkatesa Nagar, Virugambakkam, Chennai
Tamil Nadu, India
Email: director@jpc.in.net | **Website:** www.jpc.in.net

PREFACE

In the realm of business and enterprise, success is not merely an outcome but a continuous journey of growth and improvement. In a world where competition is fierce and expectations ever-evolving, organizations, especially small and medium enterprises (SMEs), face the challenge of staying ahead of the curve. They must continually enhance their processes, capabilities, and standards to thrive and deliver excellence to their customers.

It is with great pleasure and admiration that I introduce you to the insightful and invaluable work titled "Process Capability Profile based on Capability Maturity Model Integration (CMMI) for Small and Medium Enterprises," penned by the eminent authors, Dr. S. Vince Raicheal and Dr. M. V. Srinath. In this groundbreaking book, they embark on a journey to illuminate the path of success for SMEs by harnessing the power of the Capability Maturity Model Integration (CMMI).

As respected scholars and practitioners in their respective fields, Dr. Raicheal and Dr. Srinath bring a wealth of knowledge, experience, and expertise to this enlightening discourse. Their dedication to understanding the intricacies of SMEs, combined with their deep comprehension of the CMMI framework, has culminated in a work of profound significance.

Through meticulous research, comprehensive analysis, and practical insights, the authors present a step-by-step guide to assess, enhance, and optimize the process capabilities of SMEs. By integrating CMMI's principles into these enterprises' unique context, they provide a roadmap for growth and transformation.

One of the notable strengths of this book lies in its approachability. The authors skilfully navigate the complex concepts of process capability and CMMI, making them accessible to a wide audience, ranging from seasoned professionals to budding entrepreneurs. As you delve into the pages of this book, you will find a harmonious blend of theoretical foundations and real-world applications, empowering you to implement these principles within your own organization.

Furthermore, this work serves as a beacon of guidance for policymakers, industry leaders, and educators seeking to uplift SMEs and foster a culture of excellence within the business landscape. It underscores the importance of continuous improvement, innovation, and adaptability in achieving sustained success in an ever-changing global economy.

Dr. Raicheal and Dr. Srinath's dedication to sharing knowledge, their passion for advancing SMEs, and their commitment to promoting best practices shine brightly through this comprehensive work. As you immerse yourself in the content of this book, I am confident that you will gain not only valuable insights but also the inspiration to embark on your own journey of organizational excellence.

In conclusion, "Process Capability Profile based on Capability Maturity Model Integration (CMMI) for Small and Medium Enterprises" is a vital contribution to the field of business management and a testament to the author's profound dedication. It is my privilege to endorse this transformative work and extend my heartfelt gratitude to Dr. S. Vince Raicheal and Dr. M. V. Srinath for enriching our understanding of SMEs' capabilities and empowering us to chart a course towards unparalleled success.

Dr. P M Shareef, PhD
CSQA, CISA, CISM, CGEIT, CRISC, CDPSE, CSSBB, CSM, PMP, CCSK
Certified High Maturity Lead Appraiser (CHMLA)
Certified CMMI - DEV, SEC & SAF Lead Appraiser
Lead Auditor, Trainer, & Technical Reviewer
- QMS, ISMS, SMS, BCMS, PIMS+GDPR
CSA-STAR Lead Auditor, HIPAA-HITECH Assessor
QTEEM Techno Solutions Pvt Limited (ISACA Business Partner, USA)

ABSTRACT

Software development process has vital impact on Small and Medium size Enterprises (SME) which have a notable place in industry. Standards and definitions of SME are different from one country to another. Software maturity models have been a successful approach in evaluating and predicting review process capability. Many organizations in Information Technology (IT) and Information Technology Enabled Service (ITES) sectors have helped in process capability leading to improvements in statistical confidence and achievements of successful quantitative and accurate prediction. The major problem faced by several organizations is in the development of this integrated approach which is efficient and effective. Capability Maturity Model Integration (CMMI) provides the best practice-oriented approach by eliminating these barriers and issues faced by several organizations, whereas CMMI for development comprises best practices which focus an applicable advancement in software development activities for products and services provided by a company or organization. This method presents the process capability profiles of software organizations with a particular business model. The proposed method is made up of a process, artifacts and guides that assist acquisition of process profiles, based on the precise characteristics of each software company. Therefore, the application of the method permits selection of a set of attuned processes and wrinkled up with the business model proposing more objective improvement actions to a software organization. In this research work, the focus of the study is on experiments relating to the Process Maturity model which is proposed for the determination of the strong correlation among the reported process maturity to process capability profile suggesting continued and sustained process improvement with process performance initiatives subsequent to the appraisal in SMEs.

TABLE OF CONTENTS

List of Symbols

SYMBOLS		DESCRIPTION
T	-	Required Development Time in years
C	-	Parameter reliant on the development environment
E	-	Total person Months
Size	-	Standard project sizes
a and b	-	values of constants
Effort	-	Effort estimation of every PMAT level to all standard project sizes
KLOC	-	the estimated number of delivered lines of code for the project
A	-	Constant
B	-	Constant
C	-	Constant
D	-	Constant
E	-	Exponent
F	-	Stated is the schedule equation exponent derived from the fiveScale Factors
EM	-	Effort Multipliers
SF	-	COCOMO II◎s scale factor
PM	-	Person-Months
TDEV	-	Time to Develop
N	-	Effort Multiplier numbers
E	-	Cumulative of Scale Factor
PMAT	-	Process Maturity
Parameter	-	Computation effort value

List of Abbrevations

ABBREVIA-TION		DESCRIPTION
SME	-	Small and Medium size Enterprises
EU	-	European Union
IT	-	Information Technology
ITES	-	Information Technology Enabled Service
CMMI	-	Capability Maturity Model Integration
SPI	-	Software Process Improvement
SPICE	-	Software Process Improvement and Capability determination
PA	-	Process Areas
PCP	-	Process Capability Profile
PMAT	-	Process Maturity
PCDE	-	Profile Driven Process Engineering
SW-CMM	-	Software Capability Maturity Model
PP	-	Project Planning
RD	-	Requirements Development
CM	-	Configuration Management
OPF	-	Organizational Process Focus
PMC	-	Project Monitoring and Control
TS	-	Technical Solution
PPQA	-	Process and Product Quality Assurance
OPD	-	Organizational Process Definition

SAM	-	Supplier Agreement Management
PI	-	Product Integration
MA	-	Measurement and Analysis
OT	-	Organizational Training
IPM	-	Integrated Project Management
VER	-	Verification
DAR	-	Decision Analysis and Resolution
OPP	-	Organizational Process Performance
RSKM	-	Risk Management
VAL	-	Validation
CAR	-	Causal Analysis and Resolution
OPM	-	Organizational Performance Management
REQM		Requirements Management

CHAPTER – 1
INTRODUCTION

1.1 Overview

Software development process has a vital impact on Small and Medium size Enterprises (SME) which have a notable place in industry. Standards and definitions of SME are different from one country to another. In a sector, small enterprises have employees from 5 to less than 30 and medium enterprise have employees from 30 to 75 based on the standards of SME. The capacity of a small enterprises is 10 to 75 and 50 to 249 for medium enterprises in the case of European Union (EU) [1]. Software maturity models have been a successful approach in evaluating and predicting review process capability. Many organizations in Information Technology (IT) and Information Technology Enabled Service (ITES) sectors have helped process capability leading to improvements in the statistical confidence, achievements of successful quantitative and accurate prediction.

CMMI development is designed to meet the challenges seen in the changing global business landscape. The performance of V 1.3 has energized business through standards and building key capabilities [2]. The core process areas of V 1.3 CMMI development constitute an established set of universal best practices organized by critical business capabilities which develop business performance. There are major common challenges relating to the critical capabilities of any organization, including business and emerging products, performance improvements, structure and supporting skills, business management flexibility, preparation and handling work, choice and dealing providers that are meant for quality, workforce management and supporting implementation. This research work has been analyzed the metrics of data collection and derive quantifiable results in a software organization for assessing and managing the software review process efficiencies for development and their capability profiling.

Sections 1.2 and 1.3 of this chapter state the background and the need for the study described in dissertation. Next, Section 1.4, 1.5 and 1.6 summarize the statement, objectives and the scope of the study. Finally, Section 1.7 outlines the remaining chapters.

1.2 Background of the Study

Software process model serves as a groundwork for the process definition, assessment and improvement. It guarantees the handling of the concepts, significance with the finest software engineering practices and compatibility with globally accepted standards. The organizations should decide the process assessment model more appropriate to their main goal. This helps reaping the advantages of both models. The improvement of the quality of software service or products of Software Process Improvement (SPI) aims to exploit the benefits of economies followed by Small and Medium size Enterprises(SME) [3].

Almomani has proposed to the current practices of software process evaluation based on SPI which gets a chain of iterative and continuous practices. These kind of changes in continual process and advancement of novel practices have been combined for managing the activities in the process of software development, SPI gives attention to the weakness of current practices and organization’s software requirements. Outstanding quality models, namely, six sigma, ISO 9001, Capability Maturity Model Integration (CMMI) and Software Process Improvement and Capability determination (SPICE) [4].Tailoring CMMI for the benefits of SMEs is no easy task, due to its complexity. The need for enough resources such as skilled professionals, challenging deadlines and high implementation costs, is felt by SMEs firms [5].

Shareef [6] has proposed the basic framework of CMMI-Development. Capability Maturity Model Integration (CMMI) from SEI and currently administered by CMMI Institute aims at achieving organization process maturity and capability that enables process performance using a systematic Process Improvement approach. It consists of the best practices that cover the product lifecycle from Inception through delivery and maintenance. It can be used for guiding process improvement across a project, a unit / division or an entire organization. CMMI helps in setting process improvement goals and priorities and providing a reference for quality processes and for appraising current processes.

Process Maturity acts as a catalyst for the organization's success in meeting business objectives. The framework groups best practices into Constellations.

There are 3 constellations:

1. CMMI-Development
2. CMMI-Acquisition
3. CMMI-Services

CMMI-Development constellation comprises of practices that cover project management, process management, systems engineering, hardware engineering, software engineering, and other supporting processes used in development and maintenance projects. There are 16 core process areas in CMMI spread across 5 maturity levels which are highly robust.

Nowadays, the tendency in defining appropriate method to SMEs appear as evidence by considering the study of a few current Process Areas (PAs) in CMMI with the method of agile and its practices.

CMMI has an increased number of software organizations adopting agile software development methods. In 2015, CMMI appraised software organizations using one or more agile methods reported by more than 70% of CMMI [7, 8]. This method was utilized for evaluating basic capabilities and process performance with the evaluation of basic laboratory capabilities carried out through the help from software quality assurance officer or laboratory manager for generating a standard and also for discovering focus areas in project improvement. This evaluation is time consuming butis significant and also gets circumscribed with a wide design area involved with technical skill, quality management, equipment, supply chain, Bio- security and management in laboratory [9].

1.3 Need for the Study

In a current situation an organization needs to improve all kinds of components that compose a complex product or service. In general, several components are in-house creations whereas a few get accomplished in order to integrate all the components into consequent services and products.

This complicated process of improvement and maintenance has been controlled and managed in organization and the issues faced now in the organizations considered at present have been included the results to enterprise wide which is an essential approach for integration. Effective management of organizational assets is critical to business success. The major business objective is advancement of activities required for management of their essential needs in organizations including services and products as significant. Development needs attention to several aspects with the assistance of present standards, guidelines, methodologies, maturity model and market places. Moreover, the focus of any frequently applicable method of development is a business with a particular part without consideration of a general method for the issues which several organizations are facing. The focus of development of a significant area in business has unfortunately maintained the barriers of these models that occur in organizations.

SMEs form an essential subsector of strategic business services which involve services associated with information processing, computer software, advancement, marketing, research, business organization and improvement over human resources. Outsourcing improvement using main manufacturing firms has a combination of recent technologies which have helped SMEs in getting success in market places with a 10% annual growth in knowledge-based services in the present decade. Economic growth worldwide has served as an important driver for service industries whereas mature service improvement practice and advancement guidelines constitute a major source for the performance of the service providers and client satisfaction. A basic presentation which contains an important source for effective process to the organization is Capability Maturity Model (CMM) which has provided a simple description of the world that contains an essential component to effective process in organizations. These produce mature processes with improved quality and effectiveness whereas nowadays CMMI is an application of the principles introduced almost a century ago for continued improvement which has got established over time.

Organizations have experienced improved productivity, quality, and

cycle time with high accuracy and predictable schedules with low costs. The group of CMMI components collection is said to be constellation which is utilized for creating models, material training and appraisal of associated documents based on the area of interest. The constellation model on the basis of service is known as "CMMI for services" or "CMMI-SVC" whereas the practices and goal are consequently possible
significant for any organization concerned with service delivery involved enterprises in area namely defense, information technology (IT), health care, finance, and transportation. There are several model components which get grouped into three categories, namely,1. Required, 2. Expected, and 3. Informative compounds which get interpreted whereas the generic goal satisfaction is utilized for appraisal as a basic function for enabling decision on the satisfaction of the processed area. Similarly, expected components have guided the implementation development to perform appraisals using generic practices. It is often impossible to adequately describe the behavior required or expected of an organization using only a single goal or practice statement. The characteristics describe the generic goal which is presented to institutionalize process for the application of a process area whereas the goal is an essential model component which is utilized in appraisal for determining the determined process area. Moreover, a detailed description is a sub-practice which helps guidance to interpret and implement generic practices as an informative component considered only for the present ideas beneficial to the improvement process in organizations.

1.4 Statement of the Problem

Nowadays, each organization needs delivery of its services and products in the market with better, quick and cost-efficient features. Each organization generates complicated products and services which needs improvement which seems complicated. Therefore, organizations select a few components that need in-house improvement with some of them to be obtained from other organizations whereas the final service or product gets involved by integrating of all components. Moreover, all these activities in the organization should have capability and maturity for controlling and managing their own processes of complex development and maintenance. A major problem faced by several organizations is in the development of this integrated approach which is efficient and effective. Capability Maturity Model Integration (CMMI) provides the best practices-oriented approach by eliminating these barriers and issues, whereas CMMI for development comprises the best practices which focus on applicable advancement in software development activities for products and services provided by a company or organization. However, 80% of the global economy is serviced from the CMMI for obtaining a services model from CMMI Institute which has assisted these service organizations improving their processes and enabling all their resources achieving the best business results. Therefore, CMMI for Service (CMMI-SVC) model is utilized as a guide for assisting service provider organization through cost reduction, quality improvement and consistency in delivery of services. Thus, best practices of these model have helped in getting sizeable profits and improve process capability with performance.

There are several components used in CMMI but one of the key components for improving the products and services of the organization is meeting the client requirement in an appraisal. Generic goals are so called, due to similar goal statement available from multiple process areas.

The characteristics describe the generic goal which is presented to for institutionalizing a process area. These components assist the management to discover the strength and weakness of their in-house software development team. This process development has an important impact in Small and Medium size Enterprises (SME) in understanding their process performance. SMEs are part of the essential subsector of strategic business services involved in the services associated with information processing, computer software, advancement, marketing, research, business organization and improvement of human resources. The outsourcing improvement using main manufacturing firms has a combination with recent technologies which have enabled SMEs to be successful in the market place with a 10% annual growth in these services of a knowledge based industry. In this research work, previous metrics on process performances and discovery of the process weakness of the organization have been the subjects of study. The benchmark of an organization against CMMI services is targeted by achieving the maturity levels. In software organization data is collected and analyzed for deriving quantitative results that help assessment and management of the efficiency of process performance in development. Its capability profiling which leads to a high correlation between process maturity and Process Capability Profile (PCP) signifies continual improvement in process initiative followed for appraisal.

1.5 Objective of the Study

CMMI development is designed to help facing the challenges of the changing global business landscape. There are major common challenges relating to the critical capabilities of any organization, including business and emerging products, performance improvement, structure, and supporting skill, business management flexibility, preparation and handling work, choosing and dealing providers for quality, workforce management and supporting implementation. The key objectives of the research are listed below

- Providing assistance to organizations in their accomplishment of project profiles considering the specific characteristics, particularly to small software companies.
- To design the development of models for the definition of process capability profiles lined up with the specific business model offering more objective improvement actions to small software companies.
- Development of software which gets affected by the organizational Process Maturity (PMAT) level for investigating the various impact in CMMI-based process maturity levels on effort, productivity rate and diseconomy of scale for all standard project sizes.

1.6 Scope of the Research work

The software development industry is dominated by numerous Small and

Medium-sized Enterprises (SMEs) with various definitions. There are variations in the definitions based on environment, countries and the software product type, size and area of interest. CMMI implementation in SMEs represents major points in finding solutions to these problems. These issues include cost, resources, effort, and software development process in software process improvement in knowledge as an addition to the complexity of the Software Process Improvement (SPI) models like Capability Maturity Model Integration (CMMI) are applied to resolve the obstacles. The focus of this study is on the following:

- Improvement of the internal software process with two representations of which the first one is continuous representation which targets on process capability, gets measured using capability levels and staged representation for appraisal with focus on organizational maturity which gets measured by maturity levels.
- A study and review of the metrics of data collection and analysis utilized for deriving quantifiable results in a software organization for assessment and management of the software review process efficiency in development and their capability profiling.

This approach has presented the PCP of software companies with the specific business model whereas this proposed technique has been considered with a guides, process and artifacts which get assistance for acquiring profile process based on the precise features of every software organization.

1.7 Organization of the thesis

Chapter-1 describes the research work with analysis of the metrics of data collection and derivation of quantifiable results in a software organization for assessment and management of the software review process efficiency for development and its capability profiling. This chapter states the background and the need for the study. It also summarizes the problem statement, objectives with scope of the study. Section 1.2 and 1.3 of this chapter state the background and need for the study described in the dissertation. Next, Section 1.4, 1.5 and 1.6 summarize the statement, Finally, Section 1.7 provides an outline of the remaining chapters.

Chapter-2 describes characteristic of effective processes of CMMI evolution based on CMMI-Development, CMMI-services and CMMI-Acquisition process area. The specific goals and practices that satisfy the process area in achieving the associated specific goals are presented. The generic goals and generic practices that must be presented to institutionalize processes for the implementation of a process area in achieving the generic goals are presented. This chapter describes CMMI for evaluating the organizational maturity and process capability. These practices form a baseline which supports the decision on process improvements. An appraisal activity that can help organizations at any stage of the CMMI adoption is also presented. Then CMMI appraisal method, types of CMMI appraisals, CMMI levels of capability and performance and published appraisal result system is also given.

Chapter-3 deals with the quality of software service or products of Software Process Improvement (SPI) that aims to exploit the benefits of economy followed by Small and Medium size Enterprises(SME). It also describes the Capability Maturity Model Integration (CMMI) and Software Process Improvement and Capability determination (SPICE). This section reviews the definition of SME's. Factors that make SMEs unique in the provision of the basic knowledge are reviewed. This chapter presents a proposal towards a Process Capability Profile Driven Process Engineering (PCDE) as an evolution of the current Software Process Improvement based on Process Capability (and Maturity) Models. It also discusses the development of a software project with acceptable quality within a budget and on planned schedule as the main goal of every software development firm. Schedule estimation has historically been and continues to be a major problem in managing software development projects. It describes the COCOMO II Scale Factors and Cost Drivers. Review based on Software Capability Maturity Model (SW-CMM) published by SEI is used to rate an organization's process maturity.

Chapter-4 deals with the major problem and issues faced by several organizations in developing an integrated approach which is efficient and effective. This chapter describes the Capability Maturity Model Integration (CMMI) and provides details of the best practice-oriented approach by eliminating these barriers and issues faced by several organizations, whereas CMMI for development comprises best practices which focus on applicable advancement in software development activities for products and services provided by a company or organization.

Chapter-5 describes the components available in every process area and even in the generic goals which consider these components which are critical for utilizing the information. Moreover, all the model of CMMI's are created from the CMMI Framework which consists of both goals and practices which have utilized for generating the model of CMMI that existed for CMMI constellations. It discusses the applicability of generated concepts were temporarily applied to the pilot organization of partial effects found in this activity. This is the method of development illustrated using Process Capability Profile (PCP) based on Process Maturity (PMAT).

Chapter-6 refers to the evaluation measurement of effort estimation, scaling for diseconomy and productive rate of the CMMI based on process maturity rating level of all rating levels for all standard project size and the result of this evaluation has discussed in the next chapter.

Chapter-7 discusses the results by comparing the PMAT outcome for all kind of maturity rating level for SMEs and evaluating the changes in percentage of other factors with PMAT using parameters like percentage change in effort, percentage change in productivity and percentage change in scaling for diseconomy in the all standard project sizes.

Chapter-8 details the research outcomes provide conclusions, implications, and recommendations for further research. A list of references and appendices is also included at the end.

CHAPTER – 2
RESEARCH ESTABLISHMENT

2.1 OVERVIEW

This chapter describes the activities and development of CMMI services targeting achievement of the maturity levels for the organization unit. The main aim of CMMI is process development and framework of Software Process Improvement (SPI). The frame work groups are divided into 3 constellations, namely, 1. CMMI development, 2. CMMI service and 3. CMMI acquisition. CMMI-DEV covers 22 project areas that are meant for software improvement. The list of capabilities for each process area is discussed in this chapter. The knowledge of practices which would increase the capability of software development organization is outlined. Implementation of key development process, in additional definition, uses many levels of capabilities. CMMI helps in setting process improvement goals and priorities for quality process and appraisal of the current processes.

2.2 Capability Maturity Model Integration (CMMI)

CMMI plays several roles in the operational processes of characteristics and approach for process development [10] [11] [12]. It has one or more ideas based on collection of important elements of active processes. Process capability models are applied to resolve the obstacles which help in understanding organization processes [13]. CMMI plays the characterization of the business process that describes the theoretical background of particular organization. This can be viewed as the appearance of a dual role referred to as the concept based on CMMI model. CMMI is defined as the group of concepts which are mapped to a particular enterprise. The framework of CMMI modeling is not the exclusive feature of duality but it is somewhat distinctive of many other modeling frameworks. For example, specific software runtime system of UML model captures the indentation of such constants and relation among the concepts of classes. The domains of semantic web are referred to as ontology and also as the concept of annotations [14]. The product suite of CMMI is used for producing a constellation referred to as a specific model which contains several CMMI structures and components. This generates various models, assessment materials and related training. The models are characterized by continuous representation and staged representation models. It is also categorized by types of processes. Continuous representation allows collection on the order of development with respect to business objectives of organizations, and permits comparisons between organizations by process areas. The staged representation model describes progress through a predefined, improving sequence order and consecutive levels of a confirmed path. Each level functions as the next level of maturity. This allows the organizations by maturity levels that provide for appraisal results. This chapter deals with a staged representation model. It includes Software Process Improvement (SPI) models like CMMI which target on process capability gets measured using capability levels. The CMMI models namely,

- CMMI for Acquisition [15]
- CMMI for Development [16]
- CMMI for Services [17]

This research work proposes the basic framework of CMMI-Dev as shown in figure 2.1. It is used for covering project management, process management and other supporting processes used in the development and maintenance of projects.

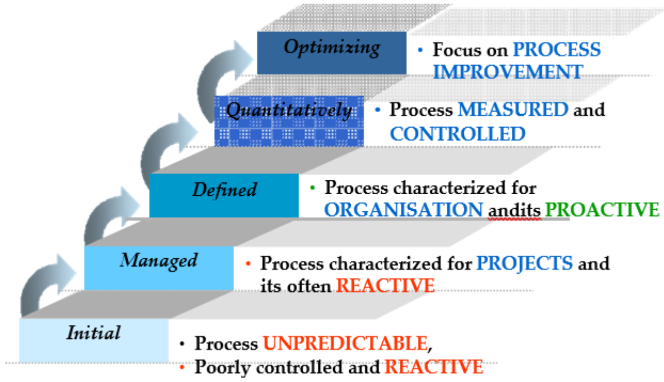

Figure 2.1 CMMI Framework

2.2.1 Evolution of CMMI

The project of CMMI was made using several CMM's to sort out the problem. In their search for enterprise-wide process development, the combination of selected models into a framework of single improvement was proposed by an organization. Developing a combination of the existing set of material models involved are less than the integrated developing set of models. The product team of CMMI can construct a structure by using a process that promotes agreement and provides accommodation for several constellations. CMMI-Dev model was the first model to be established that led to CMMI version 1.3 has shown in figure 2.2.

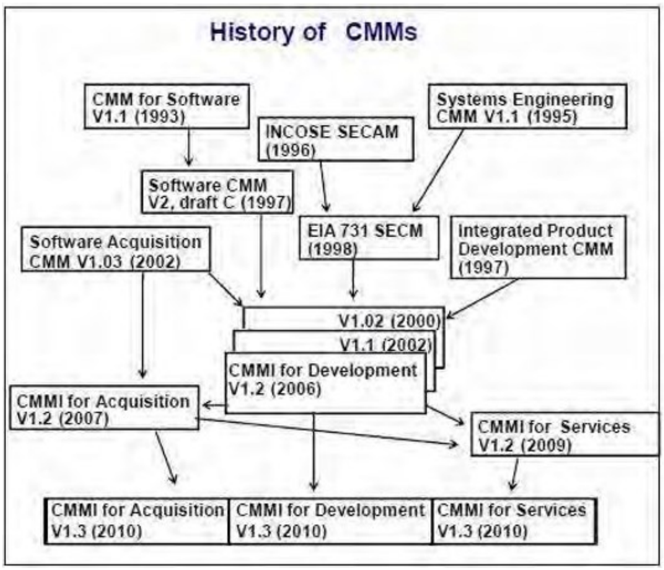

Figure 2.2 The History of CMMs

2.2.2 CMMI for Development

CMMI for Dev includes development of activities of both service and product. The reference model of CMMI-Dev is used in many organizations, namely, automobile industry, military defense, software industry, banking sector, hardware devices of the system and telecom department. It is mainly used in maintenance and development and contains the practices of hardware system engineering, process and project management, software engineering and additional supportive processes. In any organization, common sense and qualified judgment are used for interpreting a model. For many of the users, the process area defined in this model shows behavior measured as the best practices to be inferred using detailed information relating to CMMI-Dev on organization controls and business environment. It consists of 22 different process areas based on the set of process requirements that are industry considered as best practices organized across four groups as shown in Table 2.1.

The process management process areas are used across the organization for development, description and organization at the process's organization level. The process areas deploy processes that are essential considering how effectively the organization plays a large part in a new program [18].

Table 2.1: CMMI-DEV Process Areas

Project Management	**Engineering**	**Support**	**Process Management**
Project Planning (PP)	Requirements Development (RD)	Configuration	Organizational Process Focus (OPF)
Project Monitoring and Control	Technical Solution(TS)	Process and ProductQuality Assurance (PPQA)	Organizational Process Definition(OPD)
Supplier AgreementManagement	Product Integration (PI)	Measurement andAnalysis	Organizational Training (OT)
Integrated Project Management (IPM) Organizational	Verification (VER)	Decision Analysis and Resolution (DAR)	Process Performance (OPP)
Risk Management (RSKM)	Validation(VAL)	Causal Analysis and Resolution (CAR)	Organizational Performance Management
Organizational Performance Manage-	---	---	---
Requirements Management (REQM)	---	---	---

CMMI-Dev states the best practices but it does not dictate particular processes that suppliers integrate into their improvement process. In the appraisal of an organization, the degree to which an organization improvement processes co-ordinate to a representative sample of programs is measured using an appraisal for CMMI- Dev. Process capabilities and managing process improvements are the appraisals for several organizations. The appraisal can consider a number of process areas that result in a capability level profile or maturity level rating for the organization. Its use depends on the model representation. There are two representations available in CMMI-Dev for appraisals.

1. Continuous
2. Staged

The above-mentioned representation model leads to the rating of capability levels and maturity levels. Each process area has a grouping that predefines the appraisal structure known as the staged representation model. The appraisal is done independently for each selected process area known as the continuous representation model. For a variety of reasons, organizations may select one model over another. This includes the present state of the apparent needs of business objectives, continuous improvement in enterprises and specific methods for supplier's past knowledge. In general, describing and improving the processes of organization progress is measured by maturity or capability numerical levels. The indication of higher levels in the organization has been increasing the extent of process improvement efforts on institutionalization and sophistication. The representation model of CMMI-Dev has four capability levels within each process area which can be represented by the number 0-3 that measures process capability. It permits the organization to appraise and choose the process area based on its process development and business purposes. Appraisals in capability level profile achieved within each selected process areas are interpreted using continuous representation as follows.

- Capability level 0 indicates the method as either not achieved or the achievement being just moderate.

- Capability level 1 indicates the achievement of the method is to the degree that it gets the goals of the process and produces the essential yields.
- Capability level 2 indicates the accomplishment of the method in accordance with a rule.
- Capability level 3 indicates the method as personalized from the organization's set of typical methods.

The representation of staged model states that grouping of 5 maturity levels is ordered on a group of process area. It predefines the effectively appraised process area to be achieved as a maturity rating level. The effects of maturity rating level appraisal for staged model representation is as follows.

- Maturity level 1 indicates developments as typically ad hoc.
- Maturity level2 indicates the organization for each process area maturity level2 having attained capability level2. Maturity level 2 emphasizes on the support process areas and project management which are mainly established at the program level.
- Maturity level 3 indicates the organization for all the maturity level 2 and 3 as having achieved the capability level3 process area containing process management and engineering process area. It has a group of standard processes that direct the organization with approved process with emphasis at an organization level for particular programs.
- Maturity level4 indicates the organizational activities and the program as established. It makes quantitative accomplishment of the selected processes I organization that are estimated as significant and reliable with the business objectives. These can be achieved as the capability level3 is based on all process area of maturity level 2, 3 and 4.
- Maturity level 5 indicates measurement data meant for increase in the optimized selected processes that the organization has established and the sub processes having attained the capability level3 model on all process area.

The continuous representation in the organization can convert the process areas with appraisal results using equivalent staging into an organizational maturity level.

2.2.3 CMMI for Acquisition

CMMI-ACQ is the process of maturity models which are a part of the CMMI product family. It delivers the opening for organization acquisition.

- Issues and barriers can be avoided using the efficiency of the improved operational features in the acquisition process.
- Solicitations for obtaining the products and services contain supplier capability management, and help management of matters including supplier agreements and sourcing.
- Suitable technology can help exploitation of solution which are of low cost and help rapid distribution.

It was established at Carnegie-Mellon University in Software Engineering Institute (SEI). The main goal of business is to promote management development for improving or developing the process of an organization.

CMMI-ACQ Process Areas

It shows much of the current model of CMMI-Dev which is roughly around 70% commonality to address the role with various critical difference of the organization. This model based on CMMI framework, has 22 process areas with 16 shared by CMMI model and 6 specific to acquisition practices. There are six different process areas which are specific to acquisition. These are:

1. Acquisition Requirements Development (ARD) — produces, improvement, and customer evaluation and predetermined necessities.
2. Solicitation and Supplier Agreement Development (SSAD) — make a solicitation set, select one or more dealers to bring the service or product, to create and sustain the provider arrangement.
3. Agreement Management (AM) — ensures that the dealer and the supplier achieve success in the relations of the supplier agreement.
4. Acquisition Technical Management (ATM) — estimate the dealer's

mechanical solution and to accomplish selected boundaries of that solution.

5. Acquisition Verification (AVER) — ensure meeting the specified requirements based on the selected work products.

6. Acquisition Validation (AVAL) — establishes the proposed use of acquired product or service when located in its proposed environment.

Moreover, the model contains the directions on

* Acquisition Plan
* Distinctive Supplier Deliverables
* Changeover to operations and maintenance
* Groups and teaming

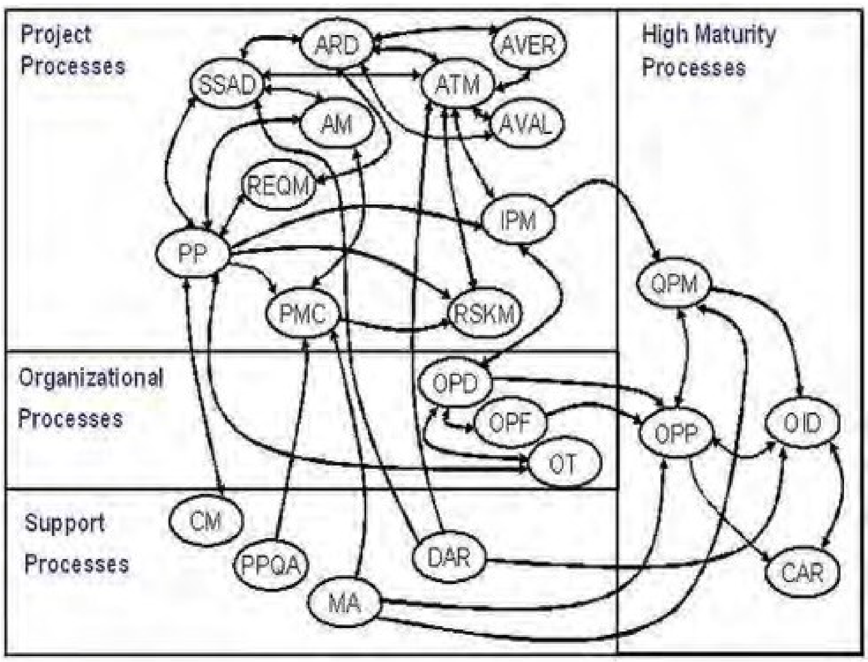

Figure 2.3 Process Areas of CMMI-ACQ

There are 16 shared process area practices contains infrastructure, support, organizational process and project management.

Table 2.2 Process Maturity Levels

Name	Abbr	ML	CL1	CL2	CL3	CL4	CL5
Agreement Management	AM	2					
Acquisition RequirementsDe-velopment	ARD	2					
Configuration Management	CM	2					
Measurement and Analysis	MA	2					
Project Monitoring andControl	PMC	2	**Target Profile2**				
Project Planning	PP	2					
Process and Product Quality Assurance	PPQA	2					
Requirements Management	REQM	2					
Solicitation and supplier Agreement Development	SSAD	2					
Acquisition TechnicalManagement	ATM	3					
Acquisition Validation	AVAL	3					
Acquisition Verification	AVER	3					
Decision Analysis andResolution	DAR	3					
Integrated ProjectManagement	IPM	3					
Organizational ProcessDefinition	OPD	3	**Target Profile 3**				
Organizational ProcessFocus	OPF	3					
Organizational Training	OT	3					
Risk Management	RSKM	3					
Organizational processperformance	OPP	4	**Target Profile4**				
Quantitative ProjectManagement	QPM	4					
Causal Analysis andResolution	CAR	5					
Organizational Innovationand Deployment	OID	5	**Target Profile5**				

2.2.4 CMMI for Service

CMMI-SVC is the process maturity model which is a portion of the CMMI product family. This structure is used for describing the maturity of their IT operations and guiding the organizations in refining their IT service delivery processes. It is applicable to any service industry as it is an industry neutral model.

CMMI-SVC helps an organization in setting up established practices into a framework for evaluation of its organizational maturity, establishment of the significance for improvement, processing of area capability and guidance to the inventiveness of process improvement.

It allows the service focused organizations with a collection of confirmed practices effectively.

* Deciding and defining the standard service to be provided for the people to known about.
* They need to provide a service that ensures everything information on equipment, processes, consumables and people.
* They need to provide a service that ensures the availability of resources at an affordable cost when required.
* Make sure nothing goes wrong, with service getting systems in place, eliminate outdated system and variations in the existing system.
* Service operate system, structure of agreements, service request should be taken care.
* Protect the organization from going in a wrong direction.
* Ensure delivery of services and getting back in the event of the occurrence of a disaster.

The model of CMMI describes "Services are valuable insubstantial and non-storable outcomes distributed through a service system". It has interactions with representations and values like ITIL, CobiT, CMMI-DEV, IT Service Continuity Management and ISO/IEC 20000.

CMMI-SVC Process Areas

- **Capacity and Availability Management:** Ensuresprovision of resources effectively for supporting service necessities and functioning of the service system presentation.
- **Configuration Management:** In order to create and maintain the reliability of work, products using variation and permission to modify formation

labelling based on formation of identification, auditing based on checks to validate changes and accounting status based on work products.

- **Analysis Decision and Resolution:** The alternative option for methodical selection using ordering, norms and appraisal method.
- **Incident Resolution and Prevention:** To ensure prevention and active determination of service incidents as suitable and ensuring with time.
- **Integrated Project Management:** Planning of work with the best practices and rules. The estimation to be based on the structural historical data. Classify needs and participants for the organization, and feed this information into an agenda or the general work plan. As work progresses, organize all key investors. Use thresholds to activate the corrective action, namely, schedule and work deviance metrics.
- **Measurement and Analysis:** To improve and ensure capability measurement used for the sustenance of the needs of management information.
- **Definition Organizational Process:** The best practices for establishing historical data into a valuable and practical archive.
- **Focus of Organizational Process:** Organize development. Take what should be learnt at the team level across the organization to establish and organize information. The outcomes are to ensure all teams perform quicker from the positive and negative modules of others.
- **Organizational Training:** Evaluate, arrange and organize training across the organization in areas which include domain-specific, knowledge and development skills desirable to decrease mistakes and develop team effectiveness.
- **Process and Creation Value Declaration:** The objective to offer the staff and management for understanding into related work products and development.
- **Requirements Management:**

I. Describe the facilities needed for the collection

II. Suggestion of distinct facilities to team actions

III. Confirm that resources, service characterization and match with the actual work done

- **Risk Management:** In a project, measure and arrange all kinds of risks that improve moderation activities for the maximum significance ones. The list of common risksstart by considering a predefined use of method for location of priorities.
- **Service Continuity:** Ensure the stability of services that establish and sustain strategies during any important disturbance of standard operations.
- **Service Delivery:** Deliver services with service contracts to make, implement and develop.
- **System Service Development:** Evaluate, plan, improve, integrate, validate, and certify service systems, including service system components, to satisfy existing or anticipated service agreements. [This is an optional Process Area]
- **System Service Transition:** Organize different or suggestively altered service system mechanisms while handling their influence on enduring service transfer.
- **Strategic Service Management:** Establish and sustain typical facilities in performance with strategic needs and plans. The remaining Level 3 Process Areas are summarized below.
- **Supplier Management Contract:** The products and services can manage the acquisition from suppliers. In the absence of tradition, uncertain or integrated suppliers can be declared in process areas as not applicable. The process areas of a particular service are summarized as follows.
- **Work Monitoring and Control:** It can be taken as a suitable corrective action when the performance turns significantly from the plan for understanding the group progress.
- **Work Planning:** Create and sustain plans (main responsibilities, evaluations, risks and properties) for service work.

Table 2.3 Process Maturity

Level	Focus	Process Area	
5 Optimizing	Continuous ProcessIm-provement	Causal Analysisand Resolu-tion,Organizational Perfor-mance Management	
4 QuantitativelyManaged	QuantitativeManagement	Organizational Process Per-formance, Quantitative Project Management	
		Decision Analysisand Reso-lution,	
3 Defined	Process Standardization	Integrated Project Man-agement, Organizational Process Definition,Or-ganizational Process Focus, Organizational Training Product Inte-gration, Requirements Development,	**Quality Productivity** Risk Rework
		Risk Management,Tech-nical Solution,Validation, Verification	
		Configuration Manage-ment, Measurement and Analysis,	
		Project Monitoringand Control,	
2 Managed	Basic Project Management	Project Planning,Process and Product Quality Assurance, Requirements Management,	
		Supplier Agreement Management	
1 Initial	---		

Value of CMMI-SVC

- It is a difficult but flexible structure
- Enhanced Service Level Agreement compliance and distribution competence
- Enhanced customer fulfillment
- Improved capability consumption

- Effective changing management
- Roadmap to service maturity
- Increase advertising/reasonable control

2.2.5 Specific Goals and Specific Practices

A specific goal must be present to fulfill the process area which has distinctive characteristics for use in appraisal. It is essential for model component to help ascertain a process area is satisfactory. For instance "Reliability of standards is sustained and established" as the only statement of specific goal required for a model component from process area configuration management. The titles based on the specific goals are considered as the informative model components and the notes are related to their goals.

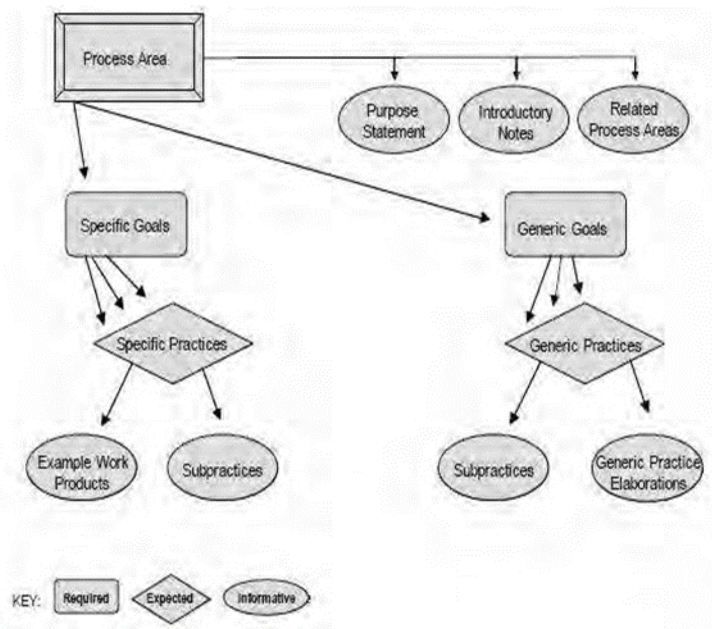

Figure 2.4 CMMI Model Components

A specific practice is found essential as the description of an activity in attaining a related specific goal. . It is expected to result in the attainment of specific goals based on the specific practices of a process area. It is also the expected model component. For instance "Monitoring promises against those well-known in the project plan" as the only statement of specific practice is the model component expected from a process area of project monitoring and control. The titles based on the specific practices are considered as informative model components and the notes are related to their practices.

2.2.6 Generic Goals and Generic Practice

Generic goals applied to multiple process area define the features whose presence is required for institutionalizing the processes to implement a process area. It is necessary for a model component to help determination of whether a process area is satisfactory. For instance "the process is established as a definite process" as the only statement of generic goal is required for modeling a component from the process area. The titles based on the generic goals are considered as the informative model components and the notes are related to their goals.

Generic practices applied to multiple process area define the features whose presence is necessary for institutionalizing the processes to implement a process area. It defines the activities to institutionalize generic goal found essential in attaining the processes related to a process area. The expected mode component of the generic practice is shown in figure 2.4. For instance, "The process based on managed process of institutionalized", delivers suitable resources for the accomplishment of the work product developing process and providing process service as the only statement generic practice expected of a model component. The titles based on the generic practice are considered as informative model components and the notes are related to their practices.

2.3 CMMI Process Improvements and Methods

CMMI has been broadly used for process capability for estimating organizational maturity during the past decades over throughout the world [19, 20]. Nowadays, most of the organizations use regular CMMI calculations and appraisals. They have self-assurance in CMMI due to its wide-ranging descriptions of how the diverse good practices fit together. Moreover, there are continuous requests from business houses for inexpensive, improved software to be carried to ever-tighter targets. Accordingly, numerous software concerns as a way of refining the quality of software need motivation for software process development, minimizing costs or speeding up their development processes [21]. Process development is defined as accepting the current processes and altering these developments to improve product value, development time and decrease cost. Two different approaches are used for process improvement and change. They are

1. Process maturity approach and
2. Agile approach

The process maturity approach is dedicated to enhancing the project management and processes that present the practice of good software engineering into software business [22]. The level of process maturity implemented in the organizational improvement processes reproduce the amount of good procedural and management training. The key objective of this method is to improve process predictability and product quality. This method is rooted in the concept of actions presented in the development of plan-driven and it normally increases overhead that is not directly related to programming. The agile approach motivates reduction in overheads and iterative development in the software practice [23]. The principal features of agile methods for changing the customer requirements are speedy distribution of responsiveness and functionality. This method is focused on the code being established and minimizing the documentation and formality [24].

Those who need improved software quality usually believe in refining the software development process. It depends on the calculation of the number of defective products and connection to the software process of these defects. The objective is to minimize product faults using examination and changes in the process. Hence, the chances of establishing faults are minimized and fault detection can be improved. Figure 2.5 shows the issues affecting the software product quality.

Figure 2.5 Issues affecting software product quality

The four important factors that affect the quality of software are:

1. Process quality,
2. People quality,
3. Development technology and cost,
4. Time and schedule [25].

The effect based on each of these issues can be influenced by type and size of the software. Many large systems established by development teams contain distinct sub-systems that may be working in various places. Hence, the major issues disturb quality of products in the software process. The main difficulties in large projects, are in, integration, communication and project management. There is generally a combination of experience and ability in team associates. Since the process of development generally takes place over many years, the development team is unstable or unsteady, with changes over the lifetime of the software project [26,27,28].

The relationship of process quality is less familiar to some extent on knowledge developments that cannot be automatic when the product is insubstantial and dependent. The software quality is not subjective but skills, experience and design process are required which are significant for the engineering process. In some of the circumstances, the method uses the maximum substantial cause of the product quality, though, particularly for original applications, the people involved in the method have more impact on software quality than the development used [29].

2.3.1 Process Improvement Lifecycle

The practice of process development which is a cyclical procedure is shown in figure 2.6 It includes three sub-processes:

1. **Process quantity:** Aspects of the existing product or the project are estimated. The objective is to improve the methods based on the goals of the organization involved in the process development. These practices constitute standards which provide support when the process improvements are active.
2. **Process study:** The existing process is measured by bottlenecks and weaknesses recognized. Process representations known as process maps define the development that may be established through this step. The study may be motivated by considering the process features, namely, quickness and robustness.
3. **Process variation:** Process variations are recommended for a report on the recognized process faults. These are presented and the cycle restarts to gather information about the efficiency of the modifications.

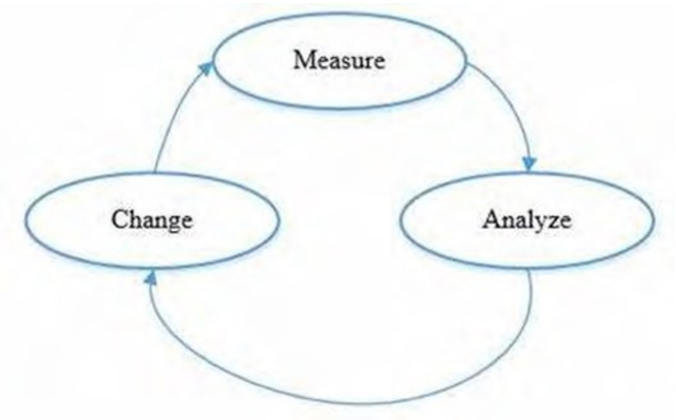

Figure 2.6 Process improvement lifecycle

Process development is an enduring activity that may last many months in the improvement process as a continuous activity. The new processes are offered, whenever the business atmosphere makes changes in the new procedures.

2.3.1.1 Process Measurement

Process measurements are quantifiable information relating to the software practice, namely, the time taken to execute some of the process action. They can be used irrespective of any developments in the effectiveness of a process measure. As this stage, 3 types of process metrics can be collected:

1. The time taken to complete the specific process. It can be calendar time, the total time dedicated to the process or the time consumed on the process by specific engineers.
2. The resources may include the complete effort in travelling costs, computer resources or personal days and also it needed for particular process.
3. The number of occurrences is based on specific events, the sample of events which may be observed, the number of defects exposed through number of necessary changes requested, average number of lines of code and the code inspection changed in response to a request change.

In the process measurement ultimate difficulty about the process in what data can be collected to sustenance of process improvement is known as Goal Question Metric(GQM) paradigm which is being broadly used as software process measurement suggested by Basili and Rombach [30]. This method is used in US space agency NASA, which defines the measurement-based process development platform has suggested by Basili and green[31]. Figure 2.7 shows the GQM paradigm.

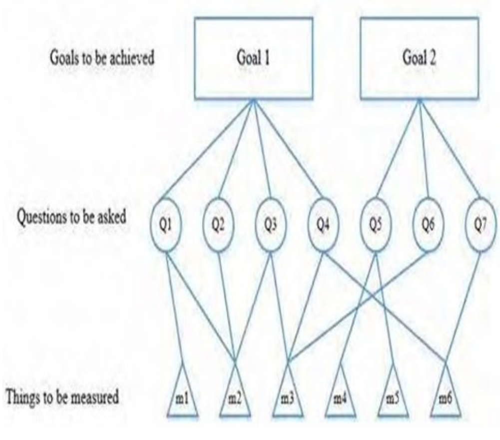

Figure 2.7 GQM Paradigm

The GQM paradigm is utilized in the process improvement to assist answering the following 3 critical questions:

1. Why we are presenting process improvement measurement?
2. What information do we need to help measurement of improvements?
3. What are the products and process measurements essential to deliver this information?

The above questions are directly related to the ideas namely, goals, queries, and metrics in the GQM paradigm: They are

1. **Goals:** A goal is accomplishment of something that the suggestion demands. It deals with how the process impacts the organization or products or may not be connected directly with the process attributes. For instance, goals may be better-quality level based on process maturity, increased product reliability or smaller product development time.
2. **Questions:** These goals are recognized where specific areas of uncertainty are linked to the modifications. Normally, a goal needs to be answered by a number of associated questions. For instance, questions related to the goal of restrictions in the product development intervals may be "How can the time necessary to decide product requirements with businesses be reduced?", "How various examinations are operative in determining product faults?" and "Where are the bottlenecks in our current process?"
3. **Metrics:** These measurements are used to check whether or not process improvements are required to help answering the question related to the predicted goal. This needs time to collect data, to implement each of the process activity, get information on the number of defects per test run and communication between customers for every change in requirements.

The benefit of the GQM method is used for dividing the organizational concerns and goals in the process improvement from detailed process concerns. It deals with the basis for defining the data to be studied and collected in various ways and based on the question it is expected to answer.

2.3.1.2 Process analysis

The study of process analysis is used for help in and appreciation of key characteristics of the processes by the people involved in the practice. Process analysis has a number of closely associated objectives:

- To know the relationships between the measurements and process actions.
- To know the relationships between these activities and the action involved in the process.
- To relate the process of examination of comparable processes as precise processes or similar type of ideal processes seen elsewhere in the organization. The most used techniques of process analysis are:
- Ethnographic studies: Process contestants are used as human activity to recognize the nature of software development detected as they work. Such type of analysis exposes refinements and difficulties which may not be exposed by surveys and discussions.
- Questionnaires and interviews: In a project, managers and software engineers work on what truly goes on interrogated. Suitable questions are developed for the answers from those involved in the process during personal interviews.

2.3.1.3 Process Change

It includes the existing process by making alterations. This can be determined by development goals during the integration reduction in the number of defects exposed by 2.5%. Process measurements are used for the measurement of the efficiency of the changes after the changes have occurred. The process change has 5 key phases which are shown in figure 2.8.

1. **Improvement identification:** This phase uses the outcome of the process analysis for tackling cost inadequacies, quality issues, schedule blocks to determine the ways that have been observed during the process analysis. Reports on the process problems are used as processes, tools, methods and new process structures. For example, an enterprise sees requirement of problems that many of its software problems stem from. With the help of best practice guide and based on the several requirements of engineering [32] practices can be changed or presented and may then be recognized.
2. **Improvement prioritization:** This phase for implementation is concerned with the estimation of the conceivable changes to the process. When

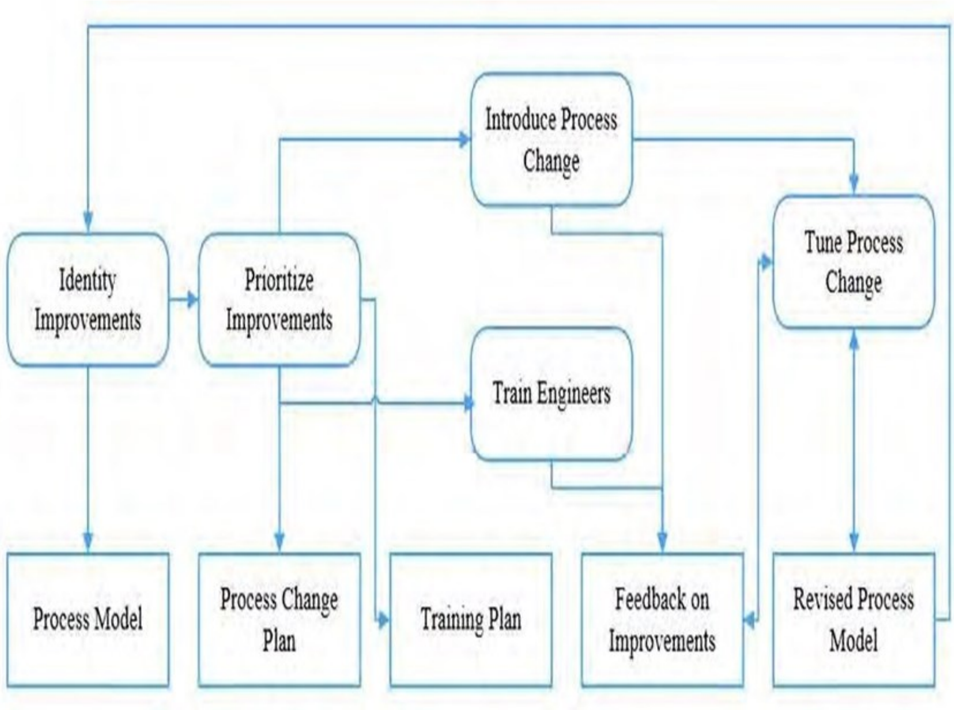

Figure 2.8 Process Change Steps

introduced all at once, it is necessary to decide on which are the most important. Decisions based on the requirement are needed to improve process areas, the impact of a change on the organization, the costs of introducing a change, or other factors. For instance, an enterprise may consider the overview of requirements management processes for managing the developing requirements as the peak priority process change.

3. Process change introduction: This phase brings in new methods, procedures, and tools into place and incorporates them with other process activities. Enough time should be allowed to present changes and confirm that these changes are compatible with other process activities and structural standards and procedures. This may include acquiring tools for requirement management and designing processes for utilizing these tools.

4. Process training: Without process training, it is impossible to reap the full benefits of process changes. Software engineers involved need to know the changes which have been suggested and how the new and changed processes

Usually, process changes are executed without sufficient training and the effect of these changes is to worsen rather than increase product quality. In this case of requirements management, a description of the process activities and an overview to the tools that have been recommended, the process training might include a discussion of the charge of requirements management.

5. Change tuning: In this stage, the proposed process changes will never be entirely operative very soon after introduction. It requires a tuning phase where minor difficulties can be identified, and modifications to the process can be comprehended and introduced. This tuning stage may last for several months until the development engineers are satisfied and happy with the new process.

2.3.2 Staged CMMI Model

CMMI valuation includes observation of the processes in an organization and rating these processes or process areas on a six-point measure which relates to the level of maturity in every process area [33].

2.3.2.1 Level 0 Incomplete

At least one of the precise goals related with the process area is not fulfilled. There are no generic goals at this level as institutionalization of an inadequate process does not make sense.

2.3.2.2 Level 1 Performed

The goals related to the process area are fulfilled, and the scope of the work to be achieved is explicitly set out for all the processes and connected to the team members.

2.3.2.3 Level 2 Managed

In this level, the goals related to with the process area structured and organizational policies which are put in place which indicate when each process should be used. There should be documented project plans which define the project goals. Process monitoring procedures and resource management should be in place across the organization.

2.3.2.4 Level 3 Defined

This stage focuses on organizational deployment and standardization of processes. Every project has a managed process which is modified to suit the project requirements using a distinct set of organizational processes. Process assets and process measurements should be collected and used for upcoming process improvements.

2.3.2.5 Level 4 Quantitatively managed

At this stage, there is an organizational concern to utilize statistical and other quantitative approaches to control sub-processes. That is, collected process and product measurements should be used in process management.

2.3.2.6 Level 5 Optimizing

This is the highest level of the CMMI model staged. Here, the organization should utilize the product and process measurements to derive process improvement. Trends must be examined and the processes modified to suit changing business requirements.

2.4 CMMI Appraisal

Activity appraisal involves a study of how closely the processes that help categorizing the positives and negatives of organization that are related to CMMI best practices, in order to find their efforts on business development and the attainment of capability level or maturity level of standards based on the eventually of earning by directing an appraised for many organizations to find the value in determining their performance and capability.

Appraisals normally lead to:

* The determination of where the development can be made on areas identifies and how well the processes organizations is compared to CMMI best practices.
* The comparison between how well is the processes of organization and CMMI best practices in the information relating to the external suppliers and customers.
* Meeting consumer predetermined necessities.

2.4.1 Benefits of Appraisal

The activities of appraisal can support the organization at any point of CMMI implementation. It enables effective and rapid improvement efforts. It ensures the most effective and efficient improvement results of CMMI journey as a proven by designed to support the CMMI appraisal method. The focus is on performance development that delivers consistent, actionable, reliable and clear result, improving the capability and building the impact on the business.
An appraisal allows the organization to:

- Prioritization of the development of business performance and design a strategy for the organization.
- A Moderate risk for improvement, transfer, observing acquisition based on service and product.
- Establishment of the reliability of processes to stakeholders and make available the Published Appraisal Results (PARS) through use of appraisal results.
- Representation of how well the processes of organization determines the CMMI levels conform to the CMMI.

2.4.2 Method of CMMI Appraisal

CMMI v2.0 is used for the estimation of an organization process defined as the official appraisal method by CMMI institute. It provides ratings relating to performance and organizational capability. The methods of CMMI appraisal are comparative to CMMI v2.0 model that delivers activities for conducting appraisals and a set of processes. The clear design of product of CMMI v2.0 to a broad variety of organizations, types of work and markets designed for availability and flexibility can be seen. This enables easier, faster and more successful adoption. Others are:

- Performance development
- Specific needs of the industry
- Project size organization
- Marketplace drivers

- ♦ Leanings (e.g., occupational, business)
- ♦ New or varying knowledge

2.4.2.1 Method features and purposes

The method of CMMI appraisal is used for categorizing process determination and process implementation as positive and negative. The model adoption is linked to established relative business performance. It is based on the types of legacy methods of appraisal that can include the best practices. A common, integrated appraisal method.

The new appraisal method comprises of supportive appraisals in varied circumstances:

- Benchmarking
- Interior presentation and process development
- Development observation
- Provider assortment
- Reduction in Risk
- An effective appraisal method.

It is implemented for capable performance subject to practical performance constraints.

Accurate results

This method provides reliable and precise results of delivery based on appraisal goals.

A effort on process operation

This method highlights finding performance gaps and challenges that certify a collaborative, through a confidential method, by directing personnel valuation against process implementation.

2.4.3 Types of CMMI Appraisals

The CMMI V2.0 Appraisal Method Definition Document describes four types of CMMI V2.0 Appraisals: 1. Benchmark, 2. Sustainment, 3. Evaluation, and 4. Action Plan Reappraisal.

Each type is designed for providing findings that describe the strengths and weaknesses of the organization's processes based on CMMI best practices. Familiarity with each type helps making good decisions about which appraisal is appropriate for the organization for meeting business objectives and fostering process improvement.

2.4.3.1 Benchmark Appraisal

Identifies opportunities for organizations to improve implementation of processes and their overall business performance.

2.4.3.2 Sustainment Appraisal

Appraisal “check-up” done at the end of two years following a Benchmark Appraisal to ascertain the organization is maintaining the appraisal level.

2.4.3.3 Action Plan Reappraisal

A “second-chance” for organizations that has narrowly failed to achieve the targeted appraisal level in a previous appraisal.

2.4.3.4 Evaluation Appraisal

An informal and flexible approach used for helping organizations in the preparation for an appraisal and determination of opportunities for improvement.

2.4.4 CMMI Levels of Capability and Performance

In an organization, maturity or capability level delivers a method for the description of its performance and capability. The focus on controllable and ordered number of practice areas at a time has shown the organization’s experience in doing its best for the process development efforts.

A Culture based on Continuous Development

The goals of process development are based on business purposes. The benefits of the organization have shown the experience from attaining the level. The focus of development is on shared objectives, performance outcomes and business. When the result of performance happens as one in the normal course, and certainly tolerable, the focus is on the accomplishment of the improved performance and business objectives.

2.4.4.1 Capability Levels

These levels can apply to individual practice area in the performance of an organization and process development achievements. These practices are classified into practice groups within the practice areas which can be provided for performance improvement of an evolutionary path labeled from level 0 to level 5. Each of the levels can be constructed by adding new objectivity or functionality on the prior levels ensuring improved capability.

1. **Capability Level 0: Incomplete**

* Incomplete process of meeting the determination of the Practice Area.
* May or may not be meeting the determination of any practice.
* Unpredictable performance.

2. Capability Level 1: Initial

* Initial method for meeting the determination of the Practice Area.
* Not a comprehensive set of practices for meeting the complete determination of the Practice Area.
* Speaks of the performance problems.

3. Capability Level 2: Managed

* Includes level 1 practice.
* Simple, but comprehensive set of practices that refers to the determination of the Practice Area.
* The use of the organizational resources are not involved.
* Categorizes and displays the development towards project performance purposes.

4. Capability Level 3: Defined

* Constructs the practices on level 2.
* Uses structural standards, work features and modification to address the project.
* Projects that use and contribute to organization assets.
* Focuses on attaining both organizational and project performance purposes.

2.4.4.2 Maturity Levels

It characterizes the effort of process development for an organization that stage paths based on predefined sets of practice areas. The predefined set of process area within each maturity level also provides a track for performance development. Each maturity level is constructed through addition of new consistency or functionality on the preceding maturity levels.

5. Maturity Level 0: Incomplete

Work gets competed but is often delayed and exceeds budget.

6. Maturity Level 1: Initial

The work schedule gets delayed frequently with a cost over-run but the job gets completed.

7. Maturity Level 2: Managed

Projects are scheduled, achieved, restrained and measured.

8. Maturity Level 3: Defined

Values of organization-wide deliver the direction across plans, packages, and collections.

9. Maturity Level 4: Quantitatively Managed

Organizations that are known for measurable performance development and meet the requirements of external and internal shareholders.

10. Maturity Level 5: Optimizing

Organization is constructed to pivot and motivated on continuous development that responds to variations and chances. The stability of organization delivers a platform for speed and innovation.

2.5 Summary

This chapter provides a description of the characteristics of effective processes of CMMI evolution based on CMMI-Development, CMMI-services and CMMI - Acquisition process area and the specific goals and practices to help the process area in achieving the associated specific goals. There is also the description of the generic goals and generic practice that must be present to institutionalize processes for implementation of a process area in achieving generic goals. This chapter has dealt with CMMI for evaluation of the organizational maturity and process capability. These practices form a baseline which supports decisions when process improvements have been effective. An appraisal activity that can help organization at any stage of the CMMI adoption has also been indicated. CMMI appraisal method, types of CMMI appraisals, CMMI levels of capability and performance and published appraisal result system have also been described.

CHAPTER – 3
LITERATURE REVIEW

3.1 Overview

This chapter discusses the quality of software service or products of Software Process Improvement (SPI) aim to help exploitation of the benefits of practices followed by Small and Medium Size Enterprises (SME). Changes of these types in continual process and advancement of novel practices have been combined for managing the activities in the process of software development. SPI gives attention to the weakness of current practices and organization's software requirements. The outstanding quality models, namely, six sigma, ISO 9001, Capability Maturity Model Integration (CMMI) and Software Process Improvement and Capability determination (SPICE) have been referred to. CMMI helps in setting process improvement goals and priorities, provide a reference for quality processes and for appraising current processes. Process Maturity acts as a catalyst for the organization's success in meeting business objectives. The framework groups best practices into Constellations. There are 3 constellations: 1. CMMI-Development, 2. CMMI-Acquisition and 3. CMMI-Services. This chapter presents a proposal towards a Process Capability Profile Driven Process Engineering (PCDE) as an evolution of the current Software Process Improvement based on Process Capability (and Maturity) Models.Software industry has been using software process improvement approaches, based on the maturity levels of SW-CMM and CMMI staged models, for improving business. The survey of the research work carried out in this chapter is based on CMMI-Based Software Process Maturity on Software Schedule Estimation.

3.2 Small and Medium Sized Enterprises (SMEs)

SMEs are not the "smaller versions of large firms" and have their own unique characteristics [34][35]. This section starts with looking at the definition of SME's. It then discusses on what makes SMEs unique to provide basic knowledge relating to the basis for the rest of the chapter.

3.2.1 Definitions of SMEs

There is no standard definition of 'SME'. Generally, there are two approaches to defining SMEs, namely,1. Quantitative and 2. Qualitative. Quantitatively, SMEs can be defined on the basis various factors such as the number of employees, working capital or annual turnover or a combination of two or more factors. The Australia Bureau of Statistics defines an SME as a business employing less than 200 employees. In Vietnam, the definition of SMEs is one with 1-100 employees (Government Decree 90/2001/ND-CP). In some countries such as China, Thailand and Singapore, the definition varies among different industries. For instance, in Thailand, to be considered as a small business, the number of employees for that business should not exceed 50 (Production and Service), 25 (Wholesale), and 15 (for Retail) and Annual revenue is not more than 50 million bath (approx. $AU 1.6 million).

As noted by Burgess, [36] there also exist issues using the number of employees as the only criterion. These include issues relating to how to count part- time staff, multiple business ownership (one owner with many different SMEs), or the need to take into account different industries have different needs (for example, construction businesses require more staff than retail businesses). SMEs can also be defined qualitatively on the basis of a combination of three factors: (1) having a small market share (in its own market), (2) being personally managed by their owners and (3) being independent (not being part of a larger business, with the owner-managers being independent in making decisions)[37]. Other categories that relate to 'small businesses' are 'micro' businesses (very small businesses) and 'medium' businesses (usually larger than small businesses but not as big as large businesses). In literature, these terminologies (i.e. small businesses and SMEs) for categorizing businesses are different fromthese relating to large organizations used interchangeably. Therefore, the researcher uses both small businesses and SMEs as keywords in the search for relevant literature reviewed in this work.

The use of the number of employees for the determination of the business size, as it is easier for the researchers to either search for that information from the database or ask organizations for their size when carrying out a study [38]. For the purposes of this study, the Vietnam definition of SMEs is used, whose details are:

A micro business is any business having 1 to 10 regular employees. A small business is any business having 11 to 50 regular employees.

A medium-sized business is any business having 51 to 100 regular employees. Therefore, SME is any business with 1-100 full-time employees.

3.2.2 Characteristics of SMEs

SMEs have been recognized as different from large businesses[39]. Researchers have identified many unique characteristics of SMEs which suggest that "a small business is not a little big business"[40].

Differences exist in many forms. Size, which also varies among SMEs[41], is not the only factor differentiating SMEs from larger ones. In addition to size, [42] suggest the presence of three perspectives which can be used differentiating SMEs from larger ones.These include task environment, organizational configuration, and managerial characteristics. The task environment consists of customers, suppliers, competitors, and regulatory bodies. Organizational environment is about the formal and informal structures of the organization. Managerial characteristics refer to the motivations, goals, objectives, and actions of the owner-manager [42]. One difference between small and large businesses, which comes from the business itself, is about the owner/manager. In SMEs, the owners/managers, who contribute most or all of the capital, have a strong influence in business dealing due to the role they played in the direction of the business [43,44,45]. Thus, their goals, operational abilities, management abilities and strategic abilities have a directeffect on not only the operation of their businesses but also the culture and atmosphere of the organizations. Further, other characteristics of owners/managers such as biographical characteristics (personalities, emotions, values, attitudes, abilities, perceptions and individual learning styles) also play a major role as they have effect on the decisions made by the owners/managers [46]. In addition, in SMEs, the owners/managers make most of the critical decisions [47].

Thus, in SMEs, decision making is generally centralized and the power of control lies with the owner/manager. As a consequence, SMEs are operated and managed in a personalized manner [48]. However, as there are fewer management layers and decision makers in SMEs, the decision-making process is often quicker [49]. Hence, this might become an advantage for SMEs in adopting/implementing changes such as KM practice in the organization.

SMEs also face difficulties that are seen in large businesses in planning, attracting, recruiting, training, retaining, and developing human resources, especially qualified staff[50][51]. This is mostly due to financial constraints, short-range management perspectives, and limited career path [52]. They tend to employ generalists rather than specialists and often rely on family labour. This may result in low productivity, high level of absenteeism, high rate of staff turnover and low level of job satisfaction [53].

In addition to scarcity of human resource, many researchers also point out to SMEs normally lacking in financial resources [54,55,56]. They do not have enough funds for necessary investment in human resources, marketing and information systems [57]. This is further compounded by difficulties in obtaining external financing, either for growth or other reasons [58].

SMEs tend to have simpler management structures – that is 'flat organizational structure' [59], 'one unit management' [60], or 'highly centralized structure' [61] is another feature associated with SMEs [62]. A simple structure leads to effective communication practices, informal face-to-face channels of communication, and direct supervision. Moreover, a flat organizational hierarchy allows owners/managers for easy keeping updated with daily business activities; and hence, quicker decisions are made [63]. In addition, most SMEs have simple planning and control systems [64][65]. Activities and operations in SMEs are governed less by formal rules and procedures with low degrees of standardization and formalization. Therefore, they are more adaptable than the larger ones, especially in implementing new technologies.

Due to their small size, SMEs normally have a unified culture which is commonly shared among a few interest groups. In general, their culture is more organic and fluid than that of any larger one. Employees are more likely to get linked to a commonly shared value and belief which influences their actions and behavior. In addition, small business culture is affected and shaped by the personality and outlook of the owner/manager. This can create either advantages or disadvantages for SMEs in adapting to changes. In summary, SMEs are characterized by resource scarcity [36]; the strong influence of Owners / Managers [66]; flat organizational structure, systems, processes and procedures and their culture and behavior[63].

3.3 Software development

Software development is a conceptually complex task [67]. Different quality assurance methods and software development methodologies are used for the achievement of bug free, reliable and high quality software [68]. Usually, this process is not just one monolithic block of work that takes some ideas about the application to be developed as input and producing a perfectly fitting solution as output. Well- defined boundaries and meanings are achieved by dividing the process into a set of basic actions. These activities are aimed at understanding the problem, carrying out the plan, examining the result for accuracy, planning the solution and correcting the possible inaccuracies or errors. Software development process involves the following:

Requirements engineering: It aims at the tasks that determine the needs.
Design: It aims at the creation of the specification of the artifact.
Implementation: Process that translates the problem into executable forms.
Testing and evaluation: It is conducted to provide information about the quality to the stakeholders.
Deployment: It makes the system available for use.
Maintenance: it is the modification of a software product after delivery to correct faults.
Evolution: It aims at updating the developed software over time, providing new inputs to the development models in the form of new requirements.

3.3.1. Software Development Process

Software development is a creative process leading to an innovative software product or system. Generally, this process is not a single monolithic block of work which is considered as input, a few ideas regarding the application are developed and produced as output with completely appropriate results. The process can be decomposed into a set of basic activities with distinct boundaries and significance. These activities focus on empathising the problem, planning a result, executing the plan, investigating the result for precision and determining possible errors or defects. The process of software development transforms necessaries into a software system.

Figure 3.1 Software Development Process structure

The figure of above indicates development and shows a few refinements also. First, these processes feed one or more individuals. The initial step in the development is to excerpt and form these concepts into a certain number of the necessaries. As the second step, the outcome of a development method is more than just software. In addition to computer sequences, the output process of the development includes user guides, program documentation and test

test belongings. User guides illustrate the use of the software. Program documentation and test cases help designers maintain and prolong the software in the future.

Software Process

This refers to the series of steps a person or organization follows in the production of a software system. A software process is the process used by an individual or organization on a specific project.

Process frameworks

The environment of software development leads to the occurrence of a narrow process that will be the best for all development processes adaptive in all surroundings as unlikely. Instead, organizations and individuals approve a method that is recommended especially for their specific needs. The Rational Unified Process (RUP) is possibly the best known and is the most widely used framework of the software development process.

Software Life Cycle

This phase of a software project goes through over time on the basis of analysis, design, and implementation, etc.

Software Development Life Cycle Model

Abstract models are used for a combination of software developments with collective characteristics. Some of the standards used for the process of software models distinctive are: 1. Timing between phases, 2. Entry and exit standards between phases and 3. The artifacts produced through each phase. The process of software model relates to development of the software as an area cover is to play basketball, both defining high-level features of performance.

3.3.2 Software Development Life Cycle (SDLC)

In any software organization, the process of SDLC is proposed for the improvement of a software project. It contains a definite plan illustrating the development, replacement, maintenance and enhancement or alteration to the particular software. The entire development process and also for improving software quality the methodology of life cycle is defined. The following are the stages in a typical SDLC.

3.3.2.1 Planning and Requirement Analysis

The fundamental and essential stage of SDLC is the requirement analysis. It is accomplished by a senior person in the team with inputs from the client, market surveys, sales department and domain experts in the industry. Data of this kind helps planning the general project approach and also the study of the performance of product feasibility in areas like technical, economical and operational. In this planning stage, quality assurance requirement is planned and recognition of risk within the project is also done. The technical possibility of study output describes the different technical approaches which help successful implementation of the project in consideration of minimum risks.

3.3.2.2 Defining Requirements

After the completion of the required analysis, the next stage is defining the requirement for a clear definition and documentation for the required product and get approval from the market analyst or client. This work is done through the Software Requirement Specification (SRS) document that contains details of the entire product requirement to be designed and developed during its project life cycle.

3.3.2.3 Designing the Product Architecture

The next stage is designing the product architecture where SRS is used as reference for product architecture to come out with the best architecture of the product which requires improvement. Based on the specified requirement in SRS, generally two or more design approaches are proposed and documented for product architecture using Design Document Specification (DDS).Time and budget constraint are reviewed from all stakeholders on the basis of parameters that include risk assessment, design modularity and product robustness using DDS. These approaches help in selecting the best design for the product. This design approach helps getting a clear definition of the entire architecture modules of the product along with communication and representation of data flow with a third party and external modules. Similarly internal design of the entire module of the proposed architecture should be distinctly defined in DDS with the minutes of the details.

3.3.2.4 Building or Developing the Product

Development of the product is the stage where the actual development begins and the product is built. Codes are generated in this stage on the basis of DDS programming. The generating code can be proficient without much difficulty, when the performance of design is in an organized and a detailed manner. The developer should follow the guidelines of the coding defined by programming tools like debugger, compilers and interpreters etc and their organization for generating the code. High level programming languages such as Pascal, C, C++, Java and PHP are used for coding. The programming language is chosen with due reference to the type of software being developed.

3.3.2.5 Testing the Product

Testing the product is the stage which is generally a subset of the complete stage as in the recent SDLC models with the activities of testing frequently involved in all the stages of SDLC. So this stage specifies only the testing of product, where defective products are tracked, reported, fixed and retested till the product reaches the standard quality defined in the SRS.

3.3.2.6 Deployment in the Market and Maintenance

After the product is tested, it is ready to be deployed and released formally in the appropriate market. Based on the business strategy of the organization, this is the occasion for product deployment. The product may be first released to a limited segment and tested in User Acceptance Testing (UAT) in the real business environment.

3.4 Software Development Effort Estimation

Capability for consistent and accurate estimation of the effort of software development particularly during the earlier stages of the development life cycle is required for planning and conducting software development from the project manager. Several estimates are involved in the effective management of the software Effort. Estimation has become essential for all community to improve a valuable model which makes effort for a good estimate. In software effort estimation, Putnam model is an empirical model introduced by Lawrence H. Putnam. This model helps estimation of the associated effort by providing size details [69]. The most widely used algorithmic model is the Constructive Cost Model (COCOMO) developed in 1981 by Barry Boehm and the community of software engineering has been using the model till now [70] [71]. Siew Hock Ow and Iman Attarzadeh have introduced ANN-COCOMO which is based on neural networks software estimation. In this model, the precision of effort estimation which can be established and the cost of estimation suggested that is very close to the actual effort [72]. Estimation of the time, effort and cost needed to advance software is done using various models of effort estimation, namely COCOMO model, Putnam model, ANN COCOMO etc. In an organization this is area that has several challenges and researchers have improved models to estimate the effort needed to build a project.

3.4.1 Description of the effort estimation process

Effort estimation technique discusses the approved manner that has come up with the usage of effort estimations. The development consists of steps essential for creating the estimation and enchanting description information from previous projects. If there are no events in place, the output can be inaccurate [73]. The basic level of effort estimation can be well-defined in terms of efforts and output it produces. Between the inputs and outputs, dissimilar types of resources are functional to the inputs using the effort estimation technique for generating the output. This estimation method at this type of basic level is presented in figure 3.2, following the definition given by [74]. Estimates made as the outcome of the effort estimation method are created on the inputs. In the estimation process, the inputs are the purposes of the estimation and the information that is used while exploiting the estimation process. This data can be both measureable and the capability of the individuals while doing the effort estimation. Both kinds of data might come from earlier projects or other development actions. The volume of information that can be employed is not the only aspect that contributes to the closeness of the approximations. The value of data is also significant. A huge dataset is not essentially more improved than a smaller one, as there are other issues that affect the quality of data, namely, its redundancy, significance and how up-to-date it. Technical fluctuations or other abrupt changes can rapidly render the collected data useless.

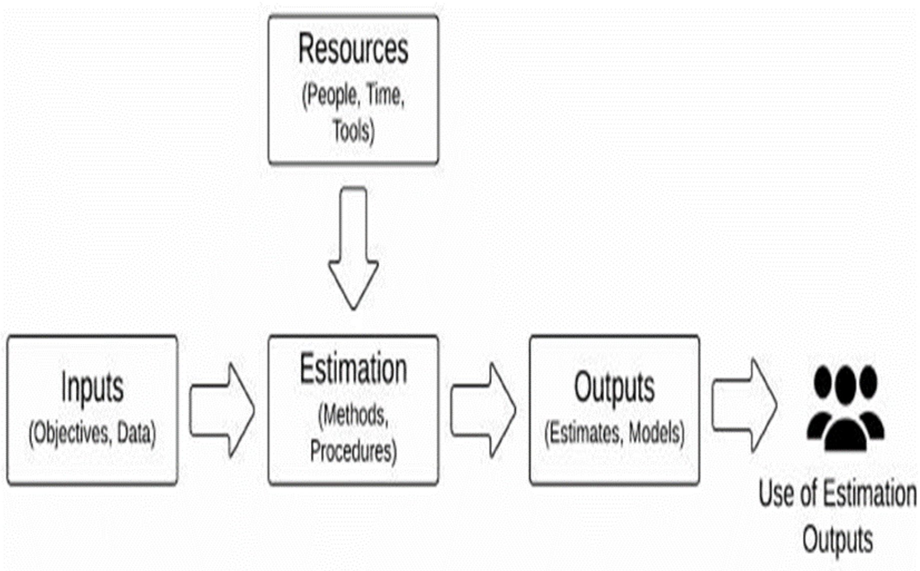

Figure 3.2. Elements of effort estimation process

Estimation in figure 3.2 denotes how the estimation is completed in the project, importance of the kind of effort estimation approaches that are realistic to the inputs [74].

Resources refer to and implements that are convoluted in effort estimation.

Also involved is the time that the people pass doing the effort estimation [74].

An output from the process of effort estimation come through the estimations themselves. If the method of effort estimation used is a method of model-based, another output is the model. In such a case, the representations can then be used in estimating other projects or development activities. These subsequent models characterize the relations between the work, effort and the disturbing characteristics existing in the work environment [74].

Context of estimation should found a place in the description of the estimation approaches used or models by regulating them to fit to the framework. For instance, the past data in the existing environment which is not good, should be taken into explanation when doing new effort estimations, as in that kind of a case it is not conceivable to use approaches that need data from past developments [74].

3.4.2 Categorization of effort estimation methods

Effort estimation methods that can be assembled are composed into distinct groups founded on the different features of the approaches, but this does not conform toone commonly acknowledged categorization. In their study, association of dissimilar reviews concerning the choice of an effort estimation technique is problematic, as education classifies the approaches inversely [75]. In their study, they grouped the approaches into three categories: expert judgment-based methods, model- based methods and "other", which contained dissimilar types of grouping methods that were not severely speaking effort estimation approaches. Others have also considered effort estimation methods to three groups, but there are frequent variations in the groups. The three categories are: 1. Expert judgment, 2. Algorithmic models and 3. Analogy systems [76]. Later some researchers have higher machine learning to be the third major group [77].

3.4.3 Expert estimation methods

Expert estimation methods (or expert judgment-based methods, expertise-based techniques, expert-based effort estimation) are created on the basis of the knowledge and experience of the specialists working on the software to offer the approximations. If there is no measured experimental data that is essential by additional methods, expert estimation-based methods can still be used. The drawback of these approaches is that the estimates are created on the basis of the opinion of the estimator(s) and, even, when a person has much experience, this does not mean that his or her approximations are essentially accurate. This method has defined an expert as a person who has the competency to estimate the software development effort, such as a professional software developer. Although the word expert is used for defining the software professional who does the effort estimation, it does not necessarily mean that the task under estimation is within the expertise area of the estimator [78].

The advantage of experts estimating the effort compared to formal methods is that as human beings they have more information available and can use it more flexibly than algorithms [79].

Some of the estimation models are mentioned below.

3.4.4 Estimation Models

- Putnam effort estimation model.
- COCOMO estimation model.
- ANN-COCOMO based software estimation model.

3.4.4.1 Putnam effort estimation model

Putnam model is the most familiar model for software effort estimation. This model form is expressed as follows:

Technical constant $C = SIZE * B^{1/3} * T^{4/3}$

Total person Months $E = \frac{1}{T4} * (size / C)^3$ . . . 3.1

Where,

T is the Required Development Time in years

Size is estimated in LOC

C is a parameter reliant on the development environment and it is resolved on the basis of historical data of the precedent projects.

Rating: C=2,000 (poor), C=8000 (good) C=12,000 (excellent).

The Putnam model is highly sensitive to the development in time, as it decreases as the person-month required for development gets increased. The major problem with this model is that it is based on the knowledge or capability for a fair estimation but the size of the software has to be developed. Uncertainty seen in the software size the effort causes inaccuracy in the estimation [80].

3.4.4.2 COCOMO estimation model

The COCOMO model which has been improved by Barry W. Bohemand is a product of software cost estimation. This model contains three hierarchies to enhance accuracy in forms. The three hierarchies are 1. Basic, 2. Intermediate and 3. Detailed, where basic COCOMO is the first level that is good for software effort of early, quick and rough order of magnitude estimates and costs but its accuracy is limited. But in the case of intermediate COCOMO, cost driver is considered and, in detailed COCOMO, further accounts for the influence of particular project phases.

3.4.4.2.1 COCOMO Model

Constructive Cost Model (COCOMO) is broadly used as an algorithmic software cost model. Boehm has proposed COCOMO [81]. It has the following hierarchy

Model 1 (Basic COCOMO Model)

COCOMO model computes the software development effort and cost as functions of program size expressed in estimated lines of code (LOC) [82]. The basic steps in this Model are: -

* Obtain a first estimation of the development effort from the estimation of thousands of delivered lines of code (KLOC).
* Determine a set of 15 multiple factors from various attributes of the project.

Adjust the estimation effort by multiplying the early estimate with all the multiplying factors. The early estimation (also called nominal estimate) is determined by an equation of the form used in the static single variable models, using KLOC as the measure of size. To determine the initial effort in person-months are expressed in the following equation,

$$EFFORT = a^* (KLOC)^b \qquad ...\ 3.2$$

The values of constants a and b depend only on the project type. It has the following three modules of software projects.

Table 3.1 Model 1 (Basic COCOMO Model)

Development Mode	Basic Effort Equation	Time Duration (TDEV)
Organic	Effort = 2.4 KLOC $^{1.05}$	TDEV = 2.50 * (PM) $^{0.38}$
Semi Detached	Effort = 3.0 KLOC $^{1.12}$	TDEV = 2.50 * (PM) $^{0.35}$
Embedded	Effort = 3.6 KLOC $^{1.20}$	TDEV = 2.50 * (PM) $^{0.32}$

Model 2 (Intermediate COCOMO Model)

Intermediate COCOMO Model computes software development effort as a function of the program size and a set of cost drivers that include subjective assessment of the hardware, products, project and personnel attributes. This model is prolonged to consider a set of cost driver attributes that can be clustered into four major categories.

A. Product attributes

- The product Complexity
- Application data base Size
- Software reliability required

B. Hardware attributes

- Required for turnaround
- Memory constraints
- Execution-time performance constraints
- Volatility of the virtual machine environment

C. Personnel attributes

- Applications experience
- Software engineer capability
- Virtual machine experience
- Analyst capability
- Programming language experience

D. Project attributes

* Software engineering application techniques

Each of the 15 attributes required for the development schedule is rated on a 6 point scale that varies from lower to extra higher (in importance or value). Depending upon the rating, an effort multiplier is determined from a table published by Boehm, and the product of all the effort multipliers results is an Effort Adjustment Factor (EAF). Typical values for EAF range from 0.9 to 1.4.

Software tools used

The intermediate COCOMO model takes the form:

$EFFORT = a * (KLOC)^b * EAF$ . . . 3.3

Where the effort in person-months and KLOC is the estimated number of delivered lines of code for the project.

Table 3.2 Model 2 (Intermediate COCOMO Model)

Development Mode	Intermediate Effort Equation
Organic	$Effort = 3.2 * (KLOC)^{1.05} * EAF$
Semi Detached	$Effort = 3.0 * (KLOC)^{1.12} * EAF$
Embedded	$Effort = 2.8 * (KLOC)^{1.20} * EAF$

Model 3 (Detailed COCOMO Model)

The detailed COCOMO Model combines all the characteristics of the intermediate version with an assessment of the cost driver's impact on every step (analysis, design, etc) of the software engineering process.

3.4.4.2.2. COCOMO II model

It is a collection of three variants, namely, 1. Application composition model, 2. Post architecture model and 3. Early design model. COCOMO II is an extension of intermediary COCOMO model [83] and defined as:-

$$EFFORT = 2.9\ (KLOC)^{1.10} \qquad \text{......}\ 3.4$$

3.4.4.3 ANN-COCOMO based software estimation

The neural network based software estimation is ANN-COCOMO which is customized from the post architecture model of COCOMO II .In the neural network, there are five input scale layers which correspond to the Scaling Factor (SF) and also Effort Multiplier (EM) with bias 1 and bias 2 values. This model is not entirely a connected network but some particular hidden layer nodes are considered in contributing account of both EM and SF separately.

3.5 Structure of CMMI

CMMI builds upon three key concepts: process areas, goals, and practices. Figure 3.3 illustrates the interaction between these structural elements. CMMI identifies 25 so-called process areas in the development process [84]. Each process area defines a set of so-called specific goals and a set of specific practices that serve to fulfill the goals. Concerning process areas, it has to be pointed out that CMMI's process areas will most likely not map one-to-one on the processes of a certain organization. Thus, it is vital to determine the best mapping of processes to CMMI's process areas. This is a matter of interpretation. In the models, although process areas depict the behavior that should be exhibited in any organization, all practices must be interpreted using an in-depth knowledge of the CMMI model being used, the organization, the business environment, and the circumstances involved [85]. As mentioned above, specific goals and practices are defined by process areas. However, there is another kind of goals and also practices. The so-called generic goals and generic practices are equivalent to the specific goals and practices, with the exception that they are not specific to a certain process area. They are of concern to more than one process area. It is also worth noting that all practices that are meant to be performed for achieving a certain goal are sequentially ordered. As an example, consider the process area Requirements Management (REQM). It defines single specific goal Manage requirements.

The practices for this goal are:

1. Obtain an understanding of requirements
2. Obtain commitment to requirements
3. Manage changes in requirements
4. Maintain bidirectional traceability of requirements
5. Identify inconsistencies between project work and requirements

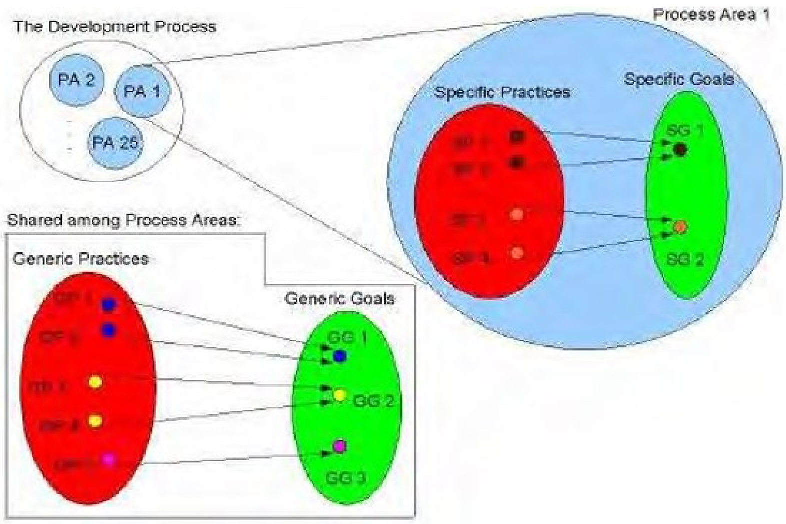

Figure 3.3 The structure of CMMI

It should be clear that no one can obtain a commitment to requirements that are not understood.

3.5.1 Need for CMMI

CMMI is about improving performance through improvement to operational processes. In particular, its improving processes are associated with managing how organizations develop or acquire solution-based wares and define and deliver their services. The CMMI has provided significant values for many organizations that have used it as a guide for improving the way they do their engineering work. It has helped them to gain control over their processes – Management, Engineering and Supporting processes – to ensure those processes serve the needs of the organization. Following the guidance of the CMMI, an organization can put them on the road to more effective processes to get the ability to achieve more consistent success in engineering projects. Successful process improvement initiatives must be driven by the business objectives of the organization. For any software development organization, the key factors for success should be delivery of the product / project on time and within the budget maintaining the quality and fulfilling the functional, non-functional requirements. CMMI covers all the process areas in such a way that ensures this. All the twenty two process areas can be grouped into four major categories: Process Management, Project Management, Engineering and Support [86],[87],[88],[89],[90],[91],[92].

3.5.2 CMMI Maturity Level

Maturity level is a well-defined evolutionary plateau toward achieving a mature software process. Each maturity level provides a layer in the foundation for continuous process improvement.

3.5.2.1 CMMI Level 1

It is characteristic of processes at this level that they are (typically) undocumented and in a state of dynamic change, tending to be driven in an ad hoc, uncontrolled and reactive manner by users or events. This provides a chaotic or unstable environment for the processes.

3.5.2.2 CMMI Level 2

CMMI Level 2 is the second of the five maturity levels in the staged representation of the CMMI. It's known as the managed level when the projects of an organization have ensured that requirements are managed and that processes are planned, performed, measured, and controlled, then that organization will be appraised as CMMI Level 2 or Managed level. CMMI Level-2 has the following process areas to conform:

1. Requirements Management
2. Project Planning
3. Project Monitoring and Control
4. Measurement and Analysis
5. Supplier Agreement Management

6. Process and Product Quality Assurance
7. Configuration Management.

Figure 3.4 CMMI Level 2 Process areas

Each process area has one or more specific goals. Each specific goal has one or more specific practices. In CMMI Level- 2, each process area has a single generic goal that contains generic practices. Generic goal of CMMI Level 2 is institutionalize a Managed Process. Generic practices for this generic goal are Establish an Organizational Policy, Plan the Process, Provide Resources, Assign Responsibility, Train People, Manage Configurations, Identify and Involve Relevant Stakeholders, Monitor and Control the Process, objectively Evaluate Adherence.

3.5.2.2.1 Requirements Management (REQM)

The policy of the Requirements Management relates to the management and documentation of the requirements and their traceability, where the requirement can be new or changes to the existing, to enable monitoring and control of the impact on the projects plans and its dependencies. The Specific Goal is to manage requirements. Specific Practices are to obtain an Understanding of Requirements, Obtain Commitment to Requirements, Manage Requirements Changes, Maintain Bidirectional Traceability of Requirements, Identify Inconsistencies between Project Work and Requirements.

3.5.2.2.2 Project Planning (PP)

The policy of the Project Planning is to define a framework for software development and software maintenance teams to develop their project plans to determine the duration, work efforts and resource efforts to identify the effective milestones/deliverables. The Specific Goal is to Establish Estimates and Specific Practices are to Estimate the Scope of the Project, Establish Estimates of Work Product and Task Attributes, Define Project Life Cycle, Determine Estimates of Effort and Cost. The Specific Goal is to develop a Project Plan and Specific Practices are establishing the Budget and Scheduling, Identifying Project Risks, Planning for Data Management, Planning for Project Resources, Planning for Needed Knowledge and Skills, Planning Stakeholder Involvement, Establishing the Project Plan. Specific Goal Obtains Commitment to the Plan. The Specific Practices are Review Plans that affect the Project, Reconcile Work and Resource Levels, Obtain Plan Commitment.

3.5.2.2.3 Project Monitoring and Control (PMC)

The policy of Project Monitoring and Control is to make a quantitative identification/monitoring of the deviations from the estimated project plan and control the progress of the project to produce timely milestones/deliverables. The Specific Goal is to Monitor Project against Plan and Specific Practices are to Monitor Project Planning Parameters, Monitor Commitments, Monitor Project Risks, Monitor Data Management, Monitor Stakeholder Involvement, Conduct Progress Reviews, Conduct Milestone Reviews. Specific Goal Manages Corrective Action to Closure and Specific Practices are Analyze Issues, Take Corrective Action, and Manage Corrective Action.

3.5.2.2.4 Measurement and Analysis (MA)

The policy of Measurement and Analysis is to make a quantitative measurement of the software engineering processes for various projects, to derive the progress indicators for the organization's performance, which will be used for future improvement in terms of both quality and productivity. Specific Goal is to Align Measurement and Analysis Activities. Specific

Practices are Establish Measurement Objectives, Specify Measures, Specify Data Collection and Storage Procedures, Specify Analysis Procedures. Specific Goal is to Provide Measurement Results and Specific Practices are to Collect Measurement Data, Analyze Measurement Data, Store Data and Results, Communicate Results.

3.5.2.2.5 Supplier Agreement Management (SAM)

The policy of Supplier Agreement Management refers to the management and documentation of the process of structured identification of the effective supplier with quantitative justification and enforce the agreement liability for required purchases of services and/or tools. Specific Goal is to Establish Supplier Agreements. Specific Practices are to determine Acquisition Type, Select Suppliers, and Establish Supplier Agreements. Specific Goal is to Satisfy Supplier Agreements and Specific Practices are to execute the Supplier Agreement, Monitor Selected Supplier Processes, Evaluate Selected Supplier Work Products, Accept the Acquired Product, Transition Products.

3.5.2.2.6 Process and Product Quality Assurance (PPQA)

The policy of Process and Product Quality Assurance relates to effective implementation of the organizations processes and adherence to the quality standards defined for each process. The Specific goal is to evaluate processes and work products and specific practices are meant for objective evaluation of work products and services. Specific goal is to provide objective insight and specific practices involve communicating and ensuring resolution of noncompliance issues, establishing records.

3.5.2.2.7 Configuration Management (CM)

The policy of Configuration Management relates to the establishment and maintenance of the integrity of the project work and deliverables for controlled changes and rollback. The Specific goal includes the establishment of baselines and specific practices are identifying configuration items, establish a configuration management system, create or release baselines. Specific goal is to track and control changes and specific practices are to track change requests, control configuration items.

3.5.2.3 CMMI Level 3

These are process areas in CMMI Level 3.

1. Decision Analysis and Resolution
2. Integrated Project Management +IPPD
3. Organizational Process Definition +IPPD 4
4. Organizational Process Focus
5. Organizational Training
6. Product Integration
7. Requirements Development
8. Risk Management
9. Technical Solution
10. Validation
11. Verification.

Figure 3.5 CMMI Level 3 Process areas

3.5.2.3.1 Decision Analysis and Resolution (DAR)

The policy of Decision Analysis and Resolution is to define a structured and documented methodology for making decisions using the evaluation method for alternative solutions against the established criteria.

3.5.2.3.2 Integrated Project Management +IPPD (IPM)

The policy of the Integrated Project Management relates to the provision of established and structured methodology for tailoring the organization's set of standard process to meet the customized SDLC requirement of the customer. Specific Goal is using the Project's Defined Process and Specific Practices are to Establish the Project's Defined Process, Use Organizational Process Assets for Planning Project Activities, Establish the Project's Work Environment, Integrate Plans, Man- age the Project Using the Integrated Plans, Contribute to the Organizational Process Assets. Specific Goal is to coordinate and Collaborate with Relevant Stakeholders and Specific Practices are to Manage Stakeholder Involvement, Manage Dependencies, Resolve Coordination Issues.

3.5.2.3.3 Organizational Process Definition +IPPD (OPD)

The policy of the Organizational Process Definition relates to the establishment of a structured and consistent approach of process implementation. The Specific Goal is Establishing Organizational Process Assets and Specific Practices are Establishing Standard Processes, Establishing Life-Cycle Model Descriptions, Establishing Tailoring Criteria and Guidelines, Establishing the Organization's Measurement Repository, Establishing the Organization's Process Asset Library.

3.5.2.3.4 Organizational Process Focus (OPF)

The policy of the Organizational Process Focus relates to planning and implementation of organizational process improvement based on a thorough understanding of the current strengths and weaknesses of the organization's processes and process assets. The Specific Goal is to determine Process Improvement Opportunities and Specific Practices are Establish Organizational Process Needs, Appraise the Organization's Processes, Identify the Organization's Process Improvements. The Specific Goal is to Plan and Implement Process Improvement Activities and Specific Practices are to Establish Process Action Plans, Implement Process Action Plans. Specific Goal is to Deploy Organizational Process Assets and Incorporate Lessons Learned and Specific Practices are to Deploy Organizational Process Assets, Deploy Standard Processes, Monitor Implementation, Incorporate Process-Related Experiences into the Organizational Process Assets.

3.5.2.3.5 Organizational Training (OT)

The policy of the Organizational Training relates to development of the skills and knowledge of people to enhance the productivity, resource usage and employee satisfaction.

3.5.2.3.6 Product Integration (PI)

The policy of the Product Integration relates to the production of a simple and effective assembly of the product and its component to maintain the integrity and validity of the system, software and process requirements, to satisfy quick and easy deployment.

3.5.2.3.7 Requirements Development (RD)

The policy of Requirements Development relates to establishment and maintenance of the requirements of system, software and processes in more presentable and conceptually clear requirements without any ambiguity. It covers the specific goal and practices for elicitation and development of the requirements, identifying interrelations, validation of the requirements and also the analysis of the requirements.

3.5.2.3.8 Risk Management (RSKM)

The policy of Risk Management refers to active identification, analysis and mitigation of the risks associated with requirements for system, software and processes and its impact on dependent software development life cycle phases. The Specific Goal is to Prepare for Risk Management and Specific Practices are to Determine Risk Sources and Categories, Define Risk Parameters, Establish a Risk Management Strategy. The Specific Goal is to Identify and Analyze Risks and Specific Practices are to Identify Risks, Evaluate, Categorize, and Prioritize Risks. The Specific Goal is to Mitigate Risks and Specific Practices are to

Develop Risk Mitigation Plans, Implement Risk Mitigation Plans.

3.5.2.3.9 Technical Solution (TS)

The policy of Technical Solution refers to effective designing and development of the solutions guided by the defined standards for the process, within the scope of defined requirements and in timely fashion to meet the schedules defined in project plan.

3.5.2.3.10 Validation (VAL)

The policy of the Validation process refers to ensuring the effectiveness of the product and its components against the specified requirements of system, software and processes. The Specific Goal is to Prepare for Validation and Specific Practices are to Select Products for Validation, Establish the Validation Environment, Establish Validation Procedures and Criteria. The Specific Goal is to Validate Product or Product Components and Specific Practices are Perform Validation, Analyze Validation Results.

3.5.2.3.11 Verification (VER)

The policy of the Verification process refers to ensuring the implementation of the selected work products for the project is in compliance with its specified requirements.

3.5.2.4 CMMI Level 4

Two Process areas of CMMI Level 4 namely,

1. Quantitative Project Management
2. Organizational Process Performance

Figure 3.6. Process areas of CMMI Level 4

3.5.2.4.2 Quantitative Project Management (QPM)

The purpose of the Quantitative Project Management (QPM) process area is to quantitatively manage the project's defined process to achieve the project's established quality and process-performance objectives. Specific Goal is to Quantitatively Manage the Project and Specific Practices are to establish the Project's Objectives, Compose the Defined Processes, and Select the Sub-processes that will be Statistically Managed, Manage Project Performance. Specific Goal is to Statistically Manage Sub-process Performance and Specific Practices are to Select Measures and Analytic Techniques, Apply Statistical Methods to Understand Variation, Monitor Performance of the Selected Sub-processes, Record Statistical Management Data.

3.5.2.5 CMMI Level 5

Two Process areas of CMMI Level 5 namely,

1. Causal Analysis and Resolution
2. Organizational Innovation and Deployment

Figure 3.7 Process areas of CMMI Level 5

3.5.2.5.1 Causal Analysis and Resolution (CAR)

The purpose of Causal Analysis and Resolution (CAR) is to identify the causes of defects and other problems and take action to prevent them from occurring in the future.

3.5.2.5.2 Organizational Innovation and Deployment (OID)

The purpose of Organizational Innovation and Deployment (OID) is to select and deploy incremental and innovative improvements that measurably improve the organization's processes and technologies. The improvements support the organization's quality and process-performance objectives derived from the organization's business objectives. The Specific Goal is to Select Improvements and Specific Practices are to Collect and Analyze Improvement Proposals, Identify and Analyze Innovations, Pilot Improvements, Select Improvements for Deployment.

The Specific Goal is to Deploy Improvements and Specific Practices are to Plan the Deployment areas, manage the Deployment, Measure Improvement Effects.

3.6 Process Capability Profile as an Evolution of Software Process Improvement

The software industry has been using software process improvement approaches, based on the maturity levels of SW-CMM and CMMI staged models, to improve its business. However, in practice, most organizations do this usually on an informal basis, doing more than just implementing the maturity levels. They use multiple models and other references for process improvement, and the additional process areas. They also use process improvement in other areas related to software processes. This work presents a proposal towards a Process Capability Profile Driven Process Engineering (PCDE) as an evolution of the current Software Process Improvement based on Process Capability (and Maturity) Models. This proposal has been constructed since 1999 using experiences from many process improvement projects in software industry. An industry-as-laboratory approach [93] was used, instead of the traditional research-then-transfer approach. This proposal aims to formalize what software intensive organizations are already doing, in an informal way.

3.6.1 Seven issues and opportunities

As a synthesis of a critical view of the issues and opportunities from the current state of the art and state of the practice of software process improvement area and related areas, seven views are presented.

- **Fixed maturity levels:** The software process improvement area was established based on the fixed "one size fits all" maturity levels of SW-CMM model [94] from around 1993 until its retirement around 2002 and of CMMI staged representation models [95] since 2000. These maturity levels guided the software industry on the improvement of the process for software development projects. As with all good pioneer work, the maturity levels establish an area based on "one size fits all" approach, as, for example, Henry Ford established the car industry with his "T model" black car. After the establishment, there is a need for more flexibility to cope up with the diversity. Most of the problems of software process improvement are identified by the community, as, for example, the ones from [96] and [97], are due to the fixed maturity levels of SW-CMM and CMMI models.
- **Flexibility of continuous architecture:** The ISO/IEC 15504, also known as SPICE (Software Process Improvement and Capability Determination) has proposed a continuous architecture as a more flexible alternative to the fixed maturity levels of staged models. The continuous architecture defines two dimensions for process capability models: one with a set of processes and another with process capability levels. An organization can choose a process capability profile, which is a subset of processes, each one in at a process capability level, to guide the improvement of its more relevant processes. Examples of continuous models are the ISO/IEC 15504-5 [2006], FAA iCMM model [98] and the continuous representation of CMMI models [95]. The continuous models need an evolution of the current approaches for software process improvement for a bigger use by the software industry. There is a need for methodological support to use the flexibility to define process capability profiles.
- **Using multiple models:** Many organizations are using simultaneously elements of multiple models as reference for process improvement. Some of these models are capability maturity models, other are certification

models, other generic process models and any other kind of model that represents best practices. Maybe the most common combination nowadays is the combined usage of CMMI-SE/SW staged representation, ISO 9001, RUP and PMBOK.

- **Define or use models:** Nowadays there is a clear cut distinction between groups that define process capability models and the ones that only uses these models. Examples of groups that defines models include SEI groups, in the development of SW-CMM and CMMI models, ISO/IEC JTC1 SC7 WG10 group, in the development of ISO/IEC 15504-5 model, FAA groups, in the development of iCMM model, and MPS-BR group, in the development of MR-MPS model [99]. Examples of groups that only use one of these models are the SEPG groups of software intensive organizations.
- **More specific models:** There is a tendency for more specific process capability models for technological segment or domain under the ISO/IEC 15504-2 framework. [100] defines six process capability levels as a measurement framework, the requirements for process reference models and process assessment process, and other orientations and requirements. In addition to the ISO/IEC 15504-5 model, CMMI models, iCMM model, MR-MPS model, there already other models under ISO/IEC 15504-2 framework, including the OOSPICE model for component-based software engineering [101] and automotive SPICE model for car's software supplies for the automobile industry [102].
- **Model-driven engineering:** Model-driven engineering is an emergent area that put model in the center of an engineering. Favre concludes a model-driven engineering overview with the definition of two terms. "Model engineering is the disciplined and rationalized production of models. Model- driven engineering is a subset of system engineering in which the process heavily relies on the use of models and model engineering" [103].
- **Fundamental concepts:** David Card observed that the process improvement approaches are "all based on very similar concepts and techniques". However, because these approaches "have evolved or been adapted to software engineering largely without the participation of the

academic research community", "they are considered competitors". "The packaging obscures the underlying principles. Eliciting and refining underlying principles is the role of science" [104].
An evolution of the current software process improvement should extend the fixed maturity levels using the flexibility of continuous models, supporting the usage of multiple models, including the development of more specific models, allowing an organization to define and use models as in model- driven engineering, by eliciting and refining the fundamental concepts and underlying principles of the current state of the practice and state of the art of software process improvement.

3.6.2 The basis and a proposal for process engineering

Process can be considered as virtually any granularity. Given an organization unit, the complete view about what people do in that organizational unit can be seen as a process, in this case, the organization unit process. Any subset of the organizational unit process can be considered also as process, as, for example, the process for unit tests. An appropriate granularity for process is related to what the ISO/IEC 15504-5 defines as "process" and what the CMMI models define as "process area". In spite of the difference in name, both concepts have similar features. ISO/IEC 15504-5:2006 defines 48 processes and CMMI-SE/SW v1.1 defines 22 process areas. The term "process area" is used in this research to mean both concepts. Therefore, a process is a set of actions that people do that could be represented by a process area, from any model.
The six process capability levels as defined by ISO/IEC 15504-2 are the best candidates for the fundamental law of process improvement. The performance of any process at an organization unit can be estimated by a characterization of the process in one of the six capability levels from level 0, incomplete, to level 5, optimizing, given that the process is abstracted as a process (area).
Process capability profile as a combination process capability level and process area, is the best candidate for the basic unified concept of a reference for a process improvement. Each element of a process capability profile represents a process, in the granularity of, and represented by, a process area, at a process

capability level. Therefore a process capability profile represents a process at a granularity of the aggregation of the processes in each element. The Figure 3.8 illustrates a candidate for the basic relationship for process engineering.
Using the M0-M1 hierarchy [105], a process is a part of the world (M0) represented by a model, in the modeling space (M1). In this case, the model is a process capability profile that represents the process under the process capability aspect. A point to note is that a process description also can represent a process, in this case under the process description aspect. A process capability profile can be a prescriptive or descriptive model. As a prescriptive model a process capability profile drives the improvement of a process towards a better process using the requirements and orientations from the process capability profile. As a descriptive model the process is represented by a process capability profile that is a result from a process assessment process.

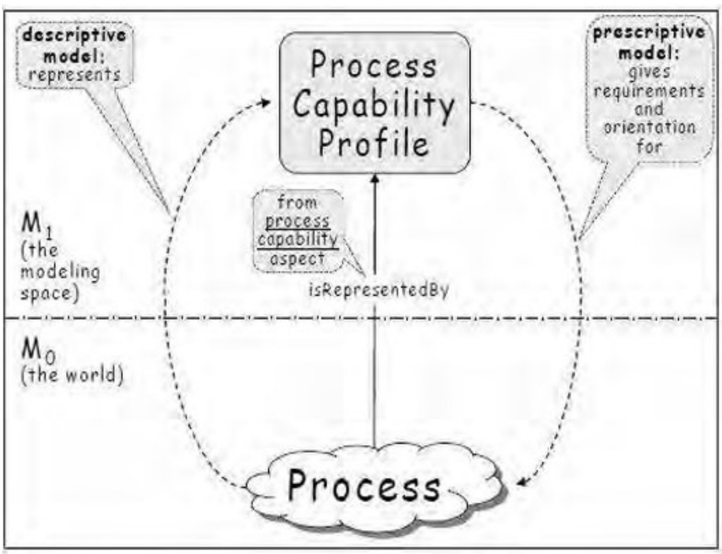

Figure 3.8 Process and process capability profile
The short term "{(Process Capability Profile)-Driven (Process Engineering)}" and a long term "{(Process Capability Profile)-Driven [Software and any other Knowledge Intensive Human Work] (Process Engineering)}", both with the same meaning and with the same initials (PCDE), are proposed as evolution of the current software process improvement.

A more complete definition for the proposed process engineering is presented below.

{(Process Capability Profile)-Driven

[Software and any other Knowledge Intensive Human Work] (Process Engineering)} is

1. The application of (engineering) which is concerned with "creating cost-effective solutions to practical problems by applying scientific knowledge to build [concrete or abstract] things in the service of mankind". To the (definition, usage, management, establishment, measurement, change, improvement and co-evolution) of consistent pair of (process capability profile) And [software and any other knowledge intensive human work] (process) Oriented by the (process capability discipline), As a mean to achieve (organizational excellence);
2. the application of (engineering), To the (definition, usage, management, establishment, measurement, change and improvement) of (process capability model) for a more specific technological context or domain; and
3. the study of approaches as in (1) and (2). Process improvement is not anymore meant only for software processes. The current versions of SEI CMMI and ISO/IEC 15504 use the term "system" as a more generic boundary that includes software. A boundary line related with the term "knowledge worker" seems to be more appropriate. This term was used by Peter Drucker in his 1959 book, Landmarks of Tomorrow as "a knowledge worker is anyone who works for a living at the tasks of developing or using knowledge". "For example, a knowledge worker might be someone who works at any of the tasks of planning, acquiring, searching, analyzing, organizing, storing, programming, distributing, marketing, or otherwise contributing to the transformation and commerce of information and those (often the same people) who work at using the knowledge so produced" [106]. The knowledge worker includes those in the information technology field, such as programmers, system analysts, technical writers, academic professionals, researchers, and so forth". Knowledge workers include people outside the area of information technology, such as lawyers,

doctors, diplomats, lawmakers, marketers, managers, bankers, teachers, scientists of all kinds, and students of all kinds.

3.7 CMMI-Based Software Process Maturity on Software Schedule Estimation

Development of a software project with acceptable quality within budget and on a planned schedule is the main goal of every software development firm. Schedule estimation has historically been and continues to be a major difficulty in managing software development projects [107]. Failure of the project is attributed mostly to failure to fulfill customers◎ quality expectations or the budget and schedule overrun. Over the last decades, several effort and schedule estimation models have been developed, and most of them have disappeared without any kind of rigorous evaluation. The reason might be that these models were not good and precise enough [108]. In fact, the presence of another important reason has to be considered, namely, people who work in software development prefer to use their own estimation techniques rather than improving and applying the work of others. According to [109], most companies have relied on experience and "Price-to-win" strategies for getting past competitors to win projects. Despite the emergence of concepts like Software Capability Maturity Model (SW-CMM) one can never rely completely on experience based estimation in the software industry due to the rapid changes in technologies, which render the experience-based estimates ineffective. Furthermore, price-to-win strategy is not very favorable for most companies. Hence, the need arises to come up with a more effective model to account for the schedule of developing software systems. A number of algorithmic models have been proposed as the basis for estimating the schedule of a software project. They are conceptually similar but use different parameter values. While most of those software models are proprietary, COCOMO II (the primary focus in this research work) is a fully documented and widely accepted model, updated from original COCOMO 81 [110] till its most recent version, COCOMO II [111].

3.7.1 COCOMO II Model

The Constructive Cost Model (COCOMO), was first published in 1981 (COCOMO 81) [110], and became one of most popular parametric cost estimation models of the 1980s. But in the 90s, COCOMO 81 faced a lot of difficulties and complications in estimating the costs of software that were developed to a new life cycle processes such as non-sequential and rapid development process models, reuse- driven approaches, and object-oriented approaches [112]. Thus, COCOMO II was published initially in the Annals of Software Engineering in 1995 with three sub models; an application-composition model, an early design model and a post- architecture model [112]. COCOMO II has, as an input, a set of seventeen effort multipliers (EM) or cost drivers which are used for the adjustment of the nominal effort (PM) to reflect the software product being developed. The seventeen COCOMO II factors (cost drivers) are shown in Table 3.3 [111].

3.7.1.1 Effort Estimation

The COCOMO II effort estimation model has been formulated. This model is used for both Early Design and Post-Architecture models for the estimation of effort. Inputs are the Size of software development, a constant A, an exponent E, and a number of effort multipliers (EM). The number of effort multipliers depends on the model being used.

$$PM = A \text{ x } size^{E} \times \prod_{i=1}^{N} EM_i \quad \text{……....3.5}$$

where the constant A=2.94, and the exponent E

3.7.1.2 Scale Factors

A study done by [113] presents the conclusion that the most critical input to the COCOMO II model is size, and so, a good size estimate is very important for any good model estimation. Size in COCOMO II is treated as a special cost driver, and so has an exponential factor, E, which is an aggregation of five scale factors and it is expressed in equation 3.6. All scale factors have rating levels. These rating levels are very low (VL), low (L), nominal (N), high (H), very high (VH) and extra high (XH). Each rating level has a weight, W, which is a quantitative value used in the COCOMO II model. The five COCOMO II scale factors are shown in Table 3.3 [111]:

$$E = B + 0.01 \times \sum_{j=1}^{N} SF_j$$

where B is a constant = 0.91. A & B are constant values devised by the COCOMO team by calibrating to the actual effort values for the 161 projects currently in COCOMO II database.

3.7.1.3 Schedule Estimation

Project Schedule months is the number of calendar months from the time the development begins and going through the time it is completed. Boehm et al. [111] have produced to estimate the project scheduling months. It has denoted as Time to Develop, TDEV:

$$TDEV = C\ ' \ (PM)^F \quad \ldots 3.7$$

Where C = 3.67, PM is the Person-Months, and F, as stated is the schedule equation exponent derived from the five Scale Factors.

$$F = D + 0.2 \text{ x } 0.01 \times \sum_{j=1}^{N} SF_j$$

where D= 0.28 and SF is the COCOMO II's scale factor. C & D are constant values devised by the COCOMO team by calibrating to the actual schedule values for the 161 projects currently in COCOMO II database.

Table 3.3 COCOMO II Scale Factors and Cost Drivers

Scale Factors	Cost Drivers	
Precedentedness (PREC)	Required Software Reliability (RELY)	Programmer Capability (PCAP).
	Data base size (DATA).	Application Experience (APEX).
Development Flexibility(FLEX)	Developed for Reusability (RUSE)	Platform Experience (PLEX).
	Documentation needs (DOCU).	Language & Tool Experience (LTEX).
Risk Resolution (RESL)	Product Complexity (CPLX).	Personnel Continuity (PCON)
	Execution Time Constraints (TIME).	Use of Software Tools (TOOL
Team Cohesion (TEAM)	Main storage Constraints (STOR)	Multisite Development (SITE).
	Platform Volatility (PVOL).	Development Schedule (SCED).
Process Maturity (PMAT)	Analyst Capability (ACAP).	

The procedure for determining PMAT – the factor of interest in this study - is organized around the Software Engineering Institutes Capability Maturity Model (SEI-CMM), Table 3.4 [111].

Table 3.4 PMAT scale factor with its rating levels and values.

PMAT Description	CMM Level 1 (lower)	CMM Level 1 (upper)	CMM Level 2	CMM Level 3	CMM Level 4	CMM Level 5
RatingLevels	VeryLow	Low	Nominal	High	Very High	Extra High
Values	7.80	6.24	4.68	3.12	1.56	0.00

According to [114], The CMM Level 1 (lower half) is for organizations that depend on “heroes” to do the task. They do not concentrate on repeatable processes. The CMM Level 1 (upper half) is for organizations that have adhered to most of the requirements that satisfy CMM Level 2. In the published definition of CMM, Level 1 (Lower half) and (Upper half) are grouped into Level 1.

3.7.2 CMMI-based Process Maturity

The Software Capability Maturity Model (SW-CMM) published by SEI is used for rating an organization's process maturity [115]. SW-CMM provides a number of requirements that all organizations can use in setting up the software processes used to control software product development. The SW-CMM specifies "what" should be in the software process rather than "when" or "for how long". There are five levels of process maturity, Level 1 (lowest half) to Level 5 (highest). The organization should demonstrate capabilities in a set of Key Process Areas (KPA) associated with a specific SW-CMM level for getting rated at a particular level.. The capabilities demonstrated in moving from lower levels to higher levels are cumulative. For example, Level 3 organizations should show compliance with all KPAs in Level 2 and Level 3. The detailed information on SW-CMM Process Maturity is available in [115].

Since release SW-CMM has seen application to many areas; therefore, several capability maturity models have been provided. These included people CMM (P- CMM), system engineering CMM (SE-CMM), the software acquisition CMM (SA- CMM), and the integrated product development CMM (IPD-CMM) (EPIC, 1996). As these models were built by different organizations, there was an overlapping in the application's scopes in addition to the lack of consistency in the terminology, assessment approach, and architecture. These problems led to the increase of time and cost to adopt multiple models. Therefore, the Software Engineering Institute, SEI, has, in 2000, released the Capability Maturity Model Integration (CMMI) in order to integrate all existing capability maturity models. On August, 2000, (CMM) was replaced by a new process model, which is the Capability Maturity Model Integration (CMMI). The Capability Maturity Model Integration (CMMI) was created to reduce redundancy, to support product and process improvement, and to eliminate undesired inconsistency experienced by organizations that have been pursuing multiple models. The CMMI combines all relevant process models found in CMM into one product suite [116].

There are two representations of CMMI: continuous representation; and staged representation. The continuous representation focuses on the capability of process areas, while the staged representation focuses on the organizational maturity. This work concentrates in CMMI staged representation which is discussed briefly below.

Like CMM, there are five maturity levels in CMMI, numbered through 1 to 5 in staged representation. Maturity levels are defined in terms of related specific and generic process areas and the achievement of their requirements. Achievement of specific and generic goals related to a process area determines the organization's maturity level. Refer to [116] for more details about CMMI.

In this research, the literature from two different perspectives has been referred to. The focus is on the calibration and improvement of the COCOMO II model, while it is on the benefits of increasing maturity levels as well as the benefits of CMMI-based software process improvement. This researcher's work is a kind of combination between the previous two perspectives, i.e. improvement of the schedule prediction power of the COCOMO II model by investigating the benefits of CMMI- based software process maturity.

COCOMO II is being revised, updated, and calibrated to be more suitable for future estimation. There are several calibrations conducted on COCOMO II [117],[118],[119],[120],[121]. Also, numerous studies have been done to enhance the predictive power of the COCOMO model by adding or reducing some influencing factors or cost drivers [122],[123],[124],[125],[126].

Chulani et al. [117] have reported a study with a regression tuning algorithm using the COCOMO project database producing estimates that are within 30% of the actual values, 69% of the time, while Clark [119] reported a study in which the Bayesian 38 tuning are within 30% of the actual values, 76% of the time after stratification by the organization. Yahya et al. [126] improved the COCOMO II's predictive power by adding a set of 16 factors to the model and considered it as the most influential factors in their local environment; they claimed that their enhanced model has improved the COCOMO II◎s predictive power by 9% as compared to the generic COCOMO II. Chen et al. [123] concluded that the COCOMO II model can be improved via WRAPPER feature subset selection method developed by the data mining community. Using data sets from the PROMISE repository, they showed in COCOMO II's predictive power a significant and dramatic improvement by WRAPPER. Huang et al. [127] have proposed a novel neuro-fuzzy Constructive Cost Model for software cost estimation. They claimed that the validation using industry project data shows that the model greatly improves estimation accuracy in comparison with the generic COCOMO model. Baik argued in [122] that disaggregation of the TOOL variable in COCOMIO II improves the prediction accuracy from 67% to 87%.

On the other hand, there has been a big discussion on the benefits of increasing maturity levels as well as the benefits of CMMI-based software process improvements [128],[129],[130].

By adopting the CMM, researchers have found a significant improvement in the control, predictability, and the effectiveness of the processes. According to [131], each CMM level enhances the quality of the product and generally reduces the development schedule. Manish and Kaushal [134] focused on CMM level 5 software projects from several organizations to investigate the effects of highly mature processes maturity on development effort, quality, and schedule. Based on historical data projects from 37 CMM level 5 of four organizations and by using a linear regression model, they found high process maturity levels, as indicated by the rating of CMM level 5, mitigating the impact of most factors that earlier were believed to affect the software development effort, quality, and schedule such as personnel capability, requirements volatility, and requirements specifications. They also claimed that the only factor found to be important in determining effort, schedule, and quality was the software size. On an average, estimated effort and schedule of their developed models were around 12% percent and defects about 49% of the actual, across organizations. In general, their results indicated some of the biggest advantages from high levels of organizational process maturity coming from the obvious reduction invariance of software development outcomes that were previously caused by some factors other than size of the software.

In order to investigate the impact of the Process Maturity on software development effort, and based on CMM with the aid of 161-project sample,

Clark [114] isolated the effects of the effort on the process maturity versus effects of other factors, and concluded that an increase of one process maturity level could reduce development effort by 4% to 11%, but this reduction seemed like a generalization across all five levels of CMM process maturity, i.e. the percentage of effort reduction is not the same among all levels. Despite the fact that several researches and case studies have shown many benefits of enhancing organizational process maturity through use of different assessment approaches [136], none has attempted to isolate individual factors that affect productivity as shown by Clark when he isolated the effects of process maturity on effort versus other factors. Nevertheless, they have indicated some considerable effects through increase in organizational maturity levels.
Donald et al. [130] have conducted an empirical research to find out the

relationship between the quality of the products, organizational process maturity, development effort, and project's schedule for a set of 30 software products in IT firms. Their findings indicated process maturity having an effect in reducing software development schedule and effort. Diaz and Sligo [135] have reported the effect of the process maturity level on software development schedule by indicating how software process improvement helped Motorola. Based on some measurements, Motorola's software development schedule was around eight times faster at CMM level 5 than at CMM level 1.
Despite numerous studies on the performance assessment results of CMM-based software process maturity and its impact on software development effort and schedule, work on the overall CMMI-based software process maturity [137] is still limited. Unlike previous studies in literature that have pointed out the benefits of CMMI-based software process maturity and, [137] in terms of six dimensions of the performance assessment considered the performance assessment for both tangible and intangible benefits of CMMI adoption. They have presented the results of performance assessment of the CMMI-based Software process improvement based on an empirical study from 18 software firms in Taiwan, which have already attained CMMI

maturity level 2 and 3 certifications. They argued that their empirical study revealed that the CMMI-based software process improvement has a positive effect on the six performance dimensions in software firms investigated by them. However, the benefits gained were in "Lighten the load of project members", "Improve product usability", "Improve product efficiency", "Improve product portability", "Increase bargain power" and "Reduce the project effort and schedule".

Another study conducted by [133] has reported some great quantitative evidence of Capability Maturity Model Integration (CMMI)-based software process improvement providing a higher quality products and better project performance with lower cost and decreased project schedule. The reported results were drawn from a set of 12 cases from 11 independent firms. Since the performance results provided by [133] were limited, [132] continued the assessment performance of CMMI-based software process improvement. Results were drawn from a variety of small and huge organizations around the world. They have reported most of their results coming from higher maturity organizations, but some notable enhancements achieved by lower maturity organizations have also been seen. Great quantitative results obtained for all six performance categories have been discussed by [132] including software cost and schedule.

Table 3.5 summarizes the benefits and impact of CMMI-based software process improvements from Schedule perspective from a sample of organizations.

Table 3.5 Summary of benefits and impact of CMMI adoption–Schedule [132]

S. No	Results	Organization
1	On-time delivery remained well over 90 percent, movingfrom 97% to 99%, with a slight improvement as the organization moved from SW-CMM maturity level 3 to CMMI maturity level 5	IBM Australia Application Management
2	70 to 80 percent reduction in the average slippage of project delivery dates as the organization achieved CMMImaturity level 2	JP MorganChase
3	Average days variance from development plan reduced from approximately 130 days to less than 20 days one yearafter reaching CMMI maturity level 2	NCR

4	Met every schedule milestone (25 in a row) on time, with high quality and customer satisfaction in a CMMI maturity level 5 organization	Northrop Grumman IT, Defense Enterprise
5	Substantially improved schedule variance over three causal analyses and resolution cycles in a CMMI maturity level 5 organization with PSP-trained engineers	Northrop Grumman IT, Defense Enterprise
6	Schedule variance improved from approximately 25 percent to 15 percent as the organization moved from SW- CMM maturity level 3 to CMMI maturity level 5	Reuter
7	On-time deliveries improved from 79 percent to 89 percent as the organization moved from SW-CMM maturity level 3 toward CMMI maturity level 4	Systematic Software Engineering
8	Schedule variation declined by 63 percent as the organization moved from SW-CMM maturity level 4 to CMMI maturity level 5	The Boeing Company

3.8 Summary of Review of Literature and Implications

The review provided a deeper insight into research aspects related to the problem chosen for the study. The following are the salient aspects brought out by the review of literature.

3.8.1 Aspects related to Small and Medium Sized Enterprises (SMEs)

On conclusion of literature review in SMEs, we conclude that the following:
The review based on the unique characteristics of SMEs which are the context of the study. The key literature in the relevant areas was reviewed. Then, the concept of SMEs are characterized by resource scarcity, the strong influence of Owners / Managers, flat organizational structure, systems, processes and procedures and their culture and behavior.

3.8.2 Aspects related to Software Development Process

Conclusions from a literature review in Software Development Process and Software Development Effort Estimation, are the following:

- This refers to the series of steps a person or organization follows to produce a software system. A software process is the process used by an individual or organization on a specific project.
- It describes the phase of a software project gone through over time based on analysis, design, implementation, etc.
- In the software organization, the process of SDLC is proposed for the improvement of software project.
- It describes the capability for consistent and accurate estimation of the

effort of software development particularly during the earlier stages of the development life cycle is required for planning and conducting software development from the project manager.

* Effort estimation technique discusses the approved process that has come up with the usages of effort estimations.

3.8.3 Aspects related to CMMI

The conclusions from a literature review in Structure of CMMI, Need for CMMI and CMMI Maturity Levels are:

* It describes the CMMI builds upon three key concepts: process areas, goals, and practices.
* Review based on maturity level is a well-defined evolutionary plateau toward achieving a mature software process.
* CMMI Level 1provides a chaotic or unstable environment for the processes.
* CMMI Level 2 is the second of the five maturity levels in the staged representation of the CMMI.
* CMMI Level 3, CMMI Level 4 and CMMI Level 5 describe the process areas.

3.8.4 Aspects related to Process Capability Profile

The conclusions from literature review towards a Process Capability Profile Driven Process Engineering (PCDE) as an evolution of the current Software Process Improvement based on Process Capability (and Maturity) Models are:

* It describes the issues and opportunities from the current state of the art and state of the practice of software process improvement.
* Review of process capability profile can be a prescriptive or descriptive model.

3.8.5 Aspects related to CMMI-Based Software Process Maturity

Conclusions from a literature review on Software Process Maturity on Software Schedule Estimation are

* It describes the COCOMO II Scale Factors and Cost Drivers.
* Review is based on Software Capability Maturity Model (SW-CMM) published by SEI is used to rate an organization's process maturity.
* Summary of benefits and impact of CMMI adoption–Schedule

CHAPTER – 4
PROBLEM DEFINITION

4.1 Introduction

Nowadays, each organization needs delivery of its services and products in the market to be better, quick and cost efficient. Each organization has generated complicated products and services but, improvement to product or service is a complex preposition and requires a significant effort. Therefore, organizations select a few components for in-house improvement and some of them to be outsourced from other organizations where in the final service or product gets involved through integration. Moreover, all these activities in the organization should have capability and maturity for controlling and managing their own processes of complex development and maintenance.

4.1.1 Current Status

The major problem faced by several organizations is in the development of this integrated approach which is efficient and effective. Capability Maturity Model Integration (CMMI) provides a good practice oriented approach by eliminating these barriers and issues faced by several organizations, whereas CMMI for development is comprised of best practices which focus an applicable advancement in software development activities for products and services provided by a company or organization. However, 80% of the global economy is serviced from the CMMI for services model from CMMI Institute has assisted these service organizations improving their process and make all their resources achieving best business results.

4.1.2 Problem State

Therefore, CMMI for Service (CMMI-SVC) model is utilized as a guide for assisting service provider organizations through cost reduction, quality improvement and consistency in delivery services. Thus, these model best practices have helped in improving huge share of profit and improve process capability with performance.

There are several components used in CMMI but one of the key components for improving the product and services of the organization is to meet the client requirement in an appraisal. Generic goals are called so due to a similar goal statement available from multiple process areas. The characteristics have described the generic goal which is presented to institutionalize a process area. These components assist in the management to discover the strength and weakness of their in-house software development team in an organization. This process development has an important impact in Small and Medium size Enterprises (SME) in understanding their process performance. SMEs are available in the essential subsector of strategic business services which is involved in services associated with information processing, computer software, advancement, marketing, research, business organization and improvement of human resources. The outsourcing improvement in using main manufacturing firms has combination with recent technologies which have permitted SMEs to be successful in market places that margin about 10% annual growth in these services of knowledge based industry.

4.2 Statement of the Problem

Thus the subject of this research work has emerged as **"DEFINING PROCESS CAPABILITY PROFILE BASED ON CMMI PROCESS PERFORMANCE FOR SMALL AND MEDIUM ENTERPRISES"**

In this research work, the study involves the review of previous metrics on process performances and discover the process weakness of the organization. The benchmark of organization against CMMI services has been targeted by achieving the maturity levels for the organization unit. In software organization data is collected and analyzed for deriving quantitative results to assess and manage the efficiency of process performance in development. Its capability profiling which lead to a high correlation between process maturity to Process Capability Profile (PCP) signify the continual improvement process initiative followed for appraisal.

4.3 Summary

This chapter has pointed out to a major problem and an issue faced by several organizations is developing this integrated approach which is efficient and effective. This chapter describes the Capability Maturity Model Integration

(CMMI) provides a best practices oriented approach by eliminating these barriers and issues faced by several organizations, whereas CMMI for development is comprised of best practices which focus an applicable advancement in software development activities for products and services provided by a company or organization.

CHAPTER – 5
RESEARCH METHODOLOGY

5.1 Overview of research motivation

The idea behind CMMI was reached from several platforms and advanced from the experience of various people whereas there are two issues that are focused in this research work, namely, increase in marketplace pressure in the organization and modifying the framework of CMMI based on appraisal methods for resolving organization issues. This focus is on organization maturity levels rather than on process capability. However, maturity levels may not be comprehensively calculated through organizational capability but can specify the risk in the process areas and also guide the development process using the description required with least activities set. In addition, several cases have high maturity ratings which do not specify effectively and practices with high maturity which is not the appraisal process with fault or the organizations are dishonest, simple in the framework of maturity does not seem to be extremely adequate into all practices in organizations. Without change, we can expect more cases where high maturity ratings will not generally correlate with better performance.

In this research work, the lessons learned from previous experience with a method for addressing this issue by controlling the precise and complex work, requires each person to be aware of detailed and precise plans and also measure and manage quality. In order to lead software developers with these application principles for their work the SEI has advanced the Personal Software Process (PSP). The developers use PSP in which the appraisers get detected these practices based on the CMMI for DEV in appraisal. Hence, the presence of the PSP has helped for further maturity in practices of developer. The SEI is now adapting the PSP to systems development and acquisition work. CMMI does not provide a definition for the improvement stage but the focus is on the PSP specification of project planning, tracking, and quality management to be performed. These issues have formed a method to involve several methods and practices into the CMMI model and method without swapping the focus from the requirement of client in the organization. This is an essential for enclosing the established principles and practices without inhibiting development organizations based on the advancement in technology. However, SEI functions on the basis of these issues as it endeavors for the development of efficient model and practices effectiveness of these methods and models.

This chapter illustrates the components available in every process area and even in the generic goals which consider these components which are critical for utilizing the information. Moreover, all the model of CMMI are created from the CMMI Framework whereas it consists of both goals and practices which have utilized for generating the model of CMMI that existed for CMMI constellations. Therefore, each process area has considered fundamental concepts which are general for improving the process in all areas of interest whereas each process area is an associated practice cluster in the area with focuses while implemented collectively that satisfies a set of goals considered as an essential to create improvement in those areas.

Software process model serves as groundwork for process definition, assessment and improvement. It guarantees the handling of the same concepts, significance with the finest software engineering practices and compatibility with globally accepted standards. The organizations should decide the process assessment model as more appropriate to their main goal but it is desirable to benefit of both models. Improvement of the quality of software service or products of Software Process Improvement (SPI) aims to exploit the benefits of economic which it follows using Small and Medium size Enterprises(SME). CMMI plays the several roles to describe the operational processes of characteristics and approach for process development [37]. It has one or more ideas based on collection of important elements of active processes. The process capability models are applied for the resolution of the obstacles which help understanding of an organization processes. CMMI plays the characterization of the business process that describe a theoretical background of particular organization. This can be viewed as appearance of a dual role referred to as the concept based on CMMI model. CMMI is defined as the group of concepts which is mapped to a particular enterprise. The framework of CMMI modeling is not an exclusive feature of duality but is somewhat distinctive of many other modeling frameworks. For example, specific software runtime system of UML model captures the indentation of such constants and relation among the concepts of classes. The domains of semantic web are referred to as ontology which is also referred as the concept of annotations. The product suite of CMMI used for producing constellation is referred to as a specific model which contains several CMMI structures and components. This generates various models, assessment materials and related training. The models are characterized by continuous representations models and staged representation models. It is also categorized by the types of processes. Continuous representation allows collections on the order of development with respect to business objectives of organizations, and permits comparisons between organizations by process areas. The staged representation model describes progress through predefined, improvement to sequence order and consecutive levels of a confirmed path. Each level functions as the next level of maturity. This allows the organizations by maturity levels that provide for appraisal results. It includes Software Process Improvement (SPI) models like CMMI which target on process capability gets measured using capability levels. The CMMI models namely, CMMI for Acquisition, CMMI for Development, CMMI for Services. This research work proposes the basic framework of CMMI-Dev., for covering project management and other supporting processes used in development and maintenance of projects. This approach presents the PCP of software companies with the specific business model whereas this proposed technique has considered with a guides, process and artifacts which get assisted for acquiring profile process based on the precise features of every software organization. To access the applicability of generated concepts were temporarily applied to the pilot organization of partial effects found in this activity. This is the method of development has illustrated with Process Capability Profile (PCP) based on Process Maturity (PMAT)[10].

5. 2 Approaches for the Research Design

In this research work, the design of the model helps access the applicability of generated concepts that temporarily applied to the pilot organization of partial effects found in this activity. There is various improvement programs have foundered simply because no action resulted from the appraisal. Improvement comes from action planning, assigning individuals to do the work, piloting and deploying improved processes, and management oversight throughout. The other design can help organizations in improving their software process but focusing on achieving a maturity level without really improving the underlying process is a danger whereas maturity levels should be a measure of improvement, not the goal of development. This is the design for a development method in service and product shown in the Figure 1 which illustrates Process Capability Profile (PCP) based on Process Maturity (PMAT) of the individual organization which assist in improving the maturity level that induces the profile capability of the individual process in the organization is mentioned in the below following steps.

Step 1: The set of features have been identified and applied in the software of small enterprises as the goal of the method whereas the value discipline, growth stage and business model are the three major features in the small enterprises even there are several feature available.

Step 2: The managed (level2) and defined (level3) maturity level in CMMI for Development (CMMI-DEV) with its process ranged it characterization gets mapped based on the selected features. Hence, the mapped characterization are compared with nine PCPs are evaluated in the small enterprises from Florianopolis by MARES.

Step 3: Based on step2 and bibliographic research, the execution process of initial (level1) version gets developed in the profile capability process works based on the definition method.

Step 4: The defined process of software applicable has been considered as an initial test is a goal of assessing the definition method.

Step 5: Detected adjustment has provided as a usage from profile capability process based on definition method for the assessment of a small enterprise.

5.2.1 Model for Process Capability Profile based on Process Maturity (PMAT)

The most essential instigators with developers have been focused to be visible SPI success, top-down commitment, i.e. visible senior management support for SPI, adequate resources, and bottom-up initiatives, indicating that developers have input into the design and planning of SPI. In addition, they value process ownership and empowerment. The most important motivational factors of senior management are visible success and meeting targets which are the basic practice of SPI does not prevent the company from meeting commercial and project goals. In order to execute the Process capability Profile based on the defined method like Process Maturity (PMAT), there is a sequence to be followed as shown in the Figure 5.1.

According to the figure 5.1, the approach has initiated with three basic characteristics namely growth stages, discipline value and model for business. This is available through previous information acquired using this method has focused for defining a Process Capability Profile (PCP). This method gets assistance from previous information for execution in the organization whereas the activity selecting the growth stage in the mind map and the associated discipline value. Once the execution begins there are various developments which should be analyzed on the basis of the business model till it reaches the final steps while it is probable for selecting the predefined profile which can be utilized as a general profile for the defined PCP. The profile of predefined has experienced modification in their startup definition based on the organization business goal of characteristics and document get utilized for guiding the particular practices that customized in every process. The model or standard has been chosen for aligning the profile with organization strategic goals.

The activities get established in terms of document sequences for accomplishing the capability profile with their practices whereas they guide and support experts by executing the approach. The method get satisfied on the basis of systematical achievement. The different sublevels for the defined PCP execution have been shown in figure 5.1. There are 8 sublevels for the execution of the defined PCP. However, the technique initially needs the following information pieces whereas the stage growth with the organization is adopted using the business model and defined discipline value. These information pieces have accomplished with method which allow the specific characteristic classification of VSE based on the required information type.

In order to define the information pieces of the method that has been incorporated for accessing must be utilized in the appraisal. Moreover, this case of growth stage definition for SPI has selected sublevels which assist the VSE at the defined PCP moments. Hence, feedback technique may be utilized from the senior employee and management of the organization constitute the essential components for the accomplishment of these sublevels. This feedback may be in terms of direct questionnaires with questions which assist information survey. During the interview with the employee, the feedback is made to modify through a list which specifies the essential items for every business model of VSEs. Similarly for the discipline value of the organization may be identified by focusing the executives through questioning them based on facilities to identify their difficulties, creative focus, price policy, generating quest for recent product, product quality feedback from client, product delivery quality to the client and focus on additional feedback from frequent and particular clients. This information assists in guiding the discipline value selection for better priority for the organization. This method provides a checklist for specified information in detecting the three required characteristics as the needed information for determining the essential standout which establish the main goal of PCP defined for supporting objective technique in appraisal of an organization.

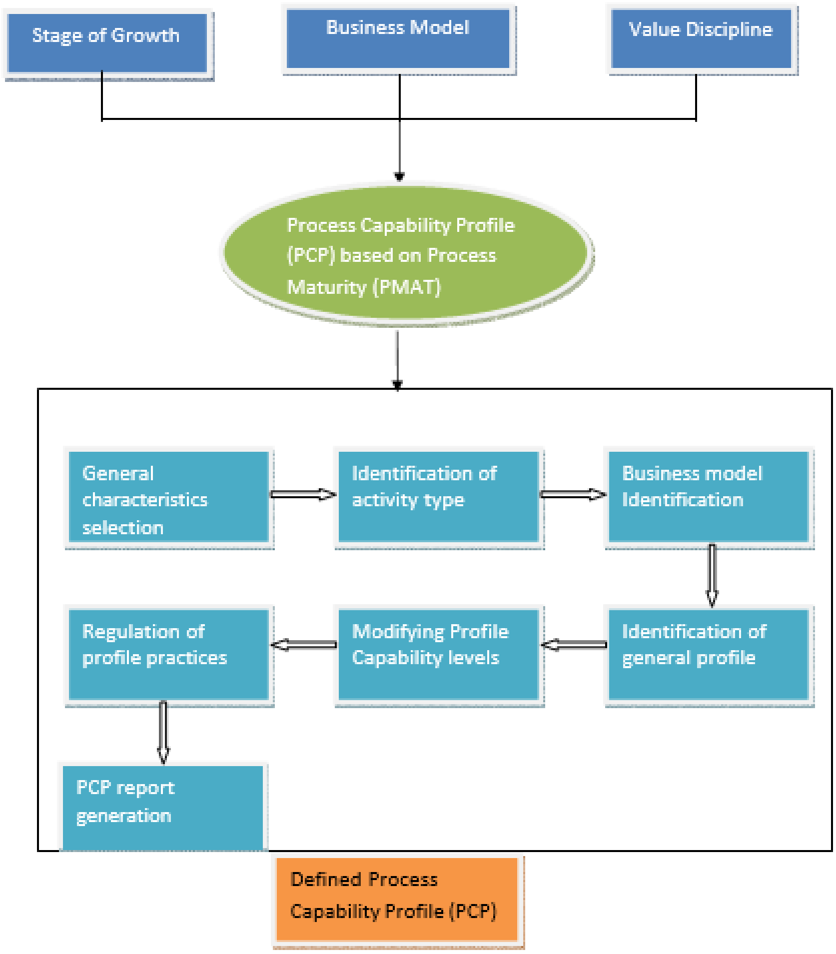

Figure 5.1 Proposed Model for Process Capability Profile based on Process Maturity (PMAT)

Based on the PCP defined, the number of characteristics to be analyzed is essential for defining a method for collecting this information pieces mainly with the business model whereas it gets subdivided in the operation field based on defined characteristics form the organization generally at the strategic definition moment of the organization goals. The information pieces with probable clear organization in the concept of defining for collecting this ideas classify them to permit use and location whereas the adopted concept in this research is the mind map which assists inducing the information associated in the manner to accomplish few classification sorting. This mind map concept gets formed using sublevels which are associated between them over lines. However, the individual sublevel with hierarchical structure is integrated from initial sublevel to final sublevels form crescent stages of details and precision. This generates the sublevels from initial to final is more usual and specific for mind mapping the characteristics. The 8 sublevels are listed below

- Sublevel 1 & 2 : General characteristics selection
- Sublevel 3 : Identification of activity type
- Sublevel 4 &5 : Business model identification
- Sublevel 6 : Identification of general profile
- Sublevel 7 : Modifying profile capability Levels
- Sublevel 8 : Regulation of profile practices

All these get associated over line by mind mapping and the report is generated using PCP report generation. The execution of this with the support of an organization as example gets processed in the organization with the Process capability Profile based on the defined method like Process Maturity (PMAT), there is a sequence to be followed

1. Sublevel 1 & 2: General characteristics Selection

The major goal of this activity is the selection of common characteristics associated with the improvement stage in the PCP based on the PMAT document and similarly the detection of discipline value to the organization. In order to determine an organization, the activity considered in the PCP based on PMAT user has selected sub level-1 which perform improvement stage identification whereas the sub level-2 get associated with discipline value. These sub levels activities have utilized for general characteristics selections in the organizations.

There are several models available but none of these has defined the stage of growth and discipline value whereas, only the defined PCP has proposed a feedback with checklist as the two questionnaires for supporting these sublevels. This is defined for every stage of growth with various general characteristics and shown in figure 5.2 which shows its selection based on the growth stage and discipline value of the executive in the organization.

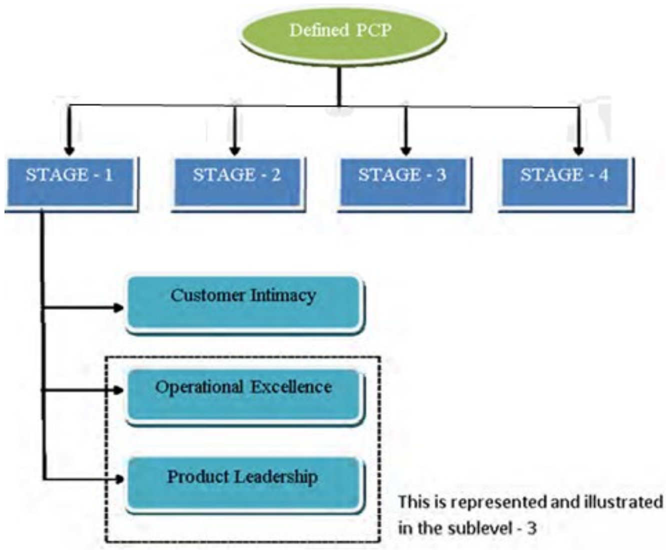

Figure 5.2 Select the general characteristics

2. Sublevel 3: Identification of activity type

This role focuses on discovering the activity type which needs attention for development in the organization whereas the activities can be segregated into two major types, namely, comprehensive service operation and comprehensive product operation. The operation for product is considered while the utilization gets distinct from the production but in services, it can be considered and depleted at an identical time and place which gets associated with the product use. This is the sub level-3 of the PCP based on PMAT model.

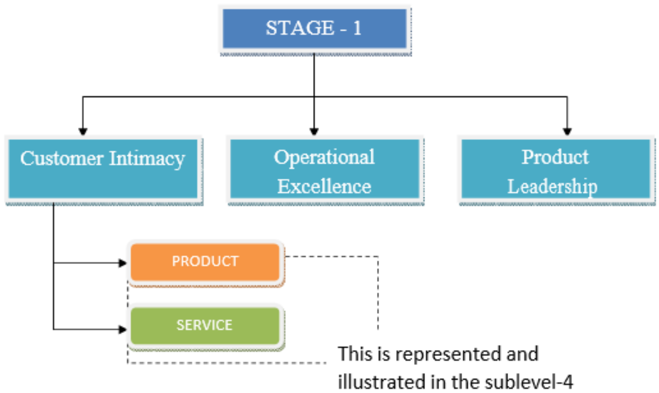

Figure 5.3 Identify type of activity

In this figure 3.3, the stage which consists of selected general characteristics whereas each characteristic consists of products and service that illustrate the sublevel 4 and 5.

3. Sublevel 4 & 5: Business Model Identification

The goal of this activity is to discover an appropriate business model for the organization whereas, in different situations, the organization has the ability to maintain, critical and more than a business model for developing their performance. In the case of PCP based on PMAT there is only one model that gets selected producing a major strategy in an organization. However, the organization which has more than a business model with similar significant features has the ability to analyze the mind map of PCP based on the PMAT model determines the combination identified with the capability profile. Therefore, this activity is accomplished using sublevel-4 and sublevel-5 whereas sublevel-4 get preferred for the organization with business model of common classification and sublevel-5 is to select the performance field from the identified business model.

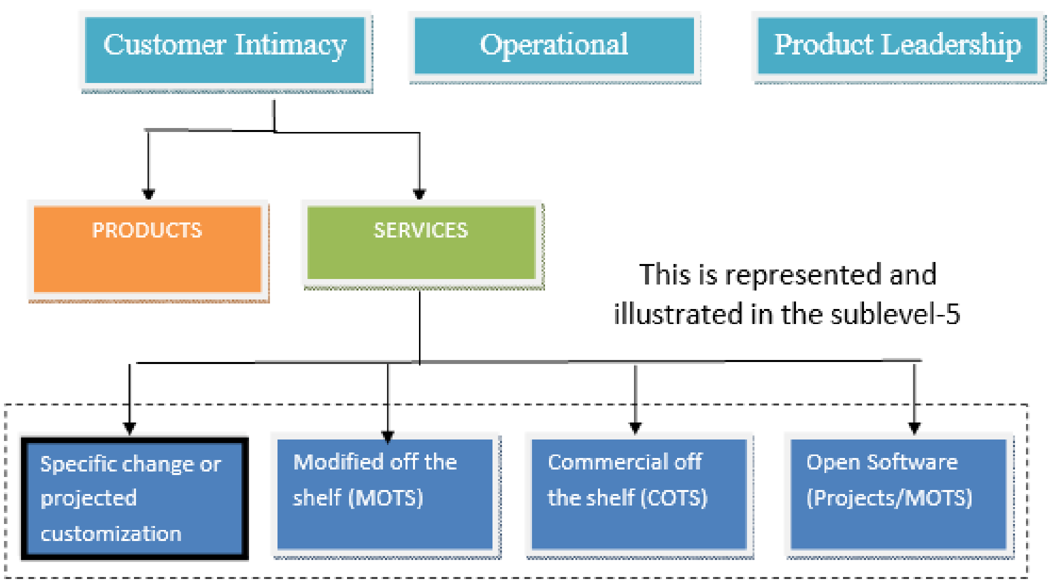

Figure 5.4 Business Model Identification

Figure 5.4 which illustrates sublevel-4 explains the various types of models selected by the organization on the basis of mind mapping. There are various business models available for understanding the requirement of the general characteristics whereas this method suggests a support questionnaire as the part of the mind map that permits decision making. Once the business model is selected using sublevel-4, sublevel-5 is used for selecting the operation areas of the business model. There are various operational areas, namely, complete software, embedded software, third party software customization and third party software partial development. In addition, the business model consists of two types, namely, with maintenance and without maintenance. The without maintenance model consists of those operational areas which are shown in figure 5.5.

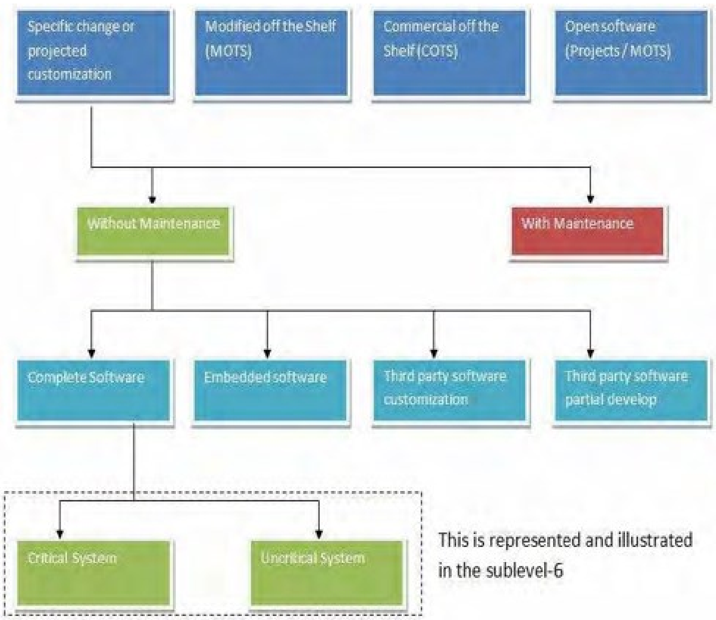

Figure 5.5 Selection of operational areas in Business Model

4. Sublevel 6: Identification of general Profile

The approach from SPI has utilized general PCP as the selected business model for focusing on this activity. At this sublevel-6, this activity is followed on the basis of the steps and information which are derived through a capability profile with general process that signifies the most relevant process that can be considered for the assessed organization is shown in figure 5.6.

This sublevel-6 activity, requires modification of the general selection of PCP and alteration based on the entrenched criteria for the next activity.

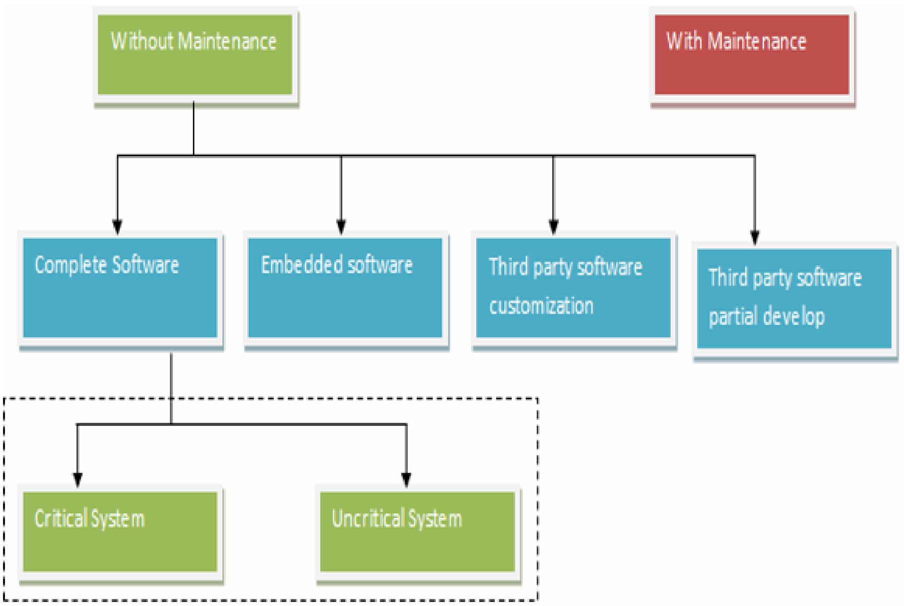

Figure 5.6 Identification of general Profile

5. Sublevel 7: Modifying profile capability Levels

This sublevel-7 activity has focused on modification or alteration of the capability level of every process of the general PCP to match the level based on organization requirement associated with the selected model whereas this activity during situations of two types in the organization. In order to accomplish the assessed level as per requirement of organization, the user has to modify the general PCP to suit the mind map of the organization. Similarly, modification has been done to the assessed level user for the level of capability greater than the demand of the organization.

Sublevel-7 activity consists of both the critical and the uncritical systems. This is shown in figure 5.7 with several general profiles for the executive in the certain operational areas. However, each general profile is analyzed and their capability levels are made to increase through training and work assessment from the management in the organization. The feedback from the appraisers will guide the management in placing of training to their executives in the organization.

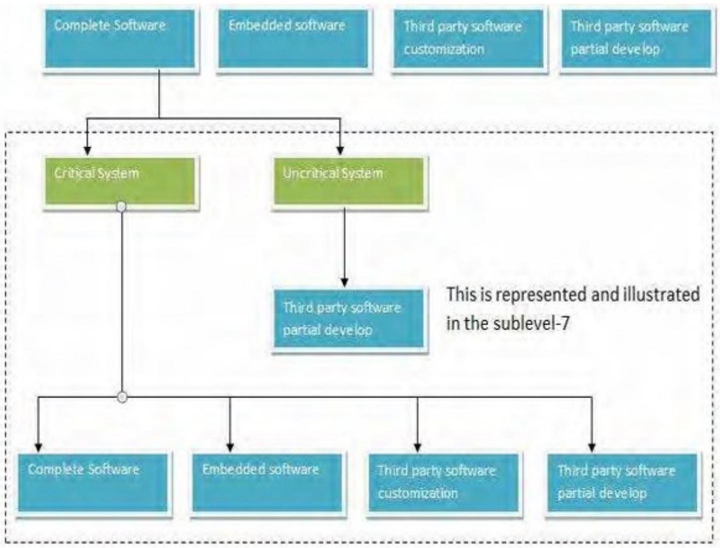

Figure 5.7 Modifying profile capability Levels

6. Sublevel 8: Regulation of profile practices

The goal of this activity is to analyze the process in detail and establish PCP for validating the essential features of every particular practice of this process to the organization. According to the identification processes, the operation of this activity relates to the goal analysis of each particular practice using the document named modification guide. Modification of these types have been performed according to the collected data from SPI approach and also with the executing team experiences. The modification in PCP is supported through mapping technique process aspects gets performed in the organization. The collection of recognized goals for every aspect is based on determined focus namely clients, financial, processes, quality and growth whereas these aspects are significant. In this sublevel-8, the regulation of the particular practices of the produced PCP focused for considering only for the practices that are legitimately relevant to the organization.

In an organization, the basic requirement from the executive is improved through the maturity level defined by the organization with the guidance of CMMI levels for better outcome which meet the customer requirement. This can be accomplished through improvement in the capability profile level of the executives through use of maturity level of management in the organization by providing the feedback and questionnaires analysis. Once the executives know their capability level, the organization tries to develop their capability level by proving maturity level training, soft skill training, leadership trains, project management training in order to improve the skills of the executive to meet the client requirements for the project in the organization. Therefore, this model is developed to enable analysis of the general characteristics of the service provider and development of the capability level of the profile by modifying or adjusting the level of the capability through process maturity for the respective process area. The three basic factors, namely, discipline value, stage growth and business model are essential key factors for improving the process area through implementation of a better model to suit identification of the capability of the executive, for improvement of the capability profile in each process. Thus the Process Capability Profile (PCP) is based on the process Maturity (PMAT) is an important business model for appraisal in each process area in the organization.

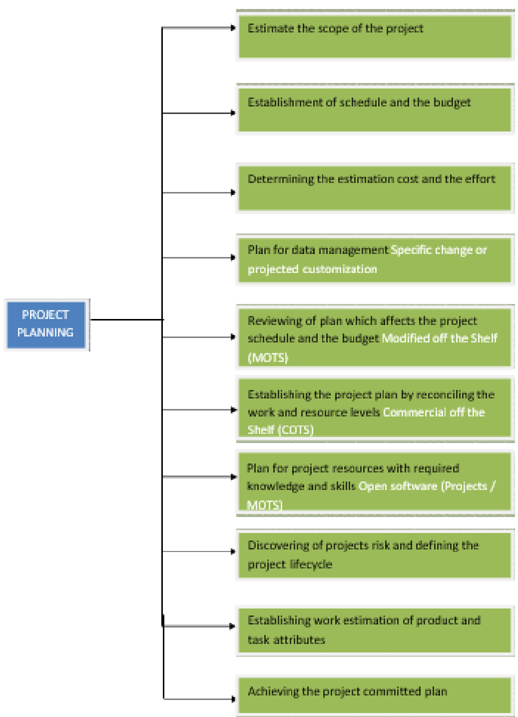

Figure 5.8 Regulation of profile practices

Project planning, one of the process area shown in figure 5.8, illustrates the identification of general profile and the activities in the respective profiles. Once each executive gets awareness of the work effectiveness and smart work to meet the client requirement due to the analysis of work capability, the organization need to monitor the capability level of the executive by using a questionnaire and feedback from the management and the executives. If the capability level does not meet the client requirements, the concerned executive is in need of training in that process area to meet the client needs. In this research work, the focus of the model is on executive service through appraisal for improving the capability profile defined in each process area based on the Process Maturity (PMAT) to support the organization of VSEs.

7. PCP report generation

The PCP based on the PMAT model illustrates all the eight sublevel activities which get implemented are summarized as reported with the description whereas this model provided a detailed explanation. Based on the eight sublevels, activities have shown an improvement in the growth of the capability profile levels using the representation of process maturity levels in the organization for the purpose of appraisal.

5.3 Summary

This chapter has dealt with the applicability of generated concepts that find provisional application in the pilot organization of partial effects found in this activity. This is the method of development that has been illustrated with Process Capability Profile (PCP) based on Process Maturity (PMAT). It also describes the components available in every process area and even in the generic goals which considerate these components as critical for utilizing the information. Moreover, all the model of CMMI been created from the CMMI Framework consists of both goals and practices which have been utilized for the generation of a model of CMMI that existed for CMMI constellations.

CHAPTER – 6
EXPERIMENTAL TECHNIQUES

6.1 Overview

This chapter discusses the fundamental concept that manipulates the factor of PMAT compared with other factors which affect the effort of software development adapted for the research study on evaluation of CMMI based Process. Depending on the variations in software product size, the impact of CMMI based process maturity level is selected for classifying the effort of software size with a suitable method. The predicting effort of software size is evaluated using the most well-known factor of the software product.

6.1.1 Process Maturity (PMAT) level in CMMI

The Process Maturity (PMAT) is rated in organization using software CMMI whereas the management of software product improvement is done through Software process settings utilized by all the organizations provided from CMMI on the basis of several requirements. However, CMMI indicates the need for software process instead of when it is utilized. Therefore, the focus of this Process Maturity is on their level based on the involvement from lower (level 1) to very high (level 5) whereas every level specifies the performance level which is expected from an organization. These are five maturity levels in CMMI namely

1. Level 1 : Initial
2. Level 2 : Managed
3. Level 3 : Defined
4. Level 4 : Quantitatively managed
5. Level 5 : Optimizing

This research work, examines the CMMI impact based on PMAT levels in the effort of software development using scale diseconomy and the productive rate of the standard project sizes.

6.2 COCOMO II Model

The Constructive Cost Model (COCOMO) is a familiar parametric cost estimating model in 1980s but, during, 1990s, COCOMO 81 model faced several problems and complexity in cost estimation software which tends to advance a novel process of the life cycle, namely, quick improvement process, non-sequential, technique of reuse driven and technique of object orientation. Therefore, COCOMO II has introduced in 1995 of the software engineering history with three sub model such as

1. Application composition
2. Early design
3. Post architecture

There are 17 Effort Multipliers (EM) or cost drivers as inputs for COCOMO II utilized for modifying the nominal effort of Person-Month (PM) in order to reflect the product of software which is being developed.

6.2.1 Effort Estimation

The estimation of effort expressed in terms of PM in COCOMO II has been defined as the amount of time spent by a person while working in the software project improvement for a month. This kind of estimating the effort using the equation is shown below

$$PM_{nominal} = A * SIZE^{E} * \prod_{i=1}^{N} EM_i$$

Where,

A = 2.94 = Multiplicative constant

N = Effort Multiplier numbers

E = Cumulative of Scale Factor (SF)

However, this multiple constant is derived from COCOMO team using the calibration of definite effort values of 165 projects recently collected from the database of COCOMO II whereas EM is utilized for modifying the effort.

6.2.2 Scale Factors (SF)

There are 17 Effort Multipliers (EM) or cost drivers as inputs of COCOMO II utilized for five SF sets considered for both scales, namely, scaling for economies and scaling for diseconomies in the projects of software development.

Scaling for economies

In the case of economy scaling, the software sizes get doubled as a result of the effort being less than twice the original sizes.

Scaling for Diseconomies

In the case of diseconomy scaling, the software sizes of the project resulted are more than the double the original project size effort that requires completion of the project.

The equation 6.1 and 6.2 are used to manipulate the project effort scale for economies or diseconomies:

$$E = B + 0.01 * \sum_{j=1}^{N} SF_j$$

Where,

B = a constant = 0.91

The multiple constant is derived from COCOMO team using calibration of the definite effort values of 165 projects is recently collected from the database of COCOMO II, whereas the exponent E in equation 6.2 is cumulative of five SF's. However, each SF has rating levels from Very Low (VL) to Extra High (XH) with every rating level having a quantitative value with weight (W) utilized in the COCOMO II technique. 5 SFs of COCOMO II represented in Table 6.1 are Risk Resolution, Precedentedness, Team Cohesion, Development Flexibility and PMAT. In order to determine the PMAT procedure, the interest factor in this investigation is organized from Software Engineering Institute's Capability Maturity Model (SEI-CMM).

The rating level of maturity in the COCOMO II is similar to CMM except for the first level of maturity rating that has been segmented into two parts as equal named as lower and upper in COCOMO II shown in Table 6.1. This research classifies the CMM maturity level 1 as lower and upper. The focus of the lower level 1 is on the job of the organization but with no focus on the process and documentation of lesson learnt. Upper level 1 is for an organization which implements several needs that satisfy level 2 of CMM. Published definition of CMM's level 1 (lower) and (upper) are grouped into level 1.

Table 6.1 Rating levels and values for COCOMO II scale factors

Scale factor	CMMI LEVELS					
	Level 1 - Lower	Level 1 - Upper	Level 2	Level3	Level4	Level 5
	Very Low	Low	Nominal	High	Very High	Extra High
Risk Resolution (RESL)	7.08	5.60	4.30	2.85	1.44	0.00
Precedentedness (PREC)	6.33	5.01	3.76	2.50	1.28	0.00
Team Cohesion (TEAM)	5.52	4.41	3.33	2.23	1.18	0.00
Development Flexibility (FLEX)	5.09	4.08	3.07	2.15	1.06	0.00
ProposedProcess Maturity (PMAT)	7.85	6.15	4.76	3.22	1.64	0.00

6.3 Hypothesis of the research

This study has focused on the hypothesis of level increase in the CMMI based on PMAT get resulted with the following:

1. The effort made in the software development is reduced.
2. The productive rate is increased.
3. Scaling for diseconomy is reduced.

As mentioned before, this research depends extremely on the proposed PMAT rating values of CMMI shown in Table 6.2.

Table 6.2 Proposed Process Maturity values.

PMAT Description	CMMI LEVELS					
	Level 1 - Lower	Level 1 - Upper	Level 2	Level3	Level4	Level 5
Rating levels	VeryLow	Low	Nominal	High	VeryHigh	Extra High
Proposed Process Maturity(PMAT) values	7.48	5.69	3.77	2.05	1.06	0.00

The consideration of CMM level 2 in the COCOMO II with nominal rating for PMAT and a similar consideration is done to the CMMI level 2. Thus, the effort, scaling for diseconomy and productivity percentage are increased or decreased compared to the nominal CMMI based PMAT ratings.

6.4 Evaluating CMMI based Process

CMMI based on PMAT impact has been examined and evaluated according to the software project sizes which is suggested for classifying the software sizes in a suitable manner due to size consideration of a significant factor in effort prediction of the software product. The classification of product size of the software namely, small, intermediate, medium, large and very large is shown in Table 6.3. The SF of PMAT gets utilized for seizure of the various PMAT level has an impact on the effort of software development to the standard project sizes details are given below.

Table 6.3 Software project sizes

Classification of projects	Sizes in KLOC
Small (S)	3
Intermediate (I)	9
Medium (M)	33
Large (L)	129
Very Large (VL)	513

6.4.1 Diseconomy of Scale

Scaling of diseconomy has been indicated with a relative increase in effort compared to the size that increased in a software product was doubled in size of the project size resulting in twice of the effort in the original project required for completing the projects. As per researcher's discussion, the standard project sizes is mentioned in terms of two thousand lines of codes (2 KLOC) and then the ratio from Small (S) to Intermediate (I), Small (S) to Medium (M), Small (S) to Large (L) and Small (S) to Very Large (VL) called from small to intermediate (from S to I), from small to medium (from S to M), from small to large (from S to L), and from small to very large (from S to VL). The small size effort in scaling for diseconomy shown in table 6.4 is segregated on the basis of the corresponding efforts.

Table 6.4 Project Size ratio of the Software

Project Classification	Sizes ratio
Small (S) to Intermediate (I)	6
Small (S) to Medium (M)	18
Small (S) to Large (L)	66
Small (S) to Very Large (VL)	258

6.4.2 Effort Estimation

In this research work, the fundamental design used in the evaluation of the PMAT compared with other factors impact manipulation of the effort of software development whereas the implementation of other isolated factors from PMAT impact is due to various improvement types considered together in the organization. However, the project management cannot found the solution for the quantum of development from PMAT compared with other factors. In this research work, COCOMO II model has been introduced for estimating the software development effort. In addition, the EM is made considered as nominal. This is illustrated in table 6.5 with all SF's except PMAT also considered as nominal. Therefore, the PMAT potential efforts present in the effort of software development are confirmed by setting the EM and SF as nominal. The calculation of nominal PMAT rating and standardized project sizes from small to very large is done using equations 6.1 and 6.2.

$$PMAT_{no\min al} = A * SIZE^{E}$$

Equation 6.3 is applied to project sizes of all types manipulated but SF can be considered using Table 6.5 which is shown below

Table 6.5 Scale factor in nominal value for all rating levels

Table 6.5 Scale factor in nominal value for all rating levels

Scale factor	CMMI LEVELS					
	Level 1 - Lower	Level 1 - Upper	Level 2	Level 3	Level 4	Level 5
	Very Low	Low	Nominal	High	Very High	ExtraHigh
Precedentedness (PREC)	3.83	3.83	3.83	3.83	3.83	3.83
DevelopmentFlexi-bility (FLEX)	3.15	3.15	3.15	3.15	3.15	3.15
Risk	4.36	4.36	4.36	4.36	4.36	4.36
Resolution (RESL)						
Team Cohesion (TEAM)	3.34	3.34	3.34	3.34	3.34	3.34
Proposed Process Maturity (PMAT)	7.45	5.61	3.71	2.18	1.13	0
Sum of all SF	22.13	20.29	18.39	16.86	15.81	14.68

According to equation 6.3, the nominal values of PMAT calculated for all kind of project sizes are shown below

Small Project size

$$PMATno \min al = 2.94 * 3^{0.91+0.01x18.39}$$

$$PMAT_{no \min al} = 9.78$$

Intermediate Project size

$$PMATno \min al = 2.94 * 9^{0.91+0.01x18.39}$$

$$PMAT_{no \min al} = 28.85$$

Medium Project size

$$PMATno \min al = 2.94 * 33^{0.91+0.01x18.39}$$

$$PMAT_{no \min al} = 134.73$$

Large Project size

$$PMATno \min al = 2.94 * 129^{0.91+0.01x18.39}$$

$$PMAT_{no \min al} = 598.57$$

Very Large Project size

$$PMATno \min al = 2.94 * 513^{0.91+0.01x18.39}$$

$$PMAT_{no \min al} = 2709.85$$

The nominal values of projects of all sized are mentioned whereas the very low, low, high, very high and extra high rating level of the projects are manipulated in a similar manner.

6.4.3 Productivity Rate

The research hypothesis is tested with CMMI based on PMAT rating level which increases the productivity of the organization and is directly proportional and applicable to every effort estimation used in equation 6.4 shown below

$$\text{Rate of productivity} = \frac{size}{Effort}$$

Where,

Size = Standard project sizes

Effort = Effort estimation of every PMAT level to all standard project sizes However, the size is measured in terms of KLOC and the effort estimated in each PMAT level to all standard sizes. But the productivity figures for all the standard sizes in the nominal rating level have been calculated as follows

Small Project size

$$Productivity = \frac{3}{9.78}$$

$$Productivity = 306.74$$

Intermediate Project size

$$Productivity = \frac{9}{28.85}$$

$$Productivity = 311.96$$

Medium Project size

$$Productivity = \frac{33}{134.73}$$

$$Productivity = 244.93$$

Large Project size

$$Productivity = \frac{129}{598.57}$$

$$Productivity = 215.51$$

Very Large Project size

$$Productivity = \frac{513}{2709.85}$$

$$Productivity = 189.31$$

In this nominal rating, productivity may decrease from small projects to very large projects as the result of the number of projects handled but, in the case of increase in maturity level, the productivity in all standard project sizes is increased. As the level increases from level 1 lower to level 5, the productivity is increased due to improvement in the capability level of the executives based on the maturity level in the process of an organization. Therefore, the productivity of an organization depends entirely on maturity level that improves the Process Capability Profile (PCP).

6.5 Summary

This chapter has dealt with the evaluation measurement of effort estimation, scaling for diseconomy and productive rate of the CMMI based on process maturity rating level of all rating levels for all standard project size and the results of this evaluation are discussed in the next chapter.

CHAPTER – 7
RESULTS AND DISCUSSIONS

7.1 Overview

This chapter has taken up PMAT rating level of the organization as the subject considering handling of all standard project sizes. In this research work, the evaluation of all factors with PMAT factor using EM and SF has been done and implementation has been done with the best factors of PMAT with all standard project sizes along with the CMMI based on PMAT rating level. This is an illustration of the defined PCP based on PMAT with a nominal rating level. Factors of this kind are required for implementation in VSE for effecting improvements in the PCP of an organization through improvements in the maturity level of the software development process for execution using the appraisal concept. This proposed method has built individual responsibility, awareness of documentation and lessons learnt for the development of the capability profile in the process area of the organization. The focus of this proposed method is on the implementation of CMMI level 2 for the use of VSEs in meeting the client requirement as fulfill and following the standardization of the software engineering institute (SEI).

7.2 PMAT rating level evaluation

The results were evaluated and compared with these of the maturity rating level based on effective measurement required for the study.But this investigation has dealt with changes in the percentage of the software development parameters such as effort estimation, scaling for diseconomy and productivity rate. The rates of percentage changes represented may be either increase or decrease in scaling for diseconomy, effort estimation and productivity rate. These significant factors have measured to determine an importance of various effects in the maturity level process of effort improvement to all software projects of all standard sizes. Computation these change of percentage haspretended the PMAT nominal rating in the base levels. Therefore, the changes in percentage of effort estimation, scaling for diseconomy and productivity are calculated using equation 7.1

$$Change\ of\ \ percentage = \frac{parameter\ - Parameter_{no\min al}}{Parameter_{no\min al}} \times 100$$

The term "Parameter" refers to the effort value computed and scaling for diseconomy and productivity at a specific rating level of PMAT, while "$Parameter_{No\min al}$" refers to the nominal value of a similar parameter. The resulting value is a combination of the two changes, namely, positive and negative. A negative value indicates reduction in the percentage of the "Parameter" value while a positive value is indicative of an increase in the percentage of "$Parameter_{No\min al}$" value.

7.2.1 Effort Estimation

The effort for all standard software project sizes in the implementation model has analyzed using COCOMO II is resulted in every PMAT rating levels have shown in Table 6. Similarly change in percentage of effort with every PMAT to all standard software project sizes is presented in table 7 are done through equation 7.1.

Table 7.1 Estimated efforts in all CMMI-based PMAT ratings for all standard sizes

Project Classification	Effort estimation in CMMI based on PMAT rating level						
	Size	Very Low	Low	Nominal	High	Very High	ExtraHigh
Small	3	9.93	9.86	9.78	9.32	9.22	9.17
Intermediate	9	30.21	29.44	28.85	27.38	26.36	25.22
Medium	33	149.38	140.67	134.73	127.98	120.82	114.21
Large	129	706.65	643.78	598.57	541.59	514.39	489.46
Very Large	513	3359.47	2992.24	2709.85	2387.96	2235.45	2097.18

The equation 7.1 is applied for small standard project size for all rating as an example to create the change of percentage in effort estimation as shown in table 7.2.

Very low rating for small size

$$Change\ in\ \ percentage = \frac{(9.93 - 9.78)}{9}.78 \times 100 = 1.53$$

Low rating for small size

$$Change\ in\ \ percentage = \frac{(9.86 - 9.78)}{9}.78 \times 100 = 0.82$$

Nominal rating for small size

$$Change\ in\ \ percentage = \frac{(9.78 - 9.78)}{9}.78 \times 100 = 0.00$$

Very high rating for small size

$$Change\ in\ \ percentage = \frac{(9.22 - 9.78)}{9}.78 \times 100 = -\ \ 5.73$$

Extra high rating for small size

$$Change\ in\ \ percentage = \frac{(9.17 - 9.78)}{9}.78 \times 100 = -\ \ 6.23$$

Table 7.2 Change in percentage of effort in CMMI based on PMAT ratings for all standard project sizes

Project Classification	Effort estimation in CMMI based on PMAT rating level						
	Size	Very Low	Low	Nominal	High	Very High	Extra High
Small	3	1.53	0.82	0.00	-4.70	-5.73	-6.23
Intermediate	9	4.71	2.05	0.00	-5.10	-8.63	-12.58
Medium	33	10.87	4.41	0.00	-5.01	-10.32	-15.23
Large	129	18.06	7.55	0.00	-9.52	-14.06	-18.23
Very Large	513	23.97	10.42	0.00	-11.88	-17.51	-22.61

The result has represented every development in the PMAT rating with reduction in effort estimation with increase in the rating level in all software project sizes which can be easily seen while these developments in the effort are related to the size of large projects than of these of small projects.

Changes in percentage are shown in Table 7.2. The effort needed with changes in percentage show 1.53% increase for PMAT rating with very low value and a decrease of 6.23% from the nominal level for a rating of extra high in small project sizes. But, in the case of very large projects, the effort needed for change in percentage shows variations from an increase of 23.97% in PMAT rating level of very low value and a decrease of 22.61% from the nominal level for a rating level of extra high.

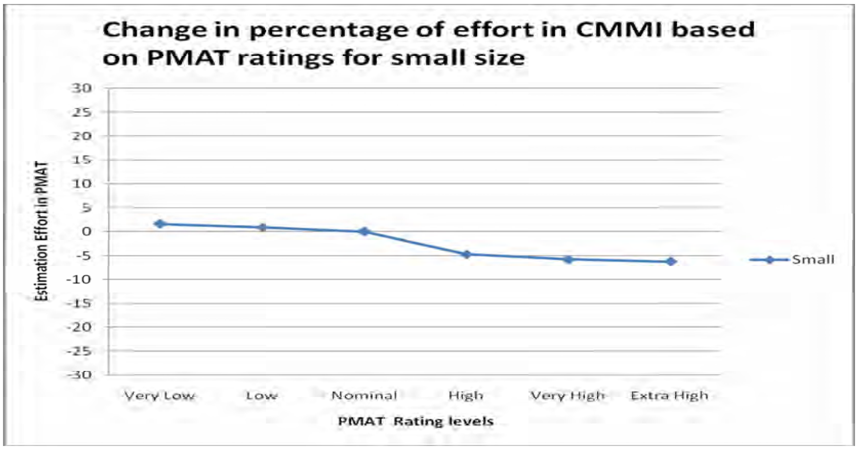

Figure 7.1 Change of Percentage in effort estimation in PMAT ratings levels for small project size

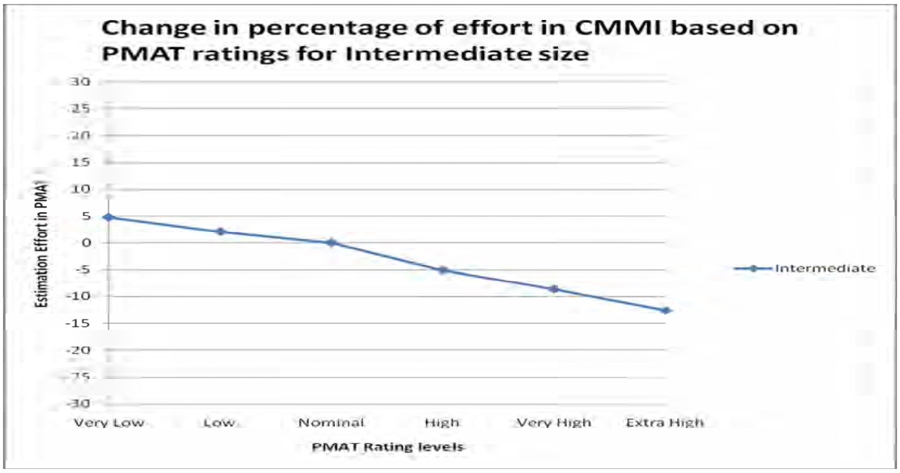

Figure 7.2 Change of Percentage in effort estimation in PMAT ratings levels for small project size

The change of percentage in effort estimation for small and intermediate have shown in figure 7.1 and 7.2 whereas the change of percentage in small project sizes is increased with 1.53% and decreases with 6.23% from the nominal level. Similarly in the case of intermediate project sizes have increases with 4.71% and decreases with 12.58% from the nominal level of PMAT rating.

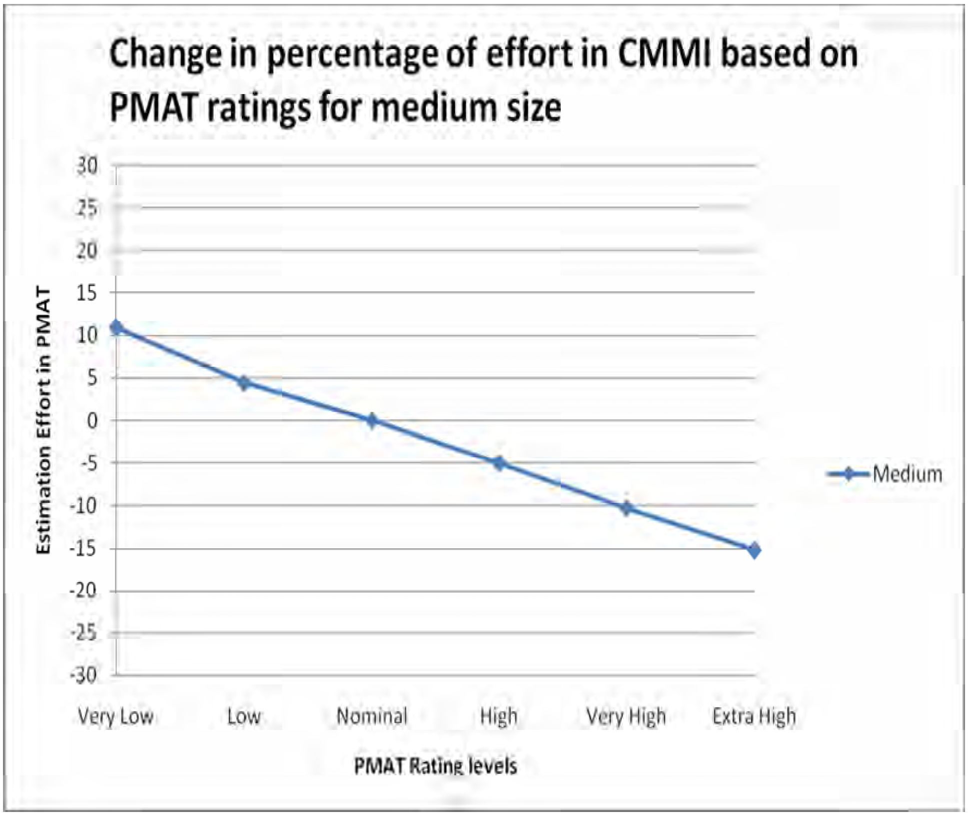

Figure 7.3 Change of Percentage in effort estimation in PMAT ratings levels for medium project size

Figure 7.3 shows, the changes in percentage as to be linear with increase with the maximum of 10.87% and decrease with the maximum of 15.23% from the nominal level of PMAT rating. Similarly in the case of Large and very large undertakings the changes in percentage increases are with 18.06% and 23.97% whereas there are decreases with 18.23% and 22.61% from the nominal value of PMAT rating as shown in figure 7.4 and figure 7.5.

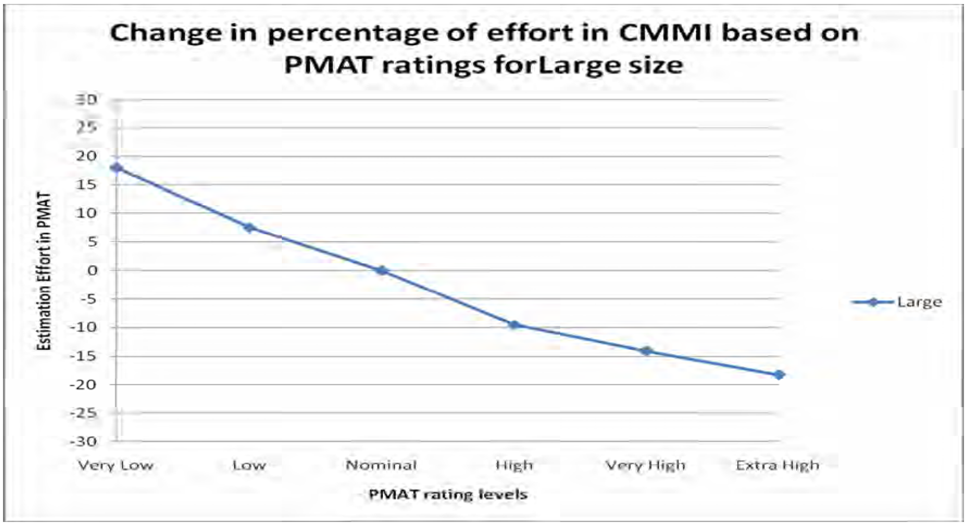

Figure 7.4 Change of Percentage in effort estimation in PMAT ratings levels for large project size

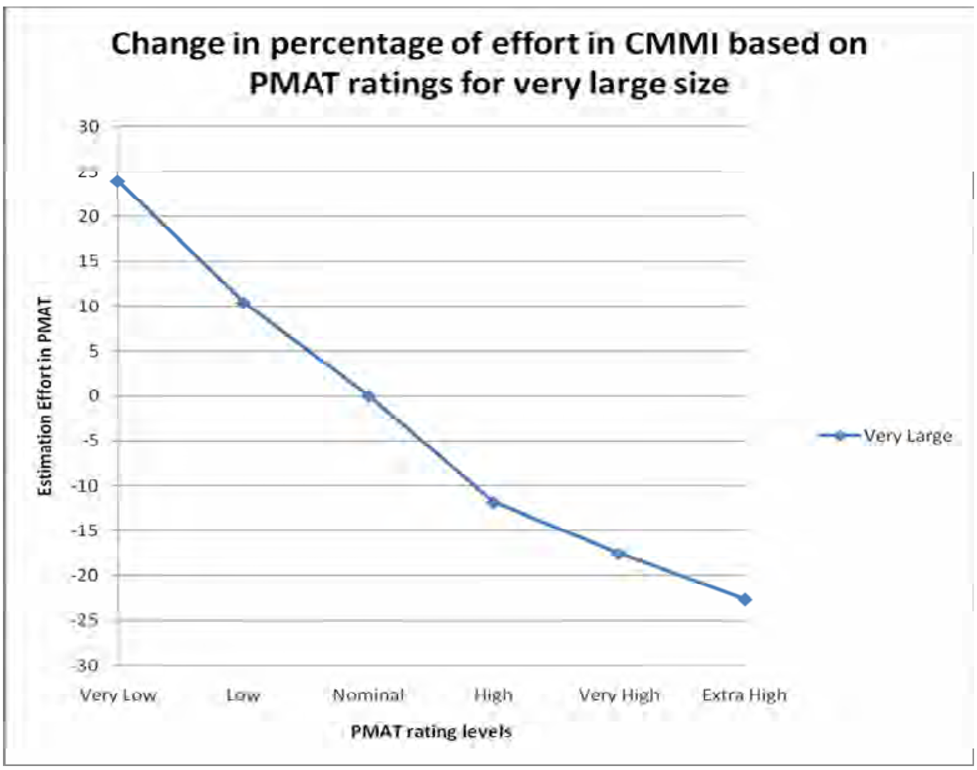

Figure 7.5 Change of Percentage in effort estimation in PMAT ratings levels for very large project size

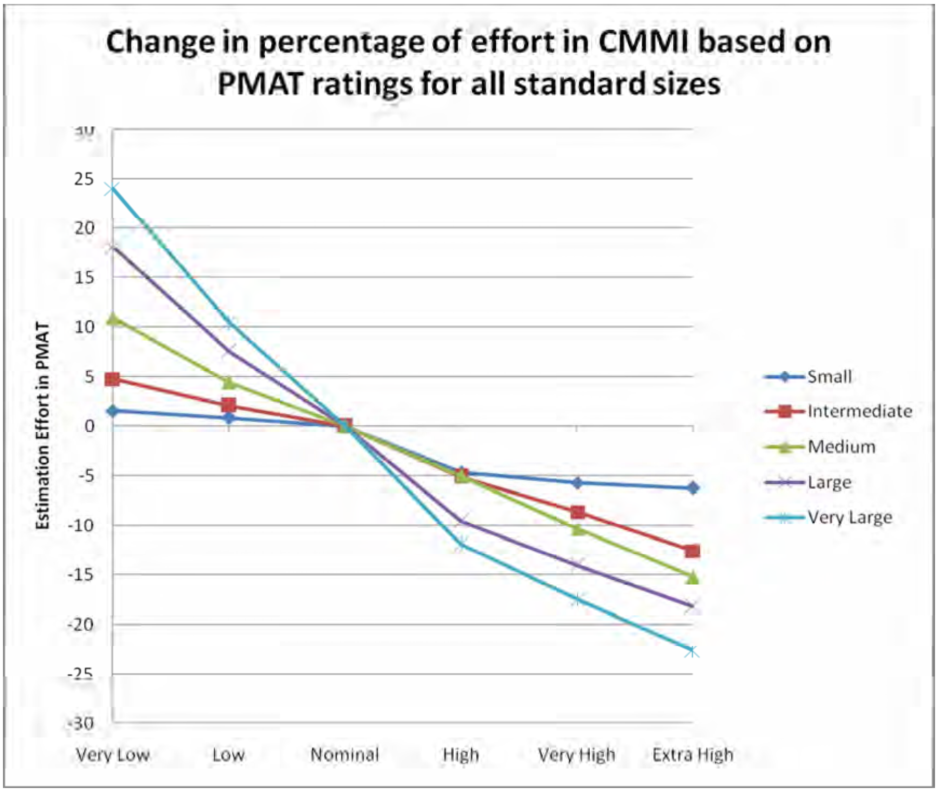

Figure 7.6 Change of Percentage in effort estimation in PMAT ratings levels for all standard project size

Variations in the percentage changes in the efforts in PMAT rating levels seen are from increase in percentage to decrease in percentage in all standard sizes of projects as shown in the figure 7.6. The change in percentage differs according to projects size classification but the effort from the very low level is more compared to the effort from extra high due to maturity level of CMMI being less. Therefore, the SMEs need to increase the process maturity level to enable development of the Process Capability Profile (PCP) of the executives using this method. This is an appropriate method for appraisal to understand the level of maturity in the process area of an organization.

7.2.2 Scaling for diseconomy

The sizes of projects standard divided between small project sizes and also the effort estimation of the standard projects with the respective efforts of small size for visualizing the scaling for diseconomy are shown in the calculation below.

The scaling for diseconomy is shown as the plotted size ratio in Table 7.3. The changes in percentage in scaling of diseconomy of scale in every PMAT rating level for all standard size projects are shown in Table 7.4.

$$Scaling\ \ for\ \ Diseconomy\ in\ sizes = \frac{S\ tan\ dard\ \ project\ sizes}{small\ project\ sizes}$$

The calculations of ratings for small to medium sizes are shown as an examples with scaling of diseconomy in sizes below

$$Scaling\ for\ \ Diseconomy\ in\ S\ to\ M\ (size) = \frac{33}{3} = 11$$

Similarly calculation of small to medium efforts is shown as an example with scaling of diseconomy in sizes are shown below

$$Scaling\ for\ \ Diseconomy\ in\ S\ to\ M(effort\ \ for\ low\ level) = \frac{140.67}{9.86} = 14.27$$

Table 7.3 Scaling for diseconomy in CMMI based PMAT ratings levels for all standard sizes

Project Classification	Size Ratio	Effort estimation in CMMI based on PMAT rating level					
		Very Low	Low	Nominal	High	VeryHigh	ExtraHigh
Small to Intermediate	3	3.04	2.99	2.95	2.94	2.86	2.75
Small to Medium	11	15.04	14.27	13.78	13.73	13.10	12.45
Small to Large	43	71.16	65.29	61.20	58.11	55.79	53.38
Small to Very Large	171	338.32	303.47	277.08	256.22	242.46	228.70

Table 7.3 shows the values of sizes and effort estimation in which the standard sizes from Intermediate to very large are divided into small size and effort estimation value. This assists creation of a change in percentage of scaling for diseconomy in PMAT rating level.

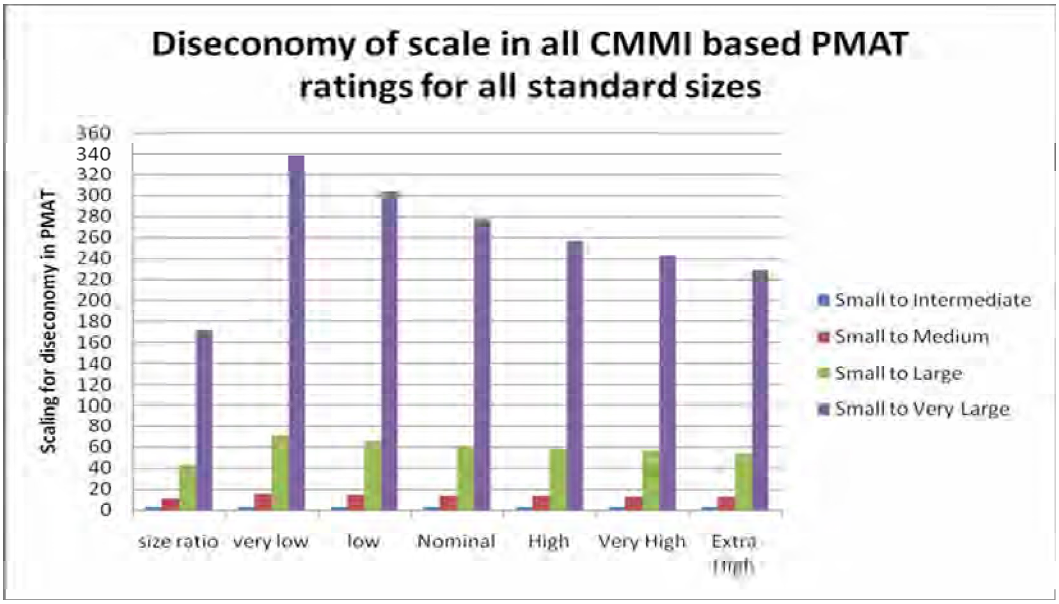

Figure 7.7 Scaling for diseconomy in all PMAT ratings for all standard sizes

Table 7.4 Percent change of diseconomy of scale in all CMMI based PMAT ratings for all standard sizes

Project Classification	Size Ratio	Effort estimation in CMMI based on PMAT rating level					
		Very Low	Low	Nominal	High	Very High	ExtraHigh
Small to Intermediate	3	3.05	1.36	0.00	-0.34	-3.05	-6.78
Small to Medium	11	9.14	3.56	0.00	-0.36	-4.93	-9.65
Small to Large	43	16.27	6.68	0.00	-5.05	-8.84	-12.78
Small to Very Large	171	22.10	9.52	0.00	-7.53	-12.49	-17.46

Table 7.4 Percent change in diseconomy of scale in all CMMI based PMAT ratings for all standard sizes.

Figure 7.7 shows the scaling of diseconomy done even for size ratio and PMAT rating levels for all project size classification. Figure 7.7 shows small to intermediate category as low in scaling ratio in diseconomy but in the case of small to very large categories, scaling ratio in diseconomy is very high in projects sizes (2KLOC). In all the project classification, the PMAT rating of very low level has a high value of scaling for diseconomy in all the standard projects sizes which explains the CMMI maturity level is needed for more concentration in very low.

Figure 7.8 and figure 7.9 present the changes in percentage in scaling for diseconomy in all CMMI PMAT rating levels with increase and decrease in percentages in project sizes namely, Small to Intermediate and Small to Medium.

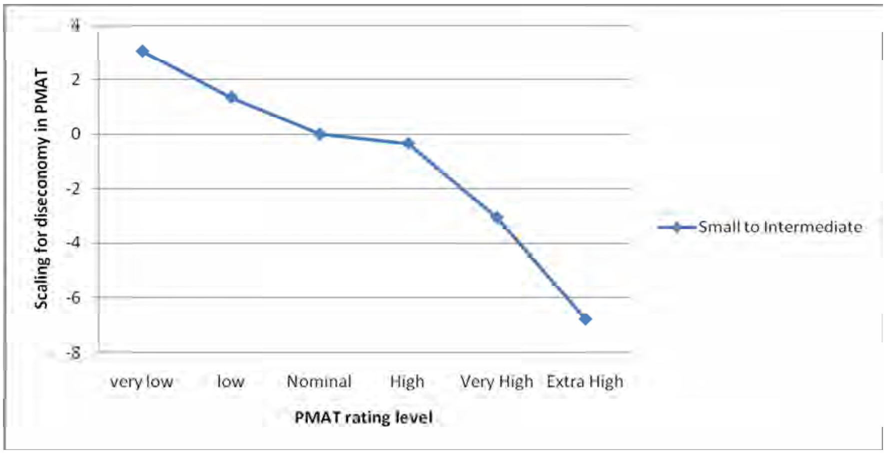

Figure 7.8 Change of percentage in scaling for diseconomy in all PMAT ratings for Small to Intermediate sizes.

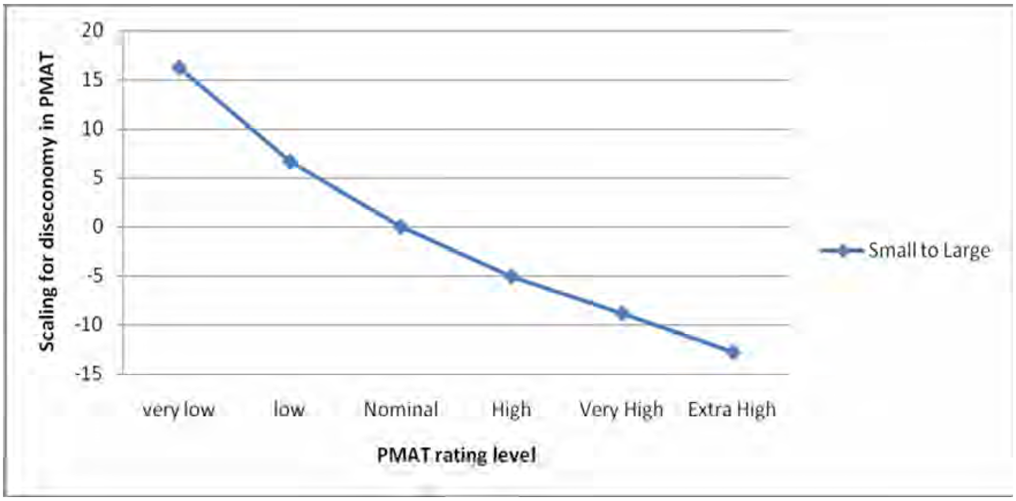

Figure 7.10 Change of percentage in scaling for diseconomy in all PMAT ratings for Small to Large sizes.

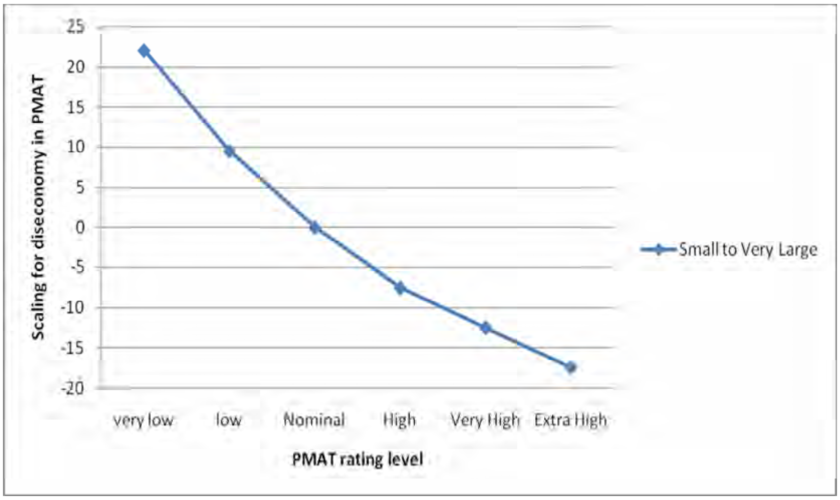

Figure 7.11 Change of percentage in scaling for diseconomy in all PMAT ratings for Small to Very Large sizes.

Figure 7.10 and figure 7.11 present the changes in percentage in scaling for diseconomy in all CMMI PMAT rating levels with increase and decrease in percentages in project sizes, namely, Small to Large and Small to Very Large. However, the increase in and decrease in percentages are high in small to very large is more while compare to other project sizes.

Figure 7.12, shows changes in percentage in CMMI PMAT rating level with different project sizes. The increase in change in percentage is low in Small to Intermediate and decrease in change in percentage is low in Small to Intermediate. The difference in percentage from the nominal level of the PMAT rating increases by 3.05% and decreases by 6.78% but, in the case of small to very large enterprises, difference in percentage from the nominal level of PMAT rating increases by 22.10% and decreases by 17.46%. The scaling for diseconomy in PMAT rating level is more in the small to very large project sizes with overall change in the percentage of 39.56% but, in the case of Small to Intermediate, it is 9.83%. The scaling for diseconomy is high in the small to very large project sizes but, in the case of maturity rating level, extra high shows a decrease in scaling for diseconomy as presented in figure7.12 with better scaling ratio than for the very low rating level of maturity. Therefore, the maturity level is better than for the scaling ratio of diseconomy.

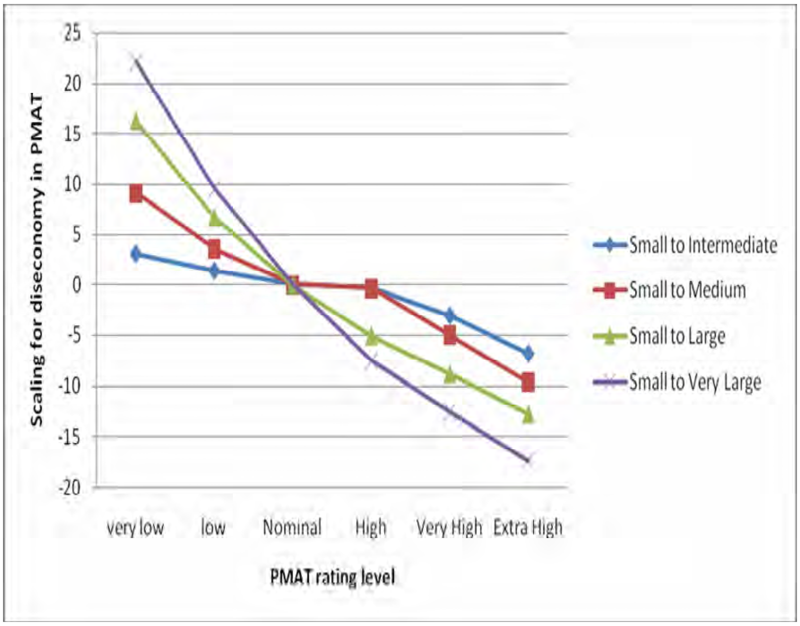

Figure 7.12 Change of percentage in scaling for diseconomy in all PMAT ratings for all project sizes.

7.3 Productivity Rate

The ratio of the size to the effort is considered as productive rate. These values after the calculation for all standard project sizes with all CMMI based on PMAT maturity rating level is shown in Table 7.5. Once the individual productivity rate is calculated in all project sizes for all CMMI, PMAT rating levels are obtained using equation 6.4. The change in percentage over productivity rate in all CMMIs based on PMAT rating levels for all project sizes provide a clear indication of the productivity rate changes (increases) as more rapid for larger projects compared to smaller projects. Table 7.6, shows the change in percentages in productivity rate in all CMMI based on PMAT rating levels for all project sizes. But, the percentage change is the reversal of the productivity rate. The estimation effort is inversely proportional to the productivity rate and is derived and presented in table 7.6 as decrease in percentage change before nominal rating level which means very low and low rating levels of PMAT simultaneous increase in the change in percentage applicable in high, very high and extra high rating level of PMAT.

Table 7.5 Productivity rate in CMMI based PMAT ratings for all standard sizes.

Project Classification	Effort estimation in CMMI based on PMAT rating level						
	Size	**Very Low**	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Small	3	302.11	304.26	306.75	321.89	325.38	327.15
Intermediate	9	297.91	305.71	311.96	328.71	341.43	356.86
Medium	33	220.91	234.59	244.93	257.85	273.13	288.94
Large	129	182.55	200.38	215.51	238.19	250.78	263.56
Very Large	513	152.70	171.44	189.31	214.83	229.48	244.61

The equation 7.1 is applied for small standard project size for all rating as an example to create the change of percentage in productivity as shown in table 7.6.

Very low rating for small size

$$Change\ in\ Percentage = \frac{(302.11 - 306.74)}{306}.74 \times 100 = 1.53$$

Low rating for small size

$$Change\ in\ Percentage = \frac{(304.26 - 306.74)}{306}.74 \times 100 = 0.82$$

Nominal rating for small size

$$Change\ in\ Percentage = \frac{(306.74 - 306.74)}{306}.74 \times 100 = 0.00$$

High rating for small size

$$Change\ in\ Percentage = \frac{(321.89 - 306.74)}{306}.74 \times 100 = -\ 4.70$$

Very high rating for small size

$$Change\ in\ Percentage = \frac{(325.38 - 306.74)}{306}.74 \times 100 = \text{-}5.73$$

Extra high rating for small size

$$Change\ in\ Percentage = \frac{(325.38 - 306.74)}{306}.74 \times 100 = \text{-}6.23$$

Table 7.6 Percentage change of productivity in CMMI based PMAT ratings for all standard sizes

Project Classification	Effort estimation in CMMI based on PMAT rating level						
	Size	VeryLow	Low	Nominal	High	VeryHigh	ExtraHigh
Small	3	-1.51	-0.81	0.00	4.94	6.07	6.65
Intermediate	9	-4.50	-2.00	0.00	5.37	9.45	14.39
Medium	33	-9.81	-4.22	0.00	5.27	11.51	17.97
Large	129	-15.29	-7.02	0.00	10.52	16.37	22.29
Very Large	513	-19.34	-9.44	0.00	13.48	21.22	29.21

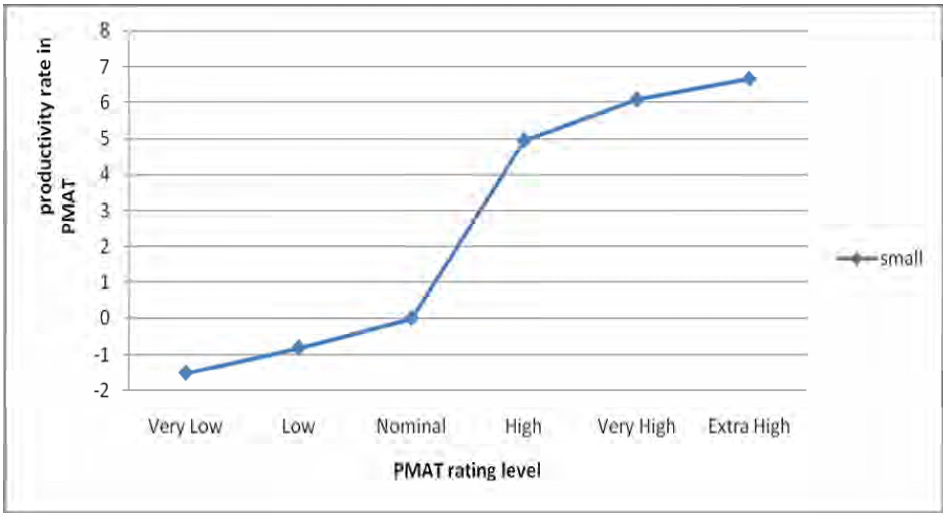

Figure 7.13 Change of percentage in productivity rate with all PMAT ratings for small project sizes.

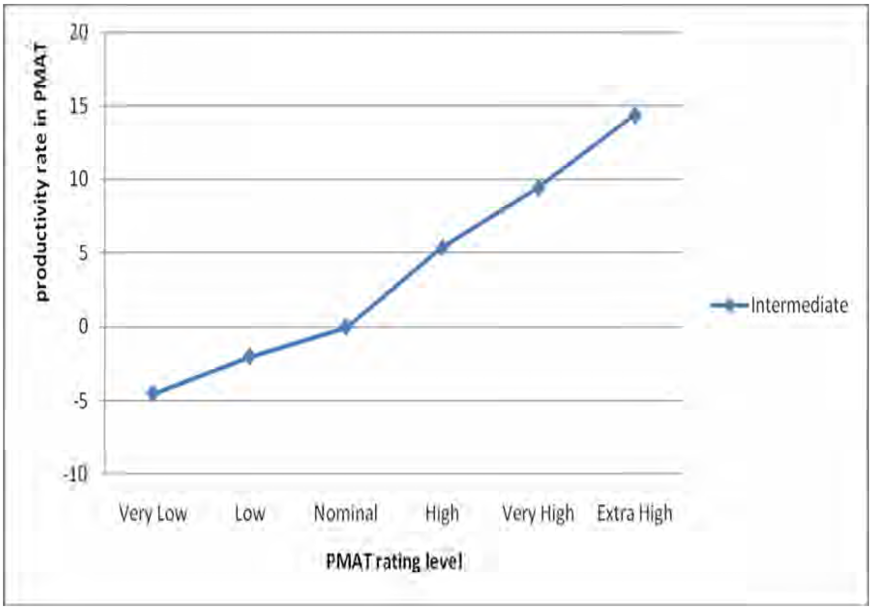

Figure 7.14 Change of percentage in productivity rate with all PMAT ratings for Intermediate project sizes.

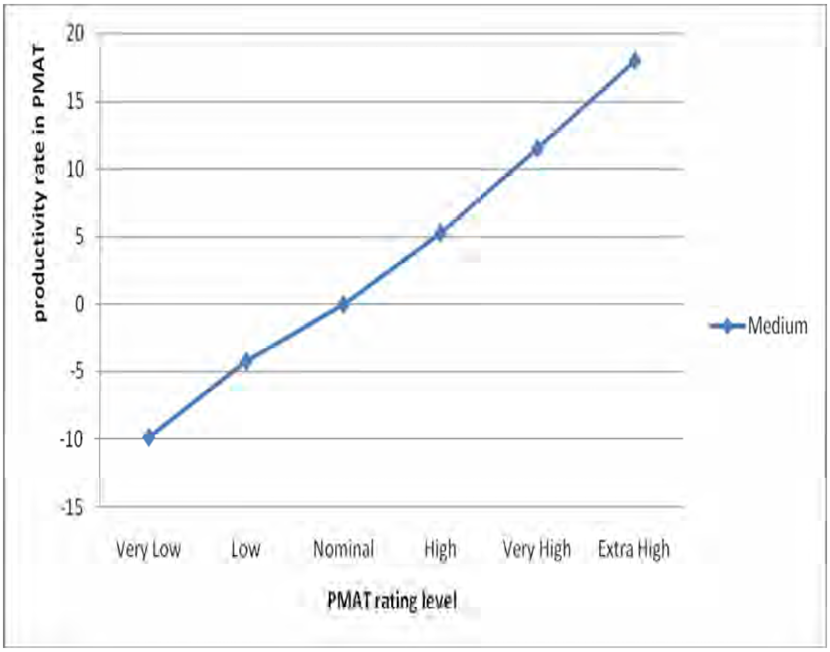

Figure 7.15 Change of percentage in productivity rate with all PMAT ratings for medium project sizes.

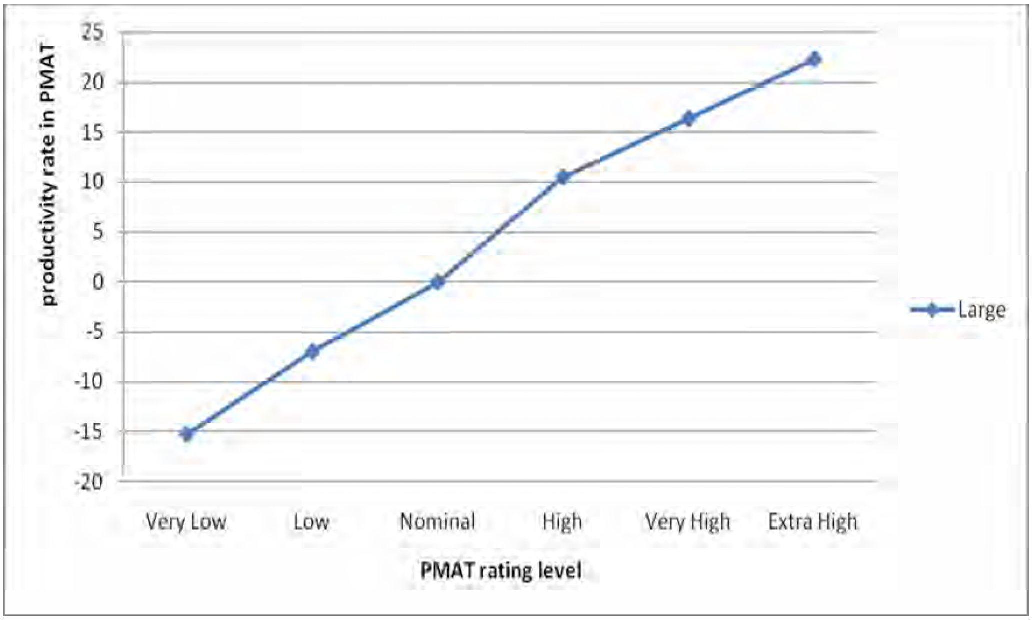

Figure 7.16 Change of percentage in productivity rate with all PMAT ratings for large project sizes.

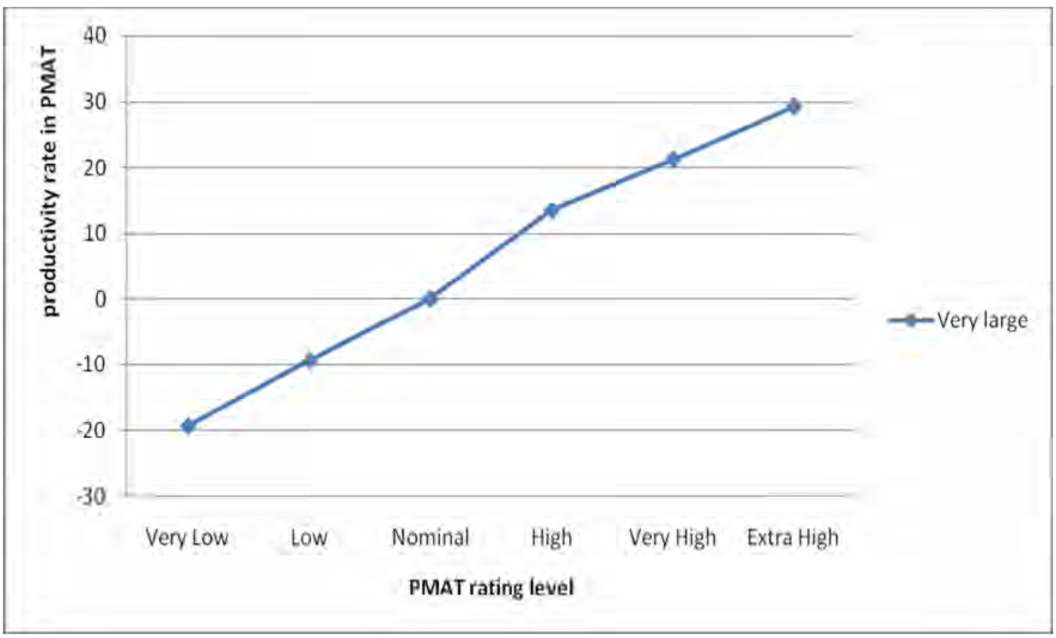

Figure 7.17 Change in percentage of productivity rate with all PMAT ratings for very large project sizes.

The percentage change in productivity rate for the medium project size shows 9.81% in very low PMAT rating level and increases to 17.97% shown in figure 7.15. Similarly change in percentage over productivity rate for large and very large project sizes have decreased to 15.29% and 19.34% whereas change of percentage in productivity rate have increased by 22.29% and 29.21% as shown in the figure 7.16 and figure 7.17.

According to the figure 7.18, the change in percentage over productivity rate in all CMMI based on PMAT rating levels for all standard project sizes are illustrated. However, the productive rate is inversely proportional to the effort which can be seen in figure 7.18 as PMAT rating level is very low with decrease in percentage change in productive rate which explains SMEs lower than the nominal level of CMMI based on process maturity having poor productivity due to undefined process capability profile for the executive in the organization. Simultaneously, the rating level of process maturity for high, very high and extra high exhibit increase in productivity rate due to CMMI based PMAT rating level. A better productivity rate is seen increase in PMAT rating level for all standard project sizes. Therefore, the productivity rate increases from very low to extra high rating level for all projects of standard sizes.

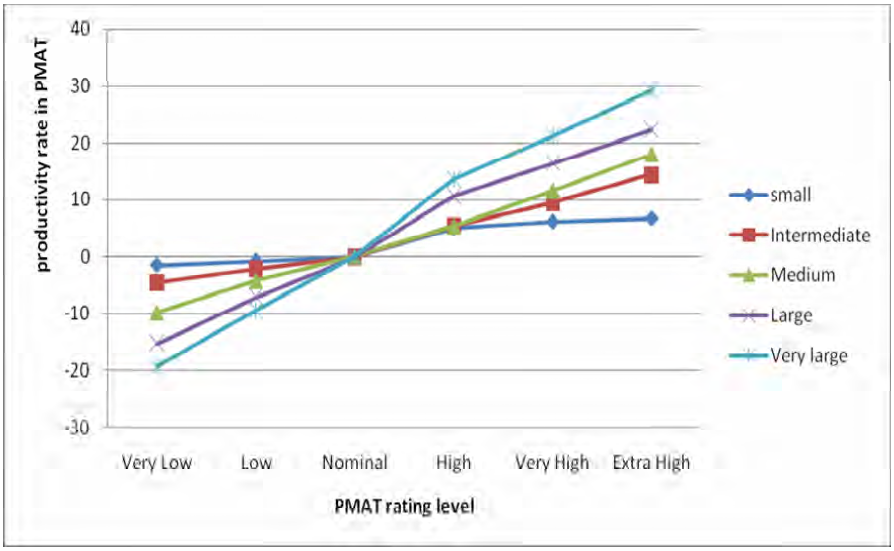

Figure 7.18 Change of percentage in productivity rate with all PMAT ratings for all standard project sizes.

7.4 Summary

This chapter has shown results through comparison of the PMAT outcomes for all kind of maturity rating levels for SMEs by evaluating the change in percentage of other factors with PMAT using parameters like percentage change in effort, percentage change in productivity and percentage change in scaling for diseconomy in the all standard project sizes.

CHAPTER – 8
CONCLUSION AND FUTURE WORK

8.1 Overview

This chapter, includes a discussion relating to a method with profile capability process performance presented in this work that has matched on the basis of software company business model which assists SMEs in the development of their software processes and indicate the effort used by the method. Hence, this process is treated as referring to renewable units which are adopted by organization for accomplishing the chosen maturity levels of CMMI. The proposed model has three stages namely, planning, analysis and implementation.

8.2 Conclusion

The basic ideas followed in CMMI are reached from several platforms and advanced from experience of various persons, but two issues are focused in this research, namely increase in marketplace pressure in the organizations and modification of the framework of CMMI based on appraisal methods for resolving organizational issues. This research has focused on organization maturity levels rather than process capability. Nevertheless, the manipulation of maturity levels through organizational capability but has the ability to specify the risk involved in the process areas and also to guide the process improvement using the needed description with the least activities set. Moreover, various cases have high maturity ratings that not specified efficiently and practices with high maturity do not make the appraisal process with fault or the organizations are dishonest, simplicity in the framework of maturity does not seem to be suitable for all practices in organizations. Without any change, one can expect more cases where high maturity ratings do not generally correlate with the performance of the improvement process.

In this research work, the results of CMMI based on PMAT of all maturity rating levels for all projects sizes by computing the changes in percentage in several factors with CMMI based on PMAT using parameters namely change of percentage in effort estimation, change of percentage in productivity and change of percentage change in scaling for diseconomy whereas the maturity rating level 2 has represented as the nominal in which the change of percentage in the PMAT nominal is zero. Effort estimation is indirectly proportional to the productivity rate whereas this kind of representation has established strong correlation that have SME's among the reported process maturity cases to process capability profile suggesting continued and sustained process improvement with process performance initiatives following to the appraisal in an organization.

BIBLIOGRAPHY

1. A. M. Al-Ashmori, B. Bashiri Rad, and I. Suhaimi, "Software process improvement frameworks as alternative of CMMI for SMEs: a literature review". Journal of SE, vol.11, no. 2, pp, 123-133, March 2017.
2. https://www.cmmiconsultantblog.com/wp-content/uploads/2018/03/ CMMI- Version-2.0-Brochure.pdf.
3. A. M. Solyman, O. A. Ibrahim, and A. A. Elhag, "Project management and software quality control method for small and medium enterprise," IEEE International Conf. on In Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE),pp.123-128,Sept. 2015.
4. M. A. T. Almomani, S. Basri, A. K. B. Mahmood, and A. O. Bajeh, "Software development practices and problems in Malaysian small and medium software enterprises: a pilot study," 5th International Conf. on IT Convergence and Security (ICITCS), IEEE. Kuala Lumpur, Malaysia, pp. 1-5, Aug. 2015.
5. J. Iqbal, R. B. Ahmad, M. H. Nasir, M. M. Niazi, S. Shamshirband, and M. A. Noor, "Software SMEs' unofficial readiness for CMMI®-based software process improvement," SQ Journal, vol.24, no.4, pp. 997-1023, Dec.2016.
6. Dr. Paloli Mohammed Shareef, "Study of Fault Injection Patterns in Software Development to Analyze Defect Leakage and Amplification", PhD Dissertation, Page 17-18, Anna University, 2014.
7. N. Nikitina, and M. Kajko-Mattsson, "Guiding the adoption of software development methods," Proc. International Conf. on Software and System Process, ACM, Nanjing, China, pp. 109-118, May 2014.
8. CROSSTALK. "Beyond the Agile Manifesto. Cross Talk," Journal of DSE, vol. 29, no. 6, pp. 40-41, Nov. 2916
9. Yeh, K., Adams, M., Stamper, P., Dasgupta, D., Richards, A., Hewson, R., Hay, J. (2016). National laboratory planning: Developing sustainable bio containment laboratories in limited resource areas. Health Security, 14, 323 – 330. doi:10.1089/hs.2015.0079.
10. M. B. Chrissis, M. Konrad and S. Shrum, "CMMI: Guidelines for Process Integration and Product Improve- ment," 3rd Edition, Addison-Wesley,

Boston, 2011.
11. Carnegie Mellon Software Engineering Institute, “What Is CMMI?” 2008. http://www.sei.cmu.edu/cmmi/general/.
12. Carnegie Mellon Software Engineering Institute, “The CMMI Version 1.2 Overview presentation,” 2008. http://www.sei.cmu.edu/cmmi/adoption/ pdf/ cmmi-overvi ew07.pdf
13. Capability Maturity Model, “Wikipedia,” 2008. http://en.wikipedia.org/ wiki/ Capability_maturity_model
14. J. Heflin, “OWL Web Ontology Language Use Cases and Requirements,” 2004. http://www.w3.org/TR/webont-req/
15. B. Gallagher, M. Phillips and, K. Richter and S. Shrum, “CMMI-ACQ: Guidelines for Improving the Acquisition of Products and Services,” 2nd Edition, Addison-Wesley, Boston, 2011.
16. CMMI for Development, “CMMI-DEV V1.3,” Technical Report, Software Engineering Institute, Pittsburgh, 2010.
17. E. Forrester, B. Buteau and S. Shrum. “CMMI for Services: Guidelines for Superior Service,” 2nd Edition, Ad- dison-Wesley, Boston, 2011.
18. S. Vince Raicheal, Dr.M.V.Srinath and Dr.P.M.Shareef, “Understanding the Conceptual Value in Adopting CMMI Process Maturity Framework”, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 12 (2015) pp. 30345-30351.
19. D. Ho, A. Kumar, and N. Shiwakoti, "Maturity model for supply chain collaboration: CMMI approach," in 2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), , 2016, pp. 845- 849.
20. A. A. Khan and J. Keung, "Systematic review of success factors and barriers for software process improvement in global software development", IET Software, vol. 10, pp. 125-135, 2016.
21. P. Domingues, P. Sampaio, and P. M. Arezes, "Integrated management systems assessment: a maturity model proposal," Journal of Cleaner Production, vol. 124, pp. 164-174, 2016.

22. D. Doss, R. Henley, D. McElreath, B. Gokaraju, R. Goza, and G. Lusk, "Process improvement: Urban vs. rural personnel perspectives of a derivative CMMi process maturity framework," in Proceedings of the Southwest Academy of Management, 2017, pp. 44-51.
23. F. S. Silva, F. S. F. Soares, A. L. Peres, I. M. de Azevedo, A. P. L. Vasconcelos, F. K. Kamei, et al., "Using CMMI together with agile software development: A systematic review," Information and Software Technology, vol. 58, pp. 20-43, 2015.
24. F. S. F. Soares and S. R. de LemosMeira, "An agile strategy for implementing CMMI project management practices in software organizations," in 2015 10th Iberian Conference on Information Systems and Technologies (CISTI), , 2015, pp. 1-4.
25. N. Fenton and J. Bieman, "Software metrics: a rigorous and practical approach: CRC Press", 2014.
26. M. Mane, M. Joshi, A. Kadam, and S. Joshi, "Software Reliability and Quality Analyser with Quality Metric Analysis Along With Software Reliability Growth Model", International Journal of Computer Science & Information Technologies, vol. 5, 2014.
27. O. Gordieiev, V. S. Kharchenko, and M. Fusani, "Evolution of Software Quality Models: Green and Reliability Issues," in ICTERI, 2015, pp. 432-445.
28. N. Paternoster, C. Giardino, M. Unterkalmsteiner, T. Gorschek, and P. Abrahamsson, "Software development in startup companies: A systematic mapping study", Information and Software Technology, vol. 56, pp. 1200-1218, 2014.
29. S.-Z. Ke, C.-Y. Huang, and K.-L. Peng, "Software reliability analysis considering the variation of testing-effort and change-point," in Proceedings of the International Workshop on Innovative Software Development Methodologies and Practices, 2014, pp. 30-39.
30. V. R. Basili and H. D. Rombach, "The TAME project: Towards improvement- oriented software environments," IEEE Transactions on software engineering, vol. 14, pp. 758-773, 1988.

31. V. Basili, M. Zelkowitz, F. McGarry, J. Page, S. Waligora, and R. Pajerski, "SEL's software process improvement program," IEEE Software, vol. 12, pp. 83-87, 1995.
32. I. Sommerville and P. Sawyer, Requirements engineering: a good practice guide: John Wiley & Sons, Inc., 1997.
33. D. Falessi, M. Shaw, and K. Mullen, "Achieving and maintaining CMMI maturity level 5 in a small organization," IEEE software, vol. 31, pp. 80-86, 2014.
34. Olejnik 2014, International small and medium-sized enterprises: Internationalization patterns, mode changes, configurations and success factors, Springer Science & Business Media.
35. Durst &Edvardsson 2012, 'Knowledge management in SMEs: a literature review', Journal of Knowledge Management, vol. 16, no. 6, pp. 879-903.
36. Burgess, Sellitto&Karanasios 2009, Effective Web Presence solutions for small businesses: strategies for successful implementation, Information Science Reference.
37. Curran & Blackburn 2001, Researching the Small Enterprise, SAGE Publications.
38. Sellitto, Banks, Bingley & Burgess 2017, Small Businesses and Effective ICT: Stories and Practical Insights, Routledge, 131737813X.
39. Yesseleva 2012, 'Small-and medium-sized enterprises: Data sources in Australia', Global Journal of Business Research (GJBR), vol. 6, no. 2, pp. 83-92.
40. Welsh & White 1981, 'A small business is not a little big business', Harvard Business Review, vol. 59, no. 4, p. 10.
41. Churchill 1983, 'The Five Stages of Small Business Growth', Harvard Business Review, vol. 61, no. 3, p. 10.
42. d' Amboise & Muldowney 1988, 'Management Theory for Small Business: Attempts and Requirements', Academy of Management Review, vol. 13, no. 2, pp. 226-40.
43. DeLone 1988, 'Determinants of Success for Computer Usage in Small Business', MIS Quarterly, vol. 12, no. 1, p. 11.

44. Poon & Huang 2004, “E-commerce and SMEs: a reflection and the way ahead”, in N Al-Qirim (ed.), Electronic commerce in Small to Medium-Sized Enterprises: Frameworks, Issues and Implications, Idea Group, Hershley, PA, USA, pp. 17-30.
45. Thong & Yap 1995, “CEO Characteristics, Organizational Characteristics and Information Technology Adoption in Small Businesses”, International Journal of Management Science, vol. 23, no. 4, p. 14.
46. Robins 1996, Organisational Behavior, Prentice-Hall International Inc.
47. Mintzberg 1979, The Structuring of Organization, Prentice-Hall, Englewood Cliffs, NJ.
48. Kuwayama 2001, E-Commerce and export promotion policies for Small and Medium-Sized Enterprises: East Asian and Latin American Experiences, International Trade and Integration Division, United Nations Publications, Santiago, Chile.
49. Wong & Aspinwall 2004, 'Characterizing knowledge management in the small business environment', Journal of Knowledge Management, vol. 8, no. 3, pp. 44-61.
50. Chin Wei, Siong Choy &Geok Chew 2011, 'Inter-organizational knowledge transfer needs among small and medium enterprises', Library Review, vol. 60, no. 1, pp. 37-52.
51. Thong 1999, 'An Integrated Model of Information Systems Adoption in Small Businesses', Journal of Management Information Systems, vol. 15, no. 4, p. 29.
52. Thong, Yap & Raman 1996, 'Top Management Support, External Expertise and Information Systems Implementation in Small Businesses', Information Systems Research, vol. 7, no. 2, p. 20.
53. Bracci & Vagnoni 2011, 'Understanding Small Family Business Succession in a Knowledge Management Perspective', IUP Journal of Knowledge Management, vol. 9, no. 1, pp. 7-36.
54. Blackburn & Kovalainen 2009, 'Researching small firms and entrepreneurship: Past, present and future', International Journal of Management Reviews, vol. 11, no. 2, pp. 127-48.

55. Molnar, Nguyen, Homolka & Macdonald 2011, 'Knowledge Management as a solution for the shortage of competent employees in SMEs at the developing country (Case study: Vietnam)', Journal of Systems Integration, vol. 2, no. 3, pp. 38-46.
56. Welsh & White 1981, 'A small business is not a little big business', Harvard Business Review, vol. 59, no. 4, p. 10.
57. Colombo, Croce & Grilli 2013, 'ICT services and small businesses' productivity gains: An analysis of the adoption of broadband Internet technology', Information Economics and Policy.
58. Carpenter & Petersen 2002, 'Is the growth of small firms constrained by internal finance?', Review of Economics and Statistics, vol. 84, no. 2, pp. 298- 309.
59. Vinten 1999, 'Corporate Communications in Small - and Medium-Sized Enterprises', Industrial and Commercial Training, vol. 31, no. 3, pp. 112-9.
60. Churchill 1983, 'The Five Stages of Small Business Growth', Harvard Business Review, vol. 61, no. 3, p. 10.
61. Mintzberg 1979, The Structuring of Organization, Prentice-Hall, Englewood Cliffs, NJ.
62. Mohd Sam, Hoshino &Hayati Tahir 2012, 'The Adoption of Computerized Accounting System in Small Medium Enterprises in Melaka, Malaysia', International Journal of Business & Management, vol. 7, no. 18, pp. 12-25.
63. Wong & Aspinwall 2004, 'Characterizing knowledge management in the small business environment', Journal of Knowledge Management, vol. 8, no. 3, pp. 44-61.
64. Cataldo, Sepúlveda& McQueen 2012, 'Exploring the IT Usage in SMEs from New Zealand, Colombia and Chile Using Action- Research Methodology', Journal of Innovation Management in Small & Medium Enterprises, pp. 1-8.
65. Ghobadian&Gallear 1997, 'TQM and organization size', International Journal of Operations & Production Management, vol. 17, no. 2, pp. 121-63.
66. Ghobadian & O' Regan 2006, 'The Impact of Ownership on Small Firm

Behaviour and Performance', International Small Business Journal, vol. 24, no. 6, pp. 555-86.

67. Nagy Ramadan Darwish, November, 2011. Improving the Quality of Applying extreme Programming (XP) Approach, International Journal of Computer Science and Information Security (IJCSIS) – ISSN, Vol. 9 No. 11, pp. 1947- 5500.
68. Usha, K., Poonguzhali, N. and Kavitha, E., 2009, A Quantitative Approach for Evaluating the Effectiveness of Refactoring in Software Development Process, International Conference on Methods and Models in Computer Science, Delhi, India.
69. Putnam, L. H., 1978. A General Empirical Solution to the Macro Software Sizing and Estimating Problem, IEEE Transactions on Software Engineering, Vol. 4, No. 4, P.345 – 361.
70. Boehm B. W., 1981.Software Engineering Economics, Englewood Cliffs, NJ, Prentice-Hall.
71. Boehm B., Abts, C., and Chulani, S., 2000. Software Development Cost Estimation Approaches – A Survey, University of Southern California Center for Software Engineering, Technical Reports, USC-CSE- P.2000-505.
72. Roger S. Pressman, 2001. Software Engineering: A Partitioner's Approach, MaGraw Hill, edition 5.
73. Popli, R., & Chauhan,N., 2014. Cost and effort estimation in agile software development. Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on, P.57–61.
74. Trendowicz, A., & Jeffery, R., 2014. Software Project Effort Estimation: Foundations and Best Practice Guidelines for Success. Springer International Publishing.
75. Moløkken, K., & Jørgensen, M., 2003. A review of surveys on software effort estimation. International Symposium on Empirical Software Engineering, Retrieved from ACM Digital Library database, P. 223-231.
76. Shepperd, M., Schofield, C., &Kitchenham, B., 1996. Effort Estimation Using Analogy. Proceeding softhe18th International Conferenceon Software Engineering, P.170–178.

77. Wen,J., Li,S., Lin,Z., Hu,Y. and Huang,C., 2012. System architecture review of machine learning based software development effort estimation models. Information and Software Technology, 54(1), P.41–59.
78. Jørgensen, M., 2004. Areview of studies on expert estimation of software development effort. Journal of Systems and Software, 70(1-2), P.37–60.
79. Jørgensen, M., 2007. Forecasting of software development work effort: Evidence on expert judgement and formal models. International Journal of Forecasting, 23(3):P.449–462.
80. Putnam, L. H., 1978. A General Empirical Solution to the Macro Software Sizing and Estimating Problem, IEEE Transactions on Software Engineering, Vol. 4, No. 4, P.345 – 361.
81. Black, R.K.D., Curnow, R.P., Katz, R., and Gray, M.D., March,1997. BCS Software Production Data, Final Technical Report, RADC-TR-77-116, Boeing Computer Services, Inc.
82. Chris F.K., May, 1987. An Empirical Validation of Software Cost Estimation Models, Management of Computing- Communications of ACM, Vol: 30, No. 5, P.416-429.
83. Singh, Y., Aggarwal, K.K.,2016. Software Engineering Third edition, New Age International Publisher Limited New Delhi.
84. The CMMI Overview by SEI. Available at http://www.sei.cmu.edu/cmmi/ general/ general.html
85. The CMMI models can be found at http://www.sei.cmu. edu/cmmi/ models/ models. html.
86. DoD 1996 Department of Defense DoD Guide to Integrated Product and process Development (Version1.0) http://www.abm.rda.hq.navy.mil/ naveyaos/content/download/1000/4448/file/ippdhdbk.pdf
87. Chrissis 2003 Chrissis, Mary Beth; Konrad, Mike; & Shrum, Sandy. CMMI: Guidelines for Process Integration and Product Improvement.
88. Dymond 2004 Dymond, Kenneth M. A Guide to the CMMI: Interpreting the Capability Maturity Model Integration. MD: Process Transition International Inc., 2004.
89. Crosby 1979 Crosby, Philip B. Quality Is Free The Art of Making Quality

Certain. New York: McGraw-Hill, 1979.

90. Process Improvement and CMMI for systems and software. Ron S.Kenett, Emanuel R.Baker.
91. CMMI for Development http://www.sei.cmu.edu
92. Carnegie Mellon University http://www.sei.cmu.edu/cmmi/
93. Colin Potts, "Software-Engineering Research Revised", IEEE Software, Volume 10, Number 5, pages 19-28, September 1993.
94. Mark C. Paulk, Charles V. Weber, Bill Curtis and Mary Beth Chrissis, The Capability Maturity Model - Guidelines for Improving the Software Process, CMU-SEI, Addison-Wesley, 441 pages, 1994.
95. Mary Beth Chrissis, Mike Konrad and Sandy Shrum, CMMI: Guidelines for Process Integration and Product Improvement, Addison-Wesley Pub Co, 2003.
96. ReidarConradi and Alfonso Fuggetta, Improving Software Process Improvement, IEEE Software, 19(4), p. 92-99, July/August 2002.
97. Stan Rifkin, Is process improvement irrelevant to produce new era software? in Software Quality - ECSQ 2002, Lecture Notes in Computer Science 2349, ed. by Jyrki Kontio and ReidarConradi, pp. 13-16, Springer-Verlag, 2002.
98. Linda Ibrahim, Using an Integrated Capability Maturity Model® – The FAA Experience, in Proceedings of the Tenth Annual International Symposium of the International Council on Systems Engineering (INCOSE), Minneapolis, Minnesota, pp. 643-648, July 2000.
99. Kival C. Weber, Eratóstenes E. R. Araújo, Ana Regina C. da Rocha, Cristina A. F. Machado, Danilo Scalet and Clênio F. Salviano, Brazilian Software Process Reference Model and Assessment Method, in Proceedings of The 20th International Symposium on Computer and Information Sciences - ISCIS'05, October 26-28th, Instambul, Turkey, 2005.
100. The International Organization for Standardization and the International Electro technical Commission, ISO/IEC 15504 - Information Technology - Process Assessment – Part 2, 2003.

101.F. Stallinger, A. Dorling, T. Rout, B. Henderson-Sellers and B. Lefever Software Process Improvement for Component-Based Software Engineering: An Introduction to the OOSPICE Project, Proceedings of the 28th EUROMICRO Conference, Germany, IEEE Computer Society, 2002.
102.Automotive SIG, Automotive SPICE Process Assessment Model, Final Release, v2.2, 144 pages, © The SPICE User Group, 21 August 2005 (available from http://www.automotivespice.com, last accessed 09/11/2005).
103.Jean-Marie Favre, Foundations of Model (Driven) (Reverse) Engineering: Models, Episode I: Stories of The Fidus Papyrus and of The Solarus, 31 pages, Pilot of the series "From Ancient Egypt to Model Driven Engineering", available at http://www-adele.imag.fr/mda, 2005. (last accessed in 20/03/2006).
104.David N. Card, Research Directions in Software Process Improvement, Proceedings of 28th International Computer Software and Applications Conference (COMPSAC 2004), Hong Kong, China, IEEE Computer Society, p. 238, September 27-30, 2004.
105.Jean Bézivin, MDA™ : From Hype to Hope and Reality, Slides from a guess talk presentation at UML´2003, San Francisco, 96 slides, 2003 (available at: www.sciences.univ-nantes.fr/info/perso/permanents/bezivin/ UML.2003/UML.SF.JB.GT.ppt, last accessed in 27/07/2004).
106.Search CRM, The searchCRM.com website, 2006. (available athttp:/ searchcrm.techtarget.com, last accessed in 30/03/2006).
107.T. K. Abdel-Hamid and S. E. Madnick, "Impact of schedule estimation on software project behavior", IEEE Software, 1986, pp. 70-75.
108.Y. Miyazaki and K. Mori, "COCOMO Evaluation and Tailoring," Proceedings, ICSE 8, IEEE-ACM-BCS, 1985 pp. 292-299.
109.R. Dillibabu, K. Krishnaiah, "Cost estimation of a software product using COCOMO II.2000 model – a case study," International Journal of Project Management, vol. 23, 2005, pp. 297-307.
110.B. Boehm, Software Engineering Economics, Prentice Hall, 1981.
111.B. Boehm, E. Horowitz, R. Madachy, D. Reifer, B. KClark, B. Steece, A.

Winsor Brown, S. Chulani, and C. Abts, “Software Cost Estimation with COCOMO II”, Prentice Hall, 2000.
112.B. Boehm, , B. Clark, E. Horowitz, C. Westland, R. Madachy, R. Selby, "Cost Models for Future Software Life Cycle Processes: COCOMO 2.0," Annals of Software Engineering Special Volume on Software Process and Product Measurement, Amsterdam, vol. 1, 1995, pp. 45-60.
113.P. Musilek, W. Pedrycz, N. Sun, G. Succi, “On the sensitivity of COCOMO II software cost estimation model,” Proceedings of the Eighth IEEE Symposium on Software Metrics, METRICS. IEEE Computer Society, Washington, DC, 13, 2002.
114.B. Clark, "Quantifying the Effects of Process Improvement on Effort," IEEE Software, vol. 17, no. 6, 2000, pp.65-70.
115.M. Paulk, C. Weber, B. Curtis, and M. chrissis, The Capability Maturity Model: Guidelines for Improving the Software Process, ddison-Wesley, Reading, Mass., 1995.
116.Capability Maturity Model® Integration (CMMI®) Version 1.2 Overview, Carnegie Mellon University, Software Engineering Institute. Accessed on 02.May.2009http://www.sei.cmu.edu/cmmi/adoption/pdf/cmmi-overview07. pdf
117.S. Chulani, B. Boehm, and B. Steece, "Bayesian Analysis of Empirical Software Engineering Cost Models," IEEE Trans. on Software Engineering, vol. 25, no. 4, 1999, pp. 573–583.
118.S. Chulani, B. Boehm, and B. Clark, “Calibrating the COCOMO II Post-Architecture Model” Proceeding ICSE98 IEEE, vol. 1, 1998, pp. 477-480.
119.B. Clark, "Calibration of COCOMO II.2003," 17th International Forum on COCOMO and Software Cost Modeling, http://sunset.usc.edu/ events/2002/ cocomo17/Calibration%20fo%20COCOMO%20I I.2003% 20Presentation%20-%20Clark.pdf.
120.Y. Yang, B. Clark, "COCOMO II.2003 Calibration Status," CSE Annual Research Review, March 2003, http://sunset.usc.edu /events/ 2003/ March_ 2003/ COCOMO_II_2003_Recalibration. Pdf.
121.Y. Yang, B. Clark, "Reducing Local Calibration Bias in COCOMO II 2004

Calibration," 19th International Forum on COCOMO and Software Cost Modeling, October 26-29, 2004, http://sunset.usc.edu/cse/pub/event/2004/ COCOMO/ files/WedAM/Wed_AM_05.ppt

122.J. Baik, "Disaggregating and Calibrating the Case Tool Variable in COCOMO II," IEEE Trans. Software Eng., vol. 28, no. 6, 2002, pp. 1009-1022.

123.Z. Chen, T. Menzies, and D. Port, "Feature Subset Selection Can Improve Software Cost Estimation Accuracy," Proc.Workshop Predictor Models in Software Eng., ACM Press, 2005.

124.J. Hale, A. Parrish, B. Dixon, and R. Smith, "Enhancing the COCOMO Estimation Models," IEEE Software, vol. 17, 2000, pp. 45-50.

125.J. Randall, "Extreme Software Cost Estimating," The Journal of Defense Software Engineering, vol. 17, No. 1, 2004, pp. 27-30.

126.M. Yahya, F. Masoud, and A. Hudaib, "The effect of Software Development Environment on Software Cost Estimation," The Proceedings of the 10th World Multi- Conference on Systematic, Cybernetics and Informatics, 2006, PP. 239-243, USA,

127.J. Herbsleb et al., Benefits of CMM-Based Software Process Improvement: Initial Results, Tech. Report CMU/SEI-94- TR-13, Software Eng. Inst., Carnegie Mellon Univ., Pittsburgh, 1994.

128.J. Brodman and D. Johnson, "Return on Investment (ROI) from Software Process Improvement as Measured by US Industry," Software Process Improvement and Practice, John Wiley &Sons Ltd., Sussex, England and Gauthier-Villars, 1995, pp. 35- 47.

129.K. Butler, "The Economic Benefits of Software Process Improvement," Crosstalk, Hill AFB, Ogden, 1995, pp. 14-17.

130.D.E. Harter, M.S. Krishnan, and S.A. Slaughter, "Effects of Process Maturity on Quality, Cycle Time and Effort in Software Product Development," Management Science, vol. 46, 2000, pp. 451-466.

131.M. Chrissis, M. Konrad, and S. Shrum, CMMI: guidelines for process integration and product improvement, Addison-Wesley, 2003.

132.D. Gibso, D. Goldenson, and K. Kost, "Performance Results of CMMI-

Based Process Improvement," CMU/SEI-94- TR-13, Software Engineering Institute TECHNICAL REPORT CMU/SEI-2006-TR-004 ESC-TR-2006-004.

133.D. Goldenson, D. Gibson, "Demonstrating the Impact and Benefits of CMMI: An Update and Preliminary Results," (CMU/SEI-2003-SR-009). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2003.

134.Manish A. and Kaushal C., "Software Effort, Quality, and Cycle Time: A Study of CMM Level 5 Projects," IEEE Transactions on Software Engineering, vol. 33, no. 3, pp. 145-156, Mar. 2007

135.(Diaz and Sligo, 1997) M. Diaz and J. Sligo, "How Software Process Improvement Helped Motorola," IEEE Software, vol. 14, no. 5, pp. 75-81, Sept.- Oct., 1997.

136.H. Wohlwend and S. Rosenbaum, "Schlumberger◎s Software Improvement Program," IEEE Trans. Software Eng., Vol. 20, No. 11, Nov. 1994, pp. 833– 839.

137.Y. Tsen, T. Sheng, C. Ching, S. Huang, "Assessing the Adoption Performance of CMMI-Based Software Process Improvement in 18 Taiwanese Firms "Journal of Software Engineering Studies, Vol. 1, No. 2, 96- 104，December 2006.