



**COMPUTER SCIENCE DEPARTMENT**  
**MSC COMPUTER SCIENCE DISSERTATION**

---

**TOWARDS A SOFTWARE PROCESS IMPROVEMENT SELECTION FRAMEWORK  
FOR SMALL AND MEDIUM ENTERPRISES IN BOTSWANA**

---

BY

MOTSOMI RAPOO

200102148

A Dissertation submitted to the Computer Science Department, University of Botswana, In  
Partial Fulfillment of the Requirements for the Degree of Master of Science.

**SUPERVISOR:** Dr. Ezekiel U. Okike

**MAY 2016**

**STATEMENT OF ORIGINALITY**

This is to affirm that the work contained in this dissertation was completed by the author by May 2016. It is the original work except in cases of where a reference has been made.

---

STUDENT SIGNATURE

---

DATE

## APPROVAL PAGE

This dissertation has been examined and is approved as meeting the requirements for a partial fulfillment of the requirements for the degree of MSc.

---

SUPERVISOR

---

DATE

---

INTERNAL

---

DATE

---

EXTERNAL EXAMINER

---

DATE

## **ACKNOWLEDGEMENT**

First and foremost I give all glory to God who has been my strength and guide during the time of working on this project. Although it was not easy, the Lord has helped me to finally finish. Secondly I would like to appreciate my supervisor Dr. Ezekiel U. Okike who took his time to supervise this work.

I will also like to thank my wife who has been so much of encouragement to me. She believed in me, that I can make it as I finally did.

## **ABSTRACT**

Software Process Improvement (SPI) models are quality models or standards for software development used to produce quality software. It involves systematic procedures to improve performance of an existing processing system by changing the current processes or updating new processes in order to avoid problems identified in the old processing system by means of a process assessment. A variety of SPI models have been suggested in the literature which have been used to improve software products. However software companies and other organisations still find it a challenge to choose or select the right SPI model to use in their organisations. In Botswana for instance the majority of software companies are Small Medium Enterprises (SMEs). For SMEs, the challenge is even worse as most of the SMEs already have challenges using the existing SPI models. This study focuses on finding out the software companies' use of SPI models in Botswana. The study also proposes a framework for choosing or selecting appropriate SPI model for the SMEs in Botswana.

Relevant data was collected from software companies that develop and sell software products to their clients. Government departments and business organizations that use software products were also studied to assess their capacity in software usage. A total number of 14 software companies were used in finding out the status of the software companies' capability and maturity in software development in Botswana. Another set of 16 government and business organizations were used to evaluate the use of the software products in Botswana. Out of the selected samples, only less than half (7) of the software companies in Botswana are familiar with the existing standards. Of these companies 4 claim to be familiar with the Capability Maturity Model/Capability Maturity Model Intergrated (CMM/CMMI). However, further assessment to certify their maturity on the CMM scale showed that only 2 companies have attained level 2 on the CMMI scale, and only 1 of these two have attained levels 3 and 4 of the Capability Maturity Model.

Further, an SPI Selection framework for SMEs is proposed in this study. This is to enable the SMEs select and adopt appropriate SPI framework when necessary.

## Contents

CHAPTER 1 .....	17
1.0 INTRODUCTION.....	17
1.1 Background.....	17
1.2 Motivation .....	22
1.3 Problem Statement .....	22
1.4 Research Objectives .....	23
1.5 Research Questions.....	23
1.6 Significance of the Study .....	24
1.7 Chapter Summary .....	24
CHAPTER 2: LITERATURE REVIEW .....	25
2.0 INTRODUCTION.....	25
2.1 Software Process Models .....	25
2.1.1 Waterfall Model .....	27
2.1.2 Rapid Application Development (RAD).....	28
2.1.3 Evolutionary Development Models.....	30
<i>Exploratory development:</i> .....	31
Incremental Model .....	31
Throw Away Prototyping .....	33
2.1.4 Spiral Model.....	34
2.1.5 Re-use Oriented Development.....	36
2.1.6 Agile Software Development .....	37
2.2 SOFTWARE PROCESS IMPROVEMENT (SPI) MODELS.....	39
2.2.1 SOFTWARE PROCESS IMPROVEMENT PROCEDURE .....	39

2.2.2	Capability Maturity Model (CMM) .....	41
2.2.3	CMMI (Capability Maturity Model Integration) .....	42
2.2.4	SPICE (Software Process Improvement and Capability Determination).....	48
2.2.5	BOOTSTRAP.....	49
2.2.6	IDEAL (Initiating, Diagnosing, Establishing, Acting and Learning) .....	51
2.2.7	TICKIT SCHEME.....	53
2.3	SOFTWARE PROCESS IMPROVEMENT FOR SMEs .....	54
2.3.1	OWPL GRADUAL FRAMEWORK.....	55
2.3.2	AN APPROACH FOR SOFTWARE PROCESS ESTABLISHMENT IN MICRO AND SMALL COMPANIES (ASPE-MS).....	56
2.3.3	iFLAP (Improvement Framework Utilising Light Weight Assessment and Improvement Planning) 57	
2.3.4	An Approach to Software process Improvement for Small to Medium Enterprises (PRISMS) 60	
2.3.5	Software Process Metrix (SPM) Model.....	61
2.3.6	MESOPYME .....	62
2.4	BUSINESS PROCESS IMPROVEMENT VERSUS SOFTWARE PROCESS IMPROVEMENT (SPI).....	65
2.5	OTHER STANDARDS AND FRAMEWORKS.....	71
2.5.1	ISO 9000.....	71
2.6	SOFTWARE PROCESS ASSESSMENT (SPA).....	73
2.7	<i>SOFTWARE ESTIMATION STANDARDS AND MODELS</i> .....	74
2.7.1	Software Effort Estimation .....	74
2.7.2	CONSTRUCTIVE COST MODEL II (COCOMO II) .....	77
2.7.3	CONSRUCTIVE QUALITY MODEL (COQUALMO) .....	77
2.8	SOFTWARE QUALITY STANDARDS .....	77
2.9	SOFTWARE STANDARDS APPLICATION IN SMALL COMPANIES.....	78

2.9.1	SOFTWARE PROCESS IMPROVEMENT IN SOME NATIONS .....	79
2.9.2.2	<i>SPI in small and medium software organizations in United Kingdom</i> .....	80
2.9.2.3	Software Process Improvement Models in Malaysia .....	81
2.9.2.4	Software Process Improvement in Bangladesh .....	82
2.9.2.5	Software Process Improvement in Botswana.....	84
2.10	CHOOSING SOFTWARE MODEL .....	85
2.10.1	CHOOSING SOFTWARE PROCESS MODELS .....	86
2.10.2	DECISION SUPPORT SYSTEM BY G. HOLODNIK-JANCZURA AND I. GOLINSKA .....	86
2.10.3	MULTI-CRITERIA DECISION AIDING - CONSTRUCTIVIST (MCDA-C) .....	91
2.10.4	A DECISION SUPPORT SCHEME FOR SOFTWARE PROCESS IMPROVEMENT PRIORITIZATION 92	
2.11	PROPOSED MODEL FOR CHOOSING SPI .....	92
2.11.1	Organisation size .....	93
2.11.2	Project Team Size .....	93
2.11.3	Kind of Project Carried Out in an organisation .....	93
2.11.4	Availability of Funds/ Financial spine .....	93
2.11.5	Staff Experience .....	93
2.11.6	The Process to be improved .....	94
CHAPTER 3: SOFTWARE PROCESSES AND PROCESS IMPROVEMENT IN BOTSWANA .....		95
3.0	INTRODUCTION.....	95
3.1	METHODOLOGY .....	95
3.1.1	Data Collection Methods .....	96
	Sampling method .....	96
3.1.2	Research Questions.....	97
3.1.3	Interviews.....	99



3.1.4	Site Selection .....	99
3.1.5	Data Analysis .....	99
CHAPTER 4:	RESULTS AND DISCUSSION .....	101
4.0	PRESENTATION OF RESULTS .....	101
4.1	MODEL FITTING.....	101
4.2	SOFTWARE COMPANIES – CERTIFICATION AND STANDARD APPLICATION .....	110
4.2.1	Companies certified with ISO 9001 .....	110
4.2.2	Companies certified with ISO 9002 .....	111
4.2.3	Companies certified with CMM .....	111
4.2.4	Companies certified with CMMI .....	112
4.2.5	Capability Maturity Level of Software Companies in Botswana .....	112
4.2.6	Companies certified with ISO 9003 .....	116
4.3	COMPANIES STAFF CONCENTRATION.....	117
4.3.1	TOTAL NUMBER OF EMPLOYEES IN A COMPANY .....	117
4.3.2	FULLTIME IT PROFESSIONALS .....	118
4.3.3	PART TIME PROFESSIONALS.....	119
4.3.4	SOFTWARE DEVELOPERS (BOTH FULL-TIMERS AND PART-TIMERS) .....	119
4.3.5	COMPUTER SCIENCE GRADUATES.....	120
4.3.6	SOFTWARE QUALITY CONTROL.....	122
4.3.7	NUMBER OF SOFTWARE PRODUCTS DEVELOPED/CUSTOMISED AND MAINTAINED BY THE COMPANIES.....	126
4.3.8	UTILISATIONS OF SOFTWARE PROCESS MODELS.....	128
4.4	ORGANISATIONAL USE OF SOFTWARE PRODUCTS.....	129
4.5	DISCUSSION .....	135
4.6	PROPOSAL OF A SME SPI SELECTION FRAMEWORK .....	137

4.6.1	Process improvement Approach Model .....	137
4.6.3	RELATING OUR MODELS .....	143
5.0	RESEARCH SUMMARY .....	144
5.1	RESEARCH CONTRIBUTION .....	144
5.2	RESEARCH CONSTRAINTS .....	145
5.3	FUTURE WORK.....	145
	BIBLIOGRAPHY .....	146

## LIST OF TABLES

Table 1 - Model summary .....	101
Table 2 - Anova.....	102
Table 3 - COEFFICIENT .....	103
Table 4 - Excluded Variables .....	105
Table 5: Botswana Software Companies use of SPI Models.....	107
Table 6 - Case Processing Summary on SPI Budget by Companies .....	109
Table 7 - Budget for Software Process usage .....	110
Table 8 - COMPANY CERTIFICATION ON ISO 9001 .....	111
Table 9 - COMPANY CERTIFICATION WITH ISO 9002 .....	111
Table 10 - COMPANY CERTIFICATION WITH CMM.....	112
Table 11 - COMPANY CERTIFICATION WITH CMMI.....	112
Table 12 - COMPANY CERTIFICATION ON ISO 9003 .....	116
Table 13 - Number of part time IT Professionals .....	119
Table 14 - Staff with Masters Computer Science .....	122
Table 15 - Company Quality Measuring methods.....	123
Table 16 - Company Quality Assurance attempts.....	124
Table 17 - Company Available Software Quality Guidelines.....	125
Table 18 - Number of software Products developed between 2012 and 2014.....	126
Table 19 - Number of software Systems maintained.....	127
Table 20 - Use of Agile Methodology .....	127
Table 21 - Company utilization of software process models.....	128
Table 22 - Usefulness of software process models.....	129
Table 23 - Foreign and local companies supporting organisations in Botswana .....	130
Table 24 - Maintenance by Local Companies .....	130

Table 25 - Availability of software products in organisation .....	131
Table 26 - Purchasing of Ready-made Products .....	132
Table 27 - Software Product Customization .....	132
Table 28 - Last Software Purchase .....	133
Table 29 - Organization annual software estimate .....	134

## LIST OF FIGURES

Figure 1 - Simple Software Development Process .....	19
Figure 2 - Software Process .....	25
Figure 3 - Waterfall basic model [Wason, C., 2006] .....	27
Figure 4 - generic framework activities for RAD- Roger S. Pressman 2005 .....	29
Figure 5 - Exploratory Development .....	31
Figure 6 - Throwaway Prototyping Model .....	34
Figure 7 - Basic Spiral model [Boehm, 1985] .....	35
Figure 8 - Agile Software Development Lifecycle.....	38
Figure 9 - Process Improvement Approach .....	40
Figure 10 - Evolution CMMI model .....	43
Figure 11 - CMMI Model.....	45
Figure 12 - Continuous Steps for software Process Improvement for SPICE, Marko Phyhajarvi, 2004 .....	49
Figure 13 - Process Area in BOOTSTRAP Architecture.....	51
Figure 14 - IDEAL Model .....	52
Figure 15 - OWPL model .....	56
Figure 16 - Overview on ASPE - MSC Approach .....	57
Figure 17 - A Process Deliverable Diagram for iFLAP Framework.....	59
Figure 18 - PRISMS SPI Approach .....	61
Figure 19 - Software Process Metrix .....	62
Figure 20 - Main Architecture of the MESOPYME Action Package .....	63
Figure 21 - MESOPYME process improvement .....	64
Figure 22 - Elements of a Process .....	66
Figure 23 - Project and Process Management Decision Making Tree.....	67
Figure 24 - Questions for Process Improvement Chart .....	68

Figure 25 - Software Process Improvement Management Decision making Tree.....	69
Figure 26 - Abstract, layered division of the decision-making system.....	88
Figure 27 - Algorithm for selecting an SDLC model.....	90
Figure 28- Staff population in Software companies in Botswana.....	117
Figure 29 - Number of full time professionals.....	118
Figure 30 - Number of full time and Part time professionals focusing on Software development.....	120
Figure 31 - Graduates in Computer Science/Engineering Graduate.....	121
Figure 32 - Staff members with Diploma/certificate in IT courses.....	121
<b>Figure 33 - Process improvement approach.....</b>	<b>137</b>
Figure 34 - SME Selection Framework portion.....	139
Figure 36 - SME Selection Framework.....	141

## **ACRONYMS AND ABBREVIATIONS**

**AI** – ARTIFICIAL INTELLIGENCE

**ANN** – ARTIFICIAL NEURAL NETWORK

**ASPE-MSc** - APPROACH FOR SOFTWARE PROCESS ESTABLISHMENT IN MICRO AND SMALL COMPANIES

**BPI** – BUSINESS PROCESS IMPROVEMENT

**CART** – CLASSIFICATION AND PROGRESSION TREES

**COQUALMO** – CONSTRUCTIVE QUALITY MODEL

**CMM** – CAPABILITY MATURITY MODEL

**CMMI** - CAPABILITY MATURITY MODEL INTEGRATION

**CRM** – CUSTOM RELATION MANAGEMENT

**EBA** – ARTIFICIAL NEURAL NETWORK

**ERP** - ENTERPRISE RESOURCE PLANNING

**IDEAL** - INITIATING, DIAGNOSING, ESTABLISHING, ACTING AND LEARNING

**IEEE** – INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS

**IFLAP** - IMPROVEMENT FRAMEWORK UTILISING LIGHT WEIGHT ASSESSMENT AND IMPROVEMENT PLANNING

**ISO/IEC** – INTERNATIONAL ORGANIZATION FOR STANDARDIZATION/ INTERNATIONAL ELECTRO TECHNICAL COMMISSION

**IT** – INFORMATION TECHNOLOGY

**KPA** – KEY PERFORMANCE AREA

**MCDA-C** - MULTI-CRITERIA DECISION AIDING – CONSTRUCTIVIST

**POS** – POINT OF SALES

**PPADB** – PUBLIC PROCUREMENT AND ASSET DISPOSAL BOARD

**PRISMS** - APPROACH TO SOFTWARE PROCESS IMPROVEMENT FOR SMALL TO MEDIUM ENTERPRISES

**QA** – QUALITY ASSURANCE

**QFD** - Quality Function Deployment

**SDLC** – SOFTWARE DEVELOPMENT LIFE CYCLE

**SDP** – SOFTWARE DEVELOPMENT PROCESS

**SEI** – SOFTWARE ENGINEERING INSTITUTE

**SME** – SMALL MEDIUM ENTERPRISES

**SPA** – SOFTWARE PROCESS ASSESSMENT

**SPI** – SOFTWARE PROCESS IMPROVEMENT

**SPICE** - SOFTWARE PROCESS IMPROVEMENT AND CAPABILITY DETERMINATION

**SPM** - SOFTWARE PROCESS METRIX

**SPSS** – STATISTICAL PACKAGE SOCIAL SCIENCES

**SPA** – SOFTWARE PROCESS ASSESSMENT

**VSE** – VERY SMALL ENTERPRISES



## CHAPTER 1

### 1.0 INTRODUCTION

#### 1.1 Background

There is need for continuous improvement in the software development process in order to come up with a successful and quality product. A quality process will produce a quality product. The understanding of this important principle has influenced companies to hire professionals with multiple skills, implement new technologies, adapt new methods, standards and technologies in order to improve their processes which will help in building sustainable and successful software in the market. Software process is a set of activities and the associated results which produce a software product [1]. Software processes are indispensable in the development of quality software product. Apart from this reason, a process provides organization stability and more control to its activities. Software process results into a software process models, which is a simplified description of a software process represented from a particular perspective [1].

A software process model is an abstraction of the actual process which is being described. There are many software process models which can be used depending on the project at hand.

Software companies need to make use of software processes and software process models in order for them to have good products. Although many software companies over the world are coming to understand the need for software process improvement [2], some organization are yet to realize this need. Many companies are spending too much effort and resources in order to improve the performance of their overall process but the disappointment rate is evident [3]. According to the business dictionary, *Software Performance* is the accomplishment of a given task measured against preset standards of accuracy, completeness, cost, and speed. In a contract, performance is deemed to be the fulfillment of an obligation, in a manner that releases the performer from all liabilities under the contract [3].

In 1992, a joint project was initiated between the Software Engineering Institute (SEI) and Siemens to investigate the impact of software process improvement methods. One of the questions that motivated the project was on how an organization should select methods for software process improvement [4]. This is still a concern to many companies as they are still struggling to select the right improvement process to adopt. Some companies in Botswana appear not to have software process improvement methods in place. Whilst this can be attributed to a number of reasons which includes: i) the expensive process of process improvement ii) unfriendliness of existing Software Process Improvement, Small Medium Enterprises (SMES) have a great challenge in choosing the right software process improvement to use. Although it is a challenge to choose SPI, SMEs can be guided by a number of factors like status of the organization, the processes to be improved and how to improve the process itself [5].

SMEs, as the name suggests, are small companies with limited budget and resources [5]. This makes it difficult for them to afford common process improvement models.

The importance of software process improvement has influenced the Information Technology (IT) industry to come up with quite a number of software process improvement methods. Examples of Software Process Improvement (SPIs) models includes Capability Maturity Model (CMM), Capability Maturity Model Integration (CMMI), Initiating, Diagnosing, Establishing, Acting and Learning (IDEAL). SMEs have a serious challenge with using the aforementioned models [6]. As a result a number of simpler models have been proposed such as; Approach for Software Process Establishment in Micro and Small Companies (ASPE-MS), Improvement Framework Utilising Light Weight Assessment and Improvement Planning (iFLAP), Approach to Software process Improvement for Small to Medium Enterprises (PRISMS), Software Process Metrix (SPM) and MESOPYME. It is of no doubt that new methods are also coming up, and they will continue to emerge. Nevertheless, the need to involve appropriate selection criteria cannot be overemphasized.

## SOFTWARE DEVELOPMENT PROCESS

Software Development Process (also known as Software Development Life Cycle (SDLC)) is the procedure taken in the development of a software product. It is often considered a subset of Systems Development Life Cycle. According to Somerville [1] it is a set of activities and the associated results which produce a software product. Software processes are very important in the software industry. Figure 1 shows a simple software development process.

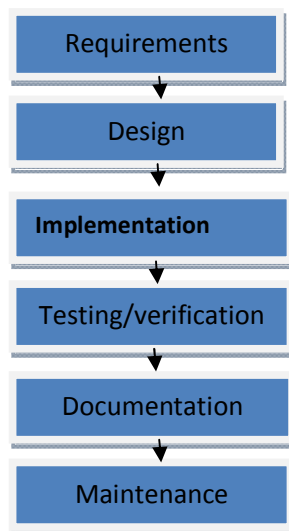


Figure 1 - Simple Software Development Process, **Source [7]**

## SOFTWARE PROCESS MODEL

This is a simplified description of a software process represented from a particular perspective [1]. The models are used to aid the development process. Examples of these models include; waterfall model, V model, Prototyping model, evolutionary development, agile model and spiral model.

Software development has increased in complexity and size and informality has proved to be a failure to many of the software products. Software processes can be beneficial for the individual, can benefit the project as well as benefiting the entire organization. For the individual, processes reduces interruptions and surprises because one can work on his or her

planned task well, there is an increased awareness as well as improved morale. Benefits for the project includes; improved ability to estimate tasks and improved chances of on-time delivery of a quality product. Organisations can benefit from the software processes in the sense that there is an improved product stability, there is reduced dependence on key individuals and improved morale.

Failure to use these standards and models can lead to products of poor quality, delayed projects and can even lead to complete project failure. Lack of adoption of these standards can lead to project delay as well. The literature shows that modern software products are accurate, faster, and cost effective. This is because they are engineered to improve the quality of the product [8].

## **CHALLENGES TO SOFTWARE PROCESSES**

Software Development life cycle faces a lot of challenges during each phase, the biggest challenge being where to start. The worst situation being starting a project with new employees, who don't have domain expertise, unproven technology and a challenging deadline. Along with technical challenges any situation might hinder a software development plan and put management in a risky and terrible crisis, products might overshoot both cost and time estimations but still ending-up in poor quality. Also products may not meet requirement specifications as defined by the consumer and finally lead to a business failure [8]. This clearly shows the danger of not having standards in place.

## **SOFTWARE PROCESS IMPROVEMENT (SPI) MODELS**

SPI models are quality models or standards for software development in order to produce quality software. Examples of these models include: Capability Maturity Model (CMM) [9], Capability Maturity Model Integration (CMMI) [9], ISO 9000-3 Series of Standards [6], the SPICE model etc.

## **WHY SOFTWARE STANDARDS**

Software standards play an important role in quality management. These standards define the required attributes of a product or a process. There are different types of standards and these are: international standards, National standards, organizational standards and project standards. Standards are important in the sense that: i) they help to avoid past mistakes – if care is not taken people can be repeating the same mistake over and over again. ii) Standards are framework for defining what quality means in a particular setting. iii) They provide continuity. vi) Process standards define how the software process should be endorsed. v) The other importance of standards is that, for the product they define the characteristics that all components should exhibit a good example is a common programming style and document structure [4].

As pointed out in [10], the choice of a particular model depends on the type of organization, its business needs, its business goals and the size of the organization. For the small organizations, the cost of investing on expensive process improvement program is often a challenge whereas large organizations can easily afford expensive SPI program.

In a developing country like Botswana, using and adopting software process improvement, assessment methodologies, technologies and other quality methodologies has been a challenge. Research also suggests that Small Medium Enterprises (SMEs) face a challenge to use traditional Software Process Improvement (SPIs) such as CMMI and SPICE [11, 5, 12, 13]. Hence the need for a framework to help SME companies to adopt or use software improvement methods. MARES [14], suggested a Methodology for Software Process Assessment in Small Software Companies in Brazil.

The success of the software development company in any environment is determined by the number of factors including the software process models used. However many software organizations in a number of countries are still encountering problems and as a result there is a need for process improvement [14]. It has also been noted that the use of software development standards, including SPI, may have been neglected by many software companies

in many countries. “Indeed many organizations still rely on adhoc process and do not take advantage of software engineering methods in their software development” [1 ]. For this reason special studies have been conducted in Brazil, Malaysia and Australia to find out if the companies in these countries make use of acceptable standards such as International Organization for Standardization/ International Electro technical Commission (ISO/IEC) 9000; the results however show that more software companies are yet to undergo the implementation of Software Process Improvement [12]. In the United Kingdom, 15/85 software companies have undergone software process implementation [12], while in Malaysia only 40 software companies out of more than 94 companies are certified with CMMI [15].

## **1.2 Motivation**

The neglect of SPI models by many software companies makes it necessary to find out the extent to which the process improvement models (specially CMM/CMMI) are applied in Botswana software companies.

Some developed countries have a large number of their software companies which are not registered with international standard [4]; and this poses a challenge to software companies in Botswana as well.

## **1.3 Problem Statement**

Software processes and process models are often ignored in many software organizations [12]. In addition there is a great challenge in choosing which process models and which process improvement models to use among many process models and Software Process Improvement methods. The inability to choose which process model and which process improvement approach to adopt by organizations is still a challenge that software organizations face especially in Botswana.

Although there are few software companies in Botswana, the problem of selecting appropriate Software Process Improvement should not be overlooked. This is because many of these companies may not be familiar with using available SPI models. The lack of proper and optimal use of software processes and software process models could lead to failure. It is therefore

necessary to find out how companies in Botswana generally carry out their software process development in order to improve it for successful outcome. The cost of choosing a wrong model may impact negatively on the small organization as the organization may not have prepared for the cost involved. This usually occurs as result of poor decisions by the project managers [16].

#### **1.4 Research Objectives**

The main objective of this study is to assess software processes and process improvement models in use in Botswana in order to develop an appropriate framework for selecting software process improvement models for SMEs in Botswana. Specifically the objectives of the study are:

- To find out the capability status of the software companies in Botswana.
- To find out how software companies apply software processes and process models.
- To suggest a framework for selecting appropriate SPI for Small Medium Enterprises in Botswana.

#### **1.5 Research Questions**

The following research questions are addressed in this study;

- Do Botswana software companies follow software process improvement models (SPI)?
- Do software companies in Botswana apply CMMI standards?
- How big or small are software companies in Botswana?
- Are the available SPI models relevant to software companies in Botswana?
- How do software companies in Botswana ensure quality?
- How do software companies in Botswana measure quality?
- Do software companies in Botswana consider software process models as important?
- How do companies choose SPI approach they want to follow?

## **1.6 Significance of the Study**

This research is very important for software companies in Botswana for a number of reasons. The main reason of carrying out this research is to find out if software companies in Botswana meet the requirements of CMM/CMMI. This research highlights some of the reasons why software development is not progressing much in the country and proposes a framework for selecting appropriate SPI in Botswana. The research also looks at the major areas of failure in many software companies and how they can improve so that they can compete with other companies abroad.

## **1.7 Chapter Summary**

This study has 5 chapters. *Chapter 1* is about the background of the topic under discussion i.e. software processes and process models. It covers the definitions of important terms used in this study. It also covers motivation, problem statement, research focus, research objectives, research question and the significance of the study. Chapter 2 introduces literature review on different models used in the software industry. Chapter 3 sets out the methodology used to gather the data, while chapter 4 covers results and discussion and the discussion of the proposed model. Finally chapter 5 covers the summary and the conclusion of this research work.



## CHAPTER 2: LITERATURE REVIEW

### 2.0 INTRODUCTION

This chapter takes a review of software processes, process models, software process improvement and international standards. It also looks at the related work and the types of Software companies. Software process, software process models, software process improvement models, SPI for Small Medium Enterprises, software process assessment and software estimation technique are common issues that require adequate study and understanding in software engineering in order to succeed in quality software products.

### 2.1 Software Process Models

A process model can be defined from two important perspectives; i.e. from the business point of view and the software engineering point of view. Software Engineering Institute (SEI) defines process as the organisation of people, automated support, procedures and standards into work activities designed to produce a specific results. Process operates by integrating people, tools and rules as shown in the figure 2 below [17], [18].

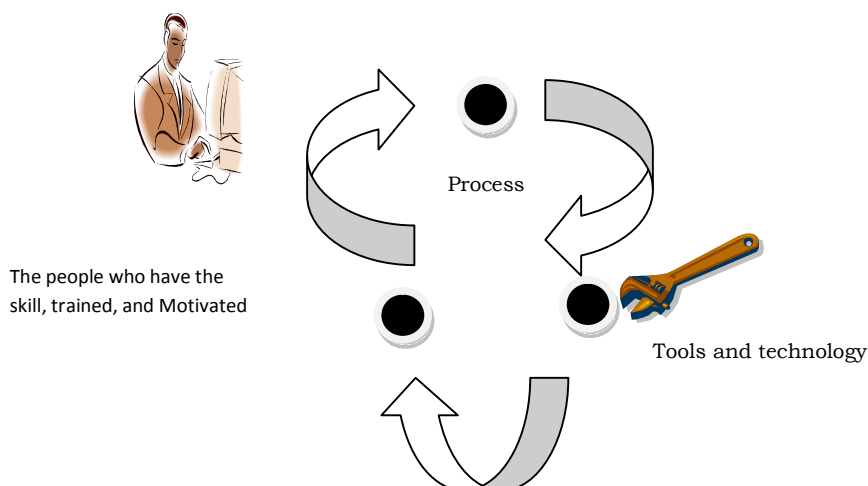


Figure 2 - Software Process, **Source [18]**

According to Watts [17] software process is a set of activities, methods and practices that guide people in the production of a software.

SPI has influenced many companies and organisation to put effort in improving processes for software development in order to develop quality software [18]. In order to have a good guidance, with the systematic coordination and control of the tasks that must be performed to achieve the end product and the objectives, processes models are required. According to Jones and Barlett [19], Processes models help in defining the following:

- The tasks to be performed
- The input and the output from each task
- The precondition and the post condition of each task.
- The sequence and how all these tasks flow.

The four points listed above are the important elements in software development. No matter the software process model, the above mentioned points are to be addressed. Having a proper model will be very useful. The question may still be why worry about all these models and processes. In every event, project or work, there must be a proper outlined way, a pattern on how things are to be done. If we view the processes as a prescriptive roadmap for generating various intermediate deliverables in addition to executable code, then it is necessary to follow the development process. These additional deliverables includes: a design document, a user guide, test cases and so on.

Software processes can be modeled through different models and which model to use will depend on various conditions. In this section, the process models are described and recommended cases where each can be used are mentioned.

### 2.1.1 Waterfall Model

This is one of the oldest models, which was introduced in the 1970s by Winson W. Royce. The name of the waterfall model is derived from the process it represents: tasks occur sequentially one after the other, with output from one task dropping into another [1]. Figure 3 represents the model picture. Although this method may be viewed as old and with a lot of shortcomings, the model has a lot of positives. Waterfall model has the following advantages:

- Simple and easy to understand and use.
- It is easy to manage due to the rigidity of the model, each phase has specific deliverables and a review process.

The model has disadvantages as well:

- Not many real projects follow the sequential flow that the model proposes.
- Customers normally have difficulties in writing all the requirements explicitly.
- Customers will have to wait until the working or the final version is produced late in the project time-span

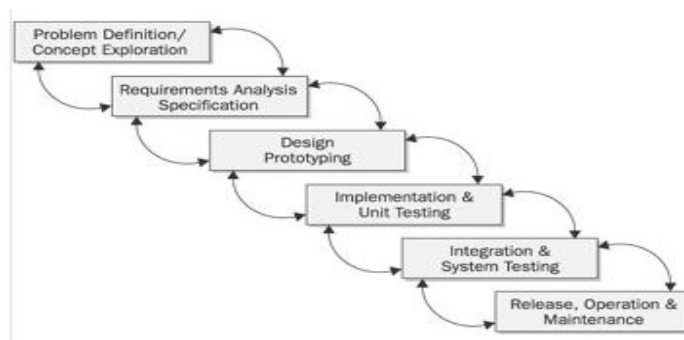


Figure 3 - Waterfall basic model, **Source [20]**

## **Recommended Projects for Waterfall**

- Waterfall model is most appropriate when the requirements can easily be defined, are well known, stable and not changing. It works better where requirements need to be defined early.
- Waterfall model is appropriate for large, expensive and complicated projects. This model keeps track of everything happening in a sequential manner.
- The model is also good for organizations which have unstable team, new staff members will be able to catch up easily.

### **2.1.2 Rapid Application Development (RAD)**

The Rapid Application Development (RAD) approach focuses on developing a sequence of evolutionary prototypes which are reviewed with the customer in order to ensure that the system is developing towards the user's requirements and to discover further requirements. The process is controlled by restricting the development of each integration to a well defined period of time, called time bore. Each timebore includes analysis, design, and implementation of a prototype. It is an incremental software process model that puts more emphasis on a short development [18]. Figure 4 shows the activities that are followed by Rapid Application Development (RAD).

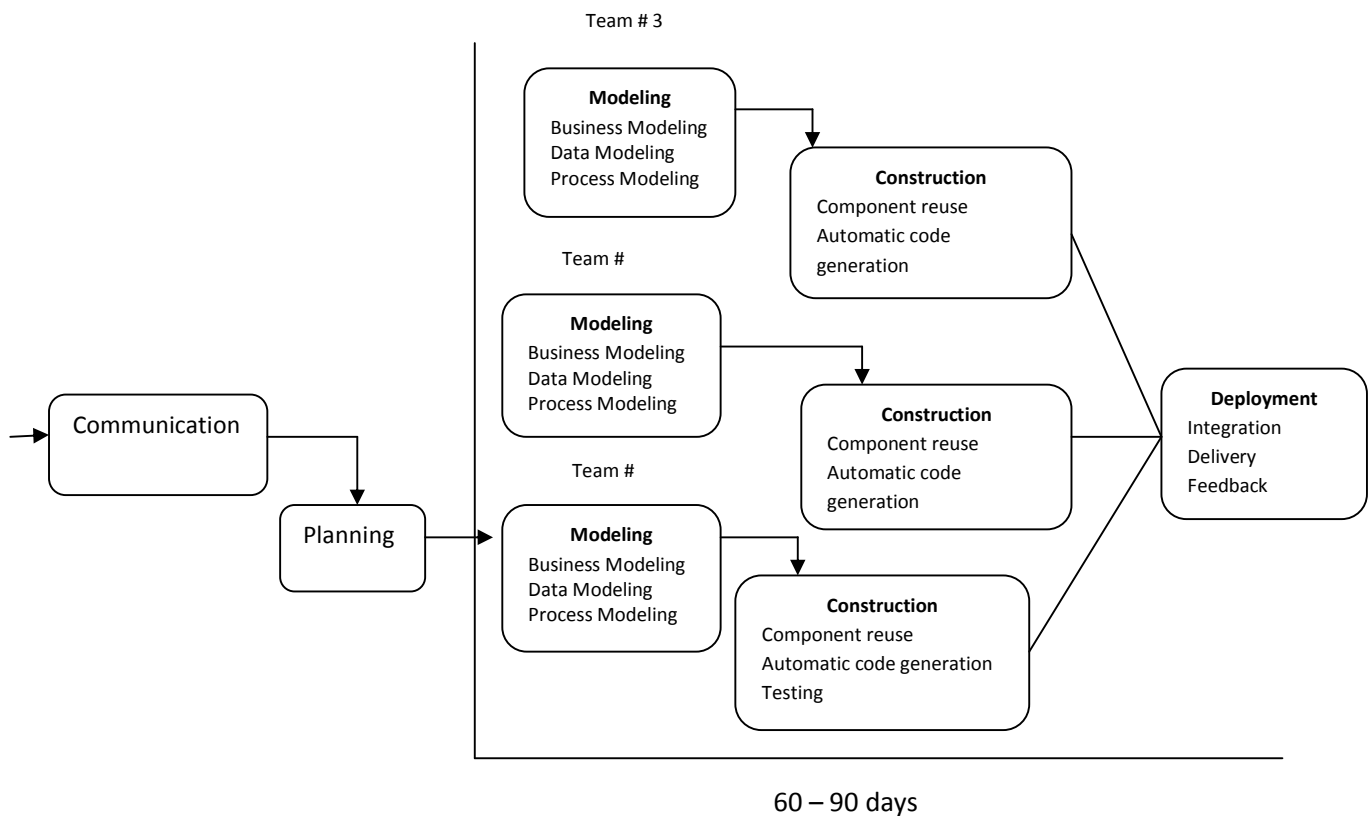


Figure 4 - generic framework activities for RAD- Roger S. Pressman 2005, Source [21]

Rapid Application Development follows the following generic framework activities:

**Communication:** This help in understanding the business problem and the information characteristics that the software must accommodate.

**Planning:** This is needed because there are multiple software teams that are involved, who work in parallel on different systems functions.

**Modeling:** Three major phases are involved here; business, data, and process modeling. This establishes design representations that serve as the basis for RAD’s construction activities.

**Construction:** Emphasizes the use of pre-existing software components and the application of automatic code generation.

**Deployment:** establishes a basis for subsequent iterations if required.

### **Recommended Projects for Rapid Application Development Model**

- As the name suggests the model is very good for projects that are needed as quickly as possible.
- It is also useful when the requirements are not clear.

### **2.1.3 Evolutionary Development Models**

These models are based on the idea of developing an initial implementation, exposing it to the user's comments and refining it through many versions until the adequate or desired system is developed. The specification development and validation activities are carried out concurrently with rapid feedback across the activities. This model is intended to explore the customers' requirements through close interaction until the final system is delivered. The development starts the parts of the system that are well understood. As the customers propose new features of the system, the developer adds them to the current system [21].

The immediate disadvantage of this model is a high risk of developing a system with poor structure. This is the fact that the system was not well planned for. The other problem with this approach is that the progress is not visible. There is nothing much to produce as a deliverable to measure the progress.

### **Recommended Projects for Evolutionary Development**

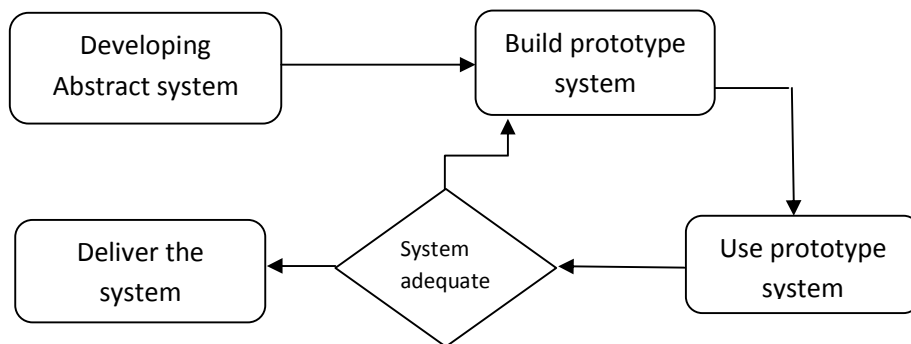
Projects for Evolutionary Development are recommended:

- Where requirements are not clear and more are still expected from the customer.
- Where users are so eager to have the system running.
- In changing user requirements.

- When the model is ideal for small and medium-sized systems (up to 500,000 lines of code).

**Exploratory development:**

This type of model is used mostly when the objective is to explore user requirements and deliver the final system. Development starts with parts of the system which are understood. As time goes on, the system will evolve as new features are added which are proposed by the customer. This helps the developer to be close to the customer and minimize the channel of developing a system that will be regular at last.



**Figure 5 - Exploratory Development, Source [21]**

**Incremental Model**

The model is more of a waterfall model with some modifications in it. Large software projects cannot be easily managed or developed as a single component. It is much easier when they are divided into sub components i.e. to subdivide the system into smaller components, which may thus be developed incrementally and iteratively. With incremental model, components are developed in an overlapping fashion. It is after these that components will be integrated into one single system, which will lead to final system test.

### ***Recommended cases for using the model***

- This model is best used for AI (Artificial Intelligence) systems which are difficult to specify. Best used when requirements are not clear.
- When developer wants users to have the feel of the system well in advance.

### **Advantages**

- The model offers the accelerated delivery of the system. Most software companies are usually faced with the challenge of delivering system quickly. Some customers are not much concerned with functionality or long term software maintainability e.g. the systems that are used in short project or used for survey.
- Model offers a good opportunity for user engagement with the system: users are more likely to make commitment to see the system work when they feel they are much involved.

### **Disadvantages**

When exploratory development is used, a number of problems can arise after the system is delivered. These problems include;

- Management problems: management will experience problems when they want to understand the way the system was developed.
- Maintenance problems: since the system was developed without following proper steps, it will be difficult to know how to maintain the system
- Contractual problems: The nature of the way the system was developed makes it difficult to come up with contract terms.



## **Throw Away Prototyping**

The model is used mainly when the objective of evolutionary development is on understanding the customer requirements so that better requirements definition for the system is acquired. The main or a principal function of the prototype is to clarify requirements and provide additional information for managers to assess process risks. In prototyping, developers concentrate on experimenting with the customer's requirements that are poorly understood. Figure 6 shows a throwaway prototype model.

The prototype is usually thrown away when the requirements are well understood, and this can really waste the developer's time. A throwaway Prototyping has the following advantages:

- Requirements can be classified
- Quick delivery
- Customers don't wait for long before having a feel of their system

Throw Away prototyping approach has the following disadvantages;

- There are a lot of implementation compromises done by developers
- Customers can get too used to the working version and later resist the proper development when a developer proposes so.

Figure 6 shows a throwaway prototype model.

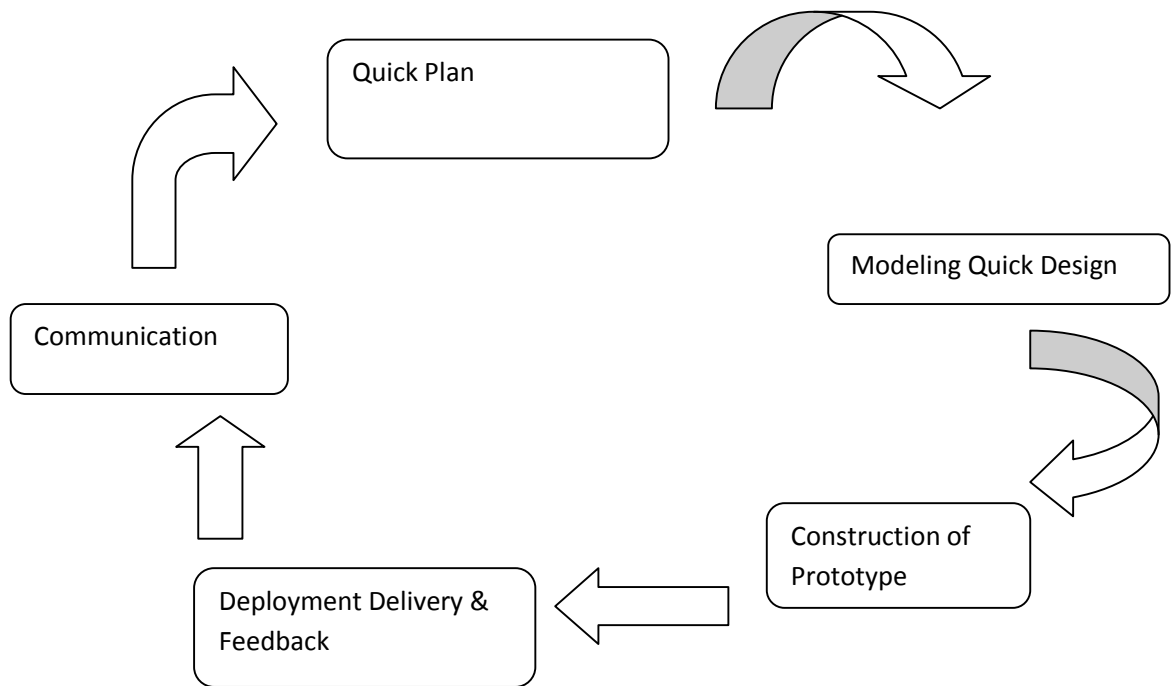


Figure 6 - Throwaway Prototyping Model, **Source [21]**

### Recommended Projects for Prototyping Model

- When the objective of the evolutionary development is on understanding customer requirements.
- When the objective of the evolutionary development is to gather better requirements

### 2.1.4 Spiral Model

The model was devised to address the weakness of the waterfall. This model was originally proposed by Boehm [22]. The model is an evolutionary software process model and it combines the iterative nature of prototyping with systematic aspect of waterfall model. Boehm describes the model in the following manner:

The spiral development model is a risk-driven process model that is used to guide multi-stakeholder concurrent engineering of software intensive systems [23]. It has two main distinguishing features. One is a cyclic approach for incrementally growing a

system's degree of definition and implementation while decreasing its degree of risk [21].

It focuses on addressing risks incrementally by repeating the waterfall model cycles.

The steps followed include:

- Determine objectives – the current situation of the business is analyzed (problem definition), definition of business process done, specification of constraints and generation of alternatives to the problem.
- Evaluate alternatives – the analysis of risks associated with the problems at hand is performed. The prototype of the system is also done here. The user or customer will also analyze the prototype and give the developer the feedback which is used to modify it until it meets the user's need.
- Development – this includes the actual design of the final product, coding, unit testing and integration. Figure 7 shows the diagram of a spiral model;

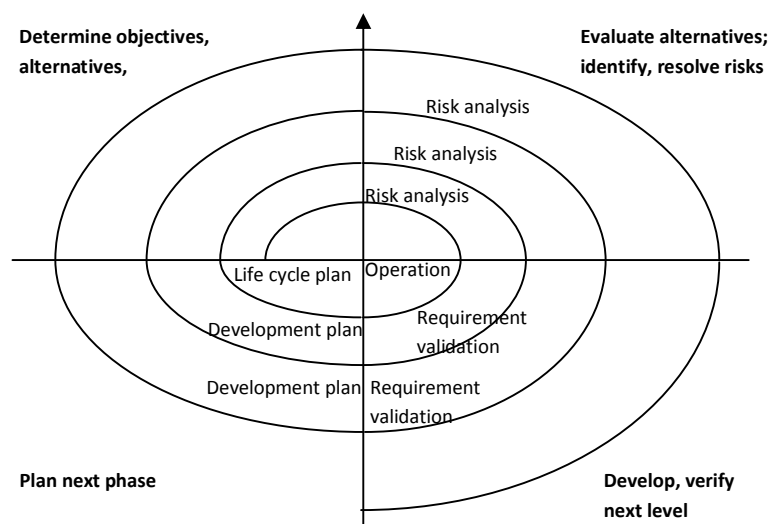


Figure 7 - Basic Spiral model [Boehm, 1985]

### **Recommended Projects for Spiral Model**

- When the requirements changes frequently and are poorly understood.
- When the requirements suggest the system complexity
- When the system will be used for a short time e.g. a questionnaire kind of system
- When there is early functionality as requirements
- Real-Time and safety-critical system
- Where risk avoidance is of high priority as a system.

### **2.1.5 Re-use Oriented Development**

Software reuse is common in most systems. Some of the design or code that is needed may be similar to those developed before. Some people opt to use the existing solutions. This is usually done by modifying the component and incorporating it into the system. In short the model uses the existing components or COTS (Commercial-off-the-shelf) systems to develop the process by systematic reuse. There are four stages that are generally covered. The four stages are as follows:

- Component Analysis:* This is the stage where the search and the specification of the suitable components, which will be used to implement the requirements is done. This should be done after requirement specification has been done and understood.
- Requirement Modification:* The requirements are modified to site the existing components
- System Design with Reuse:* At this stage, system developers have to design a new framework of the system or reuse the existing framework

- iv) **Development and Integration:** This is where the software is developed, and usually such a system is not available in the market. Components and COST are integrated to create the system.

### **Recommended Projects for Reuse Oriented Development Model**

- When the system to be developed has some commonalities with the other systems which have been developed before.
- In organizations that specializes in developing software modules for other organizations

### **2.1.6 Agile Software Development**

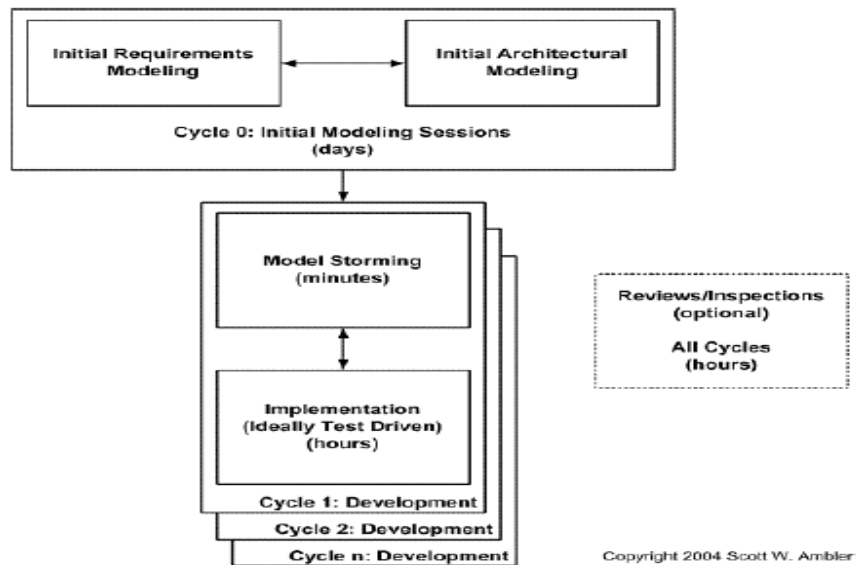
In agile software development, each phase of development is referred to as iteration where the output of each interaction is the working code that could be used to evaluate or get more changing user requirements. The model assumes that customers do not know or cannot clearly bring forth their requirements. This methodology uses iteration, use of small teams. Quick prototypes are developed to describe the problem to be solved. The team defines the requirements of the iteration, develops the code. The team also defines and runs the integrated test scripts and the users are asked to check if the results are according to their taste.

### **Recommended Cases for using Agile development methodology**

Agile software methodology is applicable in most of the IT project but it is best used in cases where:

- When needed changes are always expected in the system and are needed to be implemented. This methodology offers a freedom to change and this is very important. The frequent increments that are produced offer a great opportunity to implement changes at very minimum cost.

- When planning is not a major activity in the project. Since planning is done at a minimum level, projects which do not need much planning can be appropriate. Figure 8 shows agile software development lifecycle.



**Figure 8 - Agile Software Development Lifecycle, Source [22]**

The importance of a process cannot be overemphasised. Processes are very important in software engineering. Processes need to be modelled in order to successfully use them and that is why we need process models. Furthermore processes need to be improved over and over again for them to be highly effective both in the business world and in software engineering field. Software processes are always under improvement and this is the exercise that will continue to be experienced. Software models are also going through improvement as well. There are a number of software development methodologies that has being derived from waterfall model. SPI play an important role in the improvement of the processes. A wise application and adoption of the right SPI model can really bring good results in an organisation. Furthermore the use of the right Process Model and SPI in SMEs can as well as bring tremendous results.

## **2.2 SOFTWARE PROCESS IMPROVEMENT (SPI) MODELS**

Software Processes Improvement models may be defined as integrated collection of procedure, tools, and training for the purpose of increasing product quality or development team productivity, or reducing development time [18]. Although there is no "ideal" software process [1], there is a lot of scope for improving the software process in many organizations. Some companies/organization argue that they are all using process models but the question is how effective are they using them? Do they provide opportunity for the improvement? Software Process Improvement can help these companies to improve their Software Process.

SPI can be implemented in a number of different ways. These may be done through process standardization where the diversity in software process in an organization is reduced. Software process improvement is important here in the sense that, it can lead to improved communication, reduced training time and makes automated process support more economical.

Standardization is an important step, and actually a first step in introducing new software engineering methods and techniques and good software engineering practice.

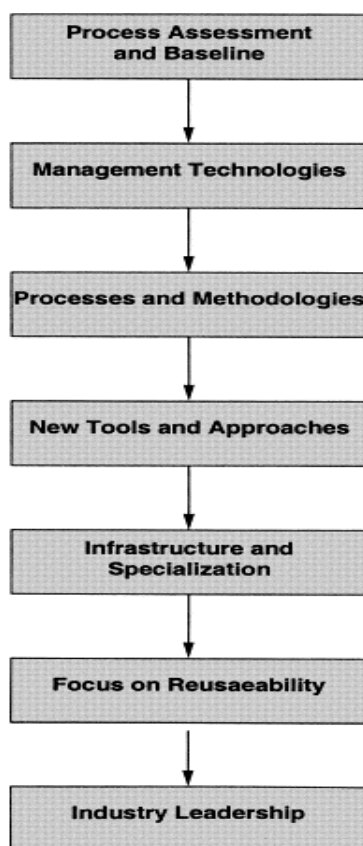
When we consider process improvement, we must understand that there is a strong relationship between quality of the developed software product and the quality of the software process used to come up with the product.

### **2.2.1 SOFTWARE PROCESS IMPROVEMENT PROCEDURE**

The *first improvement stage* in carrying out Software Process Improvement is the training of the management [10]. The stage involves training of managers to equip them with the knowledge of the critical technologies which are required for Software Process Improvement.

The *second improvement stage* deals with the approaches for dealing with user requirements, design, development and issues on quality control.

The *third stage* deals with the implementation of new tools and technologies. After the company has targeted some processes and methodologies to be improved then the organization needs to acquire improved tools and explore new technologies. The optimal sequence for software process improvement can be represented as in figure 9.



**Figure 9 - Process Improvement Approach**

**Source [7]**

All these stages are very important in the improvement process as explained by H. Saiedian and Chennupati [10].



## 2.2.2 Capability Maturity Model (CMM)

The CMM model was initiated in the US in the mid-1980s after the Software Engineering Institute (SEI) has initiated a study on ways of assessing the capabilities of contractors. The model then became very influential in convening the software engineering community in general to take process improvement seriously. The model has five levels. The levels show to what extent has the organization gone in terms of following right procedures and standards in software development. The levels are:

- i) Initial level
- ii) Repeatable level
- iii) Defined level
- iv) Managed level
- v) Optimizing level

These levels will be explained in more details under the Capability Maturity Model Integration (CMMI) model section [section 2.2.3]. The original Capability Maturity Model (CMM) was revived and adopted in 1993. This was influenced by the fact that the original CMM was said to be imprecise. CMM guides software organization on how to gain control of their processes for developing and maintaining software.

Each level in CMM needs to reach the Key Process Area (KPA) which is composed of key practices that contribute to satisfying its goals. When a company strives to reach higher level of maturity, this will result in capability increase of the organization process.

### **2.2.3 CMMI (Capability Maturity Model Integration)**

The Software Engineering Institute (SEI), describes the Capability Maturity Model Integration (CMMI) as a software process model which provides guidance for improving each organization process and the capability to control and manage the development [9].

CMMI is a process improvement maturity model for the development of products and services (ibid). It consists of best practices that address development and maintenance activities that cover the product lifecycle from conception through delivery and maintenance. This means the model integrates bodies of knowledge that are essential for development and maintenance, but that have been addressed separately in the past, such as software engineering. CMMI is actually an improved version of CMM. A number of people were using the model successfully in the areas of improving software processes and measuring the maturity of software process in an organization. This resulted in more attention to model-based process improvement in other areas and this brought about a number of other models being developed. This models includes:

- System Engineering CMM (SE-CMM)
- Software acquisition Capability Maturity Model (SA-CMM)
- Integrated Product Development Team Model (IPD-CMM)
- System Engineering Capability Assessment Model (SECAM)
- System Engineering Capability Model (SECM)

These models were used by organizations and as time went on, some organizations that wanted to use these models were confused because each model was different.

This brought about a new model – CMMI, which is the integration of all these models.

Figure 10 shows how CMMI came to be.

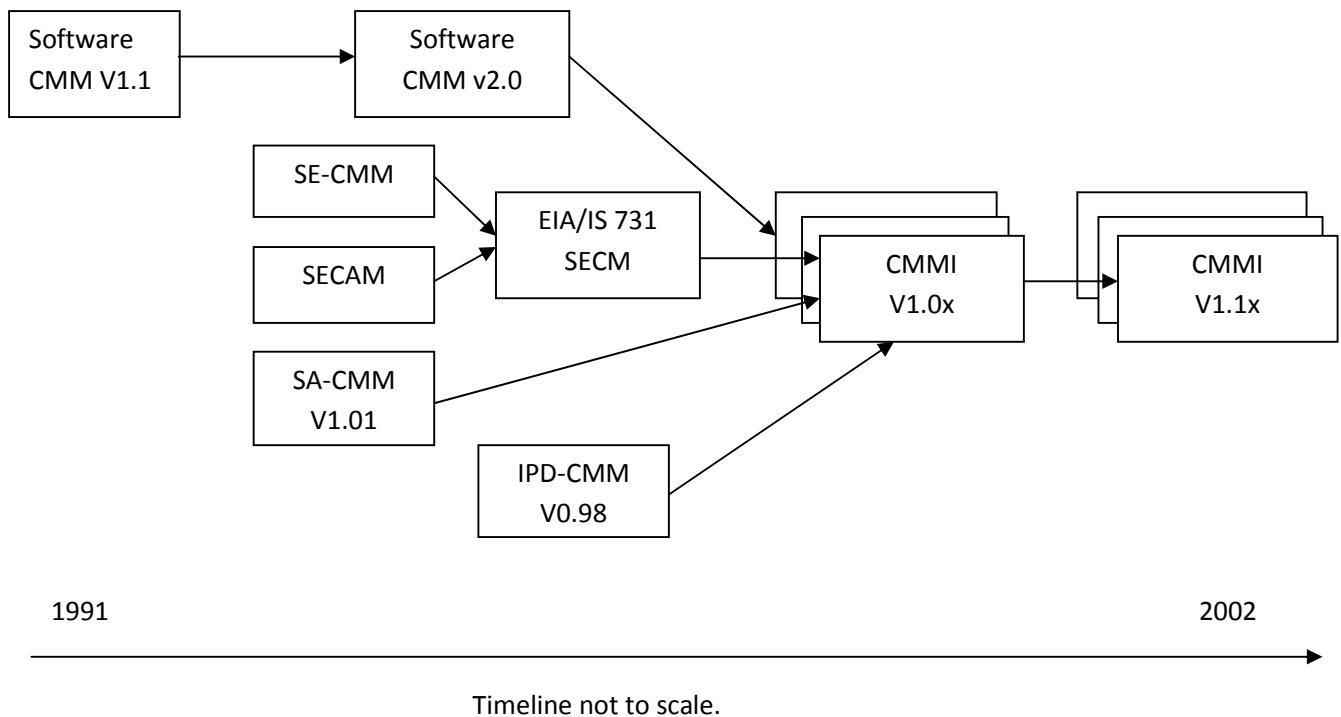


Figure 10 - Evolution CMMI model, Source [9]

### How to use the Model?

Different organizations approach the use of CMMI from two different views. These views are the assessment view and the process improvement views. Assessment view is more focused on what is the minimum required to satisfy the model and what is needed to pass the test. The test may be assessment, evaluation or SCAMPI. E.g. allowing engineers and developers perform a review of each other's work. These developers and engineers need to be trained in order to do the evaluation (ibid).

The process improvement view is based on what is the best thing to do for the organization and what is needed to improve the organization. For example; defining a

clear responsibility for quality related activity, establishing an independent quality group comprising of trained personnel (ibid).

CMMI model is used to cover many different organizational and project situations. The model is not specific but general. Some find it to be following an ambiguous style; this was done intentionally to cover many different organizational and project situation (ibid).

CMMI can be represented from two different ways. There is a staged representation and the continuous representation. Only staged representation will be addressed in this section. The staged representation focuses improvement on the process capability an organization can expect to attain. The staged representation uses five maturity levels as shown in Figure 11. This is the same structure used in CMM. A more detailed description is given in this section.

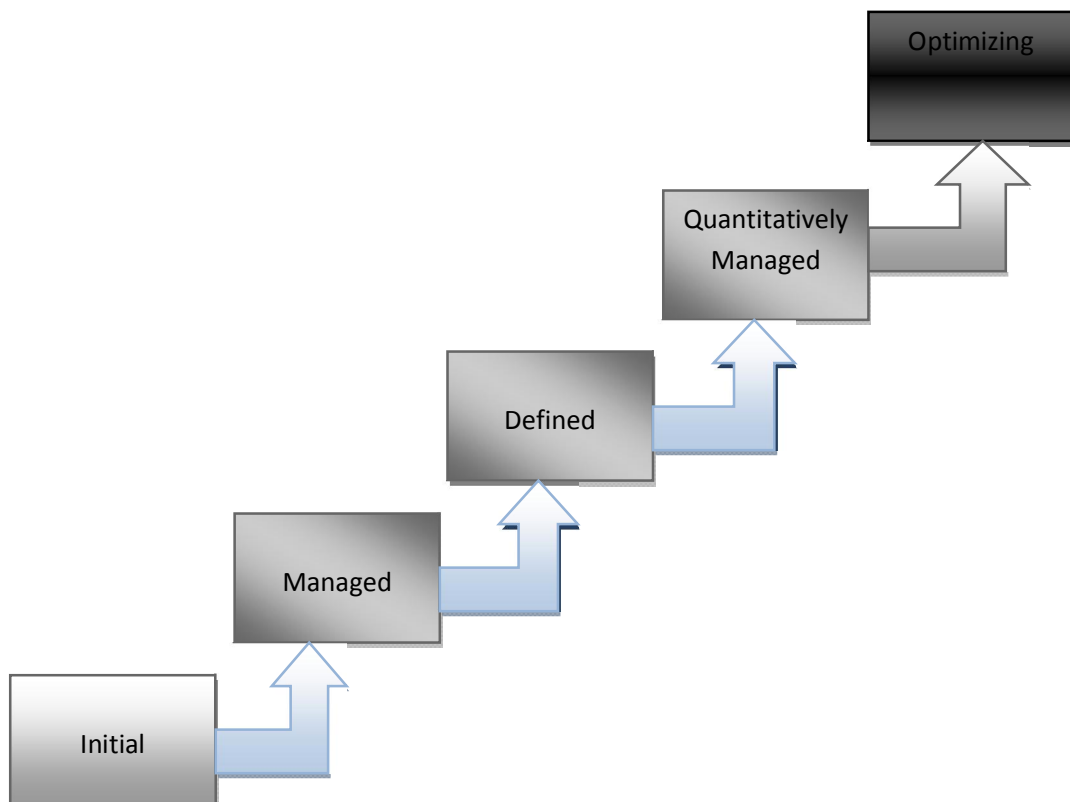


Figure 11 - CMMI Model, Source [9]

### **Maturity Level 1: Initial**

In this level, the organizations have no process in place. This is where the development is done in a chaotic and *ad hoc* manner. All this results in budgets and schedule been exceeded. Nobody can really tell how the quality of the project will be. This level is considered ad hoc because no procedure or structure is followed. One makes up as he or she goes along. CMMI model tries to avoid this kind of approach.

Any success that can be recorded in the organization of this level is as a result of few people that are smart and not that the organization follows proven processes.

### **Maturity Level 2: Managed**

This is the first maturity level that tends to address the issues of product improvement by following proven process. The level offers basic project management processes. At this level, the working teams build the foundation for the organization to become an effective service provider. This is done by institutionalizing selected; Project and work management, Support, Service Establishment and delivery processes. The service strategy is planned, work plans created and the work is monitored and controlled so that it will be done as planned. Furthermore, the institutionalization is achieved by applying adequate funding and resources, maintaining appropriate assignment of responsibility and authority, training people in the relevant fields, objectively reviewing the process, work products, and taking corrective action. The activities, status, and the results of the process are reviewed with appropriate level of management.

Also the work group, work activities, work products and the services are managed.

### **Maturity Level 3: Defined**

This maturity level is reached after the organization has achieved all of the goals of maturity level 2. The organization has a way of doing business, the methods followed are allowed under predefined conditions. The characteristics of the process are clearly stated. These are the purpose, the input, the entry criteria, activities, roles, measures, verification steps, and output and exit criteria. In this level institutionalization is achieved by successfully going through the institutionalization factors for maturity level 2 as well as achieving the following; establishing the description of the defined process, establishing a plan based on the description of the defined process, even performing the process just as it has been planned, collection all the information from

performing the process that can help in future improvement which includes; work products, measures, and improvement information.

There is a major difference between maturity level 2 and 3, this is of the fact that at level 3, processes are described in more details and more rigorously. The process in level 2 is characterized for projects in a reactive manner while in level 3, the process is characterized for the organization and it is proactive. The organization using this has an organizational identity. Organizational identity is a way of doing business particular to a certain organization.

#### **Maturity Level 4: Quantitatively Managed**

As expected, the organization at this level should have achieved all of the goals of maturity level 2 and 3. The main issue here is that the organization controls its processes by statistical and other quantitative techniques. Maintaining qualitative objectives, statistically stable and predictable, as well as maintaining a statistical understanding (to see if the process is capable of achieving the goals) is very important steps of achieving institutionalization at this level.

Metrics are used at this level to make decisions and to truly measure whether progress is occurring and the product improving.

#### **Maturity Level 5: Optimizing**

This maturity level is reached when the organization has achieved all the goals of Maturity levels 2, 3, and 4. The continuous improvement of the process, which is based on an understanding of common cause of variation within the process characterizes this level.

## **2.2.4 SPICE (Software Process Improvement and Capability Determination)**

SPICE (Software Process Improvement and Capability dEtermination) is a project which involves a number of organizations who have a joint effort to create international standards for software process improvement and it have operated since 1990 (unofficially) and since 1993 officially.

The effort was being carried out by a joint technical committee of the International Standardization Organization (ISO) and the International Electrotechnical Commission (IEC), and this is why the SPICE project culminated in new international standard referred to as ISO/IEC (International Organization for Standardization/ International Electro technical Commission) 15504.

(ISO/IEC) 15504 provides a framework for the assessment of software process. This framework can be used by organizations involved in planning, managing, monitoring, controlling, and improving the acquisition, supply, development, operation, evolution and support of software. SPICE or ISO/IEC 15504 covers a number of areas in software process improvement. It covers things like: SPICE model, processes, and process capability approach. The model also has products such as: i) Baseline Practices Guide which is similar to the SEI's Capability Maturity Model, ii) introductory guide, iii) Assessment instrument, iv) process assessment guide, v) Process improvement guide, vi) Process Capability Determination Guide and vii) Assessor Training and Qualification Guide. All these help the organization to have quality software products, guides the organization in the application of the SPICE products just for the purpose of coming up with improved software products.

SPICE also has a document suite, which contains nine different documents, which can be used in process assessment. SPICE offers steps for continuous software process improvement using the same SPICE components. Figure 12 illustrates these steps.



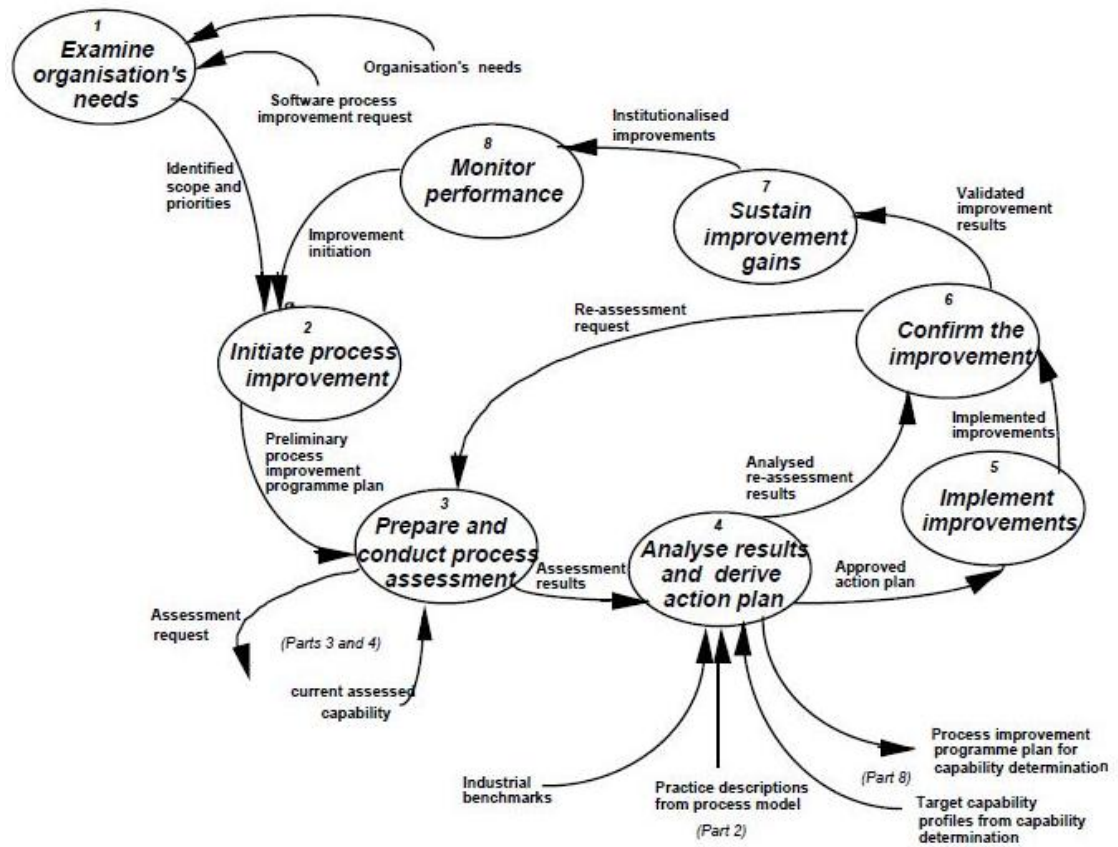


Figure 12 - Continuous Steps for software Process Improvement for SPICE, Marko Phyhajarvi, 2004, Source [23]

### 2.2.5 BOOTSTRAP

BOOTSTRAP was originally designed by using SEI model back in 1990's. It was extended on features from ISO 9001 guidelines [18].

This model is an European method for software process and improvement designed with the intension of speeding up the application of software engineering technology. BOOTSTRAP is good because it is suitable for all kinds of software organization. The model has proven to be successful in the industry probably because it does not

assume the strict adherence to a distinct key practice and allows the use of alternative approaches [25]. BOOTSTRAP has some important features which are:

- It has questionnaires for both site and project evaluation
- Uniform procedure and mandatory assessor qualification/Training
- Open questions
- Constructive instead of normative approach
- Immediate feedback and action planning

The model has the reference framework, the assessment procedure, the structure of the questionnaires, the rating and the scoring mechanisms as its main characteristics [26]. BOOTSTRAP model has proven to be suitable for all kinds and sizes of software development organizations. Figure 13 shows the process areas in BOOTSTRAP architecture.

Process Category	Process Areas	
Organization	Management Practices	
	Quality Management	
	Resource Management	
Methodology	Product Engineering	Requirements Analysis
		Definition
		Detailed Design and Implementation
		Testing and Integration
		Acceptance & Transfer
		Operational Support and Maintenance
	Process Engineering	Process Description
		Process Control
		Process Measurement
	Life Cycle Independent (Support) Functions	Project Management
		Quality Assurance
		Risk Management
	Technology	Technology Management
Product Engineering Technology		

Figure 13 - Process Area in BOOTSTRAP Architecture, Source [26]

### 2.2.6 IDEAL (Initiating, Diagnosing, Establishing, Acting and Learning)

IDEAL (Initiating, Diagnosing, Establishing, Acting and Learning) is also a Software Process Improvement model which is based on five phases as the name suggest. The model is an organizational improvement model that serves as a roadmap for initiating

planning, and implementing improvement actions [27]. Figure 14 shows the IDEAL model.

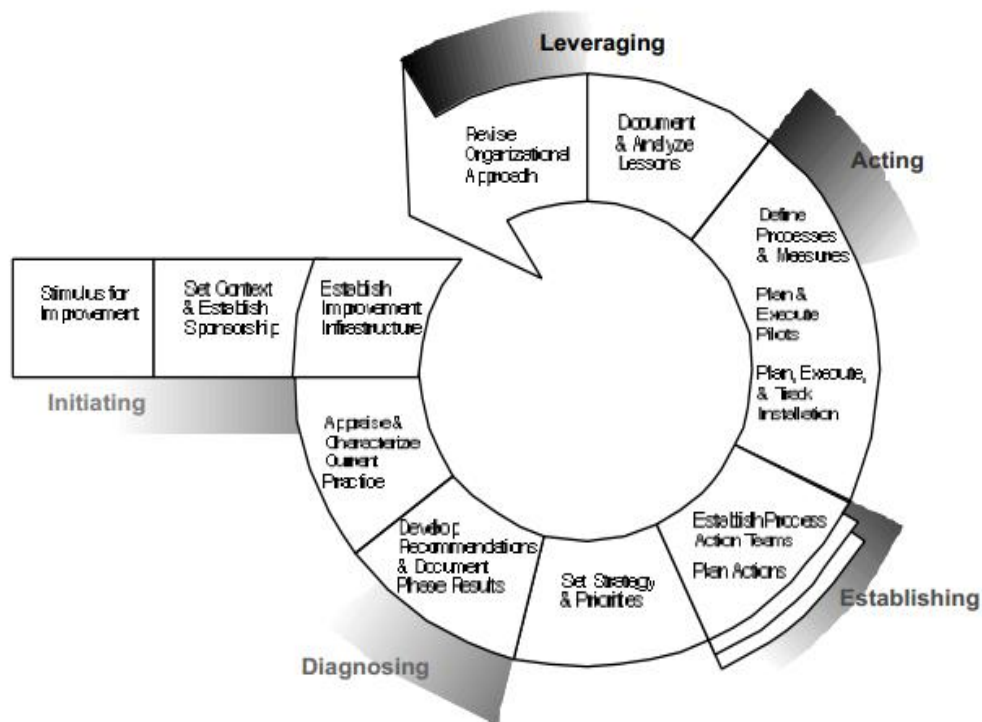


Figure 14 - IDEAL Model, Source [27]

The model has five important steps as shown in figure 14 above.

### Initiating

As the name suggest, this is the first step in this model. This is where the organization’s senior management first appreciates the need for software process improvement, commits to a SPI program, and also defines the context for SPI. The plan and the schedule for the initial SPI tasks are developed. Other things like functional elements, key interfaces and requirements are defined and agreed upon [28].

### **Diagnosing**

The organization's current software process baseline need to be diagnosed and understood so that a plan to achieve the business changes specified in organization's software process improvement goals can be achieved [28].

### **Establishing**

The strategic action plan for software process improvement is created as a critical SPI initiative. The previous strategic action plan is updated. This is done based on the past improvement efforts, past SPI actions and the budget.

### **Acting**

The improvements are developed at the Acting stage; they are put into practice and even deployed across the organization. They are also checked to see their impact.

### **Leveraging**

At this stage, after the organization has completed one cycle through IDEAL. There is a review that is done to prepare the next cycle. The tuning up of the SPI before the next cycle begins is also performed here.

## **2.2.7 TICKIT SCHEME**

TickIT is a scheme design to motivate developers and software companies to apply International Organization for Standardization (ISO) 9001. It relate directly to the requirements set out in ISO 9001. It is supported by United Kingdom and Swedish software companies. The scheme stimulates software system developers to think about:

- Quality in the context of the processes in software development
- How quality can be achieved and
- How quality management system can be achieved.

TickIT has some procedures to be followed – ISO 9001 procedures, certification practice – done by external agencies. It is relevant to all types of information systems which involve software development processes [29].

### **2.3 SOFTWARE PROCESS IMPROVEMENT FOR SMEs**

Small and Medium Enterprises (SME) often experience problems when they are to start using SPI. SMEs are companies with fewer employees. According to Fayad, Laitinen and Ward [30] companies with 50 or fewer employees are referred to as small [30]. These companies normally have few resources like finances, struggle to meet the deadlines. SMEs are challenged to do more with less, do it faster than their larger competitors.

There are a number of reasons why SMEs have a challenge in implementing the SPI:

- **Cost:** Very Small Entities or Small Medium Enterprises cannot bear the cost of implementing the Software Process Improvement programs [5].
- **Limited Resources:** A small enterprise does not have enough resources to implement SPI programs. This is because the employees are few Laporte, Seguin, Gisele, Boas and Buasung [31].
- **Deadlines :** Small Medium enterprises normally operate under strict deadlines[5]

As a result, a number of light weight models and methodologies have been developed to help SME to initiate Software Process Improvement. The following are the examples of these models;

### **2.3.1 OWPL GRADUAL FRAMEWORK**

Some refer to this model as a light weight [11] and it is also referred to as a gradual approach [5]. OWPL Stands for Observatoire Wallon des Pratiques Logicielles [32]. This model is used mainly for two objectives, which are: first, to create the awareness to the organization about the quality aspect of the software. Second, to initiate a continuous Software Process Improvement mechanism that will bring forth the results in no time, yet the results are tangible with a little burden on the use of the resources [9]. This model came about as a result of constant realisation that small and medium enterprises have a serious challenge with using the other SPI like SPICE, CMM, and CMMI etc.

The model is based on the suggestion that a key issue of success in any organisation is based on well defined goals. The goals should be structured in a hierarchical way [33]. OWPL model can easily be adopted by SMEs because it's able to initiate a continuous mechanism of SPI that will produce rapid, but tangible results within a minimum range of resources [32]. A very simplified questionnaire is also used to collect information about the current software practices in small structures to make people aware of the importance of software quality aspects. Figure 15 shows this model.

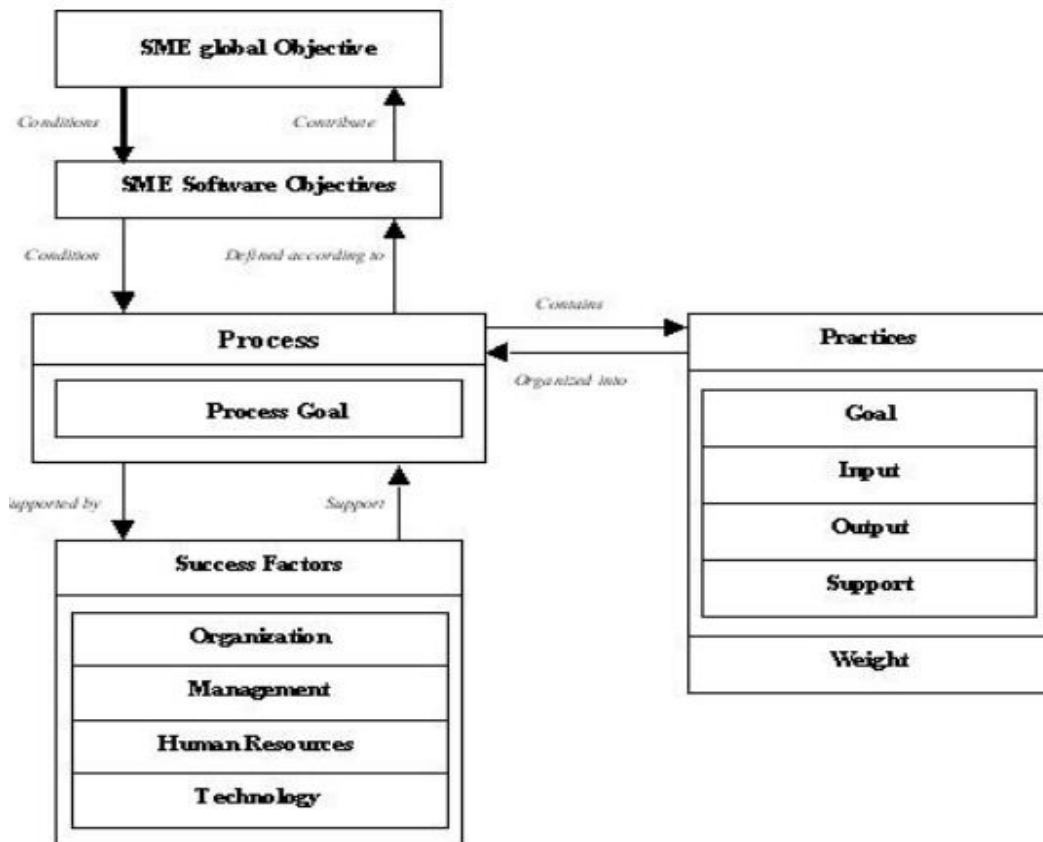


Figure 15 - OWPL model, Source [33]

This model is supported by a complete suite including: questionnaire, tool and template to perform an assessment of process, practices and success factors.

### 2.3.2 AN APPROACH FOR SOFTWARE PROCESS ESTABLISHMENT IN MICRO AND SMALL COMPANIES (ASPE-MS)

Another approach for software process establishment in micro and small companies is Approach for Software Process Establishment In Micro and Small Companies (ASPE-MS). This approach is defined by integrating and adapting existing approaches to behaviour and characteristics of small software companies [34]. This approach has phases like; diagnosing, strategic analysis, Definition, implementation, monitoring and control as well as post-mortem. Figure 16 shows the overview on ASPE-MS approach.



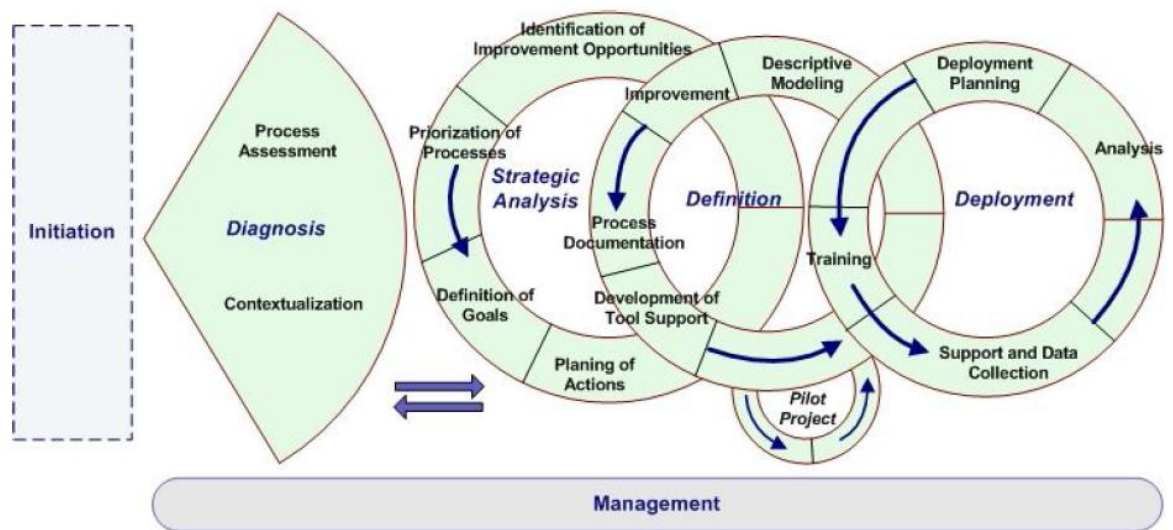


Figure 16 - Overview on ASPE - MSC Approach, Source [35]

This model uses the concept of process reference guide. This concept is considered a flexible collection of alternative processes, techniques and tools mapped to practices required by reference models and standards [36]. This can assist SMEs because the process reference guide can facilitate the improvement of the existing processes by indicating various alternatives to be tailored to the specific needs of the organization. The phases of the model are executed in an iterative and incremental way in order to establish and improve one or more process(es) within an organization [36].

### 2.3.3 iFLAP (Improvement Framework Utilising Light Weight Assessment and Improvement Planning)

The improvement is mostly used to evaluate single process area. There are three steps involved which are:

**Step1: Selection.** The assessment study needs to be carried in an organisation. During this study, it is important that the right people are selected as participants of the study [37]. This is done to allow the assessment and the improvement phases reflect the opinion of the whole organisation. This phase is performed by people who have a basic understanding of the company. The main aim of this step is to identify the

processes with the activities and the stakeholders involved. Workshops can be used to carry out the activities in this process [38].

**Step 2: Assessment.** Interviews are used to gather improvement issues from the organisation [37]. These improvement issues gathered need to be confirmed.

**Step 3: Improvement Planning.** This step also needs company's representatives to participate in it. They need to be selected. Improvements efforts focus on a limited number of issues at a time taking evolutionary steps.

iFlap can easily be adopted by SMEs because it utilises a light weight assessment. Therefore, fewer resources are used in the assessment. Figure 17 shows a process deliverable diagram for iFLAP framework.

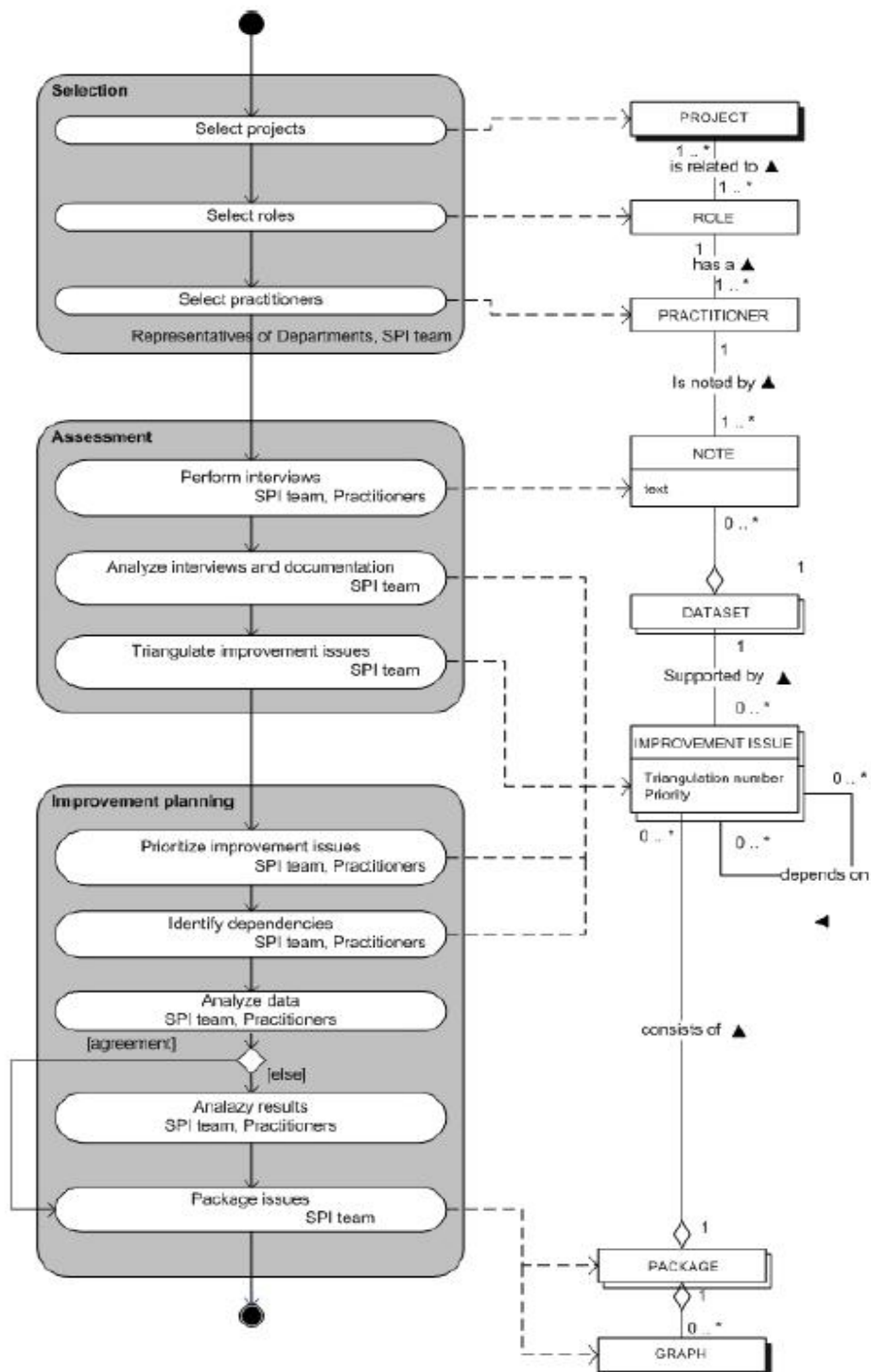


Figure 17 - A Process Deliverable Diagram for iFLAP Framework, Source [39]

### **2.3.4 An Approach to Software process Improvement for Small to Medium Enterprises (PRISMS)**

An Approach to Software process Improvement for Small Medium Enterprises (PRISMS) is an action research project, comprising of three researchers working alongside managers and developers in participating companies advising and assisting with the planning and implementation of SPI programmes, over a three year period [5]. Figure 18 shows the structure of this model. Some of the key features of this approach are:

- The existing informal process is examined, and if the resources permit, the explicit model is created
- Business goals are defined earlier
- A concurrent exercise of brainstorming, and/or questionnaire based survey, which involve all members, is carried out.
- Members of the research team carry out a tailored version of the CMMI assessment. This is done to help identify Key Process Areas (KPA)

PRISM approach is show in Figure 18.

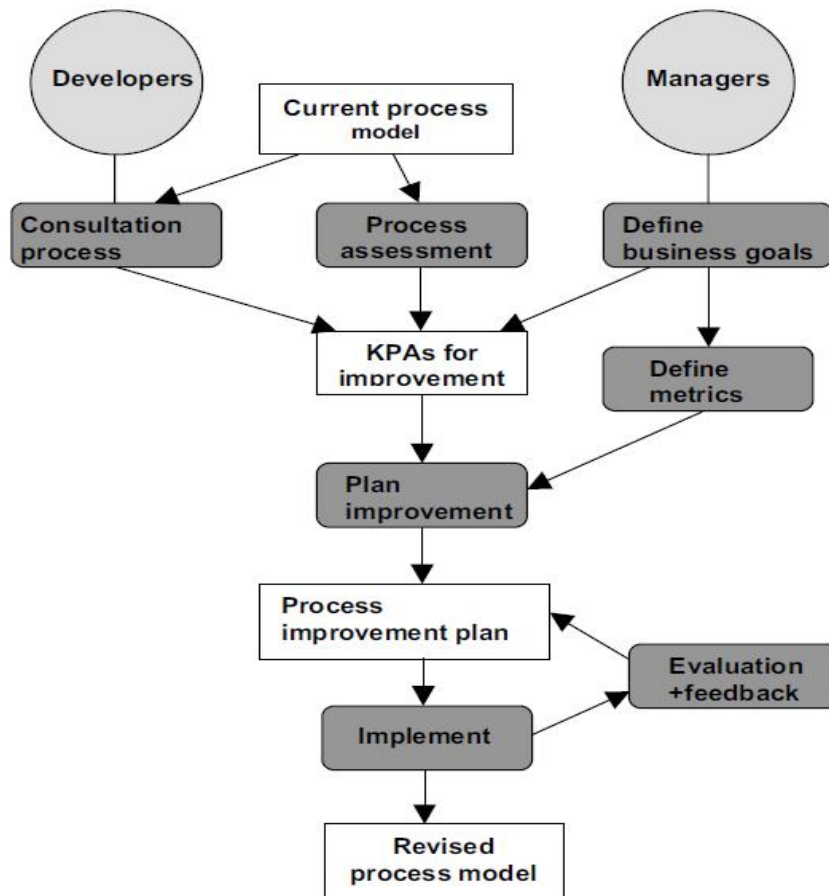


Figure 18 - PRISMS SPI Approach, Source [40]

As SMEs has limited resources, they can utilize self-assessment approach in PRISMS model. This is how the model can be friendly to most SMEs. Self-assessment approach is popularly known of its low cost, good accessibility and ownership of the results [41].

### 2.3.5 Software Process Metrix (SPM) Model

The model assist organisation in realising the importance of software processes. The model can be used to determine the practices that need to be worked on in high priority processes. In using SPM, Quality Function Deployment (QFD) is used. The voice of the customer is collected [42]. Software Process Metrix is shown in Figure 19.

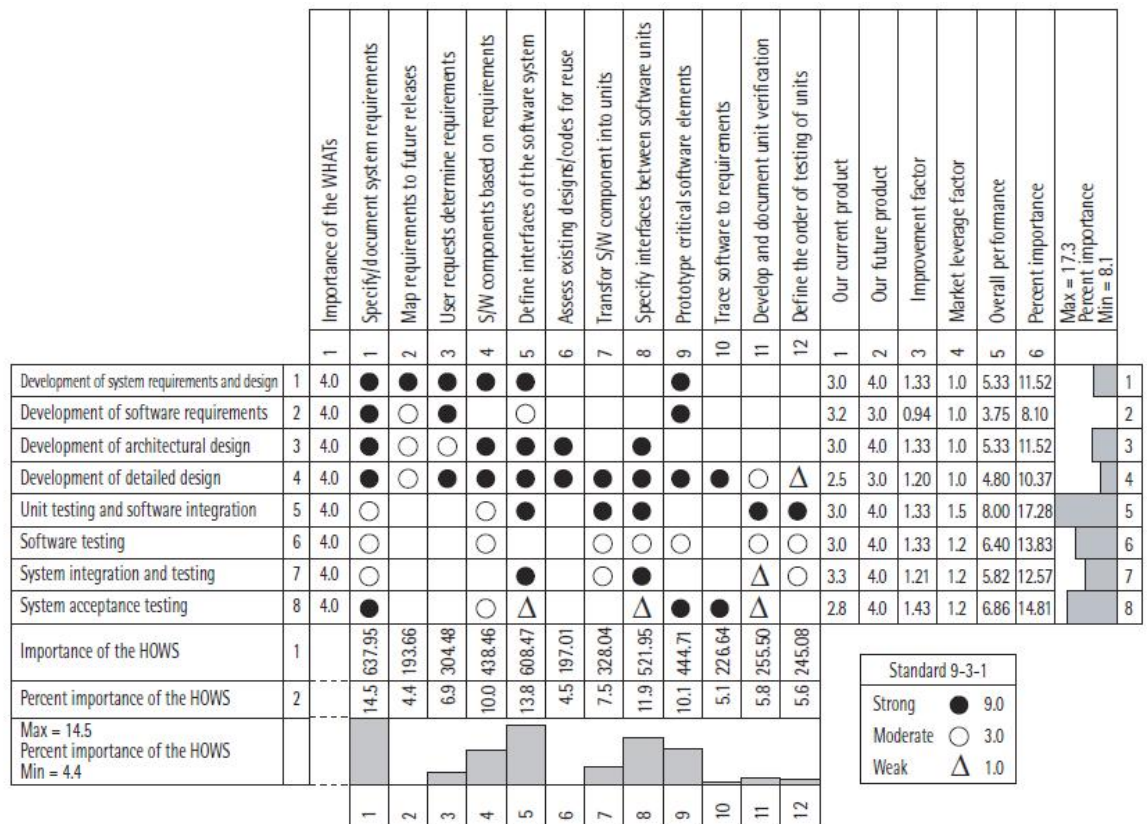


Figure 19 - Software Process Matrix, Source [43]

SPM also assist small medium enterprises to apply minimum effort. The SPM provides them with a generic section that has been completed previously and can be used in their company. A questionnaire is provided to assess the current performance, planned future performance and importance to the company for every process. From the company’s point of view, all they need to provide are the measurements for calculating the overall importance of the software process [44].

### 2.3.6 MESOPYME

MESOPYME has been developed as a SPI method that is focused on SMEs. The structure of MESOPYME model is divided into two parts. The first part is focused on

process assessment and is based on a two-staged questionnaire that is a tool to determine the current state of the process. The second part of this SPI model focuses on process improvement. It uses the action package concept to establish and also maintain a new process [6]. Figure 20 and 21 shows MESOPYME process improvement method. Figure 20 shows the action package of the model while Figure 21 shows a simple software process improvement.

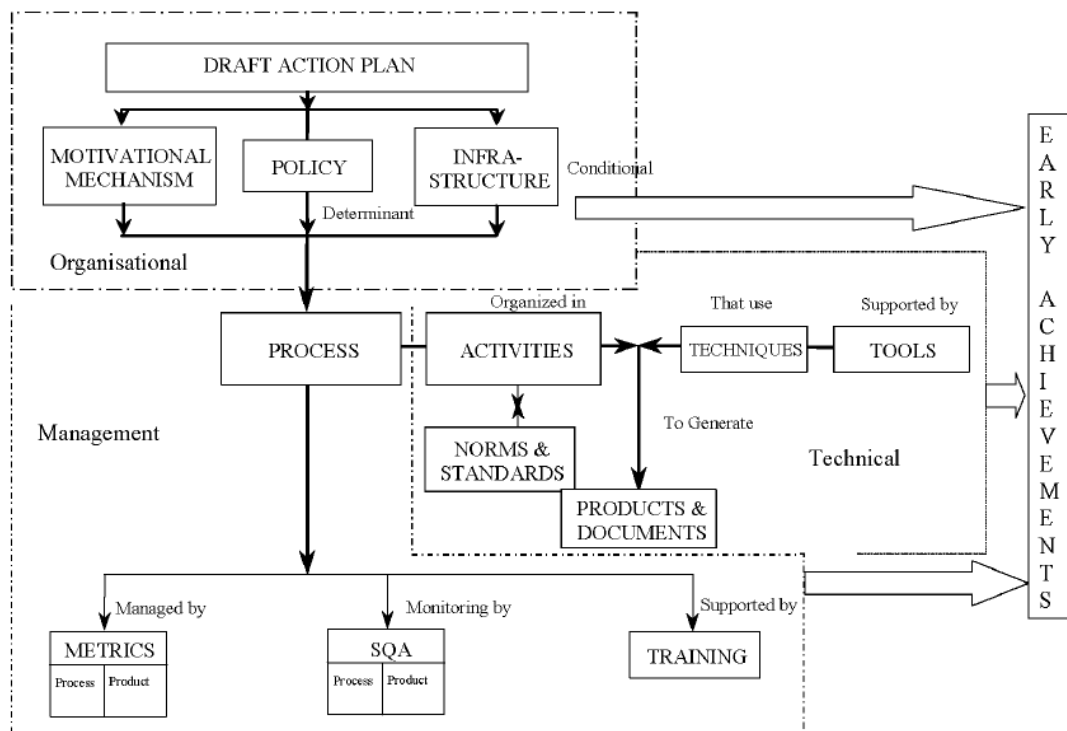


Figure 20 - Main Architecture of the MESOPYME Action Package, Source [40]

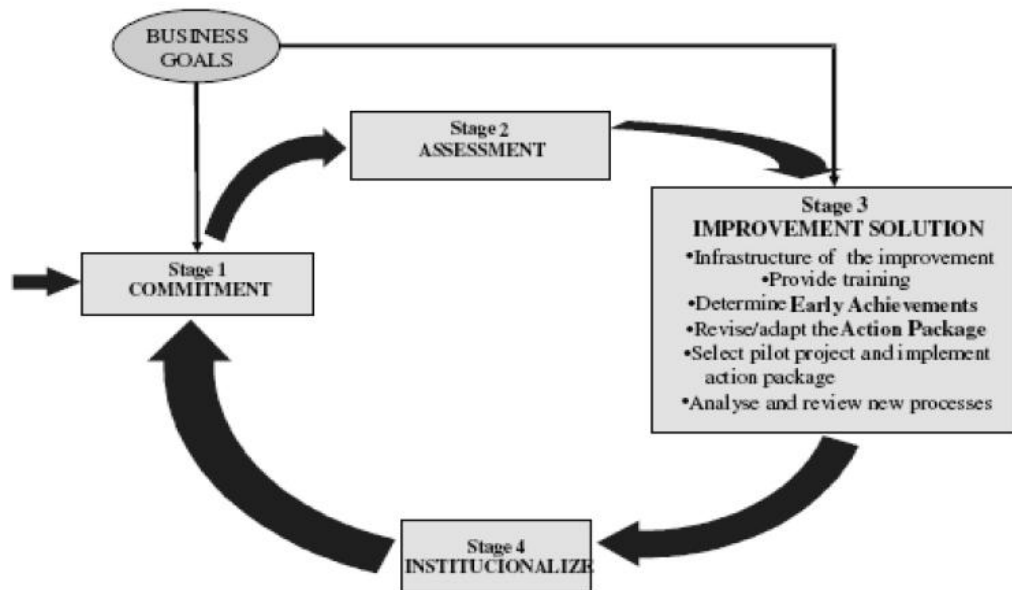


Figure 21 - MESOPYME process improvement, Source [26]

Most of the SPI approaches for SMEs try to cut down the cost by limiting the resources needed in carrying out software improvement plan. As a result, SMEs can implement the models looking at the less cost involved.



## 2.4 BUSINESS PROCESS IMPROVEMENT VERSUS SOFTWARE PROCESS IMPROVEMENT (SPI)

Business Process Improvement is a systematic approach to help organisations to archive significant changes in the way they do business [15]. Rosemann (2001) describes Business Process Improvement (BPI) as the evaluation of alternative ideas and the movement of the organisation. These definitions are in line with the aim of Software Process Improvement in software engineering.

Considering the process from business perspective, we find out that there are processes that are usually carried out in different organisation. These businesses processes are important to every organisation especially in the organisation's performance and ability to successfully execute on business strategy.

A dozen of business processes are used every day. This is usually evidenced by the same steps that are followed each time to accomplish a particular task. The example is when one wants to produce a report, resolves a customer complaint, contact a new client or manufacture a new product.

Business process can be inefficient; this is usually observed through unhappy customers, stressed colleagues, missed deadlines. This is why there is a need to improve processes when they are not working well. *Business Process Improvement* comes in to place here.

As much as there is a need for process improvement in software engineering, there is a need for process improvement in other areas of business. Figure 22 shows three basic elements of a process: the input to the process, the process under study and the output from the process. From these basic elements of a process, the idea of process improvement comes up easily.

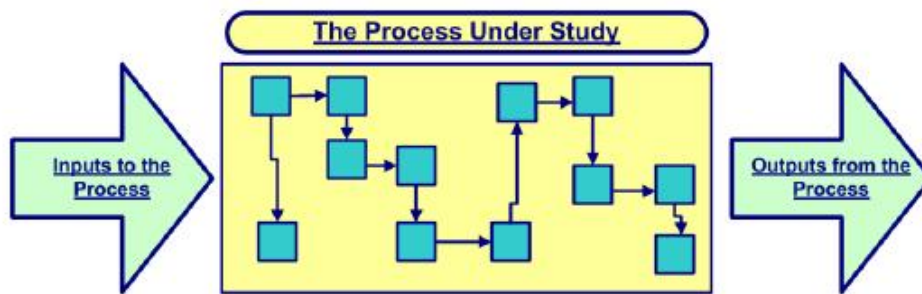


Figure 22 - Elements of a Process, Source [45]

To improve the outputs of a process, you simply improve the input and the process itself [45]. The whole idea behind this chart in Figure 22 can be represented using a function:

$$y = f(x_1, x_2, x_3 \dots x_n)$$

Where  $y$  is the output or the “Key Measure”,  $x$  is the inputs and process metrics. Furthermore, the concept that  $y$  is a function of  $x$  ( $y = f(x_1, x_2, x_3 \dots x_n)$ ) is at the core of the: Define, Measure, Analyse, Improve and Control; also known as DMAIC steps.

Hence we define a function for SPI in terms of

$$y_{spi} = (x_1, x_2, x_3 \dots x_n) \tag{1}$$

Where variables  $x_1, x_2, x_3 \dots x_n$  are the things that need to be improved for quality software product. These variables includes: business goals, budget for software development, defining of document process, assessment of suppliers process suitability, implementation of software design, planning of workflow and estimates.

Project managers are to understand the DMAIC steps. After these steps are well understood, then it will be easy to manage and improve the results of the process. Figure 23 shows a decision tree which helps the managers to make better decisions as they are able to ask the right questions for process improvement.

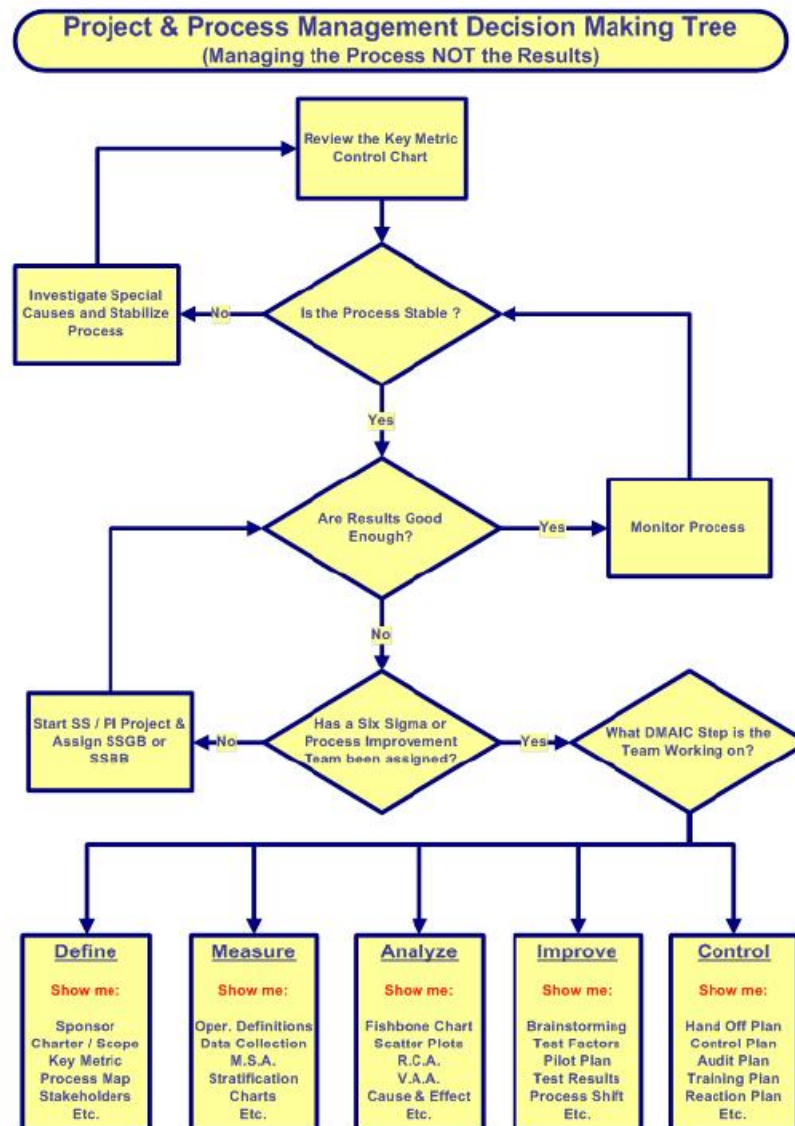


Figure 23 - Project and Process Management Decision Making Tree, Source [45]

The DMAIC are considered the proven roadmap for any process improvement project. That is why they are also relevant to software engineering processes. This is because they offer a structured approach to solving problems and improving results. The chart below shows the questions that can be addressed under each step.

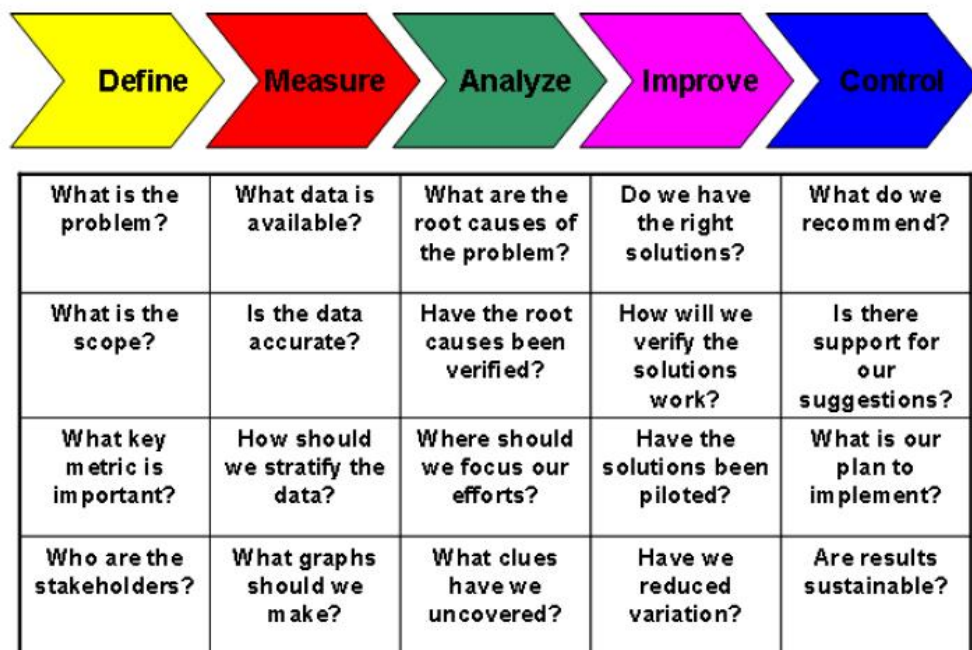


Figure 24 - Questions for Process Improvement Chart, Source [45]

Arising from figure 23 above , for SPI processes we make the following adaptations shown as shown in Figure 25:

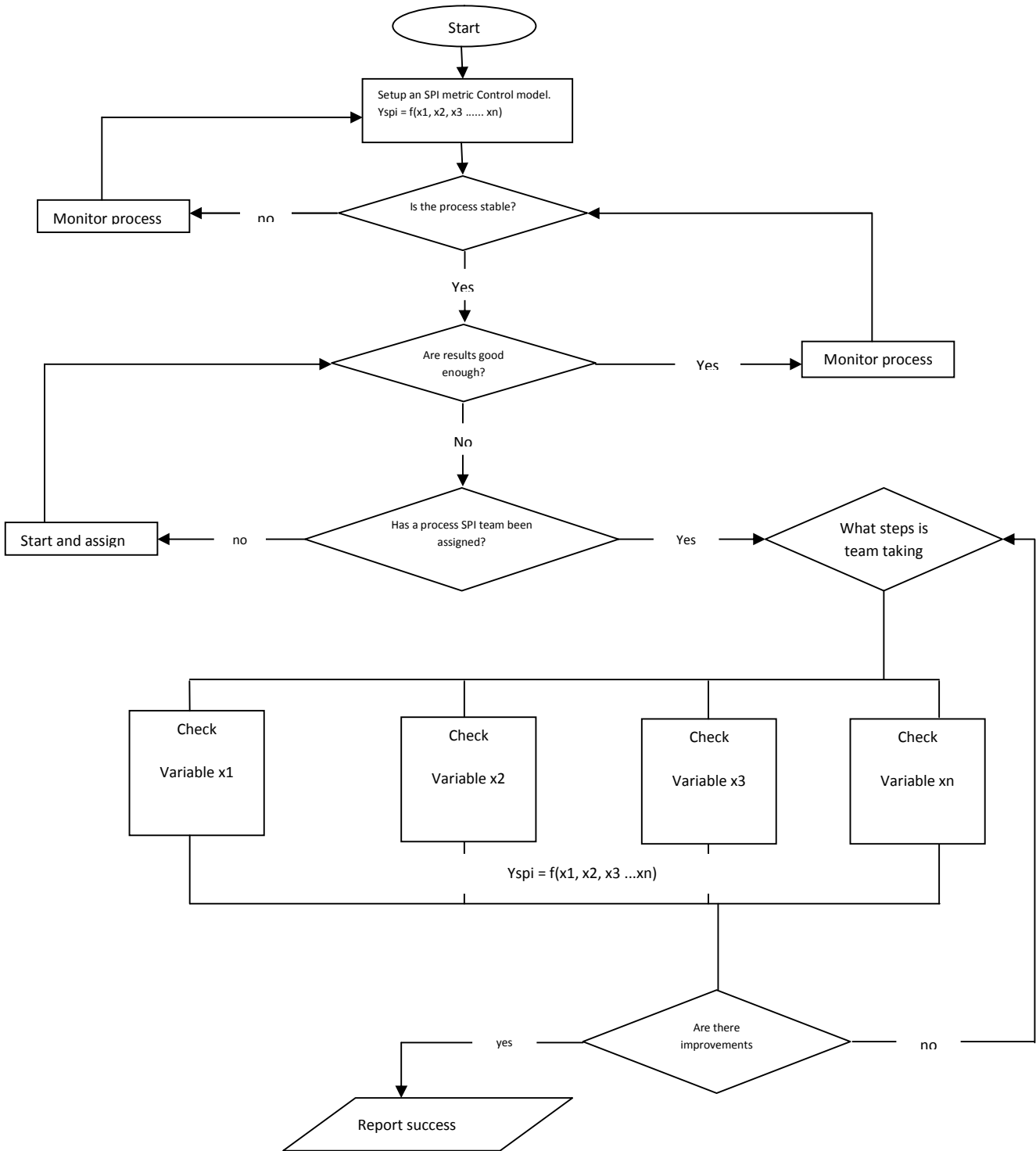


Figure 25 - Software Process Improvement Management Decision making Tree, Adapted from [45]

In this framework (Figure 25), to begin, a software company needs to do the following;

- i) Setup a metric control model  $y_{spi} = (x_1, x_2, x_3 \dots x_n)$  (2)
- ii) Verify that the process is stable
- iii) Check that the output results is good enough in equation 2, otherwise monitor the process and assign SPI process team.
- iv) Verify the steps taken up by the team in terms of process improvement variables

$$x_1, x_2, x_3 \dots x_n ,$$

Such that  $y_{spi} = (x_1, x_2, x_3 \dots x_n)$  is satisfied.

- v) Report success in improvement.

$$x_1, x_2, x_3 \dots x_n$$

## 2.5 OTHER STANDARDS AND FRAMEWORKS

There are other standards and frameworks which are also used as software process improvement techniques. According to the IEEE Comp. Soc. Software engineering standards committee, a standard can be:

- An object or measure of comparison that defines or represents the magnitude of a unit.
- A characterization that establishes allowable tolerances or constraints for categories of items
- A degree or level of required excellence or attainment.

There are a number of standards used in different circles. Standards and frameworks are used to promote software process improvement in software engineering organization. International bodies have come up with frameworks for this purpose.

### 2.5.1 ISO 9000

According to [11] ISO 9000 is a series of quality standards established by international Standards Organization (ISO) to certify that an organization practices on an acceptable level of quality control. This is a most commonly accepted model in some countries such as China. Standards are used in the case of bringing forth good software products. The ISO 9000 family addresses various quality management issues and has some ISO's best known standards which are recognized worldwide [5]. These standards provide all the guidance and tools needed by the companies and organizations to make sure they develop software projects which meet customers' requirements and for improvement of the quality [5]. The standard is concerned with a number of quality management principles including a strong customer focus, the motivation and implication of top management, the process approach and continual

improvement. When these standards are used, they will ensure that customers get consistently good quality products and services which in turn bring many business benefits. Companies need to adopt ISO 9000 standards. According to [6], ISO 9000 is a series of quality standards established by international standard Organization (ISO) to certify that an organization's practices meet an acceptable level of quality control. ISO 9000 family has many standards which includes the following:

- i) ISO 9001:2008 - sets out the requirements of a quality management system. The advantage with this standard is that it can be certified to, and it can be used by any organization whether large or small. The standard does not also consider the organization's field or activity.
- ii) ISO 9000: 2005- covers the basic concepts and language
- iii) ISO 9004: 2009 – focuses on how to make a quality management system more effective and efficient
- iv) ISO 19011:2011 – sets out guidance on internal and external audits of quality management system.

#### **ISO 9126: The standard of the Reference**

The objective of this standard is to provide a framework for the evaluation of the quality of the software. It defines a quality model which is applicable to every kind of software; this is done without providing the requirements of the software. There are six characteristics defined by this standard, which are: Functionality, reliability, portability, maintainability, efficiency, usability.

All these are important in the quality of the software.



## 2.6 SOFTWARE PROCESS ASSESSMENT (SPA)

This is an important aspect in software discipline. Software processes need constant assessment to see if Software Process Improvement is needed. Processes can always be compared and evaluated with other processes, no matter how good they are. Processes can be assessed in comparison with your earlier versions of your processes or processes to other companies. But what is software process assessment? It is an appraisal, or review, performed by a trained team of software professionals [46]. What then is the purpose of software process assessment? The purpose is to determine the current state of an organization's software process, to identify the highest priority process issues, and to facilitate improvement actions. Before any improvement can be done, there is need for assessment. Saiedian and Chennupati [10] states that process assessment helps software organizations improve themselves by identifying their critical problems and establishing improvement priorities [10].

A properly carried out process assessment will help an organization to improve by identifying the critical software problems and then establish improvement priority as it is already stated. It is the diagnostic tool to help in the organizational improvement. It is important to outline the objective of the assessment especially looking at the nature of this research. The objectives of the research are;

- First it provides a clear and factual understanding of the current state of the organization in terms of the software practice in place.
- Identifying major problems so that areas of improvement can be identified
- To initiate actions to make the necessary improvement.

The major thing in software process assessment is to review the software organization and give an advise to its management and professionals on how to improve their operation (Kitson and Humphrey, 1989). The assessment team is made up of sound leadership and cooperative organization [10]. After the assessment is done, the

assessing team will provide the recommendation on how to improve the processes in the organization.

The assessment phase includes selection, commitment, preparation, report and follows up. All these phases need to be carried out for the successful assessment.

## **2.7 SOFTWARE ESTIMATION STANDARDS AND MODELS**

Although estimation is out of focus of this study, it is necessary to how software companies do their estimates. Software estimation is another important aspect in improving the standards and the quality of the software. Estimation model is a collection of methods which uses models and tools to predict characteristics of a software project such as schedule and personnel needs before the project begins [5]. It is the processes of forecasting or approximating the time and cost of completing project deliverables or the task of balancing the expectations of stakeholders and the need for control while the project is implemented [47]. Software estimation can be done using Effort estimation and cost estimation. There are different models that can be used in effort estimation and cost estimation. This section discusses the three models of estimation: Classification and Progression Trees (CART), Estimation by Analogy (EBA) and Artificial Neural Network.

### **2.7.1 Software Effort Estimation**

There are different types of effort estimation approach that can be used. There is a non-model based and model based approach.

Non-model based method does not use models as the name suggest. This involves “expert judgment” where the software engineer examines the code or the software without using any model.

*Model - Based method* – this approach concentrates on models. These models are used in this research to estimate the effort in some of the codes that have been developed. Figure 26 shows software estimation techniques.

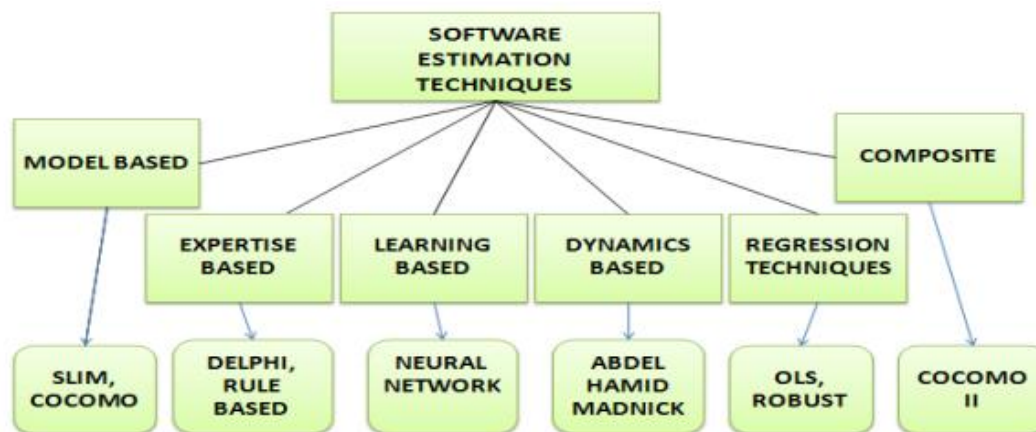


Figure 26 - Software Estimation Technique

### I) **CART (Classification and Progression Trees)**

This model is interested in predicting continuous dependent variables (regression) and categorical predictor variables (classification) [13]. This model is the ultimate classification tree that has really made a lot of impact in the entire field of classification. This method can be used in data mining as well and it has been designed for both non-technical and technical users.

### II) **EBA (Estimation By Analogy )**

This is another model used in software estimation. EbA is an estimation model which can be used in effort estimation. It means creating estimates for a new programme

by comparing new projects from the past. This model requires considerable amount of computation [47].

### III) **ANN (Artificial Neural Network)**

Artificial Neural Network (ANN) is made up of many artificial neurons that are linked together according to specific network architecture. The objective of this network is to transform the inputs into meaningful output. They can be used in software products to predict the effort.

The mathematics of the Neural Network is given by the formula.

$$y_{out} = F(x, W)$$

W is the matrix of all weight vectors [48].

### IV) **Expert Opinion**

When there is no quantified data, or there is no empirical data, then expertise based techniques are needed. In this method, the opinion of expert is taken. This model has its own drawback, the estimate is as good as the expert's opinion only. An example of this model is Delphi technique. In this method, there are four steps involved: selection of expert, briefing to the expert, collation of estimates from experts and then Convergence of estimates and finalization.

### v) **SLIM (Pulnam's Software Life-cycle Model)**

SLIM stands for Software Life-cycle Model. Introduced by L. H. Putnam in 1978 [47]. This model describes the time and the effort required for a project of a specified size. This model has other models closely related to it, which are COCOMO, PRICE-S and SEER-SEM.

### **2.7.2 CONSTRUCTIVE COST MODEL II (COCOMO II)**

**CO**nstructive **CO**st **MO**del II is a model that allows one to estimate the cost, effort and schedule when planning a new software development activity. It is the extension of the original COCOMO 81 which was established in 1981 by Boehm. The model consists of three sub models and each of these models offer increased integrity. COCOMO II is useful for a much wider collection of techniques and technologies. The model provides up to date support for business software, software created over evolutionary development models, object oriented software and the software developed using commercial- off-the-shelf application composition utilities [19]. Software cost estimation with COCOMO II model is very important for making good management decisions. Managers need to understand the use of this model in order for them to make good decisions.

### **2.7.3 CONSRUCTIVE QUALITY MODEL (COQUALMO)**

The **CO**nstructive **QU**ality **MO**del (COQUALMO) was constructed in recognition that the cost, schedule and quality are highly correlated factors in software development. This is actually created to depict defects as an extension of the COCOMO II software cost model.

## **2.8 SOFTWARE QUALITY STANDARDS**

High quality software meets the needs of users while being reliable, well supported maintainable, portable and easily integrated with other tools [49]. Achieving quality has some tradeoff. To improve software quality, there are costs implications. Time and money is required to identify the problem, removing the source of the problem, connecting it to the real source, fixing the requirements, design and the code, testing the fix for that specific problem, test if the fix has not caused new problems and changing of the documentation [49].

While striving for quality which includes satisfying our customers, probably by adding new features, we must bear in mind that other bugs can rise up. The point is when developers are given fixed amount of time and resources, adding features will add bugs and the time for testing will be reduced. This is where one needs to consider if increasing features is more important than the bugs that are likely to arise. From this, it is clear that Quality requires more time if it is not well planned for. If developers concentrate on it from the beginning, then the components produced will have less defects, then there will be less time spend in debugging and more time will be allocated for improving the software. Lack of focus on the quality of the software will tend to produce component with more hidden defects which will lead to more time being spent in the fixing of defects and poor product come as a result of these. It is good to inspect the code than to test it. That is the reason why quality is important in software development and it must be considered throughout the product lifecycle.

Better software quality can be ensured through standards. Although the quality of the software is not guaranteed through the use of standards, the standards help to make sure that no steps are skipped. Software quality standards include, SO/IEC 9126 – 1: 2001, SO 9126.

## **2.9 SOFTWARE STANDARDS APPLICATION IN SMALL COMPANIES**

Very Small Entities or small software companies cannot easily make use of the existing standards [31]. This is because they are not capable of bearing the cost of implementing these improvement standards [5, 31]. As most companies in Botswana are small, it is or it will be difficult for them to apply the common software engineering standards. Most of the standards are mostly applicable to big companies. Laporte, Seguin, Gisele, Boas and Buasung [31] have proposed a new ISO/IEC JTC/SC7

working group, WG24 to address some of these difficulties, by providing guidance for complying with ISO software engineering standards.

### **2.9.1 SOFTWARE PROCESS IMPROVEMENT IN SOME NATIONS**

A number of studies have been conducted in a number of organizations to find out how Software Process Improvement models are being implemented. In this section few studies are examined to see the impact of Software Process Improvement models in such selected countries. Different types of Software companies were considered in these nations to see how they implement SPIs. These countries includes India, Brazil, Malaysia e.t.c

#### **2.9.2.1 Software Process Improvement in small and medium software organizations in India**

Thapliyal and Pratibha [56] focused on the key issues in SPI paradigm, in comparison to the current process in media to those of the standard model used internationally. The researcher also suggested a software process practice model for the Indian small and medium software companies. According to the researcher, Indian software industry has been maturing in many dimensions. This is much attributed to the fact that many software companies in this country adopted the use of international standards. In his research, 50 software companies were selected and out of these companies 15 of them showed their eagerness for arching a CMM as their primary goal of process improvement.

#### ***Current scenario – India***

When considering the current scenario of India in Software Process Improvement, the researcher considered the importance of software industry in the country in terms of the revenue they generate in the country. The researcher also outlined the problems faced by Small and Medium Enterprises in India as well as the reasons why Small and

Medium Enterprises are relevant to engage in formal Software Process Improvement Assessment.

The general look at software industry in India was considered in respect to the age of the companies, the growth of software industry since 1990 in terms of implementing SPI, process and quality dimension. The author states that when ISO 9000 was introduced in the 90s Indian software companies quickly adopted it to improve their quality process. Hundreds of companies in the country got ISO certified. Many of the companies in this country are now CMM certified (ibid). The issue of relevant skill in SPI was also taken into consideration.

#### ***Data collection***

According to this paper (ibid) data collection was done from fifty SME companies. Questionnaires were sent by mail directly to managing directors, managers, senior development professionals. The Interview sessions were done through telephone. Open ended questions were used as well to try and gather relevant data.

In general the researcher's questions were designed to know about software process implementation and process improvement programs in India. A closer look at this study shows that there is a need to look closely at the companies in different countries. Proper guidelines should be put in place for companies that are newly coming up.

#### ***2.9.2.2 SPI in small and medium software organizations in United Kingdom***

In this research 85 companies in the United Kingdom were considered. SPI is progressing in the UK software industry. 15 of the 85 companies had undergone implementation of software process improvement in UK.

According to (Tracy, Austen, & Baddoo, 2002), generally, companies have been using SPI over a relatively long period of time. Researchers indicate that the biggest



impediment to SPI success is inadequate resources. Companies are also generally ineffective at evaluating the impact of SPI. SPI efforts are generally not focused and performance is not systematically assessed. However, findings indicate that companies have good understanding of the human factors associated with SPI. Overall findings show that SPI is progressing in the UK software industry.

### **2.9.2.3 Software Process Improvement Models in Malaysia**

In the research done in Malaysia by Shukor, Nuraminah and Nasir [15], highlighted the importance of SPI to Malaysian software organisations. They also looked at the current state of CMMI implementation in Malaysia. The researcher outlined the following points about the importance of CMMI to companies:

#### **Importance of CMMI to Companies**

CMMI is very important to software companies because:

- i) Enables company to compete successfully in the international market
- ii) Helps to improve both the quality and software product
- iii) Improves capability of the company to work with time and budget
- iv) Assist the company in their software measurement program and software process assessment.
- v) To manage in the development of the software.

The questionnaire used by the researchers was based on investigating level of organization involved, and also on the reason why the company implemented CMMI, roles in CMMI implementation and the budget used.

The report by Shukor, Nuraminah and Nasir, [15], shows that there are about 40 out of 94 or more Malaysian companies which have been certified with CMMI.

#### **2.9.2.4 Software Process Improvement in Bangladesh**

In a study done by Bernard and Sazzad [57] software development standard and software engineering practice research was carried out in Bangladesh. The study took into consideration that the company can directly or indirectly involved in developing a software. The software company can be developing for local market or for international market. In this study, only software companies which are developing software applications and selling were considered.

The study used a convenient sampling method to select fifty software companies and a questionnaire was later sent to them to gather the data. The SPSS (Statistical Package Social Sciences) were used to analyze the data. The study also looked at SPI like the IDEAL (Initiating, Diagnosing, Establishing, Acting and Learning) method (ibid).

This study revealed that 78.3% of the Software companies do business on software development and sell and the remaining 21.7% are doing general software development. The study considered attributes like: Age of the company/business, the duration the employees work in the company, number of the people working for the company and the number of software product developed.

The mean and the standard deviation were taken for all the companies after the data was collected and looking at the number of years, average number of software products produced, number of employees in the company and the duration the company has been in existence.

In a nutshell, the study considered the following attributes: Business domain, Standards: e.g. quality assurance certifications, Software quality control, Various stages of software development. The general results shows that 44.4 % companies are ISO 9001 certified, 11.1 % companies are ISO 9002 certified, 5.6 % companies are ISO 9003 certified, 8.3 % companies have CMM certification, 30.6% companies have no certification. The results from the manpower skills audit done shows that Bangladesh has a great potential in manpower.

## **Software Process Improvement in South Africa**

CMMI is becoming increasingly accepted in South Africa. In 2006, a programme was launched by Joburg Centre for Software Engineering (JCSE) at Wits University to bring CMMI in South Africa [56]. A pilot study was done with six companies and later enlarged to include other large organisation and SMEs. This pilot was done in IT companies, banks and even universities [58].

The two important elements covered by the pilot was to build a strong case to present to South African Department of Trade and industry (DTI) in order to request government's support for the mass adoption of CMMI as a tool to improve business processes within local companies. The second elements were to form a part of a broader international research project being conducted by the SEI which is aimed at exploring the role of CMMI in small organization (ibid).

One interesting thing about this pilot is that the South African government is already interested in getting involved in the encouraging the use of CMMI in the country. It is clearly pointed out that the same government has acknowledged the importance of CMMI in the country. Process improvement would be critical in assisting local software development companies in becoming more competitive internationally [58].

Another study carried out by D.J Cohen [57], significant attempts are been observed in South Africa to encourage the companies to adopt CMMI. This also attested by the pilot programme called "Bringing CMMI to South Africa" which was launched in 2006. One of the reasons pointed out in the study as of why SPI models are not adopted in South Africa is that lack of awareness of quality standard or actual demand for CMMI along with the relatively high implementation and support costs [57]. The results of the study revealed some important things worthy of consideration.

- Some of the major reasons why organizations adopt SPI or CMMI in South Africa includes: Ensuring predictable results, improve customer satisfaction,

improve service/product quality, expected process improvement, improve productivity, competitive advantage, reduce time to develop, company philosophy, improve public image, people development and increase internal controls. These are listed in the order of high frequency to low frequency.

- Impact of CMMI is seen mostly in tracking changes in the internal business process, followed by customer satisfaction and the financial impact.
- On the impact on intangible benefits of adopting CMMI, 23% perceived there to be a very positive impact, 45% a somewhat positive impact and only 5% perceived a negative impact in some way. In terms of tangible costs, on average 88% of respondents believed that the costs were worthwhile.

These results show that SPI approaches especially CMMI can be very beneficial to organizations. The adoption of standards in South Africa can be said to be improving. This is a challenge to developing countries like Botswana as how adoption of CMMI and other SPI approaches.

#### **2.9.2.5 Software Process Improvement in Botswana**

The study done by Ayelew and Motlhala [50] using a sample 10 companies reveal that only two companies in Botswana has been assessed with at least one standard i.e. CMMI. The rest of the seven companies indicated that they did not have any process assessment experience. Moreover Table 1, further shows the results as follows, i) that most of the companies in Botswana have staff less 50 employees which shows that the software companies in Botswana are SMEs, ii) most companies are still young in terms of the years they have been practicing as 3 companies out of the study shows that they are have more than 10 years of experience, iii) less than third of the companies in the study have been accessed in their process.

**Table 1 - Profiles of software Organizations in Botswana**

	Organization									
	A	B	C	D	E	F	G	H	I	J
<b>Number of Employees</b>	10	12	30	9	6	17	3	10	72	70
<b>Size of development Team</b>	7	4	24	6	4	4	3	7	40	40
<b>Years of Operation</b>	>10	2 - 5	>10	2-5	6-10	6-10	6-10	6-10	>10	>10
<b>Any Previous Assessment?</b>	YES	Don't Know	NO	Don't Know	NO	YES	NO	Don't Know	YES	YES
<b>Assessment Standard</b>	CM MI	-	None	-	None	Don't Know	None	-	CM MI	Microsoft Development

Ayelew and Motlhala [50] further states that “software process assessment is a new practice to most organizations in Botswana”. It is indicated that none of the companies have any SPI or related certification.

This research will help to examine the companies in Botswana. Most of the software companies in Botswana are small companies [50]. Literature shows that small companies have a challenge to carry out Software Process Improvement mostly because they are not able to invest a lot of money and other resources on the improvement programs [34]. On the other hand, large organisations have the ability to commit more funds in extensive SPI programs [10]. That is why it is important for the company to choose an SPI model which closely represents its aims and goals.

## **2.10 CHOOSING SOFTWARE MODEL**

The models discussed in 2.2 are diverse, ranging from software process models, software process improvement models, software quality models and software estimation models. There are also standards and frameworks been developed or that already exist in the literature. The choice of these models depends on a number of

parameters which includes the type of the organisation, size of the organisation, the project size and the availability of the funds in the organisation. When an organisation commits itself to process improvement, it looks for an SPI model which will be suitable in terms of its business needs [10]. Hossein and Kalyani further state that the choice of a particular model depends on the type of organisation, its business needs and its business goals [10]. Small organisation may not be able to invest a great amount of money and resources on an expensive process improvement program. On the other hand, large companies can easily afford to launch an extensive SPI program.

How do companies choose models. G. Holodnik-Janczura, I. Golinska suggested a model for choosing software process model or software development life cycle. His model is discussed in the section below.

#### **2.10.1 CHOOSING SOFTWARE PROCESS MODELS**

With all this variety of models to use, the literature has some suggested models that can be used to assist developers to choose which models to use. The user needs to know the parameters of the project they are undertaking. After the organisation has done the assessment and has finalised that there is a need for software process improvement in that organisation, it will then look for SPI model that will be suitable for in terms of its business needs [10].

In this section, some of these models are discussed.

#### **2.10.2 DECISION SUPPORT SYSTEM BY G. HOLODNIK-JANCZURA AND I. GOLINSKA**

There are a number of methodology present in today's world, computer scientist have already discovered the need for a decision support system for choosing a model for a software development life cycle. Holodnik-Janczura and Golinska [59] suggest an expert system to help come up with results that would help to establish which SDLC models are particularly suitable for a particular IT project . The expert system will incorporate artificial intelligence. The intension of developing the system was not to

come up with a clear answer but to obtain guidance in choosing from the known set of models. Generally, the system will achieve its goals by:

- i) Making use of the characteristics of the project which are stored in the knowledge-base
- ii) Users of the system will have to determine the characteristics of their projects by answering a set of questions asked by the system in order for them to obtain information about a suitable model.
- iii) The system have two panels; the first accessible to the administrator, the second accessible to the decision maker. The first layer act as a tool for parameterizing the system according to the models considered, criteria and management of the question facility. Second layer allows the choice of a model which is accessible by decision makers.
- iv) A successful selection of an SDLC model depends on the appropriate definition of the expert, criterion layer.

Project managers need such systems that are able to help or exempt them from headache of various characteristics of models and of examining their influence on successful project implementation. In this system the wearisome stage of determining criteria and associating them with question is completed by an expert. Figure 27 shows the abstract layered division of the decision-making system.

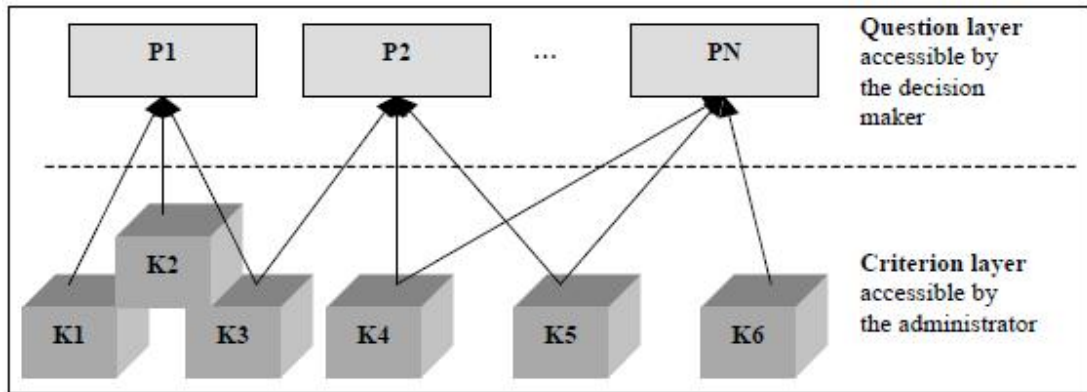


Figure 27 - Abstract, layered division of the decision-making system, Source [59]

Ibid., define their system having a knowledge-base which contain information concerning the SDLC models. The database was designed by:

- Making some simplification and assumptions were made to aid the design of the database
- The database comprises of two parts – the first stores the set of available models and the information about their features, and the second part maintains the links between the characteristics of models and project.
- The system included 11 models.

The database was filled with the appropriate components.

The system also made use of the algorithm for selecting a model. This algorithm is called a heuristic ranking algorithm (ibid). The outcome of its operation is a list models together with the ascribed score with the order. The criteria for coming up with the models is based on: Software size, membership of a life-critical system, software purpose, clarity of the requirements, main system function, project duration, financial resources, team experience, user experience, user contact.

The algorithm used by ibid., took two cases into consideration: the existence of, or the non-existence of a link between a condition and the parameters of a given model. They state that, if the link exists, the score is calculated using:

$$m_{ij} = m_{ij-1} + \frac{kw + pw}{2} \cdot \frac{sp}{100}$$



Where  $m_{ij}$  is the value of the *model\_points* field in the model table, which includes the previous sum of points collected by model  $i$  in stages 1, 2, ...,  $j-1$  of the selection process,

$kw$  is the value of *criterion\_weight* in the Criteria table, which stands for the weight of the criterion to which a given condition is referring,

$pw$  is the value of *question\_weight* in the Question table, which stands for the weight, or in other words, the importance of the question, and

$sp$  is the value of *fulfillment\_level\_parameter* in the Parameters table.

The algorithm is shown in Figure 28.

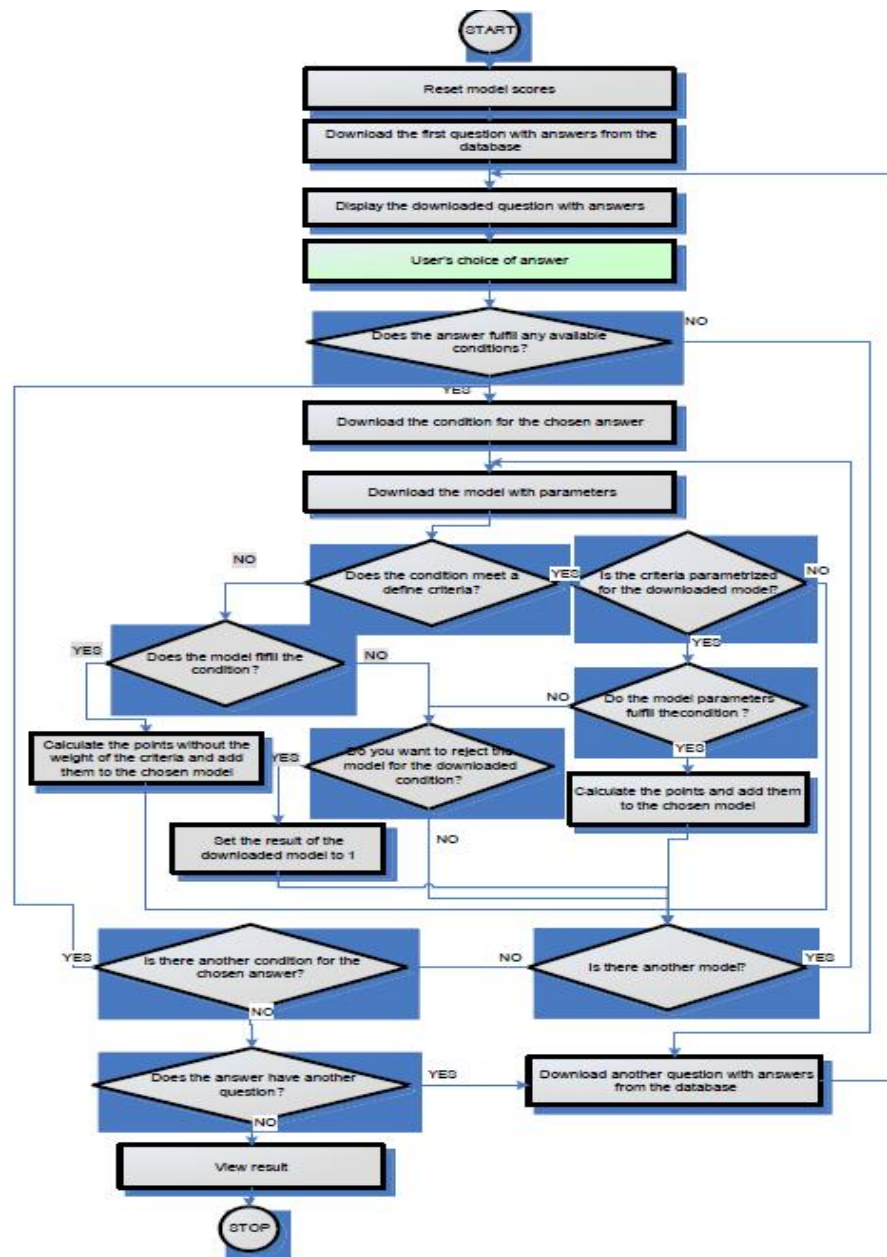


Figure 28 - Algorithm for selecting an SDLC model, Source [59]

Software process improvement models can also be of great use once there is a way of helping managers or decision makers to pick the right one. The experience with these models is that, the rest on a normative approach, where the decision maker's participation in a software organization is limited to understanding which process is relevant to each organization (ibid).

### **2.10.3 MULTI-CRITERIA DECISION AIDING - CONSTRUCTIVIST (MCDA-C)**

Multi-Criteria Decision Aiding – Constructivist (MCDA-C) is the methodology used in software process assessment and improvement to help the managers with their decisions. The methodology makes it possible to vitalize the criteria that must be considered according to the decision-makers' values in the process improvement actions. This will allow the ranking of the actions in the light of specific organizational need [60].

MCDA-C attempts to address some weaknesses in SPI model adoption which includes; one, how to identify some barriers in organisation from the perspective of software assessment and improvement process and the determination of critical success factors, two how the judgement about the current stage of an activity or process may or may not have credibility [59]. It is the alternative for software organisations, for adoption in process assessment and improvement, through the option of constructivist approach, which recognizes the need of expansion of a decision maker's knowledge about his/her specific decision context, in contrast with normative models which believe they have an optimum solution to any context [60].

MCDA-C is a branch of the MCDA, which serves as a way to aid decision-makers in complex, conflicting situations where one need to improve their understanding of the situation and no alternative exist at the beginning of the process[60].

#### ***Procedure of the MCDA-C***

It is necessary to have an understanding of how MCDA-C function. The model have three phases.

The phases include:

##### *i) Structuring*

The problem to be discussed needs to be understood in a broader way. This is where the stakeholders are identified. This helps to clarify whose perception of the context is important and for the individual whom the knowledge about the context should be improved.

##### *ii) Evaluation*

##### *iii) Recommendations*

#### **2.10.4 A DECISION SUPPORT SCHEME FOR SOFTWARE PROCESS IMPROVEMENT PRIORITIZATION**

There are several decisions that need to be taken by the management during process improvement. Managers usually find themselves in challenging situations of making decisions when attempting to improve the Quality Assurance (QA) processes. The decisions are to be made on which technique to improve first in order to achieve the highest possible quality gain. Arne Beckhaus, et al., [16] suggest an SPI decision support scheme that will provide quality managers with a toolkit to prioritise improvement activities based on expected defect reduction. Their approach relies on the constructive Quality Model (COQUALMO); a model that intends to predict defects.

Quality managers are very interested in prioritising improvement of three techniques of automatic code analysis, peer reviews, and executive testing and tools [16].

#### **2.11 PROPOSED MODEL FOR CHOOSING SPI**

In order to choose the appropriate model from the existing models, a framework is needed. The fact that there is a variety of software process improvement, there is a need to come up with an “intelligent framework” that can aid in the selection of the proper model. There are many legitimate areas for comparison; an “Evaluative framework” is needed that highlights the most visible elements for evaluation purposes [10]. That is the framework that will evaluate different SPI models in view of the given peculiar characteristics of the organisation and the recommend or guide the management to the right SPI model or models. Saiedian and Chennupati, state that the choice of a particular model depends on i) the type of the organisation, ii) its business needs, iii) and its business goals. The objective of this research is to develop, define and justify the components of this framework. The framework will take note of the following important elements to make decisions:

### **2.11.1 Organisation size**

The number of employees in an organisation has some obvious implications for the organisational culture [49]. This culture has an impact on how the processes are carried out in that organisation. Some authors claim the larger organisation use formalised quality management as a consequence of the inherent need of structure and control in such organisation [49]. Small organisations cannot afford the requirements of traditional models as they need experienced staff which cannot be easily be found in SMEs.

### **2.11.2 Project Team Size**

Larger organisations can establish specialised functional teams, such as dedicated design teams, testing teams and deployment teams while the small medium enterprises can have more autonomous and generic project teams with many responsibilities throughout the whole project [49]. This is one area which can also be used to determine whether the company is large or not.

### **2.11.3 Kind of Project Carried Out in an organisation**

The size of the budget normally correlates naturally with the size of the organisation. It has been discovered that small organisations develop fewer, smaller and less complex systems within shorter schedules and lower budgets than their larger counterparts do [48]. Smaller organisations normally develop or carry our few projects because of the limited budget, limited time as well as limited human resource.

### **2.11.4 Availability of Funds/ Financial spine**

Larger organisations are able to accumulate some financial reserves as they normally have some repeated successes. This allows the larger organisations to be able to invest in long term improvements.

### **2.11.5 Staff Experience**

Smaller organisation normally avoids traditional models like CMMI because they do not have enough manpower to explore them.

### **2.11.6 The Process to be improved**

It is important to consider and assess the processes that need to be improved. This will give information on the current status of the organisation.

## **CHAPTER 3: SOFTWARE PROCESSES AND PROCESS IMPROVEMENT IN BOTSWANA**

### **3.0 INTRODUCTION**

This section covers the methodology that was used in carrying out the study. Both interviews and questionnaires were used to gather the data from 15 software and 20 software user organisations.

The government need to enforce the use of the standard in the country by using them as way to select the companies that qualify for recognition and also in the participation of such companies in the execution of project in the country. The results of the study will help to request the government support for the mass adoption of SPI especially CMMI as a tool to improve the level of service given to customers.

On the site of the business or companies, they can perform better when they are certified or they are adopting the use of SPIs. This is also a proof that the companies are mature to handle quality projects.

For the companies that are said to be using CMMI, maturity levels of such companies should be checked regularly to see if they are maintaining them. The study will help to determine the level of awareness of the SPI in the country.

The methodology used in this study is explained in the following section 3.1.

### **3.1 METHODOLOGY**

To achieve the objectives of this study, 14 software companies and 16 governmental and business based departments were selected. Appropriate questionnaire was designed in order to collect relevant data from the software companies and the other organizations. The questionnaires were administered to the selected software companies, government departments, and business organisations. For software companies, questionnaire covered areas of: company profile, company services and specialty program, company Certification and process improvement, software development and software quality control. For the

software users in government departments and business organization the questionnaire covered organization background, type of software in use, user satisfaction, and annual budget for the software. Both questionnaires were prepared and pre-tested for validity. Pretesting was done by administering the initial questionnaire to six companies. Unclear questions were corrected before the final questionnaire was administered to selected companies.

### **3.1.1 Data Collection Methods**

The Public Procurement and Asset Disposal Board (PPADB) office maintains a list of 100% Citizen Owned IT businesses and the Central Statistics Office (CSO) also maintains a database of all business establishments in the manufacturing sector, Sentsho, Maiketso, Sengwaketse and Ndzinge-Anderson [61]. The list of 100% Citizen Owned IT business was used to select some software companies. The study focused on those companies or organizations and departments that are involved in software development and usage.

#### ***Sampling method***

Purposive Sampling method was used to select companies and organisations. According to Dolores and Tongco [62], Purposive sampling technique, also called judgment sampling, is the deliberate choice of an informant due to the qualities the informant possesses. It is a nonrandom technique that does not need underlying theories or a set number of informants. Simply put, the researcher decides what needs to be known and sets out to find people who can and are willing to provide the information by virtue of knowledge or experience. This is because some companies are not operational though they are registered with registrar of companies. There are few organisations or companies that are involved in software development in Botswana. Judgement sampling method may prove to be effective when only limited numbers of people can serve as primary data sources due to the nature of research design and aims and



objectives [63]. This explains why only companies which are operating were considered. These companies were visited to confirm that they are really operating and if they are practicing software development.

### **3.1.2 Research Questions**

The questionnaire technique was used in this research. This refers to forms filled in by respondents alone. Questionnaires can be handed out or sent by mail and later collected or returned by stamped addressed envelope [64]. Some people prefer to participate in questionnaire for comfort sake. This was also helpful in cases where the relevant people were not available. The questionnaire covered issues such as, whether the company is ISO certified, whether it is using some other standards, whether the company has implemented the software process improvement methods. The questionnaire also covered issues of finding out how the organization is applying the standards and how they are implementing the Software Process Improvement. The questionnaire also covered the following variables:

- Age of the company
- Size of the company, in terms of the number of IT professionals they have.
- Usage of existing SPI and standards
- Budget for the SPI and standards

Generally, the questionnaire addressed the following questions:

#### **Question 1**

How big/small are the software companies and IT departments in Botswana? Are they SMEs?

#### **Question 2**

What processes are been done by software companies and IT departments in Botswana?

**Question 3**

What kind of services do software companies in provide?

**Question 4**

Are software companies in Botswana using SPI? Do they follow some standards?

**Question 5**

Do companies in Botswana care about the quality of the software products they produce?

**Question 6**

To what extend do software companies and IT department use CMM/CMMI in Botswana?

The questionnaire consisted mainly of multiple choice answers. Some of the questions allowed the participants to choose more than one answer. Other multiple choice questions allowed participants to add their other possible choices. This was to allow the collection of as much information as possible. There were also open answer questions which also helped in the collection of more data.

As data was collected by the questionnaires some companies and organizations were handed the questions while other questionnaires were filled through telephone conversation. Questionnaires were also sent to some other companies through email. More than 20 software organizations, 14 responded. The questionnaires were given to the following category of companies:

- Only companies having offices or operating in Botswana were given the questionnaires
- The companies were selected from Lobatse and Gaborone.

- Organizations were selected basing on the PPADB list of registered companies while some were identified through searching the internet.
- IT Companies doing other business apart from software development were also included.

After administering the two questionnaires the response was found to be low and more data needed to be collected. This is the reason why the third questionnaire was also designed and used for both set of organisations.

### **3.1.3 Interviews**

Interview is a method of data collection where information is obtained through enquiry and recorded by enumerators [64]. The interviews were also used in cases where the organization is far or they could not be easily accessed. This has also provided a face to face interaction as some companies were visited.

### **3.1.4 Site Selection**

Software companies specializing in software development and those that do software development among other things were interviewed. Companies were selected all over Botswana especially in the cities and towns. Some other departments which are believed to be major customers to these companies were also given their special questionnaires.

### **3.1.5 Data Analysis**

SPSS package was used to analyze the data and results are presented in Chapter 4. The relevant statistical tools used include regression analysis: *R square* and *Anova*. A model of the form

$$\alpha_{ij} = \beta_0 + \beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n \quad (2)$$

Was used, where

$\alpha_{ij}$  represent a dependent variable,

$\beta_0 + \beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n$  the independent variables. This model was used to verify the appropriateness of our software improvement model presented in equation (1) page 67. The dependent variable in this study is the software process improvement (SPI).

Hence, from equation (1)

$$y_{spi} = \alpha_{ij} \text{ i.e}$$

Company  $y_{spi}$  = Company  $\alpha_{ij}$ .

The dependent variables  $\beta_0 + \beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n = x_1, x_2, x_3 \dots x_n$  i.e things that needs to be improved for quality software.

$\beta_0$  = constant, e.g. company name.

$\beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n$  are the things the company is doing to improve software process. This includes the standards the companies are following to improve software processes.

## CHAPTER 4: RESULTS AND DISCUSSION

### 4.0 PRESENTATION OF RESULTS

This chapter presents the results of our study, interprets the results, and discusses the implications as shown below;

#### 4.1 MODEL FITTING

The model;

$\alpha_{ij} = \beta_0 + \beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n$  is used in this study and is verified for validity using regression analysis as presented below.

**Table 2 - Model summary**

Model	R	R Square	Adjusted Square	RStd. Error of the Estimate
1	.600 <sup>a</sup>	.360	-.189	4.561

*a. Predictors: (Constant), Is your company registered with any other international standard apart from the one mentioned above?, Which certification does your company have? ISO 9001, Is your company registered with any of the following Software Process Improvement (SPI)? SPICE, Is your company registered with any of the following Software Process Improvement (SPI)? IDEAL, Which certification does your company have? CMM, Which certification does your company have? CMMI*

Explanation of Table 2 (results) is provided in 4.1.1

**Table 3 - Anova**

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	81.871	6	13.645	.656	.688 <sup>a</sup>
	Residual	145.629	7	20.804		
	Total	227.500	13			

*a. Predictors: (Constant), Is your company registered with any other international standard apart from the one mentioned above?, Which certification does your company have? ISO 9001, Is your company registered with any of the following Software Process Improvement (SPI)? SPICE, Is your company registered with any of the following Software Process Improvement (SPI)? IDEAL, Which certification does your company have? CMM, Which certification does your company have? CMMI*

*b. Dependent Variable: Company name?*

Explanation of Table 3 is presented in 4.1.1.

**Table 4 - COEFFICIENT**

Model	Unstandardized Coefficients		Standardized Coefficients	T	Sig.	
	B	Std. Error	Beta			
1	(Constant)	10053.476	40.688		247.089	.000
	Which certification does your company have? ISO 9001	-4.841	6.334	-.493	-.764	.470
	Which certification does your company have? CMM	-3.167	5.126	-.355	-.618	.556
	Which certification does your company have? CMMI	-4.973	5.352	-.617	-.929	.384
	Is your company registered with any of the following Software Process Improvement (SPI)? SPICE	-6.167	5.126	-.394	-1.203	.268
	Is your company registered with any of the following Software Process Improvement (SPI)? IDEAL	-7.222	7.300	-.461	-.989	.355
	Is your company registered with any other international standard apart from the one mentioned above?	.021	.029	.248	.728	.490

Model	Unstandardized Coefficients		Standardized Coefficients	T	Sig.	
	B	Std. Error	Beta			
1	(Constant)	10053.476	40.688		247.089	.000
	Which certification does your company have? ISO 9001	-4.841	6.334	-.493	-.764	.470
	Which certification does your company have? CMM	-3.167	5.126	-.355	-.618	.556
	Which certification does your company have? CMMI	-4.973	5.352	-.617	-.929	.384
	Is your company registered with any of the following Software Process Improvement (SPI)? SPICE	-6.167	5.126	-.394	-1.203	.268
	Is your company registered with any of the following Software Process Improvement (SPI)? IDEAL	-7.222	7.300	-.461	-.989	.355
	Is your company registered with any other international standard apart from the one mentioned above?	.021	.029	.248	.728	.490

*a. Dependent Variable: Company name?*

Explanation of Table 4 is presented in 4.1.1.



**Table 5 - Excluded Variables**

Model	Beta In	T	Sig.	Partial Correlation	Collinearity Statistics	
					Tolerance	
1	Which certification does your company have? ISO 9002	<sup>a</sup>	.	.	.	.000

*a. Predictors in the Model: (Constant), Is your company registered with any other international standard apart from the one mentioned above?, Which certification does your company have? ISO 9001, Is your company registered with any of the following Software Process Improvement (SPI)? SPICE, Is your company registered with any of the following Software Process Improvement (SPI)? IDEAL, Which certification does your company have? CMM, Which certification does your company have? CMMI*

*b. Dependent Variable: Company name?*

Explanation of Table 5 is presented in 4.1.1.

#### 4.1.1 MODEL SUMMARY

Table 2,3, 4 and 5 are results from regression analysis. From Table 2, the R square value explains the model to imply the variation of only 36% in SPI. This suggests that there are 64% unexplained variation which could come from other variables not investigated. This shows that our data sample may not be adequate, and our selected variables in this case, the use of SPI by these companies are also not adequate (i.e. the companies may not be making full use of SPI models in Botswana).

Although the model  $\alpha_{ij} = \beta_0 + \beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n$  is useful in this study, the selected variables are not comprehensive, meaning that the software companies are not fully utilizing SPI models in Botswana. This is further supported by the frequency analysis as shown in Table 6.

**Table 6: Botswana Software Companies use of SPI Models**

COMPANY NAME	INTERNATIONAL STANDARDS							
	ISO 9001	ISO 9002	ISO 9003	CMM	CMMI	SPICE	IDEAL	OTHER
10001	0	0	0	1	0	0	0	0
10002	0	0	0	1	0	0	0	0
10003	0	0	0	0	1	0	0	0
10004	1	0	0	0	0	0	0	0
10005	0	0	0	0	1	0	0	0
10006	0	0	0	0	1	0	0	0
10007	1	0	0	0	0	0	0	0
10008	0	1	0	0	0	0	1	0
10009	0	0	0	0	1	0	0	0
10010	1	0	0	0	0	0	0	1
10011	0	0	0	1	1	0	0	0
10012	0	0	0	0	1	0	0	0
10013	0	0	0	0	0	0	0	0
10014	0	0	0	1	1	1	0	0
<b>TOTAL</b>	<b>3</b>	<b>1</b>	<b>0</b>	<b>4</b>	<b>7</b>	<b>1</b>	<b>1</b>	<b>1</b>

From Table 6, only 3 companies out of 14 make use of ISO 9001, 1 company out of 14 uses ISO 9002, none of the companies out of 14 uses ISO 9003, only 4 of the companies use CMM, 1 company out 14 uses ISO SPICE and 1 company out 14 uses IDEAL. It is only CMMI which seems to be used by the companies with the value of 7 out of 14 companies.

The possible reasons why these companies are not using these models is likely to be attributed to fact that they are Small and Medium Enterprises (SMEs) as it is explained in section 4.3. Figure 28 shows staff members identified in these companies. Furthermore, as companies are small, many of them do not have a budget for SPI or certification as shown in Table 8 and Table 9.

**Table 7 - Reasons why companies fail to use standards**

COMPANY NAME	WHY COMPANY NOT REGISTERED OR NOT USING MODELS/STANDARDS					
	Financial Constraints	Company still new and small	No idea about standards	Don't need them	Other	No Answer
10001	0	0	0	0	1	0
10002	0	0	0	0	0	1
10003	0	0	0	0	1	0
10004	0	0	0	0	0	1
10005	0	1	0	0	0	0
10006	1	1	0	0	0	0
10007	0	0	0	0	0	1
10008	1	0	0	0	0	0
10009	0	0	0	1	0	0
10010	0	0	0	0	0	1
10011	0	1	0	0	0	0
10012	0	0	0	0	0	0
10013	0	0	0	0	0	0
10014	0	0	0	0	0	0
<b>TOTAL</b>	<b>2</b>	<b>3</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>4</b>

From table 3 and 4, the results of the formal regression are also reflected in the individual variables (ISO 9001, ISO 9002, ISO 9003, CMM, CMMI, SPICE and IDEAL) as none of them are making any significant contribution in explaining the use of SPI models.

A follow up study was made to further assess the level of SPI in the country. Appendix D shows the questionnaire that was used. Results shows that the software companies in Botswana are not using CMM/CMMI. This shows that there is a need for software companies to brace up in the use of SPI models especially CMM/CMMI.

**Table 8 - Case Processing Summary on SPI Budget by Companies**

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
CompanyID * Budget_4_Certific	12	85.7%	2	14.3%	14	100.0%

**Table 9 - Budget for Software Process usage**

		Budget 4_Certific		Total
		Yes	No	
CompanyID	10001	1	0	1
	10002	0	1	1
	10003	0	1	1
	10004	0	1	1
	10005	0	1	1
	10006	0	1	1
	10008	1	0	1
	10009	1	0	1
	10011	0	1	1
	10012	1	0	1
	10013	1	0	1
	10014	1	0	1
Total		6	6	12

## **4.2 SOFTWARE COMPANIES – CERTIFICATION AND STANDARD APPLICATION**

### **4.2.1 Companies certified with ISO 9001**

Results also suggest that only less than a quarter (3) indicated having been registered with ISO 9001. This shows that companies in Botswana are not aware of or are not able to use this model.

**Table 10 - COMPANY CERTIFICATION ON ISO 9001**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Yes	3	21.4	21.4	21.4
No	11	78.6	78.6	100.0
Total	14	100.0	100.0	

#### **4.2.2 Companies certified with ISO 9002**

Looking at ISO 9002 certification, we still find few companies certified with the same. Although this standard is now combined with other standards like ISO 9001, ISO 9003 [52] results suggest that there is less than 10% of the companies which are familiar ISO 9002.

**Table 11 - COMPANY CERTIFICATION WITH ISO 9002**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Yes	1	7.1	7.1	7.1
No	13	92.9	92.9	100.0
Total	14	100.0	100.0	

#### **4.2.3 Companies certified with CMM**

The questionnaire also included questions on the CMM and CMMI model for the sake of those companies which might be still using CMM as well as those using CMMI. Only four

companies are registered with CMM. Table 12 and Table 13 show the results of companies certified with CMM and CMMI respectively.

**Table 12 - COMPANY CERTIFICATION WITH CMM**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Yes	4	28.6	28.6	28.6
No	10	71.4	71.4	100.0
Total	14	100.0	100.0	

#### 4.2.4 Companies certified with CMMI

The table below shows company certification on CMMI. This is the model that seems to be better utilized as compared to others.

**Table 13 - COMPANY CERTIFICATION WITH CMMI**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Yes	7	50.0	50.0	50.0
No	7	50.0	50.0	100.0
Total	14	100.0	100.0	

#### 4.2.5 Capability Maturity Level of Software Companies in Botswana

The Capability Maturity Model for software (developed by Carnegie Mellon University's Software Engineering Institute) has become a major force in software development process improvement. Companies strive to increase their CMM level from an initial level of 1 to level 2 through 5 [65].

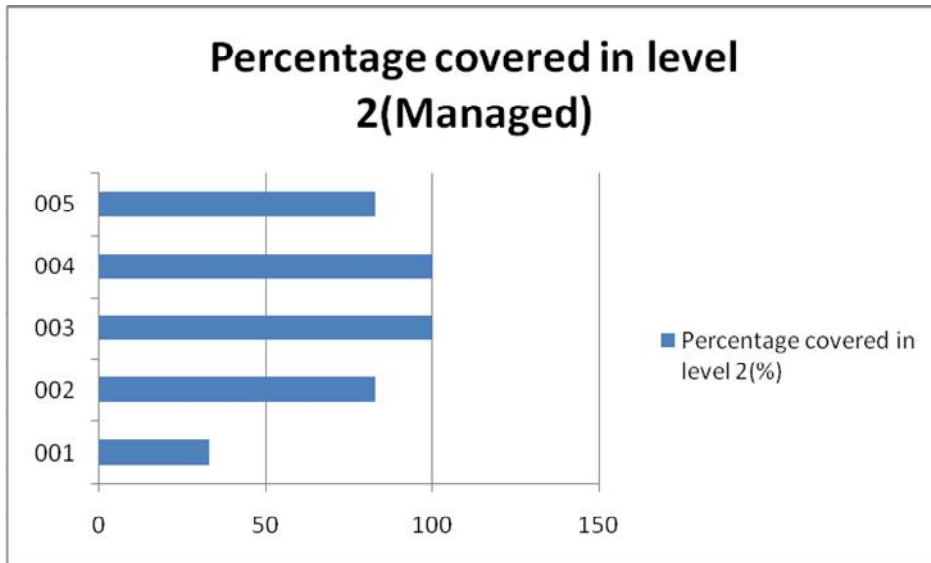


**Table 14 - CMMI level of organisation in Botswana**

Company ID	Level 2-Repeatable							Level 3 - Defined							Level 4 - Quantitatively Managed			Level 5 -Optimising				
	Requirement Management	Software Project Planning	software Project Training and Oversight	Software Subcontract Management	Software Quality Management	Software Configuration Management	Percentage covered in level 2(10%)	Organisational Process focus	Organisational Process Definition	Training Program	Integrated Software Management	Software Product Engineering	Intergroup Coordination	Peer Review	Percentage covered in level 3(3%)	Quantitative Process Management	Software Quality Management	Percentage covered in level 4%	Defect Prevention	Technology Change Management	Process Change management	Percentage covered in level 5%
001	1	1	0	0	0	1	33	1	1	1	0	0	0	0	42	0	1	50	0	0	0	0
002	1	1	1	0	1	1	83	0	0	0	0	0	0	0	0	0	1	50	0	0	0	0
003	1	1	1	1	1	1	100	1	1	1	1	1	1	100	1	1	100	0	0	0	0	
004	1	1	1	1	1	1	100	1	1	1	1	0	1	1	71	0	0	0	0	1	1	0
005	1	1	0	1	1	1	83	0	0	1	1	0	1	1	57	0	1	50	1	1	0	100

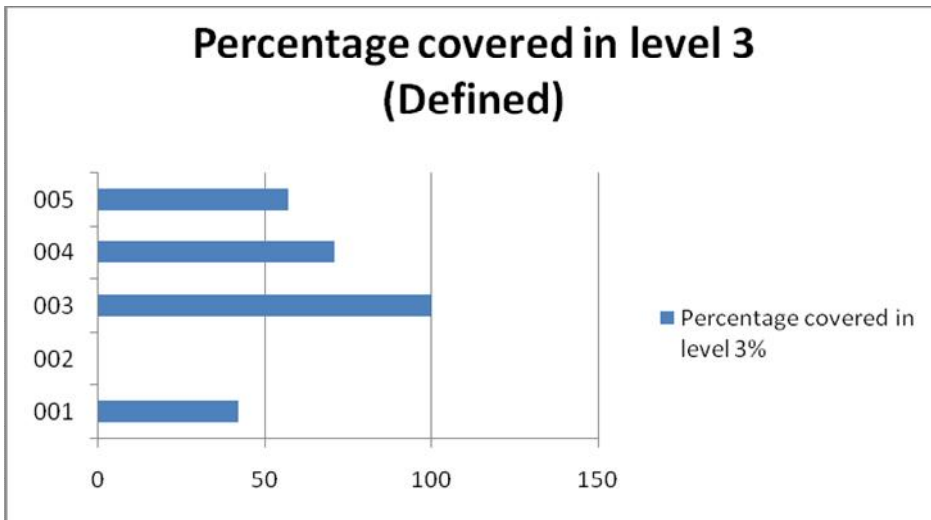
Table 14 shows Botswana software companies use of CMMI and their level of maturity. All together 7 companies namely ICL Botswana, FinDev, Baclays Bank Botswana, Stanbic Bank Botswana, DCDM, Web Logic and kuet Company indicated using CMMI as shown from table 6. Questionnaires (refer to appendix D) were administered to the companies to determine the company level of maturity using 18 key performance areas as shown in Table 14.

Results from table 14 are further analysed as shown in Figure 29 to Figure 32.



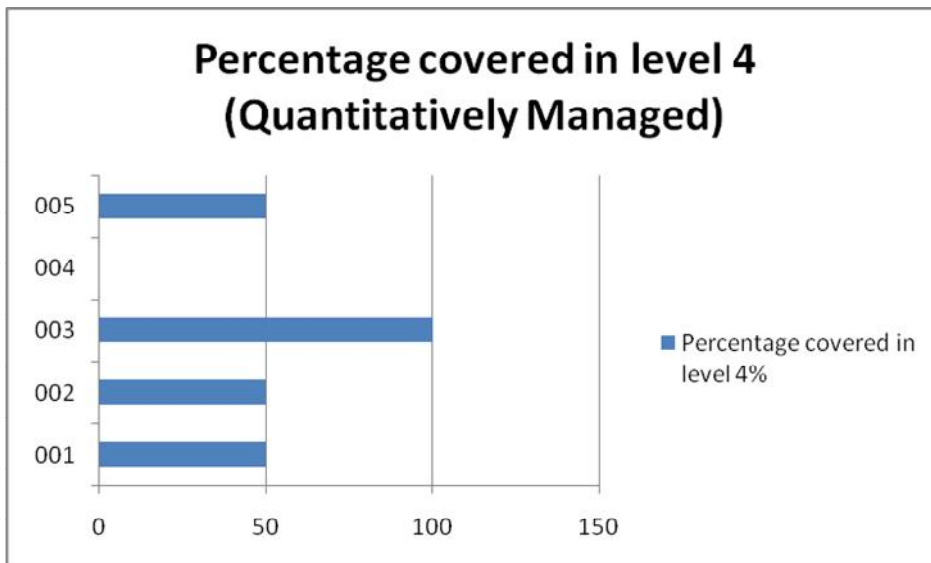
**Figure 29 - Percentage covered in level 2 (Managed)**

From figure 29, only two companies can be said to be in level 2, i.e company 004 and company 003. The rest of the companies has achieved 67% or less of level 2 (managed). According to Dangle, Laarsen and Shaw [65], an organisation that at none of the the KPAs is rated at the lowest level, called level 1(Initial).



**Figure 30 - Percentage covered in level 3 (defined)**

Figure 30 shows that company 003 is the only company that can be said to be in level 3, other companies are 71% or less. Company 003 is also the only company that has covered all the KPAs in level 4.



**Figure 31 - Percentage covered in Level 4**

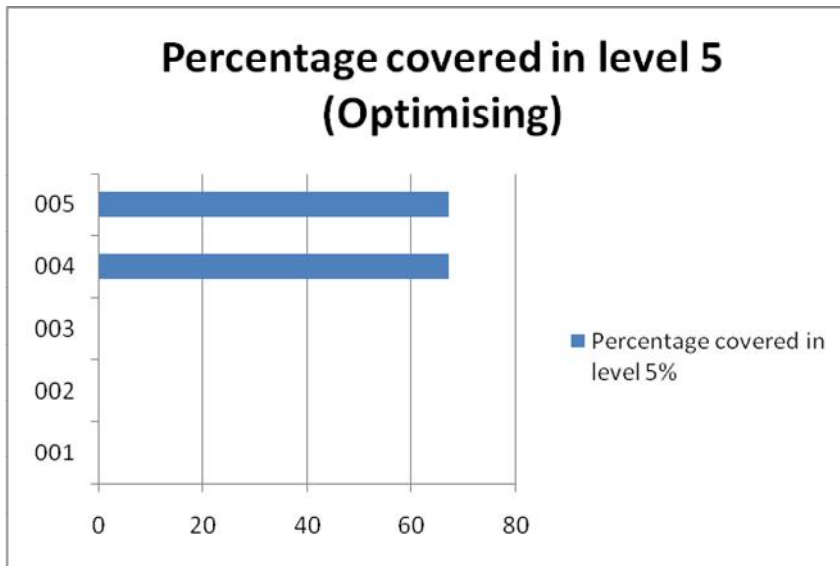


Figure 32 - Percentage covered in level 5

None of the companies is in level 5 as shown figure 32.

#### 4.2.6 Companies certified with ISO 9003

Among the companies given the questionnaires none of them is certified with ISO 9003.

Table 15 - COMPANY CERTIFICATION ON ISO 9003

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid No	14	100.0	100.0	100.0

The results of this questionnaire show that there are few companies in Botswana which are certified with international standards. Most of the companies had a challenge with completing part c of questionnaire A because they are not familiar with the many SPI models. The questionnaire provided a space for companies to fill whatever SPI model they are using, apart from the one listed, but still there were few that provided information there. Most of these companies show that they are SMEs. Botswana software companies need to use the existing models to improve their performance. Few

of these companies have very few systems they are developing and maintaining in a year.

### 4.3 COMPANIES STAFF CONCENTRATION

The options included 1 – 10, 11 – 25, 26 – 50, 51 – 100 and 101 plus. The results show that most companies have staff ranging from 26 – 50 non IT workers inclusive. This range is followed by that of 1-10. This range is in the range (10 - 50) [13], of enterprises that are referred to as (Small Medium Enterprises) SME [13]. Figure 33 to Figure 37 confirms that most of the companies in Botswana are SMEs. In countries like US, Canada, China, Finland, India, and many other countries, SMEs represent 85% of all firms [13].

#### 4.3.1 TOTAL NUMBER OF EMPLOYEES IN A COMPANY

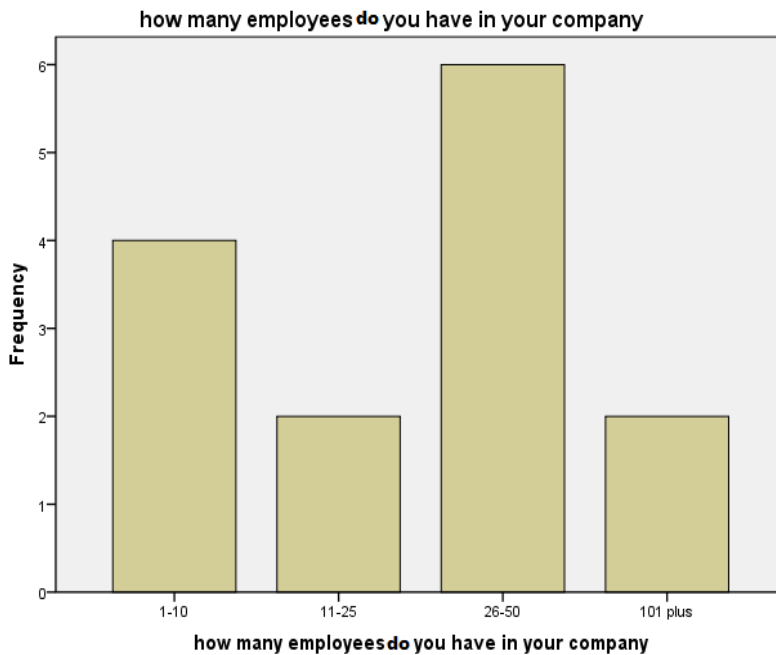


Figure 33- Staff population in Software companies in Botswana

In the Figure 33, most companies have between **26 – 50 employees** followed by the range of **1 – 10**. Only 4 companies fall in the range 11 – 25 and 101 and more. There are no companies that have number of staff ranging from 51 to 100.

### 4.3.2 FULLTIME IT PROFESSIONALS

The fulltime professionals shown in Figure 34 are IT professionals as this was extracted from Question 3 in Part A of Questionnaire A. The Figure shows that 13 companies have IT professionals ranging from 1 to 50 while only one has more 101 IT fulltime professionals. Again there is no company representing the range between 51 and 100. Most of the companies here are SMEs as pointed out by Misra D. e.t. al., [6]. Comparing results from Figure 34 and Table 16 shows that many companies have 1 – 10 fulltime IT professionals and 11 companies have 1 – 10 part-time IT professionals. This may imply that most of the work is done by the part-timers or consultants and these may not always be reliable. The fulltime employees are very few to do the work and hence it may not always be easy to adhere to the standards.

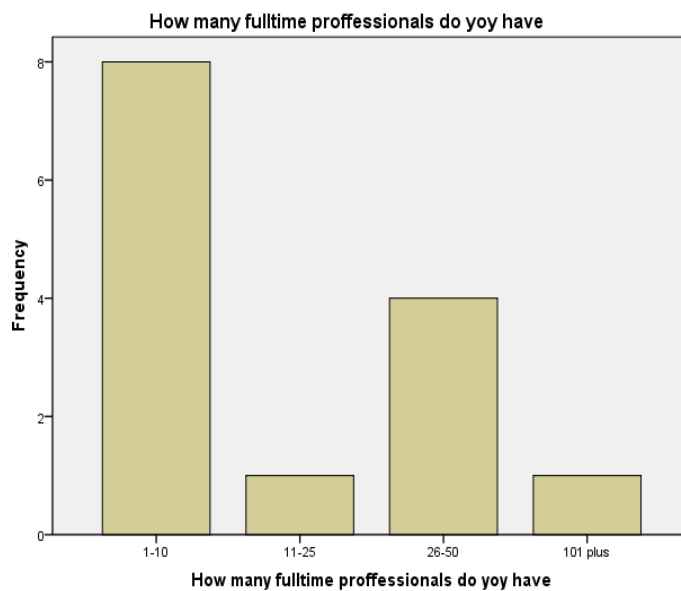


Figure 34 - Number of full time professionals

### 4.3.3 PART TIME PROFESSIONALS

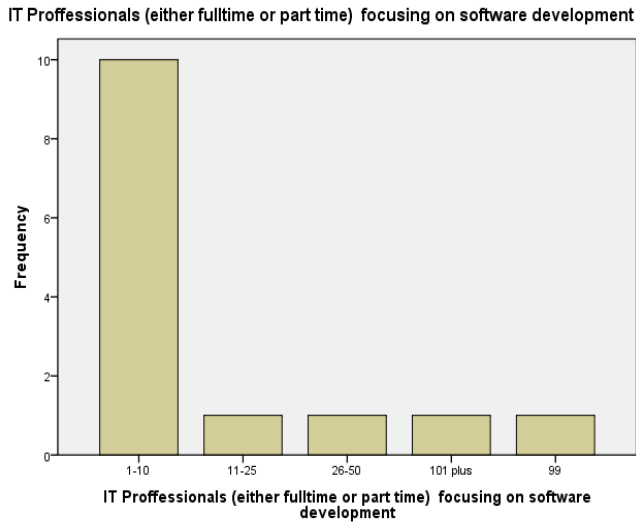
Table 16 - Number of part time IT Professionals

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid 0	3	21.4	21.4	21.4
1-10	11	78.6	78.6	100.0
Total	14	100.0	100.0	

Table 16 indicates that more than 10 companies out of 14 have part-timers ranging from 1 – 10. This part-timers includes consultants. This shows that these companies do consulting which further confirms the lacking of the local companies.

### 4.3.4 SOFTWARE DEVELOPERS (BOTH FULL-TIMERS AND PART-TIMERS)

Eleven companies here show that they have only less than 25 employees who are focus on software development. Combining the full-time and the part-time does not make much difference to the common results that the companies are SMEs.

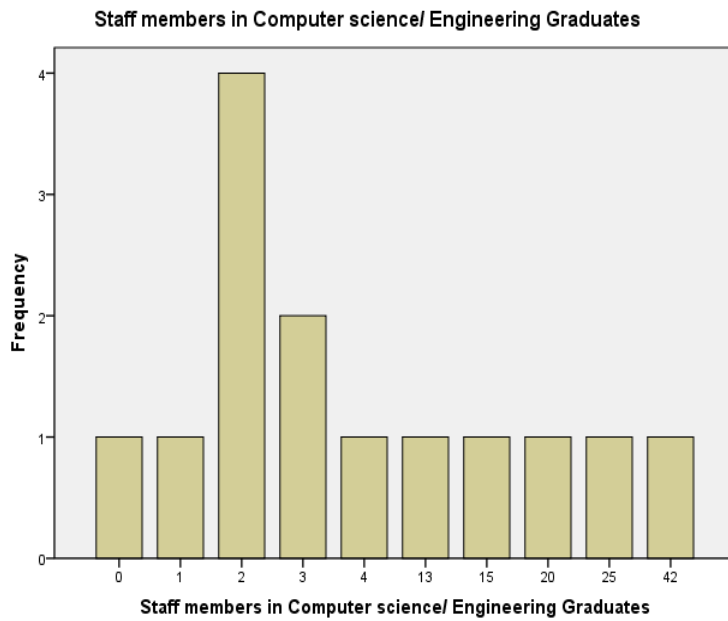


**Figure 35 - Number of full time and Part time professionals focusing on Software development**

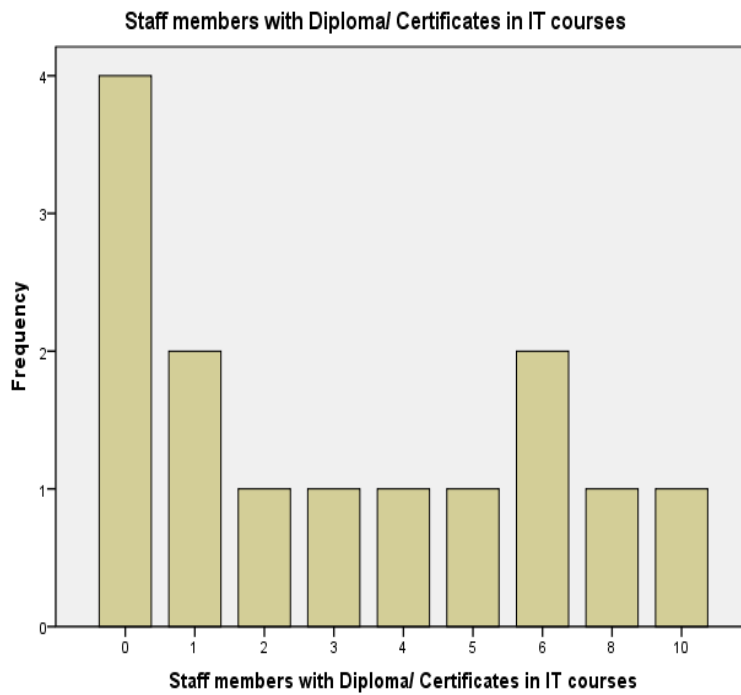
#### **4.3.5 COMPUTER SCIENCE GRADUATES**

To be more specific, actual number of diploma/certificates in computer science, computer science graduates and masters in Computer Science were collected from these companies. Generally, these companies have fewer graduates and less staff members with masters degree in computer science. This may be the reason why the standards are not applied. From all the companies, 8 of them do not have a computer science masters holder.





**Figure 36 - Graduates in Computer Science/Engineering Graduate**



**Figure 37 - Staff members with Diploma/certificate in IT courses**

**Table 17 - Staff with Masters Computer Science**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	8	57.1	57.1	57.1
	1	3	21.4	21.4	78.6
	2	1	7.1	7.1	85.7
	3	1	7.1	7.1	92.9
	4	1	7.1	7.1	100.0
	Total	14	100.0	100.0	

#### **4.3.6 SOFTWARE QUALITY CONTROL**

Software quality measures were taken into account to find out how software companies measure and assure quality. In Table 18, methods for measuring quality are tabulated against the company ID. Most companies measure quality by looking at lines of code and fog index. Half of the companies look at the number of reported faults. These issues are very important in assuring the quality of the software.

**Table 18 - Company Quality Measuring methods**

COMPANY NAME	Quality Measuring Method							
	Lines of Code	Fog Index	No. of Reported Faults	No. of Person days	Travel Cost	Computer Resources	Never consider quality	OTHER
10001	1	1	0	0	0	0	0	1
10002	1	1	0	0	0	1	0	1
10003	1	1	1	0	0	0	0	1
10004	1	1	1	0	0	0	0	1
10005	1	1	0	0	0	0	1	1
10006	1	1	1	1	0	0	0	1
10007	1	1	1	0	0	0	0	1
10008	1	1	1	0	0	0	0	1
10009	1	1	1	0	1	1	0	1
10010	1	1	1	0	0	0	0	1
10011	1	1	0	0	0	0	0	1
10012	1	1	0	0	0	0	0	1
10013	1	1	0	0	0	0	0	1
10014	1	1	0	0	0	0	1	1
<b>TOTAL</b>	<b>14</b>	<b>14</b>	<b>7</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>14</b>

Table 19 shows that companies generally don't have quality planning. Five companies consider customer view.

**Table 19 - Company Quality Assurance attempts**

COMPANY NAME	QUALITY ASSURANCE ATTEMPS				
	Quality Planning	Quality Control	Never Check Quality	Customer view	OTHER
10001	0	1	0	1	1
10002	0	0	0	0	1
10003	0	1	0	1	1
10004	1	0	0	0	1
10005	1	0	0	0	1
10006	0	1	0	1	1
10007	1	1	0	0	1
10008	0	0	0	0	1
10009	0	0	0	1	1
10010	0	0	0	0	1
10011	0	1	0	1	1
10012	0	1	0	0	1
10013	1	1	0	0	1
10014	0	0	1	0	1
<b>TOTAL</b>	<b>4</b>	<b>7</b>	<b>1</b>	<b>5</b>	<b>14</b>

The table shows that every company is concerned about quality assurance as a way of checking software products for quality.

**Table 20 - Company Available Software Quality Guidelines**

COMPANY NAME	Software Quality Guidelines						
	Quality Manual	Quality Policy	Written Org. Policy	Document Procedure for PI	Test Plan	Written Guidelines for Software design	Written Guidelines for code Generation
10001	1	1	1	1	1	1	1
10002	1	1	1	1	1	1	1
10003	1	1	1	1	1	1	0
10004	1	1	1	1	1	1	1
10005	1	1	0	0	0	0	0
10006	1	1	1	0	1	1	1
10007	1	1	1	1	1	1	1
10008	1	1	1	1	1	1	1
10009	1	0	1	1	0	0	0
10010	1	1	1	1	1	1	1
10011	1	0	0	1	1	0	0
10012	1	1	1	1	1	1	1
10013	1	1	0	0	1	0	0
10014	1	0	0	0	1	0	0
<b>TOTAL</b>	<b>14</b>	<b>10</b>	<b>10</b>	<b>10</b>	<b>12</b>	<b>9</b>	<b>8</b>

Table 20 shows that many software companies (74.5%) have some software quality guidelines. This is impressive but question is how are they using them.

#### 4.3.7 NUMBER OF SOFTWARE PRODUCTS DEVELOPED/CUSTOMISED AND MAINTAINED BY THE COMPANIES

Table 21 - Number of software Products developed between 2012 and 2014

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0-1	4	28.6	30.8	30.8
	2-5	6	42.9	46.2	76.9
	6-10	1	7.1	7.7	84.6
	11-20	1	7.1	7.7	92.3
	21 plus	1	7.1	7.7	100.0
	Total	13	92.9	100.0	
Missing	-9	1	7.1		
Total		14	100.0		

Table 21 shows that the companies are not doing much of development. This may also be the reason why companies are not familiar with the SPI standards. More companies do more maintenance than developing as shown by Table 22.

**Table 22 - Number of software Systems maintained**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid 0-1	1	7.1	7.1	7.1
2-5	1	7.1	7.1	14.3
6-10	5	35.7	35.7	50.0
11-20	3	21.4	21.4	71.4
21 plus	4	28.6	28.6	100.0
Total	14	100.0	100.0	

Agile methodology is one of the models that are doing well in the industry today, but among these companies only 5 of them are using this methodology as shown in Table 23.

**Table 23 - Use of Agile Methodology**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Yes	5	35.7	35.7	35.7
No	9	64.3	64.3	100.0
Total	14	100.0	100.0	

#### 4.3.8 UTILISATIONS OF SOFTWARE PROCESS MODELS

Although the companies are not highly utilizing the entire software process model, 12 of the companies make use of at least one model. Only two companies from the list are not using any of the mentioned models. Agile model seems to be used more than the other model. This is depicted by Table 24. In Table 25, 11 of the companies attest that the models are very useful.

**Table 24 - Company utilization of software process models**

COMPANY NAME	USE OF SOFTWARE PROCESS MODEL									TOTAL FOR EACH MODEL BY COMPANY
	Waterfall	Spiral Model	V model	RAD	Agile	Structured Programming	Dynamic system Analysis	Evolutionary Development	Reuse Based Dev.	
10001	0	0	0	1	0	1	0	0	0	2
10002	0	0	0	0	0	0	1	0	0	1
10003	1	0	0	1	1	1	0	0	0	4
10004	0	0	0	0	0	0	0	0	1	1
10005	1	0	0	0	0	0	0	0	0	1
10006	0	0	0	1	1	1	0	0	1	4
10007	0	0	0	1	1	0	0	0	0	2
10008	0	0	0	0	0	0	1	0	0	1
10009	0	0	0	0	0	0	0	0	0	0
10010	0	0	0	0	0	0	0	0	1	1
10011	0	0	0	0	1	0	0	0	0	1
10012	0	0	0	0	0	0	0	1	0	1
10013	1	1	1	0	1	0	0	0	0	4
10014	0	0	0	0	0	0	0	0	0	0
<b>TOTAL</b>	<b>3</b>	<b>1</b>	<b>1</b>	<b>4</b>	<b>5</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>4</b>	



**Table 25 - Usefulness of software process models**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid useful	2	14.3	14.3	14.3
very useful	11	78.6	78.6	92.9
not sure	1	7.1	7.1	100.0
Total	14	100.0	100.0	

#### **4.4 ORGANISATIONAL USE OF SOFTWARE PRODUCTS**

Government and business organisations/departments were also considered in this study. This is for the very reason that these organisations are the ones that are clients to most of these software companies.

There is significant number of foreign companies as shown in Table 26 and Table 27 providing service (to both the government and some business department). This shows that the local companies may still be lacking in their services. The government of Botswana usually assists SMEs by offering or reserving certain businesses, tenders to local companies [66], but the results here still shows that few local companies are involved in maintaining and providing service to the government and business department in Botswana.

**Table 26 - Foreign and local companies supporting organisations in Botswana**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Foreign	4	25.0	28.6	28.6
	Local	5	31.3	35.7	64.3
	Local and Foreign	5	31.3	35.7	100.0
	Total	14	87.5	100.0	
Missing	-9	2	12.5		
Total		16	100.0		

**Table 27 - Maintenance by Local Companies**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	6	37.5	37.5	37.5
	No	10	62.5	62.5	100.0
Total		16	100.0	100.0	

Table 28 shows that almost every organization has an information system, but they are mainly serviced by foreign companies. Of all the 14 software companies identified in this study, only a few of them have developed s/w products between the months of January 2012 and June 2014. This really shows that few of these companies are into software development. The results of the study also reveal that there are few IT professionals (developers, programmers, managers and analysts) as shown in Figure 33 to Figure 37 which may likely contribute much more to this situation.

**Table 28 - Availability of software products in organisation**

		<b>Frequency</b>	<b>Percent</b>	<b>Valid Percent</b>	<b>Cumulative Percent</b>
<b>Valid</b>	<b>Yes</b>	<b>14</b>	<b>87.5</b>	<b>87.5</b>	<b>87.5</b>
	<b>No</b>	<b>2</b>	<b>12.5</b>	<b>12.5</b>	<b>100.0</b>
	<b>Total</b>	<b>16</b>	<b>100.0</b>	<b>100.0</b>	

On issues of purchasing of ready-made software products, the results on Table 29 shows that the government departments are involved in buying products which are ready-made. Some software companies also buy ready-made products for their clients. This is worrisome as the IT professionals are not given the opportunity to learn and develop applications for their departments/organisations. There are also few cases in the organisations where IT professionals are given an opportunity to develop for their departments. Among the 16 organisations that were chosen in Questionnaire B, 13 of them have some companies (either foreign or local) that are supplying them with software products or are developing for them. Furthermore only 5 and 6 departments out of 16 do in-house development and customization respectively. This suggest that in-house development is been replaced by externally developed software [67].

**Table 29 - Purchasing of Ready-made Products**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	15	93.8	93.8	93.8
	No	1	6.3	6.3	100.0
	Total	16	100.0	100.0	

Many organisations buy ready-made products.

**Table 30 - Software Product Customization**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	6	37.5	37.5	37.5
	No	10	62.5	62.5	100.0
	Total	16	100.0	100.0	

Some government and business organisations customize the software products. This means there will be less business opportunity to software companies.

**Table 31 - Last Software Purchase**

		<b>Frequency</b>	<b>Percent</b>	<b>Valid Percent</b>	<b>Cumulative Percent</b>
<b>Valid</b>	<b>a year ago</b>	<b>12</b>	<b>75.0</b>	<b>75.0</b>	<b>75.0</b>
	<b>2 years ago</b>	<b>2</b>	<b>12.5</b>	<b>12.5</b>	<b>87.5</b>
	<b>5 years ago</b>	<b>1</b>	<b>6.3</b>	<b>6.3</b>	<b>93.8</b>
	<b>5 years plus</b>	<b>1</b>	<b>6.3</b>	<b>6.3</b>	<b>100.0</b>
	<b>Total</b>	<b>16</b>	<b>100.0</b>	<b>100.0</b>	

Table 31 shows that organisations are much active in buying software products. This implies that organisations are using software products. One other possible reasons why the software companies are struggling may be because of lack of business opportunities in the country. The annual budget presented by the organization is very low. Though the figures may be untrue, they show a picture of less organization having a sufficient annual budget for software. Only 6 organizations (Table 32) out of the 16 have an annual software budget above a million which is very low.

Table 32 - Organization annual software estimate

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	500.00	1	6.3	7.7	7.7
	5000.00	1	6.3	7.7	15.4
	60000.00	1	6.3	7.7	23.1
	200000.00	1	6.3	7.7	30.8
	400000.00	1	6.3	7.7	38.5
	500000.00	2	12.5	15.4	53.8
	1000000.00	3	18.8	23.1	76.9
	1500000.00	2	12.5	15.4	92.3
	8000000.00	1	6.3	7.7	100.0
	Total	13	81.3	100.0	
Missing	-9.00	3	18.8		
Total		16	100.0		

## 4.5 DISCUSSION

A general observation from this study is that most software companies in Botswana may not be fully making effective use SPI model. Table 5 suggests that 3 companies are familiar with ISO 9001, 1 company familiar with ISO 9002, no single company familiar with ISO 9003. SPICE and IDEAL model also recorded 1 company each. Only one company is familiar with other models not listed. It appears that the only SPI being used is CMM and CMMI with the record of 4 (28.5 %) and 7 (50%) companies respectively. This suggests that most software companies in Botswana are not using SPI models (Research Question bullet 1, 2). This is slightly above the results obtained in the study conducted in Malaysia [15], though more companies were studied.

Furthermore, using CMM/CMMI to verify the capability status of the 7 companies who claim familiarity with CMM/CMMI suggested that only 1 company is actually CMM/CMMI compliant (see figure 29 - 32)

The relevance of this SPI models to Botswana software industry is questionable as only one company seem to be compliant with CMM and CMMI (Research Question bullet 4), while 3 companies are at levels 2 and 3 of CMM/CMMI. Having few companies using these standards as shown in the Table 6 may also mean the models are not relevant to Botswana software companies as they are small (Research Question bullet 4).

The size of the software companies in Botswana is clearly shown from the Figure 33, 34, 35, 36, 37 and Table 16, 17. All figures and table shows that the average staff of Botswana companies falls in the range of 1 – 50 employees. This suggests that most of software companies in Botswana are SMEs and Very Small Entities (VSE) [67] (Research Question bullet 3).

Generally, software companies in Botswana have some level of awareness on quality control. Most of them use lines of code, fog index and faults reported by the customer

to measure the quality of their systems. They have a number of guidelines that help them to ensure quality. Table 18 shows that 75.5% of them have these guidelines (Research Question bullet 5, 6).

How companies utilize software process models was also investigated in this study (Research Question 7). From the questionnaire response, the results (Table 24 and Table 25) show that companies/developers find the software process models important. Only two companies indicated that they do not use any of the models while the rest of the companies use at least one of the models. All the companies except one (who selected *not sure*) indicated that the models are useful, or very useful.

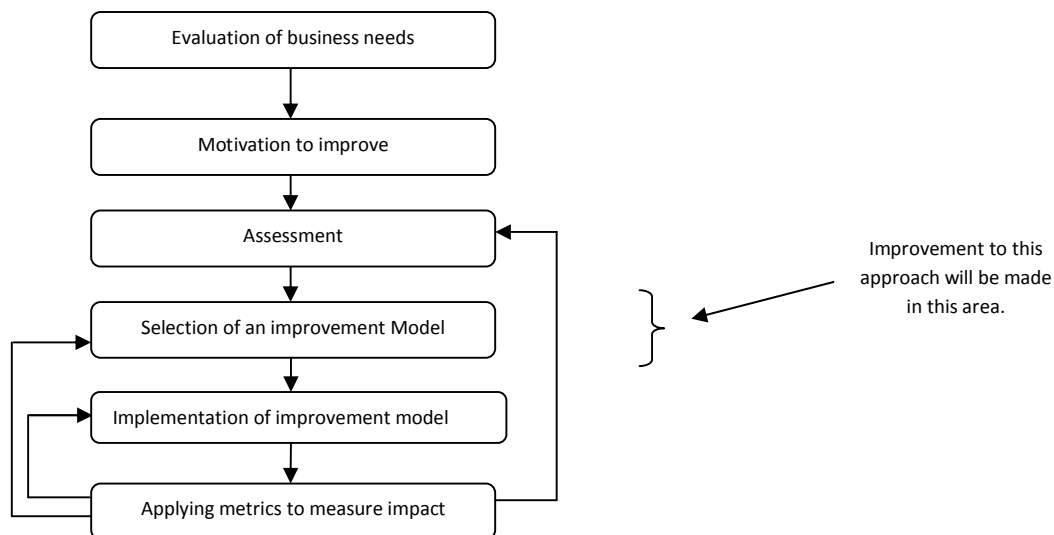
Software companies need to be motivated to use Software Process Improvement (SPIs) in order for them to improve on their products and services. They need to apply software quality procedures, use software estimation techniques, and carry software process assessment regularly so that they can do better. The results clearly show that software companies in Botswana are mostly Small Medium Enterprises (SME), and most of them do not use Software Process Improvement (SPIs). To motivate these companies to use SPIs, *SME SPI Selection Framework* is proposed.



## 4.6 PROPOSAL OF A SME SPI SELECTION FRAMEWORK

### 4.6.1 Process improvement Approach Model

In process improvement, there is a need to carry out evaluation of business needs, which will bring about motivation to improve. When there is motivation to improve, then there is a need for assessment. After that there is a need to choose from the existing models, implement the improvement model and then apply metrics to measure impact. This approach is represented in Figure 38.



**Figure 38 - Process improvement approach, Source [10]**

The systematic execution of carefully planned software improvement is needed for successful software improvement. Most successful companies use a pattern for process improvements which consists of six stage improvement program [10]. The first stage is *Evaluation of business needs*: Business needs drive companies to improve their services. They come as a result of strong completion, external regulations or a call for increased profitability [10]. *Motivation to improve* is the second stage. Business needs build up the motivation to improve and the next stage will automatically follow, which is the assessment stage. *Assessment* is the important stage in achieving the company goal of

software process improvement. Assessment is an appraisal, or review, performed by a trained team of software professionals [46]. Assessment will then lead to selection of an improvement model. Software process assessment has been explained in detail in section 2.6. After a careful assessment of the processes, the next stage that follows is the *Selection of an Improvement Model*. Model to be used is selected from the existing models. This is where we make our improvement to this approach to come up with a framework. After the model is chosen or selected, it will be then implemented. This stage is called the *Implementation of improvement model*. The stage: “*Applying metrics to measure impact*” is the final stage. This is where the metrics are applied to check if there are improvements. The cycle will go back to assessment so that more improvements will be achieved.

#### 4.6.2 PROPOSED FRAMEWORK

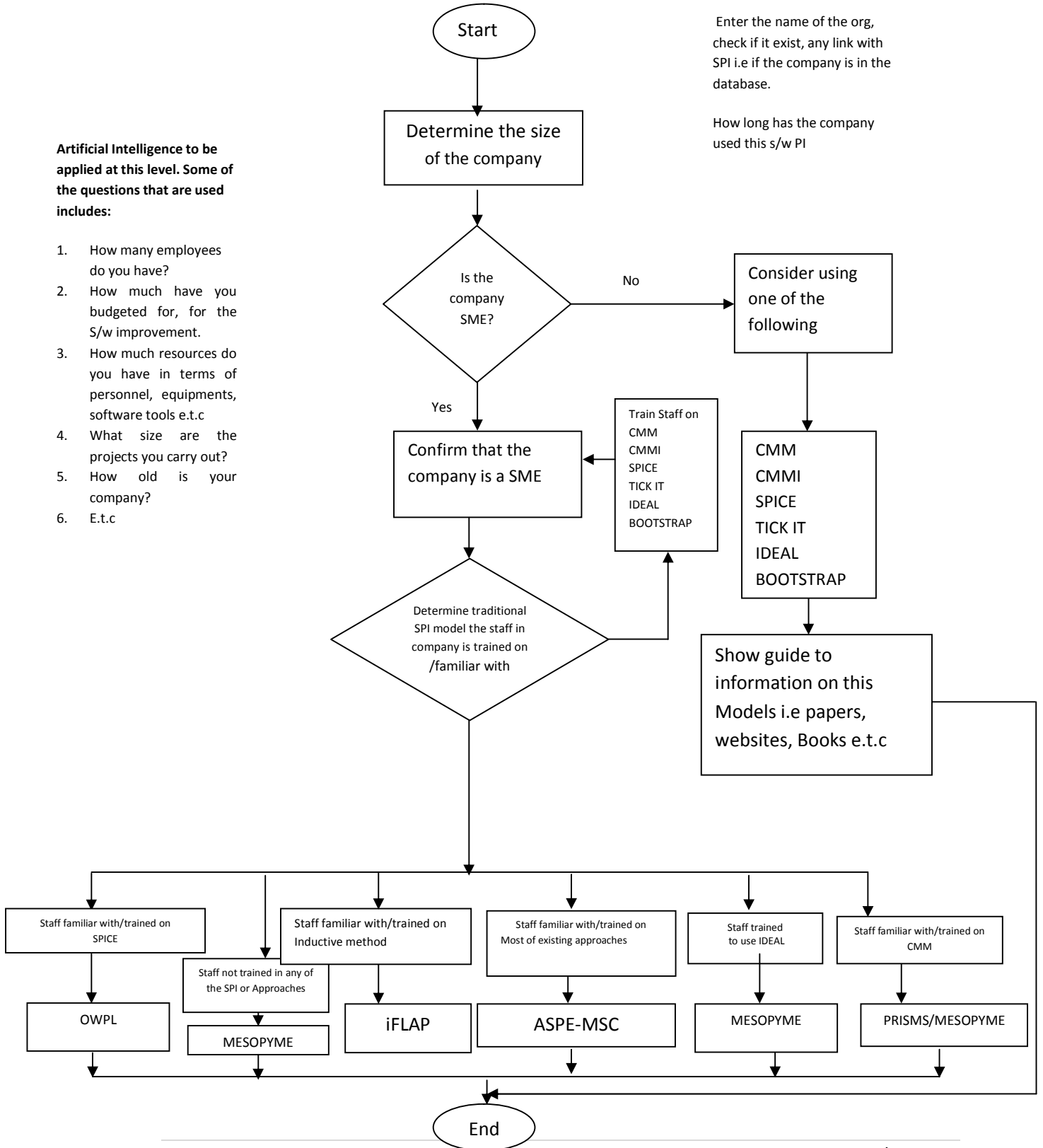
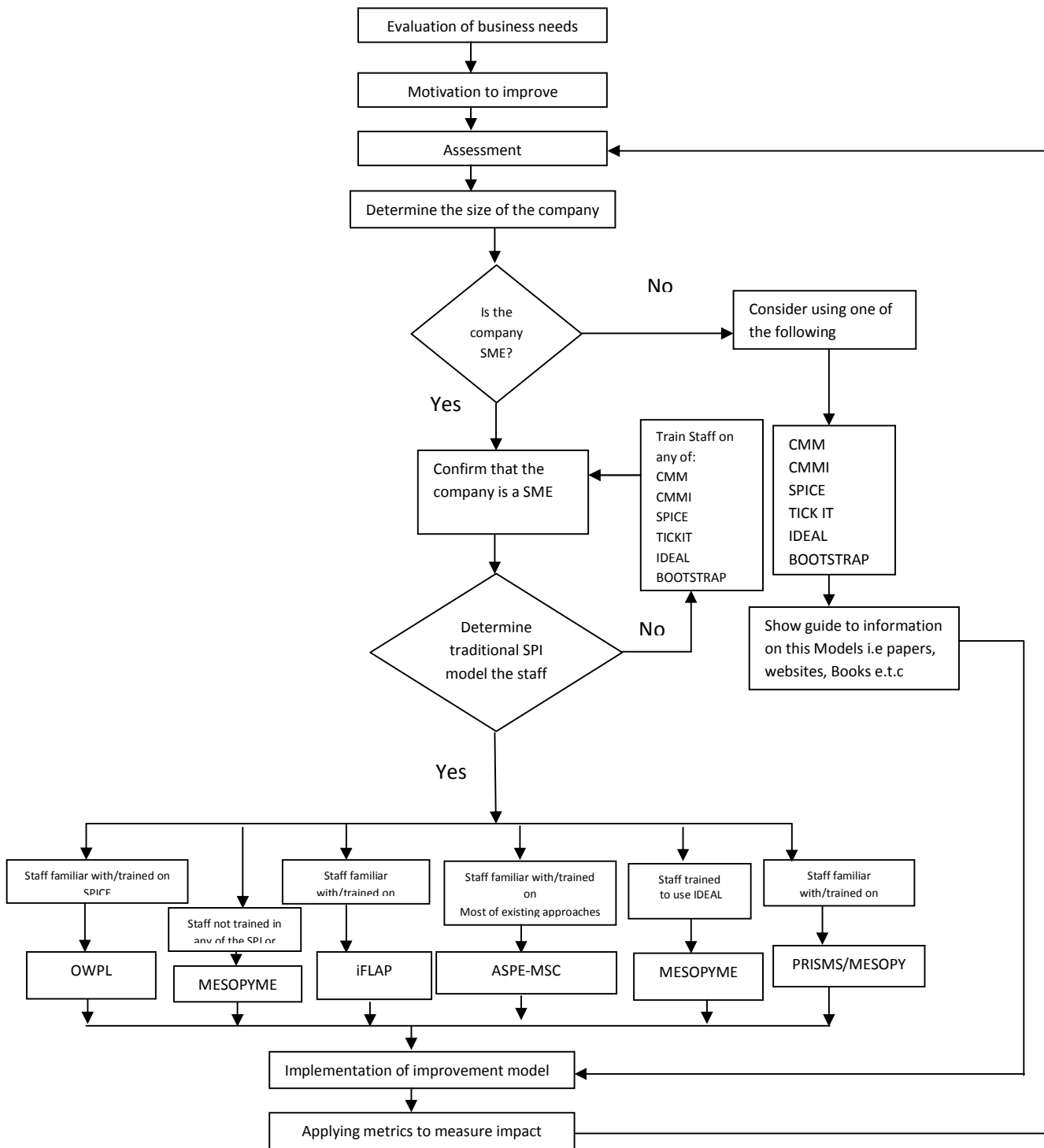


Figure 39 - SME Selection Framework portion

The need for an SPI framework for SMEs is informed by the fact that the companies have not been using available SPI standards. In particular SMEs are not using the capability maturity model (CMMI) and other standards. This study proposes that SMEs consider other models that are tailored for SMEs through this framework.

Figure 39 shows the Software Process Improvement (SPI) selection framework portion which will later be incorporated. The framework seeks to help small organization to choose the appropriate model for improvement. Improving the *Selection of an Improvement Model* stage makes it easier for the Small Medium Enterprises (SME) to select Software Process Improvement. This portion is incorporated in the approach shown in figure 38.

In this modification, the first thing is to determine whether the company is Small Medium Enterprise (SME), by determining its size which is influenced by the number of employees, other resources owned by the enterprise, the kind of projects carried out by the company – whether small or big, the annual budget they have as well as the company age. The company that does not fall under SMEs is encouraged to use traditional Software Process Improvement (SPI) models as shown in the framework. This is in line with what H Saiedian and K Chennupati pointed out [10], that large companies do not have a problem with using the common Software Process Improvement (SPI) models like CMM, SPICE e.tc. If the company is SME, then the next thing is to determine the kind of SPI the staff is familiar with. Most of the Software Process Improvement (SPI) designed for Small Medium Enterprises (SMEs) are related in a way to traditional ones [5]. Having members of staff who have been trained in the traditional models, is one guiding factor that is used in this model to select the right SPI model for the SME should be able to help selecting the right Software Process Improvement (SPI) for SME. We suggest that if the company has more staff experienced or trained in one traditional model, then the company should choose SME SPI model related to that model. This is the major idea behind this model.



**Figure 40 - SME Selection Framework**  
Adapted from Figure 38, source [10]

If none of staff members is skilled in any of the traditional models, then the company is advised to train at least one of the staff so that the right model will later be chosen.

After the SME SPI model is chosen, then it will be implemented. The implementation includes improving variables that will lead to a definite improvement. This variables include: business goals, budget for software development, defining of document process, assessment of suppliers process suitability, implementation of software design, planning of workflow and estimates. The model formula;

$$y_{spi} = (x_1, x_2, x_3 \dots x_n)$$

is applied at this stage.

After the implementation, the relevant metrics applied to measure the impact. After the metrics are applied, the assessment is done again and the cycle repeats again. This is shown in the Figure 38.

The *SME SPI Selection Framework* will help and encourage the managers to pay attention to Software Process Improvement (SPI) as they will be able to see how to choose a model among many existing Software Process Improvement (SPI) models. Software companies need motivation to successfully implement Software Process Improvement (SPI) standards and models [68]. This model will also encourage managers to read and as a result improve their knowledge on software process improvement. This will create awareness in many software organizations about the Software Process Improvement.

#### ***Using the staff experience as a mode of selection***

Most organizations employ people according to their needs, aims and the goals they have. Knowing the skills of the organization, will help to select the model that closely represent the aims and the goals of the given organisation.

### 4.6.3 RELATING OUR MODELS

Given the SPI model,

$$y_{spi} = (x_1, x_2, x_3 \dots x_n) \quad (1)$$

And the fitting model for the study

$$\alpha_{ij} = \beta_0 + \beta_1 X_1 J_1 + \beta_2 X_2 J_2 + \dots \beta_n X_n J_n \quad (2)$$

Both models are related and useful in this study. From (2) the constant factor is company name while the other variables are Software Process Improvement (SPI) in use.

## **CHAPTER 5 : SUMMURY AND CONCLUSIONS**

### **5.0 RESEARCH SUMMARY**

It is not enough to ask if software companies in Botswana make use of the Software Process Improvement (SPI), but it is important to ask if they are using it properly, fully or partially. The only way to verify this is by appropriate model fitting as this has been done in this study, in order to test the significance of the SPI models in Botswana.

The study of software companies on the use of the Software Process Improvement (SPI) model, suggest that the companies are not using the models fully or satisfactorily. This needs further investigation with more data covering more companies and more variables.

Attaining good software quality is usually a challenging goal to achieve. Nevertheless, software companies can achieve far much better products if they follow the right procedures which are recommended in the software field. Big and Small Medium Enterprises can highly benefit from the existing SPIs as long as they know how to select the proper model for them. The organizational cost can also be minimized if the right model is chosen. This is because expensive SPI programs will be avoided for SMEs.

### **5.1 RESEARCH CONTRIBUTION**

The research contributed in the following ways;

- Motivating software companies in Botswana to rise up to the challenge of using Software Process Improvement (SPIs).
- Developing a framework that will help software companies to choose from the existing model.



## **5.2 RESEARCH CONSTRAINTS**

The following were the constraints encountered while conducting the study;

1. **Reliable data:** obtaining a reliable data was a major challenge in this study. Most of the software companies just exist in paper but in a reality they are not there, or they are not fully functional. This problem limited the number of companies considered during data collection. It was also difficult to get hold of managers of the operating companies to get the more accurate data as some of the staff members were not aware of some other important facts.
2. **Research Time:** Data collection from sizable number of software companies needed more time for travelling. Some people preferred to remain with the questionnaire to complete it later. This required extra time for the collection of the completed copy.
3. **Part Time study:** Carrying out this research as a part time student has not been easy. It was not easy for me to work after long working ours as I was most of the time tired because of busy days at work.

## **5.3 FUTURE WORK**

Further study is needed with larger sample size of data. The study can be improved by using managers and senior experienced staff from companies and organisations to complete the questionnaires so that more accurate data is collected. Further improvements can also be done to the framework to suit all Small Medium Enterprises (SME) all over the globe.

Framework can be tested with more accurate data.

## BIBLIOGRAPHY

- [1] Ian Sommerville Software Engineering, 6<sup>th</sup> Edition, 2001.
- [2] Cohen, Assessing D the business value of software process improvement using CMMI in South Africa, Pretoria, 2008
- [3] Business Dictionary, Available Online:  
<http://www.businessdictionary.com/definition/performance.html>, [Accessed September 9, 2015]
- [4] E.Figueiredo, Software Standards, Available Online:  
[http://homepages.dcc.ufmg.br/~figueiredo/disciplinas/lectures/software-standards\\_v01.pdf](http://homepages.dcc.ufmg.br/~figueiredo/disciplinas/lectures/software-standards_v01.pdf), [Accessed 31 August 2015]
- [5] Mishra D, Mishra A, Software Process Improvement in SMEs: A Comparative View, Ankara, Turkey; June 2009
- [6] [http://www.iso.org/iso/iso\\_9000](http://www.iso.org/iso/iso_9000)
- [7] Software Development Process, Available online: [www.computerhope.com](http://www.computerhope.com), [Accessed May 29, 2015]
- [8] K. R. Harisha,  
[http://www.streetdirectory.com/travel\\_guide/135488/software/software\\_development\\_processes\\_and\\_its\\_importance.html](http://www.streetdirectory.com/travel_guide/135488/software/software_development_processes_and_its_importance.html), Software Development Process and its Importance [Accessed 31/08/15].
- [9] Margaret K. Kulpa, Kent A. Johnson (2006) Interpreting the CMMI, 2<sup>nd</sup> Edition
- [10] Saiedian H, Chennupati K, Towards an Evaluative framework for software process improvement models, USA, 1999.
- [11] Stambollian A, Laporte C, Najihabra, Renault. OWPL: A light Model & Methodology for Initiating Software Process Improvement, Luxemburg; 4-5, May, 2006
- [12] Paulish D. Case Studies of Software Process Improvement Methods, United State, December 1993
- [13] Allen P, Ramachandran M, Abushama H, an Approach to Software Process Improvement for Small to Medium Enterprises, Leeds; 2003.

- [14] Christiane G. Wangenheim, Alessandra Anacleto, Clenico F. Salviano, MARES – A Methodology for Software Process Assessment in Small Software, Brazil, Univali; 01 September 2004
- [15] F. Shukor Sanim, R. Nuraminah, M. Nasir, The Extent of Software Process Improvement Models in Malaysia: A Focus on CMMI Models, Malaysia, 2009.
- [16] Backhaus A, Karg L, Graf C, Grottke M, Neumann D. A Decision Support Scheme for Software Process Improvement Prioritization, Germany, 2011
- [17] SEI (2010); <http://www.sei.cmu.edu/about/about.html>
- [18] Islam Z, ZHOU X. Software Process Improvement Framework for Software Outsourcing Based On CMMI, Sweden; August 2011
- [19] Jones and Bartlett Publishers, p.p 74 – 106, Available Online:[http://samples.ibpub.com/9780763785345/85345\\_CH04\\_Tsui.pdf](http://samples.ibpub.com/9780763785345/85345_CH04_Tsui.pdf) [Accessed May 2015]
- [20] Basic Web Process Model, Available online: <http://flylib.com/books/en/4.324.1.19/1>, [Accessed May 29, 2015]
- [21] Rogers S. Pressman (2005) Software Engineering, 6<sup>th</sup> Edition
- [22] Barry Boehm, Spiral Development: Experience, Principles, and Refinements, Pittsburgh, 2000
- [23] B. Boehm and W.J Hansen, Spiral Model as a Tool for Evolutionary Acquisition, California, 2001
- [24] M. Pyhajarvi, SPICE – International Standard for Software Process Assessment, Finland, November 2004.
- [25] A. Aggarwal 1<sup>st</sup>, T. Sharma, Study and Comparison of Software Process Improvement
- [26] B. Basumatary, N. Baruah, H. Borgohain, A Study of Software Process Improvement Approaches Proposed for SMEs,
- [27] Carnegie M. The IDEAL Model, Software Engineering Institute
- [28] Mirea A, Available online: <http://www.slideshare.net/AdrianMirea/the-ideal-model>, The IDEAL Model, [Accessed November 5, 2014].

- [29] Available online: <http://www.tickit.org/scheme.htm>, TickIT Scheme, [Accessed November 5, 2014].
- [30] M. E. Fayad, M. Laitinen and R. P. Ward, "Software Engineering in the Small", Communications of the ACM, vol. 43, (2000), pp. 115-118.
- [31] Claude Y. Laporte, Normand Séguin, Gisele Villas Boas and Sanyakorn Buasung. Seizing the benefits of software and systems engineering standards, 2013
- [32] J.M. Desharnais, C.Y. Laporte, A. Stambollian, M. Zarour, N. Habra, Initiating Software Process Improvement with a light model for Small Enterprise: Our Experience, Canada
- [33] Pressroom, Available online: <https://www.cetic.be/OWPL-Assessment>, OWPL Assessment, [Accessed November 5, 2014].
- [34] Mishra D, Mishra A, Software Process Improvement Methodologies for Small and medium Enterprises, Ankara, Turkey;
- [35]Wangenheim C, Hauck J, et al, Experiences In Training for Software Process Improvement, Brazil, 2008
- [36] J.C. R. Hauck, C.G.Wangenheim, R.H Souza, Marcello Thiry, Process Reference Guides – Support for improving Software Process in Alignment with Reference Models and Standards, Brazil, 2008
- [37] Bos K, Method Description: IFlap
- [38] Koen B. Method Description: IFLAP,
- [39] K. Bos, Method Description: iFLAP, Available online: <http://foswiki.cs.uu.nl/foswiki/pub/MethodEngineering/LightWeightSoftwareProcessAssessmentAndImprovementPlanning/iFlap-3139409-kbos.pdf>, [Accessed May 2015]
- [40] D. Mishra, A. Mishra, A Study of Software Process Improvement in Small and Medium Organisations, Turkey, 2008
- [41] Software Process Improvement Standards& Specification, 2009
- [42] Calvano-Manzano J, Available online: [http://oa.upm.es/19340/1/INVE\\_MEM\\_2006\\_142237.pdf](http://oa.upm.es/19340/1/INVE_MEM_2006_142237.pdf), A Software Process Improvement Solution for Small and Medium-Size Enterprises [Accessed December 18, 2014]
- [43] ISO, Available online: <http://www.iso.org/iso/home/standards.htm>

- [44] D. Mishra and A. Mishra, Software Process Improvement Methodologies for Small and Medium Enterprises, Turkey, 2008 -  
[http://www.researchgate.net/publication/259331530\\_Lecture\\_Notes\\_in\\_Computer\\_Science](http://www.researchgate.net/publication/259331530_Lecture_Notes_in_Computer_Science)
- [45] H. Rever, Applying the DMAIC steps to Process Improvement Project, IIL
- [46] Kitson D, Humphrey W. The role of assessment in Software Process Improvement, Pittsburgh, 1989
- [47] P. Suri, P. Ranjan, Comparative Analysis of Software Effort Estimation Techniques, India, June 2012, pp. 12-19.
- [48] Peter Andras, Available online: <https://www.staff.ncl.ac.uk/peter.andras/annintro.ppt>, Artificial Neural Networks. [Accessed May 2015]
- [49] Johannesen R. Software Engineering in the Small: Is Chaos Likely to Fall?, Sunderland;
- [50] Ayalew Y. and Motlhala K, An ISO/IEC 15504 based Software Process Assessment in Small Software Companies, International Journal of Software Engineering and Its Applications Vol.8, No.6 (2014), pp.121-138
- [51] Janssen C, Available online: <http://www.techopedia.com/definition/4234/billing-software>, Billing Software [Accessed December 19, 2014]
- [52] Davoren J, Available online: [http://www.ehow.com/info\\_7934305\\_definition-inventory-management-system.html](http://www.ehow.com/info_7934305_definition-inventory-management-system.html) [Accessed December 19, 2014]
- [53] Heathfield S, Available online: <http://humanresources.about.com/od/glossaryh/a/hris.htm> [Accessed December 19, 2014]
- [54] Janssen C, Available online: <http://www.techopedia.com/definition/23889/web-development> [Accessed December 19, 2014]
- [55] Lifefyre, Available online: <http://www.entrepreneur.com/encyclopedia/point-of-sale-pos-system> [Accessed December 19, 2014]
- [56] Thapliyal M and Pratibha D, Software Process Improvement in Small and Medium Software Organisations of India, India, 2010
- [57] Bernard W and Sazzad H, Software Process Improvement in Bangladesh, Bangladesh

[58] Savant, Available Online:  
[http://www.itweb.co.za/index.php?option=com\\_content&view=article&id=9921](http://www.itweb.co.za/index.php?option=com_content&view=article&id=9921), South Africa Moves towards process improvement, [Accessed September 9, 2015]

[59] Holodnik-Janczura G, Golinska I. Decision Support System for Choosing a Model for a Software Development Life Cycle, 2010

[60] Ensslin L, Scheid L, Lacerda R. Software Process Assessment and Improvement Using Multicriteria Decision Aiding – Constructivist, Brazil, 2012

[61] Sentsho J, Maiketso J. T, Sengwaketse M, Victoria Ndzinge-Anderson V. Performance and Competitiveness of Small and Medium Sized Manufacturing Enterprises Botswana, Botswana, Gaborone; 9 March 2004

[62] Ma. Dolores and C. Tongco, Purposive Sampling as a Tool for Informant Selection, U.S.A, 2007

[63] Judgement Sampling, Available online: <http://research.methodology.net/sampling/Judgement-sampling/>

[64] Fisheries and Aquaculture Department, Guidelines for the Routine Collection of Capture Fishery Data, Available online: <http://www.fao.org/docrep/003/x2465e/x2465e09.htm>, [Accessed May 28, 2015 ]

[65] Dangle K, Laarsen P, Shaw M. Software Process Improvement in Small Organizations:A Case Study, pp 68 - 75

[66] Small, Medium And Large Enterprises Database, Sector Mapping And Validation In Botswana: “Findings And Strategies” For Local Enterprise Authority In The Republic Of Botswana 2007, Available online:  
<http://www.lea.co.bw/uploads/files/Packaged%20BIDPA%20Study.pdf?size=105115> [Accessed May 2015].

[67] A Trend Towards Lower Software Maintenance Budget, Available online:  
<http://www.methodsandtools.com/dynpoll/oldpoll.php?Maintenance> [Accessed May 2015]

- [68] Mahmood N, Babar M, Motivators of Software Process Improvement: An Analysis of Vietnamese Practitioners' Views, Ireland.
- [69] Muller, S. Dueholm, An Examination Of Culture Profiles In A Software Organization Implementing SPI, Fuglesangs.
- [70] A. Nemeth, V. Borisov, Agile Modelling in Software Engineering, May 22 2006.
- [71] M. Carnegie, Software Engineering Institute
- [72] Viju G, Elsalam M, Ibrahim K, Jassim M. The impact of Software Process Improvement in Small and Medium Scale Enterprises, 2013
- [73] RICHARDSON I, RYAN K. Software Process Improvements in a Very Small Company, University of Limerick, Ireland, 2001
- [74] J. Calvo-Manzano, G. Agustin, et al, A Software Process Improvement Solution for Small and Medium-Size Enterprises...
- [75] Helen Gillespie, Available online : <http://pubs.acs.org/subscribe/archive/tcaw/10/i08/html/08regs.html>, Tracking ISO 9000; The newly revised standards focus on business processes. [Accessed May 2015]
- [76] M. E. Fayad, M. Laitinen and R. P. Ward, "Software Engineering in the Small", Communications of the ACM, vol. 43, (2000), pp. 115-118.

## **APPENDIX A: QUESTIONNAIRE A**

### **QUESTIONNAIRE A**

#### **(Private IT Companies)**

As a master's student at the University Botswana, I am carrying out a research on the use of software processes by software companies in Botswana. Please assist by completing the following questionnaires. Please note that the information you provide will be kept confidential, and used for research purposes only.

#### **Part A Company profile**

1. How old is your company?

- 1 – 5 years
- 6 – 20 years
- 21 – 50 years
- 51 or more

2. How many employees do you have in your company?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100
- 101 or more

3. How many full time IT professionals do you have?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100
- 101 or more

4. How many part time IT professionals do you have?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100



101 or more

5. How many of the IT professionals (either full time or part time) are currently focusing on software development?

1 – 10

11 – 25

26 – 50

51 – 100

101 or more

6. Do you have IT professionals with multiple skills?

Yes

No

7. Please indicate the number of staff you have in the following area?

Graduates in non-IT subjects

Masters in non-IT subjects

Computer Science/Engineering Graduates

Masters in computer science / Engineering

Diploma/ Certificates in IT courses

Other  .....

.....

8. Do you have Botswana citizens who are professional system analyst?

Yes

No

9. Do you have Botswana citizens who are professional programmers?

Yes

No

- How many are they? (Please be specific) -----

10. Do you have Botswana citizens who are professional designers?

Yes

No

- How many are they? (Please be specific).....

11. Do you have Botswana citizens who are professional developers?

- Yes
- No
- How many are they? (Please be specific).....

12. Do you have Botswana citizens who are project managers?

- Yes
- No
- How many are they ? (Please be specific).....

13. Is your company 100% citizen owned company?

- Yes
- No

14. Is your company a local or a multinational?

- Local
- Multinational

### **Part B Company Services and Specialty areas**

1. Please choose the area of focus your company is in. (You can choose more than one area of focus)

- Medical application
- Accounting and financial management
- Inventory management
- Human Resource software
- Website/Web Application Development
- ERP (Enterprise Resource Planning)
- Software Implementation and Integration
- Billing
- Asset management
- POS (Point of Sales)
- E- Commerce
- Data Entry/Data Conversion
- CRM (Customer Relationship Management)

- E-Governance Application
- SCM (Supply Chain Management)
- Data Warehouse
- Access Control
- Mobile/Wireless Application development
- E-learning
- Data Security
- Gaming software
- Artificial intelligence
- Helpdesk
- Service desk
- Other, please specify.....
- .....

2. Which services do your company provides?

- Hardware services
- Software services
- Both Hardware and software services

3. How long has your company been in this business?

- 0 – 5 years
- 6 – 10 years
- 11 – 20 years
- 21 and more years

**PART C. Company Certification and process improvement**

4. Which certification does your company have?

- ISO 9001
- ISO 9002
- ISO 9003
- CMM/CMMI
- None

5. Is your company registered with any of the following Software Process Improvement (SPI)? Tick the relevant?

- SPICE
- IDEAL
- CMM
- CMMI

6. Is your company registered with any other international standard apart from the one mentioned above? (In 4 and 5), Please indicate

.....

7. When did you acquire the certification?

- 1 year ago
- 2 – 5 years ago
- 6 – 10 years ago
- More than ten years ago

8. Please indicate why your company decided to register with Software Process Improvement standards

- Internal compliance (Management or business)
- External compliance
- To improve the quality of service to our customers
- To reduce cost
- Don't know
- Other

9. If your answer in (4) is *none*, have you ever been certified with any international standard?

- Yes, (*please state it*).....
- Never

10. Why is your company not registered?

- Financial constraints
- Our company is still new and its small
- No idea about the standards
- As a company we do not need them
- Other.....

.....

11. Does your company have any plan to get certified?

- Yes
- No

12. Do you have any budget for the international standard certification?

- Yes
- No

**Software development and Software quality control**

1. Does your company develop software to sell commercially?

- Yes
- No

2. Is your company industrial based?

- Yes
- No

3. Does your company develop software for in-house use?

- Yes
- No

4. Who are your customers?

- Government's departments
- Business enterprises
- Individuals
- Educational institutions
- Other please specify .....
- .....

5. How many software systems/application have you developed since January 2012?

- 0 - 1
- 2 - 5
- 6 - 10
- 11 - 20
- 21 and more

6. How many Information systems are you currently maintaining?

- 0 - 1
- 2 - 5
- 6 - 10
- 11 - 20

21 and more

7. Which programming languages do you use in developing your applications?

Java

C

C++

C sharp

Other.....

8. Do have a software tools u use for the following stages in software development.

Documentation

Requirements gathering

Code generation

Testing

Project management

9. Which of the following process model do you use?

Waterfall

Spiral model

V model

Rapid Application Development

Agile Software Development

Structured programming

Dynamic System analysis

Evolutionary development

Reuse based development

Structured programming

Other please specify

10. How useful are these models to you?

Useful

Very useful

Not useful

Useless

Not sure

11. What are the complaints that you usually get from your customers (feel free to choose more than one)

- Product been too late
- Maintenance contract not clear
- Maintenance contract unfavorable
- Product too expensive as compared to its worth
- Product has many bugs
- Too much change of deadlines
- Deployment process expensive
- Product not meeting requirements
- Product been too complicated for users
- Tuning process taking too long
- User manual complicated and not clear
- Other, please

specify.....  
.....

12. How do you deal with the above customer complaints?

- Training
- Frequent visits after deployment
- We follow customer satisfaction guidelines
- Ignore until they understand

- Other, please  
specify.....  
.....

13. How do you carry out the following processes during system development?

a) Requirements

- Interviews
- Prototyping
- Scenarios
- Brainstorming
- Requirements reuse

b) Design

- Flow charts
- Use cases
- Pseudo-code
- Object oriented design

c) Implementation

- Development tool
- Programming

d) Testing

- Unit testing
- Module testing
- Sub-system testing
- System testing
- Acceptance testing

e) Maintenance

- Repair software faults
- Adapting software to a different operating system
- Adding or modifying systems functionality
- No maintenance done for customers

14. How do you measure quality of your systems in your organization?



- Lines of codes
- Fog index (a measure of the readability of a passage of written text)
- Number of reported faults in software
- Number of person-days required to develop system component
- The travel cost to meet the customer
- Computer resources
- We never consider it
- Other, please  
specify.....

15. How do you charge your customers?

- Cost estimation techniques
- Algorithm cost modeling
- Comparing with other companies
- Time needed to develop the software
- Customers charge themselves
- Rely on the previous costs for other similar projects
- Estimate
- Other, please  
specify.....

16. What contribute to your failure to meet deadlines as per your plans?

- Customer feedback delays
- Lack of resources
- Staff resignation
- User's unavailability during meetings and consultations
- Understanding of user requirements
- We rarely miss our deadlines
- We never miss our deadlines
-

Other, please

specify.....  
.....

17. Which of the following software quality management activities do you use to ensure the quality of your software?

- Quality assurance
- Quality planning
- Quality control
- Never check quality of the software
- Customer's views
- Other, please

specify.....  
.....

18. Do you have any quality manual for your software development?

- Yes
- No

19. Does your company have any quality policy for software development?

- Yes
- No

20. Do your projects follow a written organizational policy for implementing?

- Yes
- No

21. Do you have any documented procedure for process improvement?

- Yes
- No

22. Do you have any test plan for the testing of the software which is being developed?

- Yes
-

No

23. Do you have any written guideline for software design as a company?

Yes

No

24. Do you have any written procedure/guidelines for code generation?

Yes

No

**PART D SOFTWARE COST ESTIMATION TECHNIQUES**

25. Which of the following methods of estimation do you use?

Parametric estimation

Wideband Delphi

Cocomo

SLIM (Software Life-Cycle Model)

Function Point Analysis

Proxy Based Estimation (PROBE)

Program Evaluation and Review Technique

Expert judgment

Other, please specify

.....  
.....  
.....

26. Why have you chosen the estimation model you selected in (25)?

.....  
.....  
.....

27. Which estimation model (s) is most convenient for you?

.....  
.....

28. How often do you use estimation techniques?

- in every project
- it depend on the size of the project
- All members of staff
- Other, please specify.....

.....

29. Please explain more on how you carry out your software cost estimation.

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

THANK YOU SO MUCH FOR TAKING YOUR TIME TO FILL  
THIS QUESTIONNAIRE.

APPENDIX B: QUESTIONNAIRE B

**QUESTIONNAIRE B**

**(For government departments)**

As a master's student at the University Botswana, I am carrying out a research on the use of software process and models as well as adoption of international standards among other thing by companies in Botswana. Please assist by answering the following questions. Please note that the information you provide will be kept confidential.

1. Do you have a information system managing your day to day business in your organization?  
 Yes  
 No
2. Do you have some companies that usually supply/develop information system for you?  
 Yes  
 No
3. Which of the following companies have developed a software system for you?  
 ICL Botswana  
 DCM  
 Dimensions Data  
 Hi Performance systems  
 Informatix Industry Services (Pty) Ltd  
 Media Tech and Office Supplies -  
 ZBL Investments (Pty) Ltd  
 Com-Link (Pty) Ltd  
 Cumulus Technologies
4. List IT companies (Not listed in 3) which have developed an information system for you? Put a circle around 'F' to indicate if the company is foreign and 'L' for local.  
i) .....F L  
ii) .....F L  
iii) .....F L  
iv) .....F L

v) .....F L

vi) .....F L

5. Do you buy ready made software ? Y/N

6. Who are your major vendors? ..... ..

7. Do you develop for other departments? Y/N

Yes

No

8. Do you customize software products?

Yes

No

9. For how long have you been using your system?

0 – 1 year

1 – 5 years

6 – 10 years

11 – 20 years

More than 20

10. When last did you purchase a new software product system?

A year ago

2 years ago

5 years ago

More than 5 years ago

11. Do you have software developers who are currently doing software development?

Yes

No

12. How many software developers do you have?

1 – 5

6 – 10

11 – 20

21 or more

13. \*\*How do you ensure that the information systems/websites are of good quality?

- Follow DIT web hosting standards
- Follow international standards
- Other, please specify

.....  
.....  
.....  
.....

14. Who is maintaining your system?

- local company
- It officers
- Developers of the system
- Foreigners

15. List IT companies which are currently supporting or maintaining your software systems. Put a circle around 'F' to indicate if the company is foreign and 'L' for local.

- vii) .....F L
- viii) .....F L
- ix) .....F L
- x) .....F L
- xi) .....F L
- xii) .....F L

16. Give a rough estimate of what you spend on computer software annually.

.....

17. Give a rough estimate of what you spend on computer hardware annually.

.....

18. How much do you spend on your system maintenance annually?

.....

**THANK YOU SO MUCH FOR TAKING YOUR TIME TO FILL THIS QUESTIONNAIRE.**

**APPENDIX C**

**QUESTIONNAIRE C**

**(All Organization)**

I am a master’s student at the University Botswana. I am carrying out a research on the use of software processes by software companies in Botswana and software application by industries in Botswana. Please assist by completing the following questionnaires. Please note that the information you provide will be kept confidential, and used for research purposes only.

**SECTION A – ORGANISATION BACKGROUND**

1. Please provide name of your organization.

.....

2. Please choose category that best describe your organization:

- Government department
- Software Company
- Educational institution
- Banking Corporation

3. How long has your company been in this business?

- 0 – 5 years
- 6 – 10 years
- 11 – 20 years
- 21 and more years

4. Does your organization develop use software products?

- Develop
- Use
- Both

5. How many employees do you have in your company?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100
- 101 or more



6. How many full time IT professionals do you have?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100
- 101 or more

7. How many part time IT professionals do you have?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100
- 101 or more

8. How many of the IT professionals (either full time or part time) are currently focusing on software development?

- 1 – 10
- 11 – 25
- 26 – 50
- 51 – 100
- 101 or more

SECTION B – SOFTWARE QUALITY ASSURANCE (SQA)

B I (For software companies)

1. Do you use software Quality Assurance Plan? Yes  No
2. Are the software work products produced according to the project’s defined software process? Yes  No
3. Are SQA activities planned .....  Yes  No  Does not Apply  Don't Know
4. Does SQA provide objective verification that software Products and activities adhere to applicable standards, procedures, and requirements?.....  Yes  No  Does not Apply  Don't Know
5. Are the results of SQA reviews and audits provided to affected groups and individuals (e.g., those who performed the work and those who are responsible for the work)? .....  Yes  No  Does not Apply  Don't Know
6. Are issues of noncompliance that are not resolved within the software project addressed by senior management (e.g., deviations from applicable standards)?.....  Yes  No  Does not Apply  Don't Know
7. Does the project follow a written organizational policy for implementing SQA?.....  Yes  No  Does not Apply  Don't Know
8. Are adequate resources provided for performing SQA activities (e.g., funding and a designated manager who will receive and act on software noncompliance items)?.....  Yes  No  Does not Apply  Don't Know
9. Are measurements used to determine the cost and schedule status of the activities performed for SQA (e.g., work completed, effort and funds expected expended compared to the plan)?.....  Yes  No  Does not Apply  Don't Know
10. Are activities for SQA reviewed with senior management on a periodic basis?.....  Yes  No  Does not Apply  Don't Know



- POS (Point of Sales)
- E- Commerce
- Data Entry/Data Conversion
- CRM (Customer Relationship Management)
- E-Governance Application
- SCM (Supply Chain Management)
- Data Warehouse
- Access Control
- Mobile/Wireless Application development
- E-learning
- Data Security
- Gaming software
- Artificial intelligence
- Helpdesk
- Service desk

Other, Please

specify.....  
 .....

2. i) For what purpose are you using these software(s) identified in 1 above?

Software Application A (Application name ).....  
 Use:.....  
 .....

Software Application B(Application name ).....  
 Use:.....  
 .....

Software Application C(Application name ).....  
 Use:.....  
 .....

Software Application D(Application name ).....  
 Use:.....  
 .....

.....  
ii) Does the software application meet your needs?

Software Application A Yes  No

Why .....

.....  
Software Application B Yes  No

Why .....

.....  
Software Application C Yes  No

Why.....

.....  
Software Application D Yes  No

Why.....

.....  
3. Are the software in use developed within your organization or were they purchased?

- Purchased
- Developed
- Some are purchased and some are not.

4. How often has your system/software failed?

- Weekly
- Monthly
- Once in six months
- Once in a year
- Never failed

5. What are the causes of failure of your software/system?

- Electricity failure
- Hardware Failure
- Software malfunction
- Natural disaster
- User Faults

6. Has your system/software helped ease the work? Yes  No

Are there some tasks that are still done manually? Yes  No

7. What is your approximate annual budget on software in your organization?

- BWP 50, 00000 to BWP 500, 000.00
- BWP 500, 000.00 to BWP 1000, 000.00

- Above BWP 1000, 000.00
- Below BWP 50, 000.00
- No budget or No idea

8. Is there any computerized system in use in your department? Yes  No

.....  
 .....

9. Which aspect of your departmental process is computerized.

Payroll

Human resource/personnel

Order purchasing

Inventory

Others (please list)

10. Do you have software maintenance procedures ? Yes No

11. How do you maintain the software

(i) Engage software Vendors Yes No

(ii) In house maintenance Yes No

(iii) Use software tools Yes No

(iv) Others (please specify)

.....  
 .....

THANK YOU FOR TAKEN TIME TO RESPOND TO THIS QUESTIONNAIRE

## APPENDIX D

### QUESTIONNAIRE ON THE USE OF CMM/CMMI IN SELECTED SOFTWARE ORGANISATION IN BOTSWANA

#### Introduction

Your organization was selected among those that indicated using the Capability Maturity Model (CMM) or the Capability Maturity Model Integration (CMMI)

Kindly rate your level of CMM/CMMI usage according the following key performance areas: (**Note: CMM/CMMI rate organizations as to how well they succeed at meeting the key performance areas**)

1.

a) Which of the following Key Performance Areas (KPAs) do you do?

- Requirements Management
- Software Project Planning
- Software Project Tracking and Oversight
- Software Subcontract Management
- Software Quality Management
- Software Configuration Management

b) Which of the following additional Key Performance Areas (KPAs) do you have?

- Organization Process Focus
- Organization Process Definition
- Training Program
- Integrated Software Management
- Software Product Engineering
- Intergroup Coordination
- Peer reviews

c) Which of the following additional Key Performance Areas (KPAs) do you practice?

- Quantitative Process Management
- Software Quality Management

d) Do you have Defect Prevention as Key Performance Area (KPA)?

- Yes
- No

2. Why did your company specifically choose CMM/CMMI ? **(Please tick all possible reasons)**

- The model is cheaper in terms of the resources needed
- Model is easy to follow
- Many company use it
- The model has easy procedures to follow
- Don't know
- Other specify .....

3. As a developer, which of the following do you consider when choosing SPI model?

- Size of our company     Size of development team     Type of project we carry out
- Availability of funds     The processes to be improved

4. How can you rate the success/friendliness of CMM/CMMI tool?

- Excellent     Good     Fair     Poor     very poor

5. For how long have you been using CMM/CMMI ?

- 1 year ago     2 – 5 years ago     6 – 10 years ago     More than ten years ago

6. Do you experience any challenge with choosing a Software Process Improvement (SPI) model?

Explain.....  
.....

7. How experienced are your staff in the following SPI models?

a) SPICE (Software Process Improvement and Capability Determination)

- High     Moderate     Low

b) IDEAL (Initiating, Diagnosing, Establishing, Acting and Learning)

- High     Moderate     Low

c) OWPL (Observatoire Wallon des Pratiques Logicielles)



High  Moderate  Low

d) iFLAP(Improvement Framework Utilising Light Weight Assessment and Improvement Planning)

High  Moderate  Low

e) SPM (Software Process Metrix)

High  Moderate  Low

f) ASPE-MSC(AN APPROACH FOR SOFTWARE PROCESS ESTABLISHMENT IN MICRO AND SMALL COMPANIES)

High  Moderate  Low

g) PRISMS (An Approach to Software process Improvement for Small to Medium Enterprises)

High  Moderate  Low

h) TickIT SHEME

High  Moderate  Low

i) BOOTSTRAP

High  Moderate  Low

j) CMM

High  Moderate  Low

k) CMMI

High  Moderate  Low