



Provided by the author(s) and University of Galway in accordance with publisher policies. Please cite the published version when available.

Title	Reconsidering the "Software Crisis": A Study of Hypermedia Systems Development
Author(s)	Lang, Michael
Publication Date	2003
Publication Information	Lang, M. (2003) Reconsidering the "Software Crisis": A Study of Hypermedia Systems Development. In Proceedings of IADIS International WWW/Internet 2003 Conference, Algarve, Portugal, November 5-8, Vol. 1, pp. 307-313.
Item record	<a href="http://hdl.handle.net/10379/1585">http://hdl.handle.net/10379/1585</a>

Downloaded 2024-05-14T04:24:40Z

Some rights reserved. For more information, please see the item record link above.



# **RECONSIDERING THE “SOFTWARE CRISIS”: A STUDY OF HYPERMEDIA SYSTEMS DEVELOPMENT**

Michael Lang

*National University of Ireland, Galway*

*Michael.Lang@nuigalway.ie*

## **ABSTRACT**

With the advent and growth of the Web, hypermedia information systems have propagated within and beyond organizations. Much concern has been expressed about the quality of hypermedia systems being developed and the apparent absence of disciplined development practices. There has been talk that the infamous “software crisis” is afflicting hypermedia systems development, allegedly brought about by shoddy project management, inadequate requirements analysis and planning, and ad hoc “quick and dirty” development approaches. This paper reports on the findings of a survey of 438 organisations in Ireland, the objectives of which were to test the validity of assertions of a “hypermedia crisis”, and to explore what mechanisms if any are being used to guide and control hypermedia systems development. The preliminary conclusions are that hypermedia development is much more disciplined than popularly believed, and the state of practice is much healthier than depicted by many academic researchers.

## **KEYWORDS**

Hypermedia; Web development; systems development methods; software crisis.

## 1. INTRODUCTION

Although its conceptual origins can be traced back a few decades, it is only recently that hypermedia has become popularised through its ubiquitous incarnation as the Web. Of course, not all hypermedia systems are Web-based, nor can all Web-based systems be classified as hypermedia. Nevertheless, the Web is now the most significant platform for hypermedia systems implementation. Because of the sudden emergence and growth of the Web, and the newness and dynamism of development environments, concern has been expressed about the quality of systems being delivered. Numerous authors speak of a hypermedia / Web “crisis”, the symptoms of which are as follows:

*“hypermedia development is currently at the stage software development was at thirty years ago. Most hypermedia applications are developed using an ad hoc approach. There is little understanding of development methodologies, measurement and evaluation techniques, development processes, application quality and project management ... We are potentially about to suffer a hypermedia crisis”* (Lowe & Hall, 1999 p. 14)

*“In many cases, the development approaches used for Web-based systems have been ad hoc, reminiscent of early days of application software development ... Overall, software development for the Web lacks rigour and a systematic approach”* (Murugesan & Deshpande, 1999)

*“...much of the development is carried out without a true understanding of analysis and design issues. Currently, the problems of developing web-based systems are similar to those in traditional software engineering thirty years ago”* (Hadjerrouit, 2001)

*“... [hypermedia applications development] is usually quick&dirty, resulting in low correctness, robustness, and maintainability of the end products”* (Pauen et al., 1998)

*“Most WWW developers delve directly into the implementation phase, paying little or no attention to requirements acquisition and specification and going through a very informal design phase (if any)”* (Coda et al., 1998)

*“Most of the web sites are created opportunistically without prior planning or analysis. Moreover, even large mission-critical intranet projects are being started without any regard for methodology.”* (De Troyer, 2001)

It is difficult to accept these harshly generalised assertions given that there is not much firm objective data to support them. So far, there has been very little rigorous wide-scale empirical research into Web or hypermedia systems development; to the author’s knowledge, only four survey-based studies have been published in the mainstream literature (Vora, 1998; Russo & Graham, 1999; Barry & Lang, 2001; Lowe & Eklund, 2002).

This paper aims to contribute to a better understanding of the realities of hypermedia systems development in practice by reporting on the findings of a survey of 438 organisations. For the purposes of this study, hypermedia is defined as “any interactive software system that permits a user to navigate through hyperlinked information by means of various user-selected paths”. This includes such applications as interactive Web sites, electronic catalogues, intranets, courseware / CBT, interactive e-commerce systems, portals, and online information services.

## 2. RESEARCH OBJECTIVES AND METHOD

The objectives of this study were to explore:

1. the extent to which problems that characterise the alleged “crisis” in hypermedia systems development exist in practice;
2. which, if any, mechanisms are used to guide and control the practice of hypermedia systems development.

A survey was conducted in Ireland, by post and on the Web, in the last quarter of 2002. Appropriate authentication mechanisms were engaged to assure instrumental rigour and validity. As is often the case with organisational surveys, the definition of an accurate sampling frame was difficult. Here, the population includes companies engaged in general bespoke systems development; those specialising in Web, multimedia, or hypermedia systems development; those from traditional media that have branched into “new media”; and those that have internal IS departments. The initial sample was compiled from a number of classified industry databases. It was then systematically reduced, based on descriptions of activities and portfolios of work as described on Web sites and in secondary data sources. As an additional sifting mechanism, a number of questions were introduced into the questionnaire to ensure that only those respondents that developed hypermedia systems of non-trivial complexity, as defined by a combination of check variables, were included in data analysis. Prior to distribution, the survey was pilot tested with a purposefully selected group of mixed experience from mixed professional backgrounds, using the “talk aloud protocol” advocated by Dillman (2000). Professional technical writers assisted with the wording and visual layout of the questionnaire.

The final population consisted of 417 organisations, but a few of these had multiple divisions that were separately included, giving an overall tally of 438. It was decided to sample the entire population. In 425 cases the names of individuals in designated positions were known. For the remaining 13, the questionnaire was addressed to the “Head of Software Development”. The cover letter requested that the questionnaire be completed by someone in a design role, such as software design, information architecture, or creative design.

Reminders were sent out by post after 4 weeks, and again by e-mail after 6 weeks. A total of 214 valid responses were received, - 112 by postal mail, 88 via the Web, 12 by e-mail, and 2 by other media. One organisation responded twice, which was taken into consideration for those questions where the unit of analysis is the organisation rather than the individual. In addition, 23 questionnaires were returned undelivered or with a note that the organisation had ceased operations. This gives an overall organisational response rate of 51.3% (213 / 415). However, 42 of these responding organisations indicated that they had no significant experience of hypermedia systems design. Another 5 responses were insufficiently complete. Thus the usable response rate is 44.5% (166 / 373) based on the size of the true population.

### **3. SYNOPSIS OF FINDINGS**

#### **Project Management and Requirements Planning**

Respondents were asked to indicate the actual and planned time and costs of their most recently delivered project “of non-trivial complexity”. To avoid speculative responses, a “Don’t Know” category was provided. It was found that 62.8% of projects are delivered in 16 weeks or less, with a median delivery time of 10.5 weeks, which is consistent with findings in other studies (Barry & Lang, 2001; Vora, 1998).

At first glance, this lends support to the notion of “Web time”, a development context that is supposedly characterised by “headlong desperation and virtually impossible deadlines” (Constantine & Lockwood, 2002) which leads to “frantic application development” (Yourdon, 1996). One would therefore expect developers to resort to shortcuts and other time-saving devices that may not be as considered as ought be. Unconsidered actions are inherently risky and likely to cause problems over time, but when respondents were presented with a list of problems that typify the “software crisis”, it appears that few major problems are being experienced in practice (see Table 1). The three most significant troubling aspects are, in order of descending gravity: controlling project scope / feature creep; coping with requirements volatility; and preparing accurate time and cost estimates. Coping with “Web time” delivery schedules and controlling costs do not seem to cause great difficulties, as further testified by the findings in Table 2. 65.9% of projects are delivered within the agreed budget, and 32.2% are delivered on time, whereas time and cost over-runs of more than 50% arise in only 16.7% and 2.6% of cases respectively.

In response to another question, it was found that the level of guideline usage for Project Planning & Estimation and Requirements Documentation is 62.9% and 63.5% respectively. Another interesting and somewhat surprising finding is that 86.7% of respondents indicated that for their most recently completed project of non-trivial complexity, there was a written requirements specification document. These specifications are often quite detailed, with 52.4% of them being 25 or more pages in length (see Table 3).

None of these findings support the view that project management and requirements analysis is “sloppy”, “opportunistic” or in a state of crisis.

Table 1. Extent of problems in hypermedia systems design.

	N	No problems	Minor problems	Moderately problematic	Major problems
Controlling project scope / Feature creep	161	1.2%	39.1%	42.9%	16.8%
Coping with volatile and changing requirements	164	1.8%	38.4%	46.3%	13.4%
Preparing accurate time and cost estimates	156	3.8%	43.6%	45.5%	7.1%
Lack of adequate design documentation	157	22.3%	47.1%	24.8%	5.7%
Coping with accelerated timescales of Web environment	140	13.6%	55.7%	26.4%	4.3%
Lack of guidance in the use of design methods and techniques	153	30.1%	53.6%	13.1%	3.3%
Controlling and coordinating project tasks	164	11.6%	64.6%	20.7%	3.0%
Induction and training of staff in the use of design methods and techniques	146	18.5%	50.0%	29.5%	2.1%
Managing communication between team members from different professional backgrounds	166	14.5%	62.0%	21.7%	1.8%

Table 2. Variance in project duration and costs [actual / planned]

	Variance in project duration N = 137	Variance in project costs N = 76
Between 50% and 25.01% UNDER	1.5%	5.3%
Between 25% and 10.01% UNDER	1.5%	5.3%
Between 10% and 0.01% UNDER	0.0%	7.9%
Exactly ON TARGET	29.2%	47.4%
Between 0.01% and 10% OVER	5.1%	7.9%
Between 10.01% and 25% OVER	24.1%	14.5%
Between 25.01% and 50% OVER	21.9%	9.2%
Between 50.01% and 100% OVER	10.9%	0.0%
Between 100.01% and 200% OVER	5.1%	1.3%
More than 200% OVER	0.7%	1.3%

Table 3. Size of the written requirements specification (N = 124).

Less than 10 pages	18.5%
10 to 24 pages	29.0%
25 to 49 pages	20.2%
50 to 99 pages	16.1%
100 pages or more	16.1%

## Use of Processes, Methods, Approaches, Procedures and Guidelines

Terms such as “process”, “method” and “approach” are difficult to neatly define, so caution must be exercised in analysing the findings of this survey because the possibility of various interpretations may give rise to some element of measurement error. The terminology used in this paper is consistent with that in Wynekoop & Russo (1995). Nuances of nomenclature may in part explain why previous research on the use of methods and approaches in hypermedia or multimedia systems development is inconsistent. Whereas Britton et al (1997) found that “the ‘big bang’ approach to system development is rare”, Russo & Graham (1999) reported that none of their respondents used a formal system development method.

This study suggests that hypermedia systems development is much more disciplined than commonly believed. In reply to a closed multiple-choice question, 83.6% of respondents indicated that their organisation uses a hypermedia development process that has clear tasks and/or phases within it. In slightly more than a half of these organisations, these processes are explicitly documented (see Table 4). Only 16.4% of organisations do not have a clear process, 59.1% of whom consider this a problem.

Table 4. Organisation's hypermedia development process (N = 165).

There is no clear process	16.4%
Clear tasks and/or phases, though the process used is not explicitly documented	41.8%
Clear tasks and/or phases, according to an explicitly documented process	41.8%

A less clear picture emerged in response to an open-ended question that asked respondents to “list the names of any hypermedia development methods or approaches that you have used”. This question gave rise to some confusion and misunderstanding, with only 79 usable responses out of a total of 167 questionnaires returned. Whereas previous studies revealed the prevalence of in-house methods for Web and multimedia systems development (Barry & Lang, 2001; 2003), it becomes apparent from this study that these are mostly not methods in the true sense. Rather, they are an eclectic mix of approaches, process models and toolkits of techniques drawn from right across the board. Because many of the responses received were ambiguous, it was difficult to code them accurately and the categories overlap (see Table 5). Quite a few of the responses which indicated that an internal method or approach was used did not provide any details on its orientation, so caution should be taken in interpreting the table as the percentages in some categories may be understated.

There is no great surprise in the top three response categories: traditional software development methods or variants of them; in-house methods; and rapid / agile methods. There is also substantial incidence of development approaches that are focused around the use of tools, - a finding that lends some support to the assertion that developers “delve directly into the implementation phase” (Coda et al., 1998). However, there is widespread acceptance of the necessity for explicitly documented plans and considered action in preference to “ad hoc” just-do-it approaches (see Table 6). 93.9% of respondents agreed that there is an essential need for planning, and 79.3% agreed that plans and working methods should be clearly documented. 68.3% of organisations use documented guidelines or procedures for some or other purpose (Table 7).

Table 5. Use of methods and approaches in hypermedia systems design. (Usable N = 79)

Hybrid, customised, or proprietary in-house method or approach	24.1%
Traditional “legacy” software development methods and approaches, or variants thereof e.g. <i>SSADM, Yourdon, JSP, SDLC / Waterfall</i>	21.5%
Rapid or agile development methods and approaches e.g. <i>RAD, Extreme Programming</i>	17.7%
Approaches that are focused around the use of tools and development environments e.g. <i>PHP, Java, Flash, ASP, J2EE</i>	15.2%
Object-oriented development methods and approaches e.g. <i>RUP, OOA&amp;D</i>	11.4%
Approaches that are focused around the use of techniques e.g. <i>Storyboards, Flowcharts, Wireframes, UML</i>	7.6%
No method used / development approach is “ad hoc”	7.6%
Incremental or evolutionary methods and approaches e.g. <i>Spiral Model, Staged Delivery, Iterative Design, Code &amp; Fix</i>	7.6%
HCI / Human Factors Engineering methods e.g. <i>User Centred Design, Interaction Design, Goal-based Requirements</i>	6.3%
Specialised non-proprietary methods for Web and hypermedia systems development e.g. <i>Fusebox, WSDM, OOHD</i>	5.1%

Table 6. Attitudes to planning in hypermedia systems development.

	No opinion	Valid N	Firmly disagree	Disagree	Neutral	Agree	Firmly agree
Ad hoc “improvised” hypermedia development approaches <u>generally</u> result in systems of poor quality	13	153	5.2%	18.3%	7.8%	39.2%	29.4%
To combat system complexity and time pressures, there is <u>an essential need</u> for planning and considered action	2	165	0.0%	1.2%	4.8%	31.5%	62.4%
To ensure efficient and effective collaboration within the development team, plans and working methods should be <u>explicitly</u> documented	1	165	1.8%	5.5%	13.3%	36.4%	42.5%

(“No opinion” was an explicit response category, as was “Neutral”, both quite distinct from blank missing responses)

Table 7. Use of documented procedures and guidelines.

Requirements documentation	63.5%	System testing & debugging	39.5%
Project planning & estimation	62.9%	Coding practices	34.7%
Interface design / Usability	50.9%	Use of diagramming techniques	24.0%
Technical design documentation	49.7%		

## 4. CONCLUSIONS

On the basis of the findings reported herein, there seems to be little evidence to believe that hypermedia systems development is in a “crisis”; on the contrary, developers seem to be competently dealing with the pressures upon them. Although requirements management issues pose some problems, - which could be said to be an inevitable reality for all software developers, - projects are generally under control, with few major time or cost over-runs. Most organisations have some clearly visible “process”, and although that process is often undocumented, the value of documented plans and working methods is well accepted. Systems are being developed mainly using traditional software development methods or variants, in-house development methods, or rapid / agile approaches. Though there are many hypermedia-specific methodologies set forth in the academic literature (e.g. *RMM*, *OOHDM*, *WSDM*, *W3DT*), the findings of this survey reveal that only 2 of 94 respondents have ever used any of these and just 5 others are otherwise aware of them. In view of this, it must be asked if not the real “crisis” is within academia? Wynekoop & Russo (1995) have warned that “by failing to evaluate current methodologies, practices and needs, researchers may develop methodologies that are not only irrelevant, but flawed”. The academic literature is already strewn with hundreds of development methods, many of which are arcane, impractical, and unworkable. With the emergence of Web and hypermedia systems there has been talk of a “pressing need for new methods and tools” (Murugesan et al., 1999). It is doubtful if there is a genuine need for *new* methods. As this study reveals, traditional software development methods can be readily adapted to the new challenges of hypermedia systems development. Across time, the state of practice has often led the way and informed theory, rather than vice versa (Glass, 1989). Bearing this observation in mind, if academic researchers wish to make useful contributions to hypermedia development practice, perhaps the best place to start is by learning from practice through grounded empirical research.

## 5. REFERENCES

- Barry, C. & Lang, M. (2001) A Survey of Multimedia and Web Development Techniques and Methodology Usage. *IEEE Multimedia*. 8(3), 52-60.
- Barry, C. & Lang, M. (2003) A Comparison of "Traditional" and Multimedia Information Systems Development Practices. *Information & Software Technology*. 45(4), 217-227.
- Britton, C., Jones, S., Myers, M. & Sharif, M. (1997) A survey of current practice in the development of multimedia systems. *Information and Software Technology*. 39, 695-705.
- Coda, F., Ghezzi, C., Vigna, G. & Garzotto, F. (1998) Towards a Software Engineering Approach to Web Site Development. In *Proceedings of the 9th International Workshop on Software Specification & Design*, Ise-Shima (Isobe), Japan, April 16-18, 1998, pp. 8-17. IEEE Computer Society.
- Constantine, L. L. & Lockwood, L. A. D. (2002) Usage-Centered Engineering for Web Applications. *IEEE Software*. 19(2), 42-50.
- De Troyer, O. (2001) Audience-Driven Web Design. In Rossi, M. & Siau, K. (eds), *Information Modeling in the New Millennium*, pp. 442-461. Idea Group Publishing, Hershey, PA; London.
- Dillman, D. A. (2000) *Mail and Internet Surveys: The Tailored Design Method*, 2nd Edition. New York: John Wiley.
- Glass, R. L. (1989) The Temporal Relationship Between Theory and Practice. *Journal of Systems and Software*. 10, 65-67.
- Hadjerrouit, S. (2001) Web-based Application Development: A Software Engineering Approach. *SIGCSE Bulletin*. 33(2), 31-34.
- Lowe, D. & Hall, W. (1999) *Hypermedia & the Web: An Engineering Approach*. Chichester: Wiley.
- Lowe, D. B. & Eklund, J. (2002) Client Needs and the Design Process in Web Projects. *Journal of Web Engineering*. 1(1), 23-36.
- Murugesan, S. & Deshpande, Y. (1999) Preface to ICSE'99 Workshop on Web Engineering. In *Proceedings of International Conference on Software Engineering (ICSE'99)*, Los Angeles, California, USA, pp. 693-694.
- Murugesan, S., Deshpande, Y., Hansen, S. & Ginige, A. (1999) Web Engineering: A New Discipline for Development of Web-based Systems. In *Proceedings of First ICSE Workshop on Web Engineering*, Los Angeles, California, USA, May, pp. 1-9. ACM.



- Pauen, P., Voss, J. & Six, H.-W. (1998) Modeling Hypermedia Applications with HyDev. In Sutcliffe, A. et al. (eds), *Proceedings of IFIP 13.2 Working Conference / Designing Effective and Usable Multimedia Systems*, Stuttgart, Germany, September 1998, pp. 23-40. Kluwer Academic Publishers.
- Russo, N. L. & Graham, B. R. (1999) A First Step in Developing a Web Application Design Methodology: Understanding the Environment. In Wood-Harper, A. T. et al. (eds), *Methodologies for Developing and Managing Emerging Technology Based Information Systems: Proceedings of 6th International BCS Information Systems Methodologies Conference*, London, pp. 24-33. Springer.
- Vora, P. R. (1998) Designing for the Web: A Survey. *ACM interactions*. 5(3) May-June, 13-30.
- Wynekoop, J. L. & Russo, N. L. (1995) Systems development methodologies: unanswered questions. *Journal of Information Technology*. 10(2) June, 65-73.
- Yourdon, E. (1996) Developing Applications for the Internet: Advice for the Java Generation. *American Programmer*. December, 36-41.