# Persistent Problems in Information Systems Development. The Case of the World Wide Web

Karlheinz Kautz[a], Jacob Nørbjerg[b]

Department of Informatics, Copenhagen Business School
Howitzvej 60, DK-2000 Frederiksberg, Denmark
Phone: +45 3815 2400, Fax: +45 3815 2401
karl.kautz@cbs.dk[a], jacob@cbs.dk[b]

**Abstract**

*New technologies frequently move the focus of Information Systems (IS) research away from the study of the persistent and fundamental predicaments of IS development and use. In this paper World Wide Web technologies and web-based systems are used as an example of this problem. Many researchers see these technologies and systems as so different from traditional information technologies and systems that they require new approaches to systems development. We critically examine these claims in the light of the historical evolution of the practice and research of information systems and show that characteristics frequently associated with web-based systems, such as changing skill needs and more complex organization of development projects as well as a broader and more diverse user population aren't radically new developments, but extensions and variations of fundamental dynamics which have characterized IS since the inception of the discipline. We thus argue that current research ought to look at past experiences for advice on how to cope with present problems. And we urge IS researchers to be concerned with the underlying, persistent, problems in information systems development instead of the surface problems manifest in the fad or technology 'of the day'.*

**Keywords**

web based information systems, systems development methods, systems development practice, history of computing

# Introduction

The information systems (IS) discipline frequently experiences new and fast changing technologies, an evolving assembly of development tools and techniques, and a rapid diffusion of computer based applications and systems into more and more parts of the professional and social lives of humans.

IS research naturally attempts to stay as close as possible to the cutting edge of technology and researchers are therefore continuously looking for new phenomena to study or technologies and applications to develop. While this is certainly a beneficial attitude, if for no other reason because it forces the research community to be aware of the developments in the surrounding world, we consider it problematic to concentrate only on 'the issue of the day', while disregarding the persistent and fundamental concerns that have faced the field from its inception.

The current interest in the rapidly diffusing web-technologies and systems can serve as an example of this point. Many researchers see these technologies and the systems they help to build as so different from traditional information technology and systems that they require new concepts for and approaches to information systems development and implementation (Alter, Ein-Dor, Markus, Scott & Vessey 2001, Bansler, Damsgaard, Scheepers, Havn & Thommesen 2000, Baskerville & Pries-Heje 2002, Carstensen & Vogelsang 2001, Holck 2002, Lyytinen, Rose & Welke 1998, Murugesan, Deshpande, Hansen & Ginige 2001, Pressman 1998).

Some of this research looks at technical aspects, such as sound design principles for web-based systems, or the specific human computer interaction problems raised by the technology itself; e.g.. limitations of the hyper-text mark-up language (HTML) and protocol (HTTP), or secure and robust systems architectures (Pressman 1998, Pressman 2000). Other research takes a broader view and looks at the development methods and the organizational and management approaches needed to develop this 'new class' of applications and systems (Baskerville et al. 2002, Carstensen et al. 2001, Pressman 1998).

In this paper we critically examine the claims about the unique nature of web technologies and systems. Our argumentation is primarily historic; i.e. we will show how, what is claimed to be new or distinguishing with regard to web-based systems development are really variations of well known practices and methods in IS development. Likewise are the challenges facing contemporary systems development variations of the problems that have faced IS development for almost half a century. Thus we conclude that systems development in general in the age of the World Wide Web is not fundamentally different from systems development in the recent or further past and we see no evidence for an emerging new 'paradigm' for IS development (Baskerville et. al. 2002), nor a need for new concepts or theories in IS research (see alsoAlter et al. 2001). In fact, web-development research *and* practice might actually benefit from a study of the history of IS development – both as a source of inspiration and as a way to overcome well-known challenges and avoid equally well-known pitfalls. The paper therefore also serves to identify and direct attention to the underlying and persistent problems in IS development with which researchers ought to be far more concerned than with the surface problems manifest in the fad or technology 'of the day'.

We will focus our analysis on two broad themes since space does not allow a treatment of all the issues raised in the literature on web development and use. First we discuss the nature of web-based systems and the context of development of these systems (section 2), and then we take a look on the development practices observed in web based systems development and discuss whether there is a need for new methods for this type of development (section 3). Our review of the literature on web-based IS reveals a set of *persistent issues* in the fields of IS development, which we then discuss in more detail (section 4), before we conclude the paper (section 5).

# The Context and Conditions of Web Development

Research contributions point to several specific characteristics of web based systems development; e.g. the nature and diversity of skills required, the role of the users in development, the size and diversity of the user population, the close relations between web based systems and marketing, and the rate of change of the underlying technologies. We've chosen to structure our discusison of these issues under the following headings:

- the need for new skills and a more complex project organization
- the widening and diversifying user population

## New skills and organization

Web sites are often used as an organization's 'public presence' on the World Wide Web and web based systems therefore contain more than just the functionality normally associated with information systems; e.g. order management, querying, monitoring etc. They become part of the way a company presents itself and markets its products to the rest of the world. Thus, new kinds of requirements, related to the 'look and feel' and marketing capability of a site are added to the requirements to web-based systems. And new design and implementation activities are included in the life cycle to design and produce a pleasant and consistent interface able to attract and keep the attention of user. These interfaces often include advanced graphics and multimedia features such as sound, animation and video streaming (Bansler et al. 2000, Carstensen et al. 2001, Holck 2002, Murugesan et al. 2001, Vidgen 2002).

Furthermore, as several authors observe, web development projects include content production as well as content design; i.e a web development project not only produces technical frameworks and facilities for information storage and presentation, it produces the information as well (Murugesan et al. 2001). In this sense web development projects seem to transcend the traditional boundary between on the one hand producing software with some sort of data processing and storage capability, and on the other using the software to enter, process and retrieve data. The result, according to some, is that the boundary between designers and users of web based systems becomes blurred (Bansler et al. 2000).

To meet the new types of demands, web development projects depend on an increasing and increasingly varied set of skills beside the technical skills needed in 'traditional' information systems development. Web development projects are staffed with new kinds of systems developers recruited from areas such as marketing, graphics design, video- and film production, not to mention the end-users and the specialists needed to produce and communicate content within various domains; e.g. health, law, etc. (Greenbaum & Stuedahl 2000, Hansen, Deshpande & Murugesan 2001, Vidgen 2002). Often these new systems developers lack basic IT skills (Carstensen et al. 2001). At the same time, new specializations and job descriptions emerge to meet some of the special requirements of web systems development, e.g. information architect, multimedia designer, and multimedia project manager. An organization developing and using corporate intranets for information storage and retrieval furthermore face issues and conflicts related to responsibilities for information ownership, production, and maintenance (Bansler et al. 2000, Deshpande & Ginige 2001).

All this leads to the inclusion of more tasks and specializations in the development life cycle and an increasingly complex organization of web development projects. It also means that user representives are included as active participants in the development project, sometimes using advanced tools to independently produce complete or substantial parts of a web based system or application. Project organization and management furthermore tend to be unstable as IT organizations struggle to define the job categories, training programs, and development and management practices required to meet the needs of web development (Greenbaum et al. 2000).

We do not question these observations regarding skill needs, the role of the user or the difficulties involved in organizing and managing web development projects. We do,

however, argue that what is observed are evolutions or continuations of trends and changes that have taken place in information systems development over the last 50 years and not signs of a significantly different situation.

In his account of the history of information systems development Andrew Friedman illustrates how information systems have penetrated deeper and deeper into the world of users (Friedman 1989). The use of information systems has expanded from the first scientific applications built by the persons needing them, over the very early business applications to automate distinct and routine clerical processes, to present applications and systems that are deeply integrated into almost all aspects of the working – and often social – lives of humans. This development has been accompanied by a continuous expansion of the core issues and concerns of IS development away from the computing machines towards the world of the users, ongoing and interdependent changes in the internal organization of IS projects and departments, and changes in the relations between users and systems developers.

First, the main issues and concerns of ISD have gradually moved away from what Friedman calls the computer systems core: In the early days of computing, the primary concern was one of efficient utilization of the very costly, complex, and limited hardware resource. Decreasing hardware costs, more powerful development tools; e.g.. operating systems, compilers etc, and the increasing diversity and importance of information systems, shifted concerns away from the computing machines towards user related issues such as business analysis, organizational implementation and impact of information systems, and – later – user interface and ergonomic issues.

Second, these changes have been accompanied by a need for more and more skills in information systems development and an increasing specialization. The original project manager/analyst/programmer/tester/maintainer who was primarily concerned with the computing machine, has been replaced by a plethora of specializations that reflect the increasing complexity of systems development as well as the inclusion of new tasks and concerns in the development life cycle. Systems development organizations have therefore always had to recruit staff from ‚outside', to answer the need for new skills in systems development. Often the people thus recruited into information systems development, did not have basic IS development skills.

Third, systems development organizations have always had to define and redefine project organization, systems development tasks, job categories, divisions of labor, development practices and management principles in systems development in face of the changes described above (cf. Greenbaum et al. 2000, Kraft 1976). This is not, and never has been, an easy task as can be seen in for example Pettigrew's study of an IS department in the 1960s (Pettigrew 1973), or in the software engineering literature in general (cf.Naur & Randell 1969, Swanson & Beath 1990, van Genuchten 1991).

Finally, there are ongoing and dynamic redefinitions of the boundary between the responibilities and tasks of (end) users and those of developers. The general tendency has been to include more and more user related tasks under the umbrella of information systems development, but tasks are moved in the opposite direction as well. The task of data entry, for example, originally the responsibility of data processing's punch card department, has been completely taken over by the end user sitting at his or her PC, or who is using advanced equipment for data capture. Other tasks, such as systems maintenance, program installation, data querying, report generation and simple application development seem to be moving back and forth between user and systems department depending on company policies, application size and complexity, as well as user proficiency in PC-based systems and tools such as; e.g.,

Windows, Excel, Access etc. Also, end users may be temporarily included in information systems development projects as user representatives, interface testers etc.

Our claim is, that what we see unfold in web information systems development is a continuation of the trends and developments described above. Web based systems represent the latest expansion of information systems into the user domain. And this creates - as has always been the case - new issues and concerns for IS development, new tasks in the life-cycle, a need for new specialisms and skills in information systems development, new challenges for organization and management of development projects, as well as a need to reconsider user roles and the boundary between systems use and development.

This is not a radically new situation but the latest example of how information systems development needs to define new configurations of specialists, tasks and users in systems development in the light of a new application domain and technical possibilities. The challenge is, as always, to identify the special requirements and problems of the situation, and to learn from the past how similar problems and challenges have been dealt with before.

# A widening user population

Web applications on the public part of the Internet can be used by potentially everyone with access to the Internet and a browser. Internal (corporate) systems are likewise accessible from all computers on the corporate intranet. This means that web development projects must take the needs and capabilities of a very large, diversified, and for the most part unknown user group into account when designing a system and particularly its user interface. Several authors note the size and diversity of the user base as one of the distinctive characteristics of web based systems and applications (Deshpande et al. 2001, Murugesan et al. 2001).

We agree that a large and diversified user base does create problems for a systems development project in terms of identifying, describing and managing requirements as well as designing appropriate user interfaces. However, this is not a new situation, neither is it particularly associated with web-based systems. The user base has been expanding since the days of the first information systems as discussed in the previous section. And the systems have pentrated deeper and deeper into the world of users.

Several studies in the past report about the difficulties involved in developing information systems for a large and diversified user base, including the problems of managing vague and unstable requirements (c.f. Curtis, Krasner & Iscoe 1988, Pape & Thoresen 1985). As Grudin notes, however, there is a difference between systems developed for a known and clearly delimited set of users; i.e. systems developed in-house or on a contract, and systems developed as products for a market; i.e. word processors etc. (Grudin 1991). An important shift in this respect occurred with the large-scale diffusion of PCs and PC-based products in the 1980s. This triggered a general interest in usability issues and usability design and testing techniques, which are now in general use in organizations producing IT-based products for a market.

Web based applications may differ from PC-based products with respect to pricing and accessibility; i.e. the former are free and you can leave them with a click of the mouse. For the designers of web-based systems this does of course mean that they must consider issues such as how to attract and keep users on a site. The fundamental question of how to define and manage requirements and test a system intended for a large and partly unknown user population is, however, independent of whether the system in question is a ‘traditional’ information system, a product for a market, or a web-based application.

# New practices

Researchers in systems development generally agree that web based systems are produced in an ad hoc manner without attention to development methodologies, systematic planning and management practices, quality assurance, and process and product measurements (Avison & Fitzgerald 2002, Murugesan et al. 2001). A study of web-development projects in the USA and Denmark observes for example high time pressure created by a desperate rush to market, use of prototyping and parallel development, and a flexible approach to quality assurance (Baskerville et al. 2002). According to this and similar studies, this means that traditional systems development methods and management techniques are unfit for the development of web based applications. Therefore there is a need for new methods and tools for web development and web engineering (cf. Baskerville et al. 2002, Murugesan et al. 2001).

Again, we do not question the results from the studies of web systems development and we agree that many so-called traditional systems development may be unfit for the development of web-baseed systems. We do not, however, agree that this implies a need for a special 'web based' approach to systems development.

First, the problems and practices of web-development are not different from what has been consistently reported from studies of IS development for more than 30 years. Avison & Fitzgerald, for example, see clear parallels between current practices in web-development and the way information systems were developed in the 70s (Avison et al. 2002). Recent capability maturity surveys have also placed the majority of software development organizations on the lowest maturity level, indicating practices similar to those reported from web development projects (Herbsleb, Zubrow, Goldenson, Hayes & Paulk 1997). Problems with unrealistic and hard deadlines have also been repeatedly reported in the past (DeMarco & Lister 1987, Greenbaum et al. 2000). These observations, therefore, point to general problems with software development project planning and management.

Second, the practices observed in current web development projects may deviate from the *methods* recommended in textbooks, but that has also been consistently reported in several studies of systems development in the past. Systems development methods, especially those that presume a sequential progression of a predefined set of tasks, are as a rule not used in practice. Developers have been generally reported not to follow the guidelines and recommenmdations of a systems development method, but to combine elements of different methods and tools based on their own experience instead (Bansler & Bødker 1993, Fitzgerald 1996). These findings are confirmed by newer research into the use of modern, more flexible, methods as well (Kautz & Madsen 2002). Some researchers even suggest to abandon the idea of planned methodical systems development and talk about emergent and amethodical systems development instead, (Truex, Baskerville & Tavis 2000, Truex, Baskerville & Klein 1999). This, in our view, indicates a general; i.e. not web related, problem with the whole idea of systems development methods and their use in practice.

Third, researchers of web-based development note that sequential approaches to systems development, like the waterfall model, are inappropriate for the highly dynamic environment, short development cycles, and unstable requirements typical of web based systems development. Instead these researchers recommend to use alternative approaches based on evolutionary or incremental software process models (Baskerville et al. 2002). We totally agree, in fact several researchers and practitioners question the appropriateness of sequential software process models within many development contexts. But the alternatives are well described and have been known for years (c.f.Boehm 1988, Floyd 1987, Floyd, Reisin & Schmidt 1989), and have proven their usability in a variety of settings (Korsaa, Vinter & al.

2001, Nørbjerg 2002). Thus, the problem seems not to be the need of evolutionary and incremental approaches to web-development but to disseminate knowledge about how to use these approaches in a planned, manageable and consistent way (Nørbjerg 2002).

Finally, and on a more general note, we must be careful not to equate *observed* practice in web-based systems development with *good* practice. It appears that organizations that develop web-based applications and systems have been able to cope with a highly pressurized market by accepting hard deadlines, short development cycles and ad hoc approaches to planning, development and quality. We have already noted how these web practices resemble what has been observed within other parts of systems development, but we should also note that such practices are generally considered problematic because of the problems they create; e.g. poor planning and control, missed deadlines, unstable systems, and high pressure on developers (Paulk, Curtis & Chrissis 1993, Pressman 2000). The problem remains, however, to define what constitutes 'good practice' and how to assess it across the many different types of systems development projects, web or non-web, that exist today (Bach 1994, Iversen, Nielsen & Nørbjerg 1998, Nielsen & Nørbjerg 2001).

# Discussion

Systems development for the web requires other technologies, skills, and management practices than development of; e.g. accounting systems for a mainframe. Systems developers and organizations involved in the development of web based information systems therefore face many new and unique challenges of a both technical and organizational nature. It is also well documented that many of the approaches to systems development advocated in software engineering handbooks and systems development methodologies are unfit for web development. We hope to have demonstrated, however, that the nature of these challenges and the ways to overcome them are not fundamentally different from what have been experienced in information systems development in the past 40 years. In fact, rather than pointing to a new paradigm for systems development (Baskerville et al. 2002), current web development practices and problems highlight a set of persistent issues and questions for systems development research and practice.

## Coping with diversity

Systems development projects are becoming increasingly diversified in terms of size, application domain and underlying technology. We thus face the challenge of how to adapt organization, management, and systems development methodologies to a broad and very diversified set of conditions and environments. There are already a plethora of methodologies and plenty of textbook advice on how to organize projects, but we still lack knowledge of how to choose appropriate approaches in concrete situations (Mathiassen, Seewaldt & Stage 1995, Mathiassen & Stage 1992)

### Coping with knowledge

The rapid change in technology and the ongoing expansion of information systems and applications into new domains continuously introduce new challenges, tasks, specializations and knowledge to systems development. This results in a constant need to create and exchange knowledge in several ways: on the one hand must IT professionals learn about new application domains and development technologies; on the other, persons from other

professions who become involved in information systems development need to learn the basic skills required to develop quality software and systems. It is also necessary to continuously reconsider appropriate organizational and management principles to cope with the diverse and evolving demands for skill, people and task mix in development projects.

## Coping with discipline

Difficulties in adopting and maintaining disciplined approaches to systems development seem to be the exception rather than the rule. Over time several approaches to improvement of software development and management practices have been suggested. The fairly recent Software Process Improvement (SPI) initiatives based on software capability maturity models are the most recent and comprehensive (Paulk et al. 1993). Lessons from attempts to apply these ideas show, however, that this is very hard and time-consuming indeed (Mathiassen, Pries-Heje & Ngwenyama 2002).

The underlying and persistent question of course concerns how to build and preserve good professional practices in software development under conditions of rapid technological change, very different project types, dynamic organizations, inflow of new and inexperienced software developers, and market and other pressures.

New or modified methodologies may have a role to play, but as most research on method use shows, are few, if any, development methodologies used in full in practice. This points to a lack of understanding of the role of methodologies in teaching and training, as well as of what constitutes an appropriate development methodology under different circumstances.

## Coping with history

On a broader, more general level, there is a need for more systematic studies aiming to understand systems development in an organizational context. A few such studies have been published in the past (Friedman 1989, Kraft 1976), but there is a lack of contemporary research that study; e.g. organizational and managerial principles and practices, divisions of labour in systems development, the influence of market conditions, project size, application domain, development technology, and, not the least, how and why all these change over time (Bansler et al. 1993).

Such studies may help us better understand the patterns of variations in information systems development practices and problems - as well as how to deal with them.

# Conclusion

Information systems development is not a stable discipline. On the contrary, changes are frequent as information systems development must cope with rapidly changing and diversifying technologies, application domains, and organizational contexts. A closer examination reveals, however, several underlying and persistent challenges that have been with information systems development practice and research for years. The challenges facing systems development for the World Wide Web are contemporary variations of these underlying problems, and the practices observed in studies of web development projects represent current – more or less appropriate - ways to deal with them. Web based systems do not introduce fundamentally new problems, practices and paradigms for information systems

and software development, but repeat – sometimes in new disguises – and highlights persistent issues and problems in information systems development.

Thus, the study of the history of information systems development may help us better understand current practices and perhaps also point to ways to deal with the problems we face today. This may not be a very exciting approach for cutting edge research, but it is probably much cheaper than repeating past efforts.

# References

Alter, S., Ein-Dor, P., Markus, L. M., Scott, J. & Vessey, I. (2001), 'Does the Trend toward E-business Call for Changes in the Fundamental Concepts of Information Systems? A Debate', *Communications of the AIS*, vol. 5(Article 10), pp. 1-59.

Avison, D. & Fitzgerald, G. (2002), 'Reflections on Information Systems Development 1988-2002', *Proceedings of the 11th Information Systems Development Conference (ISD2002)*, Riga, pp**.** unpag.

Bach, J. (1994), 'The Immaturity of the CMM', *American Programmer*, vol. 7(9), pp. 13-18.

Bansler, J. P. & Bødker, K. (1993), 'A Reappraisal of Structured Analysis: Design in an Organizational Context', *ACM Transactions on Information Systems*, vol. 11 (2), pp. 165-193.

Bansler, J. P., Damsgaard, J., Scheepers, R., Havn, E. & Thommesen, J. (2000), 'Corporate Intranet Implementation: Managing Emergent Technologies and Organizational Practices', *Journal of the Association for Information Systems*, vol. 1(Paper 10), pp. 1-39.

Baskerville, R. & Pries-Heje, J. (2002), 'Information Systems Development @ Internet Speed: A New Paradigm in the Making!' *Proceedings of the Xth European Conference on Information Systems (ECIS 2002)*, S. Wrycza, Gdansk, University of Gdansk, vol. 1, pp**.** 282-291.

Boehm, B. W. (1988), 'A Spiral Model of Software Development and Enhancement', *IEEE Computer*, vol. 21, pp. 61-72.

Carstensen, P. & Vogelsang, L. (2001), 'Design of Web-based Information Systems – New Challenges for Systems Development?' *Proceedings of the 9th European Conference on Informaiton Systems*, Bled, pp**.** 536-547.

Curtis, B., Krasner, H. & Iscoe, N. (1988), 'A Field Study of the Software Design Process for Large Systems', *CACM*, vol. 31(11), pp. 1268-1287.

DeMarco, T. & Lister, T. (1987), *Peopleware. Productive Projects and Teams*, New York, Dorset.

Deshpande, Y. & Ginige, A. (2001), 'Corporate Web Development: From Process Infancy to Maturity - A Case Study', *WebEngineering2000*, S. Murugesan, Y. Deshpande and S. Hansen, Berlin, Springer-Verlag, pp**.** 36-47.

Fitzgerald, B. (1996), 'Formalized systems development methodologies: a critical perspective', *Information Systems Journal*, vol. 6, pp. 3-23.

Floyd, C. (1987), 'A Systematic Look at Prototyping', *Approaches to Prototyping*, R. Budde and et.al., Springer, pp**.** 1-18.

Floyd, C. F., Reisin, F.-M. & Schmidt, G. (1989), 'STEPS to Software Development with Users', *Proceedings of ESEC 1989*, Coventry, University of Warwick.

Friedman, A. L. (1989), *Computer Systems Development. History, Organization and Implementation*, Chichester, John Wiley and Sons.

Greenbaum, J. & Stuedahl, D. (2000), 'Deadlines and Work Practices in New Media Development', *Proceedings of IRIS 23*, L. Svensson,U. Snis,C. Sørensen et. al., Uddevalla, University of Trollhättan Uddevalla, pp**.** 537-546.

Grudin, J. (1991), 'Interactive Systems: Bridging the Gaps Between Developers and Users', *IEEE Computer*, vol. 24, pp. 59-69.

Hansen, S., Deshpande, Y. & Murugesan, S. (2001), 'A Skills Hierarchy for Web-Based Systems Development', *WebEngineering2000*, S. Murugesan, Y. Deshpande and S. Hansen, Berlin, Springer-Verlag, pp**.** 223-235.

Herbsleb, J., Zubrow, D., Goldenson, D., Hayes, W. & Paulk, M. C. (1997), 'Software Quality and the Capability Maturity Model', *CACM*, vol. 40(6), pp. 31-40.

Holck, J. (2002), '4 Perspectives on Web Information Systems', *Proceedings of the 25th Information Systems Research Seminar in Scandinavia.*, K. Bødker, M. Kühn Pedersen, J. Nørbjerg, J. Simonsen and M. T. Vendelø, Copenhagen.

Iversen, J. I., Nielsen, P. A. & Nørbjerg, J. (1998), 'Problem Diagnosis in Software Process Improvement', *Information Systems: Current Issues and Future Changes*, T. J. Larsen, L. Levine and J. I. DeGross, Laxenburg, IFIP, pp**.** 111-130.

Kautz, K. & Madsen, S. (2002), 'Applying Systems Development Methods in Practice - The RUP Example', *Proceedings of the 25th Information Systems Research Seminar in Scandinavia.*, K. Bødker, M. Kühn Pedersen, J. Nørbjerg, J. Simonsen and M. T. Vendelø, Copenhagen.

Korsaa, Vinter, O. & al., e. (2001), 'Iterative Software Development. A Practical View', Hørsholm, DELTA Software Engineering.

Kraft, P. (1976), *Programmers and Managers. The routinization of Computer Programming in the United States*, New York, Springer Verlag.

Lyytinen, K., Rose, G. & Welke, R. (1998), 'The Brave New World of development in the internetwork computing architecture (InterNCA): or how distributed computing platforms will change systems development', *ISJ*, vol. 8, pp. 241-251.

Mathiassen, L., Pries-Heje, J. & Ngwenyama, O., Eds. (2002), *Improving Software Organizations. From Principles to Practice*, The Agile Software Development Series, Boston, Addison-Wesley.

Mathiassen, L., Seewaldt, T. & Stage, J. (1995), 'Prototyping and Specifying: Principles and Practices of a Mixed Approach', *Scandinavian Journal of Information Systems*, vol. 7(1), pp. 55-72.

Mathiassen, L. & Stage, J. (1992), 'The Principle of Limited Reduction in Software Design', *Information Technology and People*, vol. 6(2-3), pp. 171-185.

Murugesan, S., Deshpande, Y., Hansen, S. & Ginige, A. (2001), 'Web Engineering: A New Discipline for Development of Web-Based Systems', *WebEngineering2000*, S. Murugesan, Y. Deshpande and S. Hansen, Berlin, Springer-Verlag, pp**.** 3-13.

Naur, P. & Randell, B. (1969), *Software Engineering*, Brussels, NATO Scientific Affairs Division.

Nielsen, P. A. & Nørbjerg, J. (2001), 'Assessing Software Processes: Low Maturity or Sensible Practice', *Scandinavian Journal of Information Systems*, vol. 13(1-2), pp. 23-36.

Nørbjerg, J. (2002), 'Managing Incremental Development: Combining Flexibility and Control', *Proceedings of the Xth European Conference on Information Systems (ECIS 2002)*, S. Wrycza, Gdansk, University of Gdansk, vol. 1, pp**.** 329-339.

Pape, T. C. & Thoresen, K. (1985), 'Development of Common Systems by Prototyping', *Presedings of the Working Conference on Development and Use of Computer Based Systems and Tools*, Aarhus, Aarhus University.

Paulk, M. C., Curtis, B. & Chrissis, M. B. (1993), *Capability Maturity Model for Software, v. 1.1*, Pitssburgh, PA, Software Engineering Institute, Carnegie Mellon University.

Pettigrew, A. M. (1973), 'Occupational Specialization as an Emergent Process', *Sociological Review*, vol. 21(2), pp. 0.

Pressman, R. S. (1998), 'Can Internet-Based Applications Be Engineered?' *IEEE Software*, vol. sept/oct, pp. 104-110.

Pressman, R. S. (2000), *Software Engineering. A Practitioner's approach. Eoropean Adaptation*, Maidenhead, Mc-Graw-Hill.

Swanson, E. B. & Beath, C. M. (1990), 'Departmentalization in Software Development and Maintenance', *CACM*, vol. 33 (6), pp. 658-667.

Truex, D., Baskerville, R. & Tavis, J. (2000), 'Amethodical systems development: the deferred meaning of systems development methods', *Accounting, Management and Information Technologies*, vol. 10, pp. 53-79.

Truex, D. P., Baskerville, R. & Klein, H. (1999), 'Growing Systems in Emergent Organizations', *CACM*, vol. 42 (8), pp. 117-123.

van Genuchten, M. (1991), 'Why is Software Late ? An Empirical Study of Reasons For Delay in Software Development', *IEEE Trans.on Softw.Engineering*, vol. 17 (6), pp. 582-590.

Vidgen, R. (2002), 'What's so Different about Developing Web-based Information Systems?' *Proceedings of the Xth European Conference on Information Systems (ECIS 2002)*, S. Wrycza, Gdansk, University of Gdansk, vol. 1, pp**.** 262-271.