Fitzgerald, B. (1998) An Empirical Investigation into the Adoption of Systems Development Methodologies, *Information & Management*, Vol. 34, pp. 317-328.

AN EMPIRICAL INVESTIGATION INTO THE ADOPTION OF SYSTEMS DEVELOPMENT METHODOLOGIES

Brian Fitzgerald
Executive Systems Research Centre
University College Cork,
Cork, Ireland
Email: bf@ucc.ie

Abstract

A postal survey was conducted to determine: the extent to which systems development methodologies are used in practice; the rating of their contribution to the development process, and the future trend in methodology adoption, whether the trend is towards increased usage or not. Among the main findings of the study were that 60 percent of the respondents were not using methodologies. Also, only 6 percent of respondents reported following a methodology rigorously. Finally, the forecast is not positive for methodology advocates: 79 percent of those not using a methodology indicated that they did not intend to adopt one.

Key Words: Information systems development, methods and tools, methodologies

1. Introduction

Systems development is one of the core issues in the IS field [14]. In light of this, it is not surprising that systems development methodologies (SDMs) should be the subject of extensive research. The literature has traditionally viewed them as axiomatically appropriate to improving both the process and product of systems development [20], with their use being typically seen as valuable. While there are a number of significant arguments in their favour, there are also a number of arguments and pressures that question the use of methodologies.

2. Issues in the Use of Methodologies

2.1 Arguments and Pressures in Favour

Early systems development projects often used unsystematic methods. However, as the complexity of the systems increased, more disciplined methodological approaches were advocated [23]. Some of the fundamental concepts underpinning the use of methodologies are:

- Systems development is a very complex process. Methodologies may provide a reductionist subdivision of this process into plausible and coherent steps [41].

- By rendering the development task more visible and transparent, methodologies may facilitate project management and control of the development process, thus reducing risk and uncertainty [e.g., 2, 21].
- They may provide a purposeful framework for the application of techniques and resources at appropriate times during the development process [e.g., 1, 5].
- There is an economic rationale: methodologies may allow skill specialisation and division of labour (e.g., analysis, design, coding, and testing) which can receive different remuneration rates.
- An epistemological rationale can be identified as methodologies may provide a structural framework for the acquisition of knowledge. Thus, any learning from past development experiences can be systematised and stored for future reference [6, 45, 46].
- Standardisation of the development process is possible. This facilitates interchangeability among developers. Also, it can lead to increased productivity and quality, as resource requirements can be predicted and made available as and when necessary [17].

In addition, there are a number of significant pressures, complementing these factors, which support the increased use of methodologies. They include:

- Desirability of ISO-certification in many organizations is fuelling the adoption of methodologies as a means of achieving certification [19].
- Some governments, who are significantly involved in systems development, have required that certain methodologies be used for development. For example, SSADM (UK, Ireland, Malta, Hong Kong, Israel); Dafne (Italy); Merise (France); and NIAM (Netherlands). This creates a significant pressure in practice for their adoption [25].

2.2 Arguments and Pressures Against the Use of Methodologies

It is evident that there is a consensus among many that the use of methodologies is positive and well-advised [29], and the implicit expectation is that their usage should increase over time. However, practitioners have been somewhat slow in adopting SDMs. This has been explained variously as, for example, due to the "wealth of ignorance" among developers [52], or due to the slow speed of technology transfer [12]. SDMs are attractive and have an intuitive appeal, but a methodology is merely a framework for organising the process. Some criticisms are:

- Estimates suggest that more than a thousand brand-named methodologies exist. However, there are artificial contrived differences between some, the many variations of the Structured Approach [13], for example, whereas there are fundamental differences between others; for example, soft methodologies, such those from the Scandinavian tradition, differ substantially from hard methodologies such as the Structured Approach [35].
- Generalisation has been made without adequate conceptual and empirical foundation as many methodologies have been proffered on the basis of limited use, without the twin customary safeguards of (i) expert independent review to determine their appropriateness; and (ii) their successful application on a non-trivial real-world case.

- Systems development is not actually an orderly rational process but most SDMs treat it as such, with a major emphasis on technical rationality at the expense of social aspects [54].
- There may be some means-ends inversion, as developers proceed in slavish and blind adherence to methodologies, while losing sight of the fact that development of an actual system is the real objective [19].
- Often there is an assumption that methodologies are universally applicable across all development situations—the one size fits all presumption. However, researchers have pointed out that due consideration needs to be given to the contingencies of each development situation [3, 16].
- There is an inadequate recognition of developer-embodied factors, as SDMs do not address factors critical to successful development, such as individual creativity and intuition, or learning over time [8, 9, 50].

Just as there are a number of pressures that support increased formalism, there are also a number of significant pressures for radical change in the traditional approach to development.

- At a broad level, the changing nature of the business environment means that short-term needs now dominate. Thus, traditional lifecycle approaches that result in the eventual delivery of systems after several years are no longer appropriate [10].
- There is also some evidence of an altered profile of systems development environment. Traditional in-house development of systems may no longer be the norm. Much development now consists of what could be labelled configuration development, i.e., the integration and customisation of package software to incorporate local practices [4].
- Also, there are moves towards Rapid Application Development (RAD), with the replacement of large-scale monolithic approaches by approaches based on the 'frequent tangible results' philosophy of delivering some functioning components of the system in shorter timescales [22, 26].

These arguments bring about a dilemma for systems developers: is it wise to adopt a formalised development methodology or not. This research thus focused on the behaviour and perceptions of developers in relation to methodology usage.

Previous studies have reported usage rates for methodologies of 87 percent [29] and 62 percent [40]. However, previous studies have generally taken methodology usage to be a binary factor, i.e., either organizations are using methodologies or not. Nor have previous studies typically distinguished between internal methodologies and commercial third-party methodologies. Thus, we adopted a more comprehensive categorisation of methodology usage. The study also sought to investigate the consistency and rigour with which methodologies are being applied in practice. The specific aims of the study were:

- Establish the percentage of methodology usage in practice, focusing on both commercial and internally-derived methodologies.
- Establish whether the trend is towards increased methodology usage.
- Identify the circumstances in which developers use methodologies, and the extent to which they are followed rigorously.

- Establish the profile of the systems development environment currently faced (in-house development, outsourcing, or customisation of packages).
- Determine the rating of the contribution of methodologies to the development process.

3. Research Method

A questionnaire[1] was constructed based on an extensive literature review, the researcher's commercial experience (over 12 years) as a systems developer, and a pilot study involving interviews with experienced developers in five organizations. This questionnaire was pre-tested over several months prior to the survey, and mailed to a total of 776 named individuals in different organizations. The population were known to be involved in systems development. This sample was drawn from three separate groups:

- Group 1: 107 organizations with whom the researcher's university department had some prior research contact.
- Group 2: a database of software and technology organizations (when duplicates were removed, this group comprised 331 organizations)
- Group 3: a database of the top 1000 organizations by turnover in the country, with two selection criteria applied: Firstly, duplicates were eliminated. Secondly, individuals responsible for the computing function were chosen if their job title indicated that they occupied a traditional IS department role. Thus, IS managers and systems analysts were included, whereas a financial controller would be eliminated. This selection strategy retained 338 organizations.

Thus, the overall sample size was 776 organizations.

It is important to note that this sample is a purposive, non-probabilistic one. Researchers have criticised the lack of use of random sampling strategies in IS research [31]. However, Mason [37] has pointed out that in IS research, access is often one of the problematic issues. He suggests that rather than criticising the lack of true random sampling, researchers should strive to construct samples that allow the most powerful inferences to be made. Therefore, one of the principles guiding the sample selection was that of ensuring that participants would be likely to be significantly and directly involved in the phenomenon of interest. This strategy is in keeping with that recommended by Eisenhardt [18] which recommends that samples be chosen for theoretical reasons so that the phenomenon of interest are more likely to be present.

A final total of 162 usable responses were received, giving an overall response rate of 21 percent. This level of response is in keeping with a similar UK survey [24] where response rates of 20 and 29 percent were achieved over two surveys on methodology usage. Response rates from other similar postal surveys in the US include 18 percent [36]; 22 percent [43]; 19 percent [44]; and 27 percent [47].

[1] A copy of the questionnaire is available on request.

3.1 Analysis of Survey Responses

Given that the survey questionnaire involved a good deal of nominal or categorical scale data, non-parametric methods for testing statistical significance are the most appropriate. The issue of the use of parametric or non-parametric tests is one of contention in the social sciences. For example, it has been suggested by some researchers that ordinal scales can be treated as interval, as the richer set of parametric statistical procedures which these allow more than compensate for any methodological error [32]. However, given the sample size and the failure to meet the conditions of normality and variance here, a more conservative approach was taken, and non-parametric tests were applied. However, even where the data involved interval or ratio scales, there are certain conditions with respect to normality of distribution and homogeneity of variance which need to be satisfied before parametric tests are appropriate. Given that these tests were being carried out on sub-groups of a relatively small sample, and the fact that the requisite conditions with respect to normality and variance were not satisfied for most factors, non-parametric tests such as Chi-square contingency analysis and the Mann-Whitney U test were used [7].

3.2 Investigation of Sample Validity and Non-Response Bias

The issue of representativeness and validity of the sample was highly pertinent. The following four-fold strategy, which employed both quantitative and qualitative techniques [42, 51], was adopted to investigate sample validity and possible non-response bias:

- use of late respondents as surrogates for non-respondents, and comparison of these late responses with a random sample of 'normal' responses using Chi-square contingency analysis.
- comparison of response sample characteristics with the characteristics of other similar sample populations.
- telephone follow-up with a random selection of non-respondents to ascertain reasons for non-response.
- comparison of the number of respondents who cited use of particular methodologies against vendor estimates of total numbers using these methodologies.

The weight of evidence from all four strands of this pluralistic strategy strongly supported the hypothesis that the sample was valid and that response bias was not a factor here.

4. Background and Demographic Information

Table 1 provides background information on the respondents' organizations. The sectoral distribution by business category for respondents was compared with the overall population sample and was found to be very similar in distribution. As can be seen, the largest organizational cohort in terms of numbers employed was the 100 to 1000 category, while the largest cohort in terms of IS department size was 1 to 5. Although these categories have not been cross-correlated, it would appear that relatively small IS departments often serve quite large organizations. Certainly, the numbers in IS departments are significantly lower than those reported in an earlier US

study by Sumner *et al*. However, Friedman reports a similar trend in decrease in IS department size in the 1980s. He suggests that this is due to the decentralising of developers into user departments. Also, the trend towards outsourcing may be causing a shrink in IS department size. This trend towards a reduction in IS department size continues to be reported [49].

Insert Table 1 about here

4.1 Development Profile

Table 2 provides information on the development profile in respondent organizations. Respondents were very experienced, with an average usage of more than 15 years. An interesting feature is the high-level of package customisation, and also the level of outsourcing. Given these figures, it would appear that in-house development is no longer predominant. In addition, typical development projects required about three developers for less than six months, which suggests a profile of small-scale, rapid development efforts, compared to the findings of an earlier study by Jenkins *et al*. Further evidence may perhaps be gained from the fact that the most common platform for development is the PC, again not one typically associated with large-scale development projects in the past. Thus, the development environment currently faced is radically different from that which prevailed when most current methodologies were first formulated, and these may not be ideally suited to present environment.

Insert Table 2 about here

5. Analysis of Methodology Usage

Table 3 presents the findings on the use of methodologies. The study drew upon a four-fold categorization of methodology usage:

- those using a third-party or commercial formalised systems development methodology (FSDM).
- those using an internal formalised methodology based on a commercial systems development methodology.
- those using an internal formalised methodology **not** based on a commercial systems development methodology.
- those not using any formalised systems development methodology.

This categorisation is preferable to a 'yes' or 'no' classification in terms of methodology use, as it closer represents a continuum.

Quite few third-party methodologies were actually represented in this study, with SSADM being the most popular, followed by Information Engineering, Oracle*Case, and Andersen Consulting's Foundation/Method 1. The SSADM methodology was also

the most popular method in the '*internal based on a commercial FSDM'* category, followed again by Information Engineering. In the latter category, methodologies such as Checkland's Soft Systems Method (SSM) [11] and Jackson Systems Design [27] were also mentioned as the basis for internal methodologies.

Insert Table 3 about here

5.1 Analysis of Methodology Usage

The analysis of methodology usage across a number of dimensions is presented in Table 4. The table illustrates that methodology usage is significantly more likely in larger organizations (more than 1000 employees), and in larger IS departments (more than 20 personnel). This finding is in accord with research by Zmud [55] which suggests that large organizations are more likely to introduce technical software innovations and modern software practices. The findings are also broadly in keeping with those of a 1985 NCC survey in the UK (reported by Friedman).

Not surprisingly, methodology usage is very significantly associated with both high levels of in-house development and low levels of customisation of packages and development outsourced. This is to be expected, however, as current methodologies were not originally designed for package customisation or outsourcing. Additionally, methodologies are significantly more likely to be used when more than 5 developers are involved on the project, and when the project duration is greater than 9 months. This is in keeping with the suggested project management role of methodologies, and would suggest that they play a role in controlling long projects and facilitating intercommunication among developers.

The table also shows that the relationship between methodology usage and business category is significant. Firstly, the *construction/manufacturing/distribution* category is one in which the use of methodologies is very low, while the *finance/insurance/real estate* category appears to be one in which the use of methodologies is more frequent. Some support for this finding comes from the 1985 NCC study which found that methodologies were primarily used in financial institutions. Use of methodologies, as one could perhaps reasonably expect, is predominantly associated with high levels of in-house development and low levels of outsourcing and package usage. When the profile of development environment for these categories is examined, it appears to confirm the fact that there is a low level of in-house development and higher levels of outsourcing and package customisation in the *construction/manufacturing/ distribution* category. Also, of interest is the fact that in the *finance/insurance/real estate* category, the average number of developers and average project duration are higher than in the other categories. The findings of this study extend the findings of earlier studies by identifying the threshold level on various dimensions above which methodology adoption is more likely.

Insert Table 4 about here

5.2 Development Activities and Technique Usage

Table 5(a) analyses the average percentage of time spent on various development activities, and the use of various tools and techniques by methodology users and non-users. Although the findings do not reveal significant differences, it appears that there is a slightly more even distribution of time when a methodology is being followed.

Thus, slightly more time appears to be allocated to the analysis and design activities. Several researchers have recommended that more time be allocated to early development phases [e.g. 15], with McKeen reporting a relationship between the amount of time allocated to the analysis phase and greater user satisfaction. However, the results of this study would suggest that methodologies *per se* do not ensure that this occurs, since there is so little difference in the proportion of time allocated to these activities by methodology users and non-users.

Table 5(b) analyses the percentage of respondents citing the use of various development tools and techniques. The most popular tools and techniques are prototyping, data flow diagramming, data dictionaries. and entity relationship models. The interesting finding here is that those using methodologies use **all** of these tools and techniques. This lends support to the argument that methodologies provide a suitable framework to co-ordinate the purposeful application of tools and techniques. It has been suggested that methodologies generally assume some underlying philosophy and fundamental principles in relation to the phases and activities of systems development [28]. However, given that there is no real differences in emphasis on particular development phases, the extent to which methodology users assimilate the deeper underpinning principles of methodologies is questionable.

Insert Tables 5(a) and 5(b) about here

Within the group of respondents using commercial formalised methodologies (23 respondents), some analysis was carried out on the two most popular commercial methodologies, SSADM (8 respondents) and Information Engineering (6 respondents). The results of this analysis are summarised in Table 6. In all but two cases, respondents stated that training in the use of the methodology had been provided. Given this, one would expect that there would be some uniformity in the manner in which these methodologies were applied. However, this is not supported. The minimum and maximum percentage of time allocated to different development activities in these organizations show wide discrepancies in the amount of time allocated.

Insert Table 6 about here

5.3 Adherence to Methodologies

The study also investigated the extent to which methodologies were actually followed rigorously. The overall finding was that 58 percent of those using them did not apply the methodology in a rigorous fashion. Respondents were also asked to indicate the reasons why this was so. Several reasons were given:

- *Project size/scale*: The general view was that methodologies were cumbersome and consume time and resources that were not always available. On small projects, the benefits did not seem to outweigh costs. One respondent, acknowledging the benefits, felt that productivity often had to be accorded higher priority.
- *Project type*: When the project was seen as simple, or when the system to be developed was stand-alone and not integrated with other systems, methodologies tended not to be followed rigorously.
- *Project client*: If the project was for certain clients, e.g. non-government, then methodologies were seldom followed.
- *Contingency*: Use depended on the needs of the situation. Developers needed to exercise their own judgement as to what was appropriate.

The various categories of methodology usage, from commercial to internal, were not analysed separately here as they did not exhibit significant differences. However, it is worth noting that only 10 of the 23 respondents who were using commercial methodologies stated they were following them rigorously. Thus, a mere 6 percent of respondents indicated that they were rigorously following a commercial methodology. This is important, since the implicit view is that it is poor practice to depart from the declarative sequence of steps prescribed by a methodology [48]. This finding is in keeping with recent studies by Russo *et al*., who report that almost 90 percent of those using a methodology modified it on a project-by-project basis, and Hardy *et al*. who found that the SSADM methodology was modified in 88 percent of cases. Also, Jenkins *et al.* reported that methodologies were modified in 81 percent of cases.

No clear pattern emerges in terms of business category. However, it appears that experienced developers (more than 10 years of experience) were less likely to follow a methodology rigorously, whereas developers with less experience were somewhat more likely to do so. This has been problematic in previous research. For example, Leonard-Barton's [34] study would suggest that experienced developers are more likely to use methodologies. However, Kozar's [30] study found that more experienced developers were less likely to. Lee and Kim [33] also reported that procedural formalisation is less important in older IS departments due to "the accumulation of systems development know-how" among developers. The findings of our study support these last two studies rather than the Leonard-Barton finding.

5.4 Ranking of Contribution of Methodologies

Respondents who were using methodologies were asked to indicate on a 5-point Likert-scale their level of agreement with various positive and negative factors. The negative factors that received most support were:

- methodologies can be cumbersome, and lead to inertia in the development process.
- following the methodology can interfere with actual development work.

The positive factor that received most support was:

- methodologies facilitate project control and increase visibility into the development process.

However, given the fact that development projects were typically of short duration and involving small teams, project management issues may be of less relevance and benefit.

Respondents were presented with a list of factors and asked to rank their contribution to successful systems development. Methodology users and non-users were presented with essentially the same list of factors, execept that methodology users were asked to rank the value of adherence to the methodology. The results are presented in Table 7.

There were no statistically significant differences among the various categories of methodology usage (commercial v. internal). However, the interesting feature of this table is that methodologies are ranked fourth in order of importance by methodology users, and also, both users and non users are in broad agreement on the ranking of these factors.

Insert Table 7 about here

5.5 Trend in Methodology Adoption

Table 8 indicates the different trend in methodology adoption. Of interest is the trend towards increased polarisation, that is, the majority of those who have adopted methodologies are moving towards increased usage, while the majority of those who are not using methodologies do not intend to adopt them. This seems to undermine the argument by Chikofsky that methodology usage will increase over time; rather, further polarisation is likely.

Insert Table 8 about here

6. Summary and Conclusions

The general message is that methodologies are not seen as a panacea for problems in systems development. Even those using methodologies rank them low in terms of their contribution to successful development. Their most evident contribution seems to be as a framework for the use of tools and techniques, as they do not appear *per se* to affect the allocation of time to various development activities. Additionally, methodology usage is not on the increase. However, ignorance does not seem to be an explanatory factor for this, as respondents were familiar with them. Rather, there were valid reasons for choosing not to use methodologies. Developers were aware of the limited contribution of methodologies.

Methodologies are neither applied rigorously nor uniformly, even when training in their use has been provided. This divergence in the development process in organisations purporting to use the same methodology lends support to the view that a

unique methodology-in-action is created for each development project. Methodologies are not applied in the same way by different developers. Indeed, they are probably not used in the same way by the same developer on different development projects. It would appear that developers do not really absorb any underpinning philosophy of the methodology, but adopt some of the prescribed techniques.

Many current methodologies are derived from practices and concepts relevant to the old organisational environment, and there is a need to reconsider their role in today's enviroment. Given the significant 'push' factor that this environment represents, there is an urgent need to leverage new developments, both technological and organisational, that enable new development approaches more appropriate to this climate. There is a need for more rapid systems delivery than that which is currently being achieved with the monolithic development approaches inherent in traditional system development methodologies. In fact, the latter may impose a considerable degree of inertia on the development process.

References

[1] N. Ahituv, M. Hadass and S. Neumann, (1984) A flexible approach to information system development. *MIS Quarterly*, **8**, 2, 69-78.

[2] D. Avison and G. Fitzgerald, (1995) *Information Systems Development: Methodologies, Techniques and Tools*. Blackwell Scientific Publications, Oxford.

[3] D. Avison, G. Fitzgerald and A. Wood-Harper, (1988) Information systems development: a tool kit is not enough. *The Computer Journal*, **31**, 4, 379-380.

[4] J. Bansler and E. Havn, (1994) Information systems development with generic systems, in Baets, W. (Ed), *Proceedings of Second European Conference on Information Systems*, Nijenrode University Press, Breukelen, 707-718.

[5] J. Bantleman and A. Jones, (1984) Systems analysis methodologies: a research project. In Bemelmans, T. (ed.) *Beyond Productivity: Information Systems Development for Organizational Effectiveness*, Elsevier Science Publishers B.V., North Holland Press, 213-227.

[6] R. Baskerville, J. Travis and D. Truex, (1992) Systems without method: the impact of new technologies on information systems development projects. In Kendall, K., DeGross, J. and Lyytinen, K. (eds.) *The Impact of Computer Supported Technologies on Information Systems Development*, Elsevier Science Publishers B.V., North Holland Press, 241-269.

[7] H. Blalock, (1972) *Social Statistics* (2nd Edition), McGraw-Hill, New York.

[8] B. Boehm, (1981) *Software Engineering Economics*. Prentice Hall, Englewood Cliffs, New Jersey.

[9] Brooks, (1987) No silver bullet: essence and accidents of software engineering. *IEEE Computer Magazine*, **April**, 10-19.

[10] P. Brown, (1985) Managing software development. *Datamation*, April 15, 133-136.

[11] P. Checkland, (1981) *Systems Thinking, Systems Practice*, Wiley, Chichester.

[12] E. Chikofsky, (1989) How to lose productivity with productivity tools. *Proceedings of 3rd IFAC/IFIP Workshop*, Indiana, US, 1-4.

[13] L. Constantine, (1989) The structured design approach. *Byte*, **April**, 232-233.

[14] J. Cotterman and J. Senn, (1992) *Challenges and Strategies for Research in Systems Development*, Wiley & Sons, Chichester.

[15] J. Couger, M. Colter and R. Knapp, (1982) *Advanced System Development Feasibility Techniques*, Wiley & Sons, New York.

[16] B. Curtis, H. Krasner and N. Iscoe, (1988) A field study of the software design process for large systems. *Communications of the ACM*, **November**, 1268-1287.

[17] E. Downs,P. Clare and I. Coe, (1992) *Structured Systems Analysis and Design Method: Application and Context*. Prentice-Hall International(UK), Hertfordshire.

[18] K. Eisenhardt, (1989) Building theory from case study research, *Academy of Management Review*, **14**, 4, 532-550.

[19] B. Fitzgerald, (1994), Whither systems development: time to move the lamppost?, in Lissoni *et al*., (eds), *Proceedings of Second Conference on Information Systems Methodologies*, pp.371-380.

[20] B. Fitzgerald, (1996), Formalised systems development methodologies: a critical perspective, *Information Systems Journal*, **6**, 1, 3-23.

[21] C. Floyd, (1987) Outline of a paradigm change in software engineering. In *Computers and Democracy: A Scandinavian Challenge*. Bjerknes, G., Ehn, P., and King, M. (Eds), Avebury Gower, Brookfield Vermont.

[22] S. Folkes and S. Stubenvoll, (1992) *Accelerated Systems Development*, Prentice Hall, London.

[23] A. Friedman, (1989) *Computer Systems Development: History, Organization and Implementation*, Wiley & Sons, Chichester.

[24] C. Hardy, B. Thompson and H. Edwards, (1995) Problems associated with the customisation of structured methods, in Jayaratna, N., Miles, R., Merali, Y., and Probert, S. (eds), *Proceedings of Third Conference on Information Systems Methodologies*, 211-220.

[25] S. Holloway, (1989) *Methodology Handbook for Information Managers*, Gower Technical, Aldershot.

[26] D. Hough, (1993) Rapid delivery: an evolutionary approach for application development, *IBM Systems Journal*, **32**, 3, 397-419.

[27] M. Jackson, (1983) *System Development*, Prentice-Hall, New Jersey.

[28] N. Jayaratna, (1994) *Understanding and Evaluating Methodologies*, McGraw-Hill, London

[29] A. Jenkins, J. Naumann and J. Wetherbe, (1984) Empirical investigation of systems development practices and results. *Information & Management*, **7**, 73-82.

[30] K. Kozar, (1989) Adopting systems development methods: an exploratory study. *Journal of Management Information Systems*, **5**, 4, 73-86.

[31] K. Kraemer and W. Dutton, (1991) Survey research on information systems, in Kraemer, K. (ed) *The Information Systems Research Challenge: Survey Research Methods*, Publishing Division,Harvard Business School, Boston, 3-57.

[32] S. Labovitz, (1970) The assignment of numbers to rank-order categories, *American Sociological Review*, **35**, 515-524.

[33] J. Lee, and S. Kim, (1992) The relationship bteween procedural formalisation and MIS success. *Information & Management*, **22**, 89-111.

[34] D. Leonard-Barton, (1987) Implementing structured software methodologies: a case of innovation in process technology. *Interfaces*, **17**, 3, 6-17.

[35] K. Lyytinen, (1987) A taxonomic perspective on information systems development, in Boland, R and Hirschheim, R. (eds.) *Critical Issues in Information Systems Research*, John Wiley and Sons, Chichester.

[36] M. Mahmood, (1987) System development methods- a comparative investigation. *MIS Quarterly*, **September**, 293-311.

[37] R. Mason, (1991) Commentary on survey research on information systems, in Kraemer, K. (ed) *The Information Systems Research Challenge: Survey Research Methods*, Publishing Division,Harvard Business School, Boston, 59-65.

[38] J. McKeen, (1983) Successful development strategies for business application systems, *MIS Quarterly*, **7**, 3, 47-66.

[39] P. Naur, B. Randell and J. Buxton, (1976) *Software Engineering: Concepts and Techniques*, Charter Publishers, New York.

[40] C. Necco, C. Gordon and N. Tsai, (1987) Systems analysis and design: current practices. *MIS Quarterly*, **December**, 1987.

[41] A. Olerup, (1991) Design approaches: a comparative study of information system design and architectural design. *The Computer Journal*, **34**, 3, 215-224.

[42] A. Oppenheim, (1966) *Questionnaire Design and Attitude Measurement*, Heinemann, London.

[43] P. Palvia and J. Nosek, (1993) A field examination of system life cycle techniques and methodologies. *Information & Management*, **25**, 73-84.

[44] N. Russo, J. Wynekoop and D. Walz, (1995) The use and adaptation of systems development methodologies, in Khosrowpour, M. (Ed), *Managing Information & Communications in a Changing Global Environment*, Idea Group Publishing, PA.

[45] J. Stage, (1991) The use of descriptions in the analysis and design of information systems, In Stamper *et al*. (1991) *Collaborative Work, Social Communications and Information Systems*, North Holland, 237-260.

[46] E. Stolterman, (1994) The 'transfer of rationality', acceptability, adaptability and transparency of methods, in Baets, W. (Ed), *Proceedings of Second European Conference on Information Systems*, Nijenrode University Press, Breukelen, 533-540.

[47] M. Sumner and J. Sitek, (1986) Are structured methods for systems analysis and design being used? *Journal of Systems Management*, **June**, 18-23.

[48] I. Vessey and S. Conger, (1994) Requirements specification: learning object, process and data methodologies. *Communications of the ACM*, **37**, 5, 102-113.

[49] J. Virgo, (1996) *1996 IDPM IT Skills Trends Report*, Institute of Data Processing Management.

[50] N. Vitalari and G. Dickson, (1983) Problem solving for effective systems analysis: an experimental exploration. *Communications of the ACM*, **November**, 948-956.

[51] R. Wallace and C. Mellor, (1988) Non-response bias in mail accounting surveys: a pedagogical note, British Accounting Review, 20, 2, 131-139.

[52] P. Ward, (1991) The evolution of structured analysis: Part I--the early years. *American Programmer*, **4**, 11, 4-16.

[53] D. Wastell, (1996) The fetish of technique: methodology as a social defence, Information Systems Journal, 6, 1, 25-40.

[54] D. Wastell and M. Newman, (1993) The behavioral dynamics of information systems development: a stress perspective, *Accounting, Management & Information Technology*, **3**, 2, 121-148.

[55] R. Zmud, (1983) The effectiveness of external information channels in facilitating innovation within software development groups, MIS Quarterly, 7, 2, 43-58.

Vita

Brian Fitzgerald is Senior Researcher at the Executive Systems Research Centre at University College Cork and is currently an Associate Editor for *The Information Systems Journal*. He holds a PhD from the University of London and he is actively involved in applied research projects in the areas of systems development approaches, foundations of the IS field and executive information systems. His work in these areas is published in various books and international journals, including *The Information Systems Journal*, *INFOR*, the *Journal of Information Technology*, and the *International Journal of Information Management*. He has also presented his research at international conferences and seminars in Belgium, Canada, France, Portugal, Spain, Sweden, Holland, Ireland, the US, and the UK. Having worked in industry prior to taking up a research position, he has more than 15 years experience in the IS field.

Table 1 Background Information on Respondent Organizations

Business Category	No. Of Employees	No. In IS Dept
Consult/Software House 31%	1 to 10.....12%	1 to 5 .....44%
Govt/Pub Sector/Education 7%	10 to 100.....29%	5 to 20 .....30%
Constr/Manuf/Distrib.....36%	100 to 1000.....43%	20 to 100 .....19%
Wholesale/Retail Trade.....4%	1000 to 5000 ...9%	> 100 .....7%
Finance/Insur/Real Estate 12%	> 5000.....7%	
Service/Communications.....6%		
Other 4%		

Table 2 Background Information on Development Profile

	Mean
Respondents' development experience (years)	15
Development Profile:	
Percentage development in-house	47%
Percentage development outsourced	13%
Percentage use/customisation of packages	40%
Development Project Profile:	
No. of developers per project	3.5
Duration (in months) per project	5.7
Hardware platform	
Mainframe	20%
Mini	26%
PC	33%
Mixture/Other	21%

Table 3 Methodology Usage

Percentage of Organizations not using any FSDM	60%
Percentage of Organizations using FSDM	40%
(Commercial FSDM	14%)
(Internal (based on commercial FSDM)	12%)
(Internal (not based on commercial FSDM)	14%)

Table 4 Breakdown of Methodology Usage

Dimension	Respondents Using FSDM	Respondents Not Using FSDM	Significance
Business Category	**%**	**%**	Chi-Square
Consult./Software House	15	15	
Govt./Pub Sector/Educ.	4	3	p<.0001
Constr./Manuf./Distrib.	7	29	
Wholesale/Retail Trade	1	3	
Fin./Insur./Real Estate	8	4	
Services/Communications	4	2	
Other	1	3	
No. of Employees	**%**	**%**	Chi-Square
1 to 10	4	8	
10 to 100	10	19	p<.01
100 to 1000	15	28	
1000 to 5000	6	3	
Greater than 5000	6	1	
No. in IS Department	**%**	**%**	Chi-Square
1 to 5	10	34	
5 to 20	12	18	p<.00001
20 to 100	12	7	
Greater than 100	6	1	
			Mann-WhitneyU
In-house Development	**%**	**%**	
Up to 35% in-house	8	39	p<.00001
35 - 90% in-house	24	16	
More than 90% in-house	9	4	
Development Outsourced	**%**	**%**	
Up to 10%	26	30	p<.05
More than 10%	15	30	
Package Customisation	**%**	**%**	
Less than 10%	11	6	
10 - 55%	24	28	p<.001
More than 55%	6	25	
			Mann-WhitneyU
No. of Developers	**%**	**%**	
Fewer than 5	26	55	p<.00001
5 or more	16	3	
Project Duration	**%**	**%**	
Up to 9 months	26	49	p<.00001
Longer than 9 months	17	8	

Table 5(a) Average Percentage of Development Time Allocated to Development Activities

	Avg. percentage of dev. time allocated by those using FSDM	**Avg. percentage of dev. time allocated by those not using FSDM**
Activity	**%**	**%**
Systems Planning	10	10
Systems Analysis	17	14
Systems Design	15	12
Programming	28	31
Testing	17	17
Installation	8	10
Evaluation	3	4
Other	2	1

Table 5(b) Percentage of Respondents Using Various Development Techniques

	Using FSDM	**Not using FSDM**
Tools/Techniques	**%**	**%**
Joint Application Design (JAD)	31	20
Prototyping	75	57
Data Flow Diagramming	71	37
Entity Relationship Modelling	63	19
Entity Life Histories	19	6
Flow Charting	55	35
Data Dictionary	74	34
Process Mini-Specifications	40	25
Structured Walkthrough	48	23
Other	9	5

Table 6 Discrepancies in Methodology Usage

	SSADM (n=8)		**Information Engineering** (n=6)	
	Percentage Time Allocated		**Percentage Time Allocated**	
Activity	**Min. %**	**Max. %**	**Min. %**	**Max. %**
Systems Planning	5	20	5	15
Systems Analysis	10	20	14	20
Systems Design	10	20	5	20
Programming	20	50	15	40
Testing	10	30	10	30
Installation	5	15	5	15
Evaluation	0	15	1	5
Other	0	5	1	5

Table 7 Ranking of Contribution of Various Factors to Development
(1 - most important etc.)

Factor	Using FSDM	Not using FSDM
User participation	1	1
Adequate time-scale and budget	2	3
Developer knowledge of application domain	3	2
Adherence to a formalised SDM	4	n/a
Developer technical expertise	4	4

Table 8 Trend in Methodology Adoption

Trend	Using FSDM	Not using FSDM
Moving towards increased use of FSDMs	61%	21%
Moving away from FSDMs	3%	-
Remaining as is	36%	79%