Usability: Gaining a Competitive Edge
IFIP World Computer Congress 2002
J. Hammond, T. Gross, J. Wesson (Eds)
Published by Kluwer Academic Publishers

Usability Engineering Milestones In Complex Product Development - Experiences At Nokia Mobile Phones

Industrial Experience

Pekka Ketola
Nokia Mobile Phones, Finland

Abstract: How can usability engineering be managed in highly complex innovative product development? When usability engineering is performed in Concurrent Engineering (CE) product development, there are stages where usability engineering needs to be refocused in order to perform successfully in the changing project environment. The focusing points follow the product development milestones but are not identical with those. From usability engineering perspective those points are critical for achieving effectiveness and efficiency in the product development.

Key words: Usability engineering, concurrent engineering, industry experience

1. INTRODUCTION

Product development of information appliances is often based on fast Concurrent Engineering (CE) (Valjus, 1994) due to need for cost, time and quality efficiency. The pace is given by strong competition and business situation in the industry and markets.

The phases of sequential product design, for example the Waterfall method (Royce, 1970), are requirement analysis, specification (definition), design, implementation, integration and testing. In Concurrent Engineering there are several parallel sequential design areas, for example mechanics, hardware and software, synchronized via common milestones in order to ensure an optimal timetable and to minimize implementation risks. Milestones are places where the organization decides whether it is time to continue to next phase, and sometimes design compromises need to be done (acceptable design is preferred instead of best design).

Efficient usability engineering works in an effective and competent manner in the current development phase with little wasted effort. Efficiency can be measured by comparing realization (output) against usability engineering goals, plans and amount of work (input).

Effective usability engineering aims to produce an adequate or desired result. It can be measured by studying how usable the final product is, or by analysing field feedback. During product development the effectiveness can be estimated, for example, by comparing the performed usability work against the number of usability engineering originated design changes. Planning benefits, in general, are difficult to assess using objective measures. Perceptual measures, such as fulfilment of planning objectives as a measure of planning effectiveness, have better success (Premkumar and King, 1994).

It is important to understand the particular product development environment and to know the opportunities and limitations that different product development phases and concurrency set for usability engineering, in order to perform effective and efficient usability engineering. The objective of this study is to examine what kinds of usability improvements are possible in smart phone (Figure 1) development phases and how product development stages actuate usability engineering. This research problem has been addressed in Keinonen et al. (1996). Our approach to usability engineering is holistic, i.e. it views the product usability as an entity that is built from the user interface (hardware and software), the external interface and the service interface (Ketola and Röykkee, 2001).

Figure 1. An example of a smart phone concept (left) and a smart phone (right) (Nokia 7650).

Research reports and literature (Keinonen et al., 1996; Trenner and Bawa, 1998; Mayhew, 1999; Raddle and Young, 2001), describe the practical usability engineering problems in product development and solutions for those problems. The two basic problems are:

- usability engineering is done too late and
- lack of management support for usability engineering.

If the development organization is not well adapted to human-centred design these problems are likely to appear. In addition to these problems, in fast paced CE it is simply difficult to usability engineer all needed parallel design and engineering areas. Concurrent product development is a complex engineering environment. The complexity results from (Tianfield, 2001):

- complexity of the product's (technical) structure,
- complexity of development organization and
- complexity of user requirements (late and difficult-to-identify user requirements).

Current understanding and description of sequential product development phases does not give much support for serious usability engineering in a Concurrent Engineering project. It does not match with human-centred design. Hakiel (1997b) emphasizes the need for product engineering across disciplines rather than software engineering. This raises a practical problem in concurrent product development: If the product is complex and resources limited, what usability activities should be performed and when?

Though the usability engineering lifecycle is well defined and known, it is often difficult to apply human-centred design in concurrent product development lifecycle due to the fast development pace, complexities and because the product development is not fundamentally based on human-centred approach. However, the more complex the product is technically or conceptually, the more important it is to involve elements from human-centred design and to usability-engineer the product (Keinonen et al., 1996).

1.1 Previous work

Standards and guidelines (ISO 13407; Mayhew, 1999; Daly-Jones et al., 1999) define how human-centred design should be performed to provide usable systems. The assumption is that the product development process is based on human-centred design or the desired development mode is human-centred. Hakiel (1997) claims that though we have the knowledge how to do it, the problem is that we do not routinely do what we know.

Hakiel discusses how usability engineering can be integrated with software engineering. He notes that though key development principles and processes are the same in software and usability engineering, they apply to different domains. In software processes, the emphasis is on the quality of code, i.e. defect free code, while the emphasis in usability engineering is on user requirements. Hakiel presents two contrasting approaches to product engineering: (1) usability design deliverables are aligned with software design deliverables (upper part in Figure 2) and (2) usability design deliverables are contributing to software requirements (lower part in Figure 2). He emphasizes the distinction between design for use, which leads to the specification of an information technology artefact, and software development, which leads to the implementation of the artefact in software.

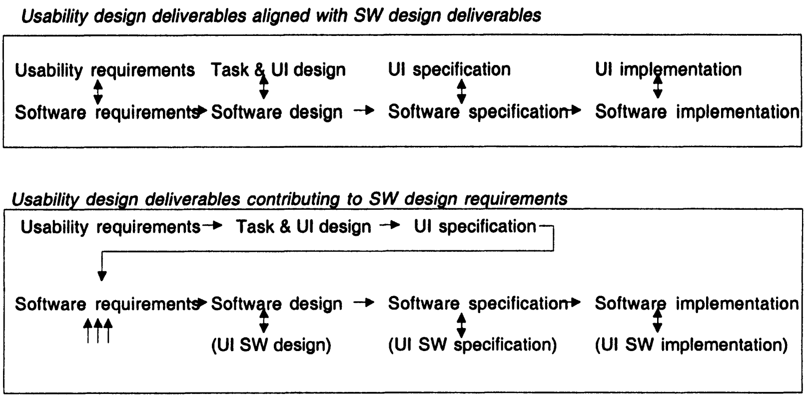

Figure 2. Contrasting usability and software engineering approaches.

It would be ideal to perform all design for use before software development. However, my observations from mobile phone development indicate that Concurrent Engineering forces UI and SW design to be aligned to concurrent development phases, especially when innovative features are designed.

Keinonen et al. (1996) studied the problem of designing increasingly complex devices. They concluded that usability can be embedded in the product development, but there must be a market-driven or intra-organizational need for change. The need for intra-organizational demand pull is also noted by Kaderbhai (1998). An industry review during 1996 showed that usability is basically a familiar concept but the essential part of usability engineering, user involvement, is still a non-utilized resource (Nieminen and Parkkinen, 1998).

While my study describes experiences at Nokia Mobile Phones, Korhonen (2000) gives an overview of usability research at Nokia Research Center. This overview gives a basic understanding of the various usability engineering activities in the research areas that precede the actual product development.

1.2 Project Approach

The findings of this study are based on several mobile phone development projects (Table 1) that were based on Concurrent Engineering and where usability was a design goal. One project served as a case project for usability engineering. In the case project, the usability work was based on a particular Usability Plan (Ketola, 2001). The results were further assessed in three other projects by doing a hands-on trial. This approach provides reliability of the results, at least, in the mobile phone development.

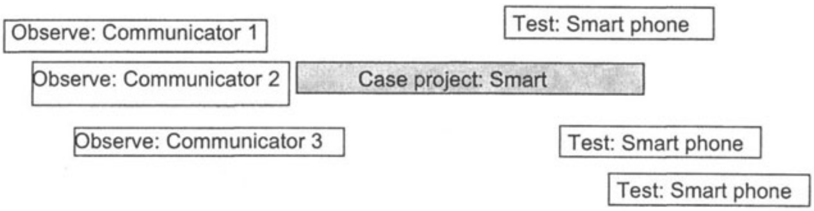

Table 1. Projects followed in the study

The research problem was to answer the following questions:

- What are the critical points for usability engineering in a concurrent product development?
- How should usability engineering be refocused in those points in order to perform effective and efficient design and usability actions?
- What is the exact difference (turning point) when moving from early to late design in Concurrent Engineering product development?

Concurrent product development sets both limitations and opportunities for usability engineering. During the projects it became clear that there are predictable turning points in product development where the usability engineering needs to be refocused. I call those usability engineering milestones. In addition, concurrency gives the perspective that there is no single early and late design phase, but one product development lifecycle contains at least two early and late phases in different engineering areas.

1.3 Description of the case project

This case study is based on one project (a smart phone product development cycle at Nokia Mobile Phones) where CE product development process is improved by integrating usability engineering activities with the process. The aim of the product development project is to create a new "leading edge" smart phone product with several novel technologies, new form factors and new interaction styles. Key functions of the phone are quality with excellent messaging and imaging capabilities, exciting and easy-to-use interfaces, compact concepts, attractive designs and entertaining. The product enables the users to create and manage personal visual content (digital pictures) in a way that enables easy picture messaging from phone to phone and from phone to any network service.

2. CONCURRENT ENGINEERING

In a CE project, several product development teams work in parallel. The aim is to provide agreed deliverables in agreed timeframes, and finally integrate the components. It must be noted that parallel design defined by Nielsen (1994) is a different concept. Nielsen handles parallel design as a method for developing and evaluating competing designs.

Figure 3 describes the basic concurrent development process model from Fujitsu (Ayoama, 1993). Concurrent Engineering is highly applicable in product development where the final product consists of multiple integrated technologies or engineering outcomes, for example integrated software, hardware and mechanics.

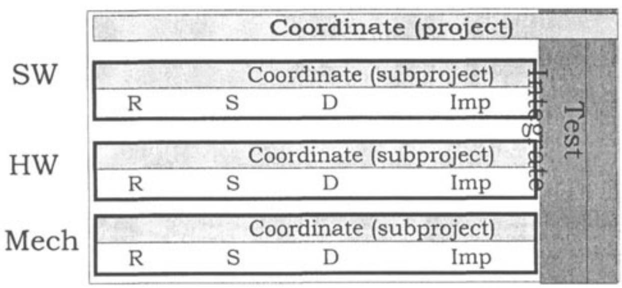

Figure 3. Concurrent development model with coordination between Requirements (R), Specification (S), Design (D) and Implementation (I).

The engineering practice inside a development team need not be sequential, but it can follow other more efficient processes. For example, incremental development may be appropriate for SW engineering, while waterfall design works better with hardware engineering. Thus, CE is more product development coordination than product implementation method.

Coordination and execution activities are performed both in the main process (project level) and sub-processes (engineering teams). The execution processes can be independent, i.e. they are not directly dependent on other sub-processes. The coordination processes are often tightly interconnected, especially in the late development phases.

2.1 Concurrent Engineering Dependencies

In mobile phone product development there are several practical dependencies between engineering areas.

- product requirements, product concept, technical product platform and industrial design define the initial development frames for all engineering areas.
- industrial design defines the overall dimensions of the product and the main factors for mechanical design.
- mechanical design defines the position and size of all product components and the available dimensions for hardware.
- hardware defines main performance issues, such as display capabilities, memory size and processor efficiency. Hardware enables certain software performance.
- software defines the software-based user interface capabilities.

The above-mentioned engineering areas have a final effect on user interface and so are potential subjects for usability engineering. From a user perspective, the product should provide quality of use. This implies that usability engineering should be done in areas that have an effect on quality of use and usability. In a mobile phone, the areas extend from software to, for example, hardware, mechanics, ergonomics, out-of-box readiness, even network services.

3. USABILITY ACTIVITIES IN PRODUCT DEVELOPMENT PHASES

Following the product development phases, the already known project milestones (M) with usability activities are:

- M0: Start of usability engineering. The milestone is typically followed (or preceded) by requirements analysis and product specification.
- M1: Start of product design. This milestone starts (or continues) early design and formative usability evaluation. Usability engineering is done with low fidelity prototypes. The units for measuring effectiveness and efficiency are the number of identified design improvements and the capability to propagate the improvements to product design.
- M2: Start of detailed design and implementation. Usability engineering is performed with high fidelity prototypes.

- M3: Start of summative usability evaluation. This milestone starts usability evaluation in order to produce summative data about the product. The measuring units for effectiveness and efficiency are the number of found usability problems and the usability of the final product.
- M4: Start of field feedback.
- M5: End of usability engineering. In most cases in the scope of this study, product usability engineering was ended before M4.

A notable observation is that the dimensions for measuring efficiency and effectiveness are different in M1 (capability to propagate the improvements to product design) and M3 (usability of the final product). The milestones (M0 to M5) and their position in product development phases are presented in Table 2.

M0		M1		M2		M3		M4		M5
	Specify product		Design		Implement and integrate		Test		Launch	

Table 2. Milestones and development phases.

Usability engineering milestones are product development stages where usability engineering needs to be refocused in order to provide efficiency and effectiveness in the changing development environment. A milestone is identified, for example, from the following characteristics:

- the project priorities or goals are defined or changed.
- a usability activity starts or ends (new usability activity starts, old activity is changed or ended).
- formal design support changes to summative design evaluation.
- target setting changes to planning, planning changes to follow-up.
- project practices for coordinating design changes is changed.
- usability engineering tool changes (for example from simulation to product prototype).

4. NEW USABILITY MILESTONES VIA HORIZONTAL AND VERTICAL REVIEW

Concurrent product development increases the number and changes the content of previously introduced usability engineering milestones. In the following the CE process is studied horizontally and vertically.

4.1 Horizontal review

By horizontal project review (lifecycle perspective) it is possible to identify the actual product development phases and their characteristics, and to estimate the potential effectiveness of usability engineering.

Figure 4 shows how many design *changes* were made in different design phases and design areas in the case project. The data was created by interviewing designers and engineering managers of the specific design areas, and by analysing the proposed and implemented design changes. The bottom line in the figure presents the situation where no changes are done.

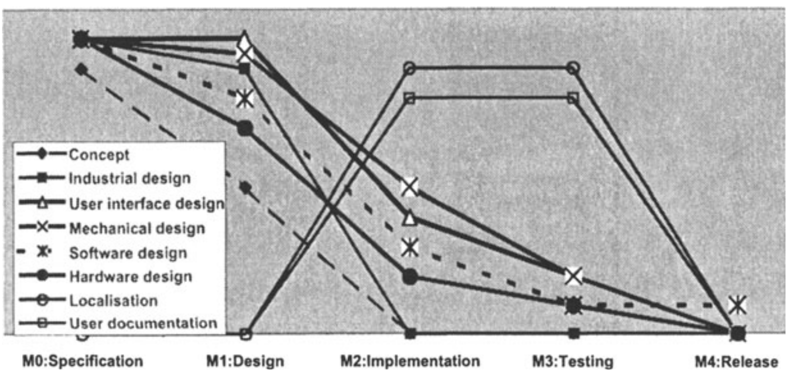

Figure 4. Doable product design changes in CE milestones

The findings based on a horizontal project view are:

1. Most product design (industrial, user interface, mechanical, software and hardware design) changes are made between M0 and M2.
2. Some changes are made even in very late design phases (software, localization, user documentation).
3. Near M2 the product concept freezes. In this milestone the development team knows exactly for the first time what the product should be like. Before concept freezing, the organization is defining the product, i.e. there are design options. After concept freezing the organization focuses on implementing the concept and there are minimal design options. The product requirements are defined much earlier (M0), but in order to design and implement new features or technologies for the first time it is necessary to leave the possibility for design changes to late design (Hakiel, 1997b). This increases design uncertainty and design teams must be alert for implementing even surprising design needs.

4. The turning point from early design (creating the design) to late design (design changes and improvements) can be estimated. The turning point can be seen at M2. In the observed projects, this was the point when the development organization started to use a design change management control mechanism.
5. Supporting engineering areas (localization, user documentation) have early and late design phases.
6. Two freezing patterns can be detected. First, the main engineering areas (software, mechanics, hardware) freeze gradually from M0 to M2. Secondly, supporting engineering areas have a main design phase between M2 and M3 with fast freezing speed. Hence, product development has two separate design phases: product design and support material design.

From this horizontal review, the following new milestones (M) can be proposed:

- M (EarlyToLate): Early design changes to late design.
- M (ConceptFreeze): Concept freezes. This gives us two important conceptual phases: pre-concept-freeze and post-concept-freeze design
- M (EarlyToLateSupportMaterial): Early design of user support material and localization changes to late design.

4.2 Vertical review

By studying a project from a vertical perspective (concurrency perspective) it is possible to identify the concurrency in product development phases, the effects of concurrency, and the potential usability engineering efficiency. The vertical findings based on Figure 4 are:

1. Development concurrency is real. For example, user interface design is concurrent with software design. User interface design is aligned with the software design instead of being a preceding action (compare Figure 2).
2. Designs freeze in a predictable order (1. concept, 2. hard design, 3. soft design, 4. Supporting designs).
3. Design changes are accepted in all design areas until M2.

4.3 Definition of early and late design phases

Product development can be divided into early and late design. The early design phase is characterized by interaction design and formative usability evaluations. Late design is characterized by detailed design and summative usability evaluations. Kiljander (1999), Nielsen (1994) and Mayhew (1999) emphasize the importance of focusing on early design phases instead of late phases due to high costs of changes in late phases. The estimate of increasing design change costs is based on findings in software engineering.

Earlier studies identify early design and late design as separate design phases. However, there is no definition of how a practitioner can identify the turning point from early to late design. Based on the observations above, I define early and late design phases for Concurrent Engineering project, and the milestone between these phases, as follows:

- **early design phase** is identified by high capability and interest of an organization to create new design, and accept and implement design changes to existing design. In the early design phase, the target of product development is to maximize the number of design improvements in the given timeframe.
- **early design phase changes to late design** when the organization decides to apply a (systematic) method for handling design change proposals.
- **late design phase** is identified by low capability and interest of an organization to create new design, and accept and implement design changes to existing design. In the late phase, the product development target is to minimize the number of design changes in order to manage project timetables and risks.

The ultimate goal of usability engineering is to support the project in achieving the project goals of cost, quality and time-to-market. The distinction and implication of early and late design phases are important: maximise design efficiency, effectiveness and quality before M1 and minimize design changes after M1.

5. MILESTONES AND SIMULATIONS

The primary usability engineering tool is a product simulation or prototype. Depending on the product development phase, different simulations and prototypes are available, low-fidelity, high-fidelity or product prototypes.

The product entity is composed from the engineering results in the late development phase (Figure 3). When a new product is developed, there is not a working product prototype before integration phase. The implication is that usability engineering on the product entity is not possible in the early phase without the aid of advanced hardware or software simulation. Meanwhile, usability engineering is possible in the non-integrated engineering areas.

In typical mobile phone development phases, the following prototypes can be produced:

- Early design: low-fidelity (and high-fidelity) prototype, design mock-up, mechanics simulation
- Late design: (low-fidelity and) high-fidelity prototype, hardware prototypes, mechanics prototypes, partially working software
- Integration phase: partially working product prototype
- Testing phase: fully working product prototype.

The availability of prototypes and prototype functionality/maturity are major factors for usability engineering. Based on the prototype availability we can set prototype-based milestones (Mp) accordingly:

- Mp1: Low-fidelity prototypes and mock-ups available.
- Mp2: High-fidelity and mechanical prototypes available.
- Mp3: Partially working product prototype available.
- Mp4: Fully working product prototype available.

Kiljander (1999) discusses the importance of prototypes for mobile phone development. In order to perform efficient usability engineering in the early development phase, a product simulation is required. Thus, the development of simulation is an essential part of early product development and should be an integral part of product development process.

6. NEW MILESTONES AND JOB DESIGN

Milestones are important for job coordination. We have now seen that there are several possible milestones where the changes in design environment have an effect on usability engineering (Figure 5). Those places are related to:

- general product development stages
- capability to accept and implement design changes
- capability to perform iterative design
- current design environment, availability of prototypes.

New usability milestones do not mean that we need more bureaucracy in the project. Knowledge and awareness about milestones and turning points can be used in planning the usability work and job design, in the same way as landmarks, map and compass are needed for navigation. In the planning phase, by identifying the opportunities such as available prototyping tools, and limitations such as project state, the usability practitioner can prepare to work in an optimal way, effectively and efficiently.

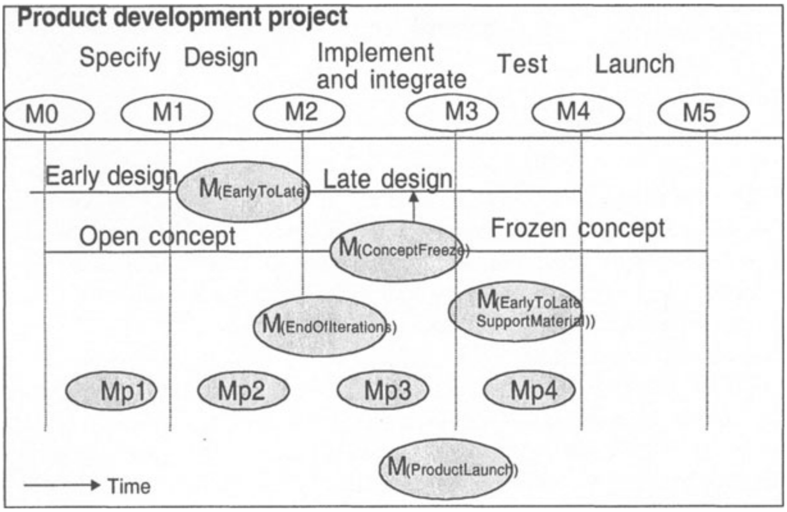

Figure 5. The effect of milestones during product development on usability engineering.

By taking milestones into consideration it is also possible to improve cost efficiency of usability engineering by focusing on areas where changes are possible and where usability engineering is mostly needed. New milestones have two practical consequences related to work organization:

- new usability engineering control points and follow-up mechanisms are needed.
- to some extent, job design or redesign is needed in each milestone.

Earlier tasks can disappear and new ones arise (Järvinen, 1980).

The identification and definition of milestone M(EarlyToLate) is especially useful. The spontaneous tendency of designers is to improve the designs as long as possible, while the will of project management is to minimize all changes after M(EarlyToLate). An explicit definition of this milestone provides improved understanding of the common goals and helps to organize the design work in a better way.

The positive effect of new milestones is the improved ability to coordinate usability engineering. The negative effect of new milestones and job design is the unproductive nature of tasks. Control and job organization do not contribute to actual product development (Järvinen, 1999).

Usability engineering could also be coordinated with less milestones. Removing milestones would potentially lead to improved capability in following the principles of human-centred design, especially iterative design. The drawbacks would be weaker capability to organize the work in fast product development, and weaker visibility and linkage to the product development entity.

7. DISCUSSION AND CONCLUSIONS

The challenge of usability engineering in complex product development is to verify that the final product is usable and that sufficient usability engineering is done in all design areas that have an effect on user experience. We need efficient but simple ways to integrate usability engineering activities into the concurrent development process following the principles of human-centred design.

In this study, I have introduced a new concept (usability milestone) that can be useful in concurrent usability engineering. Based on this new concept, I have identified and analysed several new usability milestones and discussed their effect on usability engineering.

The main result of this study is that usability engineering can be successfully performed in fast concurrent product development but it may require better control mechanisms than described by current research. The control mechanisms can be based on identified new milestones.

Concurrent Engineering projects are time-critical. The attitude towards usability activities is often that they delay product development and cause extra work (Kaderbhai, 1998). An important objective for usability engineering is to support the project in achieving the milestones in the planned timetable. This can be achieved when usability engineering is an integral part of the design process and adapted to the project limitations and opportunities.

A common problem in complex product development is timetable delays. By decreasing the design phase time and increasing the design quality and efficiency with usability engineering, it is possible to provide either more optimal or more reliable timetables and time-to-market. On the other hand, usability engineering typically requires time-consuming iterative design.

This study has taken a look to concurrent usability engineering from the perspective of only one industry and with a limited number of case projects. The introduced usability engineering milestones may be only partially applicable to other product development practices. The timing of milestones is dependent on the particular project.

By analysing potential usability milestones in the beginning of a project it is possible to make a usability plan enabling efficient and effective usability engineering within the given project targets and timetables.

New usability milestones are useful if they improve or verify the quality of the product. The efficiency and usefulness of usability engineering milestones can be measured, for example, with effort metrics (Höglund, 1999): time, cost and results. This kind of measurement is possible when there is reference data from other projects.

Based on the findings in this study, I propose questions that lead to further study. Can usability engineering decrease the uncertainty or delays in product development? How applicable and useful is the M(EarlyToLate) milestone? Are the findings of this study applicable in other industries and in other models of product development, for example Waterfall-based software engineering? And finally, can we develop better usability engineering practices for Concurrent Engineering using defined usability milestones and knowledge about the turning points?

8. REFERENCES

Ayoama M. (1993), *Concurrent Development Process Model*, IEEE Software 1993 Vol. 10, No 4, pp. 46-55.

Daly-Jones O., Bevan N. and Thomas C. (1999), *Handbook of User-Centred Design*. D6.2.1. Version 1.2.

Hakiel S. (1997), Usability Engineering and software engineering: How do they relate? in Smith M.J., Salvendy G. and Koubek R.J (eds.), *Advances in Human Factors/Ergonomics, 21B Design of Computing Systems: Social and Ergonomic Considerations,* in Proc. Seventh Int. Conf. on Human-Computer Interaction, San Francisco, California, USA. Vol. 2, pp. 521-524. Elsevier.

Hakiel S. (1997b), Delivering ease of use, *Computing & Control Engineering Journal* April 1997, pp. 81-87.

Höglund M. (1999), Efficient and Effective Use of Effort Metrics, *The European Conference on Software Process Improvement (SPI 99)*, Barcelona, Spain.

ISO 13407. (1999), *Human-centred design processes for interactive systems,* International standard.

Järvinen P. (1980), On Structuring Problems of Job Design Met, *in The Development and Maintenance of Information Systems*, BIT 20, pp. 15-24.

Järvinen P. (1999), *Oman työn analyysi ja kehittäminen,* Opinpaja Oy, Tampere.

Kaderbhai T. (1998), Overcoming Inertia within a Large Organization, in Trenner L. & Bawa J. (eds.), *The Politics of Usability,* Springer Verlag London, pp. 35-48.

Keinonen T., Nieminen M., Riihiaho S. & Säde S. (1996), *Designing Usable Smart Products*, Helsinki University of Technology, Dept. Computer Science. Report TKO-C81, Otaniemi.

Ketola P. (2001), The Usability Plan as Part of the Concurrent Engineering (CE) Process: An Empirical Study, in Callaos N., Audestad J.A. & Sanchez M. (eds.), *Proc. World Multiconference on Systemics, Cybernetics and Informatics, SCI 2001, Volume IV Mobile/Wireless Computing.* IIIS, pp. 225-230.

Ketola P. & Röykkee M (2001), Ergonomics and Usability Factors In a Mobile Handset. Proc. Annual Congress of the Nordic Ergonomics Society NES 2001, Publications 7, University of Tampere, School of Public Health, Finland, pp. 240-243.

Kiljander H. (1999), User Interface Prototyping Methods in Designing Mobile Handsets, *Human-Computer Interaction INTERACT '99*, pp.118-125.

Korhonen P. (2000), Usability Research in Nokia: Evolution, Motivation and Trust, *CHI 2000 Extended Abstracts*. ACM, pp. 219-220.

Mayhew D.J. (1999), *The Usability Engineering Lifecycle*, Morgan Kaufmann.

Nielsen J. (1994), *Usability Engineering*, Academic Press.

Nieminen M & Parkkinen J. (1998), Usability activities in product development, in Vink P., Koningsveld E.A.P. & Dhondt S (eds.), *Human Factors in Organizational Design and Management – VI*. Elsevier Science B.V, pp. 433-438.

Premkumar G. & King W.R. (1994), Organizational characteristics and information systems planning: An empirical study, *Information Systems Research* 5, No 2, pp. 75-109.

Raddle K. & Young S. (2001), Partnering usability with development: How three organizations succeeded, *IEEE Software January/February 2001*, IEEE, pp. 38-45.

Royce W. (1970), Managing the development of large software systems, IEEE WESCON, reprinted in *Proc. 9th Int. Conf. On Software Engineering (ICSE)*, Monterey, CA 1987, IEEE Computer Society Press, Washington D.C, pp. 328-338.

Tianfield H. (2001), Advanced life-cycle model for complex product development via stage aligned information-substitutive concurrency and detour, *Int. J. Computer Integrated Manufacturing*, 2001, Vol. 14, No. 3, pp. 281-303.

Trenner L. & Bawa J. (1998), *The politics of usability. A practical guide to designing usable systems in industry*, Springer, London.

Valjus P. (1994), Limittäinen tuotekehitys valtaa alaa, *Tekniikan näköalat 1*, pp. 26-27.