UNIVERSITY OF CALGARY

Software Engineering Process Benchmarking and Industrial Applications

by

Vincent N.S. Chiew

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE
SPECIALIZATION IN SOFTWARE ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE

CALGARY, ALBERTA

SEPTEMBER, 2002

Your file Votre référence
ISBN: 0-612-87211-4
Our file Notre référence
ISBN: 0-612-87211-4

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Software Engineering Process Benchmarking and Industrial Applications" submitted by Vincent N.S. Chiew in partial fulfillment of the requirements for the degree of Master of Science.

S

D eering

Far

ineering

Oct. 30, 2002

Date

ABSTRACT

This thesis studies a Software Engineering Process (SEP) benchmarking methodology and related benchmark-gap analysis techniques to effectively assist the information technology industry in their benchmarking effort in software engineering. The thesis begins with a look at the history of benchmarking and the current challenges in benchmarking. A comprehensive SEP assessment model is adopted as the foundation of SEP benchmarking to overcome these challenges. This thesis provides comprehensive background information and literature review on SEP assessment models, follows by details on the proposed SEP benchmarking methodology and benchmark-gap analysis techniques. A case study is used to validate the methodology and its effectiveness. The benchmarking techniques developed in this thesis may provide new ways for organizations on business goals achievement, costs management, continuous process improvement and knowledge assets handling.

ACKNOWLEDGEMENTS

This thesis is not possible without the mentoring of my supervisor, Professor Dr. Yingxu Wang. Dr. Wang's experience and expertise provide invaluable guidance and insight throughout this research and the thesis development.

Much appreciation goes to Lorraine Storey for helping me navigate my graduate academic journey. I would like to specifically thank Craig Parkin and Chris Kliewer for taking valuable time out of their busy schedule to help proof read and edit my thesis on short notice.

The author appreciates the software engineering process assessment data provided by all companies. Due to the nature of benchmarking, their names cannot be disclosed.

The author would like to thank the examining committee, Dr Yingxu Wang, Dr. Frank Maurer and Dr. Behrouz Homayoun Far for taking valuable research time off to examine my thesis and to provide advice and constructive comments. Extended thanks go to Dr. Frank Maurer for providing detailed feedbacks helping to greatly improve the quality of this thesis.

My gratitude extends to all my professors, colleagues, and friends, past, present and future, who have directly or indirectly influenced my graduate degree studies and contributed to my software engineering knowledge.

Much credit also goes to my Dad, Keng-Wei Chiew, and my Mom, Bebe Cheah, who always encourage and support me to be the best in all my life endeavors. I, furthermore, would like to thank my brother Robert for sharing his financial and economic acumen. Finally, words cannot express my deepest gratitude towards my wife, Susan Eng, and my two young daughters, Vinshie and Sherwinn.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Avg	Average
BM	Benchmark
BPA	Base Process Activity
CeBASE	Center for Empirically Based Software Engineering
CIO	Chief Information Officer
CL	Capability Level
CMM	Capability Maturity Model
CMMI	Capability Maturity Model Integration
CPI	Continuous Process Improvement
CRM	Customer Relationship Management
DB	Database
FP	Function Point
GQM	Goal Question Metric
GUI	Graphical User Interface
ID	Identification
IEC	International Electrotechnical Commission
ISO	ISO is not an acronym but a common name for International Organization for Standardization
KP	Key Practice
KPA	Key Practice Area
LOC	Line of Code
Max	Maximum
MI	Management Issues
Min	Minimum
MTA	Main Topic Area
MVC	Model View Controller
N/A	Not Applicable
NSDIR	National Software Data and Information Repository
OOM	Object Oriented Methodology
OU	Organization Unit
PCL	Process Capability Level

QA	Quality Assurance
ROI	Return On Investment
SAT	Satisfied
SEI	Software Engineering Institute
SEP	Software Engineering Process
SEPRM	Software Engineering Process Reference Model
SLOC	Source Line of Code
SPA	Software Process Assessment
SPI	Software Process Improvement
SPICE	Software Process Improvement and Capability dEtermination
SQL	Structured Query Language
SW-CMM	Software Capability Maturity Model
TQM	Total Quality Management
TR	Technical Report
UNSAT	Unsatisfied

One who has a thorough knowledge of oneself and one's competitors is bound to win in all competitions. One who knows oneself but not the competitors has only an even chance of winning. One who knows not oneself and the competitors is bound to perish in all battles.

Know your competitors, know yourself, and your victory will be guaranteed. Know the terrain, know the weather, your victory will be complete.

SUN TZU, THE ART OF WAR 500 B.C.
(Wee C, Lee K, Bambang W. H., 1996, modified by the author in translation)

CHAPTER 1 INTRODUCTION

Software development organizations have acknowledged that continuous self-improvement and compliance with industrial standards are important to achieve business success and to gain customer acceptance. The challenge faced by most organizations is to be the customer's preferred supplier. In order to be a preferred supplier, one has to be viewed by the customer as being the best in the industry with superior performance. Being ahead in the market place has always been synonymous with gaining approval of potential customers and increasing the quality profile of an existing customer base. Not much attention has been paid to the real opponent of a successful business – the competitors. In the market place, customers usually place emphasis on who gets a job based on vendor analysis. From a vendor's point of view, other vendors are competitors. The best vendor will probably get the contract.

This thesis is presenting Software Engineering Process (SEP) Benchmarking, based on a unified and comprehensive SEP Assessment Model, as an effective method and as an efficient implementation, to achieve superior performance ahead of competitors. The benchmark method is based on the assessment of software engineering industry standards, such as the Capability Maturity Model (CMM), ISO 9001, BOOTSTRAP, and ISO/IEC 15504. In addition, a comprehensive and integrated Software Engineering Process Reference Model (SEPRM) [Wang and King, 2000] is utilized as an effective tool for assessments [Dyck, 2001].

Effective assessment leads to effective benchmarking. The benchmarking technology developed reduces the total number benchmarking comparison processes. The reduction in the number of processes enables an organization to spend saved time and money to collect more projects appraisals for benchmarking. More benchmark results that may be collected helps an organization to understand its projects.

This chapter introduces the aims and objectives of this thesis and its underlying research motivation. The motivation is supplemented with a brief literature review of historical challenges by past benchmarking methodologies, the current state of benchmarking, and identified challenges. The general approach taken by this thesis, to overcome these challenges, is provided as an overview.

1.1 Aims and Objectives

The aims of this thesis are to develop an integrated SEP benchmarking methodology, and to apply it to the software industry efficiently and effectively. In order to reach these aims, a few objectives are considered and used to evaluate the success of this work. The main objectives are to show at least one way to efficiently implement SEP benchmarking and effectively use the SEP benchmark results.

The following list of objectives have been identified as guidelines for this work:

- To develop an SEP assessment tool and an SEP benchmarking tool
    - Validate the tools' mapping process and algorithm against the SEPRM model [Wang and King, 2000]
- To develop an efficient way to perform SEP benchmarking
    - Propose utilization of a comprehensive SEP assessment model
    - Develop a gap analysis technique
- To develop a way to effectively use SEP benchmark
    - Utilization of a comprehensive SEP assessment model approach

1.2 Motivation

Researchers are constantly benchmarking their current findings with their previous research results to determine if any progress has been made. The determination of a research breakthrough and leading the field in the discipline is usually determined by the comparison of the current research results with results produced by colleagues. The superior result will be set as the next target threshold for others to aim and shoot for as benchmark. The process of comparing the progress against one's own results is known as "internal" benchmarking, while the progress comparison with others' results is known as "external" benchmarking.

Researchers have to possess good benchmarking skills in order to manage and control multi-variables in a highly dynamic situation. Hence, it comes as no surprise that researchers have contributed tremendously in the development in various standards and models as the foundation to organize SEP. In addition, these standards and models also enable them to measure process capabilities and maturity levels. Some of the software engineering standards and models are CMM, ISO 9001, ISO/IEC 15504, BOOTSTRAP, and SEPRM. These standards and models will be used in this thesis.

It is difficult to determine a good approach to bridge the gap between academic research on standards and models and their practical applications in the industry [Eischen, 2002]. Assuming an approach is selected, the initial reaction by most organizations is which standard to adopt and implement, because there are so many standards available. After a standard has been chosen, the next usual stage is to implement it. Then, multiple standards are often required. This move is usually made for economic reasons such as market demands for a particular standard as a requirement or for marketing to solicit business. There are many organizations adhering to multiple standards. This further compounds the existing problems of how to compare multiple organizations with multiple standards and models.

The challenges in the area of SEP benchmarking are to develop a comprehensive benchmarking method, and to find a way to apply it in the industry. The challenges are the underlying motivations of this thesis.

1.3 Perspectives on SEP Benchmarking

The fundamental focus of this thesis is benchmarking the SEP. This section provides two general perspectives of SEP benchmarking, from both academic and industrial perspectives.

Benchmark is defined as a reference point (threshold) or standard, against which others can be measured and thereby judged. The comparison process between the current state and a set of thresholds is generally known as benchmarking. In a broad sense, SEP benchmarking is defined generally as a dynamic process of setting the benchmarks, striving to meet and exceed the benchmarks, and resetting the benchmarks. The benchmarking process is then continuously analyzed and evaluated. Academic focus is on benchmarking methodology and techniques.

The industry view of benchmarking relating to the aim of this thesis is that SEP benchmarking seems to be passé. Benchmarking is not viewed as an integral part of an organization's business practice. Benchmarking is mostly utilized as a one-time use only for a specific purpose. It is common for organizations to use or abuse benchmark results for propaganda purposes. Such propaganda purposes are usually used to justify a desirable need. For example, benchmark results are commonly used as an objective support by organizations to illustrate their points to employees, customers and shareholders. Industry focus is on benchmark application of the information to achieve business goals.

1.4 Literature Review

This section briefly reviews the history of quality in an attempt to understand the origin and early roles of benchmarking. The history will start with the beginning of generic benchmarking. It will then quickly progress to a more computer-field related benchmarking history. Finally, SEP benchmarking and the current state of Software Engineering benchmarking are also presented. Yingxu Wang and Graham King [Wang and King, 2000] and Juhani Kulmala [Kulmala, 2002] have identified three historical periods of quality approaches. The three periods are described as follows:

a. *Simple quality control* – This is a simple process of checking one item at a time or using a sorting technique. It has been criticized for lack of comparison and preventive elements.
b. *Systematic process control* – This system took off around World War One. Quality gurus like W. Edwards Deming and Joseph Juran introduced systematic data collection for preventive measure and continuous process improvement.
c. *Quality assurance* – This is a modern quality philosophy, e.g. Total Quality Management.

As the progression of quality practices, there is a similar maturity and progression of benchmarking practices.

1.4.1 History of Benchmarking

Benchmarking has its root in quality. The value and importance of quality can be traced back to ancient times. It is widely accepted that the concept of benchmarking originated with Sun Tzu. Robert Camp [Camp, 1989] believes that one of the earliest benchmarks originates from (THE ART OF WAR, Sun Tzu, 500 B.C.). Sun Tzu was quoted as saying, "If you know your enemy and know yourself, you need not fear the result of a hundred battles". This is loosely translated to, the need to know your own ability and your competitors' ability; the more informed you are to compete in the market place.

The author found that the concept of benchmarking was widely practiced all the way back to the Zhou Dynasty, in Chinese history, which is slightly earlier than the Sun Tzu period.

1.4.1.1 Benchmarking: An Origin of Ancient Quality Systems

China is chosen as an example from a handful of ancient civilizations to illustrate quality systems from ancient times. During the Zhou Dynasty (eleventh century to eighth century B.C.), various departments within the government were in charge of quality. Here is a list of the departments and their responsibilities obtained from "A History of Managing Quality: The Evolution, Trends, and Future Directions of Managing for Quality" [Juran, 1995]:

- The department in charge of production, collection, storage, and distribution of raw and semi-finished materials
- The department of production and manufacturing
- The department for storing and distributing completed product
- The department for formulating and executing standards
- The department of supervision and examination

This was essentially "a simple quality process". It was a checklist-based quality system, and this simple quality system was not just the motion of executing the quality process, but the quality process was strictly enforced and maintained for hundreds of years.

Another example is the building of a vehicle by a man named Xi Zhong around the seventeenth century. Xi Zhong was in charge of maintaining a standard; hence, also in charge of setting up an assessment method that other vehicles can be benchmarked and measured for compliancy. As Juran [Juran, 1995] pointed out, that maturity of quality systems over the centuries has caused China's quality system to possess unique and distinct quality features, from centuries of CPI. This further validates the value of CPI, given sufficient time. It is the hope of this thesis that benchmarking the SEP will be able to make the iteration of each CPI more efficient and effective using benchmark-based Software Process Improvement (SPI) approach [Wang and King, 2000a]. Details of the benchmark-based approach will be presented in Chapter 4 on SEP benchmarking.

Prior to outsourcing manufacturing work to the private sector, the government during the Zhou Dynasty made it a requirement to inscribe the craftsman's name and the official branch on the product produced. This was to ensure the quality was traceable. This was another breakthrough in quality tracing. It can also serve as a recognized benchmark for others to match or exceed. Then around the Jin Dynasty (281-420 A.D.), some government work was outsourced to civilians to produce. The government still maintained standard and only the product that met the standard would have a stamp of approval. For example, the porcelain pillows of the Cizhou Kiln in the Song Dynasty carried inscriptions such as "made by the Zhong family," "made by the Zhao family" and "made by the Wang family." [Juran, 1995]. In the software industry we have standards set up by international committees such as ISO 9001, ISO/IEC 15504, and IEEE, just to name a few. It must be noted that standards are governed within the law of each locality. These days, achieving such standards is great for marketing, and as a publicizing mechanism for quality recognition and for achieving potential future business prospects.

Finally, there are many more examples of quality from ancient Israel, ancient Rome, early India, early Scandinavia, and many other places around the world [Juran, 1995]. It comes as no surprise that those ancient civilizations that embraced quality systems have survived up to modern days.

1.4.2 Modern Benchmarking Concept

A study done by Andersen and Pettersen [Kulmala, 2002] over four decades of benchmarking has identified the following four strategic values:

a. 1960s – Benchmarking was based on simple comparison. The breakthrough of new management thinking in comparison of previous financial statement with the current financial statement.
b. 1970s – Benchmarking was based on competitors' analysis.
c. 1980s – Benchmarking was based on performance analysis such as time, cost and quality. Xerox is an example of this type of benchmarking.

d. 1990s to 2000s – Benchmarking is based on active process benchmarking. This is a comparison between current active process and best practice.

In the next few subsections, this thesis will take a close look at four decades of benchmarking related to the software industry.

1.4.2.1 Benchmarking in the 1960s

In the mid 1900s, the Japanese had a saying "Dantotsu; striving for the best of the best." [Camp, 1989]. The saying essentially alludes to the philosophy of benchmarking. This benchmarking philosophy is still widely practiced in Japan.

Around this period, the Japanese has a principal called *shukko*. Shukko refers to the loaning of employees to other organizations. Zairi [Kulmala, 2002] identified the following benefits of shukko:

a. Knowledge transfer between employees and organizations
b. Acquisition of specific lacking of knowledge
c. Increase the development of managers by providing them with increasingly demanding tasks.

Even though during this period (1968-1969) SEP was not a recognized discipline, a study performed by IBM [Lehman and Ramil, 1999] relating to their operating systems such as OS/360 that investigated the laws of software evolution, did recognize that software engineering issues contained factors lying outside the realm of that discipline. This included such areas as management, organizational, sociological, and user issues and activities. Hence, part of the comprehensive goal of this thesis is to capture those software engineering related processes.

1.4.2.2 Benchmarking Evolution in the 1970s

In the modern era of benchmarking, most researchers and benchmarking practitioners agree that benchmarking originated in North America in 1979 [Kulmala, 2002; Camp, 1989]. In 1979, Xerox used benchmarking as the basis for its operation to perform CPI. It is known as process benchmarking. Process benchmarking according to Xerox is the process of learning to become better.

During this period, benchmarking in the computer field focused mainly on computer system quality. By computer system, it is meant the computer hardware. At that time, firmware and software have not matured to the same degree or achieved the same level of recognition as those of computer hardware.

Most benchmarking performed around this era was hardware related [Benwell, 1975]. The performance of computer systems was due largely to the capability of the hardware that was used in its construction. Hence, a computer system builder's capability was assessed based on hardware performance. In this era, computer software or firmware was slowly maturing.

It was just a matter of time before the computer systems builders decided to compare their products with each other to determine who produced the superior product. Product superiority was commonly being assessed by simulation, instead of benchmarking. Benwell [Benwell, 1975] mentioned that Goff published an article suggesting that benchmarking as the most accurate technique available, as opposed to simulation. For a while, comparison was performed by either simulation or benchmarking. Arguments have been going back and forth regarding which is the better method of "comparing" the different hardware and software products produced by competitors. Like many other products, there was more proprietary information regarding a competitor's products that could not be divulged, thereby making it difficult to perform comparison or benchmark. In addition, it was difficult to benchmark performance of computer systems running on various platforms such as Unix, Windows, VAX, Multics, and many other platforms. The realization was benchmark should not focus on the details, particularly technologies. It is like comparing two different kinds of fruit, during each benchmarking attempt.

Furthermore, there is a realization around this time that in order to perform CPI means a breakthrough is required in computer technologies, be it hardware or software. This led to the needs to benchmark how competitors did their work or focused on their production processes. This was hard to do around this time because there were few consensuses on internationally recognized process standards that organizations agreed upon, especially SEP standards. When a technology changed, the benchmark was no longer valid to perform continuous improvement. The new approach was the transition from benchmarking technology to benchmarking process. This approach is embraced these days by the many organizations that use process standards appraisal in their benchmarking effort.

1.4.2.3 Benchmarking for Process Improvement during the 1980s

The Software Engineering Benchmark concept started to take off during the early 1980s [Juran, 1995] even though standardized processes were not used. Processes that were benchmarked were not consistent and the benchmark could not be reused. Benchmark was used as a measurement tool based on facts instead of empirical judgment. This meant quality goals could be set that were realistic, and hopefully, achievable. Obviously, to apply the concept of benchmarking required discovery of what was the better performance being achieved, whether in-house, by a competitor or by someone in a totally different industry.

1.4.2.4 Software Process Benchmarking in the Early 1990s

Around the years 1994 to 1998, a National Software Data and Information Repository (NSDIR) were set up by the government of United States. The purpose of this undertaking was to build up a comprehensive software-related information repository for prediction capability. This program failed because it was thought to be well ahead of its time. Later, Basili and Boehm took NSDIR concepts forward and established a comprehensive academic and industrial measurement database – the Center for Empirically Based Software Engineering (CeBASE). The concepts were adopted because of its data definitions and multi-source data integration, according to Boehm. An interesting point by Mosemann was the Pentagon was not searching for the best practice but was looking for predictability. Hissam believed that CeBASE was to succeed NSDIR because at this time, there was an increase in awareness of the importance of measurement. And Hissam was sure that Boehm could make it a success. Opponents to CeBASE such as Florac and Humphrey believed the world was not ready for a comprehensive repository. In addition, Humphrey believed the interpretation of the results from such a repository could be difficult. In any case, most researchers agreed that this venture provided a leg up for this field of research. Moseman said: "With fixed-price contracting, you need to be predictable." [Goth, 2001]. It is evident that many practitioners in Software Engineering still have doubts about having an ultimate repository for benchmarking, as it might not be practical.

It was around this period that Yingxu Wang proposed a comprehensive and integrated benchmark-based Software Process Improvement (SPI). This thesis expands on that research and applies it to the industry. The benchmark involves a comprehensive and integrated SEPRM [Wang and King, 2000] serving as assessment foundation feeding to the SEP benchmarking.

1.4.3 Benchmarking: A Growing Practice and Current State

In this section, the benchmarking practice and current state are described. The introduction will take a look at benchmarking in the United States and Asia. More details on the state of art of benchmarking will be presented in the next chapter.

The benchmarking concept has been widely accepted in the United States for sometime now. Benchmarking practices are still lacking in the industry due to lack of efficiency in the benchmarking process and effectiveness in the result produced by a generic benchmarking method. One of the fundamental requirements of benchmarking is to have data banks where benchmarking can be performed. There are currently many data banks storing much best practices for others to benchmark. These data banks are constantly evolving. Progress is also being made regarding methods that can be used to achieve a specific benchmark goal. The problem in the past was focusing on the data collection process instead of the type of data being collected. Hence, that led to the lack of focus on the kind of desired benchmark result. Worst of all, most benchmark results cannot be correlated across various benchmarking systems. This is another reason why this thesis approach is to use known comprehensive standards as a foundation for benchmarking, therefore, the results can be used to compare across various benchmark systems.

In 2001, a benchmarking seminar [Asian Productivity Organization, 2001] was held in India to gain an in depth understanding of the benchmarking practice and how to use it to achieve organization and business excellence. Some of the benchmarks are software-related engineering industries such as the Information Technology industry. The countries present at the seminar were: Bangladesh, Republic of China, India, Indonesia, Malaysia, Mongolia, Nepal, Philippines, Singapore, Sri Lanka, Thailand, Vietnam, Bhutan, Myanmar, and Australia. This is not only an impressive list showing the wide spread of benchmarking in Asia, but the results presented were equally impressive in their outcome. One of the most interesting papers presented was by Australia. With over 10 years of benchmarking experience, Australia has showed that a self-assessment then a benchmarking methodology approach has proven beneficial to over 400 Australian enterprises and over 1300 corporate members. One thing the enterprises all have in common is their inability to share their benchmark items. Therefore, this thesis is proposing the utilization of standards to overcome this hurdle of incompatibility of benchmark items to maximize the compatibility of benchmark partners.

1.5 Problems to be Addressed in this Thesis

One of the main drivers for this thesis is based on the author's industrial experience that most organizations are financially focused. A company will not survive without generating revenue. However, this will not provide sufficient insight into the business process of the organization or make the organization healthier. Like an entity, it is the internal flow of the entity such as the process of blood flowing that ensures the livelihood of an entity. Even though financial indicators do provide useful information to an organization [Gildersleeve, 1999]; they can be complemented with standards-based process benchmarking indicators. Based on the author's interview with several organizations' managers, the consensus is to put the focus on running an organization or project efficiently and effectively than its competitors and revenue will come by itself.

1.5.1 Problems Identified

The main problem of benchmarking is how to make it efficient and effective. The benchmarking process needs to be efficiently implemented and executed. Furthermore, the results need to be effective in providing objective support; specifically for software process decision-making that can impact an organization's goals. If benchmarking is to be used, the question is how can executives or upper management identify external industry best practice and internal improvement opportunity? If a benchmark is to be usable for upper management, the benchmark has to support top-down analysis for the organization. Ideally, a top-down decision should be verifiable by bottom-up analysis. Does a proposed benchmarking mechanism provide such dual direction decision-making mechanisms? In benchmarking terms, top-down analysis requires external benchmarking, while bottom-up analysis requires internal benchmarking. How can the threshold for benchmarking be determined for both external and internal benchmarking for the organization? Furthermore, how can the internal organization benchmark be related to the external organization benchmark? Finally, assuming that answers were found to these questions, how could these answers provide practical value to industry? This thesis does not promise all answers to business success, it does attempt to provide an effective and objective support to business decision making by using benchmark as a mechanism to determine the best practice to be emulated that hopefully results in improved performance in the industry.

1.5.2 Research Approach of this Thesis

This research approach relies upon a systematic way approach that attempts to solve those challenging problems identified in the previous section. The design of the tool, research design and framework are provided. Finally, a structure of the thesis is provided to supplement the research work approach as laid out in this thesis.

1.5.2.1 Approaches to Solve the Problems

The problems identified in Section 1.5.1 beg the questions why, what, how and when can benchmarks be used to solve the problems? Benchmarks are essentially chosen because the author believes that benchmarking holds merit in providing value-added objective support to an organization's executives, particularly by making effective continuous SEP improvement.

One of the main research areas in benchmarking is gap analysis. In general, gap analysis provides valuable insights into the magnitude of an organization's current state in comparison to its threshold state. This magnitude can be measured differently based on the benchmark used and the results could be for better or for worse.

The goal of this research is to provide suggestive evidence that benchmarking can be used as valued method that will provide an organization's executives with reasonable objective support for SEP-related decision-making. This will be achieved by first investigating benchmarking methods and the results produced. Secondly, the benchmarking methods are then implemented by using real-world data. The real-world data are obtained from industrial projects. Thirdly, and finally, the results are analyzed and interpreted as to the benefits and value in supporting an industry's executive decision-making. Executive decision-making process could be related to marketing, gathering support evidence, and improving process just to mention a few examples. For the purpose and scope of this thesis, the emphasis is on the utilization of benchmark for SPI to achieve superior performance in the market place.

1.5.2.2 Research Design

The major part of this thesis is designed to collect assessment data from industries and benchmarking them. The assessment method must be recognized by the software engineering industry. This means existing software engineering standards and models are the best candidates for deriving the assessment methods. In addition, using known standards and models provides guidelines and limits suggestions for improvement on how to achieve a specific capability level.

- Research Framework

The research framework for this thesis is illustrated in Figure 1-1. The foundation for this research is based on the Software Engineering Process Reference Model (SEPRM) developed in [Wang and King, 2000]. SEPRM is utilized to provide a comprehensive process capability assessment that can be easily mapped onto various software engineering standards and models, such as CMM, ISO 9001, BOOTSTRAP and ISO/IEC 15504.

After the assessments of various projects have been collected, the projects will then be analyzed against the benchmarks. A key part of benchmarking analysis is the gap analysis used to support benchmark-based process improvement.

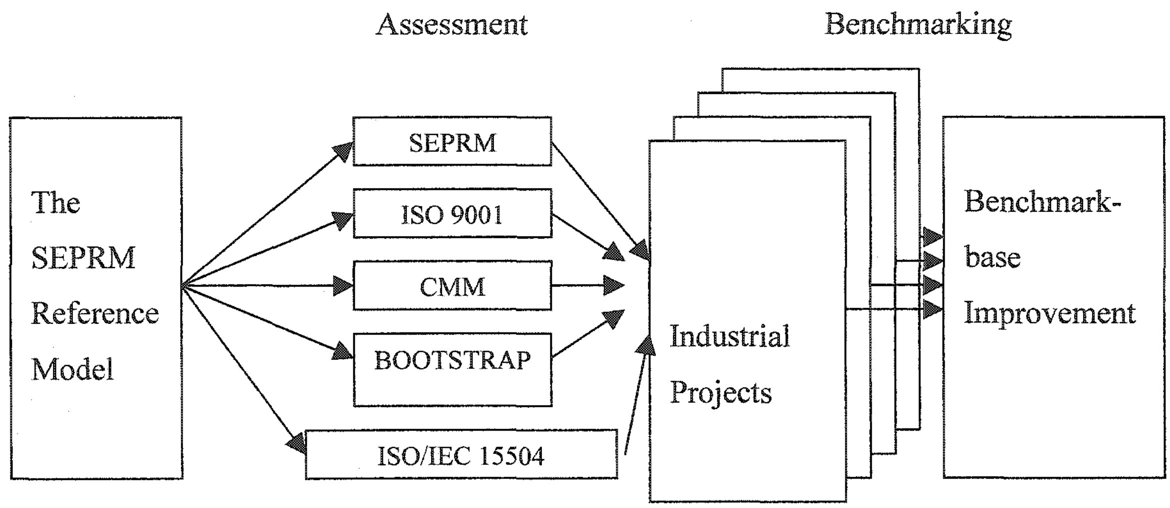

Figure 1-1 Research Framework

- *The Use of SEPRM as the Foundation to Benchmarking*

SEPRM is used as the foundation to benchmarking because it provides a comprehensive and integrated structured methodology assessment. SEPRM supported by a set of comprehensive assessment data and is ease of use in process assessment. The advantages of SEPRM are it's mapping of various standards and models process to a reference model and its process assessment algorithm.

The main benefit of performing benchmarking based on SEPRM is the ability to efficiently and effectively benchmark various well-known software engineering standards and models. SEPRM is efficient because it provides a comprehensive one-time SEP assessment per project to generate one academic SEP standards and four industrial SEP standards. This means a project does not need to be assessed five times, thus saving effort of almost five times. It is effective because it provides internationally well-known industrial standards that most organizations can work with relating to SEP. Furthermore, these standards provide a reusable benchmark that organizations can constantly refer to is especially useful for CPI. Using SEPRM can provide cross-functional benchmarking result because it is based on known standards as its basis. This is similar to building an open benchmark system, where similar software process standards can reuse each other's benchmark results.

- *Research Tools*

The research tools were selected on the common applications found on a manager or executive computer desktop. These applications include a database, a spreadsheet and a word processor. They are integrated functionally but can be used on a stand-alone basis. This is to keep in line with the philosophy of making the research tool optimally automated. The database is used to collect and store assessment data and perform limited data pre-processing. The database also performs calculation on the data to generate information related to the various assessment and benchmark standards and models. Finally, a word processor or presentation application can be used for result presentation purposes.

1.5.2.3 The Assessment and Benchmark Systems

- The System Architecture

The conceptual system architecture used in this work is the Model View Controller (MVC) Design Pattern [Gamma, et al., 1998; Larman, 1998]. In this thesis, the model is the database, the controller is a database or the spreadsheet, and the view is a database, spreadsheet or word processor.

Assessment data can be input into the database directly, via a web page or imported through a spreadsheet. Once all data is stored in the database, at a press of button, the application calculates and generates assessment and benchmark results. The results can then be presented using the database application or export to a spreadsheet or word processor for various types of presentations. The presentation aspect is currently semi-automated and not real time, due to the thesis schedule limitation.

- *The System Framework Implementation*

The system framework was designed spanning various desktop applications to maximize utilization of existing capital cost and minimize overhead cost. The next chapter will provide explanation of how benchmarking can help best allocate capital cost and better manage overhead cost. Furthermore, modification to the application is relatively easy with minimum to no programming knowledge required. Basically, the benchmark application has to be robust and easy to use yet powerful enough if an executive was to use it. Finally, adding a new standard is the hardest task. The task involves generating a table of the standard process mapping to SEPRM and write an assessment function based on SEPRM.

The original intent of this research is to implement benchmarking based on demographics and sectors, and possibly including a project's resource size. But, due to the small size of available projects for benchmarking, all of the projects from a benchmark database are used, and one project is singled out and used as a case study to show the benefits and usefulness of benchmarking in an industrial application.

- The Benchmark System Benefits

The benefits of a benchmarking system in software engineering can be described below:

- Ease of mapping of new standards and models using SEPRM
- Benchmark can be used immediately against a new standard
- Benchmark can be used to analyze how old project process framework and methodology fair against upcoming competition, other framework and methodology, other internal projects, competitors within the same industry, and others using similar processes
- Benchmark can be used to determine future business that is a standard base for cost analysis of standard implementation and adaptation
- Benchmark and assessment using SEPRM can provide a roadmap for CPI
- Evaluate strength and weaknesses (process state)

- Identify areas to implement process continuous improvement (good process)
- Identify areas to implement process overhaul (bad process)
- Identify areas to implement process installation (new process)

In the industrial context, the methodology for deriving objective support and benchmarking system has to be efficient and relatively easy to implement. The benchmark results have to show potential effectiveness in answering industrial problems. These are the practical objectives of this work with a strong industrial orientation.

1.5.3 The Structure of this Thesis

This thesis is organized into six chapters. Chapter 1 is an introduction to the thesis and initially describes the aims and objectives of the thesis. Literature review, problem identification and research, design can be found in this chapter. Chapter 2 provides background information on the state of the art of benchmarking including some historical information and the current state of benchmarking relating to software engineering. Chapter 3 proceeds with an explanation on one of the most important fundamentals of SEP benchmarking, that addressing SEP assessment. Chapter 4 follows up on SEP assessment by extending the assessment results to the benchmarking process. This chapter describes in details the newly proposed SEP benchmarking methodology and SEP benchmark gap analysis technology. Case studies are provided in Chapter 5 that illustrates the application of the benchmarking methodology and gap analysis technology in an industrial application. The last chapter, Chapter 6 concludes the achievement of this work and perspectives on future work.

A the high-level description of the outline of this thesis is provided below:

Chapter 1. INTRODUCTION: This chapter provides an introduction to the aims and objectives of this thesis and the underlying motivation to do the research work. Literature review of benchmarking historical past and current state is provided as information leading to the problems identified for this research work.

Chapter 2. BENCHMARKING: STATE OF THE ART: This chapter begins by defining benchmark and benchmarking. A general benchmark and benchmarking explanation is included in this chapter. A detailed explanation of benchmark gap is explained. Benchmark gap is the essence of benchmarking analysis. Various perspectives on benchmark values from goal focus, financial focus, process focus and knowledge focus are provided.

Chapter 3. SEP ASSESSMENT: This chapter develops general SEP assessment methodology and implement algorithm based on CMM, ISO 9001, BOOTSTRAP, ISO/IEC 15504 and SEPRM. An architectural diagram is provided to show how SEP assessment results are used as inputs to SEP Benchmarking.

Chapter 4. SEP BENCHMARKING: This chapter explores SEP benchmarking design, framework, methodology, approach and implementation. The focus of this chapter will include the theories of benchmarking and applications of the benchmarking tool.

Chapter 5. INDUSTRIAL CASE STUDIES: This chapter describes case studies that illustrate an actual SEP benchmark application in industry, accompanied by information on benchmark implementation, assessment, analysis, summarization and discussion.

Chapter 6. CONCLUSIONS: This chapter summarizes the work carried out in this thesis,, including SEP assessment and benchmarking theories and techniques and supporting tools. Perspectives on future research are provided for related future work on SEP benchmarking, which may based on this thesis.

APPENDIX C:APPENDIX D:The six chapters provide the flow of this thesis by introducing the aims and objectives relating to the problems identified. A little background information is provided on the state of the art of benchmarking that is used to solve the problems. The SEP assessment is described in an independent chapter on its own, because it is the foundation and basis from which the SEP benchmarking is based upon. The subsequent chapter on SEP benchmarking provides technical details on the proposed new benchmarking methodology and gap analysis techniques. Case studies are provided to supplement the software engineering benchmarking methodology and, to verify the usage of the research results through an industrial implementation. Finally, conclusion ensues to justify the achievement and accomplishment of the research work.

CHAPTER 2 BENCHMARKING: STATE OF THE ART

This chapter provides background information on the history and origination of benchmarking technologies. The value of benchmarking is analyzed. Industry applications and perspectives are also discussed.

2.1 Introduction

An organization may perform two kinds of benchmarking: internal benchmarking or external benchmarking. Internal benchmarking is trying to measure a given project based on internal benchmarks. External benchmarking is intended to measure a given project based on external benchmarks. Card and Zubrow [Card and Zubrow 2001] provide an interesting view of benchmarking from the customer perspective by stressing its importance. From the customer's point of view, internal benchmarking provides the customer with an organization's competency information [Spence, et al., 2002] and external benchmarking provides the customer with information if they are dealing with the best in the market place. That is why it is essential for an organization to always strive to be competent and be the best.

An International Study [Ernst and Young, 1992] on over 900 management practices covering over 500 organizations concluded that an organization must look at all processes, then select those with the most impact on financial and/or market place result, and get lean. The study stated that short-term result could be achieved by basic cost-cutting action. Long-term benefit results are achieved by elimination of non-value added time and resource that have long-term positive effects. The study also pointed out that assessments are important process in many high performance organizations that valued quality. Furthermore, processes must have an external focus. This is where benchmarking comes into play. The study found that top quality organizations frequently use benchmarking to help their high-performance achiever gain competitive advantage.

In the next sections, benchmarking concepts will be presented. It shows that a specific industrial section can gain a lot from learning what other sectors and disciplines know about benchmarking and how they are using it to help their organization.

2.2 Definitions of Benchmarking

A well know benchmark practitioner, Robert Camp [Camp, 1989] provided the following definitions for benchmark, benchmarking and best practices:

Benchmark

- Formal definition: Benchmarking is a continuous process of measuring out products, services, and practices against the toughest competitors or those companies recognized as industry leaders.
- Working Definition: Benchmarking is the search for best industry practices that will lead to superior performance.

Benchmarking

- A continuous, systematic process for evaluating companies recognized as industry leaders.
- To develop business and work processes that incorporates "best practices" and establishes rational performance goals.

Best Practices

- The methods used in work processes where outputs best meet customer requirements.

Robert's definitions are process oriented and do not focus on product or service. Two other important elements to note are the benchmarking process is a continuous process and a systematic process. These elements are keys leading to superior performance in the market place.

Andersen and Pettersen [Kulmala, 2002] provided a much simpler and basic benchmarking definition: “A predefined position, used as a reference point for taking measures against”. Others, like Spendolini [Kulmala, 2002] provided the following benchmarking definition: “Benchmarking is a continuous, systematic process for evaluating products, services and work processes of organizations that are recognized as representing the best practices, for the purpose of organizational improvement”. Notice how Spendolini’s definition is similar to Robert Camp’s definition, including the stressing of continuous process and systematic process.

Relating these definitions to industry practice of software engineering, this thesis research would define SEP benchmarking as follows:

> *SEP benchmarking is an integral part of an organization’s systematic CPI effort to achieve superior performance in the market place by measuring and analyzing its process against a given benchmark.*

With these definitions in mind, we can proceed to probe a little deeper into the meanings and applications of benchmarking. Details of this definition are explained in Chapter 4. According to Juran [Juran, 1995], the concept of benchmarking grew out of organizations’ need to establish quality goals. These goals have to be based on factual analysis instead of empirical judgment. Through benchmarking, one can discover if a goal has been achieved or not by others. If a goal has been achieved by others this implies the goal is reachable and doable. There are skeptics that say, “It can’t be done”. Similar argument is that almost all industries are not related and the business is not similar. In the early twentieth century, some German generals visited the Bailey Circus to analyze how the Bailey Circus, regarded as owner of world-renowned moving methods, transport equipment, animals, personnel and other from city to city. The German generals were trying to learn from the best movers how to transport military personnel and heavy equipment just like the Bailey Circus. This is just one of many examples of how benchmark can help in discovering potential end solution to similar problems and potential methods to achieve a solution. This example also provides a good argument to support process benchmarking. One should not focus on activities but focus on the process level. An organization that focuses mainly on its specific industry's technology might not have discovered new ways (process) of achieving similar goals from other industries.

Benchmarking goals should be aligned with an organization's goal. Similarly, a benchmarking purpose should be aligned with an organization's purpose. The fundamental underlying goals of using benchmark should be to achieve an organization's purpose. An organization should not idealize the process of benchmarking just to beat others simply for the challenge, or at all cost, during the comparison process. Benchmark should be used to assist an organization in achieving and realizing its goals. The benchmarking process can also help in determining an organization's process- related goals. The benchmarking process itself should not be an organization's goal. The customers and market feedbacks can be used to validate some of these goals.

Next, various types of benchmarking and benchmarking purposes are reviewed to determine how benchmarking can help to make an informed decision on which type of benchmarking process to use. Juhani Kulmala [Kulmala, 2002] has identified for us four types of benchmarking as follows:

1. *Strategic benchmarking* – The purpose of this type of benchmarking is to analyze how competitors are competing and the use of different competitors for long term and short term strategic planning.
2. *Performance benchmarking* – The purpose of this type of benchmarking is to perform assessment for determination of competitive position in the market place. The focus is on an organization's key processes, products and service.
3. *Process benchmarking* – The purpose of this type of benchmarking is for process improvement by selecting from similar functional operations. The focus is to improve an organization's methods and activities.
4. *Competence benchmarking* – The purpose of this type of benchmarking is to determine an organization's competency in change of actions and an individual's

behavior. This is also known as benchlearning. The philosophy is that operation effectiveness is determined by the development of an individual's skill and attitude.

Later, this thesis will show how, just by using process standards as the basis for process benchmarking, it is possible to fulfill the purposes of strategic benchmarking, performance benchmarking, process benchmarking, and competence benchmarking.

The benchmarking process will not function well without a framework. A benchmarking framework must be defined before benchmarking can be executed. Fogle, Loulis, and Neuendorf [Fogle, et al., 2001] define benchmark framework as established topic areas of benchmark study and comparison. The benchmark framework will grow as the benchmarking process is progressing, by adding details to the topic areas. In order to benchmark, best practice must be identified. They define a best practice as one that meets the following criteria:

- *Existence* - The practice must have been observed in at least one partner organization.
- *Importance* - In the benchmarking team's opinion, the practice is important to the organization.
- *Effectiveness* - In the benchmarking team's opinion, the practice appears to work well where it is used.
- *Tangible benefit* - In the benchmarking team's opinion, there is a tangible benefit to the organization that performs this practice.
- *Innovation* - Where appropriate, the practice makes use of innovation, such as use of automation instead of manual methods for accomplishing a task.
- *High perceived value* - The perceived practice value must be relatively highly based on customer feedback.

Using these criteria framework, as guidelines to identify best practice, will increase the effectiveness of the benchmarking process. One easy way to identify most of these criteria is to use process standards that are usually pre-approved and accepted by the customers and business competitors. Obviously, this is referring to the external customer. Similarly, these criteria are applicable to internal benchmarking by using organizations standard procedures that are widely acceptable by an internal organization's customers.

Now that the best practice framework for benchmarking has been identified, the next step is to determine the appropriate benchmarking techniques to use. The following list consists of various types of benchmarking techniques mentioned by Juhani Kulmala [Kulmala, 2002].

- *Internal benchmarking* – benchmarking within an organization.
- *External/ competitive benchmarking* – benchmarking external to an organization involving competitors.
- *Functional benchmarking* – benchmarking selective functions by focusing on any competitors in any businesses.
- *Generic benchmarking* – the purest form of benchmarking. This form of benchmarking is difficult to undertake. The benefits of generic benchmarking are to identify best practices in an industry and operational functional deficiencies.

A later chapter on benchmarking will show how internal benchmarking can be correlated to external benchmarking. Furthermore, the functional benchmarking is encapsulated by the utilization of a comprehensive standard of processes for benchmarking. By using a comprehensive standard that is supposed to represent "comprehensive" processes, means this form of benchmarking is also a form of generic benchmarking. Hence, the proposed benchmarking methodology of this thesis could be used for internal benchmarking, external benchmarking, functional benchmarking and generic benchmarking.

The benchmarking processes themself consist of logical systematic steps. Here are some steps as suggested by Fogle and Associates [Fogle, et al., 2001] and Kumala [Kulmala, 2002]. The first step is to get upper management to sponsor and buy into the benchmarking project. Secondly, a benchmarking plan must be developed to determine the approach to be undertaken; such as, what to benchmark and who to benchmark against. Next, a benchmarking team must be formed to collect the data, analyze the data and evaluate the data. Finally, a report will be generated and appropriate action taken. This benchmarking process can be part of an organization's CPI program, to determine if the current state of benchmark results exceeds the previous benchmark. To ensure a successful benchmark, the benchmark objectives must be well defined, carefully planned and cautiously interpreted. Objectives must be well defined which means one has to keep a narrow focus on the effort of the benchmark item to learn. This involves careful planning to ensure the benchmark process clearly states the objectives and collect the appropriate data relevant for the right analysis. With the analyzed data on hand, one should uncover and understand the limitation of the data and not to generalize beyond the scope of the benchmark study. Benchmark findings and interpretations can be validated by customer feedbacks.

Card and Zubrow [Card and Zubrow, 2001] also pointed out that it is usually due to the lack of consistency remains a serious obstacle to efficiently and effectively benchmarking. That is why standards and models are very important in assuring consistency, repeatability, manageability and predictability. These are the essence of capability maturity of level that this research thesis is trying to benchmark. This means that the benchmark result itself can be a part of an organization's metric.

2.3 Benchmark Gaps

Benchmark gap is the heart of benchmarking. How well one comprehends the gap, analyzes the gap, and evaluates the gap can make or break the benchmarking effort. Comprehension of the gap usually involves knowing the reason for the gap and recognizing the importance of the gap. This is the qualitative aspect of the gap. Gap analysis involves a gap measurement process. How and where to measure the selected gap is critical to understanding the magnitude of the gap. This is the quantitative aspect of the gap. The interpretation of the evaluated gap will determine if any action or appropriate action will be taken, can determine the faith of the gap. This is the value-added aspect of the gap. The value of the gap determines the action performs on the gap.

2.3.1 Definition of Benchmark Gaps

The essence of benchmark is essentially gap identification, analysis and evaluation. The benchmarking process measures gap magnitude of the selected processes of interest. Hence, the immediate outcome of benchmarking is the gap information. Gap is a distance measured between the current competitive gap state and the benchmark gap threshold. There are three kinds of gap. They are positive gap, parity gap and negative gap [Camp, 1989]. Positive gap is usually regarded as gap that is in favour of an organization, while negative gap refers to a weakness in an organization. Both positive and negative gaps are relative strengths and weaknesses to the item being benchmarked. In this thesis, the benchmarks are both the standards and the competitions. An organization can use one or both the benchmarks. Just like a runner can run against the clock (standard) or other runners (competitions).

Regardless of the reference benchmark point, the resultant is gap. Gap measures an internal organization's performance and the external best in industries. A positive gap should receive recognition while a negative gap should be viewed as opportunity to capitalize on the practice. Two main objectives of the gap should be to provide practice opportunity and act as a performance matrix. The practice opportunity is to encourage an organization to better a negative situation instead of blaming others for the negativity.

One should always bear in mind that the essence of benchmarking is a comparative analysis, with basic gap analysis and understanding the differences. Regardless of the gap type the benchmark gap should be analyzed as to its reliability of information and that the gap is justifiable. Organization's information could help in explaining the gap by focusing on industry sectors, geographical locations, project size, etc. In the end, the final result is to use an organization's goals to verify the justification of the gap and have the customer or market place validate the justification.

2.3.2 Gap Analysis

Any gap that has been identified must be qualified first prior to being quantified. An identified gap must make organization business sense. Once the gap has been identified as it makes business sense, then the gap is worth investigating. Looking into the gap usually involves gap analysis. As part of the gap analysis, it is very important to be able to measure the magnitude of the gap. The ability to measure the gap enables an organization to perform continuous process improvement. With measurement information, an organization can tell how far they have come from and how much more they will have to go. It is a common mistake to measure the gap first and use the gap measurement as justification for the gap. The mistake is the switch from process focus to gap focus. If a process does not make sense to the business, the organization should not be measuring the process gap but try to improve on the gap. One may focus on measurement, because the measured value can be used to determine the process selected for CPI. Hence, the selected process should be qualitatively analyzed to align with an organization's purpose and goals.

The process to be quantified must first be qualified for justification to make business sense. That is, the process to be quantified should be a process enabling current operations to be like the desire best practice at a future time. The quantified number is a synthetic number, nonetheless a true benchmark number. This number should be budgetable, and based on a desired performance measure. Achievability of the quantified process, based on the budget, is questionable and requires detailed analysis. The quantified process should be able to build up from a clean slate.

The quantified number is an indication of a desired level to obtain and indicates progression of improvement based on the magnitude of the quantified number of the achievement. The number should be tracked and monitored to ensure alignment with competition change.

Comparability is important in validating the benchmark data, not the benchmark result. If no data can be compared no data can be validated against. Caution must be taken to ensure acceptability of the comparison when relating it to cost per desired metric. Basically, the cost per desired metric can be used but must have validation and acceptance finalized for the benchmark data to ensure true benchmark statement.

Once the gap has been identified and analyzed for validity, the next step is to achieve superiority. The focus measurement is now based on the benchmark matrix. Benchmark matrix is collected based on performing more than one benchmark both internally and externally for justification. Robert Camp [Camp, 1989] pointed out that this step involves investigation of the cost of the process or lack there of, as part of the CPI effort. The idea is to quantify the size of the benchmark gap. This enables the unit cost to be calculated by averaging the cost over the average of the gap size. It is a good idea to perform more than one benchmark analyses to ensure reliability of the benchmark result. The benchmark should be performed both internally and externally to the organization for justification.

2.3.3 Gap Evaluation

Practice opportunity is the gap that has been selected as candidate for action item. Practice opportunity also refers to the CPI strategy. The strategy goes beyond determination of the current state and the future state. There is valuable information in the gap to provide effective strategy planning. One basic approach is to break down the selected gap into activities, and work on the weakest activity.

The point here is differences identified in metric cannot be understood without understanding the practices involved. The question to ask normally at this stage is to explain the differences than determine the magnitude of the gap. It is important to go to the level of detail understanding as to the key process steps, cost analysis, relation between inputs and outputs and finally, the level of measurement. To reiterate, that is why the practice should be qualified first than quantified. Performing qualitative analysis will identify the benchmark gap of interest. A quantitative analysis will determine the magnitude of the gap. This measure of magnitude of the gap corresponds with the opportunity of the gap. Frequently benchmarking of the industry can provide benchmark performance metric. Performance metric provides gap magnitude for operations to judge the degree of difficulty in achieving the desired results.

Robert Camp [Camp, 1989] identified three major components of practices. They are process practices, business practices, and operational structure. Process practices refer to the front-line activities, and executed on behalf of business practices. Business practices refer to the project level where the resource is allocated to ensure the activities are executed. It focuses on methods and contributes to efficiency. This is achieved using standard, policy, mission statements, etc. Finally, operational structure refers to the framework where all practices are operating in. It is difficult to change over time due to its rigidity.

2.3.4 Gap Interpretation

Gap itself can provide useful information. Gap can be used as a predictor for future performance level and for organizational strategic and tactical planning.

2.3.4.1 Predicting Future Performance Level

Gap can be used to project future performance level. This will not be covered by this thesis because the scope of this thesis is based on current state and the potential action that can be executed immediately. The main reason for not performing this part of the benchmark methodology is due to time constraint. This involves collecting historical data and analyzing it and will involve more than two years worth of research time, especially when it involves CPI. It will be mentioned here for completeness and provide readers ideas for potential future research.

It is not enough to just understand the current gap state, but to try and take advantage of the gap. One should try to predict and anticipate the gap dynamics over time. This entails continuous benchmarking, monitoring, and tracking of benchmark gap. Being able to anticipate gap future magnitude, such as gap widening or closing, better enables operations to manage available resources in a timely manner and perform scheduling to sustain the CPI momentum. Forecasting ability is a very important arsenal for strategic planning to ensure organizational survival by closing the gap and exceeding the benchmark threshold in a predetermined timeframe.

The ultimate goal of benchmarking is to achieve superior performance. It is important to realize that as you try to improve your own process, so do your competitors. That is why this thesis has an emphasis on analysis of how to get ahead of the competitors based on the current state of information of an organization and its competitors.

The conceptual project of the benchmark gap is illustrated in Figure 2-1 [Camp, 1989]. The horizontal axis provides a timeline and the vertical axis provides the process measures to be tracked and monitored. This chart requires historical data, and with historical data, can one then generate trend line for analysis. This chart enables one to forecast and anticipate when the gap will be closed (the intersection of the two lines). The "Endpoint" indicates the future date where the desired magnitude of competitive edge is located.

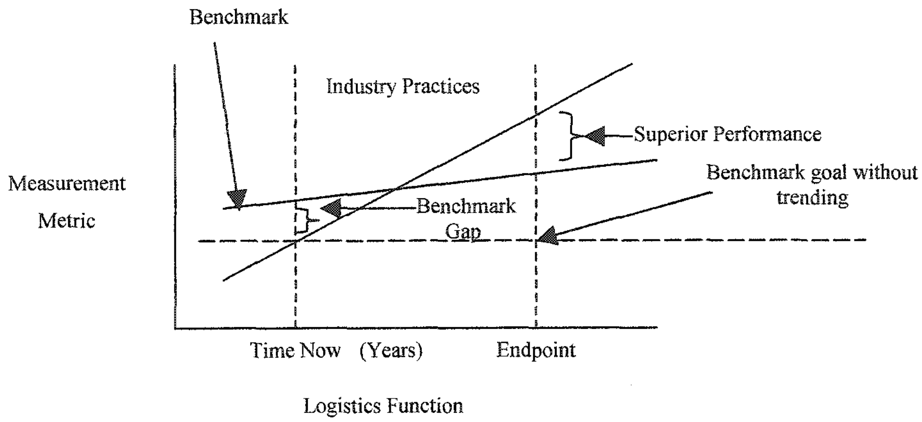

Figure 2-1 Conceptual Benchmark Gap Projection

The most important item to note if benchmark trend is not monitored, is a possibility of having the gap widening at the end of the planned CPI cycle. An arrow labeled "Benchmark goal without trending" shows where the future state would be without taking into considering trending the historical data for analysis and action planning.

There are many ways to monitor trend lines. One popular benchmark trend line is The "Z" Chart [Camp, 1989]. The "Z" Chart is similar to Figure 2-1. The "Z" Chart contains three important components consisting of such information as historical productivity, cost reduction and trend. The "Z" Chart can portray the magnitude of effort required to perform the business unit analysis and understanding.

2.3.4.2 Strategic and Tactical Planning Based on Gap Analysis

It is very important to understand the gap for strategic actions, tactical actions and practice changes. Strategic actions are important to ensure external benchmark findings correlate with internal benchmark findings. Strategic actions provide organization-wide CPI. It is a link to provide check to internal tactical practice to ensure competition will not outdistance the internal organization performance. Strategic actions involve long-term planning.

Tactical actions are translated into strategic actions. It generates data for input into metric that show the resultant effectiveness of an identified benchmarked process. The metric results are fed into the benchmark systems. Tactical actions are a more gradual change process because it is the actual working process at the activity level. Tactical actions output generates historical data for analysis and trending. Tactical actions are very important because it relies on functional and operation knowledge for execution.

Practice changes should be aligned with company goals. Practice changes are similar to the Goal Question Metric (GQM) paradigm. This involves correlating all activities related to an organization's desired goal. The goal dictates the more significant practices to be benchmarked and then improved as required. The real advantage of using benchmark result as a goal is that the goal is achievable and doable.

Furthermore, benchmark findings are excellent candidates to be used as a company's target such as goal statements. This will provide direction for CPI. This is an interesting bottom-up approach to set and to achieve strategic goals.

2.4 Industry Benchmarking Applications and Perspectives

It is very important to have upper management buy-in that benchmarking is a valuable process investment. For example, Thomas G. and Smith H. [Thomas and Smith, 2001] viewed structured benchmarking as panacea. They got management buy-in by convincing management that benchmarking will help in identifying risks and providing potential guidelines for abatement strategies. More good advice provided by Thomas and Smith is to have a cross-functional department to execute the benchmark to achieve synergy, thereby providing insight and appreciation to the overall operation of the organization to all employees. Thomas and Smith's idea will also show how the benchmarking process can span organization wide.

The view expressed by Thomas and Smith [Thomas and Smith, 2001] is a very common response by industry organizations. About 75% of the managers interviewed by the author share the same sentiment, that quality is an overhead cost with no real value. The real value of quality is only a perceived value by the customer. About half of 25% of the interviewed managers who believe in quality also believe in the value that benchmarking can provide. One of the goals of this thesis is to promote and to sell the value of benchmarking to 75% of the managers who do not believe there is a significant value return on quality investment to justify further spending on the quality cost.

The following subsections will show how benchmarking can be correlated to an organization's goals, finances, processes and knowledge management. These are the industry objectives that this work is trying to provide a way to illustrate their correlation.

2.4.1 Goal Focus

Goal focus will concentrate on the generic goal of an organization and how the benchmarking process can help an organization achieve it. An organization usually has both external goals and internal goals. That is why an organization should practice both external benchmarking and internal benchmarking.

2.4.1.1 The Ultimate Goal

Let us begin by looking into benchmarking against the ultimate goal. Conway [Conway, 1992] coined the term benchmarking utopia that is akin to a benchmarking ideal. This is similar to choosing a software engineering model and trying to achieve the highest maturity level as possible. Conway called benchmarking against utopia as the ultimate benchmark. Sometimes competitors are just busy trying to stay abreast of competitors and not worrying about being the best. This is normally how benchmarking is being used these days, without a continuous tracking and monitoring of the competitors. The point that Conway is trying to make is that sometimes one might just want to pull ahead of the rest of the field of competitors. Conway provided the story of Roger Bannister who ran the four-minute mile in Oxford, England as an example of breaking a barrier. He set a threshold for all others to strive for when Roger broke the barrier. Prior to him breaking the barrier, most runners were content with beating the rest of the runners in the field. Ten years after he broke the four-minute mile barrier, eight people surpassed the barrier in one race.

One evident benefit of benchmarking the ultimate is that it makes the field more interesting and worthwhile for those who truly want to compete. Essentially, it raises the benchmark threshold high and encourages industry to be the best, hence leading to superior practice maturity within a given industry. In addition, utilizing a known, recognized and accepted ultimate maturity model provides an organization with the best practice a new finishing line to shoot for for the rest of the field who will be coming along to catch up. In the example of running the four-minute mile, the quality of running heightens and the sporting event is more challenging to other runners. It also introduces an additional goal to the run; such as, besides being better than others, you can strive to exceed the ultimate benchmark threshold and be the best in the field. The four-minute threshold is a benchmark that everyone in the world can try to achieve even without having to run against all the runners in the world. This clearly shows the benefit of benchmarking against a maturity model as the reference measurement for benchmarking. Similarly, this thesis proposed various SEP standards and models as yardstick for measurement. This enables every Software Engineering industry in the world to compete on common ground. This has the benefit of enabling organizations potentially expand market share to different parts of the world.

Another important new point is that benchmark should not discriminate against people. The example of Roger Bannister, the runner, breaking the four-minute mile, means that others can now strive to beat the new standard threshold, instead of beating Roger Bannister. Focusing on a model provides a rational and objective benchmark; focusing on a person or an organization for benchmarking is difficult and subjective. Essentially, the focus is not on a specific particular problem requiring expert knowledge people. Standards or facts-based benchmarking enables anyone armed with the proper fact do achieve the same goals by improving the process to achieve the facts. In the Software

Engineering industry, an application can be treated as a generic product. As a generic product, all software industries can benchmark among themselves by focusing on their product manufacturing process. For example, accounting applications can benchmark against the various accounting module within the accounting package. It does not make much sense to benchmark an accounting package to a software game. That is why it is very important to have a clear goal and purpose for the benchmark that will make sense to an organization's business practice, and then appraise the business practice.

There are organizations that compile large amounts of benchmarking data from many organizations within their benchmark repositories. On such organization is the International Software Benchmarking Standards Group (ISBSG) [Lokan, et al., 2001]. ISBSG stressed that an organization must know its own performance prior to bring compared to another organization's performance. Finally, determine what performance is possible. Benchmarking should always begin with the question "Where are we?"

2.4.1.2 Benchmarking and the Goal Questions Metric Paradigm

One of the most well known goal-oriented paradigms is GQM. The GQM paradigm is useful for continuous software process improvement, among many other benefits [Grady, 1992]. Benchmarking can help with a company's strategic and tactical planning. The Strategic planning could involve the determination of an organization's goal and relevant questions that involve software process. Tactical planning could involve the determination of the priority of questions to measure first for effective return on investment (cost allocation efficiency) base on best practice. The GQM paradigm can also be used to ensure measurement relevancy and organizational support. Benchmark software process result itself can be measured as part of the GQM paradigm implementation; hence, creates a cycle for CPI both using benchmarking and measurement. For details on GQM software process related implementation, check out these articles, "Using the GQM Paradigm to Investigate Influential Factors for Software Process Improvement" [Mashiko and Basily, 1997] and "Integration of system dynamics modeling with descriptive process modeling and goal oriented measurement." [Pfahl and Lebsanft, 1999].

Generally speaking, most organizations perform short- (tactical) and long- (strategic) range goal planning. A short-range plan is usually an annual plan; a long-range plan can span many years. Benchmarking can play an important role in both forms of plans. In the short term, benchmarking provides the current state of capability level as compared to the competitors, and internal state of operations. Benchmarking helps with long-term strategic planning by monitoring external benchmarks and providing objective feedback to an organization for resource planning and scheduling of future opportunity plan. In either short or long term plans, benchmarking helps an organization with business planning by providing a factual magnitude of changes required to stay competitive.

2.4.1.3 *Strategic Benchmark Planning*

An organization's functional goals can be established and derived from benchmark findings. This is an effective approach for setting an organization's goals because the goals would have been proven achievable. These benchmark-derived functional goals are not only achievable but and are the best practices. First of all, the goals are realistic (someone has achieved it) and secondly, exceeding such a goal means potentially being the best. Obviously, achieving such a goal means the potential of outperforming the competitors. This could imply getting an edge in the market place.

2.4.1.4 *Tactical Benchmark Planning*

In almost all software-development organizations, measurement results are used to make decisions at two primary management levels namely the project level and the corporate level. Project level focuses on a project. A corporate focuses across multiple projects. Corporate desire for comparative measurement results has been the basis for many failed measurement initiatives that required all projects to use a predefined set of mandated measures. John McGarry [McGarry, 2001] says that an individual project has a different focus and goal such as schedule constraint or cost constraint therefore uses different measurement.

The causes of most bottom-up measurement fail because they focus on details such as activities that will definitely be different between projects depending on contract requirements. Even though software projects may differ approaches could be similar [Ebert, 1999]. If the focus was process based then there should be less discrepancy between projects especially if the process is related to an organization's way of doing business. Another way to increase process coherency between projects is to implement automatic measurement data collection. This way, it will force the projects to have conformal measurement across projects and the uniqueness of each project can be used as explanation for project results and for constructive criticism. As McGarry [McGarry, 2001] said, "You can't benchmark data if you haven't collected it." Data must be collected with a clear purpose and statistical analysis requirements in mind. Again, consistency and scoping of the benchmark research is critical to achieve both tactical and strategic success. Benchmarking against up-to-date data is also important [Maxwell, 2001] to keep up with the times.

2.4.1.5 Benchmark Result as Confirmation and Justification

Heires [Heires, 2001] found that performing benchmarking within an organization might not reveal any surprises, but it does provide quantifiable information for justification of the current state. The most revealing benchmarking information is that by performing stratification on that benchmark data it provides insight into various interest groups. Stratification is similar to Organization Unit (OU) separation. By selecting the various combination of OU, it is possible to derive potentially effective comparison such as project size, technology usage, geographical location, and others.

Bearing in mind that benchmark is not "enabler". "Enabler" helps in the implementation of benchmark practices [Camp, 1989]. Benchmark can be an indirect enabler when benchmarked against a known standard. A standard usually contains suggestion and recommendation to set or improve on process. That is why this thesis utilized well-known SEP standards and models as the foundation for assessment leading to the benchmarking process. As mentioned by the author, benchmark provides the direction and goal to reach but it is up to the organization to take action on the path and method to proceed along the journey of CPI.

2.4.2 Financial Focus

Benchmarking is like any other quality program or continuous improvement program; it will pay back in the long run. With continuous usage, benchmarking will become a natural process as part of the organization practice. Like those programs, benchmarking is initially expensive to implement and use. So, the question is how does benchmarking contribute to the financial of an organization and contribute to an organization's Return On Investment (ROI)? Benchmarking can help an organization better manage its cost by providing factual and correct justification for resource allocation and utilization. The ROI will hopefully increase based on the same amount of investment due to better SEP management and optimization.

It will be up to functional management to manage the financial aspect of benchmarking to justify its budget to upper senior management. This justification requires some form of measurements that can be translated to metric for budget justification. As Robert Camp [Camp, 1989] pointed out, the market place will prove the validity of the best practice business process emulation. Market place feedbacks can be measured by taking customer survey to evaluate customer satisfaction. The reason for the survey is that no two organizations are identical [Baetjer Jr., 1998]; hence, the organization's business process requires this final validation. This survey should be directly tied to a specific process that can be identified with calculated cost and scheduled implementation plans.

By knowing customer expectations and the process cost involved to meet those expectations will enable an organization to better estimate the cost of future work. During initial project bidding or work effort determination, it is common for an organization to perform estimation as to resource, cost and schedule required to complete the work.

Benchmarking can help confirm the estimation process is sound. Benchmarking usually involves multiple projects at a time, while estimation is usually concerned with one project at a time. This does not invalidate estimation [Card and Zubrow, 2001]. The organization can select the appropriate internal benchmark processes with appropriate cost association for project estimation.

2.4.3 Process Focus

Benchmarking provides business simplification. Business simplification is achieved by selecting and monitoring specific process adopted and practiced by the best in the industry. Hence, adopting these specific practices makes it cost effective for the organization by easing the needs to know how the competitors are doing through benchmarking. Knowing what others are doing and not doing can force one to evaluate the cost effectiveness of an organization's current process and practice. In Table 2-1, Robert Camp [Camp, 1989] illustrates how regular benchmarking can be extended.

Table 2-1. Uses of Benchmarking for Business Simplification

	Benchmarking	**Extended Benchmarking**
Objective	Efficiency	Effectiveness
Requirements Met	Internally Defined	End User Defined
Process	Current	Industry Standard
Practices & Technology	Best	Best
Cost of Nonconformance	Partially Reduced	Eliminated
Results	Productivity	Business Simplification

This thesis covers topics listed in the extended benchmarking column tabled above. Business simplification means the improvement of an organization's knowledge, practices, and process corresponding to best practices.

Benchmarking can help increase productivity by identifying practices or processes that make the most contribution to the capital structure. Economist defines "capital structure" as the structure of production [Baetjer Jr., 1998]. Benchmarking encapsulates and directs knowledge to a specific resource to increase productivity. Productivity is measured by decreasing the cost involved in production. The benchmark view of capital structure starts from front-line activities leading up to specific practiced processes for a given project within an organization. Essentially, benchmarking provides top-down decision making results through bottom-up assessment for an organization regarding their capital goods.

Capital is the embodied knowledge of productive processes and the processes are executed. This is the fundamental relation between knowledge and capital. From this viewpoint, benchmarking provides a means to channel the knowledge to the appropriate capital goods. Obviously, the different kind of knowledge means there are various kinds of embodiment [Baetjer Jr., 1998].

2.4.3.1 Industrial Benchmarking Implication in Software Engineering

The software industry is a dynamic industry. This means the stages of production are always in constant flux with hidden information. This is where continuous benchmarking of SEP is beneficial by providing an organization with the constant update of best practices in the industry. The process will be identified but the activities are not. Activities might contain potentially proprietary information. This means process is less susceptible to change with time. Activities are more susceptible to change with time. The change in activities will affect SEP capability level outcome. Hence, it is imperative to perform continuous benchmarking of the SEP to ensure improvement is achieved leading to superior performance in the dynamic software engineering industry.

One way to measure a benchmark process and generate benchmark metric has been researched by Yingxu Wang [Wang, 2001]. Benchmark results can be used as part of software engineering measurement and analysis. For example, Wang has compiled and developed a comprehensive Software Engineering Measurement System that includes process benchmarking. Some of the process benchmarks are listed below:

Benchmarks description:

- SEPRM process benchmarks ($Bench_{SEPRM}$ [#])
- Process benchmarks of nations ($Bench_{natn}$ [#])
- Process benchmarks of sectors ($Bench_{sect}$ [#])
- Process benchmarks of size of business ($Bench_{size}$ [#])
- The basic process level (CPL_{bas} [#])
- The competitive process level (CPL_{comp} [#])
- The advanced process level (CPL_{adv} [#])

Benchmark analysis

- Gaps against benchmarks (Δ_{bench} [#])
- Position against benchmarks (ω_{bench} [#])
- Position in a nation (ω_{natn} [#])
- Position in a sector (ω_{sector} [#])
- Position in a size group of business (ω_{size} [#])

2.4.4 Knowledge Focus

We all know that knowledge is prominent in software as in no other capital goods. Hence, it is critical for an organization to perform knowledge allocation and management well. One way that benchmarking can help in this area is to identify the specific practice and process to perform; hence, focusing on the knowledge to the relevant task. This means the human resource can focus the knowledge on only those identified processes through benchmarking. This concentration of focused knowledge will increase the productivity per person, therefore, leading to more generated goods for given effort. This essentially, increases the value of the capital goods generated by the given resource leading to gaining a competitive edge. As pointed out by Baetjer Jr. [Baetjer Jr., 1998], many centuries ago, it took 75% of the population to feed all of us. Now it only takes 3% to feed all of us. This is a clear indication of the way the future is to increase our knowledge by building knowledge into our tools. In this thesis, the tool technology is benchmarking. Benchmarking is the technology presented to optimize the effort in CPI to maximize the level of superior performance.

2.4.4.1 Benchmarking Enhances Knowledge Management

A radical point of view by Baetjer Jr. [Baetjer Jr., 1998] is that capital goods are knowledge. What this means is that a tool is a combination of knowledge and matter. Benchmarking is a knowledgeable technology. Software is the knowledge that is loaded onto a computer system that in turns causes the circuit to perform a more knowledgeable task for higher purpose. This radical view is that capital structure hierarchy just converts indestructible matter into higher advantageous forms.

We know that each process requires knowledge to execute it. The diversification of process is essentially division of knowledge. The benchmarking role helps consolidate divided knowledge/process back into superior performances hopefully leading to competitive advantage.

Thomas Sowell [Baetjer Jr., 1998] once observed that, "[T]he intellectual advantage of civilization … is not necessarily that each civilized man has more knowledge [than primitive haves], but that he requires far less". Like embodiment of knowledge in capital structure, benchmarking tries to reduce the overall effort to acquire knowledge on competitors' practices and focuses only on essential knowledge required within an organization. Essential knowledge could be reduced or increased through benchmarking, depending on the status of the gap analysis outcomes. In any case, the essential knowledge will be more focused and less generalized.

Capital must be maintained to ensure its value does not decay. Like benchmarking, CPI must be an integral part of an organization's practice to ensure its embodied knowledge does not fall behind competitors leading to decay in capital.

2.4.4.2 Capital Good Process Benchmark Ties to Knowledge Management

Fixed capital usually refers to tools such as languages, compilers and others. Working Capital usually refers to raw materials such as programmer resource, skills and others. This category fits the SEP benchmarking concept nicely, in that benchmarking focuses on working capital related items. SEP benchmarking does not focus on fixed capital because they usually represent the essence of the organization and difficult to change, but not impossible.

Modular programming is similar to division of knowledge in capital. The modules must be complementary to one another in use. Similar to SEP standards, for example, ISO 9001 standard requires that every department process must interface and flow between departments in a smooth, accountable and effective manner. A well managed process could imply a better knowledge management. The division of knowledge can be stretched and not necessarily dispersed. For example, the capital structure might be modularized from system level to subsystem level. The subsystem capital might be further lengthened by categorizing it and associated process identified. The process could be refined to better utilize the capital assets. A good capital allocation might mean better tool allocation at each stage and reduces the cost at the same time. This could imply better knowledge utilization by focusing the knowledge at the appropriate process. The process area being identified as an area to be worked on is also an area for discovering and creating knowledge. This means the resource planning must ensure that the process area consists of sufficiently the right knowledge to improve the process. In benchmarking terms, if identified processes have been specialized and optimized to execute effectively, surely the cost would be reduced at the various stages. This implies that cost is part of the process effectiveness determination.

If an organization is tries to improve upon all processes within an organization, it will stretch its resource and thereby thin out knowledge. Knowledge is a cost factor that could have an effect on the bottom line. Benchmark helps by selectively choosing the process to work on; hence, assisting in knowledge allocation that means cost is now better managed efficiently. Alternatively, extending the organization's capital structure can reduce an organization's cost. This means, for example, certain process area's activities can be analyzed to increase efficiency and effectiveness, or outsource if subcontracting makes fiscal sense. An organization's potential to gain a competitive edge in the market place is increased by reducing cost using benchmark.

In a fast changing world benchmarking plays an important role as an integral part of this process. Conventional methodologies have failed because they have focused on product or service completion, instead of process in production or providing service. CPI could lead to knowledge improvement, which will lead to organization maturity as an entity.

The software industry is like an ever-changing ecosystem. Peter Allen's work [Baetjer Jr., 1998] was quoted as saying; "... evolution does not lead to individuals with optimal behavior, but to diverse populations with the resulting ability to learn. The real world is not only about efficient performance but also the capacity to adapt. What is found is that variability at the microscopic level, individual diversity is part of evolutionarily strategy..." this seems to describe the same characteristics of benchmarking. The process of benchmarking must be an ongoing process as a quality tool for an organization to learn about their current state and how they compare to others. The benchmark result provides a channel or ability for an organization to learn and to adapt, emulating the best practice. Another important point to note from Peter Allen is it is not necessary for an organization to achieve efficient performance, but to ascertain the ability to adapt. Another note is that the microscopic activities will be different between organizations. It is the higher-level strategy, such a process, that an organization should be concerned about to stay ahead of the game. By achieving best practice essentially means one has achieved "optimal behavior". This is not necessarily a good thing if the organization decided to remain stagnant. The organization might not be staying with the game or getting ahead of the game to achieve superior performance. Stressing once again, the importance of survival is possessing the ability learn and adapt.

We can see that an organization's goals, finances, processes, and knowledge are highly inter-related. And, benchmark results have been shown as a supplemental technology to provide information to an organization to better manage its goals, finances, processes, and knowledge. An organization's business goal is usually closely tie to ensuring profitable financial outcome. An organization's goal can only be met through process execution by its knowledgeable assets, the employees. Process benchmarking helps by providing information on best practice. It is up to the organization to determine which best practices are suitable for its business. Best practice helps identify preferred process to practice and the maturity of the process helps the organization to better manage its knowledge assets. A well-allocated knowledge to a particular process for execution can hopefully help the organization meet its goal. Once the goal is met, it is up to the market place, the same market place that determines the best practice, to validate the result for financial success. Finally, the whole benchmarking process is repeated to ensure organization growth and maturity.

The state of the art of benchmarking has been explored and reviewed in this chapter. The core of benchmark analysis, namely gap analysis was also presented. The benchmark results were shown to be of value by correlating the results to an organization's goals, finances, processes and knowledge management needs. The most important thing to note in benchmarking is to appraise the current state's strength and weakness prior to benchmarking. Assessment generates appraisal results. Assessment is very important to benchmarking because it generates meta benchmarking measures. The output of assessment process is the input to the benchmarking process. Obviously, SEP benchmarking requires quantitative SEP assessment results. The next chapter will provide a comprehensive SEP-based model, which is used to overcome the difficulty of what process is to be assessed and how to perform the assessment.

CHAPTER 3 SEP Assessment

SEP assessment is a measurement process for determining what process is to be assessed and how to perform the assessment by using a standard algorithm. Since there are many SEP standards and models out in the market place the questions raised are which standards suit an organization best and how are multiple standard assessments performed and managed? It is difficult to perform multiple standards efficiently let alone trying to use them effectively. All is not lost. This thesis proposes the use of a comprehensive SEP Assessment Model to make the assessment process efficient and the results effective. This chapter will also present an assessment tool based on the assessment algorithms and its implementation.

3.1 Introduction

This chapter describes SEP assessment technologies and the precursor to the benchmarking methodology developed in Chapter 4. SEP assessment is an important precursor to benchmarking because it provides the process state information as input to benchmarking. The assessments used in this research were based on a unified SEP standards and models, Software Engineering Process Reference Model (SEPRM), which can be found in the book entitled, "Software Engineering Process: Principles and Applications [Wang and King, 2000].

The SEP assessment models used in this research are CMM, ISO 9001, BOOTSTRAP Model, ISO/IEC 15504 (SPICE) and SEPRM. SEPRM is a comprehensive and unified academic model. Table 3-1 provides the philosophies and background orientations of the SEP models [Wang and King, 2000] adopted in this thesis.

Table 3-1. Philosophies and Background Orientation of current SEP Models

Chronology	Model	Philosophy or Background Orientation
1987	CMM	To present a software project contractor's perception on the organizational and managerial capacity of a software development organization.
1991	ISO 9001	To present a generic quality system perception on software development.
1993	BOOTSTRAP	To present a combined view of software life cycle processes and quality system principles.
1998	ISO/IEC 15504 (SPICE)	To present a set of structured capability measurements for all software lifecycle processes, and for all parties such as software developers, acquirers, contractors and customers.
1998	SEPRM	To present a comprehensible and integrated process system reference model, with sound foundations and process benchmark support, for process-based software engineering.

In "Software Certification Debate: Benefits of Certification", Leonard Tripp [Tripp L., 2002] claims that certification will raise the benchmark for individual performance by achieving software quality and productivity. Otherwise, software engineering quality will suffer due to lack of understanding and awareness of accepted best practices. Tripp made a compelling argument in 1999, IEEE had over 40 software engineering standards totaling 2,400 pages in four volumes and the purpose for these standards are:

- Provide a vocabulary for communication between participants in the SEP
- Provide objective criteria for understanding claims regarding a product's nature
- Provide methods for specifying product characteristics
- Assure that quality assurance practices were applied

It is up to individual implementation of the standards in practice to improve software quality. Here certification of the practice is a means for improving the discipline by promoting the practical implementation of standards. This is important to promote awareness of a body of knowledge, recognize a code of ethics and realize the need for professional development. Furthermore, benchmarking these practices can benefit public at large. It helps in the refreshing of technical knowledge and provides opportunities to update skills. Certification is a standardized benchmark that the public can understand. That is why this work values what standards have to offer as the basis for assessment.

Each SEP standards and models sections will be presented. Next, the unification of all the SEP standards and models are mentioned to show how they were consolidated and mapped to the SEPRM. SEPRM provides a comprehensive and integrated structure and assessment methodology that the other four standards can be assessed with relative ease. Others have acknowledged the value of using multiple process standards [Aissi S., et al., 2002; Jung and Hunger, 2001; Thayer, 2002] but none have achieved the level of comprehensiveness as that of SEPRM.

SEP assessment process implementation in this thesis can be described as follow. New projects assessments data collections were told to use the raw rating scale, if possible. Existing projects with existing assessment standards ratings data were mapped to the raw rating scale. Once all assessment data were completed or collected they were stored in a relational database. A relational database was utilized because the comprehensive SEP model was mapping intensive. When the assessments needed to be computed, the raw ratings were converted to the rating scale of the appropriate SEP standards and models. The algorithm of each SEP standards and models were then applied accordingly and the results were stored back into the database. The assessment results consisting of process-level assessment results, including all intermediate results, up to the subsystems or category hierarchy level, depend on the SEP standards and models. These assessment results are the inputs to the benchmarking process.

3.2 The Capability Maturity Model (CMM)

The CMM was developed at the Software Institute (SEI) at Carnegie-Mellon University in 1987. This model is also commonly referred to as the SEI CMM. The model used in this research is SW-CMM Ver 1.1, where SW-CMM refers to Software CMM [Humphrey, 1990].

CMM process hierarchy and domain consists of five capability levels. Each capability level consists of 18 key process areas. CMM has identified 150 key practices within these 18 key process areas. CMM capability consists of five levels.

1 in Section 1.1 " Architecture" illustrates the data flow and relationship within the code implemented in the tool. An implementation of the CMM algorithm [Wang and King, 2000] is described below.

```
Sub CMM_Assessment(Project_ID)

    'Variables Declaration
       . . .

    '*** Get Recordset of the Project ***
    Extract All CMM related activities into a record set from the SEPRM BPA
       with the following SQL statement:

          SELECT BPA.Project_ID, BPA.BPA_Rating, CMM.BPA_L2, CMM.BPA_L3, CMM.BPA_L4
          FROM CMM LEFT JOIN BPA ON CMM.BPA_Number = BPA.BPA_Number
          WHERE (((BPA.Project_ID) = " + CStr(Proj_ID) + "))"
          ORDER BY CMM.BPA_L2, CMM.BPA_L3, CMM.BPA_L4

    '*** Initializing Variables ***
    PCL = 0
    CL_1 = CL_2 = CL_3 = CL_4 = CL_5 = 0
    KPA_1_1 = 0
    KPA_2_1 = KPA_2_2 = KPA_2_3 = KPA_2_4 = KPA_2_5 = KPA_2_6 = 0
       . . .
    KPA_5_1 = KPA_5_2 = KPA_5_3 = 0

    count = 0

    '*** Assign Recordset Data to Array ***
    Loop through the record set until end of record

        count = count + 1

        '*** Assessment Data Accummulation***

        '*** CL_1 ***
        KPA_1_1 = 0

        '*** CL_2 ***
        If (rst![BPA_L2] = 2) Then
            If (rst![BPA_Rating] > 2 Or rst![BPA_Rating] = 0) Then
                Select Case rst![BPA_L3]
                Case 1
                    KPA_2_1 = KPA_2_1 + 1
                Case 2
                    KPA_2_2 = KPA_2_2 + 1
                Case 3
                    KPA_2_3 = KPA_2_3 + 1
                Case 4
                    KPA_2_4 = KPA_2_4 + 1
                Case 5
                    KPA_2_5 = KPA_2_5 + 1
                Case 6
                    KPA_2_6 = KPA_2_6 + 1
                End Select
            End If
        End If
```

```
    '*** CL_3 ***
          . . .

    '*** CL_4 ***
          . . .

    '*** CL_5 ***
    If (rst![BPA_L2] = 5) Then
        If (rst![BPA_Rating] > 2 Or rst![BPA_Rating] = 0) Then
            Select Case rst![BPA_L3]
            Case 1
                KPA_5_1 = KPA_5_1 + 1
            Case 2
                KPA_5_2 = KPA_5_2 + 1
            Case 3
                KPA_5_3 = KPA_5_3 + 1
            End Select
        End If
    End If

Loop

'*** Calculate Assessment Result ***
CL_1 = KPA_1_1
CL_2 = KPA_2_1 + KPA_2_2 + KPA_2_3 + KPA_2_4 + KPA_2_5 + KPA_2_6
    . . .
CL_5 = KPA_5_1 + KPA_5_2 + KPA_5_3
PCL = 1
If (CL_2 >= 50) Then
    PCL = 2
    If (CL_3 >= 40) Then
        PCL = 3
        If (CL_4 >= 10) Then
            PCL = 4
            If (CL_5 >= 21) Then
                PCL = 5
            End If
        End If
    End If
End If

'*** Update Project Record ***
Update the CMM Assessment table with the variable being initialized earlier,
   using the SQL Update statement

End Sub
```

The CMM algorithm implementation consists of summation of the various key practices capability levels within their key practice areas. An average capability is then calculated for each respective key practice areas. The final CMM capability level is determined based on the accumulation of the average capability levels.

3.3 The ISO 9001 Model

ISO 9001 Model is a part of the ISO 9001 Suite. ISO 9000 is a set part of international standards for quality systems. ISO 9001 standard is commonly used in software development because the standard has its root in statistical quality control, Total Quality Management (TQM) and continuous improvement.

ISO 9001 process hierarchy and domain consists of three subsystems. These three subsystems consist of 20 main topic areas. ISO 9001 has identified 177 management issues within these 20 main topic areas.

ISO 9001 capability levels consist of two levels. A project is considered "Passed" when all management issues are satisfied, and a project is considered "Failed" if it did not satisfied one or more management issues.

in Section 1.1 " Architecture" illustrates the data flow and relationship within the code implemented in the tool. An implementation of the ISO 9001 algorithm [Wang and King, 2000] is described below.

```
Sub ISO9001_Assessment(Project_ID)

    'Variables Declaration
       . . .

    '*** Get Recordset of the Project ***
    Extract All ISO 9001 related activities into a record set from the SEPRM BPA
       with the following SQL statement:

    SELECT BPA.Project_ID, BPA.BPA_Rating, ISO9001.BPA_L1, ISO9001.BPA_L3, ISO9001.BPA_L4
           FROM ISO9001 LEFT JOIN BPA ON ISO9001.BPA_Number = BPA.BPA_Number
           WHERE (((BPA.Project_ID) = " + CStr(Proj_ID) + "))"
           ORDER BY ISO9001.BPA_L1, ISO9001.BPA_L3, ISO9001.BPA_L4

    '*** Initializing Variables ***
    PCL = 0
    SS_1 = SS_2 = SS_3 = 0
    MTA_1_1 = MTA_1_2 = MTA_1_3 = MTA_1_4 = MTA_1_5 = MTA_1_6 = MTA_1_7 = 0
    MTA_2_1 = MTA_2_2 = MTA_2_3 = MTA_2_4 = MTA_2_5 = 0
    MTA_3_1 = MTA_3_2 = MTA_3_3 = MTA_3_4 = MTA_3_5 = MTA_3_6 = MTA_3_7 = MTA_3_8 = 0

    count = 0

    '*** Assign Recordset Data to Array ***
    Loop through the record set until end of record

        count = count + 1
        '*** Assessment Data Accummulation***
        '*** SS_1 ***
        If (rst![BPA_L1] = 1) Then
            If (rst![BPA_Rating] > 2 Or rst![BPA_Rating] = 0) Then
                Select Case rst![BPA_L3]
                Case 1
                    MTA_1_1 = MTA_1_1 + 1
                Case 2
                    MTA_1_2 = MTA_1_2 + 1
                Case 3
```

```
                MTA_1_3 = MTA_1_3 + 1
            Case 4
                MTA_1_4 = MTA_1_4 + 1
            Case 5
                MTA_1_5 = MTA_1_5 + 1
            Case 6
                MTA_1_6 = MTA_1_6 + 1
            Case 7
                MTA_1_7 = MTA_1_7 + 1
            End Select
        End If
    End If

    '*** SS_2 ***
    If (rst![BPA_L1] = 2) Then
        If (rst![BPA_Rating] > 2 Or rst![BPA_Rating] = 0) Then
            Select Case rst![BPA_L3]
            Case 1
                MTA_2_1 = MTA_2_1 + 1
            Case 2
                MTA_2_2 = MTA_2_2 + 1
            Case 3
                MTA_2_3 = MTA_2_3 + 1
            Case 4
                MTA_2_4 = MTA_2_4 + 1
            Case 5
                MTA_2_5 = MTA_2_5 + 1
            End Select
        End If
    End If

    '*** SS_3 ***
    If (rst![BPA_L1] = 3) Then
        If (rst![BPA_Rating] > 2 Or rst![BPA_Rating] = 0) Then
            Select Case rst![BPA_L3]
            Case 1
                MTA_3_1 = MTA_3_1 + 1
            Case 2
                MTA_3_2 = MTA_3_2 + 1
            Case 3
                MTA_3_3 = MTA_3_3 + 1
            Case 4
                MTA_3_4 = MTA_3_4 + 1
            Case 5
                MTA_3_5 = MTA_3_5 + 1
            Case 6
                MTA_3_6 = MTA_3_6 + 1
            Case 7
                MTA_3_7 = MTA_3_7 + 1
            Case 8
                MTA_3_8 = MTA_3_8 + 1
            End Select
        End If
    End If

Loop

'*** Calculate Assessment Result ***
SS_1 = MTA_1_1 + MTA_1_2 + MTA_1_3 + MTA_1_4 + MTA_1_5 + MTA_1_6 + MTA_1_7
SS_2 = MTA_2_1 + MTA_2_2 + MTA_2_3 + MTA_2_4 + MTA_2_5
SS_3 = MTA_3_1 + MTA_3_2 + MTA_3_3 + MTA_3_4 + MTA_3_5 + MTA_3_6 + MTA_3_7 + MTA_3_8
PCL = SS_1 + SS_2 + SS_3

'*** Update Project Record ***
Update the ISO 9001 Assessment table with the variable being initialized earlier,
   using the SQL Update statement


End Sub
```

The ISO9001 algorithm implementation consists of summation of the various management issues that are satisfied within a given main topic areas. An ISO 9001 standard conformity satisfaction is achieved only when all management issues are satisfied for all the main topic areas.

3.4 The BOOTSTRAP Model

BOOTSTRAP is a European process methodology for SEP system assessment and improvement [Wang and King, 2000]. This BOOTSTRAP Assessment Model is BOOTSTRAP version 2.3.

BOOTSTRAP process hierarchy and domain consist of three process areas. These five process areas consist of nine process categories. Within the nine process categories, there are 32 processes. BOOTSTRAP has identified 201 quality system attributes within these 32 processes. BOOTSTRAP capability consists of five levels.

Figure 4-10 in Section 1.1 " Architecture" illustrates the data flow and relationship within the code implemented in the tool. An implementation of the BOOTSTRAP algorithm [Wang and King, 2000] is described below.

```
Sub Bootstrap_Assessment(Project_ID)

    'Variables Declaration
       . . .

    '*** Get Recordset of the Project ***
    Extract All Bootstrap related activities into a record set from the SEPRM BPA
       with the following SQL statement:

    SELECT BPA.Project_ID, BPA.BPA_Rating, Bootstrap.BPA_L5
           FROM Bootstrap LEFT JOIN BPA ON Bootstrap.BPA_Number = BPA.BPA_Number
           WHERE (((BPA.Project_ID) = " + CStr(Proj_ID) + "))"
           ORDER BY Bootstrap.BPA_L5

    '*** Initializing Variables ***
    PCL = 0
    CL_1 = CL_2 = CL_3 = CL_4 = CL_5 = 0

    count = 0

    '*** Assign Recordset Data to Array ***
    Loop through the record set until end of record

        count = count + 1
```

```
    '*** Assessment Data Accummulation***

    If (rst![BPA_Rating] > 2 Or rst![BPA_Rating] = 0) Then
        Select Case rst![BPA_L5]
            Case 1 '*** CL_1 ***
                CL_1 = CL_1 + 1

            Case 2 '*** CL_2 ***
                CL_2 = CL_2 + 1

            Case 3 '*** CL_3 ***
                CL_3 = CL_3 + 1

            Case 4 '*** CL_4 ***
                CL_4 = CL_4 + 1

            Case 5 '*** CL_5 ***
                CL_5 = CL_5 + 1
        End Select
    End If

Loop

'*** Calculate Assessment Result ***
CL_1 = 0
AC_1 = 0
AC_2 = CL_2
AC_3 = AC_2 + CL_3
AC_4 = AC_3 + CL_4
AC_5 = AC_4 + CL_5
PCL = 1
If (AC_2 >= 32) Then
    PCL = 2
    If (AC_3 >= 97) Then
        PCL = 3
        If (AC_4 >= 119) Then
            PCL = 4
            If (AC_5 >= 162) Then
                PCL = 5
            End If
        End If
    End If
End If

'*** Calculate Fraction ***
RAT = 0
If (PCL = 1) Then
    RAT = AC_5 - AC_1
    RAT = RAT / (53 + 27 + 81 + 40)
End If

If (PCL = 2) Then
    RAT = AC_5 - AC_2
    RAT = RAT / (53 + 27 + 81)
End If

If (PCL = 3) Then
    RAT = AC_5 - AC_3
    RAT = RAT / (53 + 27)
End If

If (PCL = 4) Then
    RAT = AC_5 - AC_4
    RAT = RAT / (53)
End If

If (PCL = 5) Then
    RAT = 0
End If
```

```
'*** Rounding down a quarter (1/4) ***
If (RAT < 0.25) Then
    RAT = 0
End If

If (RAT >= 0.25 And RAT < 0.5) Then
    RAT = 0.25
End If

If (RAT >= 0.5 And RAT < 0.75) Then
    RAT = 0.5
End If

If (RAT >= 0.75 And RAT <= 1) Then
    RAT = 0.75
End If

'*** Update Project Record ***
Update the Bootstrap Assessment table with the variable being initialized earlier,
   using the SQL Update statement

End Sub
```

The BOOTSTRAP algorithm implementation consists of summation of the various quality system attributes that are satisfied within a given process. These values are then rolled up to the various process areas. The final BOOTSTRAP capability level is determined by adding capability level to the rating value. The final rating value is determined by rounding it down to the nearest quarter.

3.5 The ISO/IEC 15504 (SPICE) Model

This is a new international standard developed by the international project known as Software Process Improvement and Capability dEtermination (SPICE).

ISO/IEC 15504 process hierarchy and domain consists of five process categories. These process categories in turn consist of 35 processes. ISO/IEC 15504 has identified 201 base practices within these 35 processes. ISO/IEC 15504 capability consists of six levels. Even though ISO/IEC 15504 has background in CMM, it does not follow the CMM level description. ISO/IEC 15504 rating scale consists of four levels. ISO/IEC 15504 is the first standard to implement process capability level assessment independently of the project capability maturity level assessment.

in Section 1.1 " Architecture" illustrates the data flow and relationship within the code implemented in the tool. An implementation of the ISO/IEC 15504 algorithm [Wang and King, 2000] is described below.

```
Sub ISO15504_Assessment(Project_ID)

    'Variables Declaration
       . . .

    '*** Get Recordset of the Project ***
    Extract All ISO/IEC TR 15504 related activities into a record set from the SEPRM BPA
       with the following SQL statement:

    SELECT BPA.Project_ID, BPA.BPA_Rating, ISO15504.BPA_L2, ISO15504.BPA_L3,
              ISO15504.BPA_L4
           FROM ISO15504 LEFT JOIN BPA ON ISO15504.BPA_Number = BPA.BPA_Number
           WHERE (((BPA.Project_ID) = " + CStr(Proj_ID) + "))"
           ORDER BY ISO15504.BPA_L2, ISO15504.BPA_L3, ISO15504.BPA_L4


    '*** Initializing Variables ***

    For cnt_bp = 1 To 15
        PA_1_cnt(cnt_bp) = PA_2_cnt(cnt_bp) = PA_3_cnt(cnt_bp) = 0
        PA_4_cnt(cnt_bp) = PA_5_cnt(cnt_bp) = 0
        For cnt_attr = 1 To 5
            PA_1_macro(cnt_bp, cnt_attr) = 0 ' CUS
            PA_2_macro(cnt_bp, cnt_attr) = 0 ' ENG
            PA_3_macro(cnt_bp, cnt_attr) = 0 ' PRO
            PA_4_macro(cnt_bp, cnt_attr) = 0 ' SUP
            PA_5_macro(cnt_bp, cnt_attr) = 0 ' ORG

             ' Set BP Rating Average to zero
            PA_1_avg(cnt_bp, cnt_attr) = PA_2_avg(cnt_bp, cnt_attr) = 1
            PA_3_avg(cnt_bp, cnt_attr) = PA_4_avg(cnt_bp, cnt_attr) = 1
            PA_5_avg(cnt_bp, cnt_attr) = 1
         Next
    Next

    PCL = 0
    PA_1_1 = PA_1_2 = PA_1_3 = PA_1_4 = PA_1_5 = PA_1_6 = PA_1_7 = PA_1_8 = 0

       . . .

    PA_5_1 = PA_5_2 = PA_5_3 = PA_5_4 = PA_5_5 = PA_5_6 = PA_5_7 = 0

    count = 0

    '*** Assign Recordset Data to Array ***
    Loop through the record set until end of record

        count = count + 1

        '*** Assessment Data Accummulation***

        '*** 1 CUS ***
        If (rst![BPA_L2] = 1) Then
            If (rst![BPA_Rating] > 0) Then
                rating = rst![BPA_Rating]
                PA_1_cnt(rst![BPA_L3]) = PA_1_cnt(rst![BPA_L3]) + 1
                PA_1_macro(rst![BPA_L3], 5) = IIf(rating = 4, 4, IIf(rating = 3, 2,
                                             IIf(rating = 2, 1, IIf(rating = 1, 1, 0))))
                PA_1_macro(rst![BPA_L3], 4) = IIf(rating = 4, 4, IIf(rating = 3, 3,
                                             IIf(rating = 2, 2, IIf(rating = 1, 1, 0))))
                PA_1_macro(rst![BPA_L3], 3) = IIf(rating = 4, 4, IIf(rating = 3, 4,
                                             IIf(rating = 2, 3, IIf(rating = 1, 1, 0))))
                PA_1_macro(rst![BPA_L3], 2) = IIf(rating = 4, 4, IIf(rating = 3, 4,
```

```
                                        IIf(rating = 2, 4, IIf(rating = 1, 1, 0))))
            PA_1_macro(rst![BPA_L3], 1) = IIf(rating = 4, 4, IIf(rating = 3, 4,
                                        IIf(rating = 2, 4, IIf(rating = 1, 1, 0))))
            For cnt = 1 To 5
                PA_1_avg(rst![BPA_L3], cnt) = ((PA_1_avg(rst![BPA_L3], cnt) *
                                                (PA_1_cnt(rst![BPA_L3]) - 1)) +
                                                PA_1_macro(rst![BPA_L3], cnt)) /
                                                PA_1_cnt(rst![BPA_L3])
            Next
        End If
    End If

    . . .

    '*** 5 ORG ***
    If (rst![BPA_L2] = 5) Then
        If (rst![BPA_Rating] > 0) Then
            rating = rst![BPA_Rating]
            PA_5_cnt(rst![BPA_L3]) = PA_5_cnt(rst![BPA_L3]) + 1
            PA_5_macro(rst![BPA_L3], 5) = IIf(rating = 4, 4, IIf(rating = 3, 2,
                                        IIf(rating = 2, 1,IIf(rating = 1, 1, 0))))
            PA_5_macro(rst![BPA_L3], 4) = IIf(rating = 4, 4, IIf(rating = 3, 3,
                                        IIf(rating = 2, 2, IIf(rating = 1, 1, 0))))
            PA_5_macro(rst![BPA_L3], 3) = IIf(rating = 4, 4, IIf(rating = 3, 4,
                                        IIf(rating = 2, 3, IIf(rating = 1, 1, 0))))
            PA_5_macro(rst![BPA_L3], 2) = IIf(rating = 4, 4, IIf(rating = 3, 4,
                                        IIf(rating = 2, 4, IIf(rating = 1, 1, 0))))
            PA_5_macro(rst![BPA_L3], 1) = IIf(rating = 4, 4,IIf(rating = 3, 4,
                                        IIf(rating = 2, 4, IIf(rating = 1, 1, 0))))
            For cnt = 1 To 5
                PA_5_avg(rst![BPA_L3], cnt) = ((PA_5_avg(rst![BPA_L3], cnt) *
                                        (PA_5_cnt(rst![BPA_L3]) - 1)) +
                                        PA_5_macro(rst![BPA_L3], cnt)) /
                                        PA_5_cnt(rst![BPA_L3])
            Next
        End If
    End If

Loop

'*** Round value down prior to calling function ***
' Otherwise the function parameters will be rounded to nearest integer prior
   to passing to function call
For cnt_bp = 1 To 15
    For cnt_attr = 1 To 5
        PA_1_avg(cnt_bp, cnt_attr) = Int(PA_1_avg(cnt_bp, cnt_attr))
        PA_2_avg(cnt_bp, cnt_attr) = Int(PA_2_avg(cnt_bp, cnt_attr))
        PA_3_avg(cnt_bp, cnt_attr) = Int(PA_3_avg(cnt_bp, cnt_attr))
        PA_4_avg(cnt_bp, cnt_attr) = Int(PA_4_avg(cnt_bp, cnt_attr))
        PA_5_avg(cnt_bp, cnt_attr) = Int(PA_5_avg(cnt_bp, cnt_attr))
    Next
Next

'*** Calculate Assessment Result ***
' CUS
PA_1_1 = ISO15504_Rating_Calc(PA_1_avg(1, 5), PA_1_avg(1, 4), PA_1_avg(1, 3),
           PA_1_avg(1, 2), PA_1_avg(1, 1))
PA_1_2 = ISO15504_Rating_Calc(PA_1_avg(2, 5), PA_1_avg(2, 4), PA_1_avg(2, 3),
           PA_1_avg(2, 2), PA_1_avg(2, 1))
PA_1_3 = ISO15504_Rating_Calc(PA_1_avg(3, 5), PA_1_avg(3, 4), PA_1_avg(3, 3),
           PA_1_avg(3, 2), PA_1_avg(3, 1))
PA_1_4 = ISO15504_Rating_Calc(PA_1_avg(4, 5), PA_1_avg(4, 4), PA_1_avg(4, 3),
           PA_1_avg(4, 2), PA_1_avg(4, 1))
PA_1_5 = ISO15504_Rating_Calc(PA_1_avg(5, 5), PA_1_avg(5, 4), PA_1_avg(5, 3),
           PA_1_avg(5, 2), PA_1_avg(5, 1))
PA_1_6 = ISO15504_Rating_Calc(PA_1_avg(6, 5), PA_1_avg(6, 4), PA_1_avg(6, 3),
           PA_1_avg(6, 2), PA_1_avg(6, 1))
PA_1_7 = ISO15504_Rating_Calc(PA_1_avg(7, 5), PA_1_avg(7, 4), PA_1_avg(7, 3),
           PA_1_avg(7, 2), PA_1_avg(7, 1))
PA_1_8 = ISO15504_Rating_Calc(PA_1_avg(8, 5), PA_1_avg(8, 4), PA_1_avg(8, 3),
```

```
                PA_1_avg(8, 2), PA_1_avg(8, 1))

        . . .

    ' ORG
    PA_5_1 = ISO15504_Rating_Calc(PA_5_avg(1, 5), PA_5_avg(1, 4), PA_5_avg(1, 3),
                PA_5_avg(1, 2), PA_5_avg(1, 1))
    PA_5_2 = ISO15504_Rating_Calc(PA_5_avg(2, 5), PA_5_avg(2, 4), PA_5_avg(2, 3),
                PA_5_avg(2, 2), PA_5_avg(2, 1))
    PA_5_3 = ISO15504_Rating_Calc(PA_5_avg(3, 5), PA_5_avg(3, 4), PA_5_avg(3, 3),
                PA_5_avg(3, 2), PA_5_avg(3, 1))
    PA_5_4 = ISO15504_Rating_Calc(PA_5_avg(4, 5), PA_5_avg(4, 4), PA_5_avg(4, 3),
                PA_5_avg(4, 2), PA_5_avg(4, 1))
    PA_5_5 = ISO15504_Rating_Calc(PA_5_avg(5, 5), PA_5_avg(5, 4), PA_5_avg(5, 3),
                PA_5_avg(5, 2), PA_5_avg(5, 1))
    PA_5_6 = ISO15504_Rating_Calc(PA_5_avg(6, 5), PA_5_avg(6, 4), PA_5_avg(6, 3),
                PA_5_avg(6, 2), PA_5_avg(6, 1))
    PA_5_7 = ISO15504_Rating_Calc(PA_5_avg(7, 5), PA_5_avg(7, 4), PA_5_avg(7, 3),
                PA_5_avg(7, 2), PA_5_avg(7, 1))

    PCL = 0
    PCL = SUM(PA_1_1 + PA_1_2 + . . . + PA_5_6 + PA_5_7 )
    PCL = Int(PCL / 35)


    '*** Update Project Record ***
    Update the CMM Assessment table with the variable being initialized earlier,
       using the SQL Update statement

End Sub

Function ISO15504_Rating_Calc(r5, r4, r3, r2, r1)

    If (r1 = 4 And r2 = 4 And r3 = 4 And r4 = 4 And r5 >= 3) Then
        ISO15504_Rating_Calc = 5
    ElseIf (r1 = 4 And r2 = 4 And r3 = 4 And r4 >= 3) Then
        ISO15504_Rating_Calc = 4
    ElseIf (r1 = 4 And r2 = 4 And r3 >= 3) Then
        ISO15504_Rating_Calc = 3
    ElseIf (r1 = 4 And r2 >= 3) Then
        ISO15504_Rating_Calc = 2
    ElseIf (r1 >= 3) Then
        ISO15504_Rating_Calc = 1
    Else
        ISO15504_Rating_Calc = 0
    End If

End Function
```

The ISO/IEC 15504 algorithm is the most complex algorithm to implement in comparison with the other models algorithms. The ISO/IEC 15504 algorithm requires mapping between base practices rating and base practices attribute ratings prior to performing any summing and averaging calculations. After some integer rounding is performed, then the ISO/IEC 15504 capability level for each process can be determined based on the process attributes ratings achieved.

3.6 The Software Engineering Process Reference Model (SEPRM)

SEPRM is a comprehensive and integrated SEP model developed in [Wang and King, 2000]. This model is an academic model that encompasses all the other aforementioned standards and models.

SEPRM process hierarchy and domain consists of three process subsystems. These subsystems in turn consist of 12 process categories that include 51 processes. SEPRM has identified 444 base process activities (BPAs) within these 51 processes.

SEPRM capability consists of five levels. SEPRM rating scale consists of four levels. Its scale value is very similar to ISO/IEC 15504 but the rating threshold is different. Obviously the assessment algorithm is different too. Base on the algorithm complexity, SEPRM is a lot easier to assess than ISO/IEC 15504, even though SEPRM is comprehensive.

in Section 1.1 " Architecture" illustrates the data flow and relationship within the code implemented in the tool. An implementation of the SEPRM algorithm [Wang and King, 2000] is described below.

```
Sub SEPRM_Assessment(Project_ID)

    'Variables Declaration
        . . .

    '*** Get Recordset of the Project ***
    Extract All SEPRM related activities into a record set from the SEPRM BPA
       with the following SQL statement:

    SELECT BPA.Project_ID, BPA.BPA_Rating, SEPRM.BPA_L1, SEPRM.BPA_L2,
               SEPRM.BPA_L3, SEPRM.BPA_L4
           FROM SEPRM LEFT JOIN BPA ON SEPRM.BPA_Number = BPA.BPA_Number
           WHERE (((BPA.Project_ID) = " + CStr(Proj_ID) + "))"
           ORDER BY SEPRM.BPA_L1, SEPRM.BPA_L2, SEPRM.BPA_L3, SEPRM.BPA_L4

    '*** Initializing Variables ***
    PCL = 0
    ID_1_1_1 = 0
    ID_1_1_2 = 0
    ID_1_2_1 = 0
    ID_1_2_2 = 0

        . . .

    CT_3_5_1 = 0
    CT_3_5_2 = 0
    CT_3_6_1 = 0
```

```
CT_3_6_2 = 0

count = 0

'*** Assign Recordset Data to Array ***
Loop through the record set until end of record

    count = count + 1

    '*** Assessment Data Accummulation***

    '*** 1.1 ***
    If (rst![BPA_L1] = 1 And rst![BPA_L2] = 1) Then
        If (rst![BPA_L4] > 0) Then
            t_Rating = IIf(rst![BPA_Rating] = 4, 5, IIf(rst![BPA_Rating] = 3, 3,
                   IIf(rst![BPA_Rating] = 2, 1, IIf(rst![BPA_Rating] = 1, 0, 5))))
            Select Case rst![BPA_L3]
            Case 1
                ID_1_1_1 = ID_1_1_1 + t_Rating
                CT_1_1_1 = CT_1_1_1 + 1
            Case 2
                ID_1_1_2 = ID_1_1_2 + t_Rating
                CT_1_1_2 = CT_1_1_2 + 1
            End Select
        End If
    End If

    . . .

    '*** 3.6 ***
    If (rst![BPA_L1] = 3 And rst![BPA_L2] = 6) Then
        If (rst![BPA_L4] > 0) Then
            t_Rating = IIf(rst![BPA_Rating] = 4, 5, IIf(rst![BPA_Rating] = 3, 3,
                   IIf(rst![BPA_Rating] = 2, 1, IIf(rst![BPA_Rating] = 1, 0, 5))))
            Select Case rst![BPA_L3]
            Case 1
                ID_3_6_1 = ID_3_6_1 + t_Rating
                CT_3_6_1 = CT_3_6_1 + 1
            Case 2
                ID_3_6_2 = ID_3_6_2 + t_Rating
                CT_3_6_2 = CT_3_6_2 + 1
            End Select
        End If
    End If

Loop

'*** Prevent Divide by zero ***
If (CT_1_1_1 < 1) Then CT_1_1_1 = 1
If (CT_1_1_2 < 1) Then CT_1_1_2 = 1

If (CT_1_2_1 < 1) Then CT_1_2_1 = 1
If (CT_1_2_2 < 1) Then CT_1_2_2 = 1

    . . .

If (CT_3_5_1 < 1) Then CT_3_5_1 = 1
If (CT_3_5_2 < 1) Then CT_3_5_2 = 1

If (CT_3_6_1 < 1) Then CT_3_6_1 = 1
If (CT_3_6_2 < 1) Then CT_3_6_2 = 1

'*** Calculate Assessment Result ***
ID_1_1_1 = Round((ID_1_1_1 / CT_1_1_1), 1)
ID_1_1_2 = Round((ID_1_1_2 / CT_1_1_2), 1)

ID_1_2_1 = Round((ID_1_2_1 / CT_1_2_1), 1)
ID_1_2_2 = Round((ID_1_2_2 / CT_1_2_2), 1)

    . . .
```

```
    ID_3_5_1 = Round((ID_3_5_1 / CT_3_5_1), 1)
    ID_3_5_2 = Round((ID_3_5_2 / CT_3_5_2), 1)

    ID_3_6_1 = Round((ID_3_6_1 / CT_3_6_1), 1)
    ID_3_6_2 = Round((ID_3_6_2 / CT_3_6_2), 1)

    PCL = 0
    PCL = PCL + SUM(ID_1_1_1 + ID_1_1_2 + . . . + ID_3_6_1 + ID_3_6_2)

    PCL = Round((PCL / 51), 1)

    '*** Update Project Record ***
    Update the CMM Assessment table with the variable being initialized earlier,
       using the SQL Update statement

End Sub
```

The SEPRM algorithm is clear-cut to implement in comparison with the other models algorithms, considering the advancement in the model. SEPRM employs relatively straightforward averaging calculation and capability level value rounding in its algorithm. The SEPRM algorithm can produce process capability level and project maturity level quickly in an efficient manner.

The models are unified and mapped to the comprehensive and integrated SEPRM Base Process Activities (BPAs). Table 3-2 shows the raw rating scale that was used by all the algorithms mentioned in the previous sections.

Table 3-2. Raw Rating Scale

Raw Scale	Description	Rating threshold
4	Fully adequate	75% - 100%
3	Largely adequate	50% - 74%
2	Partially adequate	25% - 49%
1	Not adequate	0% - 24%
0	Does not apply	-

The process mapping between all the standards and models and the normalized raw rating scale, enabled the readers to easily determine process capability levels. This determination derived the system level capability levels for all the standards and models. In addition, project's capability levels are also determined as part of the procedure by the algorithm for the different respective standards and models.

CHAPTER 4 SEP BENCHMARKING

The previous chapter has provided the fundamentals of SEP assessment and described the design of the SEP assessment tool. In this chapter, assessment results will be used as input data for benchmarking. Essentially, the assessment results provide information such as a project's current capability level. The benchmarking tool then uses the capability maturity level information to be compared to other projects to determine how it ranks among others and to determine the capability maturity gap magnitude. This chapter deals with the comparison process and analysis known as benchmarking.

Most software engineering standards committees believe the standards they promoted represent industry best practices [Volcker, et al, 2001]. That is why it is common to find benchmark results that reflect only one standard or model. It is becoming common to find organizations with multiple standards certifications or registrations. ISO 9001 is the most recognized standard and widely benchmarked along with other standards [Jung and Hunter, 2001] for a multi-standards organization. Trying to be compliant to multiple standards is challenging to manage; not to compound the hardship in trying to practice CPI. The new SEP benchmarking methodology presented here will hopefully make the task of compliance to multiple standards more manageable and value added. By the end of the day, hopefully, this new SEP benchmarking methodology may assist organizations to add value to an organization's SEP quality management effort.

4.1 Introduction

The SEP benchmarking methodology developed in this thesis is essentially an extension to the benchmark based SPI approach. The original concept of benchmark-based SPI using SEPRM was first introduced in [Wang and King, 2000; Wang, 2001a]. This thesis expands on the work and original concept.

This thesis defines SEP benchmarking as follows:

> *SEP benchmarking is an integral part of an organization's systematic CPI effort to achieve superior performance in the market place by measuring and analyzing its process against a given benchmark.*

The definition points out some important factors to achieving successful, efficient and effective benchmarking. Benchmarking must be an integral part of an organization's CPI effort. CPI is a long-term commitment effort and requires an organization's upper management support and buy-in in order for the benchmarking practitioner to do its job properly. The process must be systematic which means the benchmarking process must be executed in a logical manner with specific purpose in mind to achieve an organization's goals and benchmarking objectives. In order to achieve superior performance this implies action must be taken based on the benchmark results. The organization must always attempt to exceed the benchmark threshold to ensure its capability and maturity levels are constantly increasing and leading in the market place, as opposed to decaying.

4.2 Recapping Benchmarking Challenges

Let us begin by recapping some of the challenges to conventional SEP benchmarking as presented in the first two chapters. The goal is to present a new benchmark analysis technique and technology based on the proposed SEP benchmark methodology as a potential solution to take on the challenges.

SEP benchmarking problems can be broadly divided into two parts. The first part has to do with satisfying an organization's expectation from the SEP benchmarking effort. The second part has to deal with the current state of benchmarking technology such as making it more consistent, robust, verbose and acceptable. Consistent means the benchmark must be able to stand the test of time by reusing historical benchmark result. It must be robust in that it can be easily modified and extended, and reusable by existing benchmark. Verbose implies that it must be comprehensive and fully integrated in the information it provides. Acceptability is the most important factor, because a benchmark repository or technology that is not recognized by customers or organizations cannot be validated and rendered useless.

This chapter attempts to tackle these challenges as identified from literature review in Chapters 1 and 2. We will begin by taking a look at organization related problems that could potentially be aided by the newly proposed comprehensive standards and models based SEP benchmarking methodology.

An organization has two types of concern: external and internal. The former deals with competitors and the market place. The latter are usually related to processes at the various levels of an organization's projects and activities. An organization's external goal is to out perform the competitors and be accepted as the best in the market place. An organization's internal process has to be up to par to ensure the organization's external goal is achieved.

Externally, the market place is constantly demanding higher quality performance by encouraging organizations to adhere to many standards and capability maturity levels. By forcing organizations to adhere to standards is how the market place determines organization's competency and capability levels. The market place evaluates the process an organization is practicing and how well it executes its process. How an organization can manage multiple standards compliances and yet be the best? This is going to be costly and difficult, especially with the market trend favoring fixed-priced projects. Benchmarking is one possible solution. Where can we find a benchmarking organization or data bank that will meet the organization's need? Even if one is found, how can benchmarking achieve its external and internal goals?

An Internal concerns with in-house project benchmark results in contributing to the realization of organization's goal. What kind of valuable information can the benchmark gap provides? Is there any technique or technology that will make the SEP benchmarking effort efficient and effective?

The benchmarking effort usually flows down to frontline personnel, resting on the shoulders of the Quality Assurance (QA) department. QA should be concerned with the availability of benchmarking technology that can be of use to them as practitioners and value added to management. It would be nice if the benchmarking technology can be applied to multiple standards and models. Questions that QA would raise are as follows:

- Can benchmarking provide coherent comparison?
- How can overhead cost of QA be reduced or better managed while maintaining current information value?
- How can the SEP benchmarking be implemented as a CPI activity in an efficient and effective manner?

4.3 The Benchmarking Methodology for SEP

This section will propose a comprehensive SEP benchmarking methodology as a starting-point solution to the challenges identified in section 4.2, while being efficient in execution and effective in evaluation. This section first recommends the use of SEP standards and models for SEP benchmarking. Secondly, the standards and models must have an appraisal or assessment component to it. The third recommendation is to use a reference model for benchmarking. Finally, a new method of gap analysis technology will be presented in details as part of this thesis to support the proposed recommendations.

4.3.1 SEP Standards and Models

What better process to emulate than a standardized process that has already been deemed acceptable by the customer? Standardized process provides a common playing field for all to compete and focus. An organization is better off focusing on an objective standardized process as oppose to subjective competitors. Standards also provide organizations with consistency and acceptance of the process being practiced. Some of the benefits of using process standard with appraisal for benchmarking are as follows:

- Standards usually have historical credibility and can be used as a reference to best practice guide.
- Standards usually have been accepted and recognized by the customer and it is up organization's approach to implementation that will determine business effectiveness.
- Using standards or models that consist of an assessment component can provide organizations with process capability and maturity levels. Consistent assessment will provide a more objective benchmarking evaluation and judgment.
- Using the standards can ensure all organizations will be assessed similarly against the same reference point and judgment.
- Capability level requirements can serve as guidelines and suggestions for process improvement to heighten level of capability and maturity.
- Standards also serve as common ground for organizations to function accordingly and for customers to evaluate and judge superior performance and competency.

4.3.2 Appraisal and Assessment

Appraisal and assessment components of standards or models may provide suggestions and/or guidelines for improvement to achieve a higher level of capability maturity level. Since the assessment is standard based, this means that most organization will be appraised similarly as described in chapter 3.

The case studies in the next chapter propose benchmarking an organization's documented procedure as an ideal project. This internal documented procedure benchmark result will act as an ideal internal process model for all in-house projects to follow or to exceed. An organization having documented a procedure is already on its way up the capability maturity benchmark measuring stick. The internal documented procedure is usually an organization's experience of what the best practice should be for the business. This benchmark will act as an interface between internal and external benchmarking results, also making the whole benchmarking effort easier to manage. Figure 4-1 shows a conventional performance analysis would require 'n*(n-1)' number of benchmarking processes.

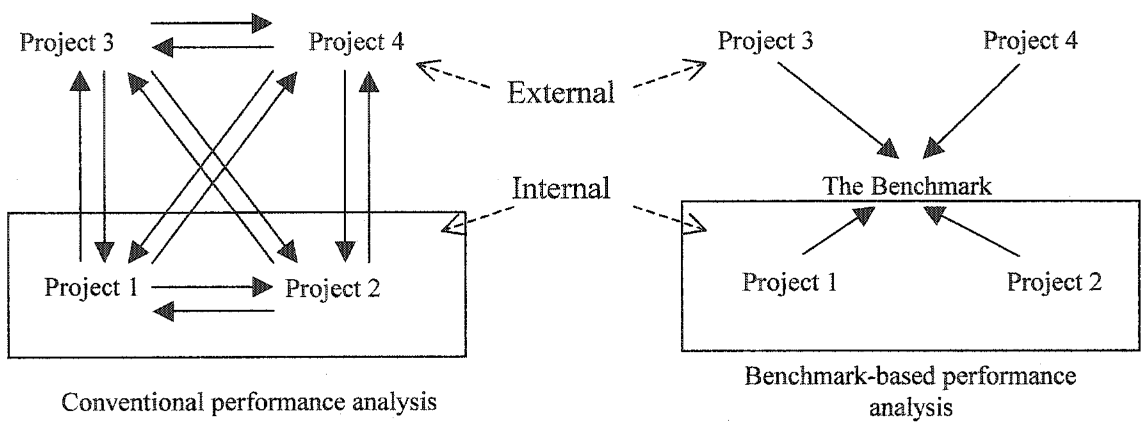

Figure 4-1 The role of benchmarks in SEP performance analysis

The Benchmarked-based performance analysis would require 'n' number of benchmarking processes. It has an advantage of nicely segregating internal and external projects by using the "The Benchmark" as an interface. Upper management can use the "The Benchmark" benchmark for strategic planning, and internal projects can use the benchmark for tactical planning.

4.3.3 Benchmarking Based on the SEPRM

A comprehensive reference model that can provide mapping to other standards and models is beneficial on time and cost saving. Assessment can be performed once and mapped to other standards and models. Less time and resource spent on assessment means less cost and more saving. In addition, the saving in time can provide faster benchmarking process turnaround time, resulting in early opportunity to be the best in the market place. A comprehensive reference model provides an organization with robustness and verboseness of SEP management in an efficient manner. Figure 4-2 and Figure 4-3 show the conventional benchmarking process and proposed benchmarking process. Conventionally, Project 1 would have to be assessed for 'N' number of times if Project 1 is interested in 'N' standards. The new proposed way would mean that Project 1 would only have to be assessed once. Furthermore, if a new standard were to be used, and this new standard is within the domain of the comprehensive reference model, means no additional assessment is required. Notice that conventional method might consist of inconsistent standard with other benchmark and might contain custom benchmark items. The new benchmark always provides consistency; assuming that all projects uses the same benchmark methodology. Conventional method work case of benchmarking is 0, which means not compatible benchmark partner was found. The average case is based on a range between 0 to N number of standards used. The best case is obviously N number of standards. Using the proposed new benchmarking means the worse case, average case and best case is always N number of standards covered by the comprehensive model.

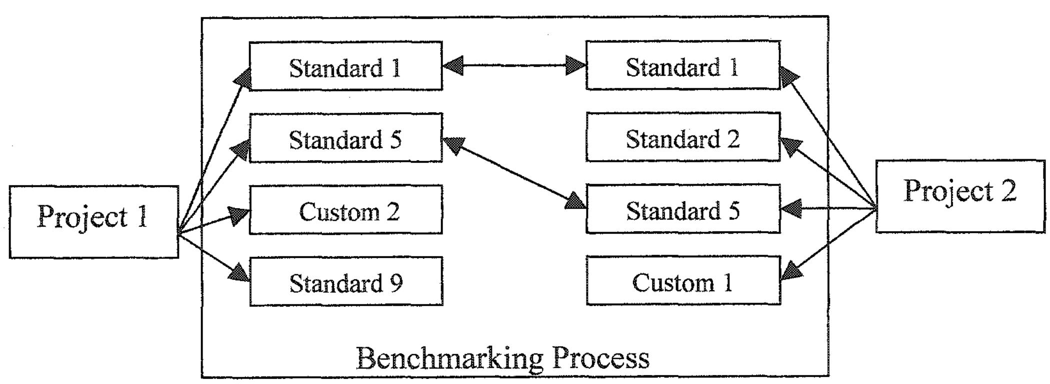

Figure 4-2 Conventional Benchmarking Framework

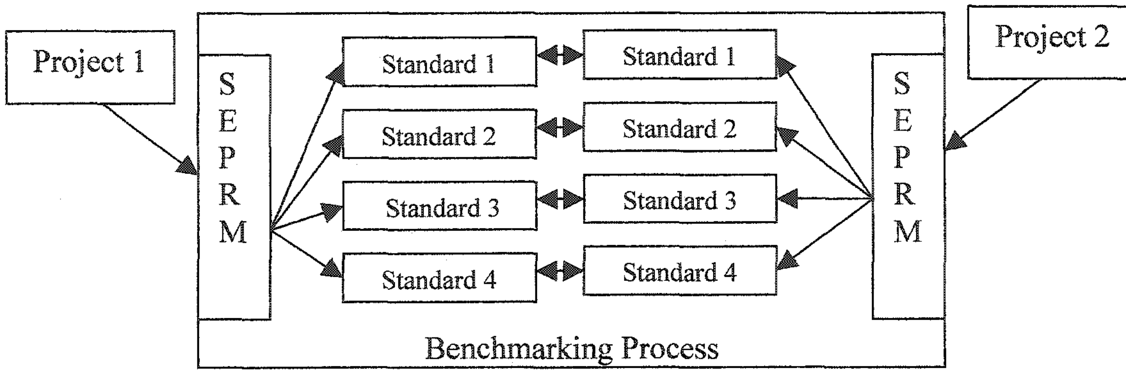

Figure 4-3 Proposed Benchmarking Framework

4.3.4 Gap Analysis

The assessment process covers the first three recommendations. Gap analysis is purely a benchmarking activity and the core of this work. A detailed gap analysis technique that will complement the recommendations previously made will be provided in section 4.4.

4.4 Enhanced Benchmarking Gap Analysis Techniques

This section develops a new gap analysis technique that works with process assessment-based benchmarking. The use of a comprehensive reference model further enhances interpretation and evaluation of the gap analysis techniques.

4.4.1 Industry Management Benchmark Description

This research begins by examining benchmarking from a management point of view with reference to Sun Tzu's Ancient Art of War. The following extraction from Sun Tzu's Ancient Art of War has relevancy and bearing to modern benchmarking.

One who has a thorough knowledge of oneself and one's competitors is bound to win in all competitions. One who knows oneself but not the competitors has only an even chance of winning. One who knows not oneself and the competitors is bound to perish in all battles.

Know your competitors, know yourself, and your victory will be guaranteed. Know the terrain, know the weather, your victory will be complete.

SUN TZU, THE ART OF WAR 500 B.C.
(Wee C, Lee K, Bambang W. H., 1996, modified by the author in translation)

The first paragraph can be loosely interpreted, as an organization must know its current state of capability and maturity levels, supplemented with the knowledge of its competitors' capability and maturity levels, in order to achieve superior performance in the market place. An important note found between the paragraph lines is: "knowing your enemy" implies knowing both your competitors' strengths and weaknesses. The desired result and situation will determine what process should warrant focus or which process should be paid attention.

The "situation" is referenced in the second paragraph and relates to knowing your terrain and weather. This means that an organization must know the environment, atmosphere, configuration, and other parameters that can help in both strategic and tactical planning. One simple technology to use is Organization Unit (OU) information. OU information such as industry type, location sector, project size, project type, technology utilization, etc., can be used to narrow down the competitors. OU information makes benchmarking efficient and benchmark evaluation more effective. A comprehensive list of organization units have been identified as part of ISO/IEC 15504 standard assessment [El Eman and Jung, 2001; El Eman, et al., 1998]. OU information has also been used successfully in an European web-based software process benchmarking server [Wang, Y, 2001a] and a national benchmarking survey in Sweden [Wang, et al., 1999] and related methodologies in software process assessment (SPA) and SPI.

4.4.2 Statistical Gap Analysis

Some of the statistical methods are examined for relevancy and applicability in industrial benchmarking. The main point is the application of statistical technique to industrial benchmarking gap analysis to generate value-added result.

The fundamentals of gap analysis are examining the magnitude of the gap and the vector of the gap. The magnitude of the gap is the range information. There are three types of gap information; namely, positive, parity and negative. The three gap types are illustrated in Figure 4-4.

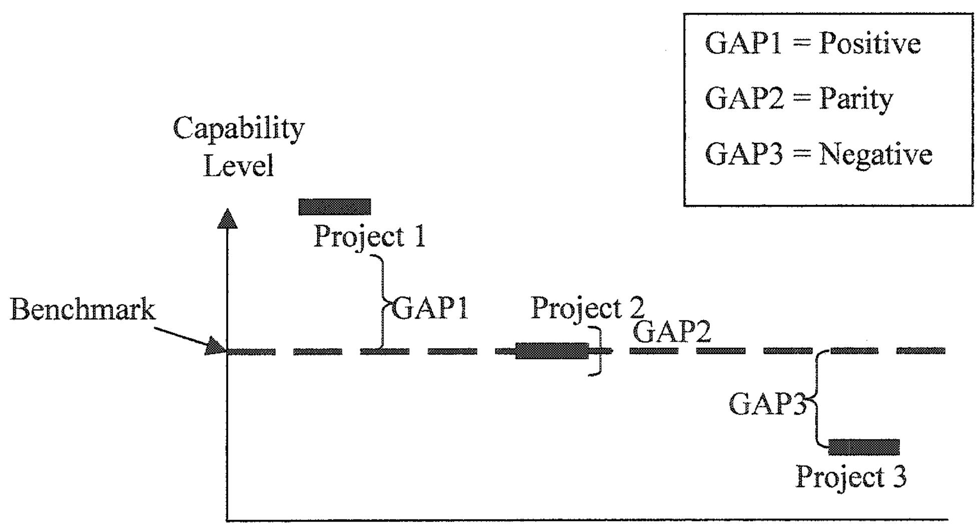

Figure 4-4 Positive, Parity and Negative Gaps

The magnitude of the gap is the range of the gap. The range is defined as the magnitude between two Capability Levels (CLs). The range is between the Project CL and the

Benchmark CL. The "range" is the simplest measure of dispersion utilizing two values [Defusco, et al., 2001].

The gap magnitude is the absolute value of the "range". The absolute value of the gap is expressed as |Gap|.

Hence,

$$
\begin{aligned}
|Gap| &= |Range| \\
&= |Value\ 1 - Value\ 2|; \\
&\qquad \text{Let Value1=Project CL and Value2=Benchmark CL} \\
&= |Process\ CL - Benchmark\ CL| \qquad (1)
\end{aligned}
$$

From Figure 4-4, we can further define the gap definition in rule or equation forms as follows:

Positive gap is when (Project CL –Benchmark CL) > 0 (Rule 1)

Parity gap is when (Project CL-Benchmark CL) = 0 (Rule 2)

Negative gap is when (Project CL-Benchmark CL) < 0 (Rule 3)

Positive gap (Rule 1) indicates that the Project CL is superior to the benchmark CL. Parity gap (Rule 2) indicates that the Project CL is at par with the benchmark. Negative gap (Rule 3) indicates that benchmark CL is superior to the project CL.

When looking at the various benchmark thresholds that are of interest to gap analysis, we need to know the following pieces of information.

a) Our current Project CL
b) The ranges of benchmarks
c) The average benchmarks
d) The maximum potential CL (Model Max CL) and minimum potential CL (Model Min CL)

These capability levels can be illustrated as follows:

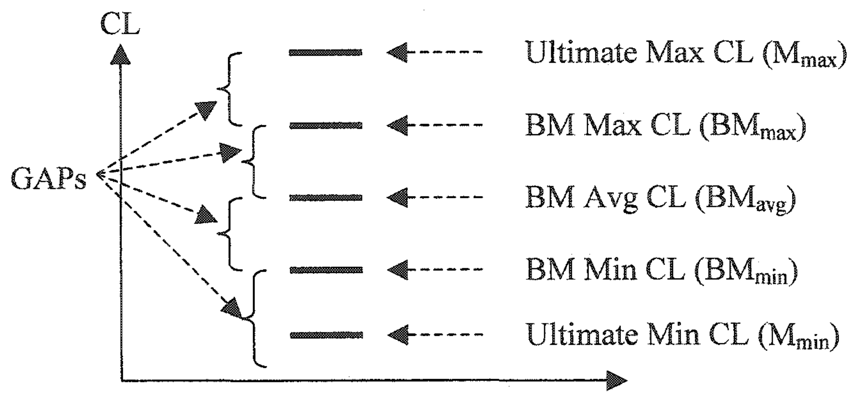

Figure 4-5 Various Benchmark CLs

Various rules can be derived for the gaps from Figure 4-5 Various Benchmark CLs, are applicable to all standards and models. The rules are as follows:

(Ultimate Max CL)	>=	(BM Max CL)	(Rule 4)
(BM Max CL)	>=	(BM Avg CL)	(Rule 5)
(BM Avg CL)	>=	(BM Min CL)	(Rule 6)
(BM Min CL)	>=	(Ultimate Min CL)	(Rule 7)
(Ultimate Max CL)	>	(Ultimate Min CL)	(Rule 8)

Notice that Rules 4 to 7 are not truly transitional because they are constrained by Rule 8. This implies that we should not have the case where,

(Ultimate Max CL) = (Ultimate Min CL) (Rule 9)

When the above rule applies to a standard or model, it means the standard or model has no provision for improvement or capability level determination; hence, cannot be benchmarked. Rule 4 to Rule 8 shows that BM Max CL, BM Avg CL, and BM Min CL is upper bounded by Ultimate Max CL and lower bounded Ultimate Min CL.

The next step is to examine the possible location of the project CL. The project CL can only lie between BM Max CL and BM Min CL exclusively as shown in Figure 4-5. The project CL can also lie right on the boundary of the gap. When the project CL is taken into consideration, more rules can be derived as follows:

(BM Max CL) >= (Project CL) =< (BM Min CL) (Rule 10)

(Project CL) >= (BM Avg CL) (Rule 11)

(BM Avg CL) >= (Project CL) (Rule 12)

(Rule 11) XOR (Rule 12) (Rule 13)

Notice that Project CL is also bounded between Ultimate Max CL and Ultimate Min CL, according to Rules 4 to 10. Rule 13 indicates that Rule 11 and Rule 12 are exclusive to each other.

Rules 11 and 12 provide indication that a project CL is superior to benchmark average or benchmark average is superior to the project CL, respectively. Rules 11 and 12 are very weak because average does not give any distribution information. The average value will be more meaningful if distribution of the benchmark data is known, such as how it deviates from the mean. We can use the mean equation such as that for standard deviation below:

$$m = \frac{\sum_{i=1}^{n}(CL_i)}{n} \tag{2}$$

Where, m - is the mean

n – is the total number of process CLs being benchmarking

CL_i – is the i'th process CLs.

The process benchmark in this thesis will provide information such as the process information and requirements for that level or the next benchmark level desired. Knowing the 'where' you are and 'what' you are, standard being benchmarked can provide suggestion as to 'how' to achieve the desired benchmark and forecast "when" to expect the desired superior result. An example of project distribution histogram is provided in Figure 4-6.

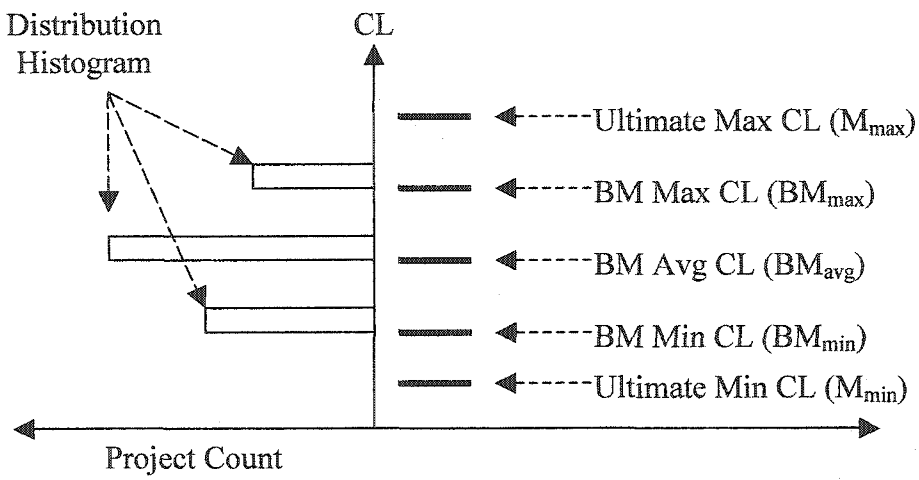

Figure 4-6 Example of Project Distribution Histogram

The four statistical methods found useful for gap analysis is the capability level maximum value, average value, minimum value and distribution histogram. The final effective presentation of the statistical information and its distribution will be presented later.

4.4.2.1 Benchmark Statistical Methods an Industry Perspective

Maximum Benchmark: This information indicates that at least one project has reached the maximum level of capability in the benchmark sample. The focus of knowing maximum capability level of a process is used to determine how well a process can mature to and the capability level that is achievable, and achieved by others. Achieving this level usually indicates that the superior performance has been met.

Average Benchmark: This information indicates the sample average of all the benchmark projects. This information can be distorted by data with extreme value. This information could be further refined by examining the distribution of the data. Ideally, benchmark mean should be used. In this thesis it is assumed that the benchmark data are representative of the best practice in the population, therefore, taking average should be sufficed for this purpose. This value will indicate to the current project a perceived distance from a group of projects' capability ranking. Normally, above average is considered good.

Minimum Benchmark: This information indicates that at least one project is at the lowest capability level in the benchmark sample. This value will indicate to the project that there exists at least one or more project at the worst level for that particular process CL. If the current project is sharing the same value as the minimum value could potentially mean that the current project is the worst of the lot. Alternate views can be that this value is an indication of practice opportunity or practice creativity.

Benchmark Capability Level Distribution Histogram: This information indicates how dispersed the projects are among the various capability levels. A standard division is not used for two reasons. First of all, this calculation is based on average and not the mean. Secondly, due to the small range of capability level values, ranging from one to five with one significant digit decimal place, it is just as effective just to plot the capability level distribution histogram. The information provided by the histogram is easier to interpret with similar value information that could have been derived from standard deviation.

This section proposes the charting of frequency histogram of the number of projects within a given capability level for a given particular process model. This is usually sufficient enough to make a decision on whether to proceed or not with an action. Most bid proposal and market position analysis can tell how many competitions are doing the same thing. Thus, this number can be useful to perform correlation between the number from market/sale research and benchmark number. We know that this does not indicate that the number represents the specific projects of interests from the market/sale research. The trade-off is benchmark provides comparative ranking information in an honourable way.

The benchmark legend presented in Figure 4-7 will be used to illustrate the information previously mentioned. The graph representation idea was based upon and developed further from articles by Khaled El Eman and his colleagues' [El Emam and Birk, 2000; El Emam and Jung, 2001] box and whisker plots. Since this implementation is slightly different, the stock graphing tool in Excel is used instead. The stock graphing tool, and box and wicker share similar characteristics, but the stock graphing tool is readily available and serves the need of this thesis benchmarking purpose for the time being.

The White block in Figure 4-7 is an indication that the project's capability level is above the Benchmark's average capability level. The Black block is an indication that the project's capability level is below Benchmark's average capability level. A horizontal line instead of a block would be an indication that the project's capability level is equal to the benchmark's average capability level. This style of presentation is extremely effective for both at a glance information and detail information analysis.

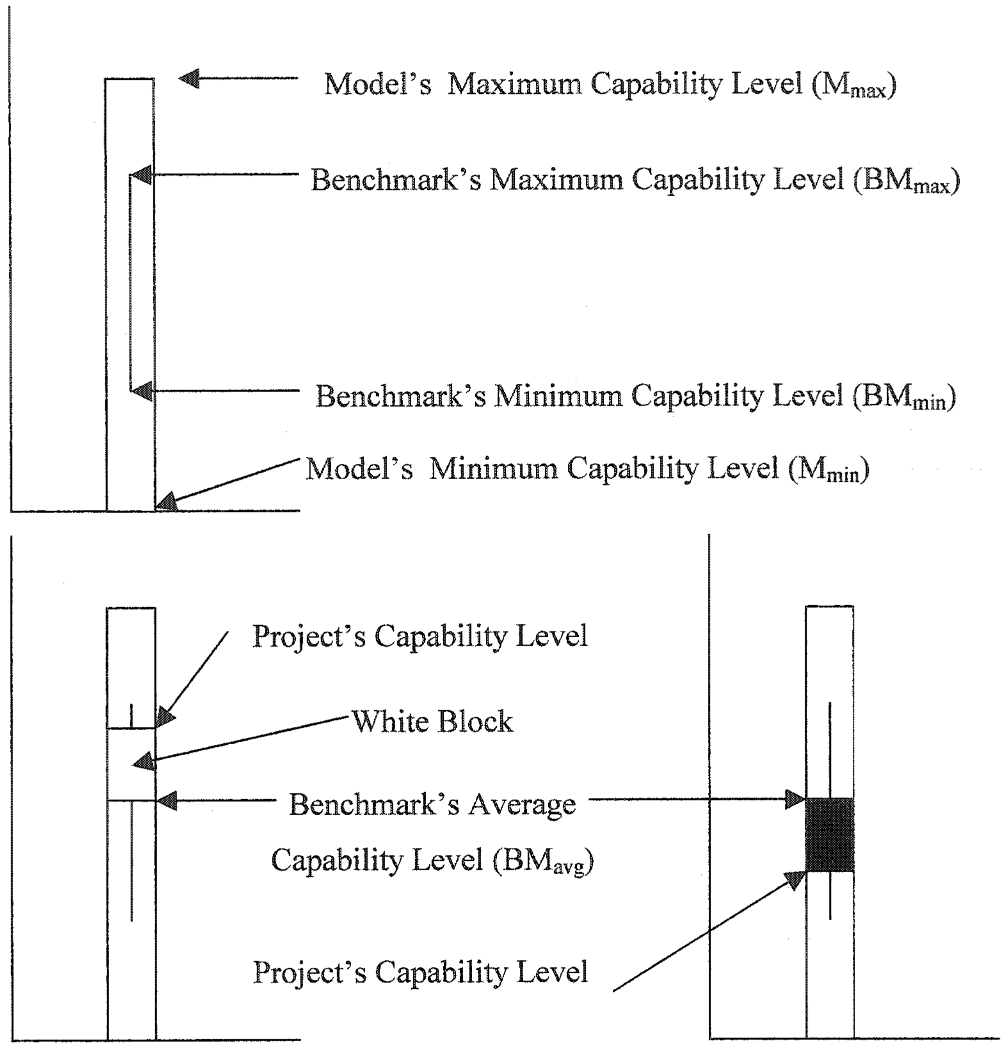

Figure 4-7 Benchmark Legend

4.4.3 The Benchmark Database

All the projects used in this research will be plotted in a bar-chart fashion to show project distribution at the project process level. The numbers and percentages of projects for a given CL are provided for each of the model used in this research. This information will indicate where all the projects rank for all the various models and standards capability level. It gives a very clear indication of the sample projects standings for the models of interest. Models and standards impact and influence on the determination of the project CL are analyzed with this information.

It is imperative for a project to benchmark itself against various models. The different models have their own agenda and philosophy on software QA improvement methodology. Comparison with percentile ranking information among various models can indicate how all the projects rank for all the various models and standards, among themselves. Furthermore, the comparison can help an organization to select a model that best suits its current state of SEP. Using these standards and models can provide suggestions and guidelines for organizations to implement or to improve its current SEP state.

4.5 The SEPRM Assessment and Benchmarking Tools

A SEP assessment tool was developed by the author to implement process assessment and measurement according to each process model described in previous sections. The SEPRM reference model provides a unified framework for designing the tool. The SEP Benchmarking Tool is a technology to assist in the process of gap analysis and gap interpretation. The tool is implemented in a spreadsheet application. The reason for choosing a spreadsheet is that it can better manipulate numbers, present tables of information and graph information, efficiently and effectively than a word processor or database application.

4.5.1 Design and Architecture

The initial validation of the tool was done by using the data provided in [Wang and King, 2000]. The algorithms and process mapping implemented in the tool were developed based on SEPRM. Later, more validations were performed based on data elected from real industry assessment results. The validated results indicate that the algorithms provided in [Wang and King, 2000] and the tools implemented are correct, but also point out some typos in the textbook and potential errors in some assessment results.

The architecture of the SEP assessment tool can be found in Figure 4-8. The two dashed boxes are the SEP assessment and benchmarking tools. The original design of the tool was to collect SEP assessment results via the Internet using existing assessment results, and performing actual informal assessment data collection. Thus, there are three forms of inputs. The Internet and Intranet inputs proof-of-concept worked but was never executed due to time limitation. It was faster to use a spreadsheet to collect data or to input existing data directly into the application.

There are two main outputs from the SEP assessment tool. The spreadsheet export is for benchmarking purposes, while the direct reports generation is for assessment result validation. Reports are useful for debugging and re-validating the tool when changes were implemented to the tool. The reasons for exporting the assessment results to a spreadsheet are for this thesis' benchmarking purpose and for other researchers to utilize the results.

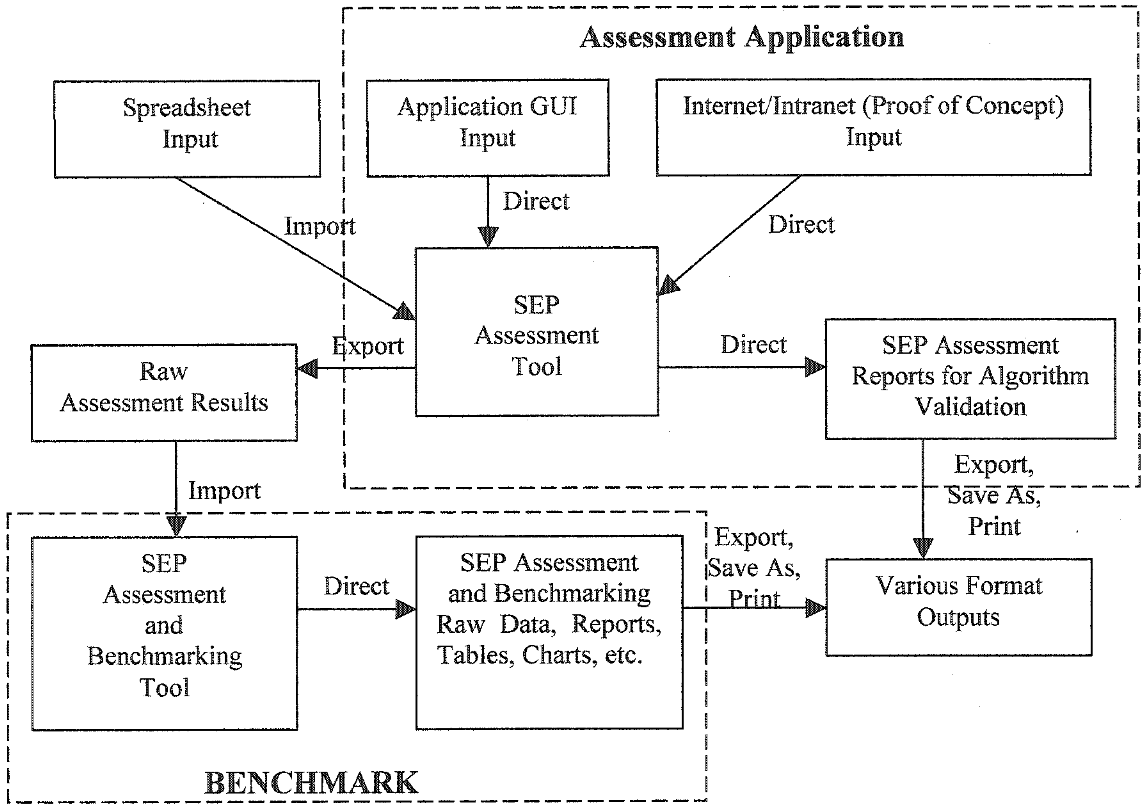

Figure 4-8 SEP Assessment Tool Architecture

Figure 4-9 is the design of the Benchmarking Tool Architecture. The dashed lines are the auto update of information to the various spreadsheets.

There are two selectors within the application. The first is the project selector;. the second is the information selector. Once a project of interest is selected, the raw assessment data will be copied to an intermediate spreadsheet. It is from this intermediate spreadsheet that all the other spreadsheets are automatically updated. The information selector allows the user to select various representations of assessment results or benchmarking results.

The SEP assessment tool exports only meta SEP assessment results and some meta SEP benchmark results. Only meta SEP assessment results were exported to provide researchers with relevant information to perform further investigation. Only some meta

SEP benchmark results were exported because the gap analysis calculation was more efficiently implemented in SQL in the SEP assessment tool.

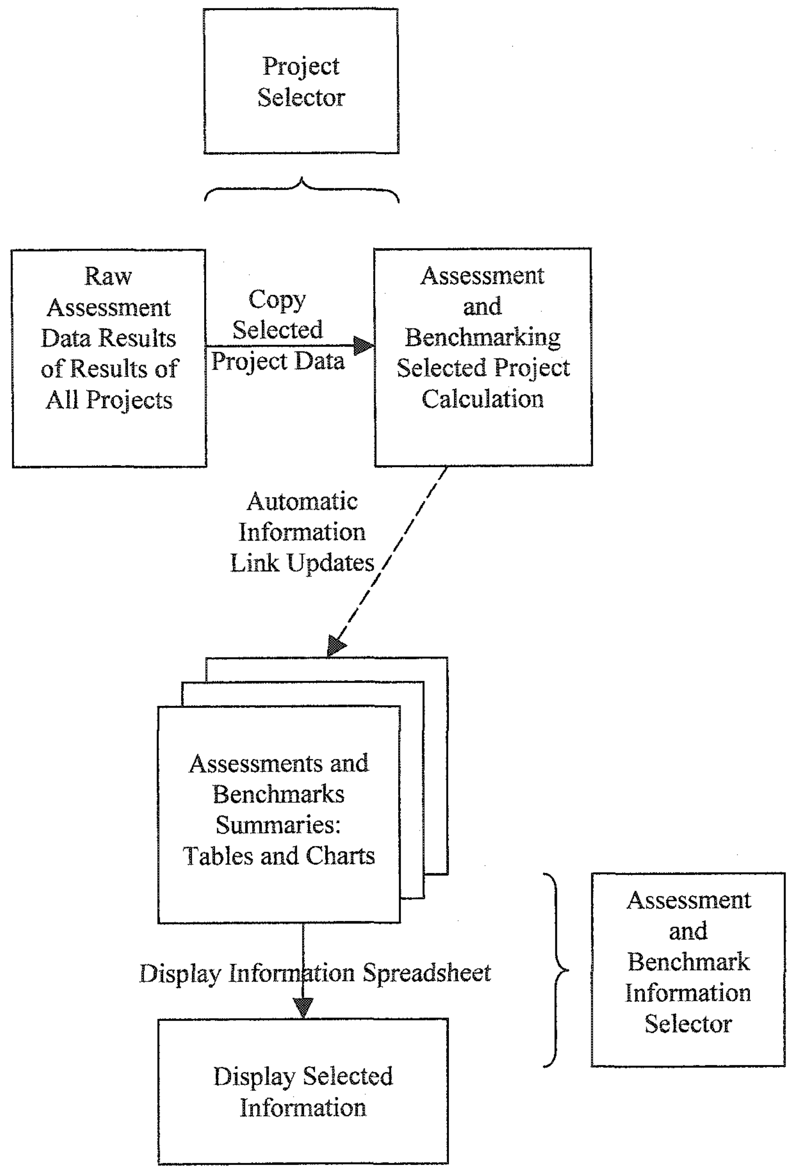

Figure 4-9 SEP Benchmarking Tool Architecture

4.5.2 Data Structures

Figure 4-10 illustrates the design of the SEPRM data relationship. On the left are tables that store the SEP assessment results. These results are linked to an associated project. On the far right, the various standards and model-based activities are mapped to SEPRM. The mappings are implemented by look-up tables that are mapped to SEPRM BPA. The project table is the main driver in the tool. The user selects the project of interest and the appropriate project's BPAs are extracted based on the look-up table and assessments are calculated and then stored in the assessment tables on the left.

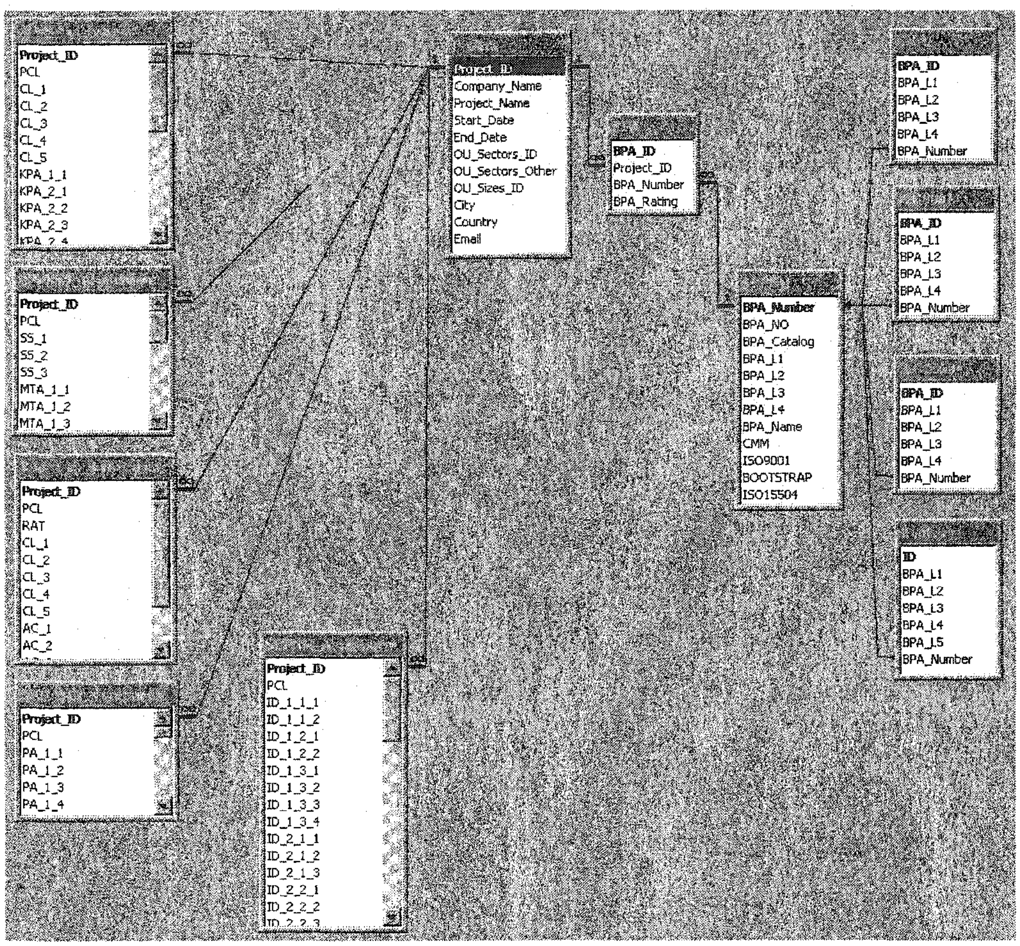

Figure 4-10 Data Relationship Diagram

4.5.3 Execution and Application

Figure 4-11 shows the main menu of the assessment tool. The first thing to do is to enter project information such as project size, location, industry sector, etc. Secondly, the assessment data is the input for the project. Finally, assessment results are produced for the project. There are quite a number of limitations in the technology used to build the

SEP assessment application such as table size limitation and limited programming language capabilities. All database access is done through Structured Query Language (SQL). Originally, the author did manage to write most of the assessment algorithms using SQL, however, due to the limitation of SQL in manipulating temporary run time variables manipulation, the final algorithms implementations were done in Visual Basic Application (VBA).

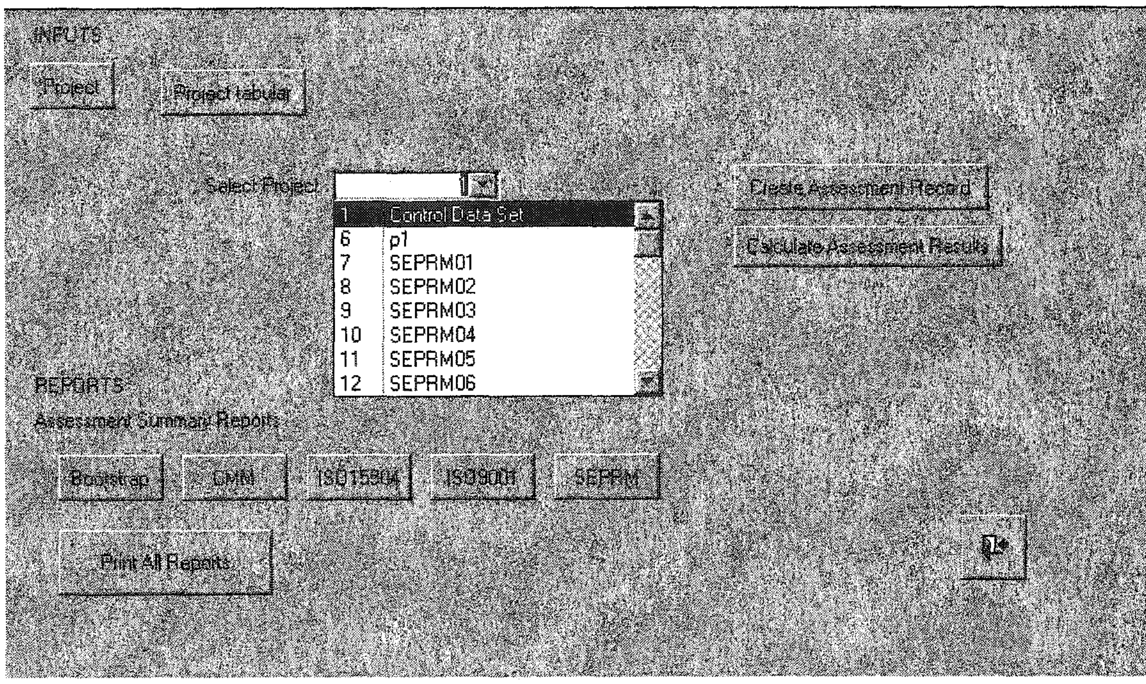

Figure 4-11 SEP Assessment Tool Main Menu

In this chapter, various SEP standards and models have been discussed. In addition, the tool implementation and its algorithms implementation were also presented. Figure 4-8 shows how the assessment results act as input to the benchmarking process. The next chapter will discuss this SEP benchmarking process that consists of the new benchmarking methodology and gap analysis techniques.

Figure 4-12 shows the hyperlink information selector and the pop-up project selector. The initial implementation of the benchmark tool using hyperlinks is to avoid programming. The pop-up project selector was eventually implemented because it is easier to prototype with all the code consolidated in one location instead of having formulas scattered all over the various spreadsheets and cells.

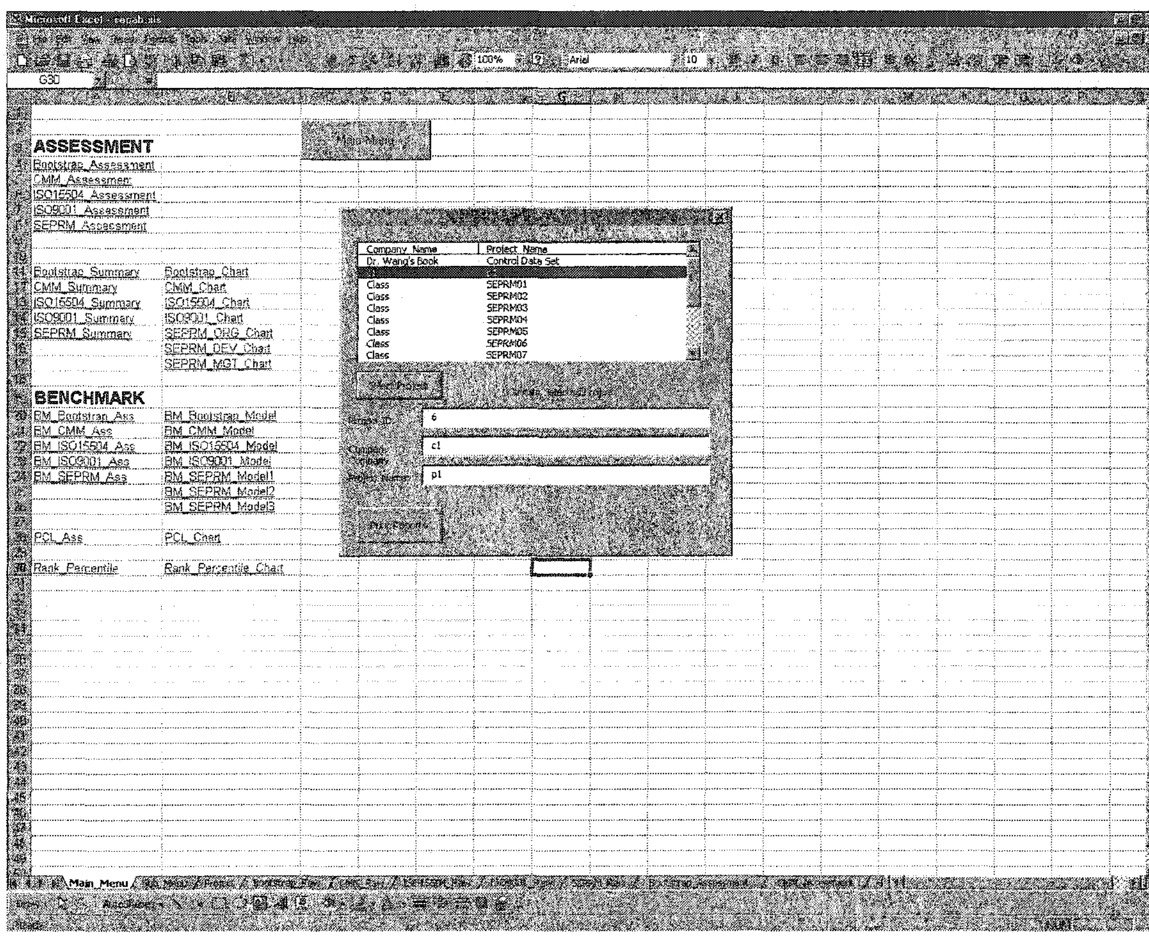

Figure 4-12 SEP Benchmarking Tool Main Menu

4.5.3.1 Benchmark Result Verification

The following rules (Rules 14 to 18) were implemented as verification to ensure assessment data integrity for benchmarking purposes were maintained. The rules were a contingency plan in case the assessment data processing process introduced bugs, such as new standard implementation. These rules must be true in order to determine if the benchmark results have been verified.

Project BM <= BM Maximum	(Rule 14)
Project BM >= BM Minimum	(Rule 15)
BM Maximum >= BM Average	(Rule 16)
BM Maximum >= BM Minimum	(Rule 17)
BM Average >= BM Minimum	(Rule 18)

The partial codes for Select Project button and Print Reports were presented to illustrate ease of programming. This level of programming is suitable for a project administrator to manage. The point is to illustrate that maximizing the embodied knowledge within capital cost can yield great dividend, in terms of quality value-added information presentation for an effective decision making process.

4.5.3.2 Partial Code for Pop-up Project Selector Button

Based on the project selected, this subroutine copies raw data from the raw data spreadsheet onto the benchmark models' spreadsheet.

```
Private Sub btn_Select_Project_Click()

Dim selected_ListIndex As Integer
Dim row_string As String
...

selected_ListIndex = ListBox_ProjectID.Value

selected_ProjectID = Worksheets("Project").Cells(selected_ListIndex + 2, 1).Value
selected_CompanyName = Worksheets("Project").Cells(selected_ListIndex + 2, 2).Value
selected_ProjectName = Worksheets("Project").Cells(selected_ListIndex + 2, 3).Value

fld_ProjectID.Value = selected_ProjectID
fld_CompanyName.Value = selected_CompanyName
fld_ProjectName.Value = selected_ProjectName

Worksheets("Sub_Menu").Cells(4, 2).Value = selected_ProjectID
Worksheets("Sub_Menu").Cells(5, 2).Value = selected_CompanyName
Worksheets("Sub_Menu").Cells(6, 2).Value = selected_ProjectName

row_string = CStr(selected_ListIndex + 2) + ":" + CStr(selected_ListIndex + 2)

    Sheets("SEPRM_Raw").Select
    Rows(row_string).Select
    Selection.Copy
    Sheets("SEPRM_Assessment").Select
    Rows("2:2").Select
    ActiveSheet.Paste

... similar code for Bootstrap
```

```
… similar code for CMM
… similar code for ISO 15504
… similar code for ISO 9001

    Sheets("Main_Menu").Select

End Sub
```

4.5.3.3 Partial Code for Print Reports Button

This subroutine essentially prints out all assessments and benchmarks results tables and charts.

```
Private Sub PrintReports_Click()
    Sheets("SEPRM_Summary").Select
    Sheets("SEPRM_Summary").Print
    Sheets("SEPRM_ORG_Chart").Select
    Sheets("SEPRM_ORG_Chart").Print
    Sheets("SEPRM_DEV_Chart").Select
    Sheets("SEPRM_DEV_Chart").Print
    Sheets("SEPRM_MGT_Chart").Select
    Sheets("SEPRM_MGT_Chart").Print
    Sheets("Bootstrap_Summary").Select

… similar code for Bootstrap
… similar code for CMM
… similar code for ISO 15504
… similar code for ISO 9001

    Sheets("Main_Menu").Select

End Sub
```

4.6 Benchmark Enhancement

Benchmarks can be enhanced in many different ways, and two ways are provided here. The first way is benchmark-based software process improvement and the second way is through software engineering organization's methodology knowledge management. The fundamental objective is to improve process because process is the benchmarking item. Even a small process improvement, without excessive additional resource can reduce project cost leading to increased revenue margin. Process improvement is essential in today's market of fixed-price project, where cost and quality are usually predetermined requirements.

4.6.1 Benchmark-based Software Process Improvement

Since the benchmarked items are processes, the gap result will identify processes of interest for improvement. Yingxu Wang [Wang and King, 2000a] indicated that conventional SPI has been goal based with the philosophy "the higher the better". So, he proposed benchmark-based process assessment and improvement provides a new approach to adaptive and relative process improvement based on a philosophy of "the smaller the advantage, the better". This is a new approach as opposed to the conventional model-based assessment. Based on the benchmarks, refined process improvement can be planned in a much more accurate manner.

A number of benchmark-based SPI features have been identified in [Wang and King, 2000a]. The philosophy now is to fill the gap as opposed to achieving "the higher the better". In order to fill the gap, the approach to SPI now is gap analysis, usually through a plotted process profile. The process in the plotted profile will be qualified then quantified for an action item. The action item will be verified based on the magnitude the gap has narrowed.

4.6.1.1 SPI with the Proposed Benchmarking Methodology

The role of SEP benchmark in software engineering life cycle is normally in the area of SPI. Figure 4-15 shows the injection points where benchmark information can be used according to part of the SPI flow diagram.

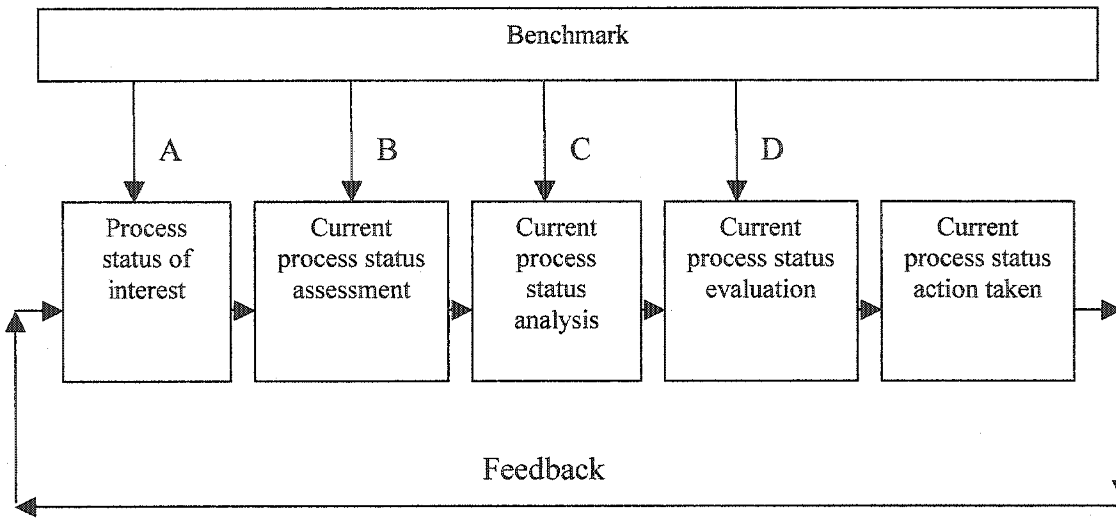

Figure 4-15 Generic SPI Flow Diagram

Table 4-1 provides a summary of where, when, and how to use benchmark information in the SPI effort.

Table 4-1. The Roles of Benchmarking in SPI

Benchmark Information Stage	Benchmark Information
Process status of interest (A)	• Benchmark based process goals • Selective benchmark process improvement
Current process status assessment (B)	• Benchmark of process and project capability level information
Current process status analysis (C)	• Benchmark process gap information • Benchmark based tactical plan
Current process status evaluation (D)	• Benchmark based strategic plan

- Process status of interest

At this state, a process of interest or concern is identified and has been qualified. A process at this stage does not exist, needs implementation, needs modification, needs deletion or needs investigation. There are two roles that benchmark plays at this stage. First of all, a process of interest at this stage could be derived from benchmark and be set as an organization's goal. This process could be part of the GQM paradigm of an organization. In this case, the process requires constant monitoring and tracking. The process capability level is software metric. Secondly, benchmark role contribution at this state, is to identify process that requires practice opportunity to get ahead of the competitor to get an edge in the market place.

- Current process status assessment

The current process assessment state is where the process is measured for analysis. In the case of this thesis, the process capability level is measured. At this stage, benchmark provides assessment capability level comparison with both competitors and standards. At this point, either external benchmarking or internal benchmarking could be performed.

- Current process status analysis

The process analysis state is where the measurement is used to generate more information such as derive measures, statistics, and distribution. The thesis proposed benchmark can produce similar information for comparative analysis for the newly generated information from measurement. The information provided by benchmark here also contributes to tactical planning for the SPI effort. This is the state where the technical detail of how to improve the process takes place. This relates to short-term planning usually on an annual basis.

- Current process status evaluation

This is the final state of SPI for one cycle or iteration. Availability of historical benchmark data could enable benchmark trend analysis. Trend analysis can enable an organization to project into the future, forecasting when superior performance ahead of the competitors might be achieved. This is essentially long term strategic planning. Strategic planning at this stage can take place annually or every couple of years.

4.6.2 Software Organization's Knowledge Management

The weaknesses identified by a benchmark gap can be improved in a number of ways. Two industry approaches will be presented here that can help narrow the benchmark gap.

The first way is the utilization of an organization's standardized software engineering methodology. The second method involves using SEP to engineer the organization's knowledge. An informed knowledge of implementation can lead to better process execution, hence, leading to gap reduction.

4.6.2.1 Standardized Software Engineering Methodology

Since we are benchmarking process it is obvious that in order to improve the benchmark result we have to improve the process. Process can only be improved through activities improvement. Activities improvement is usually a relatively easy task to perform, but to maintain the improvement capability maturity level is difficult.

One approach is to improve the activities through a standardized jobs related methodology. This means, for a given set of job processes, there is a predefined methodology that combined the right processes and activities together and executes them accordingly to the methodology.

For example, if everyone writes a document from the same template, we can ensure consistency. And we know that it is possible to improve the quality of the document by improving the template. This is similar to design patterns in object programming. But when it comes to SEP, it has to be a methodology.

A Software Engineering example would be to apply an Object-Oriented Methodology (OOM) to programming [Chiew, 2001]. The OOM is independent of the programming language, the hardware platform and the programming environment. What this means is that the process has been executed in a consistent manner, including the product quality. The author's experience with OOM has enabled consistent approach to high-level language programming down to low-level hardware device programming. One real advantage is that by explaining the methodology to one person, the person can easily read any code written that follows the methodology. If everyone follows the same methodology, then we can ensure consistency in processes and activity being carried out. Like Level 2 in CMM (Repeated), this is the first step towards capability maturity.

There are technologies that can help in this area such as a tool called Creator [Koono, et al., 2001]. Creator has many features that can be advantageous to process enhancement and can contribute greatly to the aspect of SEP knowledge management. The methodology described here is similar to one of Creator's capability which is to aid in the software engineering design process. The focus has been diverted from people focus to process focus in an objective automated way. By using Creator, one can ensure uniformity in the design process implementation and the design process execution. Using a tool technology as an aid can increase productivity and potentially quality.

By using a standard methodology for a particular job ensures process is executed in a consistent manner leading to manageability. Since the methodology relies on process and not the person doing the job or the technology involved, then an improvement in methodology can span company wide across all projects. What this means is an organization capability level may mature faster in a shorter time period with minimum effort. Time saving could potentially lead to cost saving.

4.6.2.2 Software Engineering Organization's Knowledge Management

A second way of process improvement related to the methodology approach described earlier is related to process knowledge management. The focus of this knowledge is a combination of project business specific knowledge and organization methodology knowledge.

Knowledge management of the methodology is not complete if it is not coupled with the specific project business knowledge methodology. Specific project industry could be a project of similar types. Software Engineering can play a big part in such a knowledge management by creating a software engineering framework to manage the knowledge management process [Chiew, 2002]. Basically, the software engineering framework is used to support an organization's knowledge acquisition, knowledge training, knowledge retention, knowledge configuration, knowledge implementation and knowledge reuse. The author has worked with a process system that just by entering the parameters of the projects, all software related outputs are automatically generated like install script, configuration script, code frame or shell, not only for the platform it is running on but for all related hardware that will affect the applications. This means a lot of the processes and lower level activities have been consolidated into a higher level automated process. This automated process is obviously knowledge embodiment. There are similar technologies that can be of aid in the software engineering organization's knowledge management. One such technology is the MILOS Systems [Maurer and Holz, 2002]. This system has lots of good attributes and features, but is not limited to a repository system for best practice management. It provides process framework support and is flexible enough for process model customization. The use of such technology can ensure process consistency, traceability, assurance, and improvement that are essential for progressing up the SEP capability maturity levels. This system also embodied a wealth of knowledge.

From the author's industrial experience, the knowledge implementation process is a combination of having a project's business knowledge and an organization's methodology knowledge. A framework that may be used for supporting afore mention combination can be found in [Chiew, 2002]. The examples provided essentially embody the project business specified into the automated tools or systems. Knowledge implementation process that possessed the combination described previously makes process execution relatively efficient to execute and manage.

The newly proposed benchmark methodology and gap analysis techniques provide solutions to the problems identified in Section 1.5.1. In addition, some industry process enhancement approaches that can benefit benchmark score were introduced. The process enhancement suggested is useful as an action item to increase the benchmark scores. Now, what is needed are case studies to verify the proposed benchmarking methodology and gap analysis techniques application in industry by executing benchmarking process in an efficient manner and generate value-added benchmarks for effective decision making. The case studies will be described in the next chapter.

CHAPTER 5 INDUSTRIAL CASE STUDIES

Case studies are presented in this chapter to demonstrate and verify the proposed SEP benchmarking methodology developed in the previous chapters. In addition, these case studies will also provide detailed analysis using the gap analysis techniques as described in Chapter 4. The efficiency and effectiveness of the proposed benchmarking methodology and techniques will be verified in the end of this chapter.

The assessment data of this thesis are collected based on self-assessments and internal assessments. Case in point, a benchmarking company in Australia, with over 10 years of benchmarking experience, has shown that a self assessment then benchmarking methodology approach has proven beneficial to over 400 Australian enterprises and over 1300 corporate members [Asian Productivity Organization, 2001].

An informal assessment was performed for the case studies. The author and one Quality Assurance engineer performed the informal assessment base on the organization's best practice documentation (organization's benchmark). Both assessors were familiar with the organization's processes. The informal assessment effort involved filling out the ratings for a questionnaire consisting of 444 processes. These ratings were then used as part of the project appraisal and assessments. The project assessments were then used for benchmarking. Additional assessment and benchmarking information can be found in APPENDIX C:.

5.1 Introduction

Industrial case studies are a way to verify the proposed SEP benchmarking methodology and gap analysis techniques. The case study is based on an internal project at organization X. This is a project based on the company's procedure which reflects its ideal best practice. The project is assumed to have followed the company's procedure. This is an interesting approach because a project is usually customized to specific customer requirements. By analyzing the organization ideal best practice procedure, will enable the organization to evaluate its current state of processes against all the related benchmarks in the research database. For the purpose of this case study, this ideal best practice procedure based will be referred to as projectX practice. The motion of performing internal project benchmarking will be similar to the benchmarks collected for tactical planning purposes, except for the use of external projects. External projects were used for the tactical planning portion of this thesis because the case study's organization cannot spare any more time and resource for further assessments. The internal project's goals will be to meet the organization's best practice instead of exceeding it, unless warranted by the contacts, statement of work, or situation on hand.

Internal to the organization, the projectX practice also set a threshold for internal projects to benchmark against. Advantages of this approach of benchmarking organization projectX practice are enabling the upper management to perform company-wide strategic planning and allowing projects to perform tactical planning at the project practice level.

The company's procedure was once applied to a project and managed to achieve ISO 9001 registration and SEI CMM level 3 certification. Since the actual project was too old, this ideal project is used for the case study. In addition, the company cannot afford the resources to provide actual project data. Furthermore, no detail record of the actual assessment was available but the report still exists. No detail record keeping of the assessment in itself does say something about the capability of the quality system currently in place and its role in SEP continuous improvement of the company. A few years have gone by since the organization was last certified by SEI and registered by ISO. The organization's ideal best practice documentation has gone through numerous changes. Those changes are accounted for the organization's current state of informal assessment as indicated in this chapter.

There are still some interesting points that can be obtained from studying projectX. Evaluation of the projectX practice enables it to be evaluated against the external projects' benchmarks. This further helps us to refine the company's definition of best practice at the process level by examining at the activities level for refinement.

5.2 Assessments

The assessments of projectX are conducted by using the five models; CMM, ISO 9001, BOOTSTRAP, ISO/IEC 15504 and SEPRM. The assessment for each standard will be presented first in a table that summarizes projectX process capability levels Accompanying the table is a figure illustrating the process capability level in a graphical form. The graphical form presents a summary to determine the strengths and weaknesses of the projectX practices for each respective standard. Finally, each graph has an accompanied paragraph providing some detail interpretation and analysis of the assessment of the projectX practice current state for the particular standard.

5.2.1 CMM Assessment

According to the CMM assessment, see Table 5-1, projectX achieved CMM Level 4. The projectX practice presented is doing very well overall, with strengths in managing, monitoring and tracking of the current state of events. The weaknesses of the organization are in the areas of "Process change management (KPA 5.3)" and "Organization process focus (KPA 3.1)". This information is verified by knowing the organization work on long term fixed-cost contracts and has the tendency to want to maintain status quo. The organization has to pay more attention to its organization's processes, particularly in the areas of process change management. Assuming all process require the same level of effort, to achieve CMM Level 5, the organization has to work on 23% (6/(26 CL 5 KPAs)* 100) of the level total processes. Alternatively, the organization can spend less than half the resources, ((6 CL 5 KPAs)/(3 CL 3 KPAs) = 2), to work on fully satisfying CMM Level 3 KPAs, depending on budget and resource availability.

Table 5-1. Summary Assessment Record in CMM

Capability Level	Key Practice Areas	Identified KPs	Assessment Result (SAT)	Assessment Result (UNSAT)
CL 5	Optimizing	26		
KPA_5_1	Defect prevention	8	8	0
KPA_5_2	Technology change management	8	8	0
KPA_5_3	Process change management	10	4	6
CL 4	Managed	12		
KPA_4_1	Quantitative process management	7	7	0
KPA_4_2	Software quality management	5	5	0
CL 3	Defined	50		
KPA_3_1	Organization process focus	7	4	3
KPA_3_2	Organization process definition	6	6	0
KPA_3_3	Training program	6	6	0
KPA_3_4	Integrated software management	11	11	0
KPA_3_5	Software product engineering	10	10	0
KPA_3_6	Intergroup coordination	7	7	0
KPA_3_7	Peer reviews	3	3	0
CL 2	Repeated	62		
KPA_2_1	Requirement management	3	3	0
KPA_2_2	Software project planning	15	15	0
KPA_2_3	Software project tracking and oversight	13	13	0
KPA_2_4	Software subcontract management	13	13	0
KPA_2_5	Software quality assurance	8	8	0
KPA_2_6	Software configuration management	10	10	0
CL 1	Initial	0		
KPA_1_1	Initial	0	0	0

Figure 5-1 is a graphical representation of the process profile of projectX in CMM.

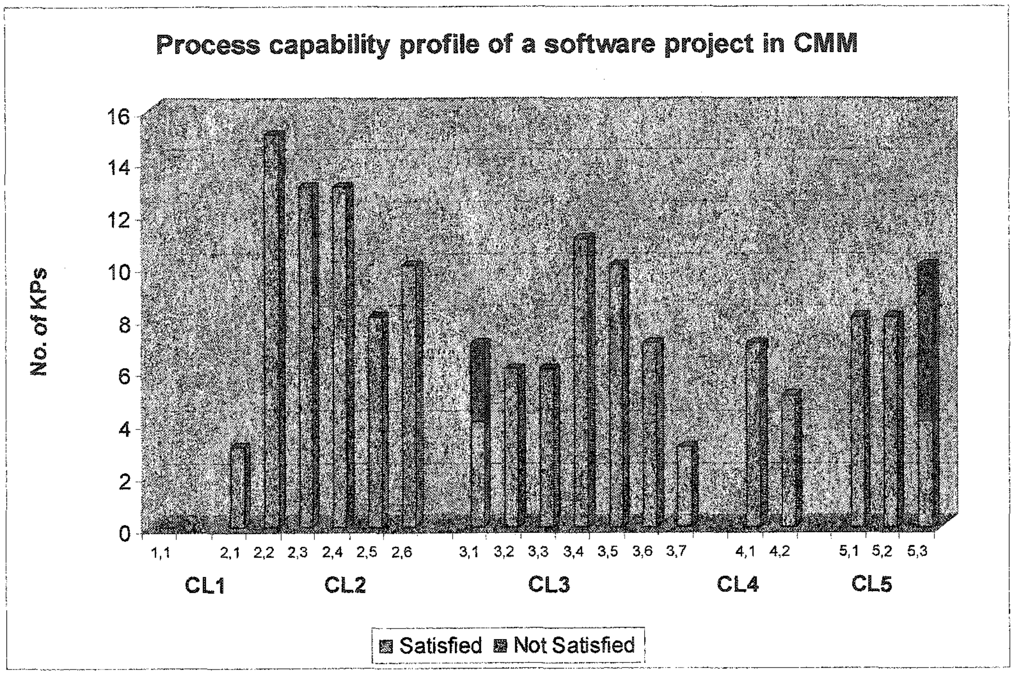

Figure 5-1 Process Capability Profile of a Software Project in CMM

5.2.2 ISO 9001 Assessment

According to the ISO 9001 assessment, see Table 5-2, projectX failed to pass the assessment. This model insists that all Main Topic Areas (MTA) must be satisfied in order to pass its assessment. Satisfaction in all MTA is required in order to be eligible for ISO 9001 registration. Again, this projectX practice strength is pretty well distributed in all ISO 9001 MTAs, similarly to its CMM assessment. Its weaknesses lie in the "Management responsibility (MTA 1.1)" and "Design and development control (MTA 3.3)". The organization projectX practice has only two MTAs to work on in order to pass the ISO 9001 assessment.

Table 5-2. Summary Assessment Record in ISO 9001

NO.	Subsystem	Main Topic Area	Pass Threshold	Assessment Result (SAT)	Assessment Result (UNSAT)
SS1	Organization Management		53	52	1
MTA_1_1		Management responsibility	15	14	1
MTA_1_2		Quality system	7	7	0
MTA_1_3		Document and data control	8	8	0
MTA_1_4		Internal quality audits	6	6	0
MTA_1_5		Corrective and preventive action	6	6	0
MTA_1_6		Quality system records	7	7	0
MTA_1_7		Training	4	4	0
SS2	Product Management		31	31	0
MTA_2_1		Product management	4	4	0
MTA_2_2		Control of customer-supplied product	4	4	0
MTA_2_3		Purchasing	8	8	0
MTA_2_4		Handling, storage, packaging, preservation, and delivery	9	9	0
MTA_2_5		Control of nonconforming product	6	6	0
SS3	Development Management		93	92	1
MTA_3_1		Contract reviews	9	9	0
MTA_3_2		Process control	23	23	0
MTA_3_3		Design and development control	30	29	1
MTA_3_4		Inspection and testing	11	11	0
MTA_3_5		Inspection and test status	2	2	0
MTA_3_6		Control of inspection, measuring, and test equipment	12	12	0
MTA_3_7		Statistical techniques	2	2	0
MTA_3_8		Servicing and software maintenance	4	4	0

Figure 5-2 is a graphical representation of the process profile of projectX in ISO 9001.

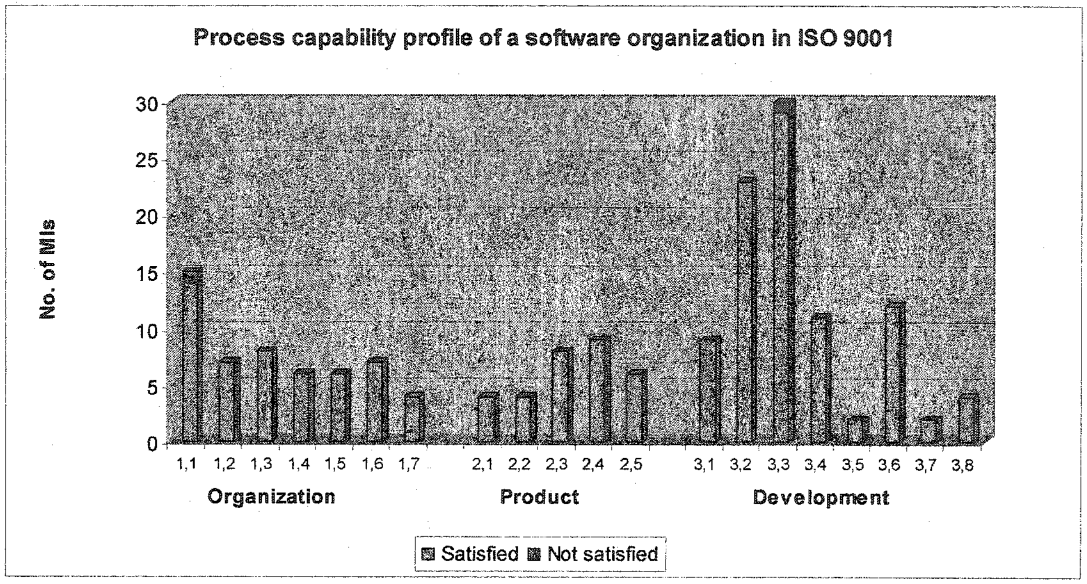

Figure 5-2 Process Capability Profile of a Software Project in ISO 9001

5.2.3 BOOTSTRAP Assessment

According to the BOOTSTRAP assessment, see Table 5-3, projectX practice has achieved BOOTSTRAP Level 5 capability maturity level. This assessment model capability is based on the cumulative process capability satisfied. Even though the organization has reached the highest level offered by the model, the organization now has the options to either try to work on maintaining status quo and work on CPI. The areas to work on would be the individual processes categorized within the given capability levels.

Table 5-3. Summary Assessment Record in BOOTSTRAP

Capability Level	Description	QSA	QSA Cumm	Pass Threshold	Pass Threshold Cumm	Assessment Result	Assessment Result Cumm	Diff	Diff Cumm
CL5	Optimizing	53	201	43	162	52	197	1	4
CL4	Managed	27	148	22	119	26	145	1	3
CL3	Defined	81	121	65	97	80	119	1	2
CL2	Repeated	40	40	32	32	39	39	1	1
CL1	Initial	0	0	0	0	0	0	0	0

Figure 5-3 is a graphical representation of the process profile of projectX in BOOTSTRAP.

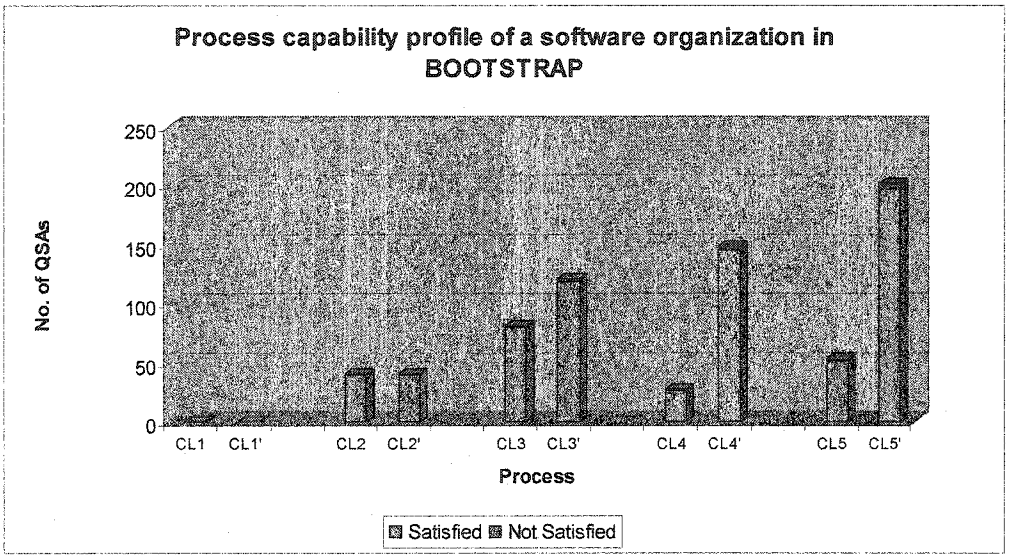

Figure 5-3 Process Capability Profile of a Software Project in BOOTSTRAP

5.2.4 ISO/IEC 15504 Assessment

According to the ISO/IEC 15504 assessments, see Table 5-4, projectX practice has achieved ISO/IEC 15504 Level 4 capability maturity. Figure 5-4 shows that projectX practice strengths are in the areas of production and supplier process practices. Its weaknesses are in specific areas such as "Perform joint audits and reviews (CUS 4)", "Provide customer service (CUS 7)", "Assess customer satisfaction (CUS 8)", "Maintain system and software (ENG 7)", "Improve the process (ORG 3)", and "Provide software engineering environment (ORG 6)". CUS 4, CUS 8, and ORG 3 all have a value of zero. It is an indication that processes have been implemented but have not been executed yet, or processes need improvement. These processes were all applicable because the assessment algorithm would have handled the situation when a process was not applicable to the organization. For improvement, the organization might need to look into these areas first to get it off the ground and let CPI build up its capability level. Generally speaking, there were six areas that required action, which constituted only about 17% (6/35 * 100) of the total process areas.

Table 5-4. Summary Assessment Record in ISO/IEC 15504

Category	Process ID	Process	Capability
CUS_1	1	Acquire software product	5
CUS_2	2	Establish contract	5
CUS_3	3	Identify customer needs	5
CUS_4	4	Perform joint audits and reviews	0
CUS_5	5	Package, deliver and install software	5
CUS_6	6	Support operation of software	5
CUS_7	7	Provide customer service	1
CUS_8	8	Assess customer satisfaction	0
ENG_1	1	Develop system requirements	5
ENG_2	2	Develop software requirements	5
ENG_3	3	Develop software design	5
ENG_4	4	Implement software design	5
ENG_5	5	Integrate and test software	5
ENG_6	6	Integrate and test system	5
ENG_7	7	Maintain system and software	1
PRO_1	1	Plan project life cycle	5
PRO_2	2	Establish project plan	5
PRO_3	3	Build project teams	5
PRO_4	4	Manage requirements	5
PRO_5	5	Manage quality	5
PRO_6	6	Manage risks	5
PRO_7	7	Manage resources and schedule	5
PRO_8	8	Manage subcontractors	5

Category	Process ID	Process	Capability
SUP_1	1	Develop documentation	5
SUP_2	2	Perform configuration management	5
SUP_3	3	Perform quality assurance	5
SUP_4	4	Perform problem resolution	5
SUP_5	5	Perform peer reviews	5
ORG_1	1	Engineer the business	5
ORG_2	2	Define the process	5
ORG_3	3	Improve the process	0
ORG_4	4	Perform training	5
ORG_5	5	Enable reuse	5
ORG_6	6	Provide software engineering environment	1
ORG_7	7	Provide work facilities	5

Figure 5-4 is a graphical representation of the process profile of projectX in ISO/IEC 15504.

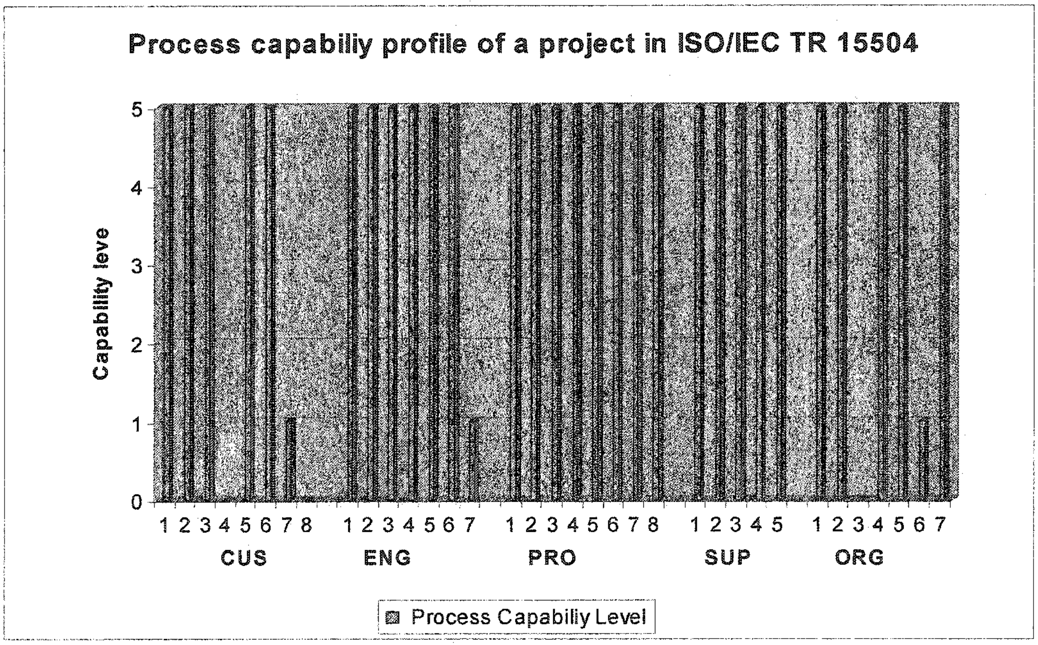

Figure 5-4 Process Capability Profile of a Project in ISO/IEC 15504

5.2.5 SEPRM Assessment

According to the SEPRM assessment, see Table 5-5, projectX practice has achieved SEPRM Level 4.8. Unlike other models that either round the capability level to one significant digit or rounding it to the nearest quartile (BOOTSTRAP), SEPRM rounds the capability level rounds the capability level to one decimal point. This essentially provided a range of 10 values between the capability levels. The strengths the projectX practice capability levels according to this model are obviously the management practice processes with the development practice processes coming in a close second. The organization practice processes is projectX practice weaknesses. Fortunately, there are not many processes within this category. The organization can work on two of its weakest process areas, which are "Organization process improvement (Category 1.2)" and "Customer relations (Category 1.3)".

Table 5-5. Summary Assessment Record in SEPRM

ID	Category/Process	Process Capability Level
1_1_1	Organization definition	5
1_1_2	Project organization	4.2
1_2_1	Organization process definition	5
1_2_2	Organization process improvement	1.4
1_3_1	Customer relations	3.1
1_3_2	Customer support	4.6
1_3_3	Software/system delivery	5
1_3_4	Service evaluation	4.2
2_1_1	Software engineering modeling	5
2_1_2	Reuse methodologies	5
2_1_3	Technology innovation	5
2_2_1	Development process definition	5
2_2_2	Requirement analysis	5
2_2_3	Design	5
2_2_4	Coding	5

ID	Category/Process	Process Capability Level
2_2_5	Module testing	5
2_2_6	Integration and system testing	5
2_2_7	Maintenance	4.5
2_3_1	Environment	4.3
2_3_2	Facilities	5
2_3_3	Development support tools	5
2_3_4	Management support tools	5
3_1_1	SQA process definition	5
3_1_2	Requirement review	5
3_1_3	Design review	5
3_1_4	Code review	5
3_1_5	Module testing audit	5
3_1_6	Integration and system testing audit	5
3_1_7	Maintenance audit	5
3_1_8	Audit and inspection	5
3_1_9	Peer review	5
3_1_10	Defect control	5
3_1_11	Subcontractor's quality control	5
3_2_1	Project plan	5
3_2_2	Project estimation	5
3_2_3	Project risk avoidance	5
3_2_4	Project quality plan	5
3_3_1	Process management	5
3_3_2	Process tracking	5
3_3_3	Configuration management	5
3_3_4	Change control	5
3_3_5	Process review	5
3_3_6	Intergroup coordination	5
3_4_1	Requirement management	5
3_4_2	Contract management	5
3_4_3	Subcontractor management	5
3_4_4	Purchasing management	5
3_5_1	Documentation	5
3_5_2	Process database/library	5

ID	Category/Process	Process Capability Level
3_6_1	Staff selection and allocation	5
3_6_2	Training	5

Figure 5-5 is a graphical representation of the organization process subsystem of projectX in SEPRM.

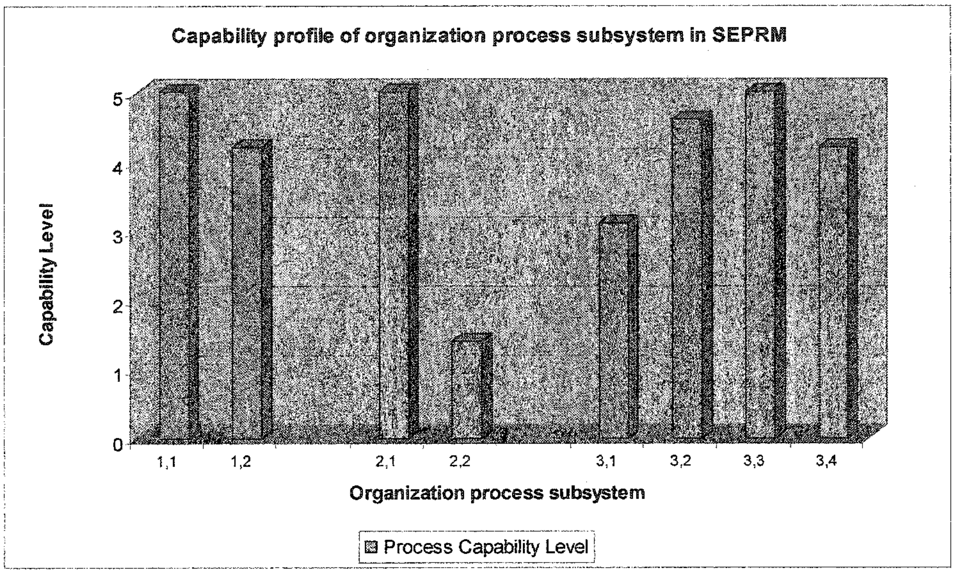

Figure 5-5 Capability Profile of an Organization Process Subsystem in SEPRM

Figure 5-6 is a graphical representation of the development process subsystem of projectX in SEPRM.

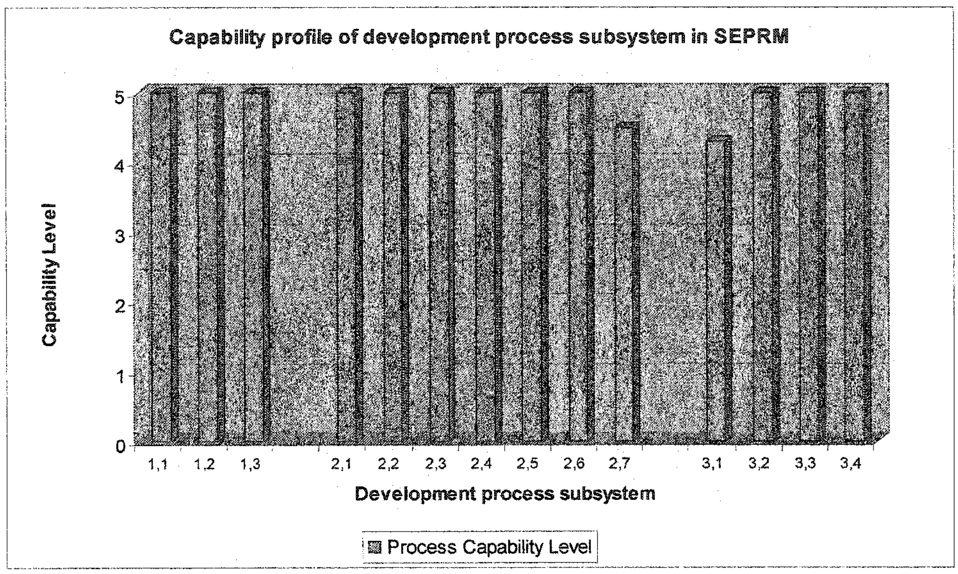

Figure 5-6 Capability Profile of Development Process Subsystem in SEPRM

Figure 5-7 is a graphical representation of the management process subsystem of projectX in SEPRM.

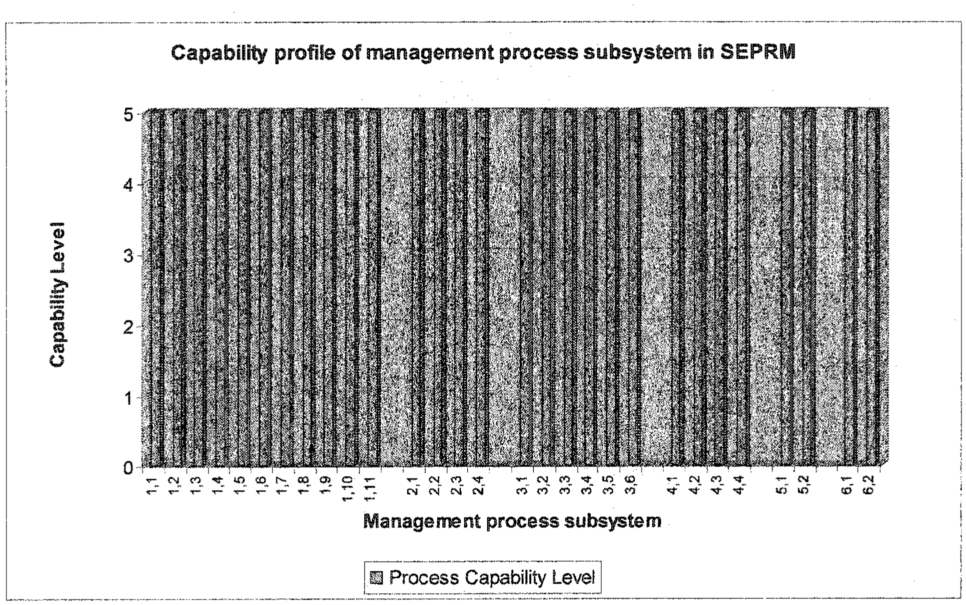

Figure 5-7 Capability Profile of Management Process Subsystem in SEPRM

5.2.6 Overall Assessment Evaluation

The following is a summary of projectX practice capability levels based on the five given models.

Table 5-6. Summary of the Organization's Best Practice Capability Levels

Models	Capability Level (CL)	CL Description
CMM	4	Managed
ISO 9001	Failed	Failed
BOOTSTRAP	5	Optimizing
ISO/IEC 15504	4	Predictable
SEPRM	4.8	Effective

The recurring theme from all the assessments, the organization needs to work on its CPI and Customer Relationship Management. Overall, projectX practices do fairly well in all models. The organization can focus improving its current status on ISO 9001 model first to achieve a better overall capability level with respect to the overall standing in all models. Also, ISO 9001 is the simplest and most common model known to most industries. Then, the organization can pick and choose the model that best reflect the way the organization is doing business.

5.3 Benchmarking

In this section, summaries of the project, SEP model, and benchmark capability are presented in a table form, followed by a graph representation of the summary. The value shown in the summaries and graphs are processes that have satisfied the given standards, passed for the given standards, or achieved the capability level for the given standards. Refer to the beginning of this thesis, "LIST OF ABBREVIATIONS" for acronym referenced and used in the benchmark summaries.

5.3.1 CMM Benchmarks

According to the CMM benchmark summary, see Table 5-7, projectX practice is below average in the "Process Change Management (KPA 5.3)". The organization is not that much ahead of the competitors in the areas of "Requirement Management (KPA 2.1)", "Organization process focus (KPA 3.1)", and "Peer reviews (KPA 3.7)". ProjectX practice is well ahead of its competitors in areas such as "Software project planning (KPA 2.2)", "Software project tracking and oversight (KPA 2.3)", "Software subcontract management (KPA 2.4), "Software quality assurance (KPA 2.5), "Organization process definition (KPA 3.2)", "Integrated software management (KPA 3.4)", " quantitative process management (KPA 4.1)", and "Defect prevention (KPA 5.1)".

Table 5-7. Summary Benchmark Record in CMM

KPA	Description	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
1,1	Initial	0	0	0	0	0	0
2,1	Requirement management	3	3	3	2	0	0
2,2	Software project planning	15	15	15	12	2	0
2,3	Software project tracking and oversight	13	13	13	9	0	0
2,4	Software subcontract management	13	13	13	9	0	0
2,5	Software quality assurance	8	8	8	5	0	0
2,6	Software configuration management	10	10	10	8	1	0
3,1	Organization process focus	4	7	7	4	0	0
3,2	Organization process definition	6	6	6	3	0	0
3,3	Training program	6	6	6	5	0	0
3,4	Integrated software management	11	11	11	7	0	0
3,5	Software product engineering	10	10	10	7	1	0
3,6	Intergroup coordination	7	7	7	5	0	0
3,7	Peer reviews	3	3	3	2	0	0
4,1	Quantitative process management	7	7	7	3	0	0
4,2	Software quality management	5	5	5	3	0	0
5,1	Defect prevention	8	8	8	4	0	0
5,2	Technology change management	8	8	8	4	0	0
5,3	Process change management	4	10	10	4	0	0

Figure 5-8 is a graphical representation of the table summary above. This graph shows the benchmarked capability for each process at each capability level.

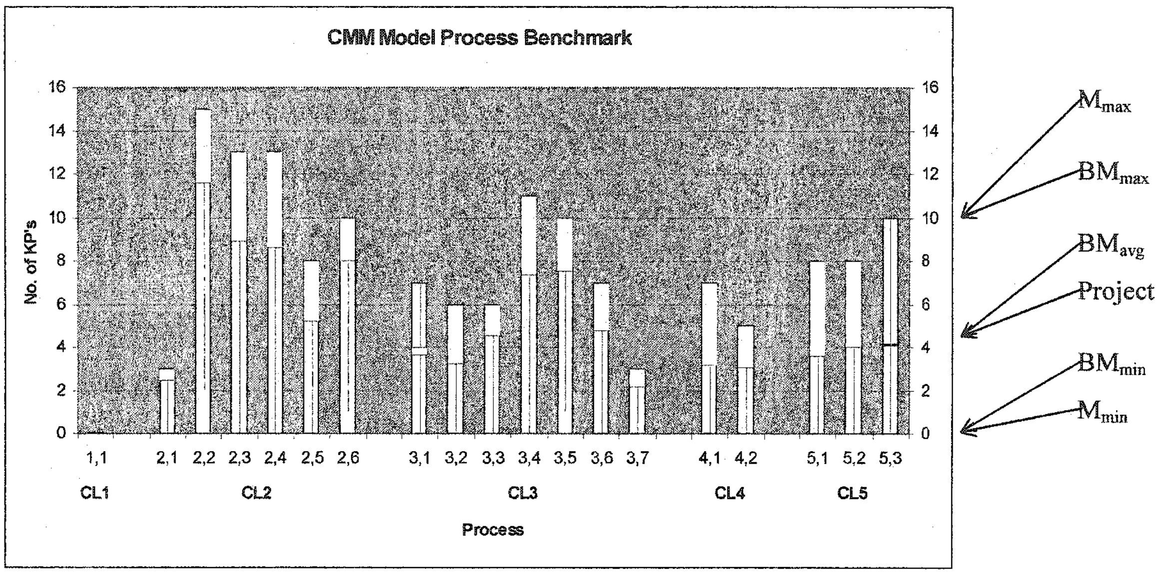

Figure 5-8 CMM Model Process Benchmark

5.3.2 ISO 9001 Benchmarks

According to the ISO 9001 benchmark summary, see Table 5-8, projectX has no practice process below average. ProjectX practice is ahead of it competitors in the process areas of "Process control (MTA 3.2)", and "Design and development control (MTA 3.3)". The competitors are close behind on all the other process areas. This mean the organization would have to ensure that the majority of its process practices are continuously maintained at the status quo, so projectX practice would not fall behind its competitors.

Table 5-8. Summary Benchmark Record in ISO 9001

Main Topic Area	Description	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
1,1	Management responsibility	14	15	15	11	1	0
1,2	Quality system	7	7	7	5	0	0

Main Topic Area	Description	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
1,3	Document and data control	8	8	8	6	1	0
1,4	Internal quality audits	6	6	6	3	0	0
1,5	Corrective and preventive action	6	6	6	3	0	0
1,6	Quality system records	7	7	7	4	0	0
1,7	Training	4	4	4	3	0	0
2,1	Product management	4	4	4	3	0	0
2,2	Control of customer-supplied product	4	4	4	3	0	0
2,3	Purchasing	8	8	8	6	1	0
2,4	Handling, storage, packaging, preservation, and delivery	9	9	9	6	0	0
2,5	Control of nonconforming product	6	6	6	4	0	0
3,1	Contract reviews	9	9	9	7	0	0
3,2	Process control	23	23	23	16	1	0
3,3	Design and development control	29	30	30	21	1	0
3,4	Inspection and testing	11	11	11	8	0	0
3,5	Inspection and test status	2	2	2	1	0	0
3,6	Control of inspection, measuring, and test equipment	12	12	12	9	4	0
3,7	Statistical techniques	2	2	2	1	0	0
3,8	Servicing and software maintenance	4	4	4	3	0	0

Figure 5-9 is a graphical representation of the table summary above. This graph provides representation at the process level and not at the project level. Graphing the process level create a range with intervals for better benchmark result evaluation and interpretation.

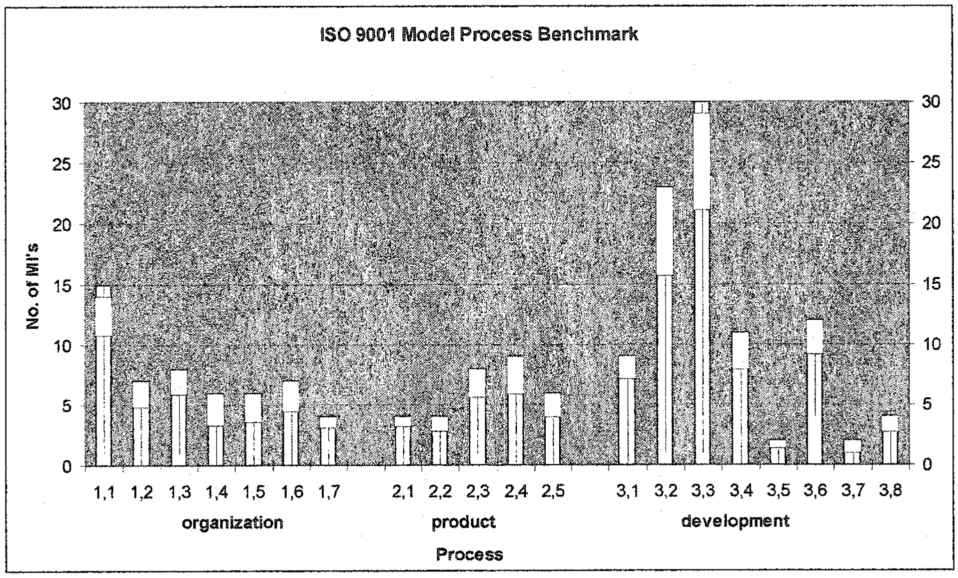

Figure 5-9 ISO 9001 Model Process Benchmark

5.3.3 BOOTSTRAP Benchmarks

According to the BOOTSTRAP benchmark summary, see Table 2-1, the organization has no practice that is below BOOTSTRAP benchmark average. ProjectX practice strength is at BOOTSTRAP Level 3 and Level 5. This means projectX practice is weak in BOOTSTRAP Level 2, and Level 4. Incidentally, this means Level 2 and Level 4 are where the competitor's strength is, based on the height of the white blocks (Level 2 and Level 4) relative to the other white blocks (Level 3 and Level 5).

Table 5-9. Summary Benchmark Record in BOOTSTRAP

CL	Description	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
PCL		5.00	5.00	5.50	2.53	1.00	0
CL1	Initial	0	0	0	0	0	0
CL1'	Initial	0	0	0	0	0	0
CL2	Repeated	39	40	40	29	5	0
CL2'	Repeated	39	40	40	29	5	0
CL3	Defined	80	81	81	51	3	0
CL3'	Defined	119	121	121	79	8	0
CL4	Managed	26	27	27	11	0	0
CL4'	Managed	145	148	148	90	8	0
CL5	Optimizing	52	53	53	27	5	0
CL5'	Optimizing	197	201	201	118	13	0

Figure 5-10 is a graphical representation of the table summary above. This graph illustrates the benchmark results for each capability level and the accumulated capability level benchmarked process assessment result.

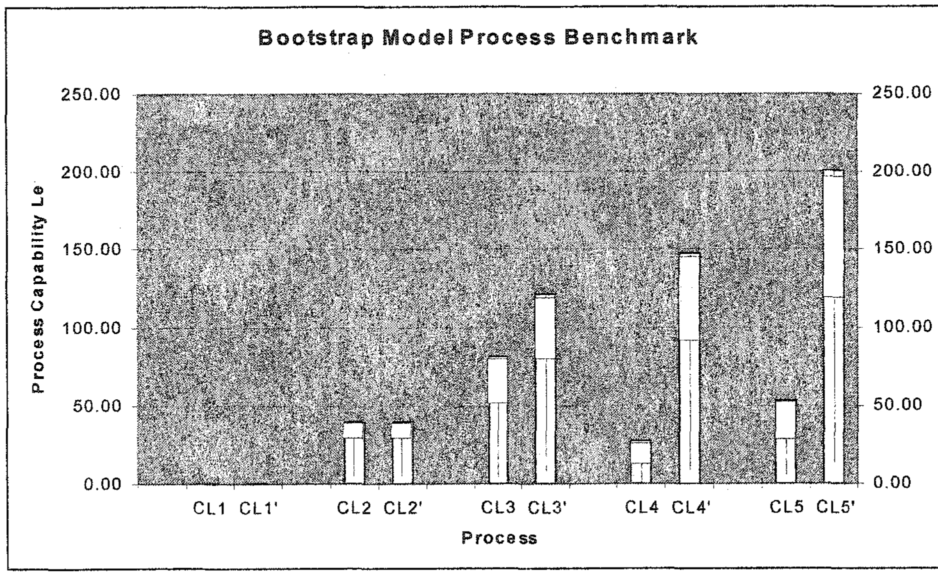

Figure 5-10 BOOTSTRAP Model Process Benchmark

5.3.4 ISO/IEC 15504 Benchmarks

According to the ISO/IEC TR 15504 benchmark summary, see Table 5-10. ProjectX practice is below average in a number of process areas. These areas are "Perform joint audits and reviews (CUS 4)", "Provide customer service (CUS 7)", "Assess customer satisfaction (CUS 8)", "Maintain system and software (ENG 7)", "Improve the process (ORG 3)" and "Provide software engineering environment (ORG 6)". It is interesting to see that "Perform joint audits and reviews (CUS 4)" average is the lowest average of the entire process category. This means the competitors are just as bad in the process category. Furthermore, "Assess customer satisfaction (CUS 8)" and "Improve the process (ORG 3)" clearly indicates that projectX practice in these process categories are either non existent or extremely weak. In addition, the organization must be aware that for "Assess customer satisfaction (CUS 8)", competitors' average is relatively high for projectX practice to catch up. Looking at the white blocks, the majority of projectX practice processes are at the highest capability level as per ISO/IEC 15504 assessment.

Table 5-10. Summary Benchmark Record in ISO/IEC 15504

Category	Description	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
CUS_1	Acquire software product	5	5	5	2	0	0
CUS_2	Establish contract	5	5	5	2	0	0
CUS_3	Identify customer needs	5	5	5	3	1	0
CUS_4	Perform joint audits and reviews	0	5	3	1	0	0
CUS_5	Package, deliver and install software	5	5	5	2	0	0
CUS_6	Support operation of software	5	5	5	3	0	0
CUS_7	Provide customer service	1	5	5	3	0	0
CUS_8	Assess customer satisfaction	0	5	5	2	0	0
ENG_1	Develop system requirements	5	5	5	3	1	0
ENG_2	Develop software requirements	5	5	5	2	0	0
ENG_3	Develop software design	5	5	5	3	0	0
ENG_4	Implement software design	5	5	5	3	0	0
ENG_5	Integrate and test software	5	5	5	3	0	0
ENG_6	Integrate and test system	5	5	5	3	0	0
ENG_7	Maintain system and software	1	5	5	3	1	0
PRO_1	Plan project life cycle	5	5	5	3	0	0
PRO_2	Establish project plan	5	5	5	2	0	0
PRO_3	Build project teams	5	5	5	3	1	0
PRO_4	Manage requirements	5	5	5	3	0	0
PRO_5	Manage quality	5	5	5	2	0	0
PRO_6	Manage risks	5	5	5	2	0	0
PRO_7	Manage resources and schedule	5	5	5	3	0	0
PRO_8	Manage subcontractors	5	5	5	3	0	0
SUP_1	Develop documentation	5	5	5	3	0	0
SUP_2	Perform configuration management	5	5	5	4	0	0
SUP_3	Perform quality assurance	5	5	5	2	0	0
SUP_4	Perform problem resolution	5	5	5	2	0	0
SUP_5	Perform peer reviews	5	5	5	2	0	0
ORG_1	Engineer the business	5	5	5	2	0	0

Category	Description	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
ORG_2	Define the process	5	5	5	1	0	0
ORG_3	Improve the process	0	5	3	1	0	0
ORG_4	Perform training	5	5	5	3	0	0
ORG_5	Enable reuse	5	5	5	1	0	0
ORG_6	Provide software engineering environment	1	5	3	2	0	0
ORG_7	Provide work facilities	5	5	5	3	0	0

Figure 5-11 is a graphical representation of the table summary above. This graph provides a quick overview of benchmark results for the five ISO/IEC 15504 process categories: Customer-supplier processes, Engineering processes, Project processes, Support processes, and Organization processes.

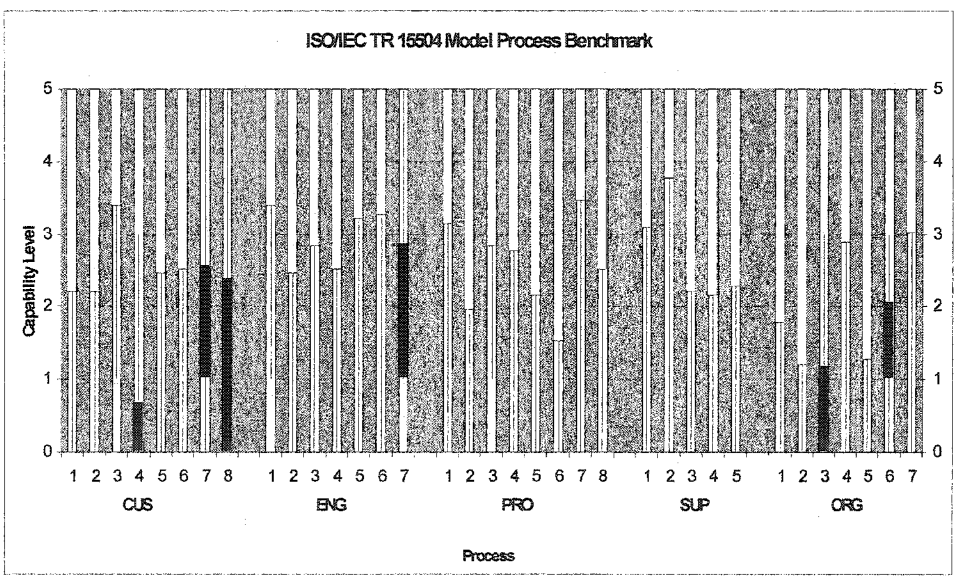

Figure 5-11 ISO/IEC 15504 Model Process Benchmark

5.3.5 SEPRM Benchmarks

According to the SEPRM benchmark summary, see Table 5-11, the organization has one weak area in the organization process subsystem, "Organization process improvement

(Organization 2.2)". The Organization process subsystem's "Project organization (1.2)" and "Customer support (3.2)" are process categories indicating that projectX practice should be concerned because the height of the white blocks is relatively short. This means the competitors are not far behind and potentially catching up to projectX practice.

Table 5-11. Summary Benchmark Record in SEPRM

Process ID	Category / Process	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
1_1_1	Organization definition	5.0	5.0	5.0	3.5	1.0	0.0
1_1_2	Project organization	4.2	5.0	4.3	3.2	1.2	0.0
1_2_1	Organization process definition	5.0	5.0	5.0	2.6	0.2	0.0
1_2_2	Organization process improvement	1.4	5.0	3.8	1.7	0.0	0.0
1_3_1	Customer relations	3.1	5.0	3.7	2.4	0.6	0.0
1_3_2	Customer support	4.6	5.0	4.7	3.1	0.0	0.0
1_3_3	Software/system delivery	5.0	5.0	5.0	3.4	0.5	0.0
1_3_4	Service evaluation	4.2	5.0	4.3	2.9	0.2	0.0
2_1_1	Software engineering modeling	5.0	5.0	5.0	2.5	0.0	0.0
2_1_2	Reuse methodologies	5.0	5.0	5.0	1.6	0.0	0.0
2_1_3	Technology innovation	5.0	5.0	5.0	2.7	0.0	0.0
2_2_1	Development process definition	5.0	5.0	5.0	3.5	0.2	0.0
2_2_2	Requirement analysis	5.0	5.0	5.0	3.2	0.0	0.0
2_2_3	Design	5.0	5.0	5.0	3.1	0.0	0.0
2_2_4	Coding	5.0	5.0	5.0	3.1	0.8	0.0
2_2_5	Module testing	5.0	5.0	5.0	3.4	0.0	0.0
2_2_6	Integration and system testing	5.0	5.0	5.0	2.4	0.1	0.0
2_2_7	Maintenance	4.5	5.0	4.5	3.1	0.8	0.0
2_3_1	Environment	4.3	5.0	4.4	3.0	0.0	0.0
2_3_2	Facilities	5.0	5.0	5.0	3.8	1.8	0.0
2_3_3	Development support tools	5.0	5.0	5.0	2.5	0.2	0.0
2_3_4	Management support tools	5.0	5.0	5.0	2.0	0.0	0.0
3_1_1	SQA process definition	5.0	5.0	5.0	2.9	0.0	0.0
3_1_2	Requirement review	5.0	5.0	5.0	3.0	0.0	0.0
3_1_3	Design review	5.0	5.0	5.0	2.6	0.0	0.0

Process ID	Category / Process	Project Performance	M_{max}	BM_{max}	BM_{avg}	BM_{min}	M_{min}
3_1_4	Code review	5.0	5.0	5.0	2.5	0.0	0.0
3_1_5	Module testing audit	5.0	5.0	5.0	1.9	0.0	0.0
3_1_6	Integration and system testing audit	5.0	5.0	5.0	2.9	0.0	0.0
3_1_7	Maintenance audit	5.0	5.0	5.0	1.9	0.0	0.0
3_1_8	Audit and inspection	5.0	5.0	5.0	2.3	0.0	0.0
3_1_9	Peer review	5.0	5.0	5.0	2.7	0.0	0.0
3_1_10	Defect control	5.0	5.0	5.0	2.8	0.0	0.0
3_1_11	Subcontractor's quality control	5.0	5.0	5.0	2.8	0.0	0.0
3_2_1	Project plan	5.0	5.0	5.0	3.5	0.9	0.0
3_2_2	Project estimation	5.0	5.0	5.0	3.7	0.6	0.0
3_2_3	Project risk avoidance	5.0	5.0	5.0	2.1	0.1	0.0
3_2_4	Project quality plan	5.0	5.0	5.0	2.8	0.0	0.0
3_3_1	Process management	5.0	5.0	5.0	2.3	0.0	0.0
3_3_2	Process tracking	5.0	5.0	5.0	2.7	0.0	0.0
3_3_3	Configuration management	5.0	5.0	5.0	3.8	0.6	0.0
3_3_4	Change control	5.0	5.0	5.0	3.0	0.0	0.0
3_3_5	Process review	5.0	5.0	5.0	2.3	0.1	0.0
3_3_6	Intergroup coordination	5.0	5.0	5.0	3.1	0.0	0.0
3_4_1	Requirement management	5.0	5.0	5.0	3.8	1.6	0.0
3_4_2	Contract management	5.0	5.0	5.0	3.2	0.0	0.0
3_4_3	Subcontractor management	5.0	5.0	5.0	2.9	0.0	0.0
3_4_4	Purchasing management	5.0	5.0	5.0	2.9	0.0	0.0
3_5_1	Documentation	5.0	5.0	5.0	3.5	0.2	0.0
3_5_2	Process database/library	5.0	5.0	5.0	2.1	0.0	0.0
							0.0
3_6_1	Staff selection and allocation	5.0	5.0	5.0	4.2	2.0	0.0
3_6_2	Training	5.0	5.0	5.0	3.4	0.0	0.0

Figure 5-12 is a graphical representation of the table summary above in the organization process subsystem.

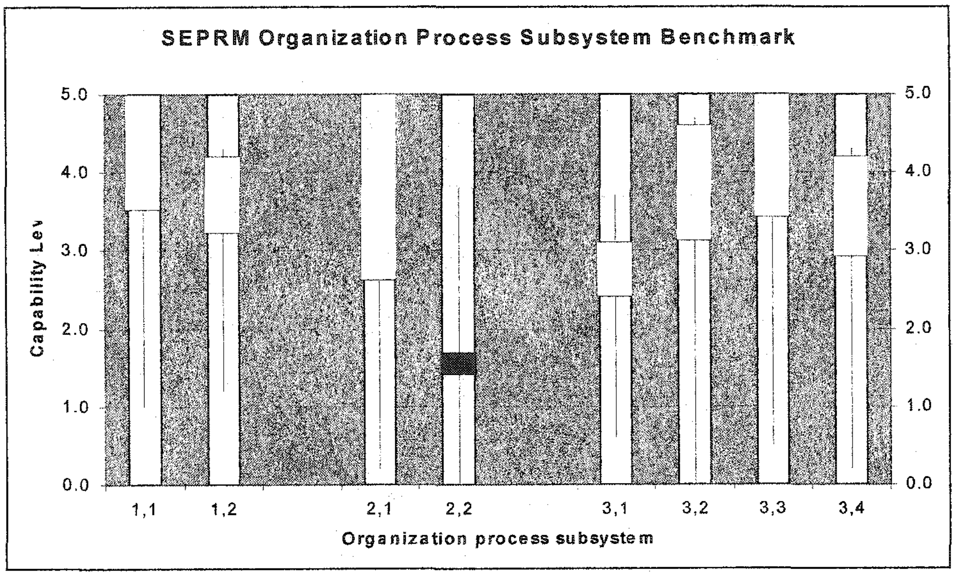

Figure 5-12 SEPRM Organization Process Subsystem Benchmark

Figure 5-13 is a graphical representation of the table summary above in the development process subsystem.

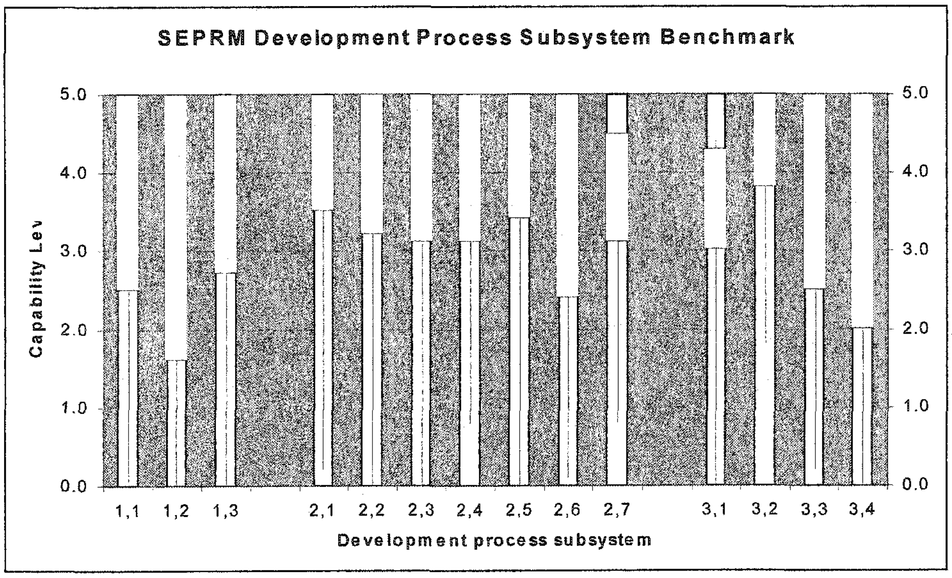

Figure 5-13 SEPRM Development Process Subsystem Benchmark

Figure 5-14 is a graphical representation of the table summary above in the management process subsystem.

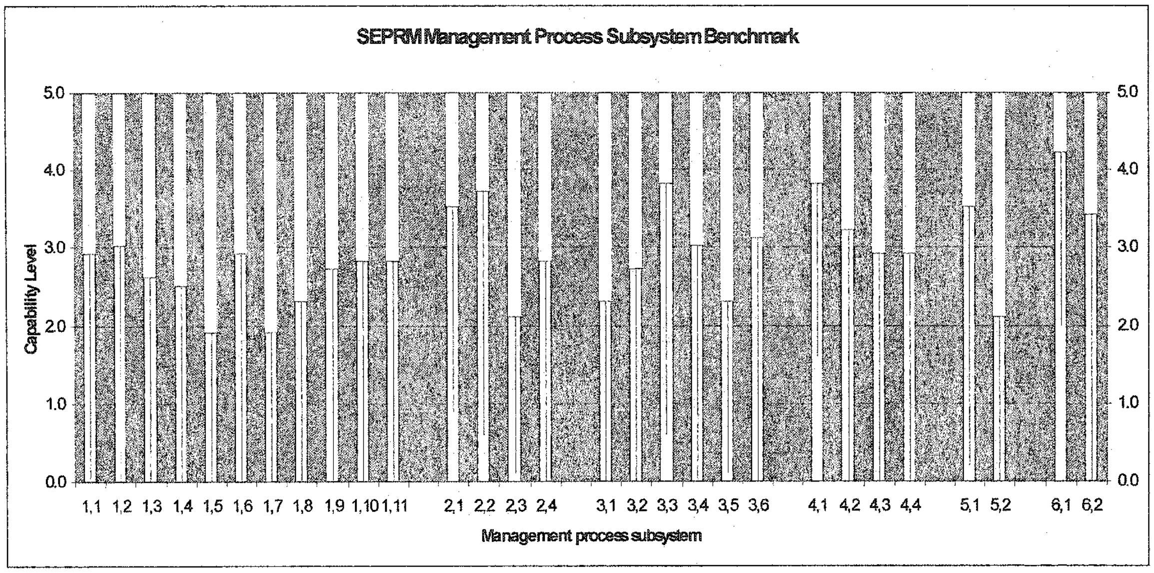

Figure 5-14 SEPRM Management Process Subsystem Benchmark

5.4 Comparative Analysis of Benchmarking Results

This section provides the comparative analysis of the benchmarking results between all the benchmarked projects.

5.4.1 Benchmarking Result Summary

Table 5-12 Summary Benchmark Database SEP CL

CMM CL	Percent	Count
1	56.25%	9
2	18.75%	3
3	6.25%	1
4	6.25%	1
5	12.50%	2
Total	100.00%	16
ISO 9001 CL	Percent	Count
Passed	18.75%	3
Failed	81.25%	13
Total	100.00%	16
BOOTSTRAP CL	Percent	Count
1	50.00%	8
2	18.75%	3
3	6.25%	1
4	6.25%	1
5	18.75%	3
Total	100.00%	16
ISO15504 CL	Percent	Count
0	12.50%	2
1	12.50%	2
2	56.25%	9
3	12.50%	2
4	6.25%	1
5	0.00%	0
Total	100.00%	16
SEPRM CL	Percent	Count
0	12.50%	2
1	0.00%	0
2	31.25%	5
3	37.50%	6
4	18.75%	3
5	0.00%	0
Total	100.00%	16

Figure 5-15 shows the distribution of the benchmark CMM capability level for all the projects. The distribution based on the five CMM levels.

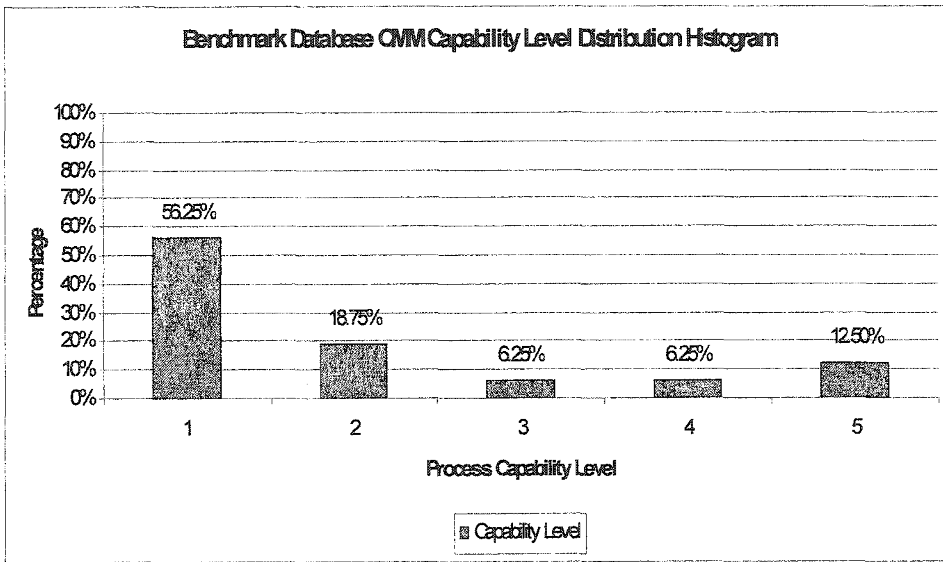

Figure 5-15 CMM Projects Capability Level Histogram

Figure 5-16 shows the distribution of the benchmark ISO 9001 capability level for all the projects. Since ISO 9001 only acknowledges satisfaction there are only two bars indicating "Pass" for satisfy, and "Fail" for not satisfied.

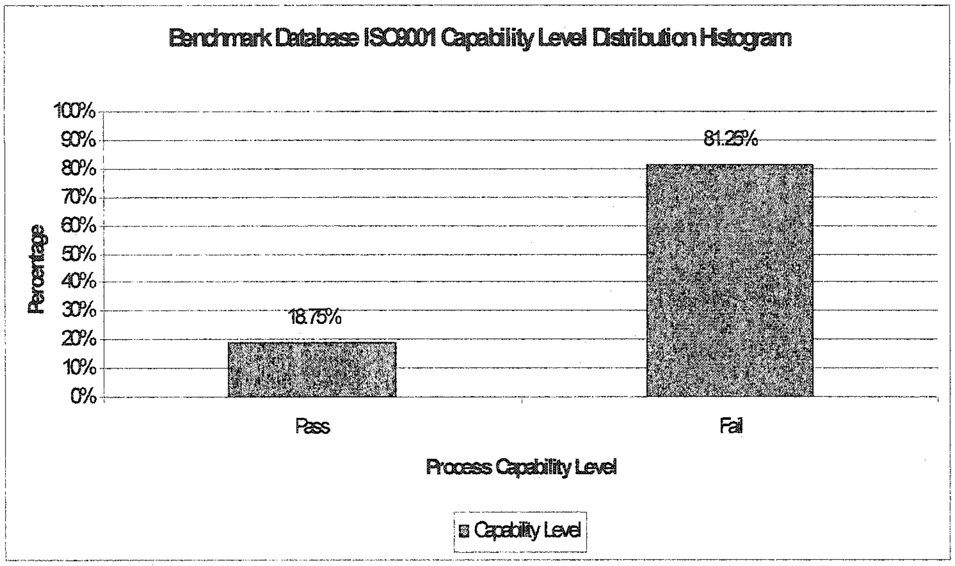

Figure 5-16 ISO 9001 Projects Capability Level Histogram

Figure 5-17 shows the distribution of the benchmark BOOTSTRAP capability level for all the projects. Similar to CMM, the projects are distributed according to their capability levels base on the BOOTSTRAP Model.

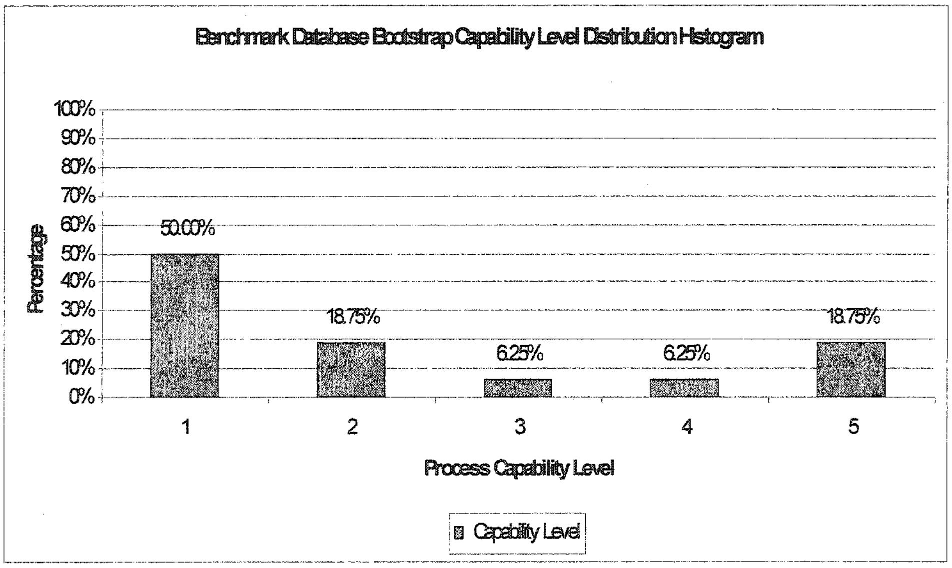

Figure 5-17 BOOTSTRAP Projects Capability Level Histogram

Figure 5-18 shows the distribution of the benchmark ISO/IEC 15504 capability level for all the projects. This model capability levels begin with level 0, indicating the process is incomplete. Level 5 is an indication that process is mature enough for change as part of CPI effort.

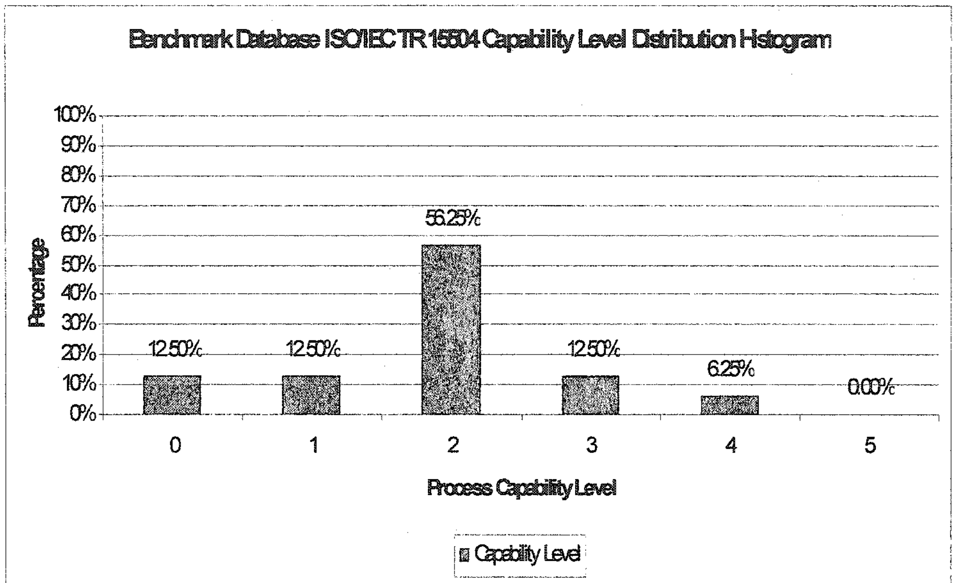

Figure 5-18 ISO/IEC 15504 Projects Capability Level Histogram

Figure 5-19 shows the distribution of the benchmark SEPRM capability level for all the projects. SEPRM's capability levels is similar to ISO/IEC 15504, where level 0 is defined as "incomplete", while level 5 is defined as "refining".

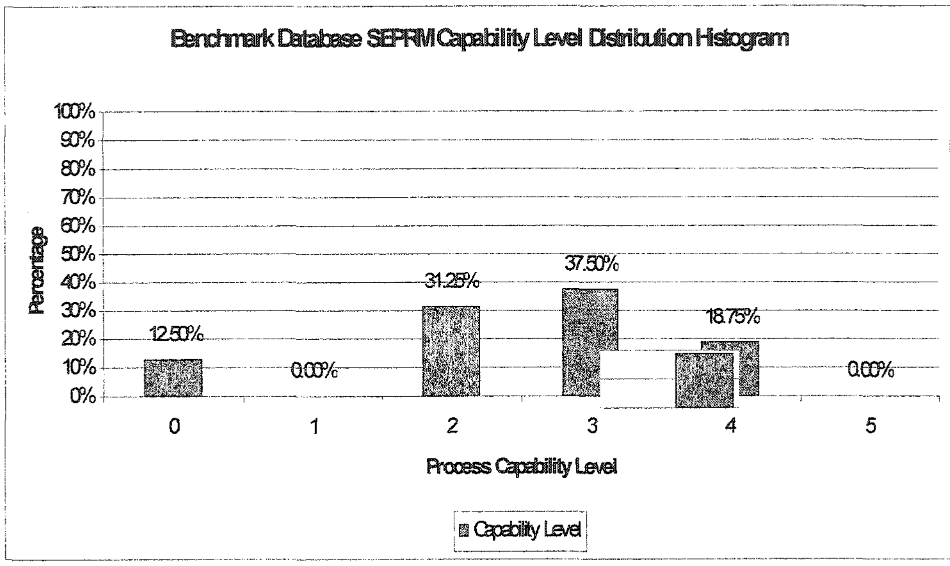

Figure 5-19 SEPRM Projects Capability Level Histogram

5.4.2 Benchmarking Result Analysis

Figure 5-15 to Figure 5-19 show the distribution of the process capability level at the project level for all the models. These distributions provide supplemental quantitative information to the benchmark information. In addition to answering the questions "Are the competitors ahead or behind the current projectX practice?" and "What is the magnitude or distance of the measurement from the current projectX practice state to the competitors?". The supplemental quantitative information answers the question "How many competitors are at each of the level?". This information is provided at the project level instead of at the process level because it provides a big picture of the project so executives can make high level CPI-related strategic decisions. The benchmark process level information is good for middle management, quality team, and practitioners to execute the required action items translated from strategic action items, as part of CPI. Table 5-6 shows that projectX practice is doing relatively well for CMM. Only 12.5% of the benchmark projects are ahead of projectX practice at capability level 5. Figure 5-16 shows that 20% of the competitors are ISO 9001 registered, ahead of the current projectX practice. The current practice is among the top 20% of the BOOTSTRAP benchmark projects, at capability level 5. Figure 5-18 and Figure 5-19 show that no project is at capability level 5. This means there is an opportunity to get ahead of the competitors. A more important observation is that more than half of the benchmark projects, approximately 60% for models, are at the heel of projectX practice. In addition, projectX practice has to concern with and competes with the benchmark projects sharing the same capability level. It is impossible to determine the projectX practice, at ISO/IEC 15504 capability level 4 maturities without a more detailed analysis at the process level.

On the other hand, projectX practice for SEPRM capability level is 4.8 providing slightly more information than ISO/IEC 15504. This information indicates the projectX practice is at the high end of the current SEPRM capability level with the rest of the benchmark projects at the low end of the capability level 4 range. Similar to ISO/IEC 15504, it requires process level analysis for further details on competitors standing.

5.4.3 Ranking Analysis

The percentile ranking analysis, see Figure 5-20, provides an overall comparative analysis at the project level.

Table 5-13. Summary of Software Engineering Project CL Benchmark Percentile Ranking

	BOOTSTRAP	CMM	ISO15504	ISO 9001	SEPRM
Project CL	5	4	4	175	4.8
Percentile	86.60%	86.60%	100.00%	-20.00%	100.00%

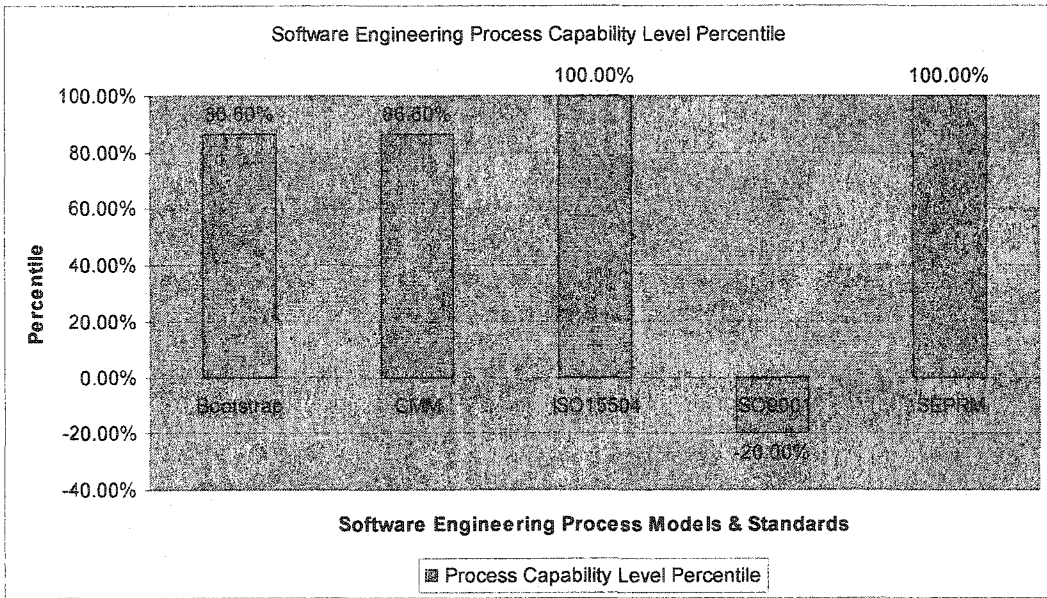

Figure 5-20 ProjectX's Practice Software Engineering PCL Percentile Ranking

Figure 5-20 provides more information regarding the overall standing of projectX's practice standing and ranking among benchmark projects (competitors). This chart is produced using the project level capability information to determine projectX's practice standing among all the competitors for the given models.

5.5 Discussions

It is possible to achieve a relatively high project capability level and yet not achieve satisfying performance. The performance refers to the validation by the customer feedback on the organization's product or service. The main reason is that most of the standards do not require all practices or activities be satisfied in order to achieve a particular capability level. Usually the requirements to achieve a particular capability level require only to satisfy or to pass predetermined thresholds of activities for a category or subsystem. The assumption is that relevant processes for projectX practices exist and are accounted for for benchmarking.

There are situations where a process is not considered for assessment because the organization has determined it is not applicable for projectX practice business. This process that is not applicable, might be applicable according to the competitors who can be identified from benchmarking. To achieve this, the benchmarking system will have to be modified and take into account the Not Applicable (N/A) practices. Alternatively, an organization can provide a process to benchmark by setting the process capability level to the lowest maturity and see how the competitors are doing in that particular process.

Let us look at an example by picking a projectX process that is relatively weak and not the "Not Applicable" case. Figure 5-11 illustrates that projectX practice is weak in "Assess customer satisfaction (CUS 8)", based on the ISO/IEC 15504 model. According to the benchmark of projectX practice, projectX practice is at capability level 0 while the competitors' average capability level is at 2.5, with at least one competitor at capability level 5. Even though projectX practice is ranked among the best according to Figure 5-20, there are still potentially significant weaknesses within the organization's best practice. Obviously, the final validation will be the customer. It is an ironic that the weak activity chosen for this example is "Assess customer satisfaction (CUS 8)". The point of this example is that competitors do consider this process significant. One way to verify this result is by eliminating non-similar projects to the organization base on OU.

- Potential New Market Penetration based on Benchmark Results

  Since projectX fairs well in ISO/IEC 15504, see Table 5-6, it might do well in the European market where the ISO/IEC 15504 standards are more prominently used. Greater assurance is achieved by looking at Figure 5-11, Figure 5-18 and Figure 5-20. Respectively, the benchmark figures say even though projectX is not as mature capability wise, according to the ISO/IEC 15504 standards, it is competing only with 6.25% of the competitors at the same level while overall it is in the top 0% percentile. A true benchmark assessment will be to benchmark projects from Europe. From the overall ranking, projectX might fair well in most markets due to its high capability levels. Bearing in mind that projectX is the organization procedural best practice. A real project should be benchmarked and used if it is more suited for the intended market sector or industry.

- Limited Budget Spending Allocation Based on Benchmark Results

If projectX has only a limited process improvement budget and can only improve on one process, which process should it try to improve? Take CMM as an example, projectX could try to improve on either its "Organization process focus (CL 3.1)" or "Process change management (CL 5.3)" based purely on assessment results, see Figure 5-1. Using the additional benchmark information from Figure 5-8, it is obvious that the process improvement budget is best spent on CMM "Process change management (CL 5.3)" because it is the only process capability level that is below the benchmark average. This is an opportunity to close or narrow down the gap. Investment on "Organization process focus (CL 3.1)" would narrow the organization's benchmark gap. However, it would provide competitors with an opportunity to widen the "Process change management (CL 5.3)" gap.

- Limited Budget Spending Allocation Based on Benchmark Reference Model

The organization can start by identifying all weak benchmark processes. Choosing the CMM process "Process change management (CL 5.3)" to work on all its related activities, the organization essentially improves the other models' processes as well. Using reverse mapping of the activities, the organization actually partially improves the CMM "Organization process focus (CL 3.1)" process, partial BOOTSTRAP process (CL 2.1) and ISO/IEC 15504 process 5.3. From an assessment point of view, much was to gain from this investment, except for increasing CMM "Process change management (CL 5.3). From a benchmarking point of view, it also raises the SEPRM process 1.2's capability level closer to best practice and makes the most significant contribution to ISO/IEC 15504 "Improve the process (ORG 3)" by raising its capability maturity level from level 0. This is one good example where benchmark coupled with reverse mapping of processes to the Unified Model yielded good benefits with optimum cost spending.

The case studies presented in this chapter have demonstrated the usage of the SEP benchmarking methodology and the gap analysis techniques developed in Chapter 4. The benchmarking results have shown that the assessment and benchmarking methods and tools show some potential and may be useful in the industrial context.

CHAPTER 6 CONCLUSIONS

In this final chapter, a summary of this thesis is provided for the technologies and tools developed. The aims and objectives originally set out Chapter 1 will be reviewed. Lastly, perspectives on future work are presented to hopefully induce the author's opinions in this exciting and interesting area of SEP Benchmarking.

6.1 Summary of this Thesis

This thesis began with an introduction to the challenges of applying benchmarking efficiently and using benchmark effectively in the industry. The initial step taken was to investigate the history of benchmarking which rooted in quality concepts and systems. History has shown that benchmarking was commonly used to achieve political goals. In software engineering even SEP benchmark has been used widely, there was no systematic way of benchmarking, in order to make it as a process that is possible of CPI and shareable between various benchmark repositories. This is considered some of the reasons why industry was slow to pick up on the benchmarking technology.

This lack of benchmarking technology transfer prompted this thesis research into finding an efficient way of benchmarking and an effective benchmark analysis that will be able to have buy-in by the software industry.

The first approach has asked industries what they expect out of benchmarking. The objectives of benchmarking have been set in Chapter 1, that consist of the academic objectives and the industry objectives. The former must be first met prior to be3ing able to meet the latter. Finally, academic objectives have been verified against industry objectives for satisfaction based on case studies for efficiency and effectiveness of the proposed academic findings. The progresses in this work have been mainly on the adoption and implementation of the comprehensive SEP assessment model known as SEPRM, and the design and implementation of the SEP assessment tool and benchmarking tool.

Effective benchmarking has been achieved by coupling the assessment results with new SEP Benchmark Gap Analysis techniques and new SEP Benchmark methodology. The results of this work have been confirmed by industry case studies. This thesis has been designed to supplement executives with information on competitions for their strategic and tactical decision-making.

The following aims and objectives have been achieved in this work:.

- Developed an SEP assessment tool and an SEP benchmarking tool
- Developed a gap analysis technique and an efficient way to perform SEP benchmarking in the industry
- Developed an approach to use SEP Benchmark effectively

6.2 Research Feedbacks from Industries and Conference

The acceptance of this thesis research has been acknowledged both by the industries and conference [Chiew V. et al, 2002.; Wang Y. et al, 2002]. The industries have provided some positive feedbacks, along with some drawbacks, to this research work. Some of the positive feedback included the time and cost saving it would require to perform the assessment and generation of the benchmark results. The effectiveness of the benchmarks in industrial setting could use more empirical studies. The validation will depend heavily on the interpretations of the benchmarks and implementation based on the interpretations. The main drawback to the benchmark is the availability of competitors, benchmark partners, for external benchmarking. The immediate action that can be taken is to perform internal benchmarking, for example between projects or departments. The next step is to approach consulting companies to inquire about availability of suitable partners' benchmarks. There are consulting companies that perform standard-based assessments, which means the assessment information should be readily available for benchmarking, both process-based or model-based benchmarking. At the time of this thesis completion, the author did not get the opportunity to approach any of these consulting companies.

6.3 Perspectives on Future Work

There are many interesting research areas that can be ventured from this work. Some of them are discussed below for future SEP benchmarking partitioning in the future.

- New Standards Integration

  Standards and models do mature along time in its best practiced coverage and assessment process advancement. Nearing the end of this thesis research, the standards and models have been updating or with expanded scope. ISO/IEC 15504 (SPICE 99) updated to ISO/IEC 15504 V.2, ISO 9001:1994 updated to ISO 9001:2000, SW-CMM has been merged with other CMM models into the Capability Maturity Model Integration (CMMI) and finally, BOOTSTRAP Ver. 2.3 to BOOTSTRAP Ver. 3.2. SEPRM is an academic model and can be extended as required to integrate all the models.

  A good starting point is to re-benchmark all the processes based on the standards or models by incorporating all new processes into the comprehensive model, SEPRM. As this thesis was SEP benchmarking capability maturity level, a new research could be to benchmark software engineering standards and models capability maturity progression and coverage; basically benchmarking the standards and models themselves. A couple of interesting investigations will be seen if new standards will favor an organization or not, and how does an organization benchmark result compare between old standards and new standards. This will be covered more in the next section.

New models, such as CMMI, seem to indicate that the trend is to expand the scope of software process engineering to include, hardware process engineering, or at a higher level of, system process engineering. Essentially, the market place is embracing comprehensiveness. Organizations start to the realization that software cannot mature on its own without taking into consideration hardware and system level engineering. Hence, benchmarking research could expand to include hardware and system processes.

- Benchmarks comparison and updating

Comparative benchmarking analysis may provide insight as to which processes and grouping of processes are more valuable for an organization to focus its attention. The organizations that match the newer standard could potentially be viewed as ahead of its time. It may also provide indications to an organization that has already well prepared for the new standards. As for the new standards, the benchmark data might be useful in evaluating its effectiveness base on the results from the current standards.

Theoretically, standards themselves can be, and should be, benchmarked for effectiveness. By benchmarking the current standards against the previous standard can provide insight into the focus of the new standard's direction and its impact on the industry. By performing the benchmark analysis of historical data and trending it, can provide the new standard efficiency in its implementation and execution, and finally its effectiveness by the rate of capability level improvement with the old standard. For example, industry-related research could be to investigate the cost involved to comply with new standard.

- Trending Analysis using Historical Data

The author investigated potential ways to analyze historical benchmark data and results. Other disciplines such as economic, finance and management provided valuable techniques for trend analysis. An example can be found in Figure 6-1. From an ordinary capability plot, two differences Diff 1 and Diff 2 can be calculated. Diff 1 is the gap magnitude and Diff 2 the rate of CL, with a unit of CL/month. Diff 1 and Diff 2 can be plotted as Performance Trend, and Capability and Maturity Trend, respectively as shown in Figure 6-1.

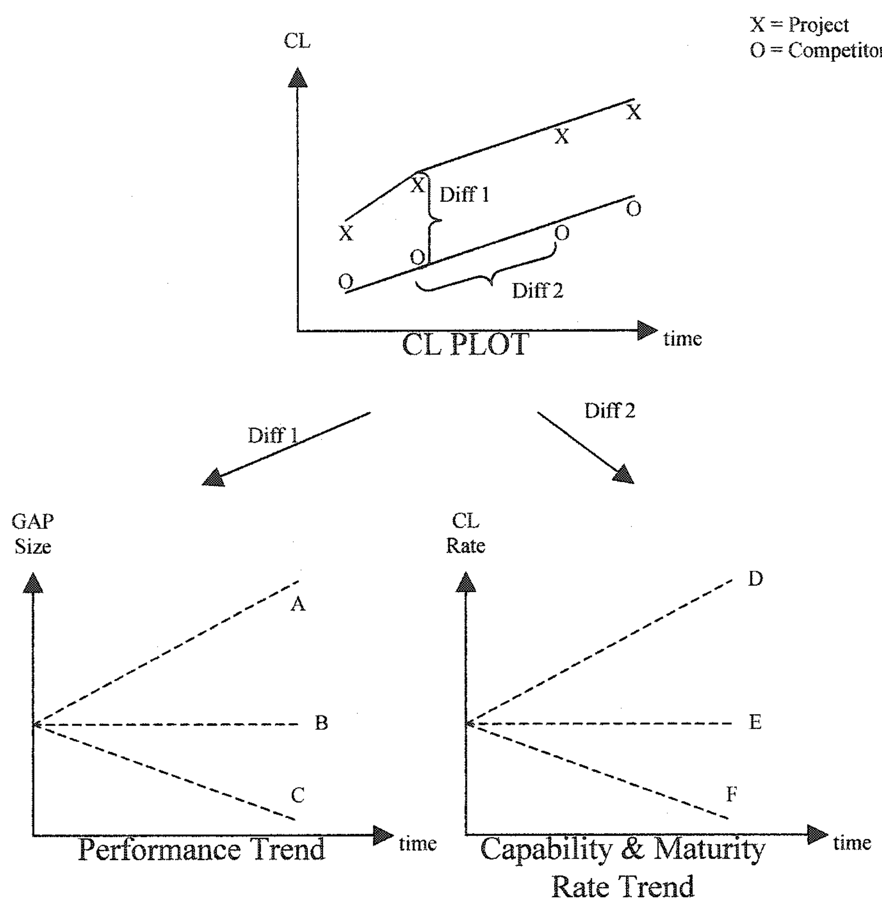

Figure 6-1 Various Trend Plots in benchmarking

Three general lines were plotted for each Capability and Maturity Rate Trend, and Performance Trend. The following trend generalizations can be made for the trends:

- Trend 'A' indicates that the project is achieving superior performance
- Trend 'B' indicates that the project is performing at parity with competitor
- Trend 'C' indicates that the competitor is achieving superior performance
- Trend 'D' indicates that the project is maturing its capability

- Trend ‘E’ indicates that the project is maintaining status quo maturity level
- Trend ‘F’ indicates that the project has maturity opportunity to practice

Even though the concept of benchmarking has been around for a while, the SEP benchmarking is still in its infancy and there is more room for growth and interesting future research.

During this work, the author has published two papers related to this thesis. They are provided in Appendixes C and D as references.

REFERENCES

Aissi S., Malu P., Srinivasan K., (2002), E-Business Process Modeling; The Next Big Step, Computer: International Technology for Computer Professional, IEEE Computer Society Publication, May 2002, ISSN 0018-9162, pp 55-62.

Asian Productivity Organization, (2001), Benchmarking: A Quality and Productivity Improvement Tool, ISBN: 92-833-2287-8, Asian Productivity Organization, pp. 3-98

Baetjer Jr., H (1998), SOFTWARE AS CAPITAL, IEEE Computer Society Press, ISBN 0-8186-7779-1, pp 1-181.

Benwell N (1975), Benchmarking, Hemisphere publishing corporation, Washington, D.C. London, ISBN: 0-470-06595-8, pp. 1-181.

Camp, R. (1989), Benchmarking: The search for industry best practices that lead to superior performance, ISBN 0-87389-058-2, ASQC Quality Press, pp. 1- 288.

Card D, Zubrow D, (2001), Benchmarking Software Organizations, IEEE SOFTWARE September/October 2001, pp. 16-17.

Chiew V., (2001), Industrial Experience: Object Oriented Methodology Approach to Remote Terminal Unit Programming, 7th International Conference on Object Oriented Information Systems, ISBN 1852335467, Springer-Verlag London Limited, pp. 457-464

Chiew V., (2002), A Software Engineering Cognitive Knowledge Discovery Framework, The first IEEE International Conference on Cognitive Informatics ICCI2002, IEEE Publishing, pp. 163-172.

Chiew V., Wang Y., (2002), Software Engineering Process Benchmarking for the Product Focused Software Process Improvement (PROFES) 2002 Conference

Conway, W. E. (1992), The Quality Secret: The Right Way to Manage, Conway Quality, Inc., USA, ISBN: 0-9631464-0-8, pp 102-103.

Defusco R., McLeavey D., Pinto J., Runkle D., (2001), QUANTITATIVE METHODS FOR INVESTMENT ANALYSIS, Association for Investment Management and Research (AIMR), ISBN 0-935015-69-8, United Book Press, pp. 101-147

Dyck, S. (2001), Experience in Assessment of a Software Project by Using Multiple Process Models, 7th International Conference on Object Oriented Information Systems, Springer-Verlag London Limited, ISBN 1852335467, pp. 405-414.

Ebert C, (1999), Technical controlling and software process improvement, The Journal of Systems and Software, Volume 49, Number 1, 1 April 1999, ISSN 0164-1212, Elsevier Science Inc. pp. 25-39.

Eischen K., (2002), Software Development: An Outsider's View, Computer: International Technology for Computer Professional, IEEE Computer Society Publication, May 2002, ISSN 0018-9162, pp 36-43.

Eman K., Drouin J., Melo W., (1998), SPiCE: The Theory and Practice of Software Process Improvement and Capability Determination, ISBN 0-8186-7798-8, IEEE Inc., pp. 1-435

Emam K., Birk A., (2000), Validating the ISO/IEC 15504 measures of software development process capability, The Journal of Systems and Software, Volume 15, Number 2, 15 April 2000, ISSN 0164-1212, Elsevier Science Inc. pp. 119-150.

Emam K., Jung H., (2001), An empirical evaluation of the ISO/IEC 15504 assessment model, The Journal of Systems and Software, Volume 59, Number 1, 15 October 2001, ISSN 0164-1212, Elsevier Science Inc. pp. 23-41.

Ernst & Young, American Quality Foundation, (1992), THE INTERNATIONAL QUALITY STUDY: Best Practices Report – An Analysis of Management Practices that Impact Performance, Ernst & Young, pp. 6-46.

Fogle S. Loulis C, Neuendorf B, (2001), The Benchmarking Process: One Team's Experience, IEEE SOFTWARE September/October 2001, pp. 40-47.

Gamma E., Helm R., Johnson R., Vlissides J., (1998), Design Patterns: Elements of Reusable Object-Oriented Software, Oxford University Press., pp.115,303

Gildersleeve R., (1999), Winning Business: How to Use Financial Analysis and Benchmarks to Outscore Your Competition, ISBN 0-88415-898-5, Gulf Publishing Company, Houston, Texas. pp. 1-322.

Gomes H (1996), Quality Quotes, ASQC Quality Press, Milwaukee, Wisconsin USA, ISBN: 0-87389-407-3, pp 1-213.

Goth G, (2001), NSDIR: A Legacy beyond Failure, Early benchmarking effort was ahead of its time, IEEE SOFTWARE September/October 2001, Pp. 40-47.

Grady R, (1992), PRACTICAL SOFTWARE METRICS FOR PROJECT MANAGEMENT AND PROCESS IMPROVEMENT, Prentice-Hall, Inc. ISBN: 0-13-720384-5, pp. 1-242.

Heires J. (2001), What I Did Last Summer: A Software Development Benchmarking Case Study, IEEE SOFTWARE September/October 2001, pp. 33-39.

Humphrey W., (1990), Managing the Software Process, ISBN 0-201-18905-2, Addison-Wesley Longman Inc., pp. 1-444

Jung H, Hunter R., (2001), The relationship between ISO/IEC 15504 process capability levels, ISO 9001 certification and organization size: An empirical study, The Journal of Systems and Software, Volume 59, Number 1, 15 October 2001, ISSN 0164-1212, Elsevier Science Inc. pp. 43-55.

Juran, J.M. (1995), A History of Managing Quality: The Evolution, Trends, and Future Directions of Managing for Quality, ASQC Quality Press, Juran Foundation, Inc., USA, ISBN: 0-87389-341-7, pp 592, 644.

Koono Z., Chen H., Abolhassani H., Far B., (2001), Design Knowledge and Software Engineering, Wuhan University Journal of Natural Sciences, Vol. 6 No1-2., pp.46-58 http://www.enel.ucalgary.ca/People/far/res-e/theme06.html [Accessed July 28, 2002]

Kulmala J. (2002), APPROACHES TO BENCHMARKING, Finnish Employers' Management Development Institute, FEMDI, Aavaranta, http://www.jyu.fi/economics/research/hankkeet/hankerekisteri/hanke5_benchmarking.htm [Accessed July 14, 2002]

Larman C., (1998), APPLYING UML AND PATTERNS: An Introduction to Object-Oriented Analysis and Design, Prentice-Hall, Inc., pp. 281-284

Lehman M., Ramil J., (1999), The impact of feedback in the global software process, The Journal of Systems and Software, Volume 46, Number 2/3, 15 April 1999, ISSN 0164-1212, Elsevier Science Inc. pp. 123-134.

Lokan C, Wright T, Hill P, Stringer M, (2001), Organizational Benchmarking Using the ISBSG Data Repository, IEEE SOFTWARE September/October 2001, pp. 22-25.

Mashiko Y., Basily V., (1997), Using the GQM Paradigm to Investigate Influential Factors for Software Process Improvement, The Journal of Systems and Software, Volume 36, Number 1, January 1997, ISSN 0164-1212, Elsevier Science Inc. pp. 17-32.

Maurer F., Holz H., (2002), Integrating Process Support and Knowledge Management for Virtual Software Development Teams, Annals of Software Engineering, Special Volume on Process-Based Software Engineering, Kluwer Academic Publishers, Vol 14, 2002 http://sern.ucalgary.ca/~milos/ [Accessed July 28, 2002]

Maxwell K, (2001), Collecting Data for Comparability: Benchmarking Software Development Productivity, IEEE SOFTWARE September/October 2001, pp. 22-25.

McGarry J, (2001), When It Comes to Measuring Software, Every Project is Unique, IEEE SOFTWARE September/October 2001, pp. 18-21.

Pfahl D., Lebsanft K., (1999), Integration of system dynamics modelling with descriptive process modelling and goal-oriented measurement, The Journal of Systems and Software, Volume 46, Number 2/3, 15 April 1999, ISSN 0164-1212, Elsevier Science Inc. pp. 135-150.

Spence D., Ciaschi R., Mantineo A. (2002), The Certification of ESOC to ISO 9001, European Space Agency (ESA) bulletin, number 101, February 2000, ESA Publications Division, ISSN 0376-4265, pp. 48-54.

Thayer R., (2002), Software System Engineering: A Tutorial, Computer: International Technology for Computer Professional, IEEE Computer Society Publication, April 2002, ISSN 0018-9162, pp 68-73.

Thomas G, Smith H, (2001), Using Structured Benchmarking to Fast-Track CMM Process Improvement, IEEE SOFTWARE September/October 2001, pp. 48-52.

Tripp L., (2002), Software Certification Debate: Benefits of Certification, Computer: International Technology for Computer Professional, IEEE Computer Society Publication, June 2002, ISSN 0018-9162, pp 31-33.

Volcker C., Cass A., Winzer L., Carranza J., Dorling A., SPiCE for SPACE: A Process Assessment and Improvement Method for Space Software Development, European Space Agency (ESA) bulletin, number 107, August 2001, ESA Publications Division, ISSN 0376-4265, pp. 112-119.

Wee C, Lee K, Bambang W. H., (1996), SUN TZU War & Management, Addison-Wesley Publisher Ltd., ISBN: 0-201-62858, pp. 1 – 298.

Wang, Y., Wickbert H., Dorling A, Kaartinen M., (1999), Establishment of a National Benchmark of software Engineering Practices, Proceedings of IEEE International Conference on Software Engineering Standards (ISES'99), Curitiba, Brazil, pp. 16-25.

Wang, Y. and King, G (2000), Software Engineering Processes: Principles and Applications, CRC Press, USA, ISBN: 0-8493-2366-5, pp 1-708.

Wang, Y. and King, G (2000a), A New Approach to Benchmark-Based Process Improvement, EuroSPI 2000, European Software Process Improvement, SPI and Assessments.

Wang, Y. (2001), Software Engineering Measurement and Analysis (II), Lecture Notes on Software Engineering Measurement and Analysis, SENG609.01, Univ. of Calgary, Canada.

Wang, Y. (~2001a), A EUROPEAN WEB-BASED SOFTWARE PROCESS BENCHMARKING SERVER, Dept. of Electrical and Computer Engineering, University of Calgary.

Wang Y., He Q., Kliewer C., Khoo T., Chiew V., Nikoforuk W., Chen L., (2002) Object-Oriented Design of a Software Engineering Measurement Expert System for the Product Focused Software Process Improvement (PROFES) 2002 Conference

APPENDIX A: INTERNET WEBSITES OF INTEREST

Carnegie Mellon Software Engineering Institute
http://www.sei.cmu.edu/

ISO 9001 resource (International Organization for Standards)
http://www.iso.ch/

BOOTSTRAP Institute
http://www.bootstrap-institute.com/

ISO/IEC 15504 Internet resource
http://www.esi.es/Projects/SPICE.html
http://www.sei.cmu.edu/iso-15504/
http://www.software.org/quagmire/descriptions/isoiec15504.asp

Official SPICE website
http://www-sqi.cit.gu.edu.au/spice/

SEPRM Internet resource
http://www.enel.ucalgary.ca/People/wangyx/ (SEPRM Creator)
http://www.enel.ucalgary.ca/People/wangyx/Books/Book1.htm (Book)

APPENDIX B: QUOTES OF BENCHMARKING

The following extracted quotes are obtained from Gomes H. book of Quality Quotes [Gomes, 1996] as inspiration to benchmarking.

Adequate is no longer good enough. If a company can't stand shoulder to shoulder with the world's best in a competitive category, it soon has no place to stand at all.

Hammer and Champy

To survive in today's environment of global competition, never-ending change and complexity, rising customer expectations and continuous cost pressures, business process effectiveness and efficiency must constantly improve. A top management focus on process quality management is no longer a choice. It is mandatory!

Juran Institute, Inc.

Historically we've said, "Quality costs money," while our competition overseas has been saying, "Quality makes money." And they've been proving it.

Harrington

The company that constantly responds to its competitors already has its back against the wall.

Aguayo

Benchmarking means out-maneuvering your competitors.

Allan Sayle

The ability to learn faster than your competitors may be the only sustainable competitive advantage.

Arie De Geus (Royal Dutch Shell)

Benchmarking is the difference between teaching yourself how to hit a golf ball and taking lessons from Jack Nicklaus.

George

The corporation that is satisfied with the status quo and makes no attempt to grow will soon find itself stagnating and unable to survive the competition. No matter how superior the technology, it is severely limited it if is confined to the skill of a single worker.

Mizuno

Companies tend to have trouble sustaining competitive advantage. Total quality, because of its focus on benchmarking customer and customer satisfaction, is basically an insurance policy for sustaining competitive advantage over the long term, even when a company might not at any given time, have a blockbuster advantage over the others, Total quality is the very essence of our long term growth strategy.

Edwin Artzt

APPENDIX C: ASSESSMENT AND BENCHMARKING DETAILS

Details provided here will only be presented in such a manner as not to jeopardize the trust and confidentiality between organizations and this research. Anonymity of the organizations will be upheld and can only be revealed to the author of this thesis.

1. Organizations' process rating were collected using the 444 SEPRM Base Process Activities (BPA).
2. Almost all assessment forms returned without much additional information accept the assessment ratings.
3. All the organizations were from Calgary and the one thing they all have in common is the project being assessed involved software development.
4. The author and a software Quality Assurance (QA) engineer performed the case studies organization's assessments. The author provided ratings for three quarters of the assessment process questionnaires while the QA engineer provided rating for the remainder quarter of the process questionnaires. The assessments covered software-related process at the organization level, management level and development level
5. The case studies' processes being assessed and their ratings can be found in APPENDIX D:
6. The "ideal best practice" documentation was used because it will serve as a common denominator for the case studies' organization's projects. Since this is an informal assessment it means there will be no official registration or certification.
7. The case studies used the "ideal best practice" based on the documented way the organization should be running the business.
8. External auditors are required to perform formal assessment on real projects for actual standards compliant registration or certification.
9. Only past project reports were kept by the organization. The auditors' consulting firm kept the original raw assessment data.
10. This thesis is suggesting the use of SEPRM for informal assessments to cut cost and time, by not performing a formal assessment. An informal assessment can be performed by the organization at the convenience of the organization with minimum

impact to the daily business operations. A formal assessment could take weeks and significant amount of resources. While it will only take a day to fill out an informal assessment questionnaire by someone who knows the organization's process and by interviewing both technical and non-technical leads of a given project.

11. The informal assessment can be performed not only on the organization's "ideal best practice" project but also on all current active projects.
12. The assessments were based on the organization's best practice documentation. The documentation has changed a number of times a year since the last time the organization was registered with ISO and certified with SEI.
13. All internal projects should follow the organization's best practice documentation, unless specified by the project contract or statement of work. In reality, a project manager usually has the veto power to make changes to project execution to ensure a successful project outcome.
14. The assessment performed on the "ideal best practice" project can be used as benchmark to evaluate other internal project's capability and maturity levels. The assessments tabled and profiled in Chapter 5 are sample examples of real-life assessment reports used to determine the continuous process improvement action item prioritization based on an ordered list of processes. The quality assurance lead or manager usually determines an ordered list of processes.
15. The internal benchmarking can be performed between projects or departments, pending the insights required by the organization.
16. The "ideal best practice" project can be used to evaluate internal projects that have deficiencies according to the benchmark, depending on whether the benchmark is model based or competitor based for continuous process improvement. Internal benchmarks have to eventually translate to external benchmarks to ensure the organization's goals are realized.
17. While the Quality Assurance department is using the "ideal best practice" project to assure all internal projects are being executed accordingly to the best practice, tactically, the upper management can use the "ideal best practice" benchmark for strategic planning. The strategic planning usually involves an assessment to determine the competitors' current state of capability and maturity level. The single

model or multi models percentile rankings are important in determining the organization's current state of assessment against the competitors with respect to the demand of the market place for the desired capability and maturity level for vendors.

18. Currently, there is no standardization or regular practice of using multi or comprehensive SEP standard-based benchmark reports, tables or profiles to be used as basis for continuous process improvement effort. Individual standard does have specific convention for performing continuous process improvement.

APPENDIX D: PROJECT X'S ASSESSMENT RATINGS

The ratings were based on the threshold indicated in Table 3-2.

Due to non-disclosure agreement (NDA) between the organization and the author, only the ratings are publishable.

No.	Base Process Activities (BPA in SEPRM)	Rating
1	Define organization structure	4
2	Establish business strategy	4
3	Define management responsibilities	4
4	Establish organization's general quality policy	4
5	Assign project managers	4
6	Define career plans	4
7	Review projects periodically	4
8	Define project teams	4
9	Define project management responsibilities	4
10	Assign SQA personnel or team	0
11	Maintain project team interactions	0
12	Management commitment on quality	4
13	Assign system analyst to management team	1
14	Define process goals	0
15	Identify current activities/responsibilities	4
16	Identify inputs/outputs of process	4
17	Establish organization's standard process	0
18	Document standard process	4
19	Report standard process	4
20	Define tailorability of standard process	4
21	Organization level process coordination	4
22	Define entry/exit criteria of processes	4
23	Define control points/milestones	4
24	Identify external interfaces	4
25	Identify internal interfaces	4
26	Define quality records	4
27	Define process measures	4
28	Establish performance expectations	4
29	Plan process improvement	4
30	Assess current process periodically	4
31	Identify improvement opportunities	1
32	Define scope of improvement activities	1
33	Prioritize improvement	1
34	Define measures of impact	1
35	Change process for improvement	4

36	Pilot trial of new process	1
37	Assess new process	1
38	Document improved process	1
39	Report/train new process	1
40	Obtain customer requirements	4
41	Document customer requirements	4
42	Define service procedures	4
43	Understand customer expectation	4
44	Define customer responsibility	1
45	Keep customers informed	4
46	Establish joint audits/reviews	1
47	Prepare for customer audits/reviews	4
48	Conduct joint management reviews	1
49	Conduct joint technical reviews	4
50	Support customer acceptance review	4
51	Perform joint process assessment	1
52	Regular interchange with customers	1
53	Identify operational risks	4
54	Support software installation	4
55	Perform operational testing	4
56	Demonstrate software operation	4
57	Resolve operational problems	4
58	Handle user requests	4
59	Document temporary workaround	4
60	Monitor system capacity and service	4
61	Train customer	4
62	Establish product support	4
63	Monitor performance	1
64	Install product upgrades	4
65	Define software replication procedure	4
66	Define installation procedure	4
67	Define delivery procedure	4
68	Identify installation requirements	4
69	Prepare site for installation	4
70	Pack software package	4
71	Deliver after conformance verified	4
72	Document acceptance of software	4
73	Deliver software on time	4
74	Verify correct receipt	4
75	Provide handling and storage procedures	0
76	SQA review with customers	0
77	Feedback customer information	0
78	Determine customer satisfaction level	4
79	Compare with competitors	1
80	Review customer satisfaction	0

81	Record customer failure reports	4
82	Aware of state-of-the-art in software engineering	0
83	Survey methodologies/ technologies adopted externally	0
84	Evaluate life cycle model	0
85	Evaluate prototype model	0
86	Evaluate OOP model	0
87	Evaluate combined model	0
88	Evaluate CASE model	0
89	Integrate methodologies and tools into process	4
90	Distinguish development category: system prototype/new system/improved version	4
91	Determine organizational reuse strategy	0
92	Identify reusable components	4
93	Develop reusable components	4
94	Establish reuse library	4
95	Certify reusable components	4
96	Integrate reuse into life cycle	0
97	Propagate change carefully	4
98	Plan technology change	4
99	Identify processes needed in technology change	0
100	Identify/replace obsolete technology/ process	4
101	Select new technology	4
102	Introduce new technology/metrics/process	4
103	Pilot trial of new technology	4
104	Incorporate trialed technology into current process	4
105	Evaluate software development methodologies	4
106	Model software process	4
107	Describe activities and responsibilities	4
108	Establish task sequences	4
109	Identify process relationships	4
110	Document process activities	4
111	Identify control point of project	4
112	Maintain consistency across all processes	4
113	Develop software according to defined process	4
114	Derive project process by tailoring organization's standard process	4
115	Approval of processes and equipment	4
116	Identify special requirements in developing special system: real-time/safety-critical/etc	4
117	Analyze requirement according to defined process	4
118	Specify formal requirements	4
119	Define requirements feasibility/testability	4
120	Prevent ambiguities in specification	4
121	Interpret/clarify requirements	4
122	Specify acceptance criteria	4
123	Allocate requirements for processes	4
124	Adopt requirements acquisition tools	4
125	Design system according to defined process	4

126	Design software architecture	4
127	Design module interfaces	4
128	Develop detailed design	4
129	Establish document traceability	4
130	Specify final design	4
131	Define design change procedure	4
132	Adopt architectural design tools	4
133	Adopt module design tools	4
134	Code according to defined process	4
135	Choose proper programming language(s)	4
136	Develop software modules	4
137	Develop unit verification procedures	4
138	Verify software modules	4
139	Document coding standards	4
140	Define coding styles	4
141	Adopt coding support/auto-generation tools	4
142	Testing according to defined process	4
143	Determine test strategy	4
144	Specify test methods	4
145	Generate test	4
146	Conduct testing	4
147	Adopt module testing tools	4
148	Integration test according to defined process	4
149	Acceptance test according to defined process	4
150	System tests generation	4
151	Test integrated system	4
152	Adopt software integration tools	4
153	Adopt module cross-reference tools	4
154	Adopt system acceptance testing tools	4
155	Determine maintenance requirements	1
156	Analyze user problems and enhancements	4
157	Determine modifications for next upgrade	4
158	Implement/test modifications	4
159	Update user system	4
160	Maintenance consistency with specifications	4
161	Maintain nonconforming products	4
162	Record nonconformance treatment	4
163	Adopt regression testing tools	4
164	Conduct regression testing	4
165	Identify environment requirements	1
166	Establish computer-supported cooperative work (CSCW) environment	4
167	Provide software engineering environment	4
168	Provide development supporting tools	4
169	Provide management supporting tools	4
170	Provide interactive communication environment	4

171	Maintain software engineering environment	4
172	Plan required resources	4
173	Identify specialized facilities	4
174	Acquire resources	4
175	Check resources availability	4
176	Provide productive workspace	4
177	Provide data backup	4
178	Provide building facilities	4
179	Provide remote access facility	4
180	Adopt software design tools	4
181	Adopt software testing tools	4
182	Ensure data integrity	4
183	Register/maintain test equipment	4
184	Control customer-supplied equipment	4
185	Record equipment condition	4
186	Ensure equipment availability	4
187	CASE tools	4
188	Software requirements acquisition tools	4
189	Software design tools	4
190	Software testing tools	4
191	SQA management tools	0
192	Software requirements review tools	4
193	Software design review tools	4
194	Software testing analysis tools	4
195	Software configuration management tools	4
196	Software documentation processing tools	4
197	Define SQA procedure	4
198	Define project s/w engineering standards	4
199	Document SQA system	0
200	Issue quality manual	4
201	Distribute quality policy	4
202	Report SQA results	4
203	Assess process quality	4
204	Take correct actions	4
205	Assign independent reviewers	4
206	Define extent of inspection	4
207	Conduct SQA for each process	0
208	Assign qualified person(s) to special process	0
209	Document quality records	4
210	Review SQA system suitability	4
211	Decisional role of SQA in processes	4
212	Decisional role of SQA in final products	4
213	Adopt SQA tools	0
214	Specification verification	4
215	Formal review requirements	4

216	Review statutory requirements	4
217	Customer accepts specifications	4
218	Adopt specification verification tools	4
219	Define design review procedure	4
220	Document design review	4
221	Verify prototypes	4
222	Measure design review coverage	4
223	Conduct code walk-through	4
224	Conduct code review	4
225	Measure code review coverage	0
226	Measure test coverage	0
227	Estimate remaining error distribution	0
228	Review test results	4
229	Static/dynamic module test analysis	4
230	Identify nonconforming software/functions	4
231	Define inspection procedure	4
232	Inspection against requirements	4
233	Document inspection/test results	4
234	Static/dynamic integration test analysis	4
235	Static/dynamic acceptance test analysis	4
236	Reinspect repaired products	4
237	Audit nonconformance records	4
238	Audit nonconformance treatment	4
239	Audit consistency with specification	4
240	Audit consistency of system documents	4
241	Audit consistency of system configuration	4
242	Audit user satisfaction with maintenance	4
243	Review regression testing results	4
244	Audit software development activities	4
245	Audit work products	4
246	Audit process quality	4
247	Audit on-site activities	0
248	Document audit results	4
249	Verify representativeness of examined samples	0
250	Plan peer review	4
251	Select work products	4
252	Identify review standards	4
253	Establish completion criteria	4
254	Establish re-review criteria	4
255	Distribute review materials	4
256	Conduct peer review	4
257	Document review results	4
258	Take actions for review results	4
259	Track actions for review results	4
260	Plan defect prevention	4

261	Defect reporting and record	4
262	Defect causal analysis	4
263	Propose process change for defect prevention	4
264	Track problem report	4
265	Prioritize problems	4
266	Determine resolutions	4
267	Correct defects	4
268	Review defect corrections	4
269	Distribute correction results	4
270	Subcontractor's quantitative quality goals	4
271	Assess/test quality of subcontractor's product	4
272	Acceptance test for subcontractor's software	4
273	Safeguard customer-supplied products	4
274	Record customer-supplied products	4
275	Assign project proposal team	4
276	Design project process structure	4
277	Determine reuse strategy	0
278	Establish project schedule	4
279	Establish project commitments	4
280	Document project plans	4
281	Conduct progress management reviews	4
282	Conduct progress technical reviews	4
283	Management commitments in planning	4
284	Determine release strategy	4
285	Plan change control	4
286	Define plan change procedure	4
287	Plan development	4
288	Plan testing	4
289	Plan system integration	4
290	Plan process management	4
291	Plan maintenance	4
292	Plan review and authorization	4
293	Assign development task	4
294	Adopt project/process planning tools	4
295	Estimate project costs	4
296	Estimate project time	4
297	Estimate resources requirement	4
298	Estimate staff requirement	4
299	Estimate software size	4
300	Estimate software complexity	4
301	Estimate critical resources	4
302	Identify project risks	4
303	Establish risk management scope	4
304	Identify unstable specification- related risks	4
305	Identify process change-related risks	4

306	Identify market-related risks	4
307	Analyze and prioritize risks	4
308	Develop mitigation strategies	4
309	Define risk metrics for probability/impact	4
310	Implement mitigation strategies	4
311	Assess risk mitigation activities	4
312	Take corrective actions for identified risk	4
313	Plan SQA	4
314	Establish quality goals	0
315	Define quality quantitative metrics	0
316	Identify quality activities	0
317	Track project quality goals	0
318	SQA team participate in project planning	4
319	Plan maintenance	4
320	Plan quantitative process management	0
321	Conduct quantitative process management	0
322	Collect data for quantitative analysis	0
323	Control defined process quantitatively	0
324	Document quantitative analysis results	0
325	Benchmark organization's baseline of process capability	0
326	Manage project by defined process	4
327	Adopt project/process management tools	4
328	Track project progress	4
329	Track development schedule	4
330	Track process quality	0
331	Track software size	0
332	Track project cost	4
333	Track critical resources and performance	4
334	Track project risks	4
335	Track process productivity	4
336	Track system memory utilization	4
337	Track system throughput	4
338	Track system I/O channel capabilities	4
339	Track system networking	4
340	Adopt process tracking tools	4
341	Document project tracking data	4
342	Identify and handle process deviation	4
343	Establish configuration management library	4
344	Adopt configuration management tools	4
345	Identify product's configuration	4
346	Maintain configuration item descriptions	4
347	Control change requests	4
348	Release control	4
349	Maintain configuration item history	4
350	Report configuration status	4

351	Establish change requests/approval system	4
352	Control requirement change	4
353	Control design change	4
354	Control code change	4
355	Control test data change	0
356	Control environment change	0
357	Control schedule change	4
358	Control configuration change	4
359	Adopt change control tools	4
360	Review processes at milestones	0
361	Document project review data	0
362	Revise project process	0
363	Conduct statistical analysis of process	0
364	Gather process data	0
365	Compare actual/forecast errors	0
366	Compare actual/forecast schedule	4
367	Compare actual/forecast resources	4
368	Define interface between project groups	4
369	Plan intergroup activities	0
370	Identify intergroup critical dependencies	0
371	Handle intergroup issues	4
372	Technical/management representatives coordination	4
373	Review last process output	4
374	Conduct intergroup representatives review	0
375	Specify system requirements	4
376	Design system based on requirements	4
377	Allocate requirements	4
378	Determine operating environment impact	0
379	Determine software requirements	4
380	Analysis of software requirements	4
381	Evaluate requirements with customer	4
382	Update requirements for next iteration	4
383	Agree on requirements	4
384	Establish requirements standard	4
385	Manage requirements changes	4
386	Maintain requirements traceability	4
387	Define contractual procedures	4
388	Prepare contract proposal	4
389	Review contract	4
390	Ensure agreement of terminology	4
391	Determine interfaces to independent agents	4
392	Assess contractor's capability	0
393	Document contractor's capability	0
394	Specify subcontracted development	4
395	Assess capability of subcontractors	0

396	Record acceptable subcontractors	4
397	Define scope of contracted work	4
398	Define interface of contracted work	4
399	Select qualified subcontractor	4
400	Approve subcontractor's plan	4
401	Maintain interchanges with subcontractors	4
402	Track subcontractor's development activities	4
403	Monitor subcontractor's SQA activities	4
404	Review subcontractor's work	4
405	Assess compliance of contracted product	4
406	Determine interfaces to subcontractors	4
407	Document subcontractor's records	4
408	Identify need for purchasing	4
409	Define purchasing requirements	4
410	Prepare acquisition strategy	4
411	Prepare purchasing document	4
412	Prepare request for proposal	4
413	Review purchasing document	4
414	Select software product supplier	4
415	Verify purchased product	4
416	Manage purchased tools configuration	4
417	Master list of project documents	4
418	Determine documentation requirements	4
419	Develop document	4
420	Check document	4
421	Control document issue	4
422	Maintain document	4
423	Documentation according to defined process	4
424	Establish documentation standards	4
425	Safety document storage	0
426	Identify current versions of documents	4
427	Adopt interactive documentation tools	4
428	Establish organization's process library	4
429	Establish organization's process database	4
430	Establish software reuse library	0
431	Establish organization's metrics database	4
432	Establish operation manual library	4
433	Establish practice benchmark database	0
434	Define qualifications for positions	4
435	Define experience for positions	4
436	Assign personnel selection group	4
437	Select staff by qualification / experience	4
438	Plan training	4
439	Identify training needs	4
440	Develop training courses	4

441	Approval of training courses	4
442	Conduct technical training	4
443	Conduct management training	4
444	Document training records	4