e-Informatica Software Engineering Journal, Volume 12, Issue 1, 2018, pages: 133–165, DOI 10.5277/e-Inf180106

A Systematic Mapping Study on Software Measurement Programs in SMEs

Touseef Tahir*, Ghulam Rasool*, Muhammad Noman*
*COMSATS Institute of IT, Department of Computer Science, Lahore, Pakistan.
touseeftahir@ciitlahore.edu.pk, grasool@ciitlahore.edu.pk, nomanyasir29@gmail.com

Abstract

Context: Software measurement programs are essential to understand, evaluate, improve and predict the software processes, products and resources. However, the successful implementation of software measurement programs (MPs) in small and medium enterprises (SMEs) is challenging.
Objective: To perform a detailed analysis of studies on MPs for highlighting the existing measurement models, tools, metrics selection methods and challenges for implementing MPs in SMEs.
Methods: A Systematic Mapping Study (SMS) is conducted.
Results: In total, 35 primary studies are comprehensively analysed. We identified 29 software measurement models and 4 tools specifically designed for MPs in SMEs. The majority of the measurement models (51%) are built upon software process improvement approaches. With respect to the measurement purposes of models, the distribution of MPs was identified as: characterization (63%), evaluation (83%), improvement (93%) and prediction (16%). The majority of primary studies discussed the use of measurement experts and experience (60%) followed by the use of measurement standards (40%) and the use of automated tools (22%) for metrics selection in MPs. It was found that the SMEs and large organizations face different challenges which was shown in studies on challenges reported in SMEs reports . The challenges existed even before the implementation of MPs and were connected with infrastructure and management processes in SMEs. The challenges reported by studies in large organizations are mostly related to the issues discovered while implementing MPs.
Conclusion: The analysis of measurement models, tools, metrics selection methods and challenges of implementing MPs should help SMEs to make a feasibility study before implementing a MP.

Keywords: software measurement process, software measurement program, small and medium enterprise (SME), software metrics, software measures, systematic mapping study, GQM

1. Introduction

The number of Small and Medium Enterprises (SMEs) in the software industry is rising quickly and contributing significantly to the Gross Domestic Product (GDP) [1]. The definition of SMEs varies from country to country. According to the European Union [2], "SMEs are those companies which employ fewer than 250 employees and which have an annual income not more than 50 million euro, and/or an annual balance sheet total not more than 43 million euro" [3]. The firms which employ fewer than 50 employees are known as small enterprises and the firms which employ a maximum of ten or in some situations five workers are known as micro-enterprises [3]. SMEs play a very important role in supporting the economy and growth of any country [4].

The software development organizations, just like any other organization, aim to deliver products and services with expected quality by effectively using resources within software development processes. Software measurement is essential to characterize, evaluate, predict and improve software products, processes and resources. Every software development process either generates or uses measurement data. The software measurement domain presents various measurement models, tools and practices to collect and analyse measurement data to estimate, monitor, control and improve software processes, products and resources. Software development organizations implement measurement programs (MPs) as part of software measurement process [5].

It is discussed in a recent SLR [6] that most of the MPs in large organizations fail to achieve measurement objectives and usually they do not sustain more than two years due to multiple reasons [6]. The rate of failure in the successful implementation of MPs is particularly exceptional in the perspective of SMEs [7, 8]. The MPs at SMEs become challenging because they usually do not have enough time, budget and resources to implement measurement plans. Software measurement knowledge is particularly poor in SMEs [7, 8]. The use of software measurement is limited in SMEs due to the lack of metric selection methods [9], a different set of metrics used in different SMEs [10], the lack of infrastructural facilities, low measurement maturity level, small development teams, higher workload [11] and limited measurement planning [12, 13].

A comprehensive Systematic Literature Review was conducted on software MPs and it was observed that there were fewer than 10 percent primary studies on implementing MPs in SMEs [6]. Therefore, this study presents a Systematic Mapping Study (SMS), which specifically focuses on measurement models, tools, metrics selection methods, and challenges of implementation software MPs at SMEs. Later, the measurement models, tools, metrics selection methods, and challenges in the implementation of MPs in SMEs and large organizations [6] are also compared. The measurement studies are analysed by answering the following research questions (RQs). There is no such study published with research questions (presented below) to the best of our knowledge.

RQ1: What measurement models, tools and practices for implementing measurement programs in SMEs are discussed in literature?

RQ2: What are the problems, challenges and issues of implementing measurement programs in SMEs?

RQ3: What metrics selection techniques, methods and approaches are used for measurement programs in SMEs?

This paper is organized as follows: Section 2 presents related work, Section 3 presents Systematic Mapping Process, Section 4 presents results and analysis and Section 5 presents conclusions.

2. Related work

Kitchenham [14] conducted a mapping study to investigate the status of software measurement research between 2000 and 2005. She identified that software MPs were the most researched area of the software measurement domain [14]. The journal papers were found to be more influential in measurement community than conference papers based on the numbers of citations. The study concluded that the there is a need for comparative studies and to serve this purpose empirical datasets should be made public. The datasets used among the primary studies were categorized as public (31%), private (61%), partial (8%) and unknown (1%). The primary studies lack the discussion on lightweight measurement methods for SMEs.

Gómez et al. [15] conducted an SLR to answer fundamental questions of what, how and when to measure. They analysed 78 primary studies. The measurement aspects discussed among the primary studies were categorized as project, process and product. They established that most of the primary studies discussed product metrics (79%) followed by project (12%) and process (9%) metrics. The software complexity and its size were identified as the most frequently measured attributes. The software metrics were mapped to typical initial, intermediate and final phases of a software project life cycle. Most of the metrics were found to be utilized for the initial phase (48%), followed by the intermediate (36%) and final (16%) phase. They concluded that software metrics need theoretical and empirical validation before being used in a measurement process. The discussion and primary studies on lightweight measurement methods and measurements used in SMEs are missing in the SLR.

The software measurement process has a key objective of predicting the use of measurement data and software defects as they are one of the most predicted attributes [6, 14]. Catal et al. [16] conducted an SLR to analyse the software defects prediction studies. They analysed 75 primary studies published between 1990 and 2009 and classified the primary studies according to methods used for fault prediction, i.e. machine learning methods/algorithms, statistical and machine learning methods and expert judgment. The machine learning and statistics are found to be the most widely used methods for software measurement. Furthermore, fault prediction metrics were classified with respect to method, class, component, file, process and quantitative-values levels. They found out that 60% of studies used method-level metrics and 24% of studies applied class-level metrics and only 4% of studies have used process-level metrics.

Malhotra [17] conducted an SLR on software defect prediction studies published between 1991 and 2013. They found that most of the studies use size, effort and object oriented metrics for prediction. Radjenović et al. [18] conducted an SLR on fault prediction studies which were published between 1991 and 2011. They identified that object-oriented metrics (49%), traditional source code metrics (27%) and process metrics (24%) are mostly used in fault prediction studies. They found out that defect prediction studies mostly used one type of metrics, e.g. method-level, class-level, process-level, or source code metrics or object-oriented metrics. Hall et al. [19] conducted an SLR to analyse 208 fault prediction studies that were published between 2000 and 2010. They established that studies which used a combined approach (where more than one type of metrics were used) performed better than the studies which used a single type of metrics. They found that the machine learning methods were mostly discussed. These methods focused on utilizing large amounts of data. They observed that the machine learning methods outperform the statistical methods because they overcome the shortcomings of traditional statistical processes. The discussion on lightweight prediction methods, which consider the minimal budget, time and resources of SMEs, are currently missing from the discussed SLRs.

Unterkalmsteiner et al. [20] conducted an SLR to analyse measurements and evaluation strategies, which are used to assess the software process improvement (SPI) initiatives. They analysed 148 primary studies that were published between 1991 and 2008. The studies were classified with respect to their measurement focus, process quality, and prediction/estimation accuracy and software measurements (such as size, effort and customer satisfaction). The SPI models are discussed and the capability maturity model (CMMI) is identified as the most studied model in the SPI domain. The primary studies mainly focused on the measurement of quality (39%), prediction accuracy (38%) and productivity (35%). Three levels of measurements are explored, i.e. product, project and organization. The measurement of SPI initiatives is mostly done at project and project-product level. The problems in SPI studies are discussed, e.g. more than half of the studies do not completely describe the SPI context (organizational size, measurements validity and scope of SPI activities, etc.). They considered that the lack of context description might hinder the reuse of learned lessons and results in similar settings. This study is different from this research in two ways; 1) it does not discuss the role of MP for SPI and 2) it mainly focuses on SPI for large companies as there is no discussion and paper found on SMEs.

Touseef et al. [6] conducted an SLR on software MPs by analysing 65 primary studies that were published between 1997 and 2014. They analysed 35 measurement planning models, 11 associated tools, and metrics selection methods, and success/failure factors for implementing MPs. Most of the models and tools extended goal-based measurement approaches. The measurement studies are categorized with respect to measurement purposes, i.e. characterization (81%), evaluation (77%), prediction (28%) and improvement (70%). The measurement planning models and tools are categorized based on mea-

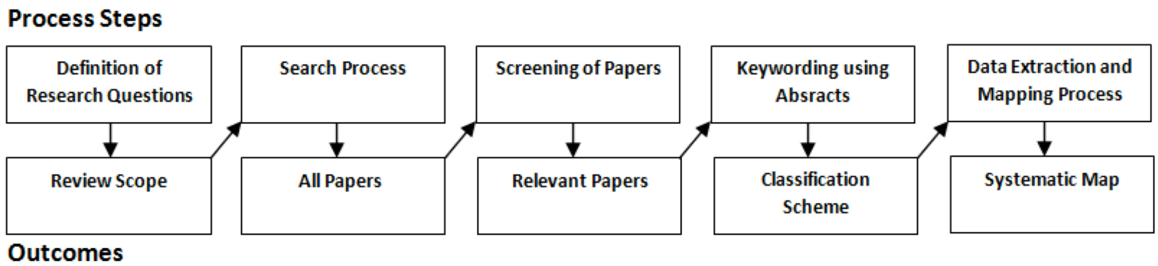


Figure 1: The systematic mapping study (SMS) process [21]

surement entities, i.e. processes (96%), products (58%) and resources (40%). The success factors for implementing MPs include organizational adoption of an MP, and Integration of an MP with SDLC, the synchronization of an MP with an SPI. Most of the measurements planning models were evaluated in case studies. They found that there are few measurement studies with the comparisons and reusability of results and the learned lessons of implementing MPs. The lack of context description (e.g. organizational size, measurement scope, and measurement analysis methods) hinders the reusability and comparative analysis of results among primary studies. The metrics datasets used in MPs are not explicitly presented in the measurement studies. In this study, only 3% of the studies discuss measurement planning models and tools for SMEs. Therefore, this SMS was conducted to specifically analyse measurement models, tools, and metrics selection methods that are proposed for SMEs while considering specific challenges in the implementation of MPs in SMEs.

Sulayman et al. [22] conducted an SLR on software process improvement (SPI) in small and medium web companies. The aim of the study was to specifically identify SPI models and techniques for small and medium web companies. They analysed only 4 primary studies after applying inclusion/exclusion criteria based on research questions. They found the limitations of SMEs, such as tight budget, ambitious deadlines and short-term strategy. The success factors include an increase in productivity, compliance with standards and overall operational efficiency. Pino et al. [23] conducted an SLR to analyse SPI approaches in SMEs by analysing 45 primary studies published between 1996 and 2006. They found CMM (38%) as the most discussed SPI standards in primary studies. They found that other standards, such as ASSESSMENT SEI (16%), IDEAL (13%), CMMI (9%), SPICE (13%), ISO/IEC 12207 (11%), GQM (2%) and PSM (2%), are not frequently used in SMEs. They also established that SPI is mostly measured in terms of employee perception instead of a formal measurement process. They claimed that the most frequently used SPI model for SMEs is CMM used as a reference model, ISO 15504 as a process assessment model and the IDEAL model for guiding improvement. It was also established that SMEs found it hard to implement SEI and ISO models. The RQs answered in these studies ([22, 23]) do not discuss the role of MPs for SPI, but rather the role of measurement for SPI. This study provides an analysis of the implementation of MPs in SMEs with respect to characterization, evaluation, improvement and prediction.

3. Systematic mapping process

This section presents the planning of Systematic Mapping Study (SMS) to analyse the existing literature regarding MPs at SMEs [21]. The overall steps of an SMS process are presented in Figure 1. The goal of this SMS is to systematically recognize, explore, and classify the studies on software MPs at SMEs and present the mapping of these MPs to highlight their possible challenges and the future scope of study [24]. The SMS was performed following the guidelines in [25] and implemented the systematic mapping process proposed by Petersen et al. [21]. Each step of the SMS process has an outcome and the overall outcome of the process is a systematic map.

Table 1: Research questions of systematic mapping study

ID	Research Question	Motivation
RQ1	What measurement models, tools and practices for implementing measurement programs in SMEs are discussed in literature?	To understand the reported measurement models, tools and practices developed in SMEs to implement software measurement programs.
RQ2	What are the problems, challenges and issues of implementing measurement programs in SMEs?	To understand problems, limitations and challenges faced by SMEs during the implementation of measurement programs.
RQ3	What metrics selection techniques, methods and approaches are used for measurement programs in SMEs?	To highlight the metric selection methods used in different SMEs for implementing their measurement programs.

Table 2: Search string

Population	Intervention
(software) AND (“measurement program” OR “measurement process”) AND “small and medium enterprise” OR SME)	(metric∗ OR measur∗ OR model OR framework OR tool OR challeng∗ OR problem OR issue OR improv∗ OR goal)

3.1. Definition of research questions

The main objective of this mapping study is to determine how software MPs are implemented in small and medium enterprises (SMEs). To answer this question, three research questions (RQ) were defined, as presented in Table 1.

3.2. Search process

A search string is used to select a potentially relevant set of primary studies. The lack of consistency for measurement concepts and terminology is a major threat to finding the relevant studies [26]. Therefore, initially the main concepts and terminology in the software measurement domain were reviewed and then the keywords considering the RQs were identified. Then, the synonyms and alternatives for each keyword were checked. Finally, “AND” and “OR” operators and wildcard character “∗” were used to create the search string. The “OR” operators were used to combine synonyms. The wildcard character “∗” was used to represents zero, one, or multiple alphanumeric characters in the position it occupies. The “AND” operator was used to combine the search string between population and intervention as shown in Table 2.

Population: In software engineering, population may refer to a particular software engineering role, the category of software engineer, an application area or an industry group [27]. In our perspective, the population is (software) AND (“measurement program” OR “measurement process”) AND (“small and medium enterprise” OR SME). In population, the keyword “Software” is used to search studies related to software engineering only. The keywords “measurement program” and “measurement process” are used to search studies which discuss a measurement program or a measurement processes. The keyword “small and medium enterprise” and SME cover small and medium enterprises.

Intervention: In software engineering, intervention refers to a software methodology, tool, technology, or procedure. In this case the intervention is clear according to the situation of this study, that is (metric∗ OR measur∗ OR model OR framework OR tool OR challeng∗ OR problem OR issue OR improv∗ OR goal). The keywords “metric” and “measur” refer to the metric/metrics and measure/measures/measuring/measurement, respectively. The keyword “improv∗” refers to the variations of improve such as improving/improves/improved. The “challeng” refers to the variations of challenge such as challenges/challenged/challenging.

The primary studies were selected by reviewing the titles, abstracts and conclusions of the search results obtained from different databases. The databases were selected based on the experience reported by [6]. Table 3 presents the number of search results per research database.

3.3. Screening of relevant papers

This step of SMS is completed by applying study inclusion and exclusion criteria.

Study exclusion criteria:

The studies which do not conform with the exclusion criteria were excluded:

- studies which are not reported in the English language;
- studies which are not accessible in full-text;
- books and grey-literature;
- studies conducted in non-software companies.

Study inclusion criteria:

General criteria:

- a study is conducted in SMEs context;
- a study is in the area of software metrics and software measurement programs/ processes;
- a study includes an empirical evaluation (experiment, case study, survey, experience report, and/or action research).

Criteria specific to research questions:

- a study presents discussion/analysis on software measurement models or tools in SMEs (RQ1);
- a study discusses challenges, issues, limitations and problems that are related to software measurements in SMEs (RQ2);
- a study discusses metric selection methods for implementing software measurement programs in SMEs (RQ3).

Figure 2 presents the selection of the final set of primary studies (35) after applying the search process, exclusion/inclusion criteria, and snowball tracking. The snowball tracking reviews the references of every primary study with respect to its relevance to research questions. Endnote, a reference management tool, is used to remove duplicates and to manage the large number of references.

3.4. Keywording

The objective of keywording is to effectively produce a classification schema and ensure that all the selected papers are relevant [21]. Figure 3 shows the systematic process that was followed to create the classification schema.

The initial step comprised reviewing the abstracts of primary studies and then allocating them a number of keywords to recognize the basic contribution topic of the article. After that, all the keywords were consolidated to establish the high-level of classification, and to understand the area of research highlighted in the primary studies. The schema experienced a continuous improvement process by logically fitting the papers into classes for new data. The resulting classification schema is presented below.

The primary studies are classified based on the following schemas:

- **Time of publication:** to map the studies based on the time of publication.
- **Empirical research method:** to map the study according to the research method used.
- **Contribution type:** to map the outcomes of different types of studies.
  - **Models/tools:** to map the models, tools, and measurement methods for building software measurement processes in SMEs.
  - **Challenges:** to map the studies, which discussed challenges, issues, limitations regarding software measurements in SMEs.
  - **Metric selection criteria:** to map the studies which discussed metrics selection methods and most commonly collected metrics in SMEs.

The *time of publication* schema describes the number of primary studies which are related to research questions.

The *Empirical research method* is the classification schema which categorizes the studies based on their research methods as presented in Table 4. The research methods are categorized as a case study, survey, industrial report and experiment.

The *contribution type* schema describes the type of contribution by study. It is classified into models/tools, measurement methods in SMEs, metric selection methods, com-

Table 3: Number of studies retrieved per research database

Research resources used	Number of potential primary studies
Search Engines	
Google Scholar	1960
Wiley Interscience	34
Science Direct Journals	06
Springer	117
One Search (Search Tool)	2372
ACM	50
IEEE Xplore	99
Journal Databases	
ACM Transactions on Software Engineering Methodology (TOSEM)	10
IEEE Transactions on Software Engineering (TSE)	2
IEEE Software	4
Software Quality Journal	3
Journal of Systems and Software	1
Empirical Software Engineering	38
Automated Software Engineering	0
Conference Databases	
IEEE International Software Metrics Symposium (2000-2005)	3
IEEE International Conference on Software Engineering (ICSE)	0
Joint International Conference on Software Process and Product Measurement (Mensura) and Workshop on Software Measurement (IWSM)	5
Empirical Software Engineering and Measurement (ESEM) (2007-2014)	0
Product Focused Software Process Improvement (PROFES)	11
Software Process and Product Measurement	0
Software Engineering and Advanced Applications (SEAA)	3
Pacific Industrial Engineering and Management Systems (APIEMS)	4
European conference on software process improvement (EuroSPI)	5
International Conference on empirical Assessment in Software Engineering (EASE)	0
International Symposium on Empirical Software Engineering and Measurement (ESEM)	0
Information Technology: New Generations (ITNG)	0
International Conference on Emerging Technologies (ICET)	1
Total	**4728**

monly selected metrics and challenges related to the implementation of MPs in SMEs. The Model/tools are further categorized into extended goal question metric (GQM) methodology or software process improvements (SPI) methodology or measurement process improvement.

The metric selection criteria are also categorized into three subclasses; use of standards, use of measurement expert and experience and use of automated tools. These three subclasses were earlier defined based on the analysis of metrics selection methods used in the measurement studies in [6]. The mapping results of the classification schema are analysed in Section 4.

3.5. Data extraction and mapping

A data extraction form was developed in MS Excel (Table 5) to extract data from the primary studies for each RQ using the classification schema.

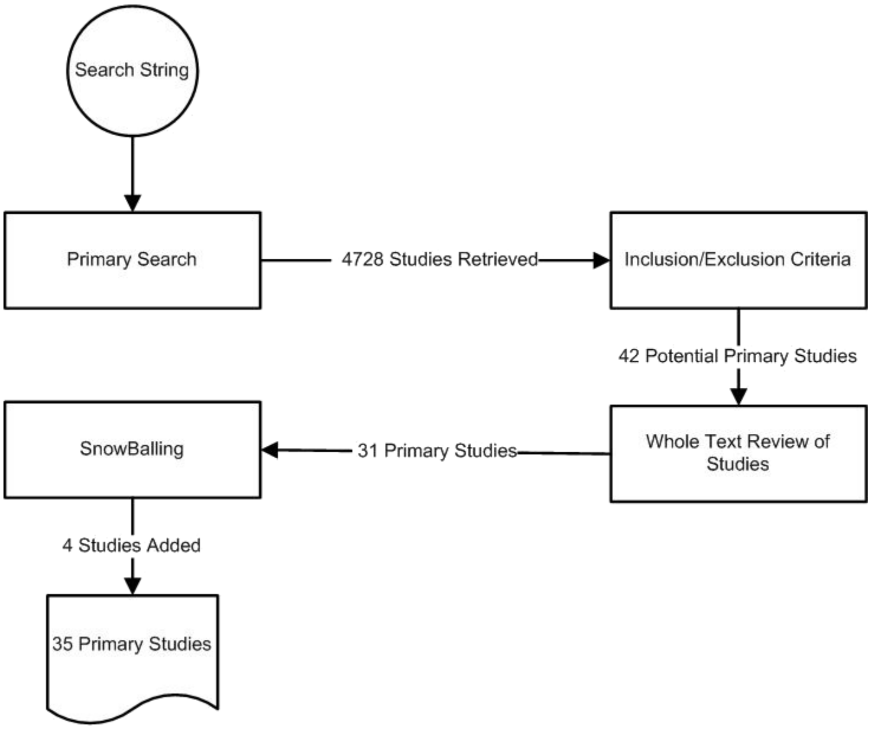


Figure 2: Process of selecting primary studies

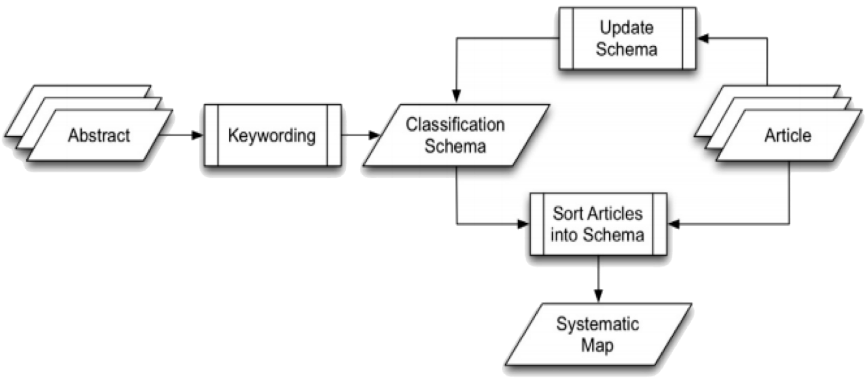


Figure 3: Creating the classification schema [21]

4. Results and analysis

In total, 35 measurement studies are analysed in this section. First a short overview of the studies is presented with respect to publication year and research method. It is followed by the presentation of results and analysis.

Publication year: The results of the systematic mapping study are presented in this section. In total, 35 primary studies are analysed and Figure 4 presents the numbers of primary studies with respect to the year of publication. The numbers of primary studies on implementing MPs in SMEs are smaller as compared to 65 primary studies on implementing MPs in large organizations in our previous study [6]. Therefore, it is important to discuss the history of software measurement domain and how it became critical of SMEs.

Software measurement is a young discipline as the history of software metrics dates back to the late 1960s [31]. It is claimed in [31] that the first book on software measurement [32] was published in 1976 and the first comprehensive report on implementing software MPs was published by Grady and Caswell [33] in 1988. The widely

Table 4: Classification schema of research methods

Purpose	Meta-data
Survey	A research method designed and performed to observe the opinions of people in a structured way [28].
Case study	A research method considered and presented to examine the opinions of people in an unstructured way [28, 29].
Experiment	A research method designed and performed to work with one or more variables and manage all other variables to measure results [30].
Industrial report	A research method used to evaluate the industrial experiences without clear research questions and objectives [30].

Table 5: Data extraction form

Purpose	Meta-data
General information	Study title, authors' names, date of publication and research methodology.
Specific information	Measurement models/tools at SMEs, metric selection methods, commonly selected metrics and challenges/problems/limitations in the implementation of measurement programs in SMEs.

used Goal Question metrics (GQM) model [34] was also introduced in 1988 and the first comprehensive guidebook on goal-oriented measurement was published by Park in 1996 [35]. Software MPs in large organizations have faced many challenges over the last three decades [6, 36].

The evolution of software engineering and software industry includes interdependencies and has impact on the emergence of SMEs. The SMEs started to influence the software development industry following the advancements in microchip technology and communication technologies (e.g. the internet) and the unbundling of software from hardware. According to [37], internet services also affected SMEs based on four factors. The first factor is access to global information sources to enable extension in a business network. The second factor is enabling faster document transfer, online transactions and faster communication channels. The third factor is enabling the search of low cost market, minimizing dependency on a local market (e.g. outsourcing, crowd sourcing and global software engineering). The fourth factor is feedback by international clients and adapting globally successful strategies.

Researchers and practitioners specifically aimed to design software development processes for SMEs during the mid-1990s. There is a plethora of studies published between 1995 and 2000 to promote iterative and incremental software development for the different structure and limitation of SMEs [38]. Basili and Larman claimed in their book ([38]) that the first book on agile software development (e.g. SCRUM, XP) was published by Cockburn [39] in 2002. SMEs represent 99 percent of businesses in Europe[1] with respect to the currently used definition of SMEs that was legislated in 2003. This definition is an updated version of the 1996 definition.

It might be argued based on the above discussion that software engineering research community initially focused on software development processes (e.g. Waterfall, Spiral) and software measurement processes in large companies. Later, the research community focused on software development processes (e.g. Agile, SCRUM) in SMEs when these processes became operational and popular, then they specifically focused on software measurement processes for the characterization, evaluation, prediction and improvement of software development processes in SMEs.

The first study meeting the inclusion criteria was published in 2001. Therefore, this paper presents the search period between 2001 and

[1] http://ec.europa.eu/growth/smes/business-friendly-environment/sme-definition_en

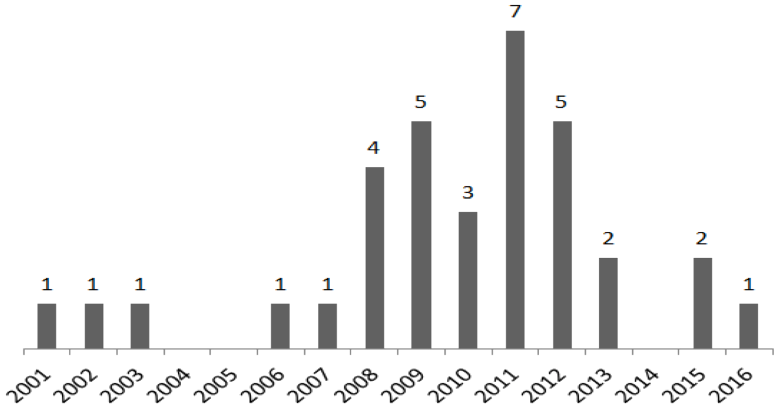


Figure 4: Distribution of primary studies with respect to time of publication

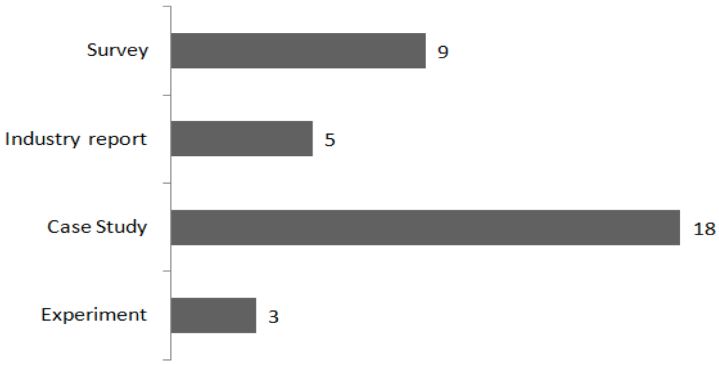


Figure 5: Distribution of primary studies with respect to research methods

2017 in Figure 4. The research databases shown in Table 3.

Research method: The most commonly used research methods in selected studies are case studies (51%) and surveys (25%). Some of the studies used industrial reports (14%) and experiments (8%) as shown in Figure 5.

RQ1: "What measurement models, tools and practices for implementing measurement programs in SMEs are discussed in literature?"

Touseef et al. [6] conducted an SLR on software MPs by analysing 65 primary studies, they studied 35 measurement planning models. In their study [6], they found only 4 specifically defined measurement models for SMEs. They observed that 83% (29 out of 35) measurement models extended the goal-oriented approach or the goal question metric model. The concept behind goal-based approaches is to identify the measurement goals of an organization and then the relevant metrics to achieve measurement goals [34, 35]. In this SMS, 29 software measurement models and 4 tools among 35 primary studies were identified.

Table 6 presents the "Base Measurement Model", of the "Measurement Model" and its "Measurement Purpose" and "Implementation Purpose". The "base measurement model" in Table 6 refers to the parent model of the identified "Measurement model" for SMEs. The "implementation level" refers to the implementation levels of MPs (i.e. project level and/or organization level). The "measurement purpose" represents the basic purpose/objective of MPs discussed in the studies (i.e. to Evaluate (E), Improve (I), Characterize (C) and/or Predict (P) the software process, product or resource entities) [34, 35].

Figure 6 presents the categorization of measurement models. These models are categorized among "goal oriented approach improvement (GOAI)", "software process improvement (SPI)" and "measurement process improvement (MPI)".

The PRISMS model is based on the goal-oriented measurement and SPI. Similarly,

Table 6: Software measurement models for SMEs

ID	Base	Measurement Model	Implementation Level	Measurement Purpose
S16	GQM	Light weight GQM	Organization	CEI
S2	GQIM, CMM	MIS-PyME MCMM	Project	CEIP
S1	GQM, GQIM	MIS-PyME	Project	CEIP
S5	GQM, GQIM	MIS-PyME	Organization	CE
S3	GQIM	MIS-PyME methodology	Project, Organization	CEI
S4	GQIM	MIS-PyME	Organization	CEI
S6	CMMI 1.2	SQIP	Project	EI
S8	GQM, CMM	PRISMS	Project	CEI
S9	CMM	MESOPYME	Project, Organization	I
S10	QFD	SPM	Organization	EI
S11	CMMI	AAHA	Organization	I
S12	TQM	LQIM	Organization	EI
S14	BSC	HSC (Holistic Scorecard)	Organization	EI
S15	No Base Model	Pro Scrum	Project	I
S20	GQM	GQM-DSFMS	Project	CEI
S19	No Base Model	Tarc	Project	C
S21	GQM	Four step framework	Organization	CI
S22	GQM	OMSD	Project	CEI
S23	GQM	SPGQM	Project	CEI
S24	No Base Model	SCAPT	Organization	CEI
S26	QIP, SME	AM-QuICk	Project, Organization	EI
S27	CMMI, PSP, XP, SCRUM	ASPISME	Project, Organization	EI
S28	ISO/IEC 12207:2008, SCRUM	Adapting ISO/IEC 12207:2008 for SCRUM	Project, Organization	CEIP
S29	SWEBOK	Adapting ISO/IEC 15939:2007	Project, Organization	CEIP
S30	ISO/IEC 15504, ISO/IEC 12207:2008 and CMMI	Hybrid Process Model	Project, Organization	CEIP
S31	GQM	GQM Adaption for SPI	Project, Organization	EI
S32	ISO/IEC 12207:2008	COMPETISOFT	Project, Organization	CEP
S33	No Base Model	PMS-IRIS	Project, Organization	EI
S35	CMMI, SCRUM	CMMIbyScrum	Project, Organization	CEI

MIS-PyME, MCMM, and 4-step framework extend goal-oriented measurement and MPI. The AAHA model is proposed to enable SPI and MPI in SMEs. An interesting finding is that the numbers of SMEs are increasing rapidly throughout the world but there are limited numbers of studies that present measurement models/tools for small and medium enterprises as compared to large organizations [6]. For instance, SMEs represent 99 percent of businesses in Europe[2] with respect to the currently used definition of SMEs that was legislated in 2003. SMEs face challenges such as having limited resources, shorter time to market, limited budget, and frequent changes in customer requirements [S1, S2, S3, S4, S5]. Therefore, there is a need for specific models/tools to deal with particular challenges to the establishment of MPs in SMEs. Pino et al. stated in an SLR [23] that ISO and SEI standards for SPI are not directly suitable for SMEs due to the com-

[2] http://ec.europa.eu/growth/smes/business-friendly-environment/sme-definition_en

plexity of recommendations and the requirement of large investment of time and resources. Therefore, there is need for widely accepted strategies to adapt these standards in SMEs [23]. It was proposed to adapt the guidelines and methods used in the measurement models that are already reported for large organizations in this SMS [6, 20]. The MIS-PyME [S2], SQIP [S6], PRISMS [S8], MESOPYME [S9], AAHA [S11], ASPISME [S27], and CMMIbyScrum [S35] models are proposed for the CMMI standard in SMEs. Irrazabal et al. proposed guidelines to adapt ISO/IEC 12207:2008 standard to SCRUM [S28] and María et al. proposed guidelines to adapt ISO/IEC 15939:2007, ISO/IEC 12207:2008 and CMMI in SMEs [S29].

Goal-oriented approach improvement (GOAI): In total, 29 measurement models are identified in this SMS and 40 percent of these models are proposed as the extension of goal-oriented approaches. For example, lightweight GQM process [S16] is an enhancement of the GQM model that is proposed to decrease measurement overhead considering the characteristics and limitations of small software companies. The OMSD [S22] model is proposed to select the optimum number of measures from the available large set of measurements within limited time and effort using meta-measures, such as collection time, cost, priority, value, and usage. The GQM model lacks a method to define measurement goals and questions in a consistent, complete, traceable and verifiable way [6]. Therefore, the SPGQM [S23] model extended the GQM model to define measurement goals and questions in a consistent, complete, verifiable and traceable way. The SPGQM model also used the OMSD model for the optimum number of metrics selection in a case study. GQM-DSFMS [S20] extended the GQM model to select the optimum number of metrics based on time, the cost and usage of metrics and the importance of measurement goal. It also presented a method to enable traceability among measurement goals, questions and metrics. Jezreel et al. [S31] proposed a method for applying the GQM model in SPI by conducting structured interviews of top management and operational management to define measurement goals, and then identify questions and metrics to achieve the goals. Similarly, the PRISMS model [S8] is proposed to relate business goals and improvement goals with measurement goals. Furthermore, the CMM model is used as a reference model to plan and implement MPs in SMEs. The MIS-PyME MCMM model [S2] is proposed to define the SMEs version of the CMM standard for SPI using the goal-oriented approach. The MIS-PyME model and its extensions are proposed with case studies to implement goal-oriented measurement processes and measurement process improvement in SMEs [S1, S3, S5].

Measurement process improvement (MPI): In total, 13 models are developed for improvements in measurement processes in SMEs. For example, the MIS-PyME [S1, S3, S4, S5] framework is presented to define the software MPs in SMEs. This model extended GQM and GIQM [40] to implement and improve the measurement process in small organizations. The MIS-PyME measurement capability maturity model [S2] was developed to support SMEs in defining MPs with respect to measurement maturity of the company and establishing a mechanism for the continuous improvement of MPs.

The LQIM [S12] model is presented based on the Total Quality Management (TQM) paradigm [41] to implement quality improvement plans in SMEs in Pakistan. It is recommended to use it with Deming"s Plan, i.e. Plan, Do, Act, Check (PDAC) for continuous improvement in quality processes. Caballero et al. [S15] present industrial experience related to MPI using agile methodology in SMEs. The study showed that Scrum might improve productivity without decreasing product quality in SMEs. The study [S15] also showed that Scrum is a good alternative for process improvement in an organization with very limited resources. A "four step framework" [S21] was presented to implement MPI in those SMEs which needed improvement in their development processes.

There are four measurement models proposed with the intentions of SPI and MPI simultaneously. AAHA [S11] is a lightweight method developed for SPI in SMEs, it is based on CMMI,

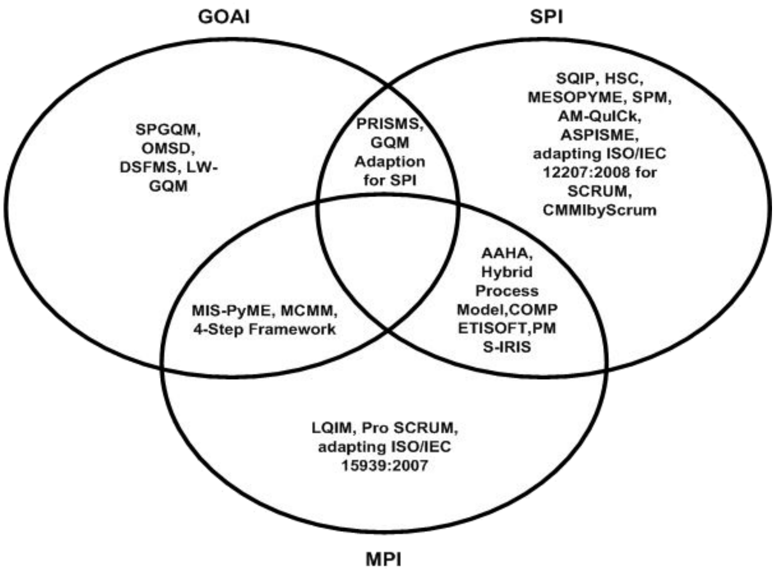


Figure 6: Categorization of measurement models with respect to goal-oriented approach (GOAI), software process improvement (SPI) and measurement process improvement (MPI)

SPICE and agile practices. It is particularly developed to provide a low cost improvement in the software development practices in SMEs. The Hybrid measurement model [S30] is proposed to adapt ISO/IEC 15504, ISO/IEC 12207:2008 and CMMI Dev 1.3 for the maturity of a measurement process and improvement in agile processes in an organization.

The COMPETISOFT model [S32] is based on the experience of using ISO/IEC 15504 and ISO/IEC 12207:2008 in 20 SMEs. It defines four steps of planning SPI, i.e. SPI definition, assessment, measurement and establishment. The improvement of documentation and project management processes is identified as the focus of most SPI initiatives in 20 companies. The PMS-IRS model [S33] proposed 9 steps of performance measurement systems in SMEs, i.e. planning the project, definition of enterprise environment, designing key improvement processes, analysis and design process, definition of measurement process levels, validation of measurements, establishing technological infrastructure, and human resource management. It defines the performance management system as a set of dynamic and integrated metrics for the measurement and evaluation of business operations enabling decision making for SPI.

There are ten key process areas and 3 themes (measurement, quality and tools) of Software Engineering Body of Knowledge (SWEBOK) [42]. Abran et al. proposed extensions in the measurement process of SWEBOK [43]. Maria et al. [S29], further extended Abran's proposal to adapt it for SMEs. They extended the key process areas of measurement by defining new measurement processes for SME, i.e. "process and business assessment", "perform measurement process", "and evaluate measurement" and "experience factor".

Software process improvement (SPI): Software Process Improvement (SPI) is a systematic approach to continuously increase the efficiency and effectiveness of processes in software development companies [20]. The SPI models proposed for establishing MPs in large organizations are not considered suitable for SMEs due to their complex nature and expensive cost [44]. SPI is one of many factors that can affect the success of software development organizations [S14]. There are multiple SPI models identified (e.g. CMMI, CMM, SPICE, PSP, TSP, Six-Sigma, QIP, TQM) in an SLR [20]. The CMM, Six-Sigma, and CMMI models are mostly discussed for implementing measurement processes in large organizations [20]. The ASPISME model [S27] is proposed to adapt CMMI and PSP for improving XP and SCRUM software development processes in SMEs. The ASPISME model contains guidelines for process improvement at three levels, i.e. enabling individuals to understand and practice SPI activities and enabling SPI at the project level and organization level. Similarly, Irrazabal et al. proposed guidelines to adapt ISO/IEC 12207:2008 standard for SCRUM based on experience in 25 SMEs.

On the other hand, there are fewer SPI models available for SMEs and they are not widely used either. For example, the PRISMS model [S8] uses the GQM model for software process improvements. It also relates improvement goals to business goals which help to choose and prioritize key process areas for improvement. The SQIP model [S6] is proposed to improve the quality and reliability of a software development process to achieve the business goals in SMEs. Specific process improvement activities are used in this project, such as requirements and change management. SQIP adopted CMMI version 1.2 as the base model for the implementation and evaluation of software process improvement in SMEs.

The SPM [S10] model is based on QFD (quality function deployment) methodology. It is proposed to define SPI plans and estimate the effect of each SPI practice on a specific software process. The MESOPYME model is proposed to improve the quality and productivity of software development processes using action package concept (i.e. a method to help faster and inexpensive SPI program implementation in SMEs). The HSC model [S14] extended the BSC [45] model to observe business success in software development in SMEs by enabling synchronization between software development processes and business operations.

Ayed et al. [S26] proposed the AM-QuICK model for improvement in agile methodologies with the help of a measurement process. They proposed customization of agile methodologies for continuous SPI at multiple levels, i.e. organizational level, process management level and product management level.

Figure 7 presents the distribution of the measurement purposes of measurement models for implementing MPs (i.e. evaluation, improvement, characterization and prediction). Characterization means that an MP is implemented to collect the data about potential causes of a problem or understand the state of processes, products or resources (e.g. to understand the delays in product delivery, MP implementation can help to collect data about the number of bugs reported, the number of change requests by customer). Evaluation means that an MP is implemented to gauge and analyse the gap between the planned and actual state of processes, products and resources (e.g. to analyse the difference between estimated and actual effort). The prediction means that an MP is implemented to use historical data to make an estimation about software processes, product and resources (e.g. to predict number of bugs in a software product). The improvement means taking actions to improve software processes based on the measurement process. The distribution of software MPs with respect to their measurement purpose are: improvement (86%), evaluation (80%), characterization (60%), and prediction (20%). When a combination of purposes (i.e. when more than one purpose was mentioned by a primary study) was investigated, it was found out that around 59% of the studies mentioned the purposes of characterization and improvement while only 17% listed all four purposes.

Figure 8 presents the distribution of the implementation levels of measurement models for implementing MPs (i.e. project and/or organization level). It was observed that most of the MPs are implemented at the organization level (45%) and the project level (45%) and only 10% of MPs are implemented at both project and organization levels.

RQ2: "What are the problems, challenges and issues of implementing measurement programs in SMEs?"

Table 7 presents the challenges of implementing MPs in SMEs.

Low measurement maturity: The implementation of software measurements processes in SMEs is limited due to low measurement maturity [S2]. It is stated in [S1, S2, S3, S4, S8, S25, S29, S30, S31, S32, S33, S35] that measurement processes are either not defined at all or poorly defined in SMEs, which hinders defining measurement indicators and measurement goals in

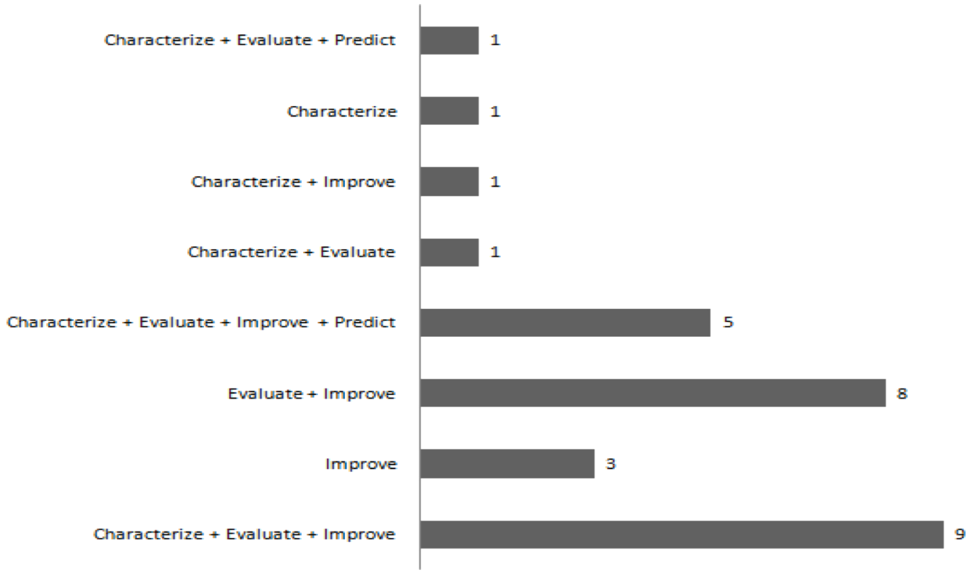


Figure 7: Distribution of measurement purposes of measurement models for implementing MPs

Figure 8: Distribution of implementation levels of measurement models for implementing MPs

SMEs. The SMEs do not have enough resources to promote serious MPI plans [S2, S9], [1, 10, 11]. All staff members are involved in the activities related to managing daily work and have no extra time for additional activities, such as implementing MPs. The implementation of MPs face major challenges such as limited resources to perform MPI [S9, S29, S33, S34, S35] and the lack of measurement experts [S12, S30, S32, S33, S35] and the lack of time for accurate estimations [S13, S29].

Poor software measurement knowledge: SMEs have poor measurement culture due to the lack of measurement knowledge, training and the perceived importance by administrators in SMEs [S12, S30, S32, S33, S35]. Therefore, a few measures are collected in these companies [S2, S21]. The lack of knowledge of measurement techniques among the software developers [S17], [11] also hinders the collection of measurement data. Developers seem to be in a great confusion about what to measure and how to measure [S17], [11]. They feel threatened by the possible adoption of a metrics program, as they perceive it as a tool that would be used for assessing their performance. Most of the developers have an insufficient knowledge of tools widely discussed and available in the literature. The management at SMEs usually do not understand the importance of a measurement process and the developers are mostly fresh university graduates equipped with insufficient knowledge about software quality and the importance of measurement [10, 46]. The people that are involved in MPs are not willing to use measurements due to their lack of knowledge of measurement techniques [S2].

Lack of experienced professionals: The stakeholders of the MPs including the measurement analyst usually come from the company implementing MPs. They usually have limited

Table 7: Challenges of implementing measurement programs at SMEs

Study ID	Challenges
S1, S2, S3, S29, S30, S33	Lack of measurement maturity for implementing software MPs. Lack of experience in using data collection tools.
S4, S29, S30, S31, S32, S33, S35	Lack of measurement maturity.
S5, S25, S29	Scope of databases containing indicators and measures is small.
S6, S29, S33	Formal process management techniques.
S8, S25, S29, S33, S35	Lack of measurement maturity. Lack of automated tool for data collection.
S9, S29, S33, S34, S35	Limited resources to perform measurement improvements.
S11, S33	Lack of formal measurement approach for software process assessment. Software process assessment is time consuming and costly at SMEs.
S12, S30, S32, S33, S35	Lack of measurement experts.
S13, S29	Lack of time for accurate estimation of projects.
S15, S30	Selected metrics are not verified for implementing measurements at SMEs.
S17, S25, S29, S33	Use of metrics is limited due to unawareness of software measurement techniques among the software developers. Measurement is considered a long-term activity. Short time-to-market. Use of metrics is limited due to lack of experienced professionals. Measurements are limited due to lack of knowledge of quality issues in development process.
S18, S33	Selected metrics are not validated for measurement and evaluation of SPI.
S19, S33	The absence of automated tool for data collection. Projects have a limited budget for empirical data collection and analysis.
S20	Cost management (time and resources needed for collection and analysis of metrics). Redundancy in metric selection process.
S22 , S23	Redundancy in metric collection. High effort required for metrics selection and collection.
S24, S29, S31, S33	Unavailability of the required assessment data to measure.
S4, S25, S28, S29, S30, S33, S34, S35	Lack of sync between measurement process and software development life cycle.
S25, S26, S27	Incorrect definition of measures.
S32, S33, S34, S35	Lack of sync between business objectives/strategies and SPI.

expertise in the measurement field [S1, S2]. The SMEs should hire experienced professionals in permanent positions to plan, organize, implement, evaluate and improve MPs [8, 47]. A few case studies (e.g. [S1, S2, S3, S29, S30, S33]) showed that all of the measurement processes proposed in measurement studies are not possible to implement yet due to poor measurement maturity, poor measurement knowledge, and the lack of experience in using data collection tools. The SMEs face difficulties in hiring experienced professionals, because the offered reward is limited. Once the developers gain some experience, they seem to be inclined to migrate to larger companies hoping for better career prospects [S12].

Time to market: The use of software metrics is limited in SMEs due to challenging time to market with tight timeframes [S17]. Software developers in SMEs are always found battling with time pressures [S13, S29]. Most of the SMEs are aware that software measures are useful for improving quality but they believe that it requires more time to implement a MP in the workplace [11].

Lack of measurement planning: Most of the SMEs have poor strategic planning processes for implementing their MPs due to barriers such as unavailability of assessment data [S24, S29, S31, S33], rapid application development [S13, S29], lack of formal process management, measurement management techniques and unwillingness to share ideas with employees [S6, S11, S29, S33], [46, 48, 49]. The lack of measurement planning also hinders linking measurement processes with business objectives and SPI [S32, S33, S34, S35].

Lack of automated tool support: The automated tools used in SMEs can be different due to multiple reasons. They can be different based on the implementation levels of MPs (i.e. organizational and/or project level), types of software entities to be measured (processes, products and/or resources), type of software development life cycle (e.g. agile, rapid application development), measurement purpose (characterize, evaluate, predict and/or improve software entities) and the business goals of software organization. There is a lack of automated tools for implementing software MPs in SMEs [11, 46, 50] as there are only four tools reported among 35 primary studies in this SMS (i.e. Tarc [S9], SCAPT [S24], SonarQube [S25], SPIALS [S35]). There is an increasing need for well understood and affordable tools that can select required metrics to implement software MPs in SMEs [S8] [46]. The automated tools might also help to overcome budget limitations, time and measurement experts in SMEs [S12].

The databases in SMEs contain a small number of measures and indicators [S5, S25, S29]. The small scope of measurement databases might be due to the lack of synchronization between a measurement process and a software development life cycle [S4, S25, S28, S29, S30, S33, S34, S35]. The lack of automation and small scope of databases causes redundancy in metrics collection and high effort is required for metrics collection [S22, S23].

Data collection problem: The unavailability of the required assessment data [S19, S24] for measurement tools is a critical challenge. This problem might not only reduce the descriptive power of the tool but also reflect company's operational problems. The tools perform effectively if the company has defined data collection and storage procedures [S19, S24]. Furthermore, projects have limited budget for empirical data collection and analysis [S19, S33]. Therefore, there is a need for automated tools, which can help to reduce the overhead associated with data collection and processing to perform measurements in SMEs [S8]. The lack of budget, time and resources also hinders the quality assurance process for the data collection process [S15, S30] and the validation of metrics for their suitability for SPI improvement [S18, S19, S20, S25, S26, S27, S33].

It was not possible to find any solution to the problem of initiating the data collection process in this mapping study, however, the SLR [6] revealed that Iversen and Mattiassen [51] discussed experiences in establishing an MP with the help of incremental application of GQM and intelligent collection and analysis of data. Therefore, the automation of data collection process can be incrementally implemented. The first step may include the collection of data with manual entries into measurement repository using a tool. In the second step, data collection may also be automated. This requires the integration of the MP with the SDLC [S4, S25, S28, S29, S30, S33, S34, S35]. There are both open source and commercial tools to automate the data collection for SDLC processes [52]. The use of automated tools for characterization, evaluation, and prediction of software processes, products and resources becomes even more important in SMEs because there is a shortage of time, human and financial resources in SMEs.

RQ3: "What metrics selection techniques, methods and approaches are used for measurement programs in SMEs?"

Table 8 presents the most commonly used metrics based on their frequency of being discussed among the primary studies. The Software metrics/measurement-attributes/measures are identified, collected and analysed based on the definition of specific measurement objectives (e.g. defect prediction, size estimation).

Gómez et al. [53] identified in a SLR that complexity and size are most discussed metrics

Table 8: Types of metrics/measures in primary studies

Metric/Measurement-attribute/Measure	Definition	Selected studies	Frequency
Defects	Errors or failures in a software product.	S1, S2, S3, S4, S5, S6, S8, S12, S17, S26, S27, S31, S33, S35	14
Productivity	The speed of software production in terms of effort and time.	S1, S3, S4, S5, S17, S14, S15, S16, S22, S26, S28, S29, S30, S33, S34	15
Customer satisfaction	The expectation of customer about the performance of software product.	S1, S3, S4, S5, S7, S10, S12, S14, S24, S31, S33, S34, S35	13
Size	The size of the product in the form of functional points or LOC.	S2, S6, S13, S15, S21, S22, S27, S33	8
Duration	The time required to construct software product.	S1, S2, S3, S4, S5, S6, S7, S9, S22, S24, S26, S29, S33, S35	14
Effort	The human effort to develop a software product.	S1, S2, S3, S4, S5, S13, S15, S16, S21, S22, S26, S29, S35	13
Reliability	Number of error-free operations in a system under particular conditions.	S1, S3, S4, S5, S24, S31	6
Traceability	A measurement that counts the software requirements that are not traced to the system requirements.	S6, S8, S18, S31, S33	5
Cyclomatic complexity	A measurement that shows the complexity of software product.	S2, S6, S8, S26	4

Table 9: Metrics selection methods

Metrics selection methods	Studies	Frequency	Percentage
Use of standards	S2, S6, S11, S17, S19, S20, S22, S23, S28, S29, S30, S32, S33, S35	14	40%
Use of measurement expert and experiences	S1, S3, S4, S5, S7, S8, S9, S10, S12, S13, S15, S16, S18, S26, S27, S28, S31, S32, S33, S34, S35	21	60%
Use of automated tools	S2, S9, S16, S19, S21, S25, S33, S35	8	22%

among primary studies on software measurement process in software development life cycle. An SLR [6] allowed to establish that defect, productivity and size are the most discussed metrics in large organizations. On the other hand, productivity, defects, effort and customer satisfaction are the most discussed metrics among primary studies in this SMS. There is an increasing need for a well understood and managed software measurement model in SMEs, to select the correct, relevant, timely, verifiable, cost-effective and valuable set of metrics [54].

In our previous study [6], metrics selection methods are classified as (i) use of standards, (ii) use of measurement experts and experience and (iii) use of automated tools. The same classification was used for metrics selection methods in this SMS as shown in Table 9. In this SMS, the use of a measurement expert and experience is the most practiced method among primary studies.

Use of standards: In an SLR on MPs [6], the primary studies discussed the role of standards such as ISO/IEC 15939:2007 [55], ISO/IEC 25000 [56], ISO/IEC 9126-x [57], ISO/IEC 14598-x [58], ISO/IEC/IEEE 24765:2010 [59], CMMI [60, 61], ISO/IEC 25021 [62], and ISO 9126 standard family [63–65] for the implementation of MPs.

In another SLR [20], the primary studies discussed the role of SPI models (SPICE, PSP, TSP, Six-Sigma, QIP, TQM) [66] and standards (e.g. CMMI, CMM, ISO 15504 [53] and ISO 9001 [53]) for the implementation of MPs. On the other hand, Pino et al. in an SLR [23] considered that ISO and SEI standards for SPI are not directly suitable for SMEs due to the complexity of recommendations, and the requirement of a large investment of time and resources. Therefore, they considered a need for widely accepted strategies to adapt these standards in SMEs and organizations. Furthermore, they considered that organizations which develop international Software Engineering standards should separately consider the measurement processes of SMEs [23].

In this SMS, multiple studies (e.g. [S2, S6, S11, S17, S19, S20, S22, S23]) stated that measurement standards (e.g. ISO/IEC 15504 [53], ISO 9001 [67]) are used to select metrics in different SMEs. The primary studies proposed multiple models to adapt those measurement standards in SMEs which are reported for MPs in large organizations. The MIS-PyME [S2], SQIP [S6], PRISMS [S8], MESOPYME [S9], AAHA [S11], ASPISME [S27], and CMMIbyScrum [S35] models are proposed to adapt CMMI standard to SMEs. Irrazabal et al. proposed guidelines to adapt ISO/IEC 12207:2008 standard for SCRUM [S28]. Similarly, Marìa et al. proposed guidelines to adapt ISO/IEC 15939:2007, ISO/IEC 12207:2008 and CMMI in SMEs [S29].

In [S1, S2, S3, S4, S8, S25, S29, S30, S31, S32, S33, S35], there is a proposal to implement MPs in SMEs according to the maturity level of software processes in the company. The MIS-PyME measurement capability maturity model [S2] for implementing MPs in SMEs uses ISO/IEC 15504 standard as a reference model [53]. The SPI models use measurements as the key component of their processes. For instance, the CMMI model contains guidelines for defining the measurement process and then using this process to monitor and control software development processes. Later, the collected measurement data is used for quantitative management and continuous improvement.

In [S20, S22, S23], the idea of using a predefined pool of standard metrics is proposed. The software companies can choose metrics from this pool based on their measurement goals using meta-metrics (importance of metrics for measurement goal, cost/time of metrics collection, and frequency of metrics usage in measurement project). The usage of a common set of metrics for different projects which have similar goals, might reduce the effort and cost of data collection.

Use of measurement experts and experience: Most of the SMEs use measurement experts and experiences to select metrics [S1, S3, S4, S5, S7, S8, S9, S10, S12, S13, S15, S16, S18, S26, S27, S28, S31, S32, S33, S34] [41].

It is challenging to implement MPs in SMEs due to their limited resources [S13, S26, S27, S28, S31, S32, S33, S34] [41]. Most of project managers in SMEs perform measurement planning (e.g. estimating budget, schedule and effort) based on their experience and knowledge from previous projects [S4, S9, S18, S27, S32], [68].

Use of automated tools: In SLR on software MPs [6], the automated tools are divided into two main categories:

1. Tools that are specifically developed for measurement processes. These tools (e.g. StepCounter, Workflow, Eclipse Metrics plug-in) also help to provide data for effective measurement implementation.
2. Tools that are a part of the processes of any organization, e.g. project management, quality assurance. These tools are usually part of the whole management information system. The limitations of such tools include lack of metrics data exchange formats, effective usage of collected data to feed the decision making process, and using collected data to effectively monitor and control the software development processes.

In [S19], project management officers used Tarc (self-assessment tool) for the selection and collection of metrics based on the predefined data collection procedure. They defined 10 fundamental metrics and 7 derived metrics (e.g. productivity, effort per day, review density, problem density, test density, bug density) to measure size, quality and effort attributes using Tarc [S19]. The collected metrics were used for quality assurance.

The SCAPT tool [S24] measures the performance of SMEs based on time, cost and reliability of software production. SCAPT depends upon the availability of the company's own data collection procedure. It is tested on 44 different SMEs and it is observed that the unavailability of assessment data is a major hindrance for performance estimation.

The SonarQube tool is proposed to collect and analyse measurement data on software quality assurance practices in SMEs [S25]. Its objective is to continuously monitor a source code for problems such as code smells, antipatterns, and unused methods. The best practices of software quality assurance based on literature and experiences are maintained in the tool.

The SPIALS tool [S35] is based on the Standard CMMI Appraisal Method for Process Improvement (SCAMPI). Its objective is to assess SPI by using the lightweight CMMIbyScrum model. It measures SPI by conducting a survey with the help of a structured questionnaire that is based on the CMMIbyScrum model.

Comparison of measurement programs in SMEs and large organizations

In [6], the authors performed an SLR on software MPs and observed that 4 out of 65 primary studies focused on the MPs in SMEs. Therefore, we conducted this SMS to analyse factors, such as measurement models, challenges and metrics selection methods for implementing MPs in SMEs.

In this section, a comparison between software MPs in SMEs and large companies is presented. The SLR [6] identified 35 measurement models and 11 tools and SMS identified 29 measurement models and 4 tools. There are 4 measurement models in SLR that are proposed for SMEs, i.e. SPGQM [69], OMSD [9], MIS-PyME [8], and GQM-DSFMS [70] and these four models are identified as common between both studies. All of these four models are based on goal-oriented approaches.

The measurement models are categorized into "goal oriented approach improvement (GOAI)", "software process improvement (SPI)" and "measurement process improvement (MPI)" in both studies. Figure 9 shows that the majority of measurement models in the SLR are GOAI followed by MPI and SPI. On the other hand, the majority of measurement models in SMS are SPI followed by GOAI and MPI.

The metrics selection methods are categorized into "use of measurement standards", "use of measurement experts and experiences" and "use of automated tools" in both studies. The SLR [6] and SMS analysed a different number of primary studies; therefore, the frequencies and percentages of primary studies discussing these standards are presented in Figure 10a and Figure 10b.

One of the reasons for the disparity in the number of studies between SLR and SMS might be the late evolution of SMEs industry in the last two decades. The history of software measurement and how it became critical of SMEs is discussed at the beginning of Section 4. The primary studies in the SLR [6] discussed ISO/IEC 15939:2007 [55], ISO/IEC 25000 [56], ISO/IEC 9126-x [57], and ISO/IEC 14598-x [58], ISO/IEC/IEEE 24765:2010 [59], CMMI [60, 61], ISO/IEC 25021 [62], and ISO 9126 standard family [63–65]. On the other hand, in this SMS there were measurement models proposed to adapt guidelines and methods of those measurement models that are already reported for large organizations [6, 20]. The MIS-PyME [S2], SQIP [S6], PRISMS [S8], MESOPYME [S9], AAHA [S11], ASPISME [S27], and CMMIbyScrum [S35] models should adapt the CMMI standard in SMEs. Irrazabal et al. proposed guidelines to adapt ISO/IEC 12207:2008 standard for SCRUM [S28] and María et al. proposed guidelines to adapt ISO/IEC 15939:2007, ISO/IEC 12207:2008 and CMMI to SMEs [S29]. Pino et al. in an SLR [23] considered that ISO and SEI standards for SPI are not directly suitable for SMEs due to the complexity of recommendations and the requirement of a large investment of time and resources. Therefore, there is a need for widely accepted strategies to adapt these standards in SMEs. The organizations that develop international Software Engineering standards should separately consider implementing measurement processes in SMEs [23].

An MP was divided into three phases for further analysis. These phases are the pre-implemen-

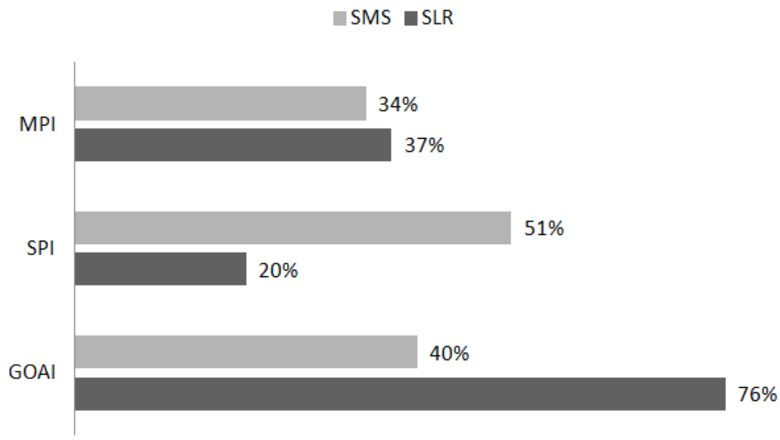
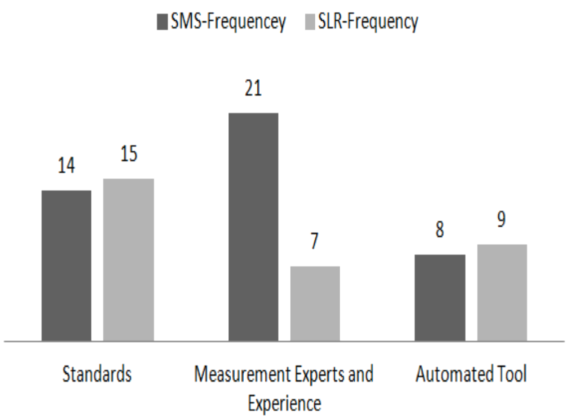
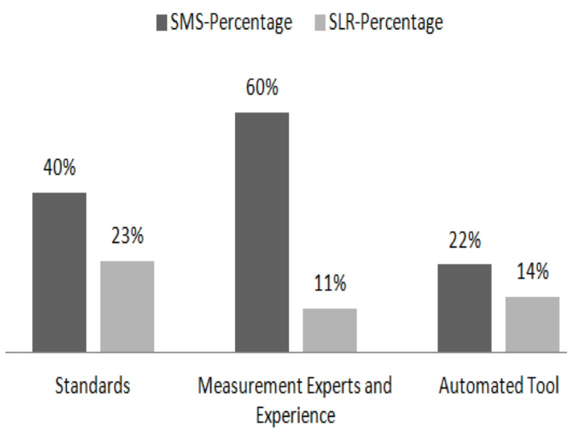


Figure 9: Comparison of categories of measurement models between SLR and SMS



(a) Metrics selection methods with respect to frequencies of primary studies discussing metric selection methods



(b) Metrics selection methods with respect to percentages of primary studies discussing metric selection methods

Figure 10: Metrics selection methods

tation, implementation and post-implementation of a MP.

Table 10 presents measurement purposes with respect to the phases of implementing an MP as shown in Figure 9. The pre-implementation phase of an MP starts with planning a software development process. In this phase, historical data from previous projects, measurement standards, measurement experts and experiences and automated tools might be used to predict the attributes of processes, products and resources (the details are in the results and analysis of RQ3). The implementation phase of an MP includes the characterization of issues/problems during software development life cycle and the continuous evaluation of project progress with respect to plans and predictions. The post-implementation phase of an MP helps in software process improvement based on lessons learned during the pre-implementation and implementation phase. The improvements can be twofold: 1) improvement in measurement processes, 2) improvement in software development processes. The prediction is the least utilized purpose among primary studies of SMS and SLR as shown in Table 10.

The measurement models for SMEs are specifically designed to implement the measurement process keeping the basic limitations of SMEs, such as budget, time, resources and low process maturity, in view. The measurement models proposed for large companies focus on broad issues, such as the measurement of customer satisfaction,

Table 10: Purposes of measurement program

Measurement studies	Pre-implementation prediction	Implementation characterization	Implementation evaluation	Post-implementation improvement
SLR	28%	81%	77%	70%
SMS	16%	63%	83%	93%

Table 11: Metrics discussed among primary studies of SMS and SLR

Measurement process	Measurement attributes	Metrics type	SMS Frequency	SMS Percentage	SLR Frequency	SLR Percentage
Pre-implementation	Size	Product	8	22.5%	10	15.4%
	Duration	Process	14	40%	7	10.7%
	Effort	Resource	13	37.1%	11	16.9%
	Cost/Budget	Process	-	-	4	6.1%
	Time to market	Product	-	-	3	4.6%
Implementation	Productivity	Resource	15	42.58%	11	16.9%
	Traceability	Product	5	14.2%	-	-
	Cyclomatic complexity	Product	4	11.4%	-	-
	Employee Commitment	Resource	-	-	8	12.3%
Post-implementation	Return on investment	Product	-	-	3	4.6%
	Customer satisfaction	Product	13	37.1%	8	12.3%
	Defects	Product	14	40%	25	38.5%
	Reliability	Product	6	17.1%	-	-

effectiveness of decisions taken based on MPs, verification and validation of the metrics collection process, building an information system for the measurement process, and the improvement of software development processes [6].

Table 11 presents the most commonly used metrics based on how frequently they are discussed in the primary studies in SMS and SLR. Fenton and Bieman [5] distinguished three types of measurement entities, i.e. process, product, and resource. Table 11 shows that the product metrics are mostly measures in the primary studies of SLR [6] and SMS. It also points out the need for more utilization of process and resource metrics for planning, organizing, monitoring, and controlling the processes and resources.

In SLR [6], they found that there is a lack of discussion of real-time metrics among primary studies (e.g. cyclomatic complexity, dynamic function calls, number of unused objects) to monitor and control the actual software development progress. Soini [71] conducted an empirical case study in the Finnish software industry to evaluate the actual use of software metrics. The software metrics are categorized into real-time and lagging metrics [71]. The real-time metrics help to monitor and control the ongoing processes in software organizations and provide indicators (e.g. cyclomatic complexity and traceability in this SMS). The lagging metrics are collected at the completion of projects (e.g. return on investment and customer satisfaction in this SMS). The balance between real time and lagging metrics might assist improvement in measurement processes [71].

Table 11 shows that all three types of metrics (i.e. process, product and resource) are only discussed for the pre-implementation phase of MPs. Furthermore, the process and product types of metrics are discussed twice as resource metrics in the pre-implementation phase. The resource and product types of metrics are discussed for the implementation phase of MPs and only product type of metrics are discussed in the post-implementation phase. The measurement of software defects is the most commonly discussed metric in both studies.

Table 12: Comparison of the challenges of implementing measurement programs in this SMS and SLR [6]

Challenges reported in SMS	Challenges reported in SLR
Pre-implementation – Lack of budget, time and resources allocated for software measurement. – Use of metrics is limited due to lack of experienced professionals. – Lack of measurement experts. – Lack of measurement maturity for implementing software MPs. – Absence of documentation and formal process management techniques. – Lack of automated tools for data collection. – Metrics are not validated for use in SMEs. Implementation – Scope of database containing indicators and measures is small as limited number of metrics are utilized in SMEs. – Limited utilization of metrics due to lack of defined process for management of quality issues in development process.	Pre-implementation – Lack of benchmarks. – Heterogeneity of SDLCs, MPs, products, culture, and priorities. Implementation – Correctness of MPs objectives. – Prioritisation of goals. – Transition to measurement culture. – Construct validity issues of metrics. – Lack of consistent definitions of measurement entities, tasks and processes. – Sync between MPs and SPI activities. – Overlapping between the metrics types. – Scalability issues in MPs. – Identification of correct measurement instrument. – Completeness, integrity, consistency of measurement data. – Lack of suitable metrics selection methods. – Lack of real time metrics (e.g. cyclomatic complexity, dynamic function calls, no of unused objects and variables) to monitor and control the actual software development progress. Post-implementation – Sustainability of MPs.

Table 12 presents the challenges of implementing MPs in SMEs and large organizations. The challenges are presented with respect to pre-implementation, implementation and post-implementation phases of an MP.

Pre-implementation challenges: The challenges which already exist in the software development organization (e.g. lack of budget, and time) or they exist in the software measurement domain (e.g. inconsistent measurement terminologies) before the implementation of MPs.

Implementation challenges: The challenges which appear during the implementation of MPs.

Post-implementation challenges: The challenges which appear after the implementation of MPs.

In the primary studies of SMS, most of the reported challenges exist even before the implementation of MPs in SMEs. They are of fundamental significance and encompass, e.g. lack of budget, time and resources. The SMEs usually hire fresh or less experienced graduates, which causes the lack of understanding and attention towards software quality and measurement issues [10,46]. The lack of defined measurement processes results in a situation when it is the higher management to decide on the importance of MPs and consequently the mechanism becomes people-oriented instead of process-oriented [S9, S18, S27]. The absence of formal documentation and automated measurement tools also hinders measurement processes because both are key sources to provide data for measurement [S19, S24]. It is also critical to learn whether the measured values are exactly the ones that were to be measured [72, 73]. The lack of metrics validation also imposes a challenge, as metrics must be mathematically correct and useful for decision-making [74], [S18, S19, S20, S25, S26, S27, S33].

The challenges faced during the implementation of MPs in SMEs include a limited scope of measurement repository (database) in terms of using metrics for the characterization, evaluation, prediction and improvement of software entities at project and organization level [S5, S25, S29]. There are only few fundamental metrics which are used mostly by SMEs to plan, monitor and control software entities such as processes, products and resources [S22, S23].

The challenges reported by studies in large companies are mostly related to the issues discovered while implementing MPs [6]. The primary studies in SLR [6], report the lack of measurement benchmarks in terms of publically available measurement datasets, measurement standards and widely accepted measurement models and tools.

The heterogeneity of software organizations might be a challenge for implementing MPs in both SMEs and large organizations, e.g. in terms of software development life cycle (waterfall, agile etc.), size of organization (small, medium and large), domain of software products (e-commerce, mainframe systems, etc.), implementation levels of MPs (project or organization-wide), measurement purposes (characterize, evaluate, predict and/or improve) and measurement culture [6,20].

Construct validity is also a key challenge while implementing an MP, however, it was not possible to find specific discussions or solutions presented to address this challenge in SMEs. Kaner defined construct validity as, “How do you know that you are measuring what you think you are measuring” [73]. The software measurement is defined as the empirical, objective assignment of numbers according to a theory or model, to characterize the attribute of processes, products and resources [73]. In an SLR on the validation of software metrics [74], the word “construct” is referred to as a tool, instrument or procedure used to collect metrics. There are 47 validation criteria of software metrics presented in the SLR [74], however, they need further evaluations by researchers and practitioners to select suitable metrics validation criteria for measurement processes in large and SMEs industry. In this study 53 citations of the SLR [74] using Google Scholar were found, however, none of these specifically focused on metrics validation for SMEs.

Table 13 presents the comparisons of the implementation of MPs at project and organization level in the measurement studies of SLR [6] and SMS. According to both studies, it is challenging for software development organizations to implement MPs at both levels [6]. It might be due to the fact that most of the measurement models are designed to solve a specific problem at project level or organization level and their implementation is usually limited to a specific project. These factors might hinder the continuity of MPs for a longer period of time and at both implementation levels of MPs. Furthermore, 51% of the primary studies in SLR [6] and SMS are case studies. It is considered in [6,20,23] that there is a lack of comparative case studies of MPs. One of the potential reasons might be the fact that there is no clear context description in the published case studies. The context description might include organizational context of case studies, such as type and size of organization, type of products, measurement stakeholders. The description of the measurement process might include the type of metrics collected and the analysed, duration of measurement processes, analysis methodologies, link between measurement processes and improvement activities [6]. A comprehensive context description will help practitioners and researchers to achieve the repeatability, extensibility, and comparisons of case studies [6,20].

Table 13: Comparison of the implementation levels of measurement programs

Implementation levels of MPs	SLR	SMS
Project	58%	30%
Organization	28%	30%
Project AND Organization	14%	40%

The challenges faced during implementation of MPs at SMEs include limited scope of measurement repository (database) in terms of using metrics for characterization, evaluation, prediction and improvement of software entities at project and organization level [S5, S25, S29]. There are only few fundamental metrics which are used mostly by SMEs to plan, monitor and control software entities such as processes, products and resources [S22, S23]. These challenges exist even before the implementation of MPs at SMEs.

In SLR [6], the incremental development of MPs is also mentioned as a solution for software organizations having no or partially defined MPs [52]. It was not possible to find any solution to the problem of initiating a measurement process in this mapping study. However, it was found in the SLR [6] that Iversen and Mattiassen [51] discussed the experiences of establishing an MP with the help of the incremental application of GQM and the intelligent collection and analysis of data. Therefore, the automation of the data collection process can be implemented incrementally. The first step may include the collection of data with manual entries into a measurement repository using a tool. In the second step, data collection may also be automated. This requires the integration of the MP with the SDLC [S4, S25, S28, S29, S30, S33, S34, S35]. There are both open source and commercial tools to automate the data collection for SDLC processes [52]. The use of automated tools for the characterization, evaluation, and prediction of software processes, products and resources becomes even more important in SMEs because there is a shortage of time, human and financial resources.

Large organizations mostly report challenges observed during the implementation of MPs while SMEs report pre-implementation challenges (e.g. budget, time, lack of measurement process maturity). The literature lacks challenges and mitigation strategies while implementing MPs at SMEs. Therefore, the SMEs can also evaluate mitigation strategies for the challenges presented in [6] according to their needs while implementing MPs.

5. Conclusion

The systematic mapping process proposed by Petersen et al. [21] is used to conduct this Systematic Mapping Study (SMS) [21]. The main objective of this mapping study is to identify and analyse the studies on software measurement programs (MPs) in small and medium enterprises (SMEs). In total, 35 primary studies are analysed to answer the following research questions:

RQ1: What measurement models, tools and practices for implementing measurement programs in SMEs are discussed in literature?

RQ2: What are the problems, challenges and issues of implementing measurement programs in SMEs?

RQ3: What metrics selection techniques, methods and approaches are used for measurement programs in SMEs?

This SMS analyses 29 measurement models and 4 tools. The measurement models are categorized into "goal oriented approach improvement (GOAI)", "software process improvement (SPI)" and "measurement process improvement (MPI)". The majority of the measurement models are built upon SPI (51%) approaches followed by GOAI (40%) and MPI (34%) approaches. As for the implementation level of MPs, most measurement models are implemented at both the project and organization level (40%) followed by project level (30%) and organization level (30%). With respect to the measurement purposes of models, the distribution of MPs is identified as: characterization (63%), evaluation (83%), improvement (93%) and prediction (16%). When the combination of purposes (i.e. when more than one purpose was mentioned by a primary study) was investigated, it was found out that around 59% of the studies mentioned the purposes of characterization and improvement while only 17% referred to all four purposes. This situation might be due to the fact that prediction based on historical data is possible if an MP lasts longer than a single project.

The metrics selection methods in primary studies are categorized into "use of measurement standards", "use of measurement experts and experiences" and "use of automated tools". The majority of primary studies discussed the use of measurement experts and experience (60%) followed by the use of measurement standards (40%) and the use of automated tools (22%). The common types of metrics discussed in the primary studies include productivity (43%), defects (40%), duration (40%), effort (37%), customer satisfaction (37%), size (22%), and cyclomatic complexity (11%). The most commonly used research methods in primary studies are a case study (51%) and a survey (25%). Most of the primary studies (80%) were published between 2006 and 2013.

Most of the SMEs face challenges, such as low measurement process maturity, limited resources to develop MPs and short time-to-market. Furthermore, the lack of measurement planning, tool support for data collection and measurement professionals are key challenges for the implementation of MPs.

In this study, the MPs in SMEs and large organizations are also compared. Most of the measurement models for SMEs are built upon the software process improvement approach. On the other hand, most of measurement models for large organizations are built upon goal-oriented approaches. The measurement models in SMS and SLR [6] focus the least on using measurement data for prediction. There is a lack of automated tools support for implementing MPs as there are 11 and 4 tools identified for large organizations and SMEs, respectively.

The SMEs and large organization face different challenges as studies in SMEs report challenges that existed even before the implementation of MPs due to different infrastructure and management processes of SMEs. Therefore, lightweight measurement models are proposed to cater for measurement processes while keeping the limitations of SMEs, such as budget, time and resources, in view. In this SMS, we found the measurement models which are proposed to adapt the guidelines and methods of those measurement models that are already reported for large organizations [6,20]. For instance, the MIS-PyME [S2], SQIP [S6], PRISMS [S8], MESOPYME [S9], AAHA [S11], ASPISME [S27], and CMMIbyScrum [S35] models are proposed to adapt the CMMI standard in SMEs. On the other hand, the challenges reported by studies in large companies are mostly related to the issues discovered while implementing MPs. These measurement studies report challenges, such as lack of measurement benchmarks in terms of measurement datasets, standards and widely accepted measurement models and tools. The challenges also include the lack of synchronization among measurement processes, software development processes and software improvement processes, and the adoption of measurement culture.

This SMS presented the findings from the existing literature. We are currently conducting online surveys in SMEs to validate the findings of SMS.

References

[1] P. Cocca and M. Alberti, "A framework to assess performance measurement systems in SMEs," *International Journal of Productivity and Performance Management*, Vol. 59, No. 2, 2010, pp. 186–200.

[2] U. Loecher, "Small and medium-sized enterprises – Delimitation and the European definition in the area of industrial business," *European Business Review*, Vol. 12, No. 5, 2000, pp. 261–264.

[3] M. Ayyagari, T. Beck, and A. Demirguc-Kunt, "Small and medium enterprises across the globe," *Small Business Economics*, Vol. 29, No. 4, 2007, pp. 415–434.

[4] M. Khalique, N. Bontis, J. Abdul Nassir bin Shaari, and A. Hassan Md. Isa, "Intellectual capital in small and medium enterprises in Pakistan," *Journal of Intellectual Capital*, Vol. 16, No. 1, 2015, pp. 224–238.

[5] N. Fenton and J. Bieman, *Software metrics: A rigorous and practical approach.* CRC Press, 2014.

[6] T. Tahir, G. Rasool, and C. Gencel, "A systematic literature review on software measurement programs," *Information and Software Technology*, Vol. 73, 2016, pp. 101–121.

[7] M. Díaz-Ley, F. García, and M. Piattini, "Implementing a software measurement program in small and medium enterprises: A suitable framework," *IET Software*, Vol. 2, No. 5, 2008, pp. 417–436.

[8] M. Díaz-Ley, F. García, and M. Piattini, "MIS-PyME software measurement capability maturity model – Supporting the definition of software measurement programs and capability determination," *Advances in Engineering Software*, Vol. 41, No. 10, 2010, pp. 1223–1237.

[9] A.M. Bhatti, H.M. Abdullah, and C. Gencel, "A model for selecting an optimum set of measures in software organizations," in *European Conference on Software Process Improvement.* Springer, 2009, pp. 44–56.

[10] M.K. Sharma, R. Bhagwat, and G.S. Dangayach, "Practice of performance measurement: experience from indian SMEs," *International*

Journal of Globalisation and Small Business, Vol. 1, No. 2, 2005, pp. 183–213.

[11] V. Claudia, M. Mirna, and M. Jezreel, "Characterization of software processes improvement needs in SMEs," in *International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE*. IEEE, 2013, pp. 223–228.

[12] I. Richardson and C.G. Von Wangenheim, "Guest editors' introduction: Why are small software organizations different?" *IEEE Software*, Vol. 24, No. 1, 2007, pp. 18–22.

[13] C.Y. Laporte, S. Alexandre, and R.V. O'Connor, "A software engineering lifecycle standard for very small enterprises," *Software Process Improvement*, 2008, pp. 129–141.

[14] B. Kitchenham, "What's up with software metrics? – a preliminary mapping study," *Journal of Systems and Software*, Vol. 83, No. 1, 2010, pp. 37–51.

[15] O. Gómez, H. Oktaba, M. Piattini, and F. García, "A systematic review measurement in software engineering: State-of-the-art in measures," in *International Conference on Software and Data Technologies*. Springer, 2006, pp. 165–176.

[16] C. Catal and B. Diri, "A systematic review of software fault prediction studies," *Expert Systems with Applications*, Vol. 36, No. 4, 2009, pp. 7346–7354.

[17] R. Malhotra, "A systematic review of machine learning techniques for software fault prediction," *Applied Soft Computing*, Vol. 27, 2015, pp. 504–518.

[18] D. Radjenović, M. Heričko, R. Torkar, and A. Živkovič, "Software fault prediction metrics: A systematic literature review," *Information and Software Technology*, Vol. 55, No. 8, 2013, pp. 1397–1418.

[19] T. Hall, S. Beecham, D. Bowes, D. Gray, and S. Counsell, "A systematic literature review on fault prediction performance in software engineering," *IEEE Transactions on Software Engineering*, Vol. 38, No. 6, 2012, pp. 1276–1304.

[20] M. Unterkalmsteiner, T. Gorschek, A.M. Islam, C.K. Cheng, R.B. Permadi, and R. Feldt, "Evaluation and measurement of software process improvementóa systematic literature review," *IEEE Transactions on Software Engineering*, Vol. 38, No. 2, 2012, pp. 398–424.

[21] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering." in *EASE*, Vol. 8, 2008, pp. 68–77.

[22] M. Sulayman and E. Mendes, "A systematic literature review of software process improvement in small and medium web companies," *Advances in Software Engineering*, 2009, pp. 1–8.

[23] F.J. Pino, F. García, and M. Piattini, "Software process improvement in small and medium software enterprises: A systematic review," *Software Quality Journal*, Vol. 16, No. 2, 2008, pp. 237–261.

[24] A. Ahmad and M.A. Babar, "Software architectures for robotic systems: A systematic mapping study," *Journal of Systems and Software*, Vol. 122, 2016, pp. 16–39.

[25] K. Petersen, S. Vakkalanka, and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update," *Information and Software Technology*, Vol. 64, 2015, pp. 1–18.

[26] F. García, M.F. Bertoa, C. Calero, A. Vallecillo, F. Ruíz, M. Piattini, and M. Genero, "Towards a consistent terminology for software measurement," *Information and Software Technology*, Vol. 48, No. 8, 2006, pp. 631–644.

[27] S. Keele *et al.*, "Guidelines for performing systematic literature reviews in software engineering," in *Technical report, Ver. 2.3 EBSE Technical Report. EBSE*. sn, 2007.

[28] J.W. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage Publications, 2013.

[29] N. Mack, C. Woodsong, K.M. MacQueen, G. Guest, and E. Namey, "Qualitative research methods: A data collectors field guide." *POPLine*, 2005.

[30] R. Conradi and A.I. Wang, *Empirical methods and studies in software engineering: experiences from ESERNET*. Springer, 2003, Vol. 2765.

[31] N.E. Fenton and M. Neil, "Software metrics: Successes, failures and new directions," *Journal of Systems and Software*, Vol. 47, No. 2, 1999, pp. 149–157.

[32] G. Tom, *Software Metrics*. Chartwell-Bratt, 1976.

[33] R.B. Grady and D.L. Caswell, *Software metrics: Establishing a company-wide program*. Prentice Hall, 1987.

[34] V.R. Basili, G. Caldiera, and H.D. Rombach, "The goal question metric approach," in *Encyclopedia of Software Engineering*. Wiley, 1994, pp. 528–532.

[35] R.E. Park, W.B. Goethert, and W.A. Florac, "Goal-driven software measurement. A guidebook," Carnegie Mellon University, Pittsburgh, PA, USA, Tech. Rep., 1996.

[36] L.C. Briand, C.M. Differding, and H.D. Rombach, "Practical guidelines for measurement-based process improvement," *Software Process Improvement and Practice*, Vol. 2, No. 4, 1996, pp. 253–280.

[37] Ø. Moen, M. Gavlen, and I. Endresen, "Internationalization of small, computer software firms: Entry forms and market selection," *European Journal of Marketing*, Vol. 38, No. 9/10, 2004, pp. 1236–1251.

[38] C. Larman and V.R. Basili, "Iterative and incremental developments. A brief history," *Computer*, Vol. 36, No. 6, 2003, pp. 47–56.

[39] A. Cockburn, *Agile software development*. Addison-Wesley Boston, 2002, Vol. 177.

[40] A. Boyd, "The goals, questions, indicators, measures (GQIM) approach to the measurement of customer satisfaction with e-commerce Web sites," in *Aslib proceedings*, Vol. 54. MCB UP Ltd, 2002, pp. 177–187.

[41] J. Motwani, "Critical factors and performance measures of TQM," *The TQM magazine*, Vol. 13, No. 4, 2001, pp. 292–300.

[42] A. Abran, J.W. Moore, P. Bourque, and R. Dupuis, Eds., *Guide to the Software Engineering Body of Knowledge (SWEBOK–2004 Version)*. IEEE Computer Society, 2004.

[43] A. Abran, L. Buglione, and A. Sellami, "Software measurement body of knowledge – initial validation using Vincenti's classification of engineering knowledge types," in *Software Measurement Conference*, 2004, pp. 1–16.

[44] S. Alexandre, A. Renault, and N. Habra, "POWPL: A gradual approach for software process improvement in SMEs," in *32nd EUROMICRO Conference on Software Engineering and Advanced Applications*. IEEE, 2006, pp. 328–335.

[45] A.H. Lee, W.C. Chen, and C.J. Chang, "A fuzzy AHP and BSC approach for evaluating performance of IT department in the manufacturing industry in Taiwan," *Expert Systems with Applications*, Vol. 34, No. 1, 2008, pp. 96–107.

[46] P. Garengo, S. Biazzo, and U.S. Bititci, "Performance measurement systems in SMEs: A review for a research agenda," *International Journal of Management Reviews*, Vol. 7, No. 1, 2005, pp. 25–47.

[47] O.T. Pusatli, "Software measurement activities in small and medium enterprises: An empirical assessment," *Acta Polytechnica Hungarica*, Vol. 8, No. 5, 2011, pp. 21–42.

[48] C. Wang, E. Walker, and J. Redmond, "Explaining the lack of strategic planning in SMEs: The importance of owner motivation," *International Journal of Organisational Behaviour*, Vol. 12, No. 1, 2007, pp. 1–16.

[49] J. Chen, "Development of Chinese small and medium-sized enterprises," *Journal of Small Business and Enterprise Development*, Vol. 13, No. 2, 2006, pp. 140–147.

[50] M. Hudson, A. Smart, and M. Bourne, "Theory and practice in SME performance measurement systems," *International Journal of Operations & Production Management*, Vol. 21, No. 8, 2001, pp. 1096–1115.

[51] J. Iversen and L. Mathiassen, "Cultivation and engineering of a software metrics program," *Information Systems Journal*, Vol. 13, No. 1, 2003, pp. 3–19.

[52] B. Daubner, "Empowering software development environments by automatic software measurement," in *11th International Symposium Software Metrics*. IEEE, 2005, p. 3.

[53] A. Coletta, "An industrial experience in assessing the capability of non-software processes using ISO/IEC 15504," *Software Process: Improvement and Practice*, Vol. 12, No. 4, 2007, pp. 315–319.

[54] H. Abushama, M. Ramachandran, and P. Allen, *PRISMS: an approach to software process improvement for small to medium enterprises*. UOFK, 2016.

[55] *Systems and software engineering – Measurement process*, International Organization for Standardization Standard ISO/IEC 15 939:2007, 2007.

[56] *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Quality measure elements*, International Organization for Standardization Standard ISO/IEC 25 021:2012, 2012.

[57] *Product quality – Part 1: Quality model*, International Organization for Standardization Standard ISO/IEC 9126-1:2001, 2001.

[58] *Information Technology – Software Product Evaluation – Parts 1-6*, International Organization for Standardization Standard ISO/IEC 14 598, 2001.

[59] *Systems and software engineering – Vocabulary*, International Organization for Standardization Standard ISO/IEC/IEEE 24 765:2010, 2010.

[60] "Capability maturity model integration (CMMI) (continuous representation)," Carnegie Mellon University, Tech. Rep. ICMU/SEI-2002-TR-011, 2002. [Online]. https://resources.sei.cmu.edu/asset_files/TechnicalReport/2002_005_001_14039.pdf

[61] "Capability maturity model integration (CMMI) (staged representation)," Carnegie Mellon University, Tech. Rep. CMU/SEI-2002-TR-012, SEI, 2002. [Online]. https://www.sei.cmu.edu/reports/02tr029.pdf

[62] *Software engineering: Software product quality requirements and evaluation (square) quality measure elements*, International Organization for Standardization Standard ISO/IEC 2502-1, 2005.

[63] *Product quality – Part 2: Quality model*, International Organization for Standardization Standard ISO/IEC 9126-2:2001, 2001.

[64] *Product quality – Part 3: Quality model*, International Organization for Standardization Standard ISO/IEC 9126-3:2001, 2001.

[65] *Product quality – Part 4: Quality model*, International Organization for Standardization Standard ISO/IEC 9126-4:2001, 2001.

[66] F.G. Wilkie, D. McFall, and F. McCaffery, "An evaluation of CMMI process areas for small-to medium-sized software development organisations," *Software Process: Improvement and Practice*, Vol. 10, No. 2, 2005, pp. 189–201.

[67] J.A. Williams, "The impact of motivating factors on implementation of ISO 9001:2000 registration process," *Management Research News*, Vol. 27, No. 1/2, 2004, pp. 74–84.

[68] M. Jørgensen, "A review of studies on expert estimation of software development effort," *Journal of Systems and Software*, Vol. 70, No. 1, 2004, pp. 37–60.

[69] T. Tahir and C. Gencel, "A structured goal based measurement framework enabling traceability and prioritization," in *6th International Conference on Emerging Technologies (ICET)*. IEEE, 2010, pp. 282–286.

[70] C. Gencel, K. Petersen, A.A. Mughal, and M.I. Iqbal, "A decision support framework for metrics selection in goal-based measurement programs: GQM-DSFMS," *Journal of Systems and Software*, Vol. 86, No. 12, 2013, pp. 3091–3108.

[71] J. Soini, "A survey of metrics use in finnish software companies," in *International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 2011, pp. 49–57.

[72] P. Carbone, L. Buglione, L. Mari, and D. Petri, "A comparison between foundations of metrology and software measurement," *IEEE Transactions on Instrumentation and Measurement*, Vol. 57, No. 2, 2008, pp. 235–241.

[73] C. Kaner *et al.*, "Software engineering metrics: What do they measure and how do we know?" in *10th International Software Metrics Symposium, METRICS*. IEEE Computer Society, 2004.

[74] A. Meneely, B. Smith, and L. Williams, "Validating software metrics: A spectrum of philosophies," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, Vol. 21, No. 4, 2012, p. 24.

[75] M. Díaz-Ley, F. García, and M. Piattini, "Implementing software measurement programs in non mature small settings," *Software Process and Product Measurement*, 2008, pp. 154–167.

[76] M. Díaz-Ley, F. García, and M. Piattini, "Software measurement programs in SMEs–defining software indicators: A methodological framework," *Product-Focused Software Process Improvement*, 2007, pp. 247–261.

[77] M. Diaz-Ley, F. García, and M. Piattini, "MIS-PyME software measurement maturity model-supporting the definition of software measurement programs," *Product-Focused Software Process Improvement*, 2008, pp. 19–33.

[78] A. Tosun, A. Bener, and B. Turhan, "Implementation of a software quality improvement project in an SME: A before and after comparison," in *35th Euromicro Conference on Software Engineering and Advanced Applications*. IEEE, 2009, pp. 203–209.

[79] E. Amrina and S.M. Yusof, "A proposed manufacturing performance measures for small and medium-sized enterprises (SMEs)," in *Proceedings of the 10th Asia Pacific Industrial Engineering and Management Systems (APIEMS) Conference*, 2009, pp. 623–629.

[80] J.A.C.M. Villalón, G.C. Agustín, T.S.F. Gilabert, A.D.A. Seco, L.G. Sánchez, and M.P. Cota, "Experiences in the application of software process improvement in SMEs," *Software Quality Journal*, Vol. 10, No. 3, 2002, pp. 261–273.

[81] I. Richardson and K. Ryan, "Software process improvements in a very small company," *Software Quality Professional*, Vol. 3, No. 2, 2001, pp. 23–35.

[82] F. McCaffery, M. Pikkarainen, and I. Richardson, "AHAA–Agile, hybrid assessment method for automotive, safety critical SMEs," in *Proceedings of the 30th International Conference on Software Engineering*. ACM, 2008, pp. 551–560.

[83] F.T. Shah, S. Shamail, and N. Ahmad Akhtar, "Lean quality improvement model for quality practices in software industry in Pakistan," *Journal of Software: Evolution and Process*, Vol. 27, No. 4, 2015, pp. 237–254.

[84] S. Bibi, I. Stamelos, G. Gerolimos, and V. Kollias, "BBN based approach for improving the software development process of an SME – A case study," *Journal of Software: Evolution and Process*, Vol. 22, No. 2, 2010.

[85] P. Clarke and R.V. OíConnor, "The meaning of success for software SMEs: An holistic scorecard based approach," in *European Conference on Software Process Improvement*. Springer, 2011, pp. 72–83.

[86] E. Caballero, J.A. Calvo-Manzano, and T. San Feliu, "Introducing scrum in a very small enterprise: A productivity and quality analysis," *Systems, Software and Service Process Improvement*, 2011, pp. 215–224.

[87] C.G. von Wangenheim, T. Punter, and A. Anacleto, "Software measurement for small and medium enterprises," in *Proceeding 7th International Conference on Empirical Assessment in Software Engineering (EASE)*, 2003.

[88] M. Sulayman, C. Urquhart, E. Mendes, and S. Seidel, "Software process improvement success factors for small and medium Web companies: A qualitative study," *Information and Software Technology*, Vol. 54, No. 5, 2012, pp. 479–500.

[89] N. Ohsugi, K. Fushida, N. Inoguchi, H. Arai, H. Yamanaka, T. Niwa, M. Fujinuki, M. Tomura, and T. Kitani, "Using trac for empirical data collection and analysis in developing small and medium-sized enterprise systems," in *ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 2015, pp. 1–9.

[90] H.M. Haddad and D.E. Meredith, "Instituting software metrics in small organizations: A practical approach," in *Eighth International Conference on Information Technology: New Generations (ITNG)*. IEEE, 2011, pp. 227–232.

[91] A. Potter, P. Childerhouse, R. Banomyong, and N. Supatn, "Developing a supply chain performance tool for smes in thailand," *Supply Chain Management: An International Journal*, Vol. 16, No. 1, 2011, pp. 20–31.

[92] A. Janes, V. Lenarduzzi, and A.C. Stan, "A continuous software quality monitoring approach for small and medium enterprises," in *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering Companion*. ACM, 2017, pp. 97–100.

[93] H. Ayed, N. Habra, and B. Vanderose, "AM-QuICk: A measurement-based framework for Agile methods customisation," in *Software Measurement and the 2013 Eighth International Conference on Software Process and Product Measurement (IWSM-MENSURA)*. IEEE, 2013, pp. 71–80.

[94] S. Suwanya and W. Kurutach, "Applying agility framework in small and medium enterprises," *Advances in Software Engineering*, 2009, pp. 102–110.

[95] E. Irrazabal, F. Vásquez, R. Díaz, and J. Garzás, "Applying ISO/IEC 12207:2008 with SCRUM and Agile methods," *Software Process Improvement and Capability Determination*, 2011, pp. 169–180.

[96] M. Díaz, F. García, and M. Piattini, "Defining, performing and maintaining software measurement programs: State of the art," in *IV Simposio Internacional de Sistemas de Informacióne*, 2006, p. 13.

[97] J.C. Ruiz, Z.B. Osorio, J. Mejia, M. Muñoz, A.M. Ch, B.A. Olivares *et al.*, "Definition of a hybrid measurement process for the models ISO/IEC 15504 – ISO/IEC 12207:2008 and CMMI Dev 1.3 in SMEs," in *Electronics, Robotics and Automotive Mechanics Conference (CERMA)*. IEEE, 2011, pp. 421–426.

[98] M. Jezreel, M. Mirna, N. Pablo, O. Edgar, G. Alejandro, and M. Sandra, "Identifying findings for software process improvement in SMEs: An experience," in *Ninth Electronics, Robotics and Automotive Mechanics Conference (CERMA)*. IEEE, 2012, pp. 141–146.

[99] F.J. Pino, F. Garcia, and M. Piattini, "Key processes to start software process improvement in small companies," in *Proceedings of the 2009 ACM symposium on Applied Computing*. ACM, 2009, pp. 509–516.

[100] R. Chalmeta, S. Palomero, and M. Matilla, "Methodology to develop a performance measurement system in small and medium-sized enterprises," *International Journal of Computer Integrated Manufacturing*, Vol. 25, No. 8, 2012, pp. 716–740.

[101] M. Lepmets and T. McBride, "Process improvement for the small and agile," in *European Conference on Software Process Improvement*. Springer, 2012, pp. 310–318.

[102] D. Homchuenchom, C. Piyabunditkul, H. Lichter, and T. Anwar, "SPIALS: A light-weight software process improvement self-assessment tool," in *5th Malaysian Conference in Software Engineering (MySEC)*. IEEE, 2011, pp. 195–199.

Appendix List of selected studies

Paper ID	Title	Empirical method	Year
S1 [7]	M. Díaz-Ley, F. García, and M. Piattini, "Implementing a software measurement program in small and medium enterprises: A suitable framework," *IET Software*, Vol. 2, No. 5, 2008, pp. 417–436.	Case study	2008
S2 [8]	M. Díaz-Ley, F. García, and M. Piattini, "MIS-PyME software measurement capability maturity model–supporting the definition of software measurement programs and capability determination," *Advances in Engineering Software*, Vol. 41, No. 10, 2010, pp. 1223–1237.	Case study	2010
S3 [75]	M. Díaz-Ley, F. García, and M. Piattini, "Implementing software measurement programs in non mature small settings," *Software Process and Product Measurement*, 2008, pp. 154–167.	Industry report	2008
S4 [76]	M. Díaz-Ley, F. García, and M. Piattini, "Software measurement programs in SMEs–defining software indicators: A methodological framework," *Product-Focused Software Process Improvement*, 2007, pp. 247–261.	Industry report	2007
S5 [77]	M. Diaz-Ley, F. García, and M. Piattini, "MIS-PyME software measurement maturity model-supporting the definition of software measurement programs," *Product-Focused Software Process Improvement*, 2008, pp. 19–33.	Case study	2008
S6 [78]	A. Tosun, A. Bener, and B. Turhan, "Implementation of a software quality improvement project in an SME: A before and after comparison," in *35th Euromicro Conference on Software Engineering and Advanced Applications*. IEEE, 2009, pp. 203–209.	Industry report	2009
S7 [79]	E. Amrina and S.M. Yusof, "A proposed manufacturing performance measures for small and medium-sized enterprises (SMEs)," in *Proceedings of the 10th Asia Pacific Industrial Engineering and Management Systems (APIEMS) Conference*, 2009, pp. 623–629.	Survey	2009
S8 [54]	H. Abushama, M. Ramachandran, and P. Allen, *PRISMS: an approach to software process improvement for small to medium enterprises*. UOFK, 2016.	Survey	2016
S9 [80]	J.A.C.M. Villalón, G.C. Agustín, T.S.F. Gilabert, A.D.A. Seco, L.G. Sánchez, and M.P. Cota, "Experiences in the application of software process improvement in SMEs," *Software Quality Journal*, Vol. 10, No. 3, 2002, pp. 261–273.	Experiment	2002
S10 [81]	I. Richardson and K. Ryan, "Software process improvements in a very small company," *Software Quality Professional*, Vol. 3, No. 2, 2001, pp. 23–35.	Survey	2001
S11 [82]	F. McCaffery, M. Pikkarainen, and I. Richardson, "AHAA–Agile, hybrid assessment method for automotive, safety critical SMEs," in *Proceedings of the 30th International Conference on Software Engineering*. ACM, 2008, pp. 551–560.	Industry report	2008
S12 [83]	F.T. Shah, S. Shamail, and N. Ahmad Akhtar, "Lean quality improvement model for quality practices in software industry in Pakistan," *Journal of Software: Evolution and Process*, Vol. 27, No. 4, 2015, pp. 237–254.	Survey	2015
S13 [84]	S. Bibi, I. Stamelos, G. Gerolimos, and V. Kollias, "BBN based approach for improving the software development process of an SME – A case study," *Journal of Software: Evolution and Process*, Vol. 22, No. 2, 2010.	Case study	2010

Paper ID	Title	Empirical method	Year
S14 [85]	P. Clarke and R.V. OíConnor, "The meaning of success for software SMEs: An holistic scorecard based approach," in *European Conference on Software Process Improvement*. Springer, 2011, pp. 72–83.	Survey	2011
S15 [86]	E. Caballero, J.A. Calvo-Manzano, and T. San Feliu, "Introducing scrum in a very small enterprise: A productivity and quality analysis," *Systems, Software and Service Process Improvement*, 2011, pp. 215–224.	Experiment	2011
S16 [87]	C.G. von Wangenheim, T. Punter, and A. Anacleto, "Software measurement for small and medium enterprises," in *Proceeding 7th International Conference on Empirical Assessment in Software Engineering (EASE)*, 2003.	Experiment	2003
S17 [47]	O.T. Pusatli, "Software measurement activities in small and medium enterprises: An empirical assessment," *Acta Polytechnica Hungarica*, Vol. 8, No. 5, 2011, pp. 21–42.	Survey	2011
S18 [88]	M. Sulayman, C. Urquhart, E. Mendes, and S. Seidel, "Software process improvement success factors for small and medium Web companies: A qualitative study," *Information and Software Technology*, Vol. 54, No. 5, 2012, pp. 479–500.	Case study	2012
S19 [89]	N. Ohsugi, K. Fushida, N. Inoguchi, H. Arai, H. Yamanaka, T. Niwa, M. Fujinuki, M. Tomura, and T. Kitani, "Using trac for empirical data collection and analysis in developing small and medium-sized enterprise systems," in *Empirical Software Engineering and Measurement (ESEM), 2015 ACM/IEEE International Symposium on*. IEEE, 2015, pp. 1–9.	Case study	2015
S20 [70]	C. Gencel, K. Petersen, A.A. Mughal, and M.I. Iqbal, "A decision support framework for metrics selection in goal-based measurement programs: GQM-DSFMS," *Journal of Systems and Software*, Vol. 86, No. 12, 2013, pp. 3091–3108.	Case study	2013
S21 [90]	H.M. Haddad and D.E. Meredith, "Instituting software metrics in small organizations: A practical approach," in *Information Technology: New Generations (ITNG), 2011 Eighth International Conference on*. IEEE, 2011, pp. 227–232.	Industry report	2011
S22 [9]	A.M. Bhatti, H.M. Abdullah, and C. Gencel, "A model for selecting an optimum set of measures in software organizations," in *European Conference on Software Process Improvement*. Springer, 2009, pp. 44–56.	Survey	2009
S23 [69]	T. Tahir and C. Gencel, "A structured goal based measurement framework enabling traceability and prioritization," in *Emerging Technologies (ICET), 2010 6th International Conference on*. IEEE, 2010, pp. 282–286.	Case study	2010
S24 [91]	A. Potter, P. Childerhouse, R. Banomyong, and N. Supatn, "Developing a supply chain performance tool for smes in thailand," *Supply Chain Management: An International Journal*, Vol. 16, No. 1, 2011, pp. 20–31.	Survey	2011
S25 [92]	A. Janes, V. Lenarduzzi, and A.C. Stan, "A continuous software quality monitoring approach for small and medium enterprises," in *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering Companion*. ACM, 2017, pp. 97–100.	Case study	2017

Paper ID	Title	Empirical method	Year
S26 [93]	H. Ayed, N. Habra, and B. Vanderose, "AM-QuICk: A measurement-based framework for Agile methods customisation," in *Software Measurement and the 2013 Eighth International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2013*. IEEE, 2013, pp. 71–80.	Case study	2013
S27 [94]	S. Suwanya and W. Kurutach, "Applying agility framework in small and medium enterprises," *Advances in Software Engineering*, 2009, pp. 102–110.	Case study	2009
S28 [95]	E. Irrazabal, F. Vásquez, R. Díaz, and J. Garzás, "Applying ISO/IEC 12207:2008 with SCRUM and Agile methods," *Software Process Improvement and Capability Determination*, 2011, pp. 169–180.	Case study	2011
S29 [96]	M. Díaz, F. García, and M. Piattini, "Defining, performing and maintaining software measurement programs: State of the art," in *IV Simposio Internacional de Sistemas de Informacióne*, 2006, p. 13.	Survey	2006
S30 [97]	J.C. Ruiz, Z.B. Osorio, J. Mejia, M. Muñoz, A.M. Ch, B.A. Olivares *et al.*, "Definition of a hybrid measurement process for the models ISO/IEC 15504 – ISO/IEC 12207:2008 and CMMI Dev 1.3 in SMEs," in *Electronics, Robotics and Automotive Mechanics Conference (CERMA)*. IEEE, 2011, pp. 421–426.	Case study	2011
S31 [98]	M. Jezreel, M. Mirna, N. Pablo, O. Edgar, G. Alejandro, and M. Sandra, "Identifying findings for software process improvement in SMEs: An experience," in *Ninth Electronics, Robotics and Automotive Mechanics Conference (CERMA)*. IEEE, 2012, pp. 141–146.	Case study	2012
S32 [99]	F.J. Pino, F. Garcia, and M. Piattini, "Key processes to start software process improvement in small companies," in *Proceedings of the 2009 ACM symposium on Applied Computing*. ACM, 2009, pp. 509–516.	Case study	2009
S33 [100]	R. Chalmeta, S. Palomero, and M. Matilla, "Methodology to develop a performance measurement system in small and medium-sized enterprises," *International Journal of Computer Integrated Manufacturing*, Vol. 25, No. 8, 2012, pp. 716–740.	Case study	2012
S34 [101]	M. Lepmets and T. McBride, "Process improvement for the small and agile," in *European Conference on Software Process Improvement*. Springer, 2012, pp. 310–318.	Case study	2012
S35 [102]	D. Homchuenchom, C. Piyabunditkul, H. Lichter, and T. Anwar, "SPIALS: A light-weight software process improvement self-assessment tool," in *5th Malaysian Conference in Software Engineering (MySEC)*. IEEE, 2011, pp. 195–199.	Case study	2012