8 *RE-SEARCHING* COMMONALITY DIFFERENTLY: Subjectively Replicating a Theory of Multimedia Systems Development

Brian Webb
Queen's University of Belfast
Belfast, N. Ireland

Abstract *Can subjective replication generate valid and worthwhile knowledge? A theory of multimedia systems development (MSD) generated using a content analysis method is tested using a grounded theory method. The theory—that two distinct communities of software engineers and graphic designers exist within MSD—is confirmed. In fact, the test finds more differences and less commonalities. This finding has implications for the development of MSD methodologies, and for the education and training of MSD practitioners. The conclusion is that subjective replication is worthwhile but must be done carefully because of problems with the application of methods in this relatively unexplored information systems research space.*

Keywords Multimedia systems development, software engineering, graphic design, grounded theory, research methods, replication

1 INTRODUCTION

In their critique of information systems research Berthon et al. (2002) argue that "while much attention has been paid to methodological rigor and pluralism in MIS research, replication has received less attention. [This is because] in our rush for new knowledge, generation rather than replication, *search* rather than *re-search*, predominates" (p. 416).

Please use the following format when citing this chapter:

Webb, B., 2007, in IFIP International Federation for Information Processing, Volume 235, Organizational Dynamics of Technology-Based Innovation: Diversifying the Research Agenda, eds. McMaster, T., Wastell, D., Ferneley, E., and DeGross, J. (Boston: Springer), pp. 101-114.

In an effort to encourage more and better replication research, they present a framework "to conceptualize, structure and guide replication efforts" (p. 418). This framework seeks to "explicate the possible research combinations of problem, theory, method and context" and maps out various research strategies on a continuum from zero degrees of freedom (pure replication) to three degrees of freedom (pure generation). The various strategies are summarized as follows:

- *Pure replication*. This strategy constrains all three dimensions of the research to be as close as possible to the original study or studies. That is, the same theoretical framework, the same methodology and the same phenomenological context are employed. ("Pure replicative studies seem to be very rare in MIS, with literature reviews identifying no pure replicative studies in any of the major journals," Berthon et al. 2002, p. 423).

- *Context-only extension.* This strategy takes an existing theory and method and applies it in a different context (e.g., the use of SERVQUAL instrument within IS to measure user satisfaction; Parasuraman et al. 1988).

- *Methods-only extension*. This strategy takes an existing theory and context and links them through a different research method. (e.g., using a different method to measure service quality in IS; Van Dyke et al. 1997).

- *Theory-only extension*. This strategy takes an existing method and context but employs a new theory to explain the results (e.g., extending the technology acceptance model to explain information technology utilization behavior; Dishaw and Strong 1998).

- *Method/context extension*. This strategy takes a new method and context, but employs existing theory to explain the results (e.g., using an existing theory of IT enabled change to explain change in a different organizational setting, using a different research method; Manzoni and Angehrn 1998).

- *Theory/context extension.* This strategy takes an existing method but applies it to a new context and employs a new theory to explain the results (e.g., using alternative theories of software project management rather than simple escalation behavior to understand the dynamics of commitment to software projects; Keil et al. 2000).

- *Theory/method extension*. This strategy takes a new theory and method and applies them to an existing context (e.g., extending the TAM to investigate behavioral intention to use an information systems, while also using a different method of data analysis; Jackson et al. 1997).

Whereas "researchers working implicitly or explicitly within the objectivist tradition have stressed the need for replication and indeed its central role in science" (Berthon et al. 2002, p. 418) replication within the subjectivist tradition (Burrell and Morgan 1979; Lee 1991) is much rarer. In the subjectivist tradition, "replication is not concerned with accuracy (the building of an ever more accurate representation of some external reality)

but depth of understanding (building richness of phenomenological experience" (Berthon et al. 2002, pp. 418-419). Although they discuss briefly the meaning of this statement, Berthon et al. readily acknowledge that their work "fits primarily into the objectivist tradition which accords replication a significant role in the research process" (p. 419) and they offer no further insights into how subjectivist replication may be done.

In this paper, the grounded theory method (Glaser and Strauss 1967) is used to test a published theory of multimedia systems development (Gallagher and Webb 2000). Since that theory was generated using a method of content analysis (Miles and Huberman 1994) and since all other research parameters remain constant, this is a methods-only extension. Thus the same problem is being investigated (differences and difficulties between software engineers and graphic designers in MSD), the same theory is being used to investigate that problem—Kuhn's (1970) theory of scientific communities or paradigms—and the same investigative context (the same data) and the same interpretive context (the same researchers) applies. Only the method is different.

Berthon et al. accept that "each dimension of research space can comprise sub-dimensions or levels." They therefore distinguish between data generation which is "methods of data production, including measurement issues, survey processes, interview techniques, observational protocols, etc." and data analysis which includes "textual analysis, statistical analysis, visual methods, etc." (p. 422). They give as an example of methods-only extension Pitt el al (1997), who responded to Van Dyke at al.'s (1997) extension of their earlier study of service quality (Pitt et al. 1995) by using the same data set and the same underlying theory, but changing the method of analyzing the data.

While "illustrations of the various strategies do exist in the literature but they are few and far between" (Berthon et al. 2002, p. 425), the goal is to make a contribution to knowledge by setting out an example of subjectivist replication in a way that is explicit and accessible to other researchers. This means identifying problems and limitations encountered. We also see value in testing a theory that has important implications for MSD methods development and project management, but that is yet to be tested, although 6 years old and cited in the literature (e.g., Lang 2003). The paper proceeds as follows. First, the test theory is outlined. Then we set out the test method, identifying points of departure with the method of data analysis used to generate the test theory. The test results are summarized and the limitations of the research acknowledged. Finally the implications of the research for IS researchers and MSD practitioners are discussed.

2 TEST THEORY

Multimedia systems development consists of a small number of generic steps—proposal, design, and production—analogous to the analysis and design phases of the software development life cycle; it involves iteration, prototyping, and top down and bottom up design. The field is characterized by complexity (of product and process), innovation (in products and processes), and methodological pluralism. Attempts to reconcile and manage multidisciplinary development teams have produced a plethora of domain-specific methods, some based on software engineering approaches (Constantine and Lockwood 2001), some based on a graphic design approaches (Mallon and Webb 2000), and some on hybrid approaches (De Troyer 2001).

Yet underlying tensions remain between the two disciplines (Lowe and Eklund 2002) and no single method is preeminent. A common feature of many of these methods is a desire to retain flexibility in response to the prevailing environment by means of contingency or cognitive fit (Fitzgerald et al. 2003), morphology (Baskerville and Stage 2001) design rationales (Firesmith and Henderson-Sellers 2002), and agile software development (Cockburn 2002). Nevertheless, there is evidence that many methodologies are not widely used in practice, or are not considered useful (Lang 2003).

Multimedia systems development is, by the very nature of the designed artefact, a multidisciplinary activity requiring the cooperation and collaboration of various professional communities, chief among which are software engineering and graphic design. Studies into multimedia systems development have identified underlying problems between software engineers and graphic designers (Barry and Lang 2001; Carstensen and Vogelsang 2001; Gallagher and Webb 1997, 2000; Lang 2003; Lowe and Eklund 2001).

Gallagher and Webb (2000) argued that two distinct communities (or paradigms) of software engineering (SE) and graphic design (GD) exist within multimedia development. They claim their theory can be used in method evaluation or feature analysis where "common elements can act as the basis of common criteria by which to judge methods, while non common elements can be used to derive features/method requirements that are specific to each community" (p. 65) and in methods integration when "method fragments [such as story-boarding] are combined to produce a single method for a particular context" (p. 67).

For Kuhn (1970), a paradigm represents a belief system that encompasses those concepts, models, assumptions, and metaphysical principles that are shared within each community (where a community is a distinct group of people composed of practitioners of a scientific speciality who have undergone a similar education and have drawn similar lessons as a result. Specifically, a paradigm (also referred to as a disciplinary matrix, or DM) may be defined as

- *Symbolic generalizations* (such as mathematical formula or scientific laws which are widely used by group members and readily justified by them)
- *Belief(s)* in heuristic and metaphysical models (preferred and permissible analogies and metaphors used to explain phenomena)
- *Values*, which are more widely shared among different communities than symbolic generalizations and models (for example a commitment to quantitative over qualitative methods)
- *Exemplars* or *paradigm as shared example* (examples of how theories are applied in practice; essential for group cohesiveness)

To generate their theory, Gallagher and Webb analyzed 12 textbooks and 20 semi-structured interview transcripts using content analysis techniques drawn from Miles and Huberman (1994). Segments of "talk" within each text interview were coded using codes that indicated key themes. For example "DM-B" indicated the presence of a belief statement. Each source was analyzed systematically and a list of DM elements was compiled. Although 67 DM elements are common across both disciplines, suggesting nondistinct communities, 58 elements for software engineering and 54 elements for

graphic design were identified as noncore elements (NCEs). The appendices to Gallagher and Webb's paper list each type of element for each discipline. This is helpful in this (test) context since it is possible for the interested reader to examine directly the correspondence of results produced in the development of the theory with the extensions to that theory reported below.

3 TEST METHOD

The theory—that two distinct paradigms exist—is unconfirmed if significant commonality is found between the two disciplines. Gallagher and Webb (2000) found some commonality but regarded it as weak. Because the main focus of Gallagher and Webb's study was on practitioners, only interview transcripts were analyzed as test data. At the practitioner level, differences between the two communities are most likely to be found, while commonalities that do exist are much more likely to be meaningful.

For the test analysis, two interview transcripts were unavailable (they simply could not be located) and, as both of these were from the same MSD community (software engineers), two graphic design transcripts were dropped. Parity of data sources was deemed important in our grounded theory analysis because of its reliance on the constant comparative analysis technique, where emerging concepts are scrutinized against similar and dissimilar concepts in order to tease out commonalities and differences in the data (Strauss and Corbin 1990). The semi-structured interview format, used in the original interviews, is given as Appendix A. Appendix B lists the 16 interviewees used to generate the test data.

A concept is a significant word, phrase, statement, or paragraph found in the transcript. At this stage the purpose of the analysis was simply to generate data that was descriptive of the phenomenon under study. No explicit attempt was made to search for and identify DM elements. An entry was created in the database by cutting and pasting the original text or by keying a summary of what was said. Each entry was given a label or name, its exact location in the transcript was identified, and three keywords recorded. In addition a memo was written further describing the concept (what Strauss and Corbin call an "operational memo"), or developing it analytically and conceptually (a "theoretical memo") and this was appended to the concept (database) record. Thus, while the concept records themselves were descriptive, each fostered analytical memos within which the original concept was further developed. Moreover the three keywords per concept provided a quick and easy way to compare concepts and an initial lead to categorization.

Although superficially similar to content analysis (a process of deconstruction and re-construction of data, using codes to represent units of text or talk), the open coding phase of grounded theory analysis differs significantly in the breadth and depth of its analysis. Content analysis is limited to a number of specific analytical techniques such as producing a contact summary sheet (Miles and Huberman 1994, p. 52), or developing codes using a mixture of an inductive approach and a "start list" approach (Miles and Huberman 1994, p. 58), but these techniques are neither informed by nor integrated with a particular methodological approach. Open coding, on the other hand, is intimately and inextricably bound up with the grounded theory method, and cannot be applied honestly except as part of that method. The application of open coding necessarily requires the

application of associated and ancillary analyses. In addition to the constant comparison of data already referred to, these are theoretical sensitivity, theoretical sampling, and theoretical saturation.

Theoretical sensitivity "refers to the attribute of having insight, the ability to give meaning to data, the capacity to understand and the capability to separate the pertinent from that which isn't" (Strauss and Corbin 1990, p. 42). Sources of theoretical sensitivity are the literature, professional and personal experiences, and through continued interaction with the data itself. Theoretical sensitivity is a personal attribute of the researcher "it indicates an awareness of the subtleties of meaning of data" but it can be developed through techniques such as "stepping back from the data" and "maintaining an attitude of scepticism" (Strauss and Corbin 1990, p. 41) No such techniques accompany content analysis, although the stance may be assumed or even encouraged implicitly.

Theoretical sampling "is sampling on the basis of concepts that have proven theoretical relevance to the emerging theory [where] theoretical relevance indicates that certain concepts are deemed significant because they are repeatedly present or noticeably absent when comparing incident after incident" (Strauss and Corbin 1990, p. 176). Even where the purpose of a grounded theory study is not to produce a theory but, for example, to produce a rich or thick description of a phenomenon, theoretical sampling drives the analysis. Again no equivalent can be found in content analysis, even where codes are developed inductively. Identifying and developing codes using a start-list approach is incompatible with both theoretical sensitivity and theoretical sampling because of the imposition of *a priori* assumptions on the data.

Theoretical saturation is knowing when to stop. Theoretical saturation (Glaser 1978, pp. 124-126; Glaser and Strauss 1967, pp. 61-62) means that no new or relevant data seems to emerge during the analysis, indicating that the analysis is conceptually dense. The distinction between first and second level analysis (Miles and Huberman 1994), wherein data is developed into codes and then codes are developed into second level codes or patterns, suggests that patterns are the pinnacle of the analysis, but it is not clear how that pinnacle is achieved, or what it looks like when you get there. The grounded theory notion of saturation is more explicit, offering greater depth to the analysis.

Compared to content analysis, open coding conducted under the grounded theory method should produce more and better (more meaningful or conceptually dense) codes and the relationships between codes should be more easily developed. That is, the rudiments of a theory should already be in place at this stage of the analysis (Strauss and Corbin 1990). In the test study, 160 concepts were produced from 16 interview transcripts. This compares to 179 DM elements produced from 20 interview transcripts and 12 textbooks in the original Gallagher and Webb study. However, although interviews are regarded as a more relevant source of data, textbooks represent important repositories of accepted knowledge within the discipline (Kuhn 1970) and it is no surprise that the majority of DM elements identified in the original study came from this source.

This paper reports a methods extension by using open coding as an alternative to content analysis. It is not necessary for either of these approaches to be labeled a method in order to be able to justify this extension (although we argue that open coding can be considered a much better approximation to a method than content analysis). Rather, as was the case with Pitt et al. (1997) and Adams et al. (1992), the methods extension may be justified even where there is only a change to the techniques of data analysis. Berthon

et al. cite both of these as examples of methods-only extensions where only the means of statistical analysis changed. For example, "Adams et al. used multiple-indicator structural equation modeling (LISREL), rather than multivariate regression of averaged scores, as the method of analysis" (Berthon et al. 2002, p. 424).

4 TEST RESULTS

The data analysis produced 160 concepts. These concepts were then compared directly to the list of DM elements produced by Gallagher and Webb. Since the ultimate purpose of the test was to test for the existence of two distinct communities of software engineering and graphic design in MSD, concepts were assigned to DM elements (by discipline) rather than vice versa. Assignment was on the basis of concept name, keywords description, and perceived fit with the DM element. Not all concepts could be assigned, and some concepts were assigned to more than one DM element. Table 1 shows the average number of concept occurrences per DM element per discipline.

The test data shows less commonality across all DM elements, for both software engineering and graphic design. This confirms the theory that there are distinct communities (major premise) and that the commonality that does exist is insignificant (minor premise). For example, whereas in the Gallagher and Webb study beliefs were found on average over five times in each software engineering transcript and over 3.5 times in each graphic design transcript, in the test study analysis this dropped to less than two times in each case. Some commonality clearly exists but this is as not strong. No symbolic generalizations or exemplars were found at all. Eleven belief statements and four value statements were found only in one or other of the disciplines, not both. Twelve value statements were not found in either discipline (see Appendix C for a full list of concepts by DM element).

Far fewer values were identified, reflecting in part the sequence of the analysis. As concepts were allocated to DM elements on the basis of best fit, they tended to be assigned as beliefs rather than values (where the distinction between beliefs and values was not always clear and beliefs were the first DM elements to be coded). Only the number of (distinct) sources (interview transcripts) within which a DM element was identified was recorded. In fact, a DM element may have multiple occurrences within the same source, which is an indication of depth of support for the target theory within discipline, rather than breadth of support across disciplines

Table 1. **Average Number of DM Elements Found by Discipline**

	SE/10	GD/10	SE/8	GD/8	SE Diff.	GD Diff.
Symbolic Generalizations	10	10	0	0	-10	-10
Beliefs	5.30	3.70	1.97	1.86	-3.33	-1.84
Values	2.95	3.75	0.95	0.66	-2.0	-3.09
Exemplars	10	0.10	0	0	-10	-0.10

GD = Graphic Design; SE = Software Engineering

5 LIMITATIONS AND IMPLICATIONS FOR IS RESEARCHERS

Berthon et al. (200, p. 4212) identify three primary IS stakeholder groups that may be interested in the application of their research framework: producers, consumers, and stewards. Each of these groups includes, but is not limited to, research practitioners. Yet Berthon et al. do not set out how their framework may be applied in each case. They claim that "the framework permits the planning of new research streams, the identification of opportunities, and the ensuing development of strategies and approaches to existing research streams" (p. 425), but there is little practical advise to researchers seeking to conduct subjectivist replication. This paper is an illustration and example of one approach to such research. It is intended to encourage more IS researchers to conduct this kind of research. Replication research is lower cost and lower risk than pure generation research, and the publication of more and better examples of such research can only increase the probability that such research will be undertaken in the future. However, there are practical difficulties and limitations that must be acknowledged.

Berthon et al. acknowledge that time inevitably changes both researcher and subject (p. 420). Although in a methods-only replication the data is held constant (the same data set is analyzed), subjective research is inevitably the product of the interaction of researcher and subject. So even, when the same researcher analyzes the same dataset, using the same method, some differences must be assumed because of time lapse. This highlights a weakness in the Berthon et al. framework, which is (necessarily) constructed on pure or ideal types. In practice, it is impossible to hold context parameters constant from study to study in social sciences research.

Berthon at al. cite two examples of methods-only extensions. In addition to Pitt et al. (1997), in which the same dataset was analyzed by the same researchers using a different method, they cite Adams et al. (1992), in which both the dataset and the researchers change. How can this be a methods-only extension to one degree of research freedom when two other parameters are also changed? More accurately, this is a methods/context extension (p. 424). Significantly, both examples of methods-only extension are in the objectivist tradition, with both using different methods of statistical analysis from the original studies. It is very much more difficult to replicate research settings in the subjectivist tradition.

One of the stated purposes of the Berthon et al. framework is to permit relevant stakeholder groups to classify and evaluate research. But if the framework itself is inconsistent—as with its treatment of method-only extensions—then it will be limited at best. At worst, an inconsistent or confusing framework, lacking clear definitions and examples, may only encourage the very misclassifications and inappropriate evaluations of research that it is designed to avoid.

6 IMPLICATIONS FOR MSD PRACTITIONERS

Whereas Gallagher and Webb (2000, pp. 65-66) postulate the possible contributions of the study to the development of MSD methods, specifically in the areas of methods evaluation and methods integration, we should now seek—on the basis of the results of

this test—to moderate their assertions. With less commonality and greater differences than originally thought, it is even less likely that methods developed predominantly upon a common core of design will be successful. On the contrary, approaches that value and accommodate differences (while still also recognizing that some commonality does exist) are much more likely to be accepted by the two communities. Methods evaluation by way of a feature analysis of common and non-common elements remains perfectly feasible but requires some adjustments to the selection criteria to reflect stronger differences and weaker commonalities. Methods integration also remains possible but will be undermined where the basis of integration relies too heavily upon a common core of design, or upon a single design process or artefact (e.g., story-boarding).

Although Gallagher and Webb did not speculate upon the possible implications of their study for MSD practitioner education and training, the replication of their theory makes such speculation more justified. Given the very real differences between the two communities in terms of beliefs, values, and language, interventions to improve mutual understanding should precede any attempt to impose a particular methodology (new or existing). This is not to argue that the two communities should become more alike. On the contrary the challenges of MSD are most likely to be met where the distinctive capabilities and contributions of the two communities are fully recognized, but to call upon researchers, educationalists, method developers, and practitioners to study and act upon a better appreciation of what we now know to be the same and to be different about software engineering and graphic design approaches to MSD.

7 CONCLUSIONS

A theory of MSD is replicated along one degree of freedom of the Berthon et al. (2002) replication space: methods-only extension. Gallagher and Webb's (2000) theory, that two distinct communities of software engineering and graphic design exist within MSD, is confirmed but the test data indicates that the theory is understated. Less commonality and more differences were found between the two communities. This test has implications for methods development and for the education and training of practitioners, since the importance of the "common" element in any such initiative is questioned. A relatively neglected corner of the IS research space has been illustrated and further research outlined.

We conclude that subjective replication is worthwhile because it can be used to test theories that are important to IS practice. These theories may otherwise remain untested because of a lack of insight into how such testing may be done. However, the application of subjectivist replication must be done carefully. Test theories and test methods are often difficult to understand, to reconcile, and to demonstrate. Conceptual frameworks such as the one put forward by Berthon et al. can help the researcher but will be limited in their applicability until they themselves have been empirically tested.

References

Adams, D. A., Nelson, R. R., and Todd, P. "Perceived Usefulness, Ease of Use, and Usage of Information Technology: A Replication," *MIS Quarterly* (16:2), 1992, pp. 227-247.

Barry C., and Lang, M. "A Survey of Multimedia and Web Development Techniques and Methodology Usage," *IEEE Multimedia* (8:3), 2001, pp. 52-60.

Baskerville, R., and Stage, J. "Accommodating Emergent Work Practices: Ethnographic Choice of Method Fragments," in N. L. Russo, B. Fitzgerald, and J. I. DeGross (eds.), *Realigning Research and Practice in Information Systems Development: The Social and Organizational Perspective*, Boston: Kluwer Academic Publishers, 2001, pp. 11-28.

Berthon, P., Pitt, L., Ewing, M., and Carr, C. L. "Potential Research Space in MIS: A Framework for Envisioning and Evaluating Research Replication, Extension and Generation," *Information Systems Research* (13:4), 2002, pp. 416-427.

Burrell G., and Morgan, G. *Sociological Paradigms and Organizational Analysis*, London: Heinemann, 1979.

Carstensen, P. H., and Vogelsang, L. "Design of Web-Based Information Systems: New Challenges for Systems Development?," in *Proceedings of the 9th European Conference on Information Systems*, Bled, Slovenia, June 27-29, 2001, pp. 536-547.

Cockburn, A. *Agile Software Development*, Reading, MA: Addison Wesley, 2002.

Constantine, P. H., and Lockwood, L. A. "Usage-Centered Engineering for Web Applications," *IEEE Software* (19:2), 2001, pp. 42-50.

De Troyer, O. "Audience-Driven Web Design," in M. Rossi and K. Siau (eds.), *Information Modeling in the New Millennium*, Hershey, PA: Idea Group Publishing, 2001, pp. 442-461.

Dishaw, M. T., and Strong, D. M. "Supporting Software Maintenance with Software Engineering Tools: A Computed Task-Technology Fit Analysis," *Journal of Systems and Software* (44:2), 1998, pp. 107-120.

Firesmith, D. G., and Henderson-Sellers, B. *The OPEN Process Framework*, Reading, MA: Addison Wesley, 2002.

Fitzgerald, B., Russo, N. L., and O'Kane, T. "Software Development Method Tailoring at Motorola," *Communications of the ACM* (46:4), 2003, pp. pp. 64-70.

Gallagher, S., and Webb, B. "Competing Paradigms in Multimedia Systems Development: Who Shall Be the Aristocracy?," in *Proceedings of the 5th European Conference on Information Systems*, Cork, Ireland, June 19-21, 1997, pp. 1113-1120.

Gallagher, S., and Webb, B. "Paradigmatic Analysis as a Means of Eliciting Knowledge to Assist Multimedia Methodological Development," *European Journal of Information Systems* (9), 2000, pp. 60-67.

Glaser, B. *Theoretical Sensitivity*, Mill Valley, CA: Sociology Press, 1978.

Glaser, B., and Strauss, A. S. *The Discovery of Grounded Theory: Strategies for Qualitative Research*, London: Wedenfeld and Nicholson, 1967.

Jackson, C. M., Chow, S., and Leitch, R. A. "Towards an Understanding of the Behavioral Intention to Use Information Systems," *Decision Sciences* (28:2), 1997, pp. 357-254.

Keil, M., Mann, J., and Rai, A. "Why Software Projects Escalate: An Empirical Analysis and Test of Four Theoretical Models," *MIS Quarterly* (24:4), 2000, pp. 631-664.

Kuhn, T. S. *The Structure of Scientific Revolutions* (2nd ed.), Chicago: University of Chicago Press, 1970.

Lang, M. "Multimedia Systems Development: A Comparative Study of Software Engineers and Graphic Designers," *Communications of the AIS* (12), 2003, pp. 242-257.

Lowe, D. B., and Eklund, J. "Client Needs and the Design Process in Web Projects," *Journal of Web Engineering* (1:1), 2002, pp. 23-36.

Mallon, B., and Webb, B. "Structure, Causality, Visibility and Interaction: Propositions for Evaluating Engagement in Narrative Multimedia," *International Journal of Human-Computer Studies* (53), 2000, pp. 269-287.

Manzoni, J. F., and Angehrn, A. A. "Understanding Organizational Dynamics of IT Enabled Change: A Multimedia Simulation Approach," *Journal of Management Information Systems* (14:3), 1998, pp. 109-140.

Miles, M. B., and Huberman, A. M. *Qualitative Data Analysis: An Expanded Sourcebook*, London: Sage Publications, 1994.

Parasuraman, A., Zeithaml, V. A., and Berry, L. L. "SERVQUAL: A Multiple Item Scale for Measuring Customer Perceptions of Service Quality," *Journal of Retailing* (64), Spring 1988, pp. 12-40.

Pitt, L. F., Watson, R. T., and Kavan, C. B. "Measuring Information Systems Service Quality: Concerns for a Complete Canvas," *MIS Quarterly* (21:5), 1997, pp. 209-219.

Pitt, L. F., Watson, R. T., and Kavan, C. B. "Service Quality: A Measure of Information Systems Effectiveness," *MIS Quarterly* (19:2), 1995, pp. 173-187.

Strauss, A., and Corbin, J. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*, Newbury Park, CA: Sage Publications, 1990.

Van Dyke, T. P., Kappelman, L. A., and Prybutok, V. P. "Measuring Information Systems Service Quality: Concerns Othe Use of the SERVQUAL Questionnaire," *MIS Quarterly* (21:2), 1997, pp. 195-208.

About the Author

Brian Webb is a senior lecturer in Information Systems, School of Management and Economics, Queen's University of Belfast, N. Ireland. He is a former Distinguished Erskine Fellow, in the Department of Accounting, Finance, and Information Systems, Faculty of Commerce, University of Canterbury, New Zealand. He holds a Bachelor's degree from Queen's, an MBA from the University of Ulster, and a Ph.D. from University College London. Prior to becoming an academic, he worked as a systems analyst in both the United Kingdom and the United States. He may be contacted at b.webb@qub.ac.uk.

Appendix A: Interviews

Interviewee	Experience†	Education
GD1	12 years	BA Graphic Design
GD2	24 years	BA Graphic Design
GD3	20 years	BA Graphic Design
GD4	11 years	BA Graphic Design
GD5	08 years	BA Graphic Design
GD6	03 years	BA Visual Communications.
GD7	04 years	BSc Computing & Design
GD8	01 year	BA Visual Communications.
SE1	18 years	HND Computing
SE2	10 years	BSc Eng. MSc Computing.
SE3	14 years	BSc Eng. MSc Computing.
SE4	13 years	BSc Computer Science
SE5	06 years	BSc Computer Science
SE6	10 years	BSc Computer Science
SE7	01 year	BSc Computing & IS
SE8	03 years	BSc Computer Science

†Experience refers to total years experience in the referent field of graphic design or software engineering. As the number of years experience in the subfield of digital interactive multimedia was not always clear, this is not shown. In general, those with the least experience overall tend to be those with experience only in the multimedia field.

Appendix B: Interview Questions

[1] Attitudinal Information
- Q1: How would you define software engineering/graphic design?
- Q2: How would define multimedia?
- Q3: What is your attitude concerning the current state of these fields?
- Q4: What do you believe your field has to offer multimedia?
- Q5: What are the differences between your traditional field and that of Multimedia? What is the same?

[2] Development Activity
- Q6: Does multimedia development differ from traditional development within your field? (For example, the process?)
- Q7: Could you describe the multimedia development process (i.e., stages)?
- Q8: What are the most important aspects (stages) of this multimedia process? Why?
- Q9: Do you rely on a formal approach or methodology during a project?
- Q10: Do you have any particular views on how multimedia development should be tackled?
- Q11: What do you see as the key problems that are faced during multimedia development?

[3] Development Environment
- Q12: How would you describe the nature of your development environment (for example, formal/informal; team/individual)?
- Q13: If you undertake work as part of a team, do you have any particular attitudes or concerns regarding team development (for example, team leadership/team skills/team structure)?
- Q14: What do you see as the essential characteristics of an ideal development environment?

[4] Project Management
- Q15: How do you monitor/measure progress during a project? (Criteria for success? Formal reviews? Meetings?)
- Q16: How would define the success/failure of a project?

[5] Design
- Q17: How would you define the term/activity *design* within the context of your particular discipline?
- Q18: How does this concept change within the context of interactive mMultimedia?
- Q19: Do you have any particular views about how design should be carried out?
- Q20: How would you distinguish good design from bad design?
- Q21: What do you feel are the essential principles of design?
- Q22: What factors influence design?

[6] Quality
- Q23: How do you define *quality*?
- Q24: How do you distinguish between high quality and poor quality?
- Q25: If asked to evaluate the quality of a product, how would you approach this?
- Q26: What particular aspects would you focus on?

Appendix C: Allocation of Concepts to Elements

	Theory Data (n = 10)	**Test Data (n = 8)**
Authoring packages/scripting languages **[Symbolic Gen.]**	10	10
A formalized design approach is worthwhile in some cases **[Beliefs..]**	9	22
Abstraction	3	321
Almost always the client doesn't know what they want	4	644
Analyze before design	4	232
Awareness of practical constraints	8	825
Choice of design method/strategy depends on nature of problem	10	523
Collaborative design	8	1024
Compromise	4	212
Decomposition	7	122
Design for change	4	112
Design for the lowest denominator	3	41
Design is a continuous process with no clear barriers to indicate where it begins and ends	2	25
Design is a creative problem solving activity	1	424
Design is a pivotal step in any development	7	135
Design is an iterative process—adding greater formality and detail	8	652
Design is open to interpretation	1	423
Design process is built up from theory, principles and/or heuristics	4	131
Design process is difficult to formalize and define	3	24
Design reviews are a good way to monitor progress	7	311
Design usually progresses from the higher levels to the lower levels	10	422
Designer's goal is to produce a representation or model of an entity that will later be built	6	353
Designers need direct contact with customers	7	233
Designers should avoid detail too early in the design process	6	211
Everything that is designed should be justified or have a reason for being	1	916
Fitness for purpose	1	212
Generate alternatives	4	71
It is generally good design practice to attempt some sort of design on paper first	5	61
It is vital to get agreement on the design specification as early as possible	2	221
Modularity	5	11
Planning—producing plans	2	411
Prescriptive design approaches are unhelpful	3	211
Prototyping	9	612
Separation of concerns	10	42
Some level of client/user involvement in the design process is required	7	632

	Theory Data (n = 10)	**Test Data (n = 8)**
Successful design requires the early involvement of all concerned.	7	42
Team effort in design can create problems	8	52
Testing is an integral aspect of designing	7	31
Accountability in design **[Values..]**	1	111
Accuracy	1	111
Appropriateness	5	8
Clarity	3	5
Coherence	2	81
Cohesion of design elements	1	3
Consistency	5	621
Elegance	4	12
Flexibility on part of the designer	4	7
Functionality	3	2
Independence of design components	3	2
Inventiveness/Innovation	3	622
Originality	1	9
Pragmatism	1	222
Precision	1	12
Professionalism	1	1
Readability	1	5
Relevance	1	2
Reuse	5	12
Robustness	7	1
Simplicity	6	432
Understandability	1	2
Usability	8	442
Visual aesthetics/appeal beauty	3	824
Analyze design cycle **[Exemplars]**	10	1