SACJ 34(1) July 2022
Research Article

The state of systems development methodologies use within the systems development organisations in South Africa

Benson Moyo[a], Magda Huisman[b], Lynette Drevin[b]

[a] School of Mathematical and Natural Sciences, University of Venda, South Africa
[b] School of Computer Science and Information Systems, North-West University (Potchefstroom Campus), South Africa

ABSTRACT
The adoption of the most appropriate systems development methodology (SDM) for a systems development project is one of the most critical decisions in systems development practice. An appropriate SDM for a systems development project is adopted to avoid a misfit of a systems development methodology to the systems development project contextual settings. An SDM misfit is caused by an SDM whose basic assumptions have been violated by the targeted systems development project contextual settings. A misfit may increase the effort required for an SDM in use to fit the systems development project contextual settings. The study aims to investigate the contingent use of SDMs as an approach to minimise SDM misfit in a systems development project. The purpose of the article is to present insights on the contingent use of SDMs derived from a survey of 155 organisations involved in systems development. The results show that both plan-driven and agile SDM class instances are in use and that hybrid SDMs characterise most SDM implementations. The findings have implications for both research and practice as they unify the use of the two SDM classes under the contingent use of SDMs approach.

Keywords: Systems development methodology, systems development methodology contingent use, project contextual stressors

Categories: • **Software and its engineering ∼ software creation and management, software development process management, software development methods**

Email:
Benson Moyo Benson.Moyo@univen.ac.za (CORRESPONDING),
Magda Huisman Magda.Huisman@nwu.ac.za,
Lynette Drevin Lynette.Drevin@nwu.ac.za

Article history:
Received: 25 November 2021
Accepted: 20 March 2022
Available online: 22 July 2022

1 INTRODUCTION

Research and practice present two main perspectives on how the systems development project contextual settings can be dealt with. The first asserts the existence of a universal SDM

Moyo, B., Huisman, M and Drevin, L. (2022). The state of systems development methodologies use within the systems development organisations in South Africa. *South African Computer Journal 34*(1), 103–123. https://doi.org/10.18489/sacj.v34i1.926

SACJ is a publication of the South African Institute of Computer Scientists and Information Technologists. ISSN 1015-7999 (print) ISSN 2313-7835 (online).

that can deal with any systems development project contextual settings. However, systems development is a complex problem, and therefore, there is no single systems development methodology that can address all the systems development project contextual settings (Brooks, 1987; Špundak, 2014). Evidence from practice and research in the systems development field has dismissed the existence of a one-size-fits-all systems development methodology (Aitken & Ilango, 2013). The quest for context-setting independent SDMs for systems development projects has been categorically rejected as infeasible by research and practice evidence (Gill et al., 2018; Marks et al., 2017; Molina-Ríos & Pedreira-Souto, 2020; Moyo, 2020; Serrador & Pinto, 2015). The proliferation of SDMs is perhaps partly a result of an attempt to find a universally applicable SDM to different systems development contextual settings (Diebold et al., 2015). The systems development project contextual setting consists of a unique set of systems development constraints, characteristics, and concerns that have to be met to achieve optimal interaction between an SDM characteristics and the systems development project contextual factors (Moyo, 2020; Špundak, 2014). These systems development constraints, characteristics, and concerns are hereafter referred to as systems development contextual stressors.

The second perspective hypothesises that each set of systems development project contextual stressors is unique, and should be treated as such. The systems development practitioner's challenge is to adopt the most appropriate SDM from a variety of SDMs for each systems development project's contextual stressors. Therefore, the critical factors in the adoption of an SDM for a systems development project are contextual stressors (Špundak, 2014). Furthermore, the adoption of an SDM is not a once-off event nor a straightforward process. SDM adoption is the outcome of a decision process that involves continuous SDM use assessment and evaluation of available alternatives (Moyo, 2020; Špundak, 2014). The adoption of an SDM for a systems development project may not be as ideal and logical as expected due to contextual stressor framing or misinterpretation (Ashok et al., 2012). Furthermore, the adopted SDM implementation may vary from one system development project to another or even within the same systems development project (Schmidt, 2016; Špundak, 2014).

The systems development practitioner is mostly presented with two classes of SDMs from which they can adopt an appropriate SDM, namely the plan-driven SDM and the agile SDM classes. The plan-driven SDM class emphasises the freezing of the system development project scope. In contrast, the agile SDM class emphasises the freezing of cost, schedule, and quality, with scope regarded as a variable. The two classes have been compared and contrasted through project contextual stressors such as requirements dynamics (Boehm, 2004), systems development project type (Dyba & Dingsoyr, 2009), addressing change (Boehm, 2004), and organisation culture (Nerur et al., 2005). Both classes have their strengths and limitations (Henderson-Sellers et al., 2014; Janes & Succi, 2012; Špundak, 2014) when evaluated on specific systems development project contextual stressors.

Research on agile SDM has received extensive attention (Abrahamsson et al., 2009; Dingsøyr et al., 2012; Tam et al., 2020), and some of the studies seek to prove the superiority of the agile SDM class over the plan-driven SDM class (Janes & Succi, 2012; Jiang & Eberlein, 2008; VersionOne, 2018). The comparison to prove that one SDM class is superior to the other in

https://doi.org/10.18489/sacj.v34i1.926

all systems development project situations has been rejected as not objective and counterproductive (Jiang & Eberlein, 2008). Each project is unique and it has to be treated as such, by adopting an SDM appropriate to its contextual stressors, and not simply adopting an SDM because it belongs to a specific SDM class. Different SDMs can be combined to achieve synergies that would be impossible to achieve with a single SDM (Janes & Succi, 2012). Even though the plan-driven SDM class is well-established, it requires ongoing attention to improve. The agile SDM class is intended to complement, rather than replace, the plan-driven SDM class (Janes & Succi, 2012).

In South Africa, the Agile Africa conference is an annual event organised and hosted by the Joburg Centre for Software Engineering (JCSE) that advances the agile SDM class (JCSE, 2016). There is a gap in empirical evidence concerning the current state of SDMs in use, in particular within the South African systems development industry. This article investigates and presents empirical findings on SDMs in use and their pattern of use within the South African systems development industry. The findings are relevant for both practice and research since they reveal a repository of SDMs and their use patterns. The findings allow practitioners to evaluate and compare their current SDM(s) with local and international trends to improve SDMs and their application. Researchers may have the opportunity to gain an unbiased perspective on the use of both the agile SDM class and the plan-driven SDM class in systems development projects. The findings could advance understanding the adoption and use of SDMs based on project contextual stressors.

The overall research question for the study is:

How can the contingent use of SDMs be investigated?

The overall research question is further subdivided into two sub-research questions, which are as follows:

RQ1: What systems development methodologies are in use within the South African systems development industry?

RQ2: What is the SDMs use pattern in systems development projects within the South African systems development industry?

This article has seven sections. A conceptual definition of an SDM as used in the article is proposed in the next section. The evolution of SDMs is presented in the third section. The research design and methodology for the study are described in section four. Results are presented in section five and section six discusses the results. Finally, conclusions and recommendations are presented in section seven.

2 CONCEPTUAL DEFINITION OF SDM

There has been a considerable effort to arrive at an authoritative definition for systems development methodology (Avison & Fitzgerald, 2003; Geambaşu et al., 2011; Huisman & Iivari, 2006; Iivari et al., 2000). A proposal of system development methodology definition that has a broader scope that encapsulates most of the definition conceptualisations found in literature has been presented (Huisman & Iivari, 2006). The definition provides a perspective based on the components of a systems development methodology that are not restricted to instances of systems development methodology classes (Iivari et al., 2000). The definition presents a unique approach to the definition of systems development methodology found in the literature that is neither biased towards the plan-driven systems development methodology class nor the agile systems development methodology class. The definition provides a systematic way of understanding, comparing, and evaluating systems development methodologies. This article's working definition adopts the definition proposed in Huisman and Iivari, 2006 as follows:

> A system development methodology is a dynamic framework for developing systems consisting of the systems development approach, the systems development method(s), the systems development process model(s), and the systems development technique(s) guided and uniquely shaped by a philosophy.

This definition can be used as a criterion against which systems development methodologies are identified, assessed, and evaluated.

3 SDM EVOLUTION

The evolution of systems development methodologies is classified into four eras: the pre-SDM era, the early-SDM era, the SDM era, and the post-SDM era or SDM reassessment era (Avison & Fitzgerald, 2003). Each era is characterised by a specific focus on the systems development contextual stressors. The pre-SDM era was characterised by systems development based on the build and fix approach (Aitken & Ilango, 2013). There was significant dependence on the experience and insight of individual experts and management. When systems complexity increased, this unsystematic and informal systems development practice of the pre-SDM era was challenged to develop systems that could address the systems development contextual stressors. To address the weaknesses of the pre-SDM era, a more disciplined approach to systems development was introduced (Avison & Fitzgerald, 2003). This period marked the early-SDM era. The use of the systems development life cycle (SDLC) as a de jure systems development methodology for all systems development situations characterized the early SDM era (Aitken & Ilango, 2013). However, in reality, the deployment of an SDM is dependent on contextual stressors (Clarke & O'Connor, 2015; Marks et al., 2017). Contextual stressors are not static, universally applicable, or equally important in all systems development projects (Marks et al., 2017). One of the criticisms of the early-SDMs was the underestimation of the variability of the contextual stressors (Gill et al., 2018). The importance and emphasis on a specific contextual stressor may vary from one organisation to another, from one system development project to another, and within the same systems development project over time (Aitken & Ilango, 2013). The SDM era saw a proliferation of SDMs in search of a universally applicable SDM. The failure to find an SDM that was appropriate to all the spectrum of possible systems development contextual stressors configuration led to the post-SDM era. The post-SDM era was characterised by the reassessment of the relevance of SDMs in the systems development field. The current period is the post-SDM era, where SDMs are no longer viewed as complete packages that can be matched with systems development projects. The post-SDM era views SDMs as complementing each other; SDMs can be combined to achieve an ideal fit to systems development project contextual stressors. An SDM achieves an ideal fit if it matches the contextual stressors at the organisational level, systems development project level, and systems development team level (Vijayasarathy & Butler, 2016). Each SDM is suitable for addressing systems development contextual stressors that do not violate its underlying assumptions (Clarke & O'Connor, 2015).

Systems development practitioners often tailor systems development methodologies to fit the specific circumstances of a systems development project (Gill et al., 2018; Serrador & Pinto, 2015). Each systems development project is unique, and the choice of an SDM or a variant thereof is contingent on contextual stressors (Henderson-Sellers et al., 2014). It is not common for an SDM to be used rigidly as per its declared use (Henderson-Sellers et al., 2014; Serrador & Pinto, 2015). In practice, each SDM, even the one regarded as the most appropriate, is tailored (Henderson-Sellers et al., 2014) or adapted (Diebold et al., 2015) or combined with another SDM (Cooper, 2014) to suit specific project contextual stressors (Henderson-Sellers et al., 2014). In the light of the preceding, this research investigated SDMs in use and the SDMs use pattern within the South African systems development industry.

4 RESEARCH DESIGN AND METHODOLOGY

This research is based on a comprehensive study that uses contingency theory and the innovation diffusion theory to investigate the contingent use of systems development methodologies in South Africa (Moyo, 2020). The underlying principle of contingency theory within the context of systems development methodologies is based on the proposition that no one SDM can address all systems development contexts (Henderson-Sellers et al., 2014). An SDM's suitability is defined as the temporal achievement of an ideal fit between an SDM and the contextual stressors. To maintain this optimal fit, an SDM must be adapted, tailored, or substituted in response to the dynamics of the contextual stressors. The contingency theory's Task-Technology Fit model (TTF) is used in this study. Task-technology fit (TTF) is the correspondence between the task requirements and the functionality and features of a contingent innovation (Goodhue & Thompson, 1995). The antecedents of TTF are the interactions between contingent innovation and contextual stressors. The contingent innovation in this study is an SDM. Therefore the TTF construct provides a framework to investigate the SDM choices developers make for a project based on their experience.

With regards to TTF, after using an SDM, a developer gains direct performance experience with it. The performance experience evidence is evaluated against performance expectations (Goodhue & Thompson, 1995). When discrepancies between expected and actual performance are observed, the developer may reconsider the initial expectations, tailor an SDM to reduce the discrepancies, or discontinue its use entirely (Rogers, 2003). The experience of the developer influences the future use behaviour of an SDM (Goodhue & Thompson, 1995; Rogers, 2003). Experience is related to the knowledge accumulation on how to use and why use a specific SDM (Goodhue & Thompson, 1995; Rogers, 2003). In this study, experience is measured through SDMs intensity of use, vertical SDM use, and horizontal SDM use, and the total number of years the developer has been using a particular SDM.

Contextual stressors in systems development projects evolve and take on different levels of importance at different points in the project's life cycle. Changes resulting from the interaction of an SDM and the contextual stressors necessitate a continuous assessment of previous and current decisions on the appropriateness of an SDM to the contextual stressors. The continuous assessment of previous and current decisions on the appropriateness of an SDM to the contextual stressors can be investigated using the innovation diffusion theory. The innovation diffusion theory describes an innovation-decision process that explains how an innovation is adopted and used (Rogers, 2003). The innovation-decision process can be used to investigate decision options that a developer can take to adopt an SDM, use an SDM as-is, tailor and use an SDM, or create and use an appropriate alternative SDM.

The TTF model (Goodhue & Thompson, 1995) and the innovation-decision process (Rogers, 2003) form the basis for the creation of survey questionnaire items.

The research methodology selection is informed by the positivist research paradigm (Saunders & Lewis, 2019). According to the positivist research paradigm, scientific study is based on observable and repeatable facts, and the phenomenon under study does not necessitate the presence of an observer (Saunders & Lewis, 2019). The use of SDMs and their use patterns are independent of the researcher. As a result of the structured nature of the positivist research paradigm, it was considered appropriate to address the research questions posed for this study. The survey was used as the research method (Oates, 2006) and a questionnaire was developed as a survey data-generating instrument.

The survey questionnaire consisted of demographic questions, SDM adoption questions, and SDM deployment questions. Some survey questionnaire item responses ranged from 1, representing totally disagree, to 6, representing totally agree.

Systems development organisations were the target population. However, at the time of the study, the Department of Trade and Industry (DTI) did have a separate economic sector allocation for the systems development industry in South Africa (DTI, 2008; MICT SETA, 2017). This presented a challenge in establishing an authoritative total number of systems development organisations in South Africa. The possible systems development organisations were identified through their web presence and some were identified from the ITWeb directory of companies website (ITWeb, 2019). A random list-based frame sampling technique was used to establish the sample size. A total of 573 systems development organisations were considered eligible to participate in the survey and were invited to participate. The refusal rate was 35.6%. A questionnaire package consisting of a consent letter and a self-administered questionnaire was sent to each one of the 369 eligible organisations that agreed to participate in the survey. The first preference was the systems development project manager. However,

https://doi.org/10.18489/sacj.v34i1.926

in the case of the systems development project manager not being available, other systems development practitioners were co-opted to complete the questionnaire. The unit of analysis is the organisations whereas the unit of inquiry is the systems development practitioners.

In total, 162 questionnaires were completed and returned, giving an acceptable response rate of 28.3% (Sekaran & Bougie, 2016). The first cycle of data analysis constituted data cleaning. 155 (27.1%) were usable and 1.2% of the received questionnaires were discarded due to missing key data values. The discarded cases were within the acceptable data loss range (Bannon, 2015). The Cronbach's alpha was used to indicate good internal consistency of the items in the scale, in which all the items indicated Cronbach's alpha greater than 0.7. The dimensionality of the scale was determined by Factor Analysis. The second cycle of data analysis constitutes descriptive statistics.

The size of an organization, among other important characteristics, determines the type and amount of resources available, as well as communication protocols. The respondents came from organisations of varying sizes. The majority (42.6%) of the respondents came from organisations with 251 or more employees followed by organisations with 51-250 employees which constituted 33.5% of organisations and lastly, organisations with 1-50 employees constituted 23.9% of the total organisations.

5 RESULTS

The following subsections present the empirical results of the study.

5.1 Respondents' roles in systems development projects

A summary of roles is presented in Table 1. The roles in systems development projects are each associated with accountability in a team, systems development project, or in an organisation as a whole. Table 1 represents the role of respondents in systems development projects. The systems development managers were 38.1% and the systems analysts were 26.5% of the total respondents.

Table 1: Respondents' roles in systems development projects

Respondents' role in systems development projects	Percentage
Systems Development Manager	38.1%
Systems Developer	21.2%
Systems Designer	5.2%
Systems Analyst	26.5%
Other	9.0%

5.2 Experience of respondents in the use of SDMs

The respondents had varying levels of experience in the use of systems development methodologies. The experience of the respondents is associated with the technical knowledge on SDMs acquired over the years. The majority (81.3%) of the respondents had experience of six years and above, as shown in Table 2. The significant experience in the systems development practice of the respondents strengthened the relevance of the data collected in terms of its validity and generalisability.

Table 2: Respondents' experience in the use of SDMs

Experience of respondents in the use of SDMs in systems development projects	Percentage
0-5 years	18.7%
6-10 years	21.9%
11-15 years	27.7%
16-20 years	20.7%
21 or more years	11.0%

5.3 The size of the systems development team

The majority of the systems development project teams (78.7%) had more than five members at the time of the study, as shown in Table 3. The largest systems development project teams had more than 50 members representing about 8.4% of the total number of respondents. The small systems development project teams (between one and five team members) were 21.3% of the total number of respondents.

Table 3: Systems development project team size

Systems development project team size SDMs in systems development projects	Percentage
1-5 members	21.3%
6-15 members	43.2%
16-30 members	20.6%
31-50 members	6.5%
51 or more members	8.4%

https://doi.org/10.18489/sacj.v34i1.926

5.4 Systems development project team members involved in the selection of an SDM in an organisation

One of the questions in the questionnaire aimed at establishing the number of people responsible for the SDM selection decision-making process. The results are presented in Table 4 which shows that 43.9% of the respondents state selecting SDMs as a responsibility of a few members (between one and five team members) of the systems development team.

Table 4: Systems development team members involved in the selection of SDMs

Number of systems development project team members involved in SDM selection	**Percentage**
1-5 members	43.9%
6-15 members	40.0%
16-30 members	5.8%
31-50 members	4.5%
51 or more members	5.8%

5.5 Decision-making practices in systems development projects

Respondents were asked to indicate their perception of systems development failure and the level of accountability per role. The question sought to understand the responsibility and accountability associated with the systems development roles in a systems development project. The assumption is that the contribution to the selection of a systems development methodology is associated with the level of responsibility for the actions and decisions embedded within a role. Figure 1 shows that the systems development managers and the systems analysts bear most of the accountability for systems development projects.

5.6 SDM adoption approaches and SDM use patterns

The results for SDM adoption approaches are displayed in Table 5, and respondents were allowed to indicate more than one approach. The majority (78.7%) of organisations indicated that their SDM selection was informed by best practices and experience. The SDM adoption based on adoption frameworks and the "Other" option was chosen by 11% and 0.6% of respondents, respectively. The adoption frameworks assist systems development project teams to adopt SDMs based on the assumptions made by the adoption frameworks on systems development project contextual stressors. However, these framework assumptions may not hold in some situations. Notably, 24.5% of the organisations selected SDM adoption based on guidelines or policies for adopting SDMs.

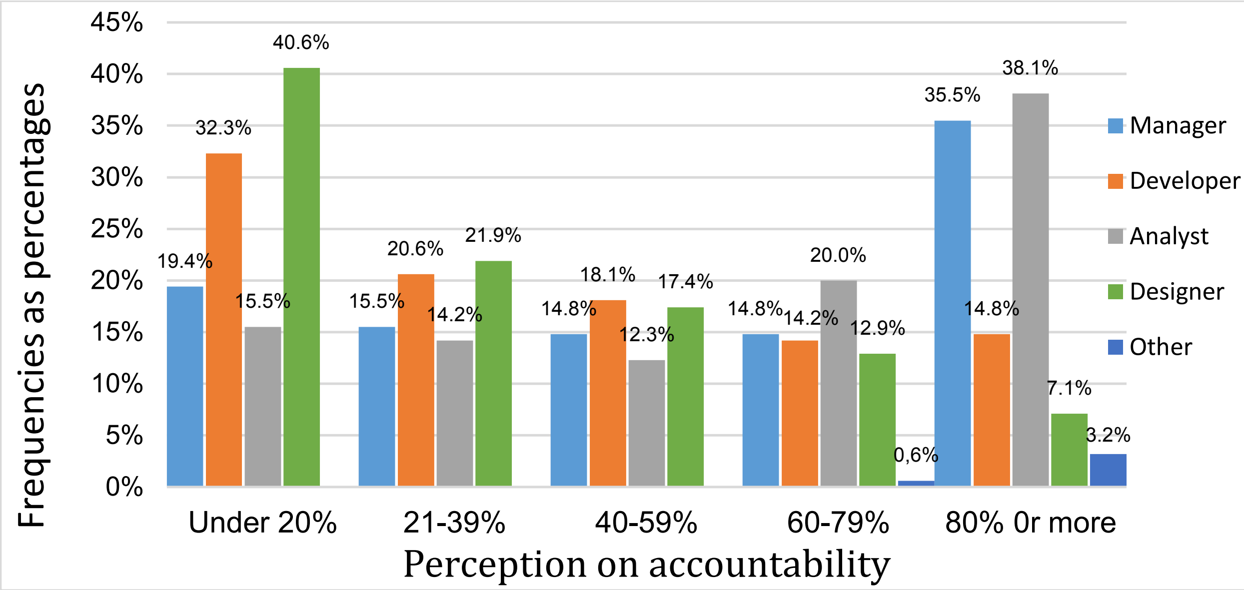

Figure 1: Level of accountability per role in a systems development project.

Table 5: SDM adoption approaches

SDM adoption approaches	Percentage
Guidelines and polices	24.5%
SDM adoption frameworks	11.0%
Based on SDM adoption best practices	78.7%
Other	0.6%

5.7 SDM use patterns

More than three-quarters (77.4%) of the respondents adopted an SDM and tailored it for each systems development project situation. 42.6% of the respondents did not only adopt a single SDM but a set of SDMs and combined them for each systems development project situation, while 39.4% adopted an SDM from a standard set of SDMs for each systems development project situation. Finally, 17.4% create an alternative SDM for each systems development project situation. Table 6 shows that no respondents indicated non-use of SDMs in their systems development projects.

5.8 Horizontal use of SDMs

Table 7 shows that not even one (0.0%) of the respondents indicated non-use of SDM knowledge in systems development projects in their organisation. The question gathered information on the horizontal use of SDMs, that is, the proportion of people using SDM knowledge

https://doi.org/10.18489/sacj.v34i1.926

Table 6: SDM use patterns

SDM use patterns	**Percentage**
Not using any SDM	0.0%
Create a new SDM for each systems development project	17.4%
Select an SDM from a standard set of SDMs for each systems development project situation	39.4%
Select an SDMs from a standard set of SDMs for each systems development situation	42.6%
Select an SDM from in-house developed SDMs for each systems development situation	70.3%
Select an SDM and tailor it for each systems development situation	77.4%

across systems development projects in organisations. The results reveal that more than two-thirds (67.1%) of the respondents indicated that the intensity of SDM knowledge use across systems development projects in their organisations was above 60%.

Table 7: Horizontal use of SDMs in organisations

Horizontal use of SDMs	**Percentage indicating the SDM knowledge application across systems development projects in an organisation for the interval**
0%	0.0%
1-20%	1.3%
21-40%	7.1%
41-60%	24.5%
61-80%	34.2%
Over 80%	32.9%

5.9 Vertical use of SDMs

The question gathered information about the vertical use of SDM, that is, the intensity of SDM knowledge application in each systems development project within an organisation. A summary of the results is presented in Table 8. The results indicate that the application of SDM knowledge is above 60% in 65.2% of the organisations that participated.

https://doi.org/10.18489/sacj.v34i1.926

Table 8: Vertical use of SDMs in organisations

Vertical use of SDMs	Percentage indicating the intensity of SDM knowledge application in each systems development project in an organisation for the interval
0.0%	0.0%
1-20%	2.6%
21-40%	1.2%
41-60%	27.1%
61-80%	40.0%
Over 80%	25.2%

5.10 Contingent use of SDMs

A question was posed to measure the variability of SDM implementation between system development projects and within the same systems development project. The variability in the use of SDM indicates the extent of the contingent use of SDMs in systems development projects. The responses are summarised in Table 9. The results indicate 63.9% of the respondents completely agreed that they adopt and tailor SDMs for each systems development project ($M = 4.0$, $SD = 1.11$), 95% CI $[3.8069,4.1587]$, 54.9% create alternative SDMs ($M = 3.8$, $SD = 1.71$), 95% CI $[3.5153, 4.0589]$, and 33.9% use SDMs without any modification ($M = 3.0$, $SD = 1.70$), 95% CI $[2.7564,3.2952]$.

Table 9: Contingent use of SDMs in organisations

N = 155	Mean	Std Dev	Frequencies as percentages					
			1	2	3	4	5	6
SDM tailoring	4.0	1.11	1.3	7.1	27.7	26.5	29.7	7.7
Create alternative SDMs	3.8	1.71	12.3	14.8	18.1	14.2	18.1	22.6
SDM use as-is	3.0	1.70	19.4	31.0	16.8	7.1	12.3	13.5

5.11 SDMs in use within organisations

A list of plan-driven SDM class instances and agile SDM class instances was presented to the respondents to indicate SDM instances their organisations were using. The aim was to gather data related to the specific SDMs used within the systems development organisations and the intensity of use of the SDMs. The responses are summarised in Table 10. The intensity of use for Scrum SDM indicated the highest calculated mean score of 4.58 and had a low standard deviation of 0.973. The low standard deviation concerning the calculated mean scores indicated a small variability on observed data, which meant that the mean scores were closer to the observed data (Hair et al., 2008). The “Other” option data and the data for Yourdon Systems Method (YSM) were removed for further analysis as constituted outliers in the data set (Hair et al., 2008).

Table 10: SDMs in use within organisations

N = 155	Mean	Std Dev	Frequencies as percentages					
			1	2	3	4	5	6
Other	5.50	0.707	0.0	0.0	0.0	0.6	0.5	0.0
Scrum	4.58	0.973	1.3	2.6	6.5	29.0	47.7	12.9
Rapid Application Development	3.85	1.144	6.5	4.5	20.6	34.8	32.9	0.6
Rational Unified Process (RUP)	3.68	1.791	23.9	4.5	9.0	20.0	27.7	14.8
In-house developed	3.68	1.347	12.3	5.2	18.7	32.9	27.7	3.2
PRINCE2 (Projects In Controlled Environments)	3.65	1.126	6.5	6.5	28.4	32.9	25.8	0.0
Water-Scrum-Fall	3.32	1.329	10.3	15.5	29.0	29.0	9.0	7.1
Microsoft Solutions Framework (MSF)	3.28	1.540	20.6	10.3	20.0	23.9	20.0	5.2
Kanban	3.19	1.221	11.0	14.0	36.1	21.9	14.8	1.3
Structured Systems Analysis and Design Methodology (SSADM)	3.16	1.760	27.7	11.6	16.1	19.4	11.6	13.5
XP (Extreme Programming)	3.10	1.252	13.5	18.1	27.7	27.1	12.8	0.6
Crystal family	2.94	1.323	21.9	12.9	25.8	29.0	9.7	0.6
Jackson Systems Development (JSD)	2.51	1.388	36.1	14.8	20.0	20.0	9.0	0.0
Merise	2.20	1.281	39.4	22.6	26.5	5.2	3.2	3.2
Yourdon Systems Method (YSM)	1.19	0.507	86.5	8.4	5.2	0.0	0.0	0.0

A Cronbach’s alpha level for these thirteen items was 0.801, suggesting good internal consistency reliability within the items (Hair et al., 2008). After performing PCA suitability tests, the Kaiser-Meyer-Olkin measure of sampling adequacy (KMO) generated a value of 0.832, which is regarded as great (Hair et al., 2008). The results of Bartlett’s Test of Sphericity, $\chi^2(78) = 475.522$, $p < 0.0001$, indicated that the correlations between the subscale items were adequate (Hair et al., 2008). The Kaiser criterion extracted three components with eigenvalues greater than 1 (Hair et al., 2008), explaining a total of 51.4% of the variance. The scree plot showed a clear break between the third and the fourth components, suggesting the retention of the three components.

The Promax with Kaiser Normalization rotation was performed on the three components to generate a pattern of loadings that was easy to interpret without changing the underlying solution. The three components were subjected to Cronbach's alpha level analysis, and the first component had the highest Cronbach alpha level of 0.757, followed by the second component with a Cronbach alpha level of 0.723 and the third component had a low Cronbach alpha level of 0.489. However, by treating Scrum SDM as a single item component and removing it from the third component, the Cronbach's alpha level of the third component increased to 0.502. The resulting four components were named as: 1) less used known SDMs, 2) Adaptive SDMs, 3) Popular structured SDMs, and 4) Scrum SDM. This information is presented in Table 11.

Table 11: Component structure of SDM intensity of use

SDM intensity of use (F0)	**Less used known SDMs (F1)**	**Adaptive SDMs (F2)**	**Popular structured SDMs (F3)**	**Scrum SDM (F4)**
All items	XP (Extreme Programming)	Rapid Application Development (RAD)	Structured Systems Analysis and Design Methodology (SSADM)	Scrum SDM
	Kanban	PRINCE2	Water-Scrum-Fall	
	Jackson Systems Development (JSD)	In-house developed	Rational Unified Process (RUP)	
	Crystal family Merise			
	Microsoft Solutions Framework (MSF)			
Cronbach α	**0.757**	**0.723**	**0.502**	**-**

5.12 The duration of the current SDM

The respondents were asked how much time they had spent using the SDM. The results presented in Table 12 show that 71.6% of the respondents indicated that they had been using their SDM for at least three years.

6 DISCUSSIONS OF RESULTS

More than a third (38.1%) of the respondents held senior positions in systems development projects in their organisations. The majority (79.4%) of the respondents had the experience of at least six years in systems development projects. Experience is related to knowledge accumulation on how to use and why use specific SDMs (Rogers, 2003). The study found a

Table 12: Period the current SDM has been in use

Time interval	Percentage
Less than 1 year	12.9%
1-2 years	9.0%
3-5 years	22.6%
6-10 years	33.5%
Over 10 years	15.5%
Not known	6.5%

significant positive relationship between the experience of systems development practitioners and the tailoring of SDMs ($r = .36$, $p < .001$). The result is consistent with the innovation diffusion theory, which states that innovations are adopted, used, and tailored based on their context (Rogers, 2003). SDMs are adopted and tailored on a project-by-project basis, according to 77.4% of respondents. This result is consistent with the contingency theory, which states that no single SDM can address all system development contexts (Henderson-Sellers et al., 2014).

67.1% of respondents indicated a high level of horizontal use of SDMs in organisations. The high horizontal use of SDMs implies that SDM knowledge is used by the majority of systems development practitioners across all systems development projects in organisations. 65.2 percent of respondents indicated a high vertical use of SDM, indicating the depth of SDM knowledge application among systems development practitioners. Together the horizontal and the vertical use of SDMs indicated the breadth and the intensity of SDM knowledge application in systems development projects. According to the contingency theory, a high horizontal SDM use and a high vertical SDM use are indicators of how appropriate an SDM addresses the task requirements of a systems development project (Goodhue & Thompson, 1995).

When it came to the use of SDMs in systems development projects, the Scrum SDM was the most popular. The Scrum SDM, however, was used as a hybrid SDM, that is, it is combined with other SDM instances from the same agile SDM class or the plan-driven SDM class. These findings are consistent with Kuhrmann et al., 2017 and Rodríguez et al., 2019 who found that agile SDM class instances are combined with plan-driven SDM class instances to form hybrid SDMs. Consistent with other studies (Henderson-Sellers et al., 2014; Janes & Succi, 2012; Moyo, 2020), as well as the innovation-decision process (Rogers, 2003), the study found that an SDM can be selected and used as-is, tailored and used as an emergent SDM, or a new alternative SDM can be created depending on the prioritized specific systems development project contextual stressors.

A crosstabulation analysis revealed that 10% of the systems development managers indicated that their systems development team sizes were less than six members. However, 33.9% of systems development managers indicated that less than six people were involved in the selection of an SDM for a systems development project. In the case of systems developers, 36.6%

indicated that their systems development teams were less than six members. Lastly, 54.5% of the systems developers indicated less than six people being involved in the selection of SDMs. This demonstrates that the selection of SDMs for systems development projects is handled by a smaller group of people than the entire systems development team. The evidence was significant ($\chi^2(df = 4, N = 155) = 10.821$, $p < 0.05$). The results suggest that SDM selection decision was not necessarily devolved to all the members of a systems development project team. This is also shown by the higher responsibility and accountability associated with systems development managers and analysts. In general, organisations use a role-based punitive system where a few individuals make decisions and are held accountable for the outcome. This is mostly associated with the plan-driven SDM management approach.

Given the degree of variability in system development project contextual stressors, it is not surprising that a wide variety of SDMs was indicated as being used in the South African systems development industry. A significant negative relationship exists between the use of Scrum SDM and its use in its original version ($r = -0.26$, $p < 0.01$), as indicated in Table 13. The findings suggest that Scrum SDM is not used as-is, but rather is tailored to the specifics of the systems development project, which is one of the options provided by the innovation-decision process in the innovation diffusion theory (Rogers, 2003). However, a significant positive relationship was found between the Scrum SDM use and the creation of hybrid SDM for a systems development project ($r = 0.28$, $p < 0.01$). Scrum SDM is combined with other SDM instances to gain synergies not possible through the sole use of Scrum SDM (Moyo, 2020). The results also confirmed the assertion made by Martin Fowler that Scrum is not used as proposed (Fowler, 2018). It has also been found that what organisations claim as the use of Scrum SDM in most cases is a variation of Scrum SDM, not the version of Scrum SDM as proposed by its creators (Schwaber, 2010). A significant moderate positive relationship was found between the creation of alternative SDMs and the use of Scrum SDM ($r = 0.30$, $p < 0.01$). The creation of alternative SDM is consistent with the innovation diffusion theory's innovation-decision process. The use of SDMs as originally documented (as-is) had a moderate negative relationship with the Adaptive SDMs use ($r = -0.34$, $p < 0.01$), a negative relationship with the less used known SDMs usage ($r = -0.25$, $p < 0.01$), and a weak negative relationship with the usage of popular SDMs ($r = -0.19$, $p < 0.05$). SDM use as-is is a decision option in the innovation-decision process. The Scrum SDM, Adaptive SDMs, less used known SDMs, and popular structured SDMs, all indicated statistically significant negative relationships being used rigidly according to prescription. Perhaps this is because systems development is a knowledge-intensive process where systems development project details are progressively tailored depending on the level of understanding of the systems development project contextual stressors involved. Each systems development project is unique, therefore, according to the contingency theory (Goodhue & Thompson, 1995) and innovation diffusion theory (Rogers, 2003), SDMs are tailored to the demands of the systems development project contextual stressors. A statistically significant positive relationship was found between SDM tailoring and the use of Adaptive SDMs ($r = 0.22$, $p < 0.01$). This suggests that Adaptive SDMs respond continuously to the changing systems development project contextual stressors. The creation of alternative SDMs had a moderate

Table 13: Correlations of the SDMs in use and the variations in SDM use

Variable	Create alternative SDM	Tailor SDM	SDM use as-is
Scrum SDM	0.297[a]	0.280[a]	-0.261[a]
Adaptive SDMs	0.084	0.218[a]	-0.339[a]
Less used known SDMs	0.210[a]	0.140	-0.245[a]
Popular structured SDMs	0.301[a]	0.106	-0.185[b]

[a] $p < 0.01$
[b] $p < 0.05$

positive relationship with the use of popular structured SDMs ($r = 0.30$, $p < 0.01$), a positive relationship with less used known SDMs ($r = 0.21$, $p < 0.01$). The variability of SDM use based on the contextual stressors is consistent with the contingency theory (Goodhue & Thompson, 1995) and the theory of innovation diffusion theory's innovation-decision process (Rogers, 2003).

7 CONCLUSIONS AND RECOMMENDATIONS

This study looked into the contingent use of SDMs in South African systems development organisations. An overarching research question was developed, then subdivided into two sub-research questions that were answered using empirical evidence from the South African systems development industry.

RQ1: What systems development methodologies are in use within the South African systems development industry?

There was no conclusive evidence to suggest that organisations exclusively use a specific SDM class instance. According to the findings, SDMs cannot be split into two mutually exclusive categories of plan-driven SDM class and agile SDM class in the context of systems development in South Africa. The study dataset indicated the use of hybrid SDMs. Both the agile SDM class instances and the plan-driven SDM class instances coexist as asserted by Janes and Succi (2012). The results indicate that organisations enact base SDMs and create hybrid SDMs as determined by the specific systems development project contextual stressors. The hybrid SDMs are derived from combining the base SDM with SDM components from the same class of SDMs or a different class of SDMs.

RQ2: What is the SDMs use pattern in systems development projects within the South African systems development industry?

The study found that SDMs are used on a case-by-case basis. Depending on the prioritized specific systems development project contextual stressors, an SDM can be selected and used as-is, tailored, and used as an emergent SDM, or a new alternative SDM can be created. An organisation can select a base SDM a priori and tailor it on an ad hoc project-by-project basis (Henderson-Sellers et al., 2014). The base SDM is usually a variant of an instance of either an agile SDM class or plan-driven SDM class or a hybrid SDM (Gill et al., 2018; Henderson-Sellers et al., 2014; Janes & Succi, 2012; Moyo, 2020).

It can be concluded that instances of the plan-driven SDM class and the agile SDM class are used in the South African systems development industry. The use of SDMs varies from system development project to system development project and within a systems development project based on the systems development project-specific contextual stressors that may be prioritized at any given time (Henderson-Sellers et al., 2014; Molina-Ríos & Pedreira-Souto, 2020). As a result, SDMs are used on a case-by-case basis in system development projects. The variations in SDMs implementation make identifying the specific SDM in use difficult, as an SDM instance may be a temporary hybrid of various SDM components from different SDMs or/and SDM class instances. As a result, when viewed in the context of system development contextual stressors, the SDM concept is critical. This is typical of the post-methodology era, in which the SDM is relevant and applicable in a specific project contextual stressors configuration (Marks et al., 2017; Serrador & Pinto, 2015).

Because the findings unify the two SDM classes based on project contextual stressors, they have implications for both research and practice. The findings reveal a repository of SDMs as well as their usage patterns. This gives practitioners a criterion for evaluating and comparing their current SDM(s) with local and international trends in order to improve the SDMs and their use. In addition, researchers are given an unbiased viewpoint on the use of the agile SDM class and the plan-driven SDM class in system development. These findings provide an opportunity to further our understanding of the adoption and use of SDMs based on systems development project contextual stressors.

This study makes no claims about the generalizability of its findings because it is limited to those who voluntarily participated. Respondents in a survey may introduce bias due to differences in their experiences and roles within their organisations. This study can be expanded in the future to look at how the contingent use of SDMs is implemented in system development projects. More empirical research is needed to investigate the SDMs in use at the component level rather than the entire SDM.

References

Abrahamsson, P., Conboy, K. & Wang, X. (2009). Lots done, more to do: The current state of Agile systems development research. *European Journal of Information Systems, 18*. https://doi.org/10.1057/ejis.2009.27

Aitken, A. & Ilango, V. (2013). A comparative analysis of traditional software engineering and Agile software development. *2013 46th Hawaii International Conference on System Sciences*, 4751–4760. https://doi.org/10.1109/HICSS.2013.31

Ashok, A., Bangal, S. & Sumangala, D. (2012). The study of prescriptive and descriptive models of decision making. *International Journal of Advanced Research in Artificial Intelligence*, *1*. https://doi.org/10.14569/ijarai.2012.010112

Avison, D. & Fitzgerald, G. (2003). *Information systems development: Methodologies, techniques & tools*. London, UK: McGraw-Hill.

Bannon, W. (2015). Missing data within a quantitative research study: How to assess it, treat it, and why you should care. *Journal of the American Association of Nurse Practitioners*, *27*. https://doi.org/10.1002/2327-6924.12208

Boehm, B. (2004). Balancing agility and discipline: A guide for the perplexed. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *3026*. https://doi.org/10.1007/978-3-540-24675-6_1

Brooks, F. (1987). No silver bullet essence and accidents of software engineering. *Computer*, *20*(4), 10–19. https://doi.org/10.1109/MC.1987.1663532

Clarke, P. & O'Connor, R. V. (2015). Changing situational contexts present a constant challenge to software developers. *Communications in Computer and Information Science*, *543*. https://doi.org/10.1007/978-3-319-24647-5_9

Cooper, R. G. (2014). What's next? after stage-gate. *Research Technology Management*, *57*. https://doi.org/10.5437/08956308X5606963

Diebold, P., Ostberg, J. P., Wagner, S. & Zendler, U. (2015). What do practitioners vary in using scrum? *Lecture Notes in Business Information Processing*, *212*. https://doi.org/10.1007/978-3-319-18612-2_4

Dingsøyr, T., Nerur, S., Balijepally, V. & Moe, N. B. (2012). A decade of Agile methodologies: Towards explaining Agile software development. *Journal of Systems and Software*, *85*. https://doi.org/10.1016/j.jss.2012.02.033

DTI. (2008). Annual Review of Small Businesses in South Africa 2005-2007 [Department of Trade and Industry (DTI), Pretoria (South Africa)].

Dyba, T. & Dingsoyr, T. (2009). What do we know about Agile software development? *IEEE Software*, *26*. https://doi.org/10.1109/MS.2009.145

Fowler, M. (2018). *The State of Agile Software in 2018. The 10th Annual Agile Australia Conference, 18-19 June, Melbourne* [Accessed on 12 June 2020]. https://martinfowler.com/articles/agile-aus-2018.html

Geambaşu, C. V., Jianu, I., Jianu, I. & Gavrilă, A. (2011). Influence factors for the choice of a software development methodology. *Accounting and Management Information Systems*, *10*.

Gill, A. Q., Henderson-Sellers, B. & Niazi, M. (2018). Scaling for agility: A reference model for hybrid traditional-agile software development methodologies. *Information Systems Frontiers*, *20*, 315–341. https://doi.org/10.1007/s10796-016-9672-8

Goodhue, D. L. & Thompson, R. L. (1995). Task-technology fit and individual performance. *MIS Quarterly: Management Information Systems*, *19*. https://doi.org/10.2307/249689

Hair, J. F., Black, W. C., Babin, B. J. & Anderson, R. E. (2008). *Multivariate Data Analysis* (8th ed.). London, UK: Cengage.

Henderson-Sellers, B., Ralyté, J., Ågerfalk, P. J. & Rossi, M. (2014). *Situational method engineering*. Heidelberg: Springer. https://doi.org/10.1007/978-3-642-41467-1

Huisman, M. & Iivari, J. (2006). Deployment of systems development methodologies: Perceptual congruence between is managers and systems developers. *Information and Management*, *43*. https://doi.org/10.1016/j.im.2005.01.005

Iivari, J., Hirschheim, R. & Klein, H. K. (2000). A dynamic framework for classifying information systems development methodologies and approaches. *Journal of Management Information Systems*, *17*. https://doi.org/10.1080/07421222.2000.11045656

ITWeb. (2019). *ITWeb company directory by industry sector* [Accessed on 25 August 2019]. http://v2.itweb.co.za/index.php?option=com_sobi2%5C&sobi2Task=searchSector&Itemid=176

Janes, A. & Succi, G. (2012). The dark side of Agile software development. *SPLASH 2012: Onward! 2012 - Proceedings of the ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*. https://doi.org/10.1145/2384592.2384612

JCSE. (2016). *Joburg Centre for Software Engineering. The transformation issue Joburg Centre for Software Engineering 2015-2016 Annual Report* [Accessed on 28 November 2020]. https://www.jcse.org.za/wp-content/uploads/2018/05/JCSE_AR_2016.pdf

Jiang, L. & Eberlein, A. (2008). Towards a framework for understanding the relationships between classical software engineering and agile methodologies. *Proceedings - International Conference on Software Engineering*. https://doi.org/10.1145/1370143.1370146

Kuhrmann, M., Diebold, P., Münch, J., Tell, P., Garousi, V., Felderer, M., Trektere, K., McCaffery, F., Linssen, O., Hanser, E. & Prause, C. R. (2017). Hybrid software and system development in practice: Waterfall, scrum, and beyond. *ACM International Conference Proceeding Series*, *Part F128767*, 30–39. https://doi.org/10.1145/3084100.3084104

Marks, G., O'Connor, R. V. & Clarke, P. M. (2017). The impact of situational context on the software development process – a case study of a highly innovative start-up organization. *Communications in Computer and Information Science*, *770*. https://doi.org/10.1007/978-3-319-67383-7_33

MICT SETA. (2017). MICT SETA Sector Skills Plan for 2018-2023 [Media, Information and Communication Technologies Sector Education and Training Authority (MICT SETA), Government Printing Works].

Molina-Ríos, J. & Pedreira-Souto, N. (2020). Comparison of development methodologies in web applications. *Information and Software Technology*, *119*, 106238. https://doi.org/10.1016/j.infsof.2019.106238

Moyo, B. (2020). *The contingent use of systems development methodologies in South Africa* [Potchefstroom: North-West University, (Doctoral thesis)].

Nerur, S., Mahapatra, R. & Mangalaraj, G. (2005). Challenges of migrating to Agile methodologies. *Communications of the ACM*, *48*. https://doi.org/10.1145/1060710.1060712

Oates, B. J. (2006). *Researching Information Systems and Computing*. London, UK: Sage.

Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L. E., Seppänen, P. & Kuvaja, P. (2019). Advances in using agile and lean processes for software development. *Advances in computers* (pp. 135–224). Elsevier. https://doi.org/10.1016/bs.adcom.2018.03.014

Rogers, E. M. (2003). *Information Systems Development: Methodologies, techniques & tools*. NY, USA: Simon; Schuster.

Saunders, M. A. & Lewis, P. (2019). *Research methods for business students* (8th ed.). Pearson.

Schmidt, C. (2016). Agile software development. *Agile Software Development Teams* (pp. 7–35). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-26057-0_2

Schwaber, K. (2010). *On Shoulders of Giants, Interview, Agile Collab* [Accessed on 19 August 2019]. http://mattcallanan.blogspot.de/2010/02/ken-schwaber-on-scrum.html

Sekaran, U. & Bougie, R. (2016). *Research methods for business: A skill building approach*. Haddington, UK: John Wiley and Sons.

Serrador, P. & Pinto, J. K. (2015). Does Agile work? - a quantitative analysis of Agile project success. *International Journal of Project Management*, *33*. https://doi.org/10.1007/s10796-016-9672-8

Špundak, M. (2014). Mixed agile/traditional project management methodology – reality or illusion? [Selected papers from the 27th IPMA (International Project Management Association), World Congress, Dubrovnik, Croatia, 2013]. *Procedia - Social and Behavioral Sciences*, *119*, 939–948. https://doi.org/10.1016/j.sbspro.2014.03.105

Tam, C., da Costa Moura, E. J., Oliveira, T. & Varajão, J. (2020). The factors influencing the success of on-going agile software development projects. *International Journal of Project Management*, *38*. https://doi.org/10.1016/j.ijproman.2020.02.001

VersionOne. (2018). *VersionOne. The State of Agile Survey* [Accessed on 20 October 2020]. https://explore.versionone.com/state-of-agile/12th-annual-state-of-agile-report-overview

Vijayasarathy, L. R. & Butler, C. W. (2016). Choice of software development methodologies: Do organizational, project, and team characteristics matter? *IEEE Software*, *33*. https://doi.org/10.1109/MS.2015.26