IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

An Evaluation of Software Measurement Processes in Pakistani Software Industry

Touseef Tahir[1], Ghulam Rasool[1], Waqar Mehmood[2], Cigdem Gencel[3]
[1]COMSATS University Islamabad, Lahore Campus, Pakistan
[2]COMSATS University Islamabad, Wah Campus, Pakistan
[3]Free University of Bozen-Bolzano, Italy

ABSTRACT Implementing a successful measurement process is a challenging task. Most measurement studies report measurement models, experiences, and lessons learned based on pilot projects or case studies in a limited scope to overcome specific challenges. This study identifies 14 basic measurement practices and proposes a model of 18 success factors for implementing measurement processes with respect to the identified measurement theories in our systematic literature review (SLR) i.e., "A systematic literature review on software measurement programs", by Tahir et al., 2016. In addition**,** a survey is conducted to evaluate the state of measurement practices and to validate the proposed model based on the feedback from 200 software professionals working in Pakistani software industry. The state of measurement practices in the industry is mostly not according to the identified measurement theories in the SLR. For instance, more than 50 measurement models reported in the literature but only 10% software organizations follow any measurement model. 75% of organizations do not follow any measurement standard. 80% software organizations do not use any measurement tool. The proposed model is validated by applying structural equation modeling on the survey data. Furthermore, among 18 success factors, it is statistically significant that Pakistani software professionals strongly believe in necessity of 3 factors for successful implementation of a measurement process i.e., synchronization between measurement process and software process improvement, use of measurement standards and use of measurement models. In addition, they also believe that a successful measurement process will improve prediction, monitoring and management of software projects, and support in achievement of individual and organization-wide objectives. Software organizations might consider this study in planning and improving their measurement processes.

INDEX TERMS Software Measurement, Software Measurement Program, Software Metrics, Software Measurement Survey.

1. INTRODUCTION

The use of measurement is common in all aspects of human life such as scales for buying vegetables, fruits, grocery items, gold or making estimations for reaching to the home of a friend. The applications of measurement are obvious in all disciplines such as sciences, engineering, medical and others. The measurements play a vital role in software organizations for the improvement of software processes [1], [2]. It is reported by many software quality assurance agencies that the quality of software product is strongly influenced by the quality of software process followed to develop the software product [3]. Software measurement has become an essential process in software organizations for characterizing, evaluating, predicting and improving software products, processes, and resources [4]–[6]. ISO/IEC 15939:2007 [4] defines measurement process as "a process for establishing, planning, performing and evaluating measurement within an overall project, enterprise or organizational measurement structure". Despite the acknowledged importance and advocated benefits of measurement processes for software engineering community, their actual applications in the software industry are limited [4], [5], [8], [9]. It is also reported in a systematic review on measurements in software engineering that software process is the least measured entity [35].

There are as high as 70% software organizations which are not implementing software measurement processes in spite of all the reported benefits [8], [10], [11]. In addition, the studies on implementing measurement processes have reported various challenges [2]. Popović & Bojić [12] have reported that most of the measurement processes are implemented as an additional process which lacks a systematic structure. Díaz-Ley et al. [13] reported

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

that implementation of measurement process can fail because software organizations find it difficult to collect required data. It is reported that 80% of the measurement processes fail due to indecision in what should be measured, how to use measurement data in decision making and lack of awareness regarding fundamental concepts of software measurement [14] [15]). The lack of awareness impacts their decisions of not implementing measurement process or adopting it inappropriately [11], [16]. Most of the failed measurement programs lack long-term sustainability, clear objectives, correct measurement instruments, resources, time and budget [4], [5], [9].

There are a number of published studies which focused on improving different aspects of software measurement for successful implementation of measurement processes by proposing measurement models, tools, theories, guidelines, and practices, to implement software measurement processes in software organizations [1], [5]. For instance, sustainability of measurement process [16]–[21], transition to measurement culture [21], [22], use of measurement standards [4], [5], [7], synchronization between measurement process and business strategies [23]–[25]. Although, the significance of measuring and evaluating the outcome of software measurement programs is paramount, but still there exist no standard guidelines and models of success factors that may be generalized for all type of software products and software development organizations.

Most software organizations initiate measurement programs (MPs) as pilot projects to plan and establish their measurement processes [5], [26]. The aim of MPs is to evaluate the measurement processes for their benefits and improvements in software processes and products [4], [5], [26]. However, sustaining the MPs organization-wide for all/multiple projects and gauging their actual impact to characterize, evaluate, improve and predict software processes, products and resources remain challenging [4], [5], [26]. Moreover, there are many organizations which implicitly use the data generated through their software processes (e.g., source lines of code, function points, number of defects, number of defects fixed, time spent on specification/design/code/test etc.) for project management purposes even though they do not have a defined measurement process. Unterkalmsteiner et al. [4] and Touseef et al. [5] reported that the majority of MP studies lack the description of context in which they were conducted. The context description includes organizational context (e.g., size of organization, measurement scope, and project/organization level of measurement), stakeholders of measurement datasets, measurement process details (e.g., metrics collection and analysis methods, business goals, measurement goals, source/timestamp of measurement, and artifacts used for measurement). Due to lack of these details, it is very difficult to generalize the state measurement practices in the software industry. Therefore, based on the findings and analysis of our systematic literature review (SLR) on MPs [5] and the results of this survey, this paper has the following contributions:

- The analysis and evaluation of previously identified 3 key components and 14 sub-components of MPs that should be considered while planning and evaluating MPs.
- A model of comprehensive success factors is proposed that should be considered while planning a MP.

A questionnaire containing 45 questions was sent to 200 software professionals working in Pakistani software development organizations to evaluate and validate the proposed model of success factors. It consists of three types of questions:

- Demographic information about the participant and his/her organization.
- Evaluation of the state of measurement processes by asking specific questions that are based on identified measurement components.
- Validation of the model of success factors based on the opinions of participants on likert-scale regarding the importance of success factors for the implementation of a measurement process. The structural equation modeling is applied to the responses to identify statistically significant factors.

Table 1 presents research questions of this study.

TABLE 1. RESEARCH QUESTIONS

ID	Research Question	Motivation
RQ1	What is the state of measurement practices in software organizations in Pakistan?	To identify, analyze, classify and evaluate standard practices and components of software measurement programs in software organizations.
RQ2	What are significant success factors for implementing measurement programs according to software professionals?	To identify key success factors and validate the model of success factors for software measurement programs.

This paper has two main contributions for researchers and practitioners: 1) to provide basic components of a MP which should be considered while planning a MP and evaluating the state of measurement practices in Pakistani software industry with respect to the identified components, 2) to support in planning a successful MP by proposing a model of success factors based on synthesizing and consolidating reported success factors in SLR [2] and later identifying statistically significant success factors for Pakistani software organizations. The practitioners and researchers can further evaluate, extend and refine the measurement components and the model presented in this paper.

This paper is organized as follows. Section 2 presents related work. We build background and motivation for the survey as a whole and theoretical model of success factors in Section 3. Section 4 elaborates research methodology and Section 5 describes survey results and analysis. Section 6 presents discussions in terms of a summary of findings and mitigation strategies to key issues, threats to validity and managerial implications and future work of this study. Section 6 presents conclusions.

2. RELATED WORK

Soini [8] presented a survey to investigate the use of software measurement in the Finish software industry. The problems, challenges, limitations are evaluated. There is an overlap between the metrics used for measuring processes and products and most of the process metrics are used for product measurement too which can affect the effectiveness of decision making. Most of the metrics are extracted from the data that was not collected with the intention/context of measurement. Metrics are not collected for their intended use at some management level e.g., middle or upper management. Most of the metrics are lagging (calculated at the end of a process) but there is a strong need of leading metrics (real-time) that can support effective monitoring and controlling. The metrics selection method is based on the information system available for project management in software organizations, or metrics are extracted from already collected data. The metrics are mostly collected to characterize and evaluate while they are rarely used for prediction and improvement of software processes, products, and resources. Most of the metrics are collected and analyzed regarding processes and products entities while resource metrics are least collected and analyzed. We have also evaluated related issues in Pakistani Software industry including purposes of measurement processes, types of metrics, real-time metrics, scope of measurement process, availability of measurement tools, and defined measurement process (Section 3.1).

The Software Engineering Institute (SEI) conducted a survey consisting of 17 questions with the help of random sampling [9]. The implementations of measurement processes are investigated with respect to participant beliefs and organization type. It is observed that administration and staff have different opinions regarding measurement process. The administration believed more strongly that they understand the purpose of the measurement process and it should be documented, measurements should be consistently defined and indicators and thresholds set on measurements should be used for corrective actions and decision making and it helps to improve team performance. The larger organizations (size >= 500 employees) had better measurement infrastructure. Only 40% of respondents believed that corrective actions are taken based on measurement indicators while 20% believed corrective action is rarely or never taken. The use of CMMI guidelines for identification, collection, and analysis are believed to be used by 56% of respondents. There were 21% respondents who believed to have no measurement method for identification, collection, and analysis of software metrics. Most respondents (97%) believed that tracking scheduled progress is most reported metric while 93% respondents believed that effort is the most reported metric. In our study, we have also investigated similar issues and also state of the art issues of measurement processes such as sustainability of MP, use of measurement models and real-time metrics etc (Section 3.1) in different organization types. The organization types in [9] are defined as department of defense, commercial and government. In our study, we have defined organization types as start-up, SME, large, and very large, because we have observed in a systematic mapping study on SMEs that MPs in small organizations face more challenges of time, budget, and resources as compare to large organizations [27].

Gopal et al. [21] proposed a theoretical model to evaluate the impact of institutional forces and management commitment on successful adaptation of measurement process in an organization. They also analyzed the impact of successful adaptation of measurement process on decision making. They conducted a survey among 214 metrics managers of software organizations and applied structural equation modeling (SEM) on responses. It is observed that institutional forces, metrics adaptation, and management commitment are essential for successful implementation of measurement process. This model is proposed based on experience of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

implementing MPs and it is also validated in software organizations having already defined MPs. The responses are collected regarding frequency of adapting measurement processes. On the other hand, our model of success factors is proposed based on SLR on MPs. It contains three success factors for implementing a measurement process i.e., external factors, measurement adaptation factors and measurement acceptance factors. In our model of success factors, we have adapted external factors and measurement acceptance factors from the model of Gopal et al. [21]. They defined three possible dimensions of measurement acceptance i.e., 1) frequency of adapting measurement process, 2) mental attitude towards impact of adapting a measurement process and 3) intention to adapt measurement process [21]. We applied SEM on the responses of Pakistani software professionals regarding their mental attitude towards the importance of external factors and measurement adaptation factors for successful implementation of MP and impacts of measurement adaptation towards measurement acceptance factors.

Dyba [28] presented an empirical study to investigate major success factors that may be used to evaluate software process improvement (SPI) using survey method. The proposed model comprises on six independent, one dependent and two contextual variables. A questionnaire with 47 questions related to independent, dependent and contextual variables was sent to different software organizations. A total of 120 respondents from 55 software organizations returned the filled questionnaire. Based on the feedback of respondents, the author presented a theoretical model to analyze major success factors and they succeeded to provide more insight on influence of organizational issues by empirically showing that they are equally essential similar to technology. He concluded that evaluating success factors for SPI depends on organizational factors and organization culture plays a paramount role in this regard. The RQ answered in this study focuses on the success factors of implementing SPI rather than discussing success factors of implementing MPs. Our study has proposed and validated a model of success factors for implementing MPs.

This study differs from the above surveys in two perspectives. First, the above surveys are mainly based on the experience of implementing MPs while this study first identified 14 basic measurement practices based on a SLR on software MPs (1997-2015) [2] and later analyzed the frequencies of these practices in software organizations. Moreover, a model of success factors is also proposed based on the SLR and validated using SEM. Second, we checked Google Scholar, Wiley Interscience, Science Direct Journals, Springer, One Search, ACM, and IEEE for similar studies in *Pakistani software industry*. To the best of our knowledge, there is no similar study previously conducted according to research questions in Table 1.

3. BACKGROUND OF THEORETICAL MODEL

We stem motivation and build background for survey and theoretical model of success factors from our concrete findings highlighted in SLR [2] in this section.

3.1. Background

In our previous research, we conducted a SLR on software MPs [5]. We identified 35 different measurement-planning models and 11 associated tools while answering the following research questions.
"RQ1: Which measurement planning models, tools, and practices are discussed in the literature?"
"RQ2: Which techniques/methods/models are developed for metrics selection when implementing MPs?"
"RQ3: What are the success and failure factors for MPs implementation?"
"RQ4: Which mitigation strategies are discussed for MP implementation?"

We found that there is no clear structure of measurement programs that are reported in primary studies of SLR. Therefore, in this paper, we identified key components of a measurement program that should be considered while planning a measurement program. We have analyzed the state of measurement practices based on these key components. Table 2 presents key components, sub-components and description of components that are analyzed in the software development industry as shown in Appendix A. We have analyzed these components based on descriptive statistics (e.g., percentages and averages) with respect to the size of software organizations. We divided organizations into four categories, such as start-ups, SMEs (less than 250 employees), large (250 -500 employees) and very large (greater than 500 employees).

3.2. MODEL OF SUCCESS FACTORS

Among the primary studies of SLR, we identified 18 success factors for implementing MPs. In this paper, we have mapped these success factors among three main factors and propose a model for implementing a MP. The main factors include external, measurement adaptation, and measurement acceptance factors.

Later, we have conducted a survey to identify statistically significant success factors of the model with respect to beliefs of software professionals in the Pakistani software industry. The structural equation modeling is a multivariate statistical analysis technique that is used to empirically validate the model based on the survey data. A five points likert-scale is used to collect response of participants i.e., strongly agree (5), Agree (4), Neutral (3), Disagree (2), and Strongly disagree (1).

The successful implementation of a MP is mainly affected by external factors and the internal (measurement adaptation) factors. First, this model presents external and measurement adaptation factors that should be considered while planning a MP in a software organization. Second, this model presents measurement acceptance factors to evaluate the successful implementation of MP.

Figure 1. Proposed Abstract Theoretical Model

3.2.1. EXTERNAL FACTORS

The external factors/constructs that may impact the successful implementation of a measurement process consists of 3 sub-factors (EF1-EF3). Table 3 presents a brief description of each sub-factor as shown in Appendix B. The external environment might also strongly influence the software organizations because they need to comply with certain industry-wide practices and policies in order to gain legitimacy in the market [21]. In this survey, we are measuring the participant's beliefs regarding the influence of external factors on the successful implementation of a measurement process.

3.2.2. MEASUREMENT ADAPTATION FACTORS

The measurement adaptation factors consist of 11 sub-factors (MAD1-MAD11). Table 4 presents a brief description of each sub-factor as shown in Appendix B.

3.2.3. MEASUREMENT ACCEPTANCE

In a study, there are three possible dimensions of measuring measurement acceptance i.e. 1) mental attitude towards the impact of adapting a measurement process, 2) intention to adapt measurement process and 3) degree of adapting measurement process [21]. In our context, we are measuring measurement acceptance factor in terms of mental attitude (beliefs) towards the impacts of measurement adaptation. The measurement acceptance construct is measured in terms of four sub-factors. Table 5 presents a brief description of each sub-factor as shown in Appendix B.

4. RESEARCH METHODOLOGY

Below, we discuss the details of the survey design, data collection, and analysis procedures. We have used the guidelines of the software Engineering Institute (SEI) for conducting survey specifically in software organizations [29].

4.1. SURVEY DESIGN

We designed a questionnaire[1] that consists of three types of questions: a) demographic information about the participant and his/her organization (Section-1 of survey), b) specific information about the state of measurement process implementation in the participant's organization (Section-2 of survey), c) opinions of participants regarding the importance of success factors for measurement process implementation (Section-3 of survey). In total, the survey contains 45 questions.

The survey was sent to 542 software professionals. We made the survey available online on the web for three months. In total, we have collected 200 responses. We made the design of the survey to ensure that the respondents provided answers to all necessary questions in a section. For example, the respondents could not continue to the next section of the survey if necessary questions in a current section were left blank.

The opinions of the participants regarding success factors of measurement process implementation (Section-3 of survey) are analyzed using Structural equation modeling (SEM). It is a multivariate statistical analysis method that is effectively used to evaluate the structural relationships between variables [30]. The minimum dataset size for applying SEM is 200 [30].

The compulsory questions were asked as closed questions while optional questions were asked as open-ended questions to get further information. For example, a close-ended question is followed by an open ended question:

- Does your organization use any measurement model? (Yes, No).
- If yes, does your organization use any specific measurement model (e.g., Goal Question Metric)? (Open Ended).

[1] The questionnaire of the survey is available at: https://goo.gl/forms/9kvnVh7E6ir617e72

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

Most of the questions were multiple-choice questions. We prepared the choices considering the terms used in well-known standards or commonly used terms in software industry to avoid ambiguity. For example, ISO 12182 software classifications are used for classification of IT systems and software development domains. Similarly, we used the term "measurement process" instead of "measurement program" in our questionnaire as many organizations may not have implemented a specific measurement program but yet have been performing a measurement process.

For getting answers to opinion questions, we used a five-point likert-scale. Likert scale is a psychometric scale, which is the most commonly used scale in questionnaire and survey designs. It is used when respondents are asked to rank their opinion about something such as customer are asked to rank their opinion about a product quality from high to low or best to worst using five or seven levels [31]. The five-point likert-scale used in this survey included opinions ordered such as following:

- Always (5), Generally (4), Sometimes (3), Seldom (2), and Never (1)
- Strongly Agree (5), Agree (4), Neutral (3), Disagree (2), Strongly disagree (1)

4.2. DATA COLLECTION

We used a survey facility of Google Forms to design the questionnaire. The survey was available online for three months. When distributing the link of the survey, we decided to make purposive homogeneous sampling, which is a non-probabilistic sampling technique [32], [33]. It is suitable to collect a homogeneous sample that contains instances of similar characteristics and traits. This technique is suitable for our study, as we have collected data from software professional working in software development organizations. It was explicitly described to the participants to fill up the survey only if they were part of a measurement process or had observations of it in their respective organization.

4.3. DATA ANALYSIS

First, we first transferred the collected data to MS Excel and transformed the likert-scale answers into ordinal data. Second, we transformed data to Statistical Package for the Social Sciences (SPSS[2]) and performed both descriptive and predictive statistics for analysis towards answering our research questions.

[2] Statistical Package for the Social Sciences

5. RESULTS AND ANALYSIS

The questionnaire consists of three types of questions:

1. Demographic information about the participant and his/her organization,
2. Specific information about the level of measurement process implementation in the participant’s organization,
3. Opinion questions on the three factors, which the participant considers as important for successful measurement process implementation.

In the following sections, we analyze demographic information about the participants (Section 5.1) and their organization (Section 5.2) and specific information about the level of the measurement process implementation in the their organizations (Section 5.3) and validation of the model of success factors of implementing MPs (Section 5.4).

5.1. PARTICIPANT'S DEMOGRAPHIC INFORMATION

Figure 2 presents a summary of the demographic data about the survey participants and their role in measurement process implementation in their organization.

More than 50% of participants in this survey have 1-3 years of experience and their job title as software engineers. The reason for this is that most university alumni working in software industry in Pakistan are contacted and a majority of them were fresh graduates having minimum 1-3 years of experience. A majority of the participants responded being observers of the measurement process.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

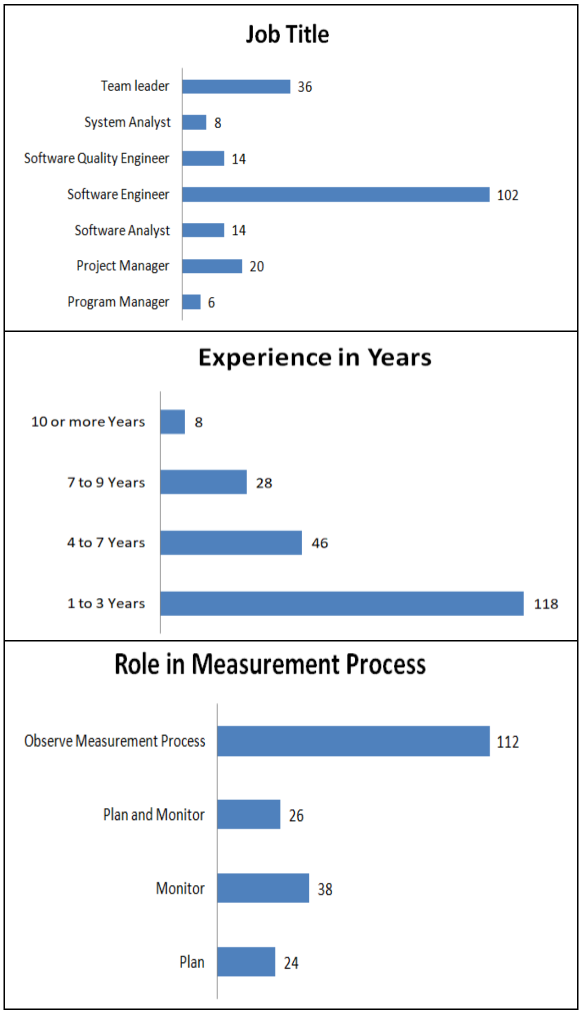

Figure 2. Demographics of Participants

5.2. DEMOGRAPHICS OF PARTICIPANT'S ORGANIZATION

Figure 3 presents demographics of software organizations. We divided organizations into four categories i.e., start-ups, small and medium enterprises (SMEs) (less than 250 employees), large (250 -500 employees) and very large (greater than 500 employees). According to small and medium enterprises development authority (SMEDA[3]), SMEs constitute nearly 90% of all the enterprises in Pakistan and their share in annual GDP is approximately 40%. According to [34], there are approximately 1500 registered IT organizations and approximately 10,000 Computer Science graduates enter IT market every year. The IT exports were 100 million in 2003 but exports have grown to US$2.2 billion in 2015, according to Pakistan Software Export Board (PSEB, www.pseb.org.pk). These figures indicate an evolution in Pakistani software development industry.

In our survey, 42% of the organizations were SMEs. These SMEs are further categorized as very-small (less than 25 employees), small (25-100 employees) and medium (100-250 employees), which constitute 15%, 57% and 28% of SMEs in our survey. There are 6% startups, 24% large and 28% very large organizations.

The list of domains for which software applications are developed in software industry is extracted from another survey [35]. The information technology (70%), professional and business services (44%) and financials (33%) are the most common domains of software application development in this survey.

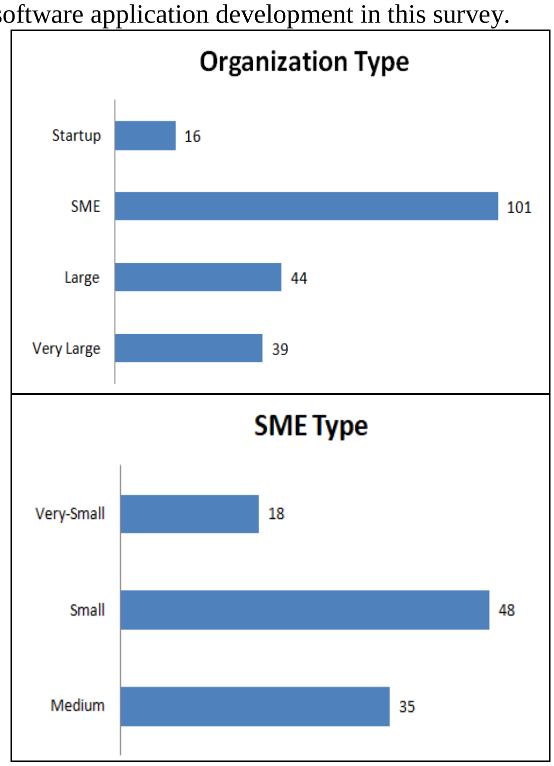

Figure 3. Demographics of Software Organizations

5.3. DEMOGRAPHICS OF MEASUREMENT PROCESSES

In this survey, we have evaluated the state of 3 key components of measurement programs in software development industry. The distribution of key components of a measurement program into measurement practices is depicted with the help of a work breakdown structure as shown in Figure 4. Each measurement practice is elaborated below.

[3] www.smeda.org

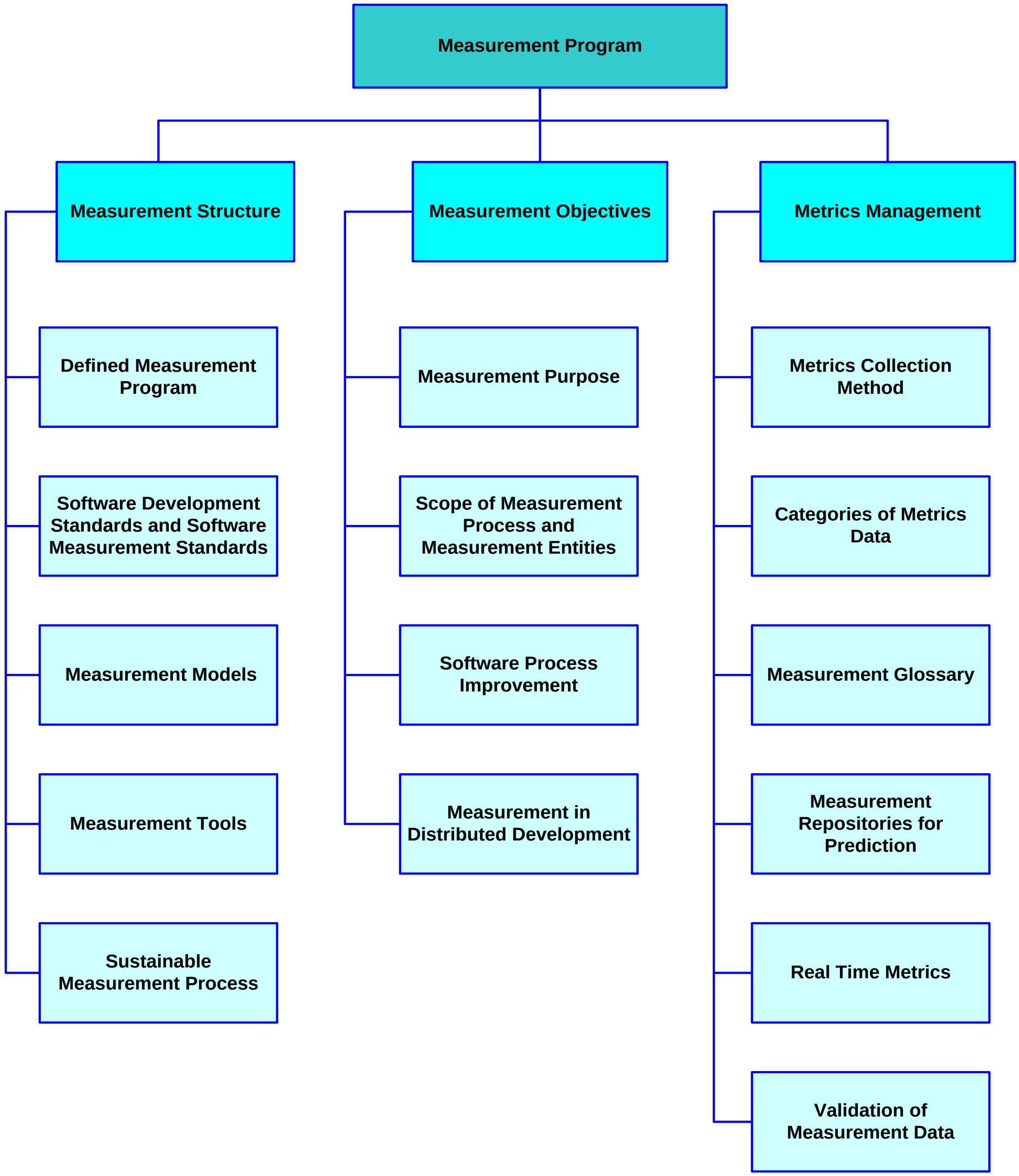

Figure 4. **Key Components of Measurement Programs Identified from SLR**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

5.3.1. MEASUREMENT STRUCTURE

A. DEFINED MEASUREMENT PROCESS

Most of the participants (61%) answered that there is no defined measurement process in their organization. The large (17%) and very large (18%) organizations have higher tendencies of defined measurement process than SMEs (4%) while none of the startup organizations in the survey have defined measurement process as shown in Figure 5.

ISO/IEC 15939:2007 [4] defines measurement process as "a process for establishing, planning, performing and evaluating measurement within an overall project, enterprise or organizational measurement structure". However, it is observed in the SLR [5], that there is no standard definition of software measurement process followed among reported measurement studies in software development organizations. However, organizations might define their measurement processes with respect to their needs and objectives.

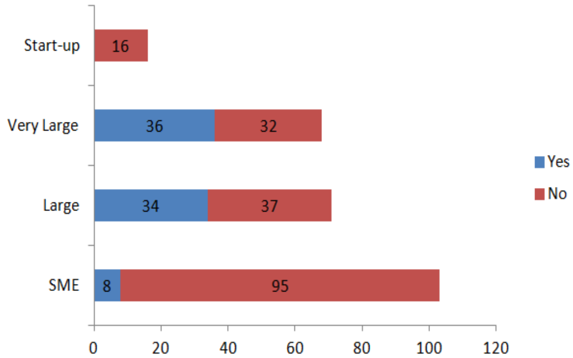

Figure 5. **Frequency of Defined Measurement Processes**

B. SOFTWARE DEVELOPMENT STANDARDS

We observed in the SLR, that software measurement studies discuss the use of few software development standards. Therefore, in this survey, a list of IEEE glossary of standards is presented. A checkbox with 'other' option and textbox is also given to identify mostly used standards. There are 41% organizations in the survey which do not follow any international standard. According to organization types, there are startups (1%), SMEs (47%), large (30%), very large (22%) organizations, which do not follow any software development standard as shown in Figure 6. ISO/IEC 12207, ISO/IEC 23026 (IEEE std 2001-2002) and ISO/IEC 15393-2002 are used by 18, 18, and 9% of the organizations. There are 5% responses with 'I don't know' which might mean that they are unaware about the standard used by their organization. The other standards mentioned by respondents include: Microsoft Gold Partner (1%), PCI standards (2%), ISO 9001:2008 (4%), ISO 27001 (3%), 3GPP and 5 GPP (1%), SOC2 (1%), ISO 26262 (1%), and Scrum (1%). The overall figures indicate lack of software development standards used among the software development organizations in this survey.

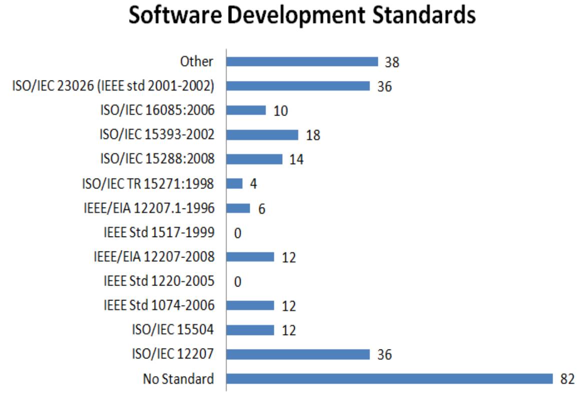

Figure 6. **Frequency of Adapted Software Development Standards**

C. SOFTWARE MEASUREMENT STANDARDS

In this survey, we presented IEEE glossary of measurement standards in a similar way of software development standards to indentify mostly used measurement standards.

In total, there are 75% of organizations which do not follow any measurement standard as shown in Figure 7. Among the remaining 25% organizations, SMEs (30%) have a lesser tendency to use measurement standards as compared to large (61%) and very large (60%) organizations. IEEE Std 982.1-2005 is used by highest (9%) number of the organizations. Overall, these figures indicate lack of measurement standards used among the software development organizations in this survey.

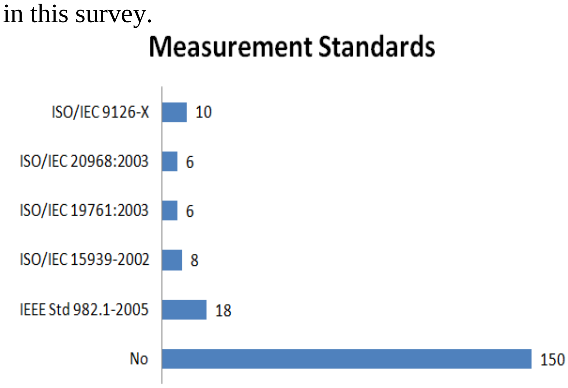

Figure 7. **Frequency of Adapted Measurement Standards**

D. MEASUREMENT MODELS

We have identified 35 measurement models and 11 tools being reported in the primary studies in the SLR [5]. Goal Question Metrics (GQM) model is considered as a base model for the majority of those 35 measurement models but in this survey, there are only 4% of the organizations using the GQM model.

There are 95% of the organizations which do not follow any measurement models as shown in Figure 8. This indicates lack of measurement models used among the software development organizations in this survey.

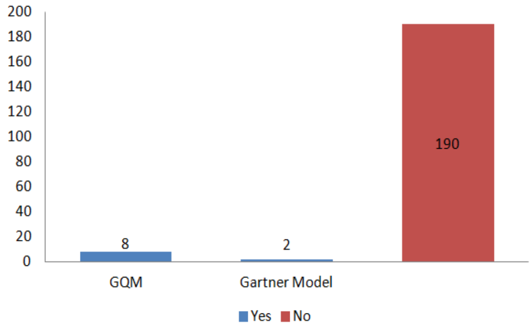

Figure 8. **Frequency of Adapted Measurement Models**

E. MEASUREMENT TOOLS

The automated tools play a very important role in all fields of software engineering including measurement programs but surprisingly, only 20% software organizations are using automated measurement tools according to results of this survey. The software organizations either develop their own tools or use commercial or open source tools available as shown in Figure 9. MS Excel is used by 12% of the organizations. According to organization type, the SMEs (16%), large (34%) and very large (50%) organizations practicing automated measurement tools. However, in this survey, 80% of the software organizations don't use any measurement tool. In the SLR, we could only identify 11 tools among 65 measurement studies. We cannot find any concrete reason for lack of interest of researchers and software organizations regarding applications of automated tools. *Therefore, we recommend that software organizations and researchers need to focus on the automation of measurement process with help of measurement tools.*

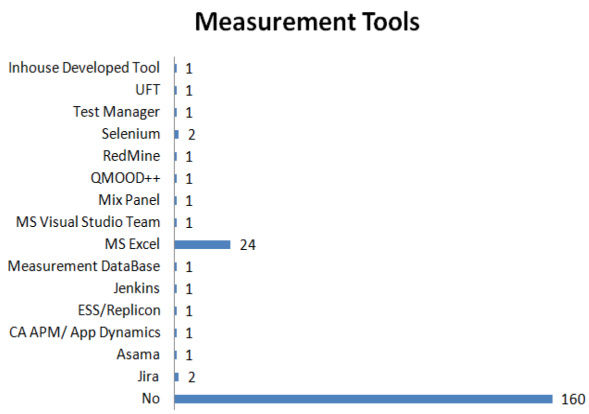

Figure 9. **Frequency of Adapted Measurement Tools**

F. SUSTAINABLE MEASUREMENT PROCESS

In this survey, the sustainability of measurement process is focused by only 6% of the organizations. In the SLR [5], it is also one of the least focused objectives of measurement studies. Irrespective of organizational-size, the sustainability is not mainly focused by all organizations as presented in Figure 10. There are 29% participants who were unaware about the presence/absence of sustainable measurement process in their organization. It reflects from results that sustainability of measurement processes requires the attention of software practitioners and software development organizations.

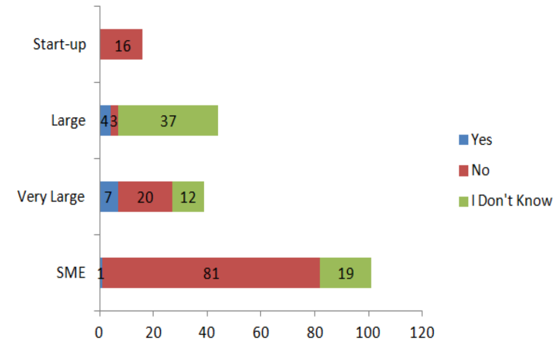

Figure 10. **Sustainability of Measurement Processes**

5.3.2. MEASUREMENT OBJECTIVES

A. MEASUREMENT PURPOSE

In the SLR [5], the distribution of measurement studies according to measurement purposes of the MP are as follows: Characterization (81%), evaluation (77%), improvement (70%) and prediction (28%). Figure 11 represents Venn diagrams to analyze the measurement purposes with respect to four types of software organizations. It is evident from the diagram that less than half of organizations e.g., large (39%), very large (29%), SMEs (20%) and start-ups (0%) use measurement process for all the four purposes. Characterization, evaluation, prediction, and improvement of software entities are the purpose of 53, 59, 52 and 61% of organizations. While measurement processes used for all four measurement purposes are in 26% organizations.

This indicates that measurement possesses implemented in software industry are not fully aligned with theoretical perspective of reported measurement studies which advocate that software measurement process should be used to achieve all four measurement purposes [5], [6].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

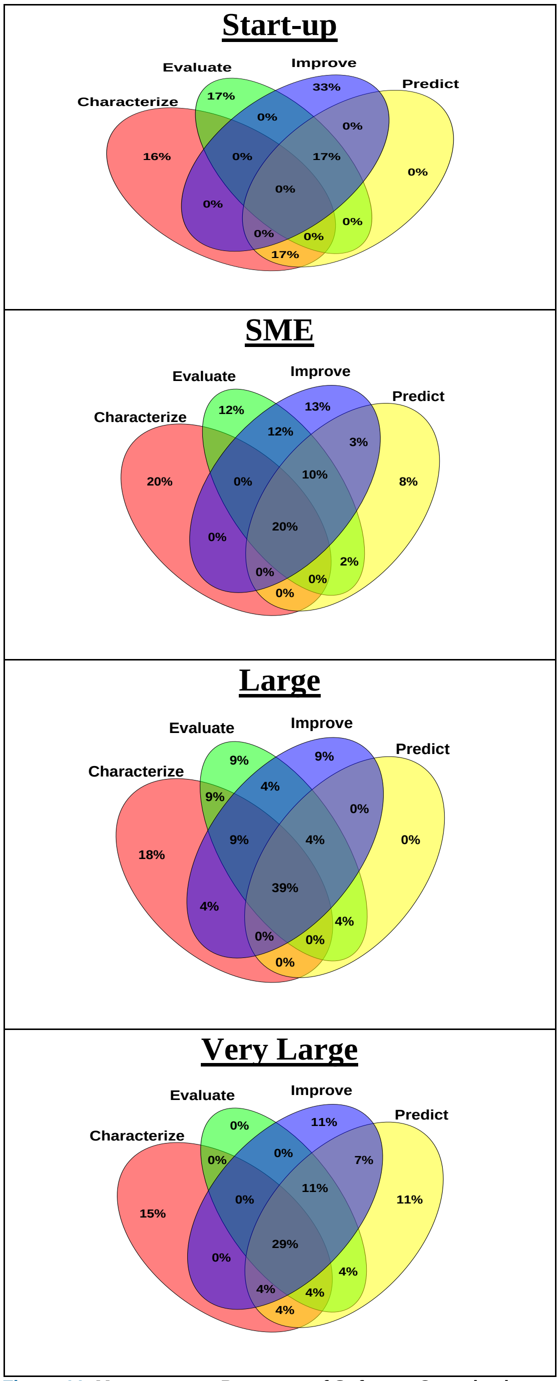

Figure 11. Measurement Purposes of Software Organizations

B. SCOPE OF MEASUREMENT PROCESS

The use of measurement process at project and organizational level simultaneously helps to plan, implement, monitor, control and improve software development processes, products and resources at each project-level and achievement of organizational goals such as return over investment, and employee satisfaction. In this survey, software measurement process is observed at 56, 14, and 30% of organizations at project, organization and simultaneously at both levels respectively. In the SLR [5], most of the primary studies discussed measurement programs at project-level, followed by organization-level and simultaneously at both levels respectively 58, 28 and 14%. *The results of this survey and SLR* [5] *illustrate that the scope of software measurement process in software organizations is mostly at project level.*

Figure 12 represents the scope of measurement processes in four types of software organizations with the help of Venn diagram. The large (39%) and very large (41%) organizations have higher tendency to implement measurement process at both levels as compared to SMEs (20%). On the other hand, SMEs (65%) have higher tendency to implement measurement process at project level as compared to large (47%) and very large (44%). Measurement process at sole organization level is least among SMEs (13%), large (14%) and very large (14%).

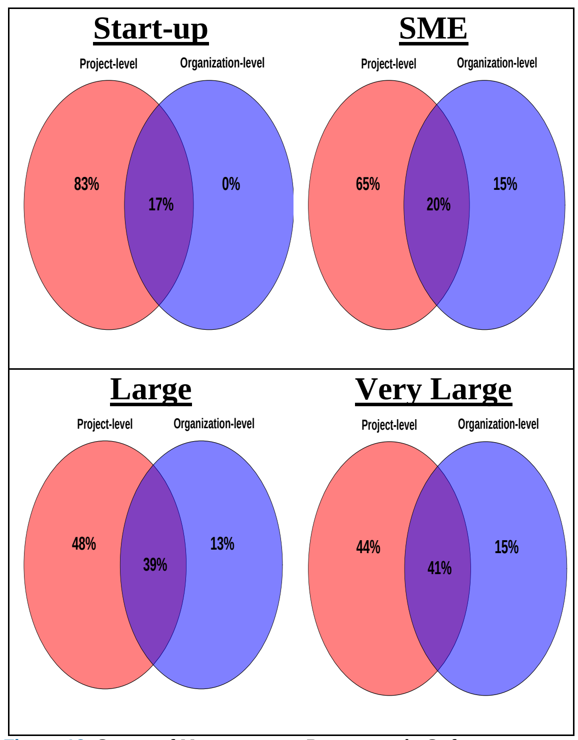

Figure 12. Scope of Measurement Processes in Software Organizations

C. MEASUREMENT ENTITIES

This survey identifies that processes, products, and resources are measured by 74%, 48% and 36% of organizations respectively. While all three measurement entities combined (i.e., processes, products and resources) is the focus of 23% organizations. In the SLR [5], software process, product and resources entities are measured by 96, 58, and 40% of empirical studies while all three combined are measured by 56% of studies. The results of this survey and the SLR illustrate that process entities followed by product and resource entities are mostly measured. The resource is a least measured entity in both of these studies. Figure 13 represents measurement of the software entities in four types of organizations with the help of Venn diagram. The collection of all three metrics by software organizations is less than 30% among the software organizations which might indicate lesser collection of data to effectively characterize, evaluate, predict and improve basic categories of software entities. The process metrics are collected more than twice as compared to product and resource categories which might indicate that focus of organizations is to manage their processes while management of products and resources is not generally focused.

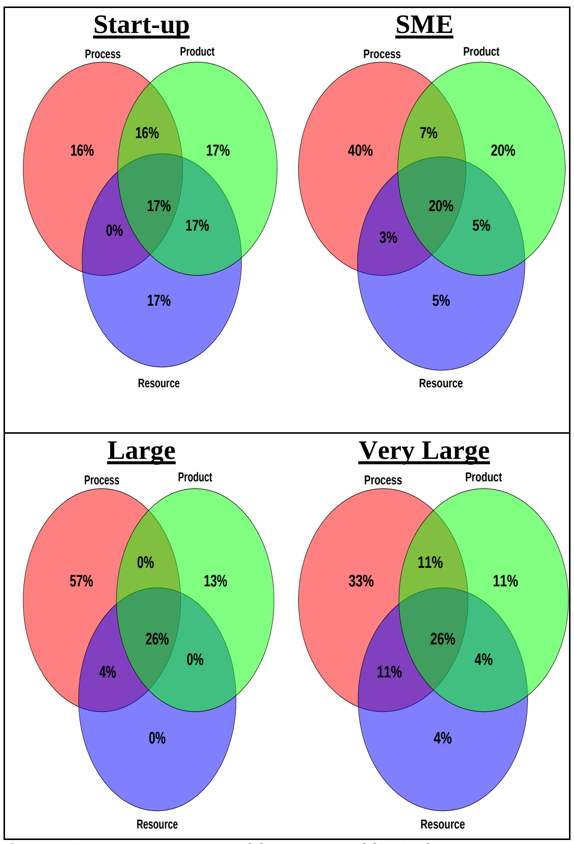

Figure 13. Measurement Entities Focused by Software Organizations

D. SOFTWARE PROCESS IMPROVEMENT

CMMI is identified as most widely used software process improvement (SPI) model in the SLR on software measurement programs [5] and SLR on evaluation and measurement of SPI [4]. In this survey, CMMI is used as software process improvement initiative by 29% organizations while 71% of organizations don't follow any software improvement methods, models or standards. Among the remaining 29% organizations, CMMI is used by SMEs (20%), large (47%) and very large (30%) organizations as shown in Figure 14.

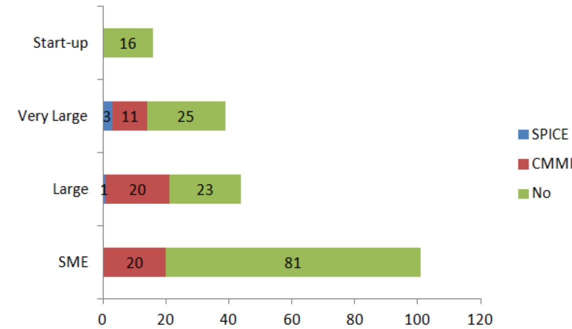

Figure 14. Frequency of Adapted Software Process Improvement Models

E. MEASUREMENT IN DISTRIBUTED DEVELOPMENT

In Pakistan, mostly software industry constructs software products using distributed development [34]. In this survey, 67% of organizations follow distributed development. These 67% of organizations are further categorized as near-shoring (19%), off-shoring (42%), outsourcing (33%), and on-shoring (6%) as shown in Figure 15. There are only 38% of participants who believe that their organizations have a defined measurement process for distributed software development.

The measurement process becomes even more important for distributed development because characterization, evaluation, prediction, and improvements of software processes, products and resources are generally based on data collected at a distant organization. However, in the SLR [5], there are less than 10% measurement studies in distributed development organizations.

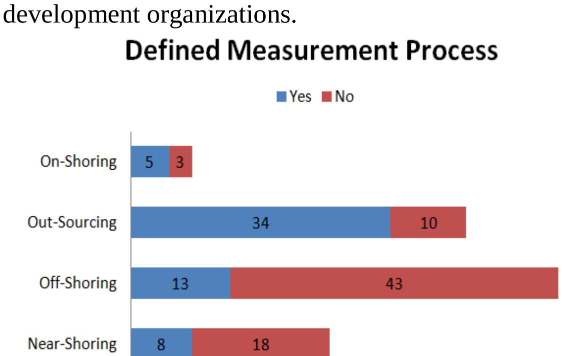

Figure 15. Software Measurement Processes in Distributed Development Organizations

5.3.3. METRICS MANAGEMENT

A. METRICS COLLECTION METHOD

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

In the SLR, we generalized metrics collection methods among six categories. The collected metrics are a key input for evaluation, prediction, characterization, and improvement of software development process. In this survey, majority of the organizations only collect most common metrics (23%) or necessary metrics (30%) for software process improvement. The SMEs (20%), large (22%) and very large (37%) organizations collect metrics that are necessary for software process improvement. The most common metrics are collected by SMEs (30%), large (30%) and very large (11%) organizations as shown in Figure 16. The use of measurement tools and expert judgment is used by 14% of organizations. The use of automated tools, measurement models and expert judgment is under 15% among the organizations. *The use of measurement models is least used method for metrics collection while on the other hand, there are 35 measurement models proposed in literature.*

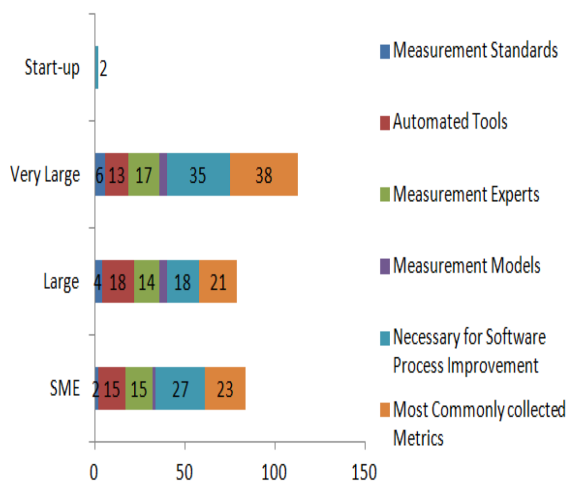

Figure 16. Frequency of Metrics Collection Methods in Software Organizations

B. CATEGORIES OF METRICS DATA

We have extracted common measurement attributes that are discussed in the studies on implementing measurement processes [5]. The actual use of the measurement attributes in the software industry is evaluated in this survey. The duration, cost, and effort are most measured attributes as shown in Figure 17. A text-box labeled "other" was also presented but no participant provided any other measurement attribute.

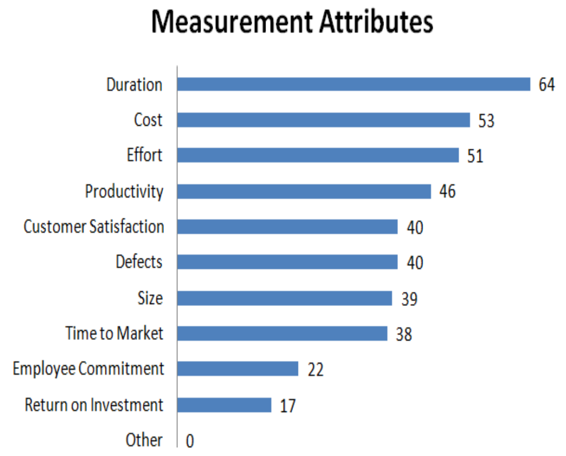

Figure 17. Frequency of Attributes Measured in Software Organizations

C. REAL-TIME METRICS

In SLR [5], 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 is observed to be least discussed. A survey in Finish software industry specifically reported lack of real-time metrics used during software development process. In this survey, we realized that 89% of organizations don't use real-time metrics as shown in Figure 18.

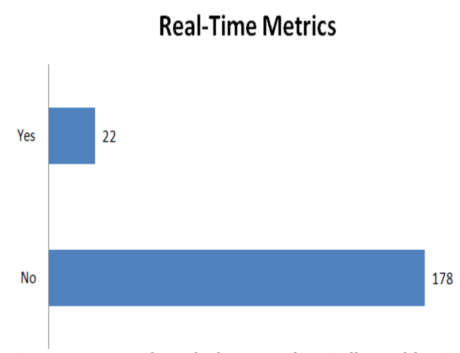

Figure 18. Frequency of Real-Time Metrics Collected in Software Organizations

D. MEASUREMENT GLOSSARY

The lack of usage of measurement glossary might cause ambiguities in understanding measurement objectives, tasks and process. A measurement glossary containing definitions of used metrics is maintained completely by 38% of the organizations, while 19% partially and 16% don't use measurement glossary at all. There are 28% respondents who are not aware of measurement glossary usage in their organization. A few SMEs completely (30%) or partially (18%) maintain measurement repository as shown in Figure 19. The large organizations completely (52%) or partially (22%) maintain a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

measurement repository. The large organizations completely (44%) or partially (11%) maintain a measurement repository.

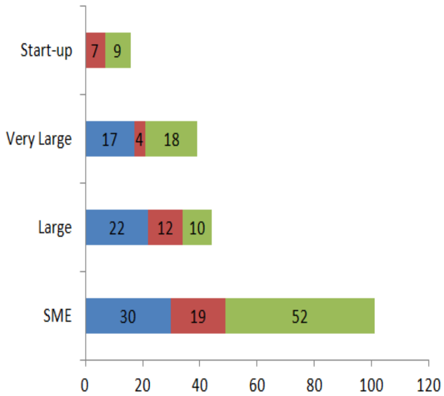

Figure 19. Frequency of Organizations Maintaining Measurement Glossary

E. MEASUREMENT REPOSITORIES FOR PREDICTION

It is observed in the SLR [5] that prediction is the least focused purpose of measurement processes in the empirical studies. In this survey, the degree of past data usage for predicting software attributes/entities (e.g., size, cost etc.) is measured on a likert-scale of 5 i.e. "Never", "Rarely", "Occasionally", "Frequently" and "Always". Most of the participants working in very large organizations believe that they frequently (40%), or occasionally (40%) use measurement data for prediction. While few believe that they rarely (12%) use measurement data for prediction. Similarly, most of the participants working in large organizations believe that they frequently (32%), or occasionally (48%) use measurement data for prediction as shown in Figure 20. A few participants believe they rarely (15%) or never (5%) use measurement data for prediction. Most of the participants working in SMEs believe that they frequently (45%), or occasionally (35%) use measurement data for prediction. A few believe that they rarely (15%) or never (5%) use measurement data for prediction. Overall, less than 50% of the organizations frequently use past data for prediction. None of the participants responded with "Always" on the scale.

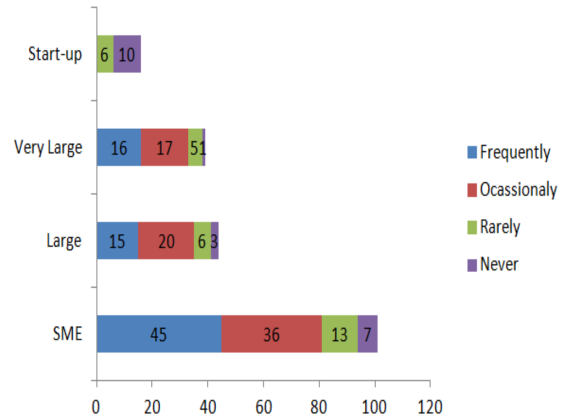

Figure 20. Frequencies of Organizations using Measurement Data for Prediction

F. VALIDATION OF MEASUREMENT DATA

It is important to learn that we are actually measuring what we think we are measuring. The metrics validation in software measurement processes is vital yet it is observed in the SLR [5] that it is least discussed issue in empirical studies. This survey also shows that 58% of participants believe that their organizations don't have any metrics validation while other 15% participants don't know if there is a metrics validation mechanism exists. While only 27% of organizations consider metrics validation as an important measure. A small number of SMEs (13%), large (6%) and very large (6%) organizations validate their metrics as shown in Figure 21. We have not collected any further details on validation methods as the purpose of this survey is to collect data on basic software measurement processes in software development organizations.

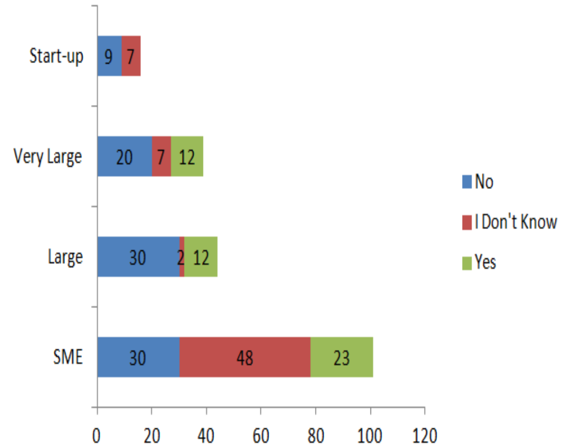

Figure 21. Frequency of Measurement Data Validation in Software Organizations

5.4. RESEARCH MODEL OF SUCCESS FACTORS

This section presents the results of applying SEM. It is a multivariate statistical analysis technique that is used to empirically validate the factors of successful

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

measurement process implementation based on practitioner's beliefs. The five-point likert-scale is used in this survey i.e., Strongly disagree (1), Disagree (2), Neutral (3), Agree (4), and Strongly agree (5).

We analyzed participant's opinion after proposing a theoretical model of 18 success factors for implementing MPs and identifying relationships among its constructs i.e., external forces, measurement adaptation, and measurement acceptance as shown in Figure 22. The relationships among constructs are validated based on hypothesis testing. Each of these constructs is measured using a distinct set of questionnaire items. Tables 3-5 present the questionnaire items in Appendix B.

The relationships among constructs of proposed models are evaluated with the help of two hypotheses as shown in Table 6.

TABLE 6
PROPOSED HYPOTHESES FOR EMPIRICALLY VALIDATING THE THEORETICAL FRAMEWORK

Proposed Hypothesis	Association Paths
H1: 'External Forces' construct is Significantly Related to Measurement Adaptation in Software Organizations	EF → MAD
H2: 'Measurement Adaptation' is Significantly Related to Metrics Acceptance in Software Organizations	MAD → MA

5.4.1. RESEARCH HYPOTHESES

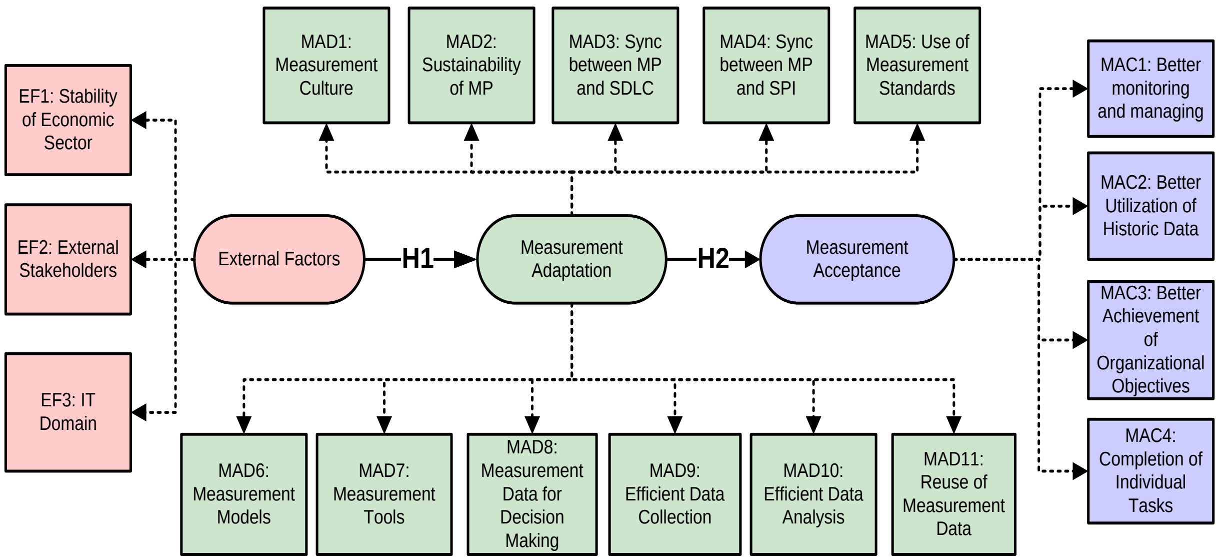

Figure 22. Proposed Theoretical Model with Success Factors

5.4.2. STRUCTURAL EQUATION MODELING

Structural Equation Modeling (SEM) is a statistical practice that explains the relationships among multiple variables. It is a multivariate statistical analysis method that is effectively used to evaluate the structural relationships between variables [30]. In this research, SEM is used for factor analysis.

The factor analysis is a technique used to identify a structure that describes the relationships among different observed variables. The factor analysis is performed by two methods i.e., Exploratory Factor Analysis and Confirmatory Factor Analysis, which are presented below.

A. EXPLORATORY FACTOR ANALYSIS (EFA)

EFA is concerned with how many factors are necessary to explain the relations among a set of indicators (variables) and with estimation of factor loadings [30], [36]. It explores the inter-relationship among variables to discover if those variables can be grouped into a smaller set of underlying factors. The KMO (Kaiser-Meyer-Olkin) test is used to test the factor loading (bonding of variable in a factor) and its value that is greater than or equal to 0.50 is acceptable [36], [37].

There are three constructs (factors) of the proposed model (Figure 22). The validity of each factor is tested with the help of EFA. The validity of a factor is indicated as good if a set of questionnaire items associated with it are loaded appropriately on it. The KMO value of our dataset is 0.726, which is acceptable for EFA[30], [36], [37]. Furthermore, Eigenvalues of the factors should be greater than 1

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

and total variance explained should be greater than 60 [30], [36], [37]. Table 7 shows the total variance explained (i.e., 66.248) and Eigenvalues greater than 1 for three factors as shown in Appendix C.

The principal component extraction method is used to extract the factors because the principal component method is most simple and suitable method for EFA [30], [36], [37]. Table 8 represents the rotated component matrix in which all the extracted components are mentioned with their suitable observed variables as shown in Appendix C. The correlations between different factors in the path diagram are known as factor loadings. Factor loadings can be positive or negative.

B. CONFIRMATORY FACTOR ANALYSIS (CFA)

CFA means testing the number of factors to confirm whether the loaded factors are according to the pre-developed theory (i.e., either the resulted latent variables actually cause impact on the observed factors). CFA is used to validate the scale with a specified hypothesis structure [38]. The extracted components are designed in AMOS 24 to perform CFA. AMOS 24 is an appropriate statistical tool to perform CFA. Figure 23 shows the path diagram of the proposed model in which relationships among the latent and observed variables, and error variables are shown. These variables are defined below:

- **Latent variables:** These variables are also known as unmeasured variables because these are difficult to measure directly. In our proposed model, there are three latent variables i.e., external factors (EF), measurement adaptation (MAD) and measurement acceptance (MA). They are shown in ellipses in the path diagram.
- **Observed variables:** The variables that are measured or observed by the researchers are known as measured variables or observed variables. These variables are presented in Tables 3-5 as questions/items for each construct/factor and also shown in rectangles in the path diagram.
- **Error variables:** Each observed variable have some kind of errors in measurements. The measurement errors of variables are shown in small circles in the path diagram.

In the path diagram (Figure 23), the EF, MAD and MA factors are acting as independent, mediator and dependent variables respectively.

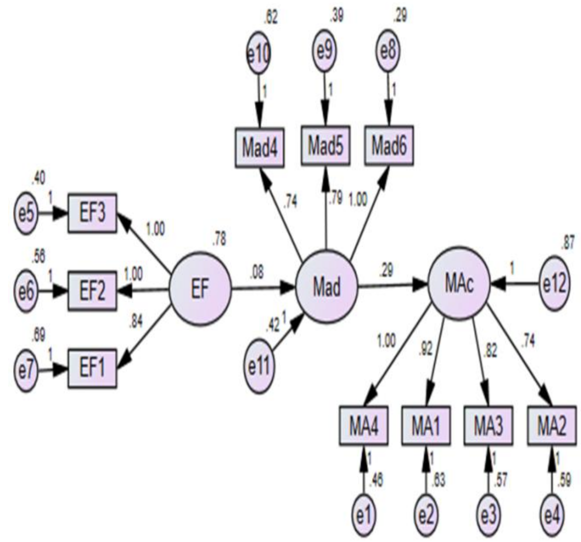

Figure 23. Path Diagram of Proposed Theoretical Model

In SEMs, model fits are used to indicate whether the underlying model is acceptable. If the resulting model is accepted then relationship paths of factors are tested for significance [39]. Table 9 presents fit indices with their standard and estimated values [40], [41]. The fit indices values are greater than the recommended values. The method to make a good model fit is iterative, which means to delete one indicator/questionnaire-item at a time and check again to make a model fit. The results of the model fits show that measurement model is a good fit with the data of the proposed model as shown in Table 9.

TABLE 9
MODEL FIT INDICES

Fit Indices	Recommended values [70]	Measurement Model
Goodness-of-Fit Index (GFI)	> 0.90	0.968
Adjusted Goodness-of-Fit Index (AGFI)	> 0.80	0.946
Comparative Fit Index (CFI)	> 0.90	0.995
Root Mean Square Error of Approximation (RMSEA)	< 0.08	0.048
Normed Fit Index (NFI)	> 0.90	0.941
Parsimony Normed Fit Index (PNFI)	> 0.60	0.690

C. HYPOTHESIS TESTING

The mode fit indices above represent the extent to which the hypothesized model fits the survey data. Later, the estimation of the paths between constructs is used for hypothesis testing as shown in Table 10. The empirical validation of the hypothesis is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

significant if the value of P of a relationship between constructs is less than .001, .01 or .05 [41].

Table 10 shows that the hypothesis H2 is significant with (P=.035). *It depicts that there is a strong empirical validation that metrics adaptation is associated with metrics acceptance.*

TABLE 10
HYPOTHESIS PATH ESTIMATIONS

	Hypothesis	
	H1	**H2**
Path	EF → MAD	MAD → MA
Estimates	.085	.292
S.E	.069	.138
C.R	1.226	2.114
P	.220	**.035**
Analysis	Non-significant	**Significant**

6. DISCUSSION

In this section, we discuss and analyze the results in terms of a summary of our findings, key issues and mitigation strategies, statistical and general validity threats and managerial implications and future work of this study.

6.1. SUMMARY OF FINDINGS

In this section, we summarize the state of measurement practices and statistical validation of the model. Analysis and findings of SLR on software measurement programs are the basis of this study [5].

First, we have identified three key components and 14 sub-components of measurement programs as shown in Figure 4. Second, a survey is conducted to analyze the state of measurement processes in Pakistani software development industry according to the identified components. As a whole, large and very large scale organizations are observed with more measurement practices as compared to SMEs (see details in Section 5.3). *Following are the highlights:*

A. DEMOGRAPHICS OF PARTICIPANTS AND SOFTWARE ORGANIZATIONS

- 50-60% of the participants are software engineers with experience of 1-3 years and they have responded being observers of measurement processes.
- SMEs are nearly 50% of the organizations in this survey. IT (70%), professional and business services (44%) and financials (33%) are the most common domains of software applications development.

B. DEMOGRAPHICS OF MEASUREMENT PROCESSES

1) MEASUREMENT STRUCTURE

- 61% of the organizations do not have defined measurement process.
- 41% of the organizations do not follow any software development standard. ISO/IEC 12207, ISO/IEC 23026 (IEEE std 2001-2002) and ISO/IEC 15393-2002 are used by 18%, 18%, and 9% of the organizations.
- 75% of the organizations do not follow any measurement standard. IEEE Std 982.1-2005 is used by highest (9%) number of the organizations.
- 95% of the organizations do not follow any measurement model.
- 80% of the software organizations don't use any measurement tool.
- 94% of the organizations don't focus on the sustainability of their measurement processes.

2) MEASUREMENT OBJECTIVES

- Characterization, evaluation, prediction, and improvement of software entities are the purpose of 53%, 59%, 52% and 61% of software organizations. 26% of software organizations use measurement process for all four purposes.
- Software measurement process is observed at 56%, 14%, and 30% of organizations at project, organization and simultaneously at both levels respectively.
- Process, product, and resource entities are measured by 74%, 48% and 36% of organizations respectively. Combination of all three entities is measured by 23% of organizations.
- CMMI is used by 27% of organizations and 71% organizations don't use any software process improvement approaches.
- 67% of organizations follow distributed software development and 62% of these organizations do not have defined measurement processes.

3) METRICS MANAGEMENT

- Most of the organizations use necessary and most commonly used metrics and expert judgment instead of measurement models, and automated tools.
- Duration (32%), cost (27%), and effort (27%) are the most measured attributes.
- 89% of the organizations don't use 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.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

- A measurement glossary is completely maintained by 38% of the organizations, while 19% partially and 16% don't use measurement glossary at all.
- Less than 50% of the organizations frequently use past data for prediction.
- 58% of software organizations don't have any metrics validation criteria.

C. HIGHLIGHTS OF MODEL VALIDATION

We have also proposed a model of success factors of implementing a MP based on analysis of SLR [2]. We have mapped the success factors among three main factors of a model i.e., external, measurement adaptation and measurement acceptance factors. The model is empirically validated based on the responses of 200 software professionals by using a multivariate statistical analysis technique (i.e., structural equation modeling (SEM)). This technique internally checks the factors of a theoretical model based on Exploratory Factor Analysis (EFA) by exploring the inter-relationship among the items to discover their bonding to constitute a factor. The Kaiser-Meyer-Olkin (KMO) test is used to test the factor loading (bonding of items in a factor) and its value that is greater than or equal to 0.50 is acceptable [36], [37]. The factor loadings of all the items of "external forces" and "measurement adaptation" factors have acceptable bonding (KMO>0.5) to constitute both factors. However, the EFA identifies 3 out of 11 items i.e. MAD4, MAD5, MAD6) which are necessary to constitute measurement adaptation factor based on factor loadings. This means that practitioners strongly believe that synchronization between measurement process and software process improvement (MAD4), use of measurement standards (MAD5), and use of measurement models (MAD6) are necessary to be adapted for successful implementation of a measurement process. The EFA is followed by confirmatory factor analysis (CFA), which verifies that the theoretical model is also statistically significant based on model fit indices. The factors of the models are connected based on hypotheses. Later, hypothesis path estimations are used for empirical validation of hypotheses. The H2 hypothesis is empirically validated (P<0.05) which means practitioners strongly believe that adaptation of measurement process is significantly related to improvement in monitoring and management of software projects (MA1), better utilization of historical data for prediction (MA2), achievement of organization-wide objectives (MA3) and enabling individuals to complete their tasks (MA4).

6.2. KEY ISSUES AND MITIGATION STRATEGIES

Table 11 presents key issues identified in this survey and their mitigation strategies discussed in measurement studies. These strategies are presented from our comprehensive literature reviews on software measurement programs [5], [27].

TABLE 11
MITIGATION STRATEGIES FOR KEY ISSUES FOUND IN THE SURVEY

Key Issues	Mitigation Strategy
Lack of defined measurement programs.	Large and very large software organizations should study following before planning a MP [5]: Available Measurement Planning models/tools Reported Challenges of MP planning Reported Success Factors of MP planning SMEs should study following before planning a MP [27]: Available Measurement Planning models/tools for SMEs Reported Challenges of MP planning in SMEs Reported Sucess Factors of MP planning Difference between the challenges of implementing MPs in large and very large organizations as compared to SMEs
Lack of suitable metrics selection methods.	Integration of MP with Software Life Cycle [21], [23], [26], [42], [43], Tool support: [24], [43]–[48], Use of standards: e.g., ISO/IEC 15939:2007 [7], ISO/IEC/IEEE 24765:2010 [49], [50], ISO/IEC 9126-x [51]–[53], ISO/IEC 25000 [54] and ISO/IEC 14598-x [54] (Most commonly discussed measurment standards for metrics selection methods for measuring software processes, products and resources),using measurement models [5], [27]
Lack of measurement data validation.	First, it should be analyzed that a software organization is measuring what it thinks it is measuring [55], [56]. Second: software organizations should adapt from 47 metrics validation criteria according to their context and goals of measurement. Meneely et al. [55] devised these criteria based on a comprehensive analysis of the last 40 years of research on metrics validation.
Lack of software development standards and software measurement standards.	Adaptation of software development standards according to size of an organization and its processes, product and resources e.g., IEEE glossary of software development standards and software measurement standards, and ISO/IEC Standards. The most commonly used measurement standards in large and SMEs are analyzed in [5], [27].
Lack of using measurement data for characterizatio n, evaluation, prediction,	Defining measurement goals of characterization, evaluation, improvement and prediction according to business goals [24], [25], [57] and then using measurement data to achieve these goals [6], [57]–[59]. Effective and efficient use

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

and improvement of software processes, products, and resources.	of historical [measurement] data for achieving prediction goals is a key input for planning a MP.
Mostly MPs are implemented at project level.	MPs are more beneficial if they are used at the project and organizational level simultaneously [4], [5], [27].
Lack of MP for distributed software development.	It is one of the least focused areas in this study and also among measurement studies [2], [77]. However, we could not find a study that specifically focuses on this issue.
Lack of Measurement Glossary	Consistent and complete definitions of measurement concepts and measurement entities [55], [60]–[62].
Lack of Sustainable MP	Identifying potential challenges of sustainable MP [16], [18], [21], [26], [63]–[66].
Lack of real time metrics	It is one of least focused area of measurement studies [2], [77] and in this survey. However, we could not find a study that specifically focuses on this issue.
Lack of measurement repository for prediction	Establishing a measurement repository [59], [67]–[71]. Complete automation of measurement process and establishing measurement repository at once might not be possible due to lack of budget, time and resources [72]. Therefore, incremental measurement process implementation is proposed in literature [44], [73]–[76].
Lack of Software process improvement initiatives	CMMI is the most commonly used SPI model [4]. However, there are many other SPI models analyzed in SLR [4]. The SMEs face fundamental challenges of available time, resources and budget for SPI. Therefore SMEs should adapt lightweight SPI models [27].

6.3. THREATS TO VALIDITY

In this section, we present statistical and general validity threats of the survey study. We have evaluated state of 14 measurement practices and validated a model of 18 success factors for implementing MPs with respect to responses of Pakistani software professionals.

6.3.1. CONSTRUCT VALIDITY

The construct validity relates to whether we are actually measuring the real-world software measurement practices. Furthermore, it involves reliability of the questionnaire-items that are based on the success factors of the proposed model. Reliability of the measures expresses the degree to which it is consistent and repeatable.

The questionnaire-items for evaluating the state of 14 basic measurement practices are designed with respect to findings of peer-reviewed and empirical research in SLR. The purpose of questionnaire-items was to count the frequencies of measurement practices in startups, SMEs, large and very organizations. The frequencies of measurement practices directly convey the presence/absence of measurement practices without any addition of biasness. The impact of participant's role, experience and organization type on their opinions regarding success factors of MPs are statistically evaluated in Section 6.3.5.

The Cronbach Alpha test measures the internal consistency (reliability) of questionnaire-items regarding the proposed model of success factors [77], [78]. It indicates the extent to which the questionnaire-items truly measure what it claims to measure, i.e., the underlying construct. In other words, it tests whether the questionnaire-items measuring a construct are inter-correlated well enough to jointly form the construct. Table 12 presents Cronbach Alpha test results of the three factors/constructs of the proposed model. The coefficient of alpha is calculated to measure the level of closeness among the related items in a group. This coefficient varies from 0 to 1 and higher value of alpha indicates higher reliability of scale as given below.

- ($\alpha \geq 0.9$): Excellent
- ($0.7 \leq \alpha < 0.9$): Good
- ($0.6 \leq \alpha < 0.7$): Acceptable
- ($0.5 \leq \alpha < 0.6$): Poor
- ($\alpha < 0.5$): Unacceptable

TABLE 12
CRONBACH ALPHA TEST RESULTS

Factors	No of Questionnaire-items	Cronbach alpha value
External forces	3	0.789
Measurement acceptance	4	0.827
Measurement adaptation	11	0.672

6.3.2. SAMPLING VALIDITY

There is possible sampling bias as purposive sampling is used instead of random sampling in this study. The purposive sampling involves selection of participants by the researchers as compared to random sampling. There are few surveys on software measurement processes which have used databases of clients and their employees working in collaboration with research centers and they have also used random sampling to an extent due to having very large data size e.g., Software Engineering Institute [8] and Finland software industry [8]. But in our case, there is no such database/repository available for research purpose and the software industry in Pakistan is going through an evolution [34]. Therefore, we applied the purposive sampling method [33] in which participants with specific characteristics are contacted based on their availability and convenience for the survey.

In order to partially mitigate this risk: first, we defined software measurement and software measurement process at the start of the survey and given the choice to give no response if they think they did not support or be part of a software organization, where measurement process is applied and/or observed as defined. There are 15 respondents that did not fill up the survey as they could not visualize measurement process in their organizations or as practitioners. The authors being university instructors mostly contacted computer science and software engineering alumni students. In addition, professionals working in software industry within the social circle are contacted. Furthermore, LinkedIn contacts of software professionals are also requested. The participants were assured of anonymity and confidentiality of data so that they could answer questionnaire-items without any reluctance.

6.3.3. MODEL VALIDITY

SEM is used to validate the model of success factors. The SEM performs model validation by estimating two separate models i.e., the measurement model and the structural model.

A. MEASUREMENT MODEL

It addresses the degree to which questionnaire-items capture the underlying constructs/factors. This model involves construct and discriminant validity. The construct validity is already presented above. Discriminant validity tests if the questionnaire-items measuring one construct are sufficiently uncorrelated from other constructs, thereby establishing the ability to discriminate between constructs. The tests results of KMO and eigenvalues in EFA ensure discriminant validity among success factors of the model (see details in Section 5.4.2). EFA involves a factor analysis of all the questionnaire-items together to examine the factor structure and loadings. The KMO test is used to test the factor loading and its value is 0.726 which is greater than the minimum acceptable value i.e. 0.50 [26], [36], [37]. Furthermore, Eigenvalues of each of three factors is greater than 1 and total variance explained is also greater than 60 [30], [36], [37] (Section 5.4.2).

B. STRUCTURAL MODEL

It describes the relationship between the constructs of the proposed model and tests the research hypotheses (Section 5.4.1).

CFA is used to estimate the structural model (Section 5.4.2). CFA can be applied to survey data provided that the responses are interval scaled, continuous data and sample size is great than 199 [79]. We have used five-point likert-scale and prior research has shown that this scale can be treated as continuous for the purposes of CFA [80]. West et al. evaluated the impact of categorization of a continuous variable in a covariance matrix with maximum likelihood estimation using a set of Monte Carlo simulations and estimations based on categorized variables were sufficiently robust [81]. Bollen and Barb have shown that a five-point likert-scale used to approximate underlying continuous variable has small distortion in correlation coefficients [82]. Therefore, we consider five-point likert-scale data in this study as continuous[21].

CFA is used to estimate the statistical fitness of the proposed model. The goodness of fit indices for the model (Section 5.4.2) depicts that the overall hypothesizes structure (Figure 22) appropriately fits the data. Later, we check individual hypothesis based on estimations of paths between constructs. Hypothesis testing shows that there is a statistically significant association (P=.035) between the metrics adaptation factor and metrics acceptance factor as shown in Table 10. Thus overall hypothesized structure of the proposed model and there is a strong empirical validation that metrics adaptation factor is associated with metrics acceptance factor.

6.3.4. CONCLUSION VALIDITY

Conclusion validity involves reliability and credibility of the conclusions drawn from the results of each phase of the research work [83]. It is related to the degree of confidence regarding the validity of the study that researchers can gain from the appropriateness of empirical methods [84]. We have taken a systematic approach to this research for managing the construct validity threats. First, we have conducted a SLR which is considered vital to accumulate evidences from the research literature. It systematically identifies and analyses evidences from primary studies (Section 3.1). The identification of basic measurement practices and the proposed model of success factors are based on the results and analysis of SLR (Section 3.2). Second, the survey was administered to 5 practitioners, who provided valuable feedback. The changes and feedback were incorporated before conducting the survey. Third, a structured survey instrument is used to statistically evaluate the frequency of 14 measurement practices in software industry and validation of statistically significant success factors in the proposed model based on the practitioners' beliefs (Section 4.1). The investigation of 14 measurement practices provides a key insight of state of measurement practices in 200 software organizations without drawing a general conclusion (Section 5.3). Finally, SEM provided a statistical validation of the model.

6.3.5. IMPACT OF PARTICIPANT'S DEMOGRAPHICS ON THEIR RESPONSES

One-Way ANOVA test is applied on the survey data to analyze the impact of participants' experience, role and organization type on their opinions regarding importance of three success factors. This test is used to compare mean scores of three or more groups, such as comparing the mean scores of different types of organizations; startups, SMEs, large, very large, regarding the impact of external factors on implementing software measurement process.

A. PARTICIPANT’S EXPERIENCE

The One-Way ANOVA test is conducted to compare the effect of different levels of participant's experience on their opinions regarding the impact of external factors, measurement adaptation factors and measurement acceptance factors on successful implementation of measurement process. Table 13 presents the results of the One-Way ANOVA test as shown in Appendix D.

The analysis of variance showed that the effect of participant's experience on their opinion regarding the impact of external forces was not significant, $F(3,196) = .35$, $p= .771$. Similarly, the analysis of variance showed that the effect of participant's experience on their opinion regarding the impact of measurement adaptation factors was not significant, $F(3,196) =1.808$, $p= .147$. The analysis of variance showed that the effect of participant's experience on their opinion regarding the measurement acceptance factors was also not significant, $F(3,196) =1.808$, $p= .147$. The p-value greater than 0.05 for each factor indicates that there is no experience level of participant's (1-3, 4-6, 7-9, 10 or more years) which impacts his/her opinion regarding the external factors, measurement adaptation factors, and measurement acceptance factors.

B. PARTICIPANT'S ROLE

The One-Way ANOVA test is conducted to compare the effect of participant's role on their opinions regarding the impact of external factors, measurement adaptation factors and measurement acceptance factors on successful implementation of measurement process. Table 14 presents results of the One-Way ANOVA test as shown in Appendix D.

The analysis of variance showed that the effect of participant's role on their opinions regarding impact of external forces was not significant, $F (5,194) = .35$, $p= .539$. Similarly, analysis of variance showed that the effect of participant's role on their opinion regarding the impact of measurement adaptation factors was not significant, $F (5,194) =.751$, $p= .586$. Furthermore, analysis of variance showed that the effect of participant's role on their opinion regarding the measurement acceptance factors was not significant, $F (5,194) =1.696$, $p= .137$. The p-value greater than 0.05 for each factor indicates that there is no participant's role (Program Manager, Project Manager, Team Leader, Software Engineer, Software Analyst, Software Quality assurance) which impacts his/her opinion regarding the external factors, measurement adaptation factors, and measurement acceptance factors.

C. PARTICIPANT'S ORGANIZATION

The One-Way ANOVA test is conducted to compare the effect of participant's organization type on their opinions regarding the impact of external factors, measurement adaptation factors and measurement acceptance factors on successful implementation of measurement process. Table 15 presents results of the One-Way ANOVA test as shown in Appendix D.

The analysis of variance showed that the effect of participant's organization type on their opinions regarding impact of external forces was not significant, F (3,196) = 1, p= .394. Similarly, analysis of variance showed that the effect of participant's role on their opinion regarding the impact of measurement adaptation factors was not significant, F (3,196) =1.442, p= .232. Moreover, analysis of variance showed that the effect of participant's role on their opinion regarding the measurement acceptance factors was significant, F (3,196) =3.773, p= .012. The p-value greater than 0.05 for external factor and measurement adaptation factor indicates that there is no participant's organization type (start-up, SME, large, very large) which impacts his/her opinion regarding both factors. The p-value less than 0.05 for measurement acceptance factor requires further investigation to identify which organization type effects the opinion of the participants and this is done with the help of Post Hoc Tukey Test.

The Post Hoc Tukey Test is applied on the job-organization of participants and their opinion regarding measurement acceptance factor. Table 16 presents results of the Post Hoc Tukey Test as shown in Appendix D. The test showed that the opinions of participants working in SMEs and very large organizations differed significantly (at $p < .05$) regarding measurement acceptance factors. Similarly, opinions of participants working in large and very large organizations differed significantly (at $p < .05$) regarding measurement acceptance factors. Furthermore, the effect size of participant's job-organization on their opinion regarding measurement acceptance is calculated by Etta-Square value ($\eta 2$). The univariate analysis in the SPSS tool is used to calculate $\eta 2=0.055$. According to Cohen ($0.02 > \eta 2 > 0.13$), the effect size is small [85]. In a nutshell, this means that effect of participant's job-organization on her/his opinion regarding measurement acceptance factors is statistically significant but the size of the effect of job-organization is small.

6.4. MANAGERIAL IMPLICATIONS AND FUTURE WORK

Software managers are key stakeholders of measurement process while planning, monitoring, and controlling software development process and making decisions. Our research informs them with a first level description of basic measurement practices based on a SLR on MPs and aims to support planning and setting up measurement processes. Furthermore, it presents the state of Pakistani software industry according to the basic measurement practices. The researchers and practitioners from other countries can reuse this study 1) to evaluate and compare the state of measurement practices in their industry 2) extend the 14 measurement practices or modify the survey-questionnaire according to their context. The software industry in Pakistan is small and going through an evolution as compared to USA and European countries [8], [9], [26], [34]. The survey results depict lack of basic measurement practices such as defined measurement processes, measurement standards, measurement models and automated measurement tools. We have also proposed mitigation strategies for observed challenges based on analysis of literature on MPs. On the other hand, measurement processes are considered key component for software process improvement in research and software industries of modern world [4], [21], [26], [42].

In this survey, we have found out that nearly half of 200 software organizations are SMEs. Most of MP studies in the SMEs report fundamental challenges of budget, time and resources as compare to large organizations. Therefore, we have also conducted a systematic mapping study on MPs to investigate measurement models, tools, standards, practices, challenges, and success factor in SMEs [27] and using it to design a questionnaire for specifically investigating MPs in SMEs. Moreover on a broader spectrum, this study, SLR [5] and SMS [27] are the first step towards research and development of measurement processes specifically for Pakistani software industry. In future, we are planning to evaluate the software development practices and measurement practices with the help of interviews

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

and observations and propose tailor-made measurement model and practices.

It is also observed that 58% of participants in this survey have 1-3 years of experience which might affect their responses regarding measurement practices. In this study, we have applied One-Way ANOVA test on the survey data to analyze the impact of participants' experience, role and organization type on their opinions regarding importance of three success factors (Section 6.3.5). The test results reveal that only participants' job-organization has a statistically significant effect on their opinions regarding measurement acceptance but the size of the effect of job-organization is small (Section 6.3.5). We have distributed the survey among 542 potential participants keeping in mind the experience range of 1-3, 4-7, 7-9 and 10 or more years. However, more university alumni confirmed to fill up the survey. The participants were assured of anonymity and confidentiality of data so that they could answer questionnaire-items without any reluctance. However, only 37% of the contacted software professionals responded. The reason might include lack of industry and academia coordination in research and software professionals are not used to participate in survey-based research.

It is important for software managers to consider success factors of implementing MPs while implementing basic measurement practices (Section 6.2). The successful adaptation of measurement practices requires structural changes to improve the chances of measurement acceptance e.g., establishing measurement culture, synchronizing measurement process with software development process and software process improvement and sustainability of measurement process across all software projects in an organization (Section 3.2). Moreover, it is important for software managers to facilitate adequate budget, time and resources for successful implementation of MPs [4], [5], [27]. This approach can support them to improve chances of successful implementation of MPs.

The proposed model of success factors has multiple opportunities for future work. First, the model is proposed purely based on literature review. More work by practitioners and researchers is required to refine the factors and possibly add more factors in this model. Second, we are measuring the success of MP in the form of measurement acceptance factor. With more research, the success of MP can also be measured in terms of return over investment, software process improvement and product quality improvement. Third, context of implementing MP in SMEs and large organizations face different challenges [27], therefore, organizational settings should also be considered while implementing the proposed model of success factors.

6. CONCLUSIONS

In this paper, we evaluated key components and success factors of software measurement programs extracted from our previous SLR and proposed a model of success factors. In general, the state of measurement practices in software organizations is not at a good status around the globe and requires special attention for Pakistani software development organizations. We discovered very conclusive outcomes through this survey. For example, there are more than 50 measurement models reported in the literature but only 10% software organizations in this survey follow any measurement model. Similarly, 75% organizations do not follow any measurement standard. There are 80% software organizations which do not use any measurement tool. There are multiple software process improvement (SPI) models but 71% software organizations do not use any SPI model. The resource metrics are least utilized by software organizations. There are only 38% software organizations which frequently use measurement data for prediction. As a whole, large and very large scale organizations are observed with more measurement practices as compared to SMEs. Later, we have also presented mitigation strategies for key issues identified in software organizations. To the best of our knowledge, this survey is a first attempt to evaluate measurement processes in Pakistani software industry.

In this paper, we have proposed a model of 18 success factors for implementing a MP. The 18 factors are mainly categorized under external, measurement adaptation and measurement acceptance factors. The model is empirically validated by applying Structural Equation Modeling (SEM) on the responses of 200 software professionals and it is revealed from responses (strongly believe) that synchronization between measurement process and software process improvement (MAD4), use of measurement standards (MAD5), and use of measurement models (MAD6) are necessary to be adapted for the successful implementation of a measurement process. In addition, software professionals strongly believe that adaptation of measurement process are significantly related to improvement in monitoring and management of software projects (MA1), better utilization of historical data for prediction (MA2), achievement of organization-wide objectives (MA3) and enabling individuals to complete their tasks (MA4).

According to PSEB[4], IT industry in Pakistan is going through a revolution as IT investments have currently risen up to $5.1138 billion from $100 million in 2003 and growth rate of IT remittances in Pakistan was 16% in the year 2016-2017. Software measurement is vital for software process improvement (SPI) and effective monitoring and controlling of processes. They can provide key support towards meeting international quality standards of processes and products. For instance, CMMI is most commonly used SPI model and its certification by a organization cannot only improve its processes but also improves its chances of winning international project bids. The software process capability and maturity level-4 and level-5 of CMMI cannot be achieved without effective and efficient measurement processes. Therefore, software organizations might plan and evaluate their measurement processes according to three key components presented in this study and they should also consider the model of success factors before planning their measurement programs. In future work, this study might also be conducted in the software industries of other countries to evaluate and compare the basic states of measurement processes and to identify the most important success factors.

[4] http://www.pseb.org.pk/

ACKNOWLEDGMENTS

The authors are thankful to anonymous reviewers for thought-provoking feedback on improving the earlier version of this paper.

REFERENCES

[1] C. G. Von Wangenheim, M. Thiry, and D. Kochanski, "Empirical evaluation of an educational game on software measurement," *Empirical Software Engineering*, vol. 14, no. 4, pp. 418–452, 2009.

[2] A. Rainer and T. Hall, "Key success factors for implementing software process improvement: a maturity-based analysis," *Journal of Systems and Software*, vol. 62, no. 2, pp. 71–84, 2002.

[3] M. Söylemez and A. Tarhan, "Challenges of software process and product quality improvement: catalyzing defect root-cause investigation by process enactment data analysis," *Software Quality Journal*, vol. 26, no. 2, pp. 779–807, 2018.

[4] M. Unterkalmsteiner, T. Gorschek, A. M. M. Islam, C. K. Cheng, R. B. Permadi, and R. Feldt, "Evaluation and Measurement of Software Process Improvement: A Systematic Literature Review," *IEEE Transactions on Software Engineering*, vol. 38, no. 2, pp. 398–424, Mar. 2012.

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

[6] R. E. Park, W. B. Goethert, and W. A. Florac, "Goal-Driven Software Measurement. A Guidebook.," Aug. 1996.

[7] "IEEE Standard Adoption of ISO/IEC 15939:2007 Systems and Software Engineering Measurement Process." Jan-2009.

[8] J. Soini, "A Survey of Metrics Use in Finnish Software Companies," in *2011 International Symposium on Empirical Software Engineering and Measurement (ESEM)*, 2011, pp. 49–57.

[9] M. Kasunic, "The state of software measurement practice: results of 2006 survey," 2006.

[10] P. Bourque, S. Wolff, R. Dupuis, A. Sellami, and A. Abran, "Lack of consensus on measurement in software engineering: Investigation of related issues," in *14th International Workshop on Software Measurement (IWSM)*, 2004.

[8] W. G. Linda and S. D. Sheetz, "The adoption of software measures: A technology acceptance model (TAM) perspective." Information & Management, 2014.

[12] J. Popović and D. Bojić, "A comparative evaluation of effort estimation methods in the software life cycle," *Computer Science and Information Systems*, vol. 9, no. 1, pp. 455–484, 2012.

[13] M. Díaz-Ley, F. García, and M. Piattini, "MIS-PyME Software Measurement Maturity Model-Supporting the Definition of Software Measurement Programs," in *Product-Focused Software Process Improvement*, A. Jedlitschka and O. Salo, Eds. Springer Berlin Heidelberg, 2008, pp. 19–33.

[14] M. Bundschuh and C. Dekkers, *The IT measurement compendium: estimating and benchmarking success with functional size measurement*. Springer Science & Business Media, 2008.

[15] W. Goethert and W. Hayes, "Experiences in implementing measurement programs," 2001.

[16] C. A. Dekkers and P. A. McQuaid, "The dangers of using software metrics to (mis)manage," *IT Professional*, vol. 4, no. 2, pp. 24–30, Mar. 2002.

[17] M. Berry and R. Jeffery, "An Instrument for Assessing Software Measurement Programs," *Empirical Software Engineering*, vol. 5, no. 3, pp. 183–200, Nov. 2000.

[18] F. Niessink and H. V. Vliet, "Measurement Program Success Factors Revisited," *Information and Software Technology*, vol. 43, pp. 617–628, 2001.

[19] F. Niessink, "Perspectives on Improving Software Maintenance," in *Proceedings of the IEEE International Conference on Software Maintenance (ICSM'01)*, Washington, DC, USA, 2001, pp. 553–.

[20] A. J. Bianchi, "Management indicators model to evaluate performance of IT organizations," in *Portland International Conference on Management of Engineering and Technology, 2001. PICMET '01*, 2001, vol. Supplement, pp. 217–229 vol.2.

[21] A. Gopal, T. Mukhopadhyay, and M. S. Krishnan, "The impact of institutional forces on software metrics programs," *IEEE Transactions on Software Engineering*, vol. 31, no. 8, pp. 679–694, Aug. 2005.

[22] F. van Latum, R. van Solingen, M. Oivo, B. Hoisl, D. Rombach, and G. Ruhe, "Adopting GQM-Based Measurement in an Industrial Environment," *IEEE Softw.*, vol. 15, no. 1, pp. 78–86, Jan. 1998.

[23] V. R. Basili *et al.*, "Linking Software Development and Business Strategy Through Measurement," *Computer*, vol. 43, no. 4, pp. 57–65, Apr. 2010.

[24] V. Mandić and M. Oivo, "SAS: A Tool for the GQM+Strategies Grid Derivation Process," in *Product-Focused Software Process Improvement*, M. A. Babar, M. Vierimaa, and M. Oivo, Eds. Springer Berlin Heidelberg, 2010, pp. 291–305.

[25] K. Petersen, C. Gencel, N. Asghari, and S. Betz, "An elicitation instrument for operationalising GQM+Strategies (GQM+S-EI)," *Empir Software Eng*, vol. 20, no. 4, pp. 968–1005, Mar. 2014.

[26] M. Staron and W. Meding, "MeSRAM – A method for assessing robustness of measurement programs in large software development organizations and its industrial evaluation," *Journal of Systems and Software*, vol. 113, pp. 76–100, Mar. 2016.

[27] T. Tahir, G. Rasool, and M. Noman, "A Systematic Mapping Study on Software Measurement Programs in SMEs," *e-Informatica Software Engineering Journal*, vol. 12, no. 1, pp. 133–165, 2018.

[28] T. Dyba, "An empirical investigation of the key factors for success in software process improvement," *IEEE Transactions on Software Engineering*, vol. 31, no. 5, pp. 410–424, 2005.

[29] M. Kasunic, "Designing an effective survey," CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 2005.

[30] B. M. Byrne, *Structural Equation Modeling With EQS: Basic Concepts, Applications, and Programming, Second Edition*. Routledge, 2013.

[31] I. E. Allen and C. A. Seaman, "Likert Scales and Data Analyses," *Quality Progress; Milwaukee*, vol. 40, no. 7, pp. 64–65, Jul. 2007.

[32] J. W. Creswell and J. D. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications, 2017.

[33] R. D. Fricker, "Sampling methods for web and e-mail surveys," *The SAGE handbook of online research methods*, pp. 195–216, 2008.

[34] J. A. Shamsi and Z. Nasir, "Unleashing the Pakistan Software Industry: Growth Prospects and Challenges," *IT Professional*, vol. 18, no. 5, pp. 12–14, Sep. 2016.

[35] N. Phaphoom, X. Wang, S. Samuel, S. Helmer, and P. Abrahamsson, "A survey study on major technical barriers affecting the decision to adopt cloud services," *Journal of Systems and Software*, vol. 103, pp. 167–181, May 2015.

[36] I. Izquierdo Alfaro, J. Olea Díaz, and F. J. Abad, "Exploratory factor analysis in validation studies: uses and recommendations," *Psicothema*, 2014.

[37] J. F. Hair, W. C. Black, and B. J. Babin, *Multivariate Data Analysis: A Global Perspective*. Pearson Education, 2010.

[38] H. Gatignon, *Statistical analysis of management data*. Springer, 2003.

[39] S. Moss, “Fit indices for structural equation modeling,” *Website: http://www. psych-it. com. au/Psychlopedia/article. asp*, 2009.

[40] M. Maqableh, R. M. T. Masa’deh, and A. B. Mohammed, “The Acceptance and Use of Computer Based Assessment in Higher Education,” *Journal of Software Engineering and Applications*, vol. 08, no. 10, p. 557, Oct. 2015.

[41] D. Hooper, J. Coughlan, and M. Mullen, “Structural equation modelling: Guidelines for determining model fit,” *Articles*, p. 2, 2008.

[42] M. Staron, “Critical role of measures in decision processes: Managerial and technical measures in the context of large software development organizations,” *Information and Software Technology*, vol. 54, no. 8, pp. 887–899, Aug. 2012.

[43] T. Kilpi, “Implementing a software metrics program at Nokia,” *IEEE Software*, vol. 18, no. 6, pp. 72–77, Nov. 2001.

[44] L. Lavazza and G. Barresi, “Automated support for process-aware definition and execution of measurement plans,” in *27th International Conference on Software Engineering, 2005. ICSE 2005. Proceedings*, 2005, pp. 234–243.

[45] M. Auer, B. Graser, and S. Biffl, “A survey on the fitness of commercial software metric tools for service in heterogeneous environments: common pitfalls,” in *Software Metrics Symposium, 2003. Proceedings. Ninth International*, 2003, pp. 144–152.

[46] D. Homchuenchom, C. Piyabunditkul, H. Lichter, and T. Anwar, “SPIALS: A light-weight Software Process Improvement self-assessment tool,” in *2011 Malaysian Conference in Software Engineering*, 2011, pp. 195–199.

[47] B. Daubner, “Empowering software development environments by automatic software measurement,” in *Software Metrics, 2005. 11th IEEE International Symposium*, 2005, pp. 3 pp. – 45.

[48] J. C. Ruiz, Z. B. Osorio, J. Mejia, M. Munoz, A. M. Ch´vez, and B. A. Olivares, “Definition of a Hybrid Measurement Process for the Models ISO/IEC 15504-ISO/IEC 12207:2008 and CMMI Dev 1.3 in SMEs,” in *2011 IEEE Electronics, Robotics and Automotive Mechanics Conference*, 2011, pp. 421–426.

[49] “Systems and software engineering – Vocabulary,” *ISO/IEC/IEEE 24765:2010(E)*, pp. 1–418, Dec. 2010.

[50] “Systems and software engineering – Vocabulary,” *ISO/IEC/IEEE 24765:2010(E)*, pp. 1–418, Dec. 2010.

[51] H. Al-Kilidar, K. Cox, and B. Kitchenham, “The use and usefulness of the ISO/IEC 9126 quality standard,” in *2005 International Symposium on Empirical Software Engineering, 2005*, 2005, pp. 7 pp.-.

[52] “ISO/IEC 9126-1:2001 - Software engineering -- Product quality -- Part 1: Quality model.” [Online]. Available: https://www.iso.org/standard/22749.html. [Accessed: 02-May-2018].

[53] “ISO/IEC TR 9126-2:2003 - Software engineering -- Product quality -- Part 2: External metrics.” [Online]. Available: https://www.iso.org/standard/22750.html. [Accessed: 02-May-2018].

[54] “ISO/IEC 25000:2014 - Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE.” [Online]. Available: https://www.iso.org/standard/64764.html. [Accessed: 02-May-2018].

[55] F. García *et al.*, “Towards a consistent terminology for software measurement,” *Information and Software Technology*, vol. 48, no. 8, pp. 631–644, Aug. 2006.

[56] C. Kaner, S. Member, and W. P. Bond, “Software Engineering Metrics: What Do They Measure and How Do We Know?,” in *In METRICS 2004. IEEE CS*, 2004.

[57] V. Basili, “Software modeling and measurement: the Goal/Question/Metric paradigm,” 1992.

[58] J. Heidrich and J. Münch, “Goal-Oriented Setup and Usage of Custom-Tailored Software Cockpits,” in *Product-Focused Software Process Improvement*, A. Jedlitschka and O. Salo, Eds. Springer Berlin Heidelberg, 2008, pp. 4–18.

[59] Z.-K. Görgülü and S. Pickl, “Adaptive Business Intelligence: The Integration of Data Mining and Systems Engineering into an Advanced Decision Support as an Integral Part of the Business Strategy,” in *Business Intelligence and Performance Management*, P. Rausch, A. F. Sheta, and A. Ayesh, Eds. Springer London, 2013, pp. 43–58.

[60] A. Meneely, B. Smith, and L. Williams, “Validating Software Metrics: A Spectrum of Philosophies,” *ACM Trans. Softw. Eng.*

Methodol., vol. 21, no. 4, pp. 24:1–24:28, Feb. 2013.

[61] V. R. Basili *et al.*, "Linking Software Development and Business Strategy Through Measurement," *Computer*, vol. 43, no. 4, pp. 57–65, Apr. 2010.

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

[63] W. M. Miroslaw Staron, "Factors Determining Long-term Success of a Measurement Program: An Industrial Case Study." [Online]. Available: http://gup.ub.gu.se/publication/154397-factors-determining-long-term-success-of-a-measurement-program-an-industrial-case-study. [Accessed: 28-Jul-2015].

[64] M. Staron, W. Meding, G. Karlsson, and C. Nilsson, "Developing measurement systems: an industrial case study," *J. Softw. Maint. Evol.: Res. Pract.*, vol. 23, no. 2, pp. 89–107, Mar. 2011.

[65] D. Hamann, A. Beitz, M. Müller, and R. van Solingen, "Using FAME Assessments to Define Measurement Goals," in *New Approaches in Software Measurement*, R. Dumke and A. Abran, Eds. Springer Berlin Heidelberg, 2001, pp. 220–232.

[66] M. Staron and W. Meding, "Factors Determining Long-term Success of a Measurement Program: An Industrial Case Study," *e-Informatica Software Engineering Journal*, vol. 5, pp. 7–23, 2011.

[67] F. Niessink and H. Van Vliet, "Measurements should generate value, rather than data [software metrics]," in *Software Metrics Symposium, 1999. Proceedings. Sixth International*, 1999, pp. 31–38.

[68] W. Harrison, "A flexible method for maintaining software metrics data: a universal metrics repository," *Journal of Systems and Software*, vol. 72, no. 2, pp. 225–234, Jul. 2004.

[69] L. Chirinos, F. Losavio, and J. Bøegh, "Characterizing a data model for software measurement," *Journal of Systems and Software*, vol. 74, no. 2, pp. 207–226, Jan. 2005.

[70] N. Fenton, M. Neil, W. Marsh, P. Hearty, L. Radlinski, and P. Krause, "Project Data Incorporating Qualitative Factors for Improved Software Defect Prediction," in *Proceedings of the Third International Workshop on Predictor Models in Software Engineering*, Washington, DC, USA, 2007, pp. 2–.

[71] 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, pp. 1397–1418, Aug. 2013.

[72] M. Diaz-Ley, F. Garcia, and M. Piattini, "Implementing a software measurement program in small and medium enterprises: a suitable framework," *IET Software*, vol. 2, no. 5, pp. 417–436, Oct. 2008.

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

[74] J. C. Abib and T. G. Kirner, "A GQM-based Tool to Support the Development of Software Quality Measurement Plans," *SIGSOFT Softw. Eng. Notes*, vol. 24, no. 4, pp. 75–80, Jul. 1999.

[75] F. García, F. Ruiz, J. A. Cruz, and M. Piattini, "Integrated Measurement for the Evaluation and Improvement of Software Processes," in *Software Process Technology*, F. Oquendo, Ed. Springer Berlin Heidelberg, 2003, pp. 94–111.

[76] K. Petersen and C. Wohlin, "Software process improvement through the Lean Measurement (SPI-LEAM) method," *Journal of Systems and Software*, vol. 83, no. 7, pp. 1275–1287, Jul. 2010.

[77] Y. Liu, H. Li, and C. Carlsson, "Factors driving the adoption of m-learning: An empirical study," *Computers & Education*, vol. 55, no. 3, pp. 1211–1219, 2010.

[78] S. L. Pfleeger and B. A. Kitchenham, "Principles of survey research: part 1: turning lemons into lemonade," *ACM SIGSOFT Software Engineering Notes*, vol. 26, no. 6, pp. 16–18, 2001.

[79] R. G. Lomax and R. E. Schumacker, *A beginner's guide to structural equation modeling*. Routledge Academic New York, NY, 2012.

[80] D. R. Johnson and J. C. Creech, "Ordinal measures in multiple indicator models: A simulation study of categorization error," *American Sociological Review*, pp. 398–407, 1996.

[81] S. G. West, J. F. Finch, and P. J. Curran, "Structural equation models with nonnormal variables: Problems and remedies.," 1995.

[82] K. A. Bollen and K. H. Barb, "Pearson's r and coarsely categorized measures," *American Sociological Review*, pp. 232–239, 1981.

[83] T. D. Cook, D. T. Campbell, and W. Shadish, *Experimental and quasi-experimental designs for generalized causal inference*. Houghton Mifflin Boston, 2002.

[84] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in*

software engineering. Springer Science & Business Media, 2012.

[85] J. Cohen, “A power primer.,” *Psychological bulletin*, vol. 112, no. 1, p. 155, 1992.

[86] N. Fenton and J. Bieman, *Software Metrics: A Rigorous and Practical Approach, Third Edition*. CRC Press, 2014.

[87] L. Buglione and A. Abran, “ICEBERG: a different look at Software Project Management,” in *Proc. 12th Int’l Workshop Software Measurement*, 2002, pp. 153–167.

[88] L. Buglione and A. Abran, “Software measurement body of knowledge–overview of empirical support,” in *Proceedings of the 15th International Workshop on Software Measurement–IWSM*, 2005, pp. 353–368.

[89] J. D. Herbsleb, “Global Software Engineering: The Future of Socio-technical Coordination,” in *2007 Future of Software Engineering*, Washington, DC, USA, 2007, pp. 188–198.

[90] P. Ardimento, M. T. Baldassarre, D. Caivano, and G. Visaggio, “Multiview Framework for Goal Oriented Measurement Plan Design,” in *Product Focused Software Process Improvement*, F. Bomarius and H. Iida, Eds. Springer Berlin Heidelberg, 2004, pp. 159–173.

[91] T. Murphy and K. Cormican, “An analysis of non-observance of best practice in a software measurement program,” *Procedia Technology*, vol. 5, pp. 50–58, 2012.

[92] R. J. Offen and R. Jeffery, “Establishing software measurement programs,” *IEEE Software*, vol. 14, no. 2, pp. 45–53, 1997.

[93] T. Olsson and P. Runeson, “V-GQM: a feed-back approach to validation of a GQM study,” in *Software Metrics Symposium, 2001. METRICS 2001. Proceedings. Seventh International*, 2001, pp. 236–245.

[94] 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, pp. 3091–3108, Dec. 2013.

[95] O. Gómez, H. Oktaba, M. Piattini, and F. García, “A Systematic Review Measurement in Software Engineering: State-of-the-Art in Measures,” in *Software and Data Technologies*, J. Filipe, B. Shishkov, and M. Helfert, Eds. Springer Berlin Heidelberg, 2008, pp. 165–176.

[96] M. G. Mendonca and V. R. Basili, “Validation of an approach for improving existing measurement frameworks,” *IEEE Transactions on Software Engineering*, vol. 26, no. 6, pp. 484–499, Jun. 2000.

[97] P. Parviainen, J. Jarvinen, and T. Sandelin, “Practical experiences of tool support in a GQM-based measurement programme,” *Software Quality Journal*, vol. 6, no. 4, pp. 283–294, Dec. 1997.

[98] A. Fuggetta, L. Lavazza, S. Morasca, S. Cinti, G. Oldano, and E. Orazi, “Applying GQM in an Industrial Software Factory,” *ACM Trans. Softw. Eng. Methodol.*, vol. 7, no. 4, pp. 411–448, Oct. 1998.

[99] F. Niessink and H. van Vliet, “Towards mature measurement programs,” in *Proceedings of the Second Euromicro Conference on Software Maintenance and Reengineering, 1998*, 1998, pp. 82–88.

[100] 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. 10–11, pp. 1223–1237, Oct. 2010.

[101] M. Díaz-Ley, F. García, and M. Piattini, “Implementing Software Measurement Programs in Non Mature Small Settings,” in *Software Process and Product Measurement*, J. J. Cuadrado-Gallego, R. Braungarten, R. R. Dumke, and A. Abran, Eds. Springer Berlin Heidelberg, 2008, pp. 154–167.

[102] R. Van Solingen and E. Berghout, “Integrating goal-oriented measurement in industrial software engineering: industrial experiences with and additions to the Goal/Question/Metric method (GQM),” in *Software Metrics Symposium, 2001. METRICS 2001. Proceedings. Seventh International*, 2001, pp. 246–258.

[103] J. W. Moore, “Software Engineering Standards,” in *Encyclopedia of Software Engineering*, John Wiley & Sons, Inc., 2002.

[104] F. García, M. Serrano, J. Cruz-Lemus, F. Ruiz, and M. Piattini, “Managing software process measurement: A metamodel-based approach,” *Information Sciences*, vol. 177, no. 12, pp. 2570–2586, Jun. 2007.

[105] E. Arisholm, L. C. Briand, and E. B. Johannessen, “A systematic and comprehensive investigation of methods to build and evaluate fault prediction models,” *Journal of Systems and Software*, vol. 83, no. 1, pp. 2–17, Jan. 2010.

[106] M. Ciolkowski, J. Heidrich, F. Simon, and M. Radicke, “Empirical Results from Using Custom-made Software Project Control Centers in Industrial Environments,” in *Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and*

Measurement, New York, NY, USA, 2008, pp. 243–252.

[107] S. Komi-Sirvio, P. Parviainen, and J. Ronkainen, "Measurement automation: methodological background and practical solutions a multiple case study," in *Software Metrics Symposium, 2001. METRICS 2001. Proceedings. Seventh International*, 2001, pp. 306–316.

[108] A. M. Bhatti, H. M. Abdullah, and C. Gencel, "A Model for Selecting an Optimum Set of Measures in Software Organizations," in *Software Process Improvement*, R. V. O'Connor, N. Baddoo, J. C. Gallego, R. R. Muslera, K. Smolander, and R. Messnarz, Eds. Springer Berlin Heidelberg, 2009, pp. 44–56.

[109] 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, pp. 3091–3108, 2013.

[110] D. R. Goldenson, A. Gopal, and T. Mukhopadhyay, "Determinants of success in software measurement programs: initial results," in *Software Metrics Symposium, 1999. Proceedings. Sixth International*, 1999, pp. 10–21.

[111] M. Staron and W. Meding, "Ensuring Reliability of Information Provided by Measurement Systems," in *Software Process and Product Measurement*, A. Abran, R. Braungarten, R. R. Dumke, J. J. Cuadrado-Gallego, and J. Brunekreef, Eds. Springer Berlin Heidelberg, 2009, pp. 1–16.

[112] R. Braungarten, R. Dumke, and M. Kunz, *An Approach to Classify Software Measurement Storage Facilities*. 2005.

[113] A. Birk, R. van Solingen, and J. Jarvinen, "Business impact, benefit, and cost of applying GQM in industry: an in-depth, long-term investigation at Schlumberger RPS," in *Software Metrics Symposium, 1998. Metrics 1998. Proceedings. Fifth International*, 1998, pp. 93–96.

[114] M. Berry and M. F. Vandenbroek, "A targeted assessment of the software measurement process," in *Software Metrics Symposium, 2001. METRICS 2001. Proceedings. Seventh International*, 2001, pp. 222–235.

[115] L. C. Briand, C. M. Differding, and H. D. Rombach, "Practical guidelines for measurement-based process improvement," *Softw. Process: Improve. Pract.*, vol. 2, no. 4, pp. 253–280, Dec. 1996.

[116] G.S. Sureshchandar and Rainer Leisten, "A framework for evaluating the criticality of software metrics: an analytic hierarchy process (AHP) approach," *Measuring Business Excellence*, vol. 10, no. 4, pp. 22–33, Oct. 2006.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

Appendix A

TABLE 2
COMPONENTS OF MEASUREMENT PROGRAMS ANALYZED IN THE SOFTWARE INDUSTRY

Main-Component	Sub-Component	Description and Response Collection Method
Measurement Structure	**Software development and measurement standards**	The software development standards encourage the use of software measurement processes for planning, monitoring, controlling and improving software processes, products and resources. Initially, we identified the use of software development and measurement standards for implementing MPs through primary studies in our SLR [5]. Later on, we also consulted IEEE glossary of standards[5] and IEEE glossary of measurement standards[6] for a complete list of available standards in this survey to analyze the state of the standardization in software development industry. **Response collection method:** List of software development standards and measurement standards are presented with checkboxes. A textbox with ***"Other"*** label is added to make sure that participant could add a standard if it is missing in the presented options.
	Measurement models	The use of measurement models helps systematic implementation of a measurement process. In the systematic literature review, we have identified 35 measurement models [5]. **Response collection method:** In a close-ended question, "Does your organization use any measurement model?" (Yes, No), a follow up open-ended question is provided a text box to capture model name.
	Measurement tools	The application of measurement tools automates collection, storage, analysis and re-usability of the data [5]. **Response collection method:** An open-ended question is presented with a text box i.e., "Which automated tools your organization use for collecting metrics? Please specify "
Measurement Objectives	**Measurement purpose**	MPs support informed decision making by identifying four type of purposes: characterize, evaluate, predict and utilize resources [6]. In the SLR and in this survey, we analyzed primary studies and software development organizations according to these four purposes. The use of only four purposes generalizes most of MP activities. **Response collection method:** List of four measurement purposes is presented with check boxes.
	Scope of measurement process and measurement entities	Fenton and Pfleeger [86] distinguished three types of measurement entities i.e. process, product, and resource. Abran and Buglione [87], [88] added project and organization to measurement entities as well. In the SLR and this survey, we distinguished measurement entity focus (i.e. process, product, and resource) from the level of MP implementation (project level or organizational level). **Response collection method:** List of three types of metrics is presented with checkboxes. Similarly, in another question list of three possible measurement scopes i.e. Project level, organization level and both-levels is presented. A textbox with ***"Other"*** label is added to make sure that participants could add a type of metrics if they believe it is missing in the presented options.
	Software process improvement	It is observed through data of the primary studies in SLR that software measurement is considered as an integral part of every software process improvement (SPI) initiative. Software measurement enables assessment of SPI strategies. **Response collection method:** An open-ended question is presented with a text box i.e., "Has your organization been involved in a software process improvement initiative (e.g., CMMI, SPICE, etc.)? please specify:"
	Defined measurement process	Software measurement is pervasive as every single software process generates data and/or uses data generated about processes, products and resources. There is a possibility that software organizations use measurement practices and measurement data without a formal definition of a MP. **Response collection method:** Binary response (Yes, No) is collected on the existence of a defined measurement process.
	Measurement in distributed development	In the SLR, we could not find a measurement study on implementation of MPs in distributed development of software projects. However, software MPs are critical for distributed software development because they face more challenges of planning, monitoring, controlling and improving due to constraints such as time zone differences, geographical distances, cultural differences etc [89]. In this survey, we analyzed which type of distributed development (e.g., outsourcing, nearshoring, offshoring, onshoring) use distributed/centralized measurement processes.

[5] https://standards.ieee.org/findstds/standard/software_and_systems_engineering.html
[6] https://standards.ieee.org/findstds/standard/instrumentation_and_measurement_all.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

		Response collection method: A close-ended question is presented, "Does your organization involve distributed development of software projects? (Yes, No) In a first follow-up question, we have presented a list of distributed software development methods with checkboxes. In a second follow-up question, we have asked a close-ended question that either they defined distributed measurement processes across all contractors or a centralized measurement process.
Metrics Management	**Metrics collection method**	Based on the analysis of SLR, the metrics collection methods are classified into five categories. 1) using the guidelines of measurement models 2) measurement experts deciding on what measures to collect 3) using automated tools 4) collecting necessary metrics for software process improvement 5) most commonly collected metrics. **Response collection method:** A List of five metrics collection methods is presented with checkboxes. A textbox with ***"Other"*** label is added to make sure that participants could add their in-house metrics collection method or if they believe it is missing in the presented options.
	Categories of metrics data	In the SLR, we have identified following types of metrics discussed in primary studies: duration, cost/budget, effort, size, productivity, time to market, defects, customer satisfaction, return on investment, employee commitment. In this survey, we analyze the use of these metrics in the software industry. **Response collection method:** A List of types of metrics data is presented with checkboxes. A textbox with ***"Other"*** label is added to make sure that participants could add a new type of metrics data or if they believe it is missing in the presented options.
	Measurement glossary	It is observed in SLR that there is a lack of consistent terminology in software measurement domain [55]. Therefore, it is important for software organizations to define measurement glossary for a consistent and complete understanding of metrics, measurement tasks, and activities. In this survey, we analyzed the use of measurement glossary in the software industry. **Response collection method:** Binary response (Yes, No) is collected on the existence of a measurement glossary.
	Measurement repositories for prediction	It is observed in SLR that prediction is the least focus of measurement studies. In addition, there are a few publically available datasets such as Architecture Research Facility, DACS Productivity, NASA Ames, NASE/SEL, Software Reliability, ISBSG Benchmarking datasets. In this survey, we analyze the use of measurement data of past project for prediction in the future projects. **Response collection method:** A lickert-scale question is presented with radio buttons: Never, Rarely, Occasionally, Frequently and Always. The question is presented as, "Does your organization use [measurement] data of past projects to plan/predict future projects? (e.g., software size, number of defects)"
	Real-time metrics	In the SLR, we have observed that there is a lack of discussion on the use of real-time metrics in MPs. Mostly software metrics are discussed in the context of planning phase or at the completion of software development. The real-time metrics (e.g., cyclomatic complexity, dynamic function calls, no of unused objects and variables) help to monitor and control actual software development in progress. In this survey, we analyzed the use of real-time metrics in the software industry. **Response collection method:** An open-ended question is presented with a text box i.e., "Does your organization use software metrics to monitor and control the software development progress in real-time?. Please specify those metrics:"
	Validation of measurement data	It is important to learn that we are actually measuring what we think we are measuring. We have observed in SLR that the metrics validation in software measurement processes is vital yet it is one of the least discussed issues in empirical studies. **Response collection method:** Binary response (Yes, No) is collected on the existence of a metrics validation criteria.

Appendix B

TABLE 3
EXTERNAL FACTORS FOR SUCCESSFUL IMPLEMENTATION OF MEASUREMENT PROCESSES IN SOFTWARE ORGANIZATIONS

Factor	Factor ID	Sub-Factors	Description	References

External Factors (EF)	EF1	Stability of economic sector	The strong pressure of economic situation of the environment might also affect the policies and practices of implementing measurement processes in the organization.	[17]–[21], [90]
	EF2	IT domain	The industrial practices in a domain are affected by strong industrial institutions. The software industry is also affected by influential institutions such as software engineering journals (ACM, IEEE, Science Direct, Springer, Elsevier etc.)	[17]–[21], [73], [90], [91]
	EF3	External stakeholders	The software industry is also affected by software organizations such as IBM, Oracle and Motorola and academic institutes such as Software Engineering Institute (SEI) and customers.	[14]–[18]

TABLE 4
MEASUREMENT ADAPTATION FACTORS FOR SUCCESSFUL IMPLEMENTATION OF MEASUREMENT PROCESSES IN SOFTWARE ORGANIZATIONS

Factor	Factor ID	Sub-Factors	Description	References
Measurement Adaptation (MAD)	MAD1	Measurement culture	It means that measurement processes are actually trusted and used for characterization, evaluation, prediction and improvement of software development processes, products and resource utilization.	[21], [22], [25], [92], [93]
	MAD2	Sustainability of measurement process	The sustainability of measurement process **(MAD2)** is of key importance as most of the MP fail or usually do not last more than a project. It is also observed in the SLR [5], that most of the measurement processes are implemented only for a specific project or product. The sustainability of a measurement process to characterize, evaluate, improve and predict the processes, products and resources in the organization-wide scope might help to gain full advantages of measurement process.	[7], [16]–[21], [94]–[96]
	MAD3	Synchronization between measurement process and software development life cycle (SDLC)	A number of studies stated that synchronization between MP and SDLC **(MAD3)** is critical for effectiveness of measurement process. The measurement processes should be linked with each and every process of SDLC because every process creates or uses data for its characterization, evaluation, prediction and improvement.	[8], [44], [45], [58], [74], [75], [97]
	MAD4	Synchronization between measurement process and software process improvement initiatives (SPI)	The synchronization between MP and SPI (**MAD4)** is considered as a significant success factor. The measurement process is usually related as data collection activity to characterize and evaluate software processes, products and resources but its critical success factor is its ability to help in decision making and software process improvement.	[13], [67], [72], [74], [76], [91], [93], [98]–[102]
	MAD5	Use of measurement standards	The use of measurement standards **(MAD5)** to implement measurement process is considered as significant success factor. A well-defined structure for implementing measurement process is given key importance and measurement standards serve this need by providing guidelines, processes, and tasks of implementing measurement process.	[26], [51], [60], [63], [63], [67], [72], [74], [76], [98]–[100], [103], [104]
	MAD6	Use of measurement models to identify, collect and analyze measures in measurement process	The use of measurement models **(MAD6)** to identify, collect and analyze measures to implement a measurement process is considered as a key success factor, as we identified 35 measurement planning models [5] that are proposed to implement measurement processes in the empirical studies.	[5], [60], [105]
	MAD7	Use of measurement tools	The use of automated tools **(MAD7)** is considered as key success factor for the effective and efficient collection, storage, analysis, and prediction of data for the improvement of software processes.	[24], [25], [44], [58], [74], [97], [98], [104], [106]
	MAD8	Use of measurement data to improve	The essence of a successful measurement process is the ability to effectively use measurement data for decision making **(MAD8)**.	[8], [60], [67], [72],

		decision making	There are multiple studies in the SLR [5] which link the success of a measurement process with its ability to support in decision making.	[76], [93], [97], [98], [107]–[109]
	MAD9	Efficient data collection processes	A well-defined data collection procedure **(MAD9)** is considered as a success factor because only high quality of data (i.e. timeliness, completeness, correctness, accuracy) can support characterization, evaluation, prediction and improvement of software development processes.	[21], [42], [56], [63], [64], [72], [90], [94], [108], [110], [111]
	MAD10	Efficient data analysis processes	An efficient data analysis process **(MAD10)** is considered as a success factor among multiple studies in the SLR [5] because it enables correct decision-making process.	[21], [42], [98], [102], [109]
	MAD11	Reuse of measurement data for planning and prediction	The reuse of measurement data for planning and prediction **(MAD11)** is a major success factor because it enables data-driven predictions that are redoable and comparable as compare to expert judgment that is based on one's experience and tacit knowledge.	[17], [56], [60], [87], [94], [96], [107], [112], [113]

TABLE 5
MEASUREMENT ACCEPTANCE FACTORS FOR SUCCESSFUL IMPLEMENTATION OF MEASUREMENT PROCESSES IN SOFTWARE ORGANIZATIONS

Factor	Factor ID	Sub-Factors	Description	References
Measurement Acceptance	MA1	Improved monitoring and management of software projects	The fundamental purpose of implementing a measurement process is to improve monitoring and management **(MA1)** of the software development processes, products and resources.	[4], [8], [17], [19], [21], [44], [75], [97], [104], [106], [107], [114], [115]
	MA2	Efficient utilization of historical data for project planning and estimation	The measurement process enables storage of measurement data which helps better utilization of historical data for project planning and estimation/prediction **(MA2)**.	[8], [21], [42]–[44], [76]
	MA3	Achievement of organization wide objectives	The measurement process helps in achievement of organization wide objectives such as increased market-share and reworks reduction **(MA3)**.	[17], [21], [25], [64], [64], [76], [76], [99], [101]
	MA4	Completion of tasks	The implementation of a measurement process helps individuals such as project manager, software engineer and software analyst by providing necessary data and to perform their tasks **(MA4)**.	[16], [20], [21], [43], [58], [104], [113], [116]

Appendix C

TABLE 7
EIGENVALUE AND TOTAL VARIANCE

Factors	Initial Eigenvalues: Total	Initial Eigenvalues: % of Variance	Initial Eigenvalues: Cumulative %	Extraction Sums of Squared Loadings: Total	Extraction Sums of Squared Loadings: % of Variance	Extraction Sums of Squared Loadings: Cumulative %	Rotation Sums of Squared Loadings: Total	Rotation Sums of Squared Loadings: % of Variance	Rotation Sums of Squared Loadings: Cumulative %
1	**2.766**	27.662	27.662	2.766	27.662	27.662	2.661	26.605	26.605
2	**2.152**	21.515	49.178	2.152	21.515	49.178	2.129	21.293	47.898
3	**1.707**	17.070	66.248	1.707	17.070	66.248	1.835	18.350	**66.248**
4	.700	6.998	73.246						
5	.557	5.574	78.821						

(Total Variance Explained)

6	.512	5.118	83.938
7	.488	4.882	88.821
8	.449	4.485	93.306
9	.352	3.519	96.825
10	.318	3.175	100.000

TABLE 8
ROTATED COMPONENT MATRIX

	Component		
	1	2	3
MA4	.845		
MA1	.827		
MA3	.795		
MA2	.768		
EF3		.857	
EF2		.849	
EF1		.805	
Mad6			.806
Mad5			.787
Mad4			.733

Appendix D

TABLE 2
EFFECT OF PARTICIPANT'S EXPERIENCE ON THEIR OPINIONS REGARDING THE IMPACT OF EF, MAD AND MA

		Sum of Squares	Degree of Freedom (F)	Mean Square	F	Sig.
External Factors	Between groups	9.060	3	3.020	.375	.771
	Within groups	1576.440	196	8.043		
	Total	1585.500	199			
Measurement Adaptation Factors	Between groups	118.241	3	39.414	1.808	.147
	Within groups	4272.554	196	21.799		
	Total	4390.795	199			
Measurement Acceptance Factors	Between groups	71.668	3	23.889	1.818	.145
	Within groups	2574.887	196	13.137		
	Total	2646.555	199			

TABLE 3
EFFECT OF PARTICIPANT'S ROLE ON THEIR OPINIONS REGARDING THE IMPACT OF EF, MAD AND MA

		Sum of Squares	Degree of Freedom (F)	Mean Square	F	Sig.
External Factors	Between groups	21.736	5	4.347	.539	.746
	Within groups	1563.764	194	8.061		
	Total	1585.500	199			
Measurement Adaptation Factors	Between groups	83.372	5	16.674	.751	.586
	Within groups	4307.423	194	22.203		
	Total	4390.795	199			
Measurement Acceptance Factors	Between groups	110.850	5	22.170	1.696	.137
	Within groups	2535.705	194	13.071		
	Total	2646.555	199			

TABLE 4
EFFECT OF PARTICIPANT'S ORGANIZATION ON THEIR OPINIONS REGARDING THE IMPACT OF EF, MAD AND MA

		Sum of Squares	Degree of Freedom (F)	Mean Square	F	Sig.
External	Between groups	23.901	3	7.967	1.000	.394

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872956, IEEE Access

Factors	Within groups	1561.599	196	7.967		
	Total	1585.500	199			
Measurement Adaptation Factors	Between groups	94.815	3	31.605	1.442	.232
	Within groups	4295.980	196	21.918		
	Total	4390.795	199			
Measurement Acceptance Factors	Between groups	144.480	3	48.160	3.773	***.012***
	Within groups	2502.075	196	12.766		
	Total	2646.555	199			

TABLE 5
MULTIPLE COMPARISON USING POST HOC TUKEY TEST

Organization Type (I)	Organization Type (J)	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
1	2	-.22340	.91924	.995	-2.6054	2.1585
	3	-.41667	.98750	.975	-2.9755	2.1422
	4	1.85000	1.01407	.265	-.7777	4.4777
2	1	.22340	.91924	.995	-2.1585	2.6054
	3	-.19326	.63384	.990	-1.8357	1.4492
	4	**2.07340***	.67450	**.013**	.3256	3.8212
3	1	.41667	.98750	.975	-2.1422	2.9755
	2	.19326	.63384	.990	-1.4492	1.8357
	4	**2.26667***	.76491	**.018**	.2846	4.2487
4	1	-1.85000	1.01407	.265	-4.4777	.7777
	2	-2.07340*	.67450	**.013**	-3.8212	-.3256
	3	-2.26667*	.76491	**.018**	-4.2487	-.2846

*. The mean difference is significant at the 0.05 level.