Mind the gaps: increasing the impact of IS research on ISD performance improvement

Niamh O Riordan
UCD School of Business
University College Dublin
Dublin 4, Ireland
niamh.oriordan@ucd.ie

Garry Lohan
Lero Research Centre & Whitaker Institute
NUI Galway
Galway, Ireland
garry.lohan@nuigalway.ie

Kieran Conboy
JE Cairnes School of Business & Economics
NUI Galway
Galway, Ireland
kieran.conboy@nuigalway.ie

Abstract

Poor performance has pervaded the last forty years of software development, evident across industry sector, project size, budget, geographic location, system quality and functionality, and exacerbated by increased criticality of Information Technology (IT) in organisational mission and strategy. A significant body of research has investigated the potential of emerging development methodologies to address these shortcomings but the effectiveness of these methods is largely supported by anecdotal evidence. At the same time, metrics and measurement are known to affect ISD performance but the existing literature on Information Systems Development (ISD) metrics is misaligned with practitioners' needs, leading to a lack of clarity about ISD metrics in practice. This paper presents an interdisciplinary literature review on ISD metrics to identify the underlying reasons for this misalignment and evaluate the extent to which existing literature can be used to better understand the impact of emerging software development methodologies on ISD performance.

Keywords: Information systems development, performance improvement, metrics, measurement

1 Introduction

Investments in ISD are substantial but poor performance is a pervasive and significant problem (Bharadwaj, Keil and Mähring, 2009). In the past decade, a significant body of research has investigated the potential of emerging software development methodologies (e.g. XP and Scrum) to address these issues but partly due to uncertainty regarding the applicability of traditional measurement practices (Kulas, 2012), the effectiveness of these methods is largely supported by anecdotal evidence (Lee and Xia, 2010). At the same time, an extensive body of research attests to significance of measurement in software development processes (Scotto *et al.*, 2005; Harjumaa *et al.*, 2008; Hazzan and Hadar, 2008). However, the existing literature on ISD metrics and measurement practices is misaligned with practitioners' needs (Johnson *et al.*, 2005). This has led to a persistent lack of clarity about ISD metrics in practice, which is recognized as one of main causes of poor ISD performance (Jalali and Wohlin, 2011; Olague *et al.*, 2007; Bertoa *et al.*, 2006). To address these issues, this paper presents an interdisciplinary review of the peer reviewed studies on ISD metrics that have been published in leading Information Systems (IS) and Software Engineering (SE) journals from 2001 to 2013. Its main aim is to identify some of the underlying reasons for this misalignment and also to evaluate the extent to which existing literature can be used to better understand the impact of emerging software development methodologies on ISD performance.

The paper begins with a discussion of the role of emerging software development methodologies and software measurement programs in achieving ISD performance improvements, highlighting the need for research on the misalignment of the existing literature on ISD metrics with practitioners' needs. The literature review process and analytical framework are described. In the next section, the analysis identifies and explains the existence of significant gaps in the literature, arguing that these gaps effectively limit the potential impact of the literature on practice. The paper concludes with a discussion of directions for future IS research on ISD metrics.

2 Problem Statement

Information Systems have become ubiquitous and are recognized as important enablers of both global productivity improvements and lifestyle advancements. Thus, investments in ISD are not only significant but are set to increase in the coming years (Gartner, 2013). As a result, ISD practitioners have a growing need to protect their investments in ISD project, to maximise the return on their investments and to ensure that their investments have a significant and sustained effect on their competitive positioning. By extension, IS research that can maximise the return on those investments is of great practical significance.

Nevertheless, practitioners have also struggled to overcome persistent and significant problems with poor ISD project performance: organizations continue to undertake projects that consume more resources than they should before they are either cancelled or turned around, produce systems and platforms that are poorly aligned with stakeholder requirements or non-compliant with demands unique to the domain. Thus, the direct annual cost of ISD project failure has been estimated to be over $6 trillion (Sessions, 2009). In 1995, for example, the Standish Group Report estimated that more than half of all ISD projects would cost nearly twice the originally estimated cost and that 30% would ultimately be cancelled (The Standish Group, 1995). Though a more recent study (CHAOS, 2011) indicates that performance improvements have been achieved, it reports that only 37% of ISD projects meet scope, time and cost goals. Further, that one fifth of all projects (21%) are ultimately cancelled.

The persistence of these problems is due to the numerous challenges faced by IS developers including variability in project size and duration, emerging trends in the areas of open systems, web services and distributed configurations, decreasing product lifecycles, and increasing requirements volatility (Port and Bui, 2008; Petersen and Wohlin 2010). As well as this, there is a lack of strong theoretical and conceptual foundations to many studies of IS development methods (Abrahamsson *et al.*, 2009, Conboy, 2009). In the absence of sound, systematic research, there are few lessons learned across studies, and thus, the existing body of knowledge is somewhat fragmented and inconclusive.

2.1 Agile methods: a possible solution?

In the past decade, a significant body of research has investigated the potential of emerging software development methodologies (e.g. XP and Scrum) to address these issues. These agile methodologies promise to deliver higher customer satisfaction, lower defect rates, faster development times and a solution to rapidly changing requirements by means of a shift in the "centre of gravity" of software engineering from creating a technology-cantered solution toward satisfying the stakeholders (Suryn and Abran, 2003).

Though agile methodologies are now used by the majority of software development teams, the effectiveness of these methods is largely supported by anecdotal evidence and rhetorical argument (Lee and Xia, 2010). This is partly due to uncertainty regarding the applicability of traditional measurement practices in agile settings (Kulas, 2012). More specifically, it is argued that (continuous) measurement of both product and process is inherent in agile software development processes (Kulas, 2012) and that traditional modes of evaluation may either be incompatible with agile values and principles (Hartmann and Dymond, 2006) or cannot be directly transferred into agile development (Nerur and Balijepally, 2007; Kunz *et al.*, 2008). The use of agile methods is also often superficially judged as used or not used, whereas the actual implementation can be subtle, partial and inconsistent, and so categorising a method as used or not used may be overly simplistic (Conboy and Fitzgerald, 2010).

As a result, few agile methods studies have used metrics to evaluate software product or process quality (Olague *et al.*, 2007) and the research on the effectiveness of agile methods "is yet to yield significant systematic and insightful knowledge that can either guide future research or inform effective adoption and use of these methods in practice" (Wang, Conboy *et al.*, 2010). In practice, agile teams therefore continue to struggle to measure performance (Downey and Sutherland, 2013).

2.2 ISD metrics: a possible solution?

In recent years, there has also been a dramatic increase in the number of ISD metrics used to drive ISD performance improvements in practice (Gencel et al, 2013). These metrics, sets of figures or statistics used to measure results (OED, 2013), are used by developers to provide a quantitative basis to support the development, control and validation of software processes and products. Similarly, project managers can also use them to assess status, track risk, uncover problems before they become critical, adjust work flows and tasks, and evaluate the team's ability to control quality. In addition, other organisational stakeholders including customers and program managers can use metrics to better understand and control the software process and products as well as the relationships between them (Basili, 1992).

Though the bulk of the literature on ISD metrics focuses on the metrics themselves, a significant body of research investigates the efficacy of ISD metrics and evaluates their impact on ISD performance (cf. Harjumaa *et al.*, 2008; Hazzan and Hadar, 2008). However, the results of these studies are inconclusive. On the one hand, these studies show that ISD metrics can have a positive impact on project performance by facilitating improved resource estimation (Boehm *et al.*, 2002), improved resource allocation (Basili, 1996) and improved information for decision making (Appari and Benaroch, 2010). On the other hand, they problematise the validity of the metrics themselves in some instances (e.g. Fenton, 1994). (More recently, Meneeley *et al.*, (2012) identify 47 validation criteria to help identify valid metrics). They also reveal that ISD metrics can have a negative effect on ISD performance. For example, Hartmann and Dymond (2006) argue that inappropriate measurement practices can drive dysfunctional behaviours, such as wasting resources and distorting team behaviours in counter-productive ways. Similarly, Harrison (2004) cautions that particular metrics may become obsolete over time or may require different interpretations depending on the given context.

Thus, a number of authors have pointed to this lack of consistency and clarity surrounding ISD metrics (cf. Jalali and Wohlin, 2011; Olague *et al.*, 2007; Bertoa *et al.*, 2006). For example, Johnson et al (2005) argue that practitioners face various barriers in selecting and implementing ISD metrics that are inadequately addressed in the literature, pointing to the existence of a misalignment or gap between ISD metrics research and practice. To address these issues, this study presents a literature review that identifies some of the underlying reasons for this misalignment and evaluates the extent to which existing knowledge of ISD metrics can be used to better understand the impact of emerging software development methodologies (e.g. Agile methods) on ISD performance. Our approach is described in the next section.

3 Research Design

In order to better understand the reasons for the misalignment or gap between the ISD metrics research and practice, we conducted an interdisciplinary review of the literature on ISD metrics. Our approach was informed by both Fink's general recommendations and by the recommendations of Kitchenham *et al.* (2007) and Petersen (2011) in terms of SE reviews. More specifically, we began by defining (i) a search strategy and (ii) the inclusion and exclusion criteria that could be used to govern the selection of specific articles. Once we had identified the relevant articles, we analysed them using an analytical framework in order to both ‘map’ and ‘review’ the literature (cf. Petersen, 2011). More specifically, we mapped the literature by imposing a structure upon it and analysing the distribution of individual studies within that structure and we reviewed the literature by evaluating the evidence within those studies in order to derive particular recommendations to guide future research. Each of these activities is discussed in turn.

3.1 Search strategy and paper selection

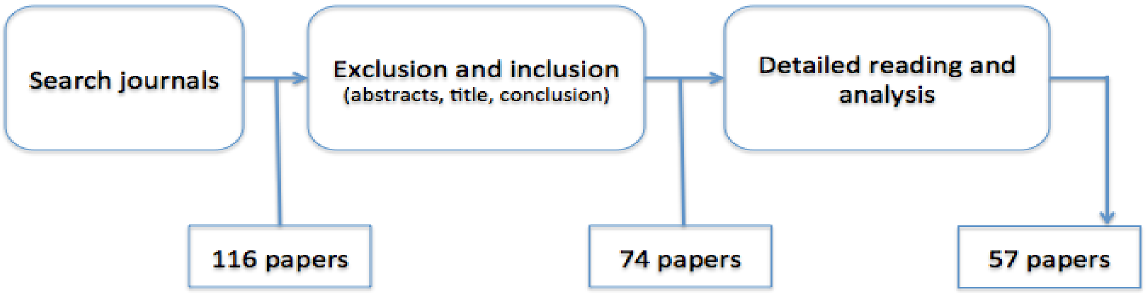
Dyba˚ *et al.* (2007) propose four phases to define a comprehensive search strategy. In the first phase a search string is defined to search for relevant articles. In the second and third phase, inclusion and exclusion criteria are applied to the articles based on title (phase 2) and abstract (phase 3). Finally, the relevant articles are retrieved for an in-depth evaluation. In this study, we began by identifying the leading Information Systems (IS) and Software Engineering (SE) journals. It was necessary to consider both schools as they both occupy the domain of ISD in terms of research and practice (Barry and Brown, 2003). In IS, we analysed eight leading IS journals in the Senior Scholars' basket of journals: Management Information Systems Quarterly (MISQ), Information Systems Research (ISR), Journal of the Association for Information Systems (JAIS), European Journal of Information Systems (EJIS), Information Systems Journal (ISJ), Journal of Management Information Systems (JMIS), Journal of Strategic Information Systems (JSIS), Journal of Information Technology (JIT). We were less familiar with the leading journals in SE so our selections were informed by similar literature reviews in the area (cf. Petersen, 2011). More specifically, we analysed the ACM Transactions on Software Engineering and Methodology (TOSEM), IEEE Transactions on Software Engineering (TSE), IEEE Software (IEEESw), Journal of Software Evolution and Process - formerly Software Process: Improvement and Practice - (JSEP). We subsequently added two journals because our analysis of the literature revealed that many of the most heavily cited articles on ISD metrics in both IS and SE appear in these journals. These were the Journal of Systems and Software (JSS) and the Journal of Information and Software Technology (IST).

Study selection criteria are intended to identify those primary studies that provide direct evidence about the research question. In order to reduce the likelihood of bias, we defined the selection criteria prior to the review. We developed a protocol which specified the search strategy, search string, and inclusion and exclusion criteria. We implemented the research protocol by searching all articles published in the chosen journals between 2001, the year the Agile Manifesto was released, and 2013. The titles, abstracts and keywords of the journal papers were searched and the search string used was formulated as follows (search strings were adapted to match the individual requirements of each of the search engines used):

((“metric” or “measure” or “indicator”) AND (“agil” or “incremental” or “scrum” or “extreme” or “software develop*” or “information systems develop” or “iterative”))*

The abstracts, titles and keywords of the result set were merged and duplicates removed. In order to minimise any bias in the search procedure and to ensure that the result set covers the literature on ISD metrics, we preselected a set of papers that should be picked up by a thorough search procedure and then checked to see that they were returned by the search. We also analysed the reference lists of the included articles to identify any additional papers that might have been inappropriately excluded from the search. Figure 1 shows the number of studies identified at each stage.

In phase 2 and 3 of the study, inclusion and exclusion criteria were applied to the titles and abstracts of the papers (see Figure 1). As noted by Brereton (2007) the standard of abstracts is often too poor to rely on when selecting primary studies. We therefore examined the conclusions of papers at this stage. The reading of titles, abstracts and conclusions was done by two researchers to minimise the possibility of bias when including or excluding articles. When there was any disagreement we discussed the issues until we reached agreement.



Inclusion criteria: *The paper appears in one of the journals listed above*

AND

(The paper addresses one or more ISD metrics, defined as figures or statistics used to measure results in a substantial way	*OR*	*The paper addresses one or more ISD measures, defined as standard units used to express the size, amount or degree of some aspect of an ISD process or product, in a substantial way*	*OR*	*The paper addresses one or more ISD indicators, defined as a thing that indicates the state or level of some aspect of an ISD process or product, in a substantial way)*

Exclusion criteria:

The paper is an introduction to a special issue or a book review	OR	*The paper is not centrally concerned with ISD metrics or measurement issues*

Figure 1: The process of applying inclusion and exclusion criteria to papers analysed in this review

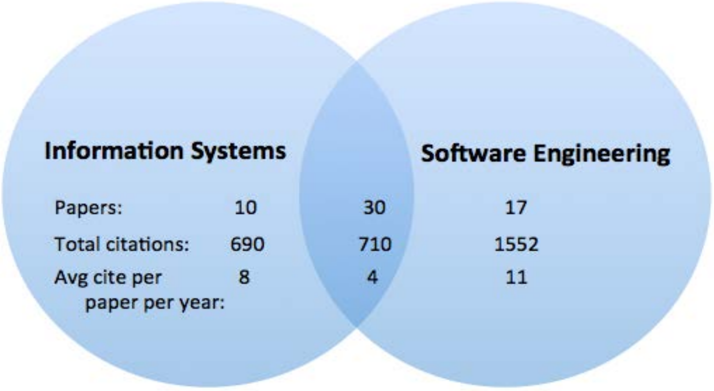
3.2 In-depth evaluation

At this stage, the relevant articles were retrieved for an in-depth evaluation. A predefined extraction form was used to extract data from each of the 57 primary studies included in this review. In the remainder of this document, sources that are referenced using square brackets are part of the literature review whereas sources referenced in the normal fashion were not. The following details were extracted: Bibliographic reference: author, year, title, source; Number of citations; primary research focus: technical, non-technical; Primary focus of the paper: validation, selection, utilization, evaluation; Metrics used; Research method: qualitative, quantitative (experiment, survey); Data collection method: questionnaire, observation, interview. We also extracted the type of metric used. Two types of metric are commonly distinguished in the literature: (i) process metrics are aimed at reducing risk and gaining confidence in the process as the ISD process progresses and (ii) product metrics are aimed at ensuring a high-quality product (Hazzan and Hazard, 2008). In our initial evaluation of the literature, we found that a third type of metric is prevalent. We refer to this as a product-in-process metric. This type of metric is calculated using the product as it is being created during the ISD process. As such, our analysis distinguishes between three types of metric: (i) process metrics, (ii) product-in-process (mid-process or interim product) metrics, and (iii) product-in-use (post release) metrics. Finally we assessed the research design.

The evaluation was also conducted by two researchers working independently at first and then collaboratively. Where ambiguities arose, the argumentation presented in the paper itself was analysed and the apparent perspective of its authors was adopted. In [12] for example, one of the researchers felt that architectural quality over time could be classified as product-in-use metric but the intent of the paper was to attempt to "foresee the possible impact of changes before their realization" and so the metric was classified as a product-in-process metric.

4 Findings and Discussion

We analysed 57 publications, the bulk of which were published in the two journals straddling Information Systems (IS) and Software Engineering (SE). Though the number of relevant publications is small given the duration and number of journals studied, Figure 2 illustrates that the significance of the topic manifests in terms of citations. In the field of IS, each of the papers receives an average of eight citations per year taking into account the number of years since publication. This is significantly greater than the impact factor for each of the IS journals included in the sample and cannot be explained in terms of outliers: all ten papers had an average of at least three citations per year.



Information Systems (n=10)				Info Sys / Software Eng (n=30)		Software Engineering (n=17)	
EJIS (1.59)	n=3	MISQ (4.66)	n=1	JSS (1.24)	n=19	TSE (3.37)	n=11
ISJ (1.38)	n=3	ISR (2.14)	n=0	IST (1.42)	n=11	IEEESw (2.58)	n=1
JMIS (1.26)	n=2	JAIS (2.25)	n=0			TOSEM (1.55)	n=0
JIT (3.53)	n=1	JSIS (1.50)	n=0			SWEP (1.27)	n=5

Figure 2: The configuration of found papers across IS and SE

Figures 2(a) and 2(b) present an overview of the literature. Figure 2(a) compares the fields, analysing the metric type(s) studied (Row 1) and the research design employed (Row 2) in each field. Our analytical framework distinguishes between several experimental designs that may not be familiar to IS researchers including in vitro, in silico, in vivo and in situ studies(see above for further information). In our assessment of research design, we wanted to be able to clearly distinguish the different types of experimental methods that are used in the studies so we distinguished between eight different research designs. In particular, our analytical framework distinguishes between *in vitro, in silico, in vivo* and *in situ* studies, which may not be familiar to IS researchers. *In vitro* studies use data gathered from industrial setting but generate observations outside of those settings; *in silico* studies use simulated data rather than data from industrial settings for simulation, *in vivo* studies approximate industrial settings to an extent (e.g. by using student participants) and *in situ* studies are carried out in an industrial setting. Figure 2(b) examines the literature as a whole, analysing the research designs used to study each category of metric.

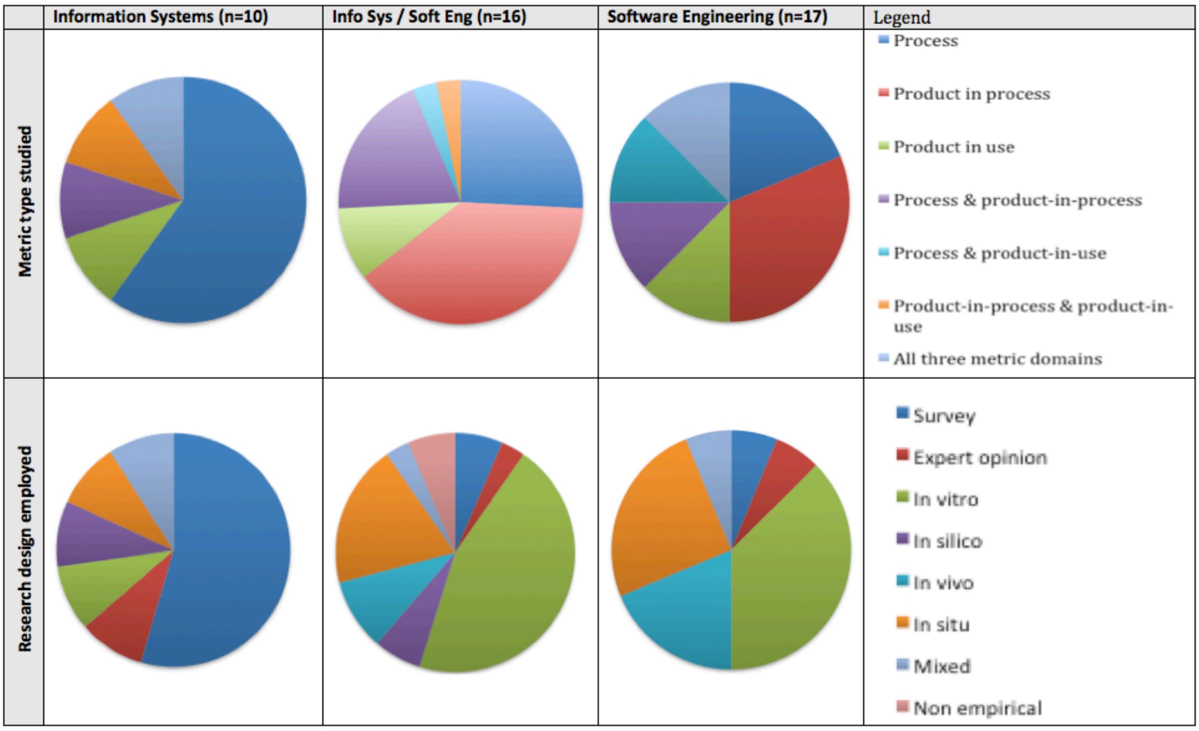


Figure 2a: Comparing the metrics studied and methods used in IS and SE

	In vitro	In silico	In vivo	In situ	Expert	Survey	Mixed	Non empirical	Total
Process	2	2	3	3	1	3	0	2	**16**
Product in process	5	1	0	2	2	1	1	0	**12**
Product in use	**12**	0	0	0	0	0	1	0	**13**
Process & product-in-process	2	0	0	4	0	2	0	0	**8**
Process & product-in-use	0	0	2	0	0	2	0	0	**4**
Product-in-process & product-in-use	0	0	0	1	0	0	0	0	**1**
All three metric domains	0	0	1	1	0	1	0	0	**3**
Total:	**21**	**3**	**6**	**11**	**3**	**9**	**2**	**2**	**57**

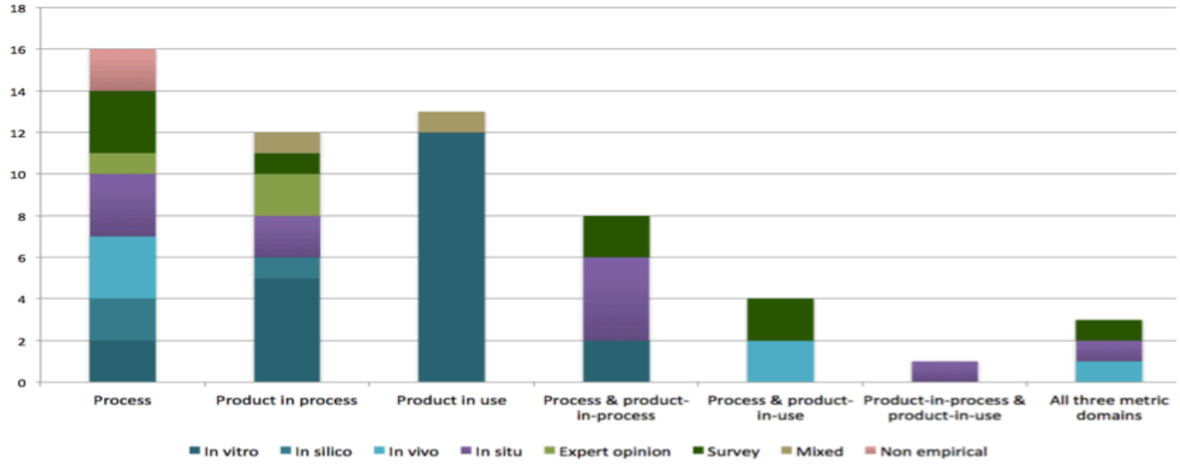


Figure 2b: Analysing the methods used to study software metrics in the two fields

In terms of the metrics, 40 studies focus on a single type of metric with the remainder focusing on two or more metric types. A closer inspection of Figure 2(a) reveals that the prevalence of studies investigating a single type of metric is similar in both fields. But there is a stronger emphasis on process metrics in IS than in SE, which is more balanced in its coverage of the different metric types. 17 studies focus on multiple metric types and they are evenly distributed across the fields. Over half examine process metrics together with product-in-process metrics but only three examine all three metric types and only one combines product-in-process and product-in-use metrics. A wide variety of research designs has been used but 21 of the 57 studies use an *in vitro* approach. Survey-based studies (n=9) and *in situ* studies (n=11) are also common, though mixed methods are rare (n=2). A closer inspection of Figure 2(a) reveals that there are more *in vitro* studies in the IS/SE domain (n=14) than in IS (n=1) or in SE (n=5) while Figure 2(b) reveals that there is less diversity in the methods used to study product-in-use metrics or to study different combinations of metric types compared to the methods used to study process and product-in-process metrics. In particular, 12 of the 13 papers studying product-in-use metrics rely on in vitro methods. Due to space constraints, we are unable to present a more in-depth analysis of the studies, but would like to draw the reader's attention to some key points that help explain the misalignment of literature with practitioner needs and suggest particular recommendations for IS research on metrics going forward.

4.1 The literature is heavily focused on the identification and validation of metrics

Many of the papers in both IS and SE are primarily designed to identify and validate new metrics ([18] and [55] are typical of papers in this style). In addition, the majority of the papers in our sample focused on metrics that would evaluate the functionality of software rather than the usability of software ([23] is an interesting exception). This is unfortunate because significant preventable losses are incurred every year when companies build highly functional and technically successful systems that are nevertheless unused or underused because of poor usability (Markus and Keil, 2004). In addition, a significant portion of the studies focus on a specific metric with only a handful of papers comparing individual metrics (e.g. [13]) to one another or comparing suites of metrics to one another (e.g. [35]). As a result, there is little or no integration across studies and the literature as a whole lacks coherence. This means that there are important gaps in the literature that go undetected - our analysis did not find any papers that proposed metrics for program level management, for example. But it is especially problematic for practitioners who must integrate assessments of (sets of) process and product metrics into the software development process and struggle to clearly define measures that can be derived directly from evolving engineering artefacts and fully integrated into all activities and teams (Royce, 2002).

4.2 The literature is heavily focused on particular types of research question

Many of the papers have a strong technical focus (our analysis found that 38 papers had an exclusively technical focus), frequently overlooking managerial and human (individual or social) measures of performance. This is problematic because of the complex relationship that exists between social interaction and software development success [44] and because "human aspects are the source of the majority of problems associated with software development projects" [19]. As a result, many of the papers fail to provide an adequate description of the context in which their studies are carried out. This is problematic given the diversity of the domain and the broad range of contextual factors such as application domain and application criticality, research methodology employed, programming language used, age and physical distribution of developers that affect the software development product and process. One exception is [12] which takes into consideration the fact that different components of the software being measured are to be used in mixed critical contexts. Thus, many of the papers fail to provide adequate insights to practitioners on when, where and how to use a particular metric or suite of metrics. In particular, the literature provides very little guidance on maximizing the utility of metrics in practice and sheds very little light on the impact of measurement programs on organisational behaviour.

4.3 The literature is heavily focused on particular types of research design

Finally, our analysis suggests that many of the shortcomings listed above have arisen because a significant proportion of the studies rely on data that has been extracted from industrial settings but is analysed in artificial settings. Whilst this approach is well suited to the identification and validation of software metrics, it is of little use in terms of (i) identifying principles to guide practitioners on the selection, utilization and evaluation of metrics and measurement programs in practice and in term of (ii) developing a better understanding the organisational impact of measurement programs and practice. At the same time, a number of studies carried out either *in situ* or in collaboration with practitioners have produced some significant findings. Several studies, for example, report on practitioners' efforts to streamline their measurement programs and to reduce the number of metrics used without reducing measurement program performance (cf. Concas *et al.*, 2008; Staron, 2012; Kehan *et al.*, 2011). In one well known study that was carried out in Microsoft, eight organisational metrics were found to be more effective predictors of fault proneness than traditional metrics (cf. Nagapan *et al.*, 2008).

5 Conclusion

This paper highlights the positive impact that metric and measurement programs have had on ISD project performance outcomes and the potential of ISD metrics research to contribute to the reduction of costly ISD project performance issues in practice. The paper also suggests that ISD metrics may have a role to play in evaluating the impact of emerging IS development methodologies (e.g. agile and model driven development) on ISD project performance, particularly in comparison with more traditional approaches. Nevertheless, our initial analysis reveals the existence of a significant gap between the contributions that are made by the existing ISD metrics literature and the needs of practitioners. The paper therefore presents an interdisciplinary review of 57 peer reviewed studies on ISD metrics that have been published in leading Information Systems (IS) and Software Engineering (SE) journals between 2001 and 2013.

One of the main features of the paper is that it both maps and reviews the literature. The mapping of the literature is done so that a clear picture can emerge of the topography of the field and the gaps within it. This analysis is then complemented by the literature review which done so that the underlying reasons for these gaps can also emerge. Though the study has its limitations, the composition of the sample size and the exclusive reliance on journal publications are perhaps its most significant limitations, a number of important observations arise from the analysis that significantly enhance the capacity of IS research to impact upon ISD performance improvement practice. The citation analysis confirms that there is significant interest in existing literature on ISD metrics. This underscores the practical and theoretical significance of research in this area and should encourage both editors and researchers to publish in the domain. The analysis also shows that in many instances, researchers fail to provide adequate information on software development methods that had been used in their studies. This is at least partly due to the reliance on research designs that are based on artificial data or are carried out in artificial settings but if the research is to be used to evaluate the impact of IS development methodologies on ISD project performance then researchers should be encouraged to more carefully distinguish between traditional, plan driven, ISD settings and other settings.

The main contribution of the paper, however, is in confirming that the majority of ISD metrics studies in the sample are designed to contribute a narrow technical validation of specific ISD metrics rather than to inform the selection, utilization or evaluation of them. It also identifies some of the reasons why this is the case. First, the primary research focus of the studies is an issue with the majority of the studies in the sample having a purely technical focus and with only a tiny minority of published papers examining purely non-technical (i.e. individual, social, behavioural and organisational) issues. Second, the number and configuration of metrics studied is an issue. More specifically, a significant proportion of the studies focus on a single metric or on a single type of metric. The studies that focus on two types of metric typically focus on particular combinations of metric type (e.g. they investigate process metrics in conjunction with product-in-process metrics) and very few studies examine all three types of metric at once. This helps to explain the lack of integration across studies and the lack of coherence in the research as a whole. Third, the analysis reveals that a significant proportion of the literature is based on empirical studies that rely on artificial data and/or artificial settings and are illsuited to the investigation of the efficacy of ISD processes, the usability of ISD products and to quantify the impact of ISD products on the generation of business value.

Acknowledgements

This work was supported, in part, by Science Foundation Ireland grant 10/CE/I1855 to Lero - the Irish Software Engineering Research Centre (www.lero.ie).

References

Abrahamsson, P., Conboy, K., and Wang, X. 2009. "*"Lots Done, More to Do": The Current State of Agile Systems Development Research,*").

Barry, C., and Brown, J. 2003. "Finding a Home for Web-Based Information Systems-Perusing the Landscape," *ECIS*, pp. 126-138.

Basili, V.R. 1992. "*Software Modeling and Measurement: The Goal/Question/Metric Paradigm*".

Bharadwaj, A., Keil, M., and Mähring, M. 2009. "Effects of Information Technology Failures on the Market Value of Firms," *The Journal of Strategic Information Systems* (18:2), pp. 66-79.

Brereton, P., Kitchenham, B.A., Budgen, D., Turner, M., and Khalil, M. 2007. "Lessons from Applying the Systematic Literature Review Process within the Software Engineering Domain," *Journal of systems and software* (80:4), pp. 571-583.

Conboy, K. 2009. "Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development," *Information Systems Research* (20:3), pp. 329-354.

Conboy, K and Fitzgerald, B. 2010. "Method and Developer Characteristics for Effective Agile Method Tailoring: A Study of XP Expert Opinion," *ACM Transactions on Software Engineering and Methodology* (20:1)

Concas, G., Di Francesco, M., Marchesi, M., Quaresima, R., and Pinna, S. 2008. "Study of the Evolution of an Agile Project Featuring a Web Application Using Software Metrics," in *Product-Focused Software Process Improvement*. Springer, pp. 386-399.

Downey, S., and Sutherland, J. 2013. "Scrum Metrics for Hyperproductive Teams: How They Fly Like Fighter Aircraft," *46th Hawaii International Conference on System Sciences (HICSS)*, pp. 4870-4878.

Dybå, T., and Dingsøyr, T. 2008. "Empirical Studies of Agile Software Development: A Systematic Review," *Information and software technology* (50:9), pp. 833-859.

Dyba, T., Dingsoyr, T., and Hanssen, G.K. 2007. "Applying Systematic Reviews to Diverse Study Types: An Experience Report," *First International Symposium on Empirical Software Engineering and Measurement*: IEEE, pp. 225-234.

Fenton, N. 1994. "Software Measurement: A Necessary Scientific Basis," *IEEE Transactions on Software Engineering* (20:3), pp. 199-206.

Fink, A. 1998. "Conducting Research Literature Reviews: From Paper to the Intern,").

Harjumaa, L., Markkula, J., and Oivo, M. 2008. "How Does a Measurement Programme Evolve in Software Organizations?," in *Product-Focused Software Process Improvement*. Springer, pp. 230-243.

Harrison, W. 2004. "A Flexible Method for Maintaining Software Metrics Data: A Universal Metrics Repository," *Journal of Systems and Software* (72:2), pp. 225-234.

Hartmann, D., and Dymond, R. 2006. "Appropriate Agile Measurement: Using Metrics and Diagnostics to Deliver Business Value," *Agile Conference, 2006*: IEEE, pp. 6 pp.-134.

Jalali, S., and Wohlin, C. 2012. "Global Software Engineering and Agile Practices: A Systematic Review," *Journal of Software: Evolution and Process* (24:6), pp. 643-659.

Kitchenham, B.A., and Charters, S. 2007. "Guidelines for Performing Systematic Literature Reviews in Software Engineering".

Kulas, H. 2012. "Product Metrics in Agile Software Development," in: *School of Information Sciences*. University of Tampere.

Kunz, M., Dumke, R.R., and Zenker, N. 2008. "Software Metrics for Agile Software Development," *19th Australian Conference on Software Engineering*: IEEE, pp. 673-678.

Markus, M.L., and Keil, M. 1994. "If We Build It, They Will Come: Designing Information Systems That People Want to Use," *Sloan Management Review* (35), pp. 11-11.

Meneely, A., Smith, B., and Williams, L. 2012. "Validating Software Metrics: A Spectrum of Philosophies," *ACM Transactions on Software Engineering and Methodology (TOSEM)* (21:4), p. 24.

Nagappan, N., Ball, T., and Zeller, A. 2006. "Mining Metrics to Predict Component Failures," *Proceedings of the 28th international conference on Software engineering*: ACM, pp. 452-461.

Nerur, S., and Balijepally, V. 2007. "Theoretical Reflections on Agile Development Methodologies," *Communications of the ACM* (50:3), pp. 79-83.

Petersen, K. 2011. "Measuring and Predicting Software Productivity: A Systematic Map and Review," Information and Software Technology (53:4), pp. 317-343.

Petersen, K., and Wohlin, C. 2010. "The Effect of Moving from a Plan-Driven to an Incremental Software Development Approach with Agile Practices," *Empirical Software Engineering* (15:6), pp. 654-693.

Royce, W. 2002. "Cmm Vs. Cmmi: From Conventional to Modern Software Management," *The Rational Edge*, pp. 2-9.

Sessions, R. 2009. "The IT Complexity Crisis: Danger and Opportunity," White paper, November).

Staron, M. 2012. "Critical Role of Measures in Decision Processes: Managerial and Technical Measures in the Context of Large Software Development Organizations," *Information and Software Technology* (54:8), Aug, pp. 887-899.

Wang, X., Conboy, K., and Cawley, O. 2012. "“Leagile” Software Development: An Experience Report Analysis of the Application of Lean Approaches in Agile Software Development," *Journal of Systems and Software* (85:6), pp. 1287-1299.

Appendix A

The 57 papers that were analysed in this study are listed below. These papers are referenced in the paper using square brackets whereas all other sources are referenced in the normal fashion.

List of publications analysed as part of the literature review (n=57)	
[1]	Alshayeb, M., and Li, W. 2003. "An Empirical Validation of Object-Oriented Metrics in Two Different Iterative Software Processes," IEEE Transactions on Software Engineering (29:11), Nov, pp. 1043-1049.
[2]	Alshayeb, M., and Li, W. 2005. "An Empirical Study of System Design Instability Metric and Design Evolution in an Agile Software Process," Journal of Systems and Software (74:3), Feb 1, pp. 269-274.
[3]	Andersson, Carina, and Per Runeson. "A spiral process model for case studies on software quality monitoring—method and metrics." Software Process: Improvement and Practice 12.2 (2007): 125-140.
[4]	Bavota, G., De Lucia, A., and Oliveto, R. 2011"Identifying Extract Class Refactoring Opportunities Using Structural and Semantic Cohesion Measures," Journal of Systems and Software (84:3), Mar, pp. 397-414.
[5]	Bertoa, M.F., Troya, J.M., and Vallecillo, A. 2006. "Measuring the Usability of Software Components," Journal of Systems and Software (79:3), Mar, pp. 427-439.
[6]	Bibi, Stamatia, et al. "BBN based approach for improving the software development process of an SME—a case study." Software Process: Improvement and Practice (2009).
[7]	Binkley, D., Feild, H., Lawrie, D., and Pighin, M. 2009. "Increasing Diversity: Natural Language Measures for Software Fault Prediction," Journal of Systems and Software (82:11), Nov, pp. 1793-1803.
[8]	Chen, C.B., Lin, C.T., Wang, C.H., and Chang, C.W. 2006. "Model for Measuring Quality of Software in Dvrs Using the Gap Concept and Fuzzy Schemes with Ga," Information and Software Technology (48:3), Mar, pp. 187-203.
[9]	Chilton, Michael A., Bill C. Hardgrave, and Deborah J. Armstrong. "Person-job cognitive style fit for software developers: the effect on strain and performance." Journal of Management Information Systems 22.2 (2005): 193-226.
[10]	Conejero, J.M., Figueiredo, E., Garcia, A., Hernandez, J., and Jurado, E. "On the Relationship of Concern Metrics and Requirements Maintainability," Information and Software Technology (54:2), Feb, pp. 212-238.
[11]	Cubranic, D., Murphy, G.C., Singer, J., and Booth, K.S. 2005. "Hipikat: A Project Memory for Software Development," IEEE Transactions on Software Engineering (31:6), Jun, pp. 446-465.
[12]	Durisic, D., Nilsson, M., Staron, M., and Hansson, J. 2013. "Measuring the Impact of Changes to the Complexity and Coupling Properties of Automotive Software Systems," Journal of Systems and Software (86:5), May, pp. 1275-1293.
[13]	Etzkorn, Letha H., et al. "A comparison of cohesion metrics for object-oriented systems." Information and Software Technology 46.10 (2004): 677-687.
[14]	Gencel, C., Petersen, K., Mughal, A.A., and Iqbal, M.I. "A Decision Support Framework for Metrics Selection in Goal-Based Measurement Programs: Gqm-Dsfms," Journal of Systems and Software (86:12), Dec, pp. 3091-3108.

[15] Gopal, Anandasivam, et al. "Measurement programs in software development: determinants of success." Software Engineering, IEEE Transactions on 28.9 (2002): 863-875.

[16] Gotel, Olly, et al. "Quality indicators on global software development projects: does ‘getting to know you’really matter?." Journal of Software: Evolution and Process 24.2 (2012): 169-184.

[17] Gou, Lang, et al. "Quantitative defects management in iterative development with BiDefect." Software Process: Improvement and Practice 14.4 (2009): 227-241.

[18] Han, A.-R., Jeon, S.-U., Bae, D.-H., and Hong, J.-E. "Measuring Behavioral Dependency for Improving Change-Proneness Prediction in Uml-Based Design Models," Journal of Systems and Software (83:2), Feb, pp. 222-234.

[19] Hazzan, O., and Hadar, I. 2008. "Why and How Can Human-Related Measures Support Software Development Processes?," Journal of Systems and Software (81:7), Jul, pp. 1248-1252.

[20] Henderson, Peter, Yvonne Margaret Howard, and Robert John Walters. "A tool for evaluation of the software development process." Journal of Systems and Software 59.3 (2001): 355-362.

[21] Iversen, J., and Mathiassen, L. 2003. "Cultivation and Engineering of a Software Metrics Program," Information Systems Journal (13:1), Jan, pp. 3-19.

[22] Johnson, P.M., Kou, H.B., Paulding, M., Zhang, Q., Kagawa, A., and Yamashita, T. 2005. "Improving Software Development Management through Software Project Telemetry," IEEE Software (22:4), Jul-Aug, pp. 76-+.

[23] Joshi, A., Sarda, N.L., and Tripathi, S. "Measuring Effectiveness of Hci Integration in Software Development Processes," Journal of Systems and Software (83:11), Nov, pp. 2045-2058.

[24] Kakarontzas, G., Constantinou, E., Ampatzoglou, A., and Stamelos, I. "Layer Assessment of Object-Oriented Software: A Metric Facilitating White-Box Reuse," Journal of Systems and Software (86:2), Feb, pp. 349-366.

[25] Kumar, K.V., Ravi, V., Carr, M., and Kiran, N.R. 2008. "Software Development Cost Estimation Using Wavelet Neural Networks," Journal of Systems and Software (81:11), Nov, pp. 1853-1867.

[26] Lee, G., and Xia, W. "Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data on Software Development Agility," Mis Quarterly (34:1), Mar, pp. 87-114.

[27] Lee, G., and Xia, W.D. 2005. "The Ability of Information Systems Development Project Teams to Respond to Business and Technology Changes: A Study of Flexibility Measures," European Journal of Information Systems (14:1), Mar, pp. 75-92.

[28] Li, M., and Smidts, C.S. 2003. "A Ranking of Software Engineering Measures Based on Expert Opinion," IEEE Transactions on Software Engineering (29:9), Sep, pp. 811-824.

[29] Lin, C.-T., and Huang, C.-Y. 2008. "Enhancing and Measuring the Predictive Capabilities of Testing-Effort Dependent Software Reliability Models," Journal of Systems and Software (81:6), Jun, pp. 1025-1038.

[30] Madeyski, L. "The Impact of Test-First Programming on Branch Coverage and Mutation Score Indicator of Unit Tests: An Experiment," Information and Software Technology (52:2), Feb, pp. 169-184.

[31]	Manadhata, P.K., and Wing, J.M. "An Attack Surface Metric," IEEE Transactions on Software Engineering (37:3), May-Jun, pp. 371-386.
[32]	Mendes, E., Mosley, N., and Counsell, S. 2005. "Investigating Web Size Metrics for Early Web Cost Estimation," Journal of Systems and Software (77:2), Aug, pp. 157-172.
[33]	Miranda, E., and Bourque, P. "Agile Monitoring Using the Line of Balance," Journal of Systems and Software (83:7), Jul, pp. 1205-1215.
[34]	Na, K.-S., Simpson, J.T., Li, X., Singh, T., and Kim, K.-Y. 2007. "Software Development Risk and Project Performance Measurement: Evidence in Korea," Journal of Systems and Software (80:4), Apr, pp. 596-605.
[35]	Olague, H.M., Etzkorn, L.H., Gholston, S., and Quattlebaum, S. 2007. "Empirical Validation of Three Software Metrics Suites to Predict Fault-Proneness of Object-Oriented Classes Developed Using Highly Iterative or Agile Software Development Processes," IEEE Transactions on Software Engineering (33:6), Jun, pp. 402-419.
[36]	Palviainen, M., Evesti, A., and Ovaska, E. "The Reliability Estimation, Prediction and Measuring of Component-Based Software," Journal of Systems and Software (84:6), Jun, pp. 1054-1070.
[37]	Petersen, Kai. "Measuring and predicting software productivity: A systematic map and review." Information and Software Technology 53.4 (2011): 317-343.
[38]	Port, D., and Bui, T. 2009. "Simulating Mixed Agile and Plan-Based Requirements Prioritization Strategies: Proof-of-Concept and Practical Implications," European Journal of Information Systems (18:4), Aug, pp. 317-331.
[39]	Procaccino, J.D., Verner, J.M., Darter, M.E., and Amadio, W.J. 2005. "Toward Predicting Software Development Success from the Perspective of Practitioners: An Exploratory Bayesian Model," Journal of Information Technology (20:3), Sep, pp. 187-200.
[40]	Quah, Tong-Seng, and Mie Mie Thet Thwin. "Prediction of software development faults in PL/SQL files using neural network models." Information and Software Technology 46.8 (2004): 519-523.
[41]	Raffo, D.M. 2005. "Software Project Management Using Prompt: A Hybrid Metrics, Modeling and Utility Framework," Information and Software Technology (47:15), Dec, pp. 1009-1017.
[42]	Sarkar, S., Kak, A.C., and Rama, G.M. 2008. "Metrics for Measuring the Quality of Modularization of Large-Scale Object-Oriented Software," IEEE Transactions on Software Engineering (34:5), Sep-Oct, pp. 700-720.
[43]	Sarker, S., Munson, C.L., Sarker, S., and Chakraborty, S. 2009. "Assessing the Relative Contribution of the Facets of Agility to Distributed Systems Development Success: An Analytic Hierarchy Process Approach," European Journal of Information Systems (18:4), Aug, pp. 285-299.
[44]	Sawyer, S., Guinan, P.J., and Cooprider, J. "Social Interactions of Information Systems Development Teams: A Performance Perspective," Information Systems Journal (20:1), Jan, pp. 81-107.
[45]	Sobel, Ann E. Kelley, and Michael R. Clarkson. "Formal methods application: an empirical tale of software development." Software Engineering, IEEE Transactions on 28.3 (2002): 308-320.
[46]	Stamelos, Ioannis, et al. "Code quality analysis in open source software development." Information Systems Journal 12.1 (2002): 43-60.

[47]	Staron, M., Meding, W., and Soderqvist, B. 2010. "A Method for Forecasting Defect Backlog in Large Streamline Software Development Projects and Its Industrial Evaluation," Information and Software Technology (52:10), Oct, pp. 1069-1079.
[48]	Subramanyam, R., and Krishnan, M.S. 2003. "Empirical Analysis of Ck Metrics for Object-Oriented Design Complexity: Implications for Software Defects," IEEE Transactions on Software Engineering (29:4), Apr, pp. 297-310.
[49]	Succi, G., Pedrycz, W., Stefanovic, M., and Miller, J. 2003. "Practical Assessment of the Models for Identification of Defect-Prone Classes in Object-Oriented Commercial Systems Using Design Metrics," Journal of Systems and Software (65:1), Jan 15, pp. 1-12.
[50]	Teasley, Stephanie D., et al. "Rapid software development through team collocation." Software Engineering, IEEE Transactions on 28.7 (2002): 671-683.
[51]	Tomaszewski, P., and Lundberg, L. 2005. "Software Development Productivity on a New Platform: An Industrial Case Study," Information and Software Technology (47:4), Mar 15, pp. 257-269.
[52]	Tunkelo, Teemu, Ari-Pekka Hameri, and Yves Pigneur. "Improving globally distributed software development and support processes–A workflow view." Journal of Software: Evolution and Process 25.12 (2013): 1305-1324.
[53]	van Koten, C., and Gray, A.R. 2006. "Bayesian Statistical Effort Prediction Models for Data-Centred 4gl Software Development," Information and Software Technology (48:11), Nov, pp. 1056-1067.
[54]	Van Rompaey, B., Du Bois, B., Demeyer, S., and Rieger, M. 2007. "On the Detection of Test Smells: A Metrics-Based Approach for General Fixture and Eager Test," IEEE Transactions on Software Engineering (33:12), Dec, pp. 800-817.
[55]	Wijayasiriwardhane, T., and Lai, R. "Component Point: A System-Level Size Measure for Component-Based Software Systems," Journal of Systems and Software (83:12), Dec, pp. 2456-2470.
[56]	Wood, S., Michaelides, G., and Thomson, C. "Successful Extreme Programming: Fidelity to the Methodology or Good Teamworking?," Information and Software Technology (55:4), Apr, pp. 660-672.
[57]	Xia, W.D., and Lee, G.H. 2005. "Complexity of Information Systems Development Projects: Conceptualization and Measurement Development," Journal of Management Information Systems (22:1), Sum, pp. 45-83.

An earlier version of this paper was presented at the Australasian Conference on Information Systems (ACIS) 2014 in Auckland, New Zealand.