MultiCom, a platform for the design and the evaluation of interactive systems.

Jean Caelen
Laboratoire CLIPS-IMAG, Groupe GEOD
Domaine universitaire, BP 53, 38041 Grenoble Cedex 9 – France
Téléphone : 04.76.51.46.27, Fax : 04.76.63.55.52
E-mail : Jean.Caelen@imag.fr

Summary :

This paper presents a platform, named MultiCom, for the design and the evaluation of user interfaces and more generally of interactive systems or products dedicated to the end user. MultiCom is a kind of "usability lab" which contains various distributed resources on network and dedicated software to capture and analyse the human working with a computer. It can be notice that very few generic usability labs exist now in the world. The methodology of MultiCom is based on the *cycle life* used in software engineering domain, by having multidisciplinary approach, unifying several methods on sociology, ergonomics, and computing engineering. The cycle life starts from the original idea of product or system to need and ends by the evaluation after the complete exploitation by the client. The results show that it is very important to design all innovative product or service by integrating the end user in the "development loop" of the system. This point of view implies that the design process have to start from the situated work where a subject is observed in order to capture is behaviour. The data resulting of the capture stage are analysed in terms of usability factors defined a priori for the diagnostic and evaluation. Then, the results allows the designer to fix the precise system functionality. After the prototype realisation, a lot of users is observed again, and so on, then the prototype is modified until a convergence. The paper describes all the tools needed and used in MultiCom platform to manage a complete design session.

1. INTRODUCTION

Bernsen (Bernsen, 97) claims : "It is a well-recognised fact that the production of a new software engineering tool or method is difficult and time consuming. The difficulties lie not only in the initial conception of, for instance, a new tool, or in tool drafting and early in-house testing. Even if these stages yield encouraging results, there is a long way to go before the tool can stand on its own and be used as an integral part of best practice in the field. One central reason why this is the case is the problem of *generalisation*. A tool which only works, or is only known to work, on a single system, in a highly restricted domain of application, or in special circumstances, is of little interest to other developers. In-house testing will inevitably be made on a limited number of systems and application domains and often is subject to other limitations of scope as well. To achieve and demonstrate an acceptable degree of generality, the tool must be iteratively developed and tested on systems and application domains, and in circumstances that are significantly different from those available in-house. Achievement of generality therefore requires access to other systems, corpora and/or development processes. Such access is notoriously difficult to obtain for several reasons, including commercial confidentiality, protection of in-house know-how and protection of developers' time. A second reason why software engineering tool or method development is difficult and time consuming is the problem of *objectivity*. It is not sufficient that some method or tool has been trailed on many different cases and in widely different conditions. It must also have been shown that different developers are able to use the new method or tool with approximately the same result on the same corpus, system or development process. The benefits from using a new tool or method should attach to that tool or method rather than to its originators."

Since many years, the designers of interactive systems need tools and methods for system design and evaluation. This point is crucial in the industrial context where the constraints are hard to satisfy, due to the variety of resources and competencies to collect, in the area of human factors and sociology.

The academic research teams are sought frequently by the industry for studies on user needs, and on user evaluation. It is important to notice this recent evolution : in the past, the evaluation was done at the end of the cycle of development, now the ergonome acts more later and continuously during the lifecycle of the system. So, the industry asks to academic teams about new services or competencies as design studies and usability studies.

It would be more benefit, to create a common platform for the both partners. The platform would be more a *service centre* devoted to the experimentation than a classical academic or industrial institution. These institutions should be clients of the centre. With this definition, the platform must include : a set of observation rooms, some tools for capturing and analysing the user behaviour, that means a complete usability lab.

The envisioned industrial benefits of this platform will be :

- Progress towards the integration of systems best practice into software engineering,
- Improved feasibility assurance of development projects (risk minimisation) and more exact feasibility assessment,
- Improved procedures, methods, concepts and software tools,
- Reduced development costs and time, improved maintenance and reusability,
- Improved product quality and increased flexibility and adaptability,
- Progress towards the establishment of software engineering standards,
- Improved guarantees to end-users that a product has been developed following best software and cognitive engineering practice. Enabling end-users to objectively assess different systems and components technologies against one another and choose the right product according to quality, price and purpose.

2. THE APLICATION FIELD OF MULTICOM

Increasing importance is being attached to the ability of ensuring that user interfaces (UIs) behave as expected, as they are being used in more applications where safety and security are critical. Prototyping is the preferred approach to creating UIs because of the need to allow the user to test many of the issues associated with usability. Unfortunately it is often difficult to verify that prototype-based implementations always behave as expected. In this paper we present a platform for the experiment of UI-based applications.
MultiCom is an approach to software development which emphasises the use of operational models throughout the lifecycle. Software development consists, therefore, in building, testing and refining models within a seamless process that leads the analyst/developer from analysis to design and finally to the implementation of the system.

MultiCom is a complete platform for the design and the evaluation of the interactive systems, including all the stages of the development cycle and supporting tools for sociologic observations and ergonomic tests. MultiCom provides tools and assistance to the following steps of the design :

- to observe and diagnose the usage,
- to define the needs, and the style guide,
- to write the test scenario,
- to experiment different test scenarios,
- to simulate the scenarios,
- to refine the specifications,
- to validate the specifications,
- to evaluate the usability.

The application field of MultiCom are :

- Text processing,
- Machine translation,
- Natural Language processing,
- Multimodal interfaces,
- Man-machine dialogue,
- Speech server and multimodal server,

• Information retrieval,
• Groupware (CSCW),
• Mediaspace and visioconferencing
• Hypermedia,
• Computer education aided.

3. PLATFORM PRESENTATION

The platform contains :

- An observation lab for capturing the behaviour of human subjects, using the "Wizard of Oz" method or the "Direct Observation" method. The lab allows to do objective measurements as : sensory tracking, response delay, activity level, task strategy, error measures, etc.
- A numeric usability lab which allows to do data analyses after annotating observations and filtering. Several tools provide some facilities to the expert in order to obtain results concerning the task model or the user model.
- A studio of audio-video recording,
- Dedicated rooms and offices to the experimentation,
- A software tool collection as statistics, virtual reality simulation, networking, etc.

Hardware and software environment

The observation lab has several computers distributed on network (Unix and Windows) supporting the application to test itself and all the systems of observation needed (cameras, videos, microphones, and others sensors). The Wizards, if needed for the experimentation, are hidden in others rooms than the subject. They dispose all the means of observation on the network, images and sound from the subject.

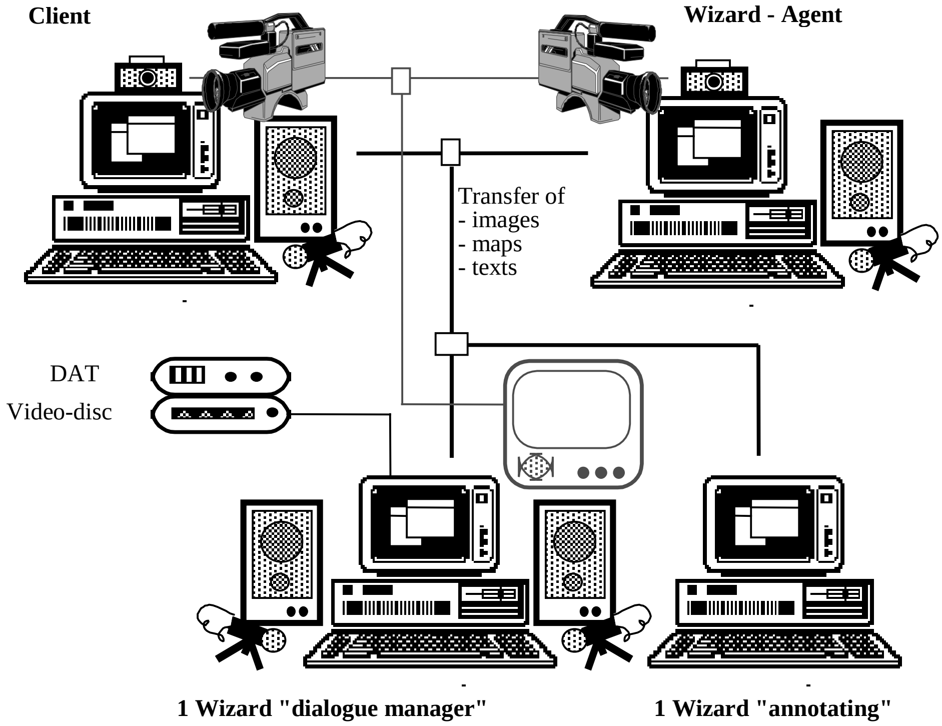

Fig. 1 : Example of configuration for capturing the dialogue between two users. The situation could be an agent and an client negotiating tourism information. Two wizards assist the transaction and complete functions unavailable in the system, for instance, the speech recognition or language translation if needed.

Observation, Evaluation, Recording and Visualisation Environment :
The platform provides an all-inclusive interface usability evaluation environment. It consists of a set of tools designed to facilitate and accelerate the usability evaluation process. The tools support the usability evaluator or analyst by automating a good part of the recording and analysis work.

The tools consist of :

- **Logging tools** to record human-computer interaction. These tools range from event logging to video recording. When multiple recording streams are used, they are automatically synchronised based on their timestamps,
- **Analysis tools** to analyse the recorded interaction. These tools go beyond the traditional statistical analyses by looking at usability evaluation needs and implementing both traditional and innovative usability metrics,
- **Visualisation tools** to display the results of the logs and of the analyses in adequate forms. Our visualisation tools include tools to display analysis results in contextual forms, and tools to combine and synchronise various sets of results and logs to replay or assist the analysis of usability sessions,

The platform is running for Microsoft Windows and X Windows systems. Although not every tool is available for every system, the tools all exchange data in the same format. Therefore, you could, for example, record a session on a Macintosh, analyse it on a Unix system, and view the results on a PC running Windows. Our current tool development effort concentrates on the Microsoft Windows system. This is in response to the wishes expressed by our industrial partners.

MultiCom is being developed by university researchers in a research environment. We listen to our corporate partners to assess their needs, and then our team of scientists (faculty and graduate students) develop and validate new methods and tools to provide novel solutions for our partners. It's a win-win combination: the industrial partners get leading edge tools and methods for usability, and our faculty and students pursue their research interests.

Modularity and reusability

Based on separate and independent modules, the numeric lab if completely reusable and configurable for all types of applications.

4. METHODOLOGY

The methodology of MultiCom is based on the observation of situated utilisation. A set of users is observed when they are working and their actions are recorded. The results from the data analysed, provides parameters and constraints for the design. We consider the following parameters :

- the value and the sense of usage,
- the usability,
- the technological constraints.

Inspection is widely believed to be the most cost-effective method for detecting defects in documents produced during the software development lifecycle. However, it is by its very nature a labour intensive process. This has led to work on computer support for the process which should increase the efficiency and effectiveness beyond what is currently possible with a solely manual process, but no single tool available fills all the identified needs of inspection.

MultiCom uses several and complementary methods :

- the *predictive method* which starts from cognitive theories and takes into account the technology feasibility in order to help the designer in the specification step,
- the *simulated method* (by virtual reality or by wizard of Oz technique) which provides data for the specification refinement step,
- the *direct inspection method* on the prototype for the formal validation and the ergonomic evaluation of the system,
- the *indirect method*, as the reverse engineering, which allows to check the system regarding the specifications.

The design cycle

The methodology of MultiCom goes with the design cycle of the system in order to integrate as soon as possible the human factors in the design process. The benefit expected, is to reduce the time of development and to obtain a best quality of the final product. The design cycle runs on three phases :

Information Elicitation and Analysis Phase: involves collecting and analysing device dependant information (using techniques such as task analysis) concerning the existing tasks of users and the positive and negative features of the systems that they currently use and of other selected systems. The information is summarised in a device-independent (i.e. high level) task model of the existing systems; the system to be designed is modelled at a similar level based on the Statement of Requirements, i.e. the Project Brief.

Design Synthesis Phase: the human factors requirements are established and the semantics of the task domain recorded. A conceptual design for the target system is produced; this is based on the task models produced earlier, and informed by the information collected about the task domain and the human factors requirements. This is then split to separate those sub-tasks that are to be performed using the system under development from those that are performed off-line.

Design Specification Phase: a detailed and device-dependent specification of the user interface is produced; this includes descriptions of the screens, widgets and the interactions. The design is then assessed and refined in an iterative fashion.

More detailed, the design cycle is the following :

A • Observing the use of current systems and predicting the future use with the system to design,
B • Defining the specification notebook,
C • Defining the using scenario and assessing it in situated work with some subjects,
D • Refining the specification on the diagnostic delivered in C,
E • Developing a mock up,
F • Testing the mock up,
G • Evaluating the use and the usability of the mock up,
H • Modifying after debriefing , debugging,
I • Developing another version of the mock up,
J • Returning at the step F while it is necessary,
K • Evaluating the final mock up,
L • Developing a prototype,
M • Pushing it on the market,
N • Training end users if needed,
O • Analysing the market returns.

The cycle (fig. 2) starts generally from an idea proposed by the marketing team. This idea is relatively imprecise : she do not take into account the use constraints nor the technological limits. Therefore, it is necessary to enrich and concretise the starting idea before developing the first model. The steps A, B, C are, in this perspective, a kind of preparatory cycle for testing the feasibility of the idea. Then, the steps D to J are the core of the design cycle running under the trial-error strategy. The step K is critical because it is the decision node before the market proof.

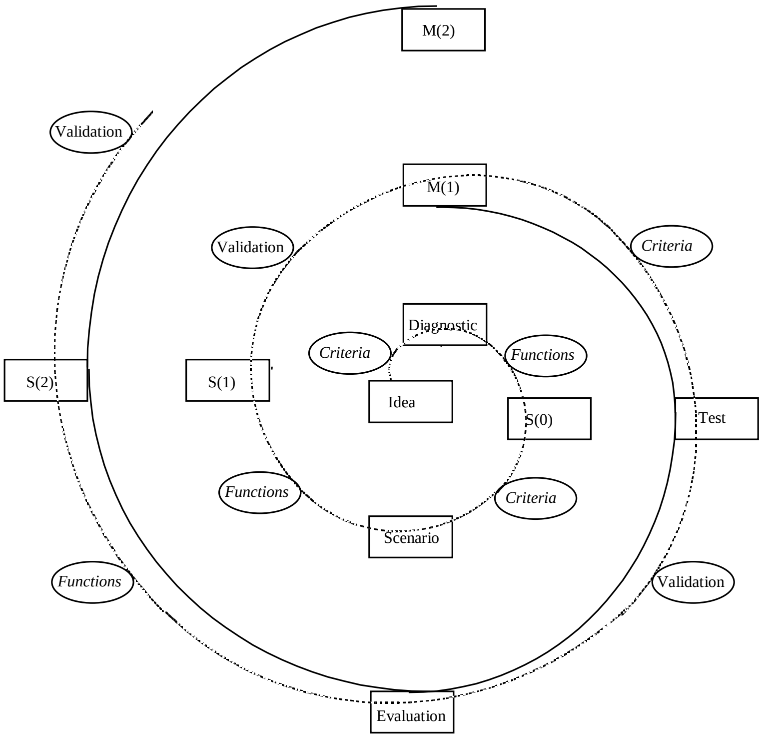

Fig. 2 : The design cycle has a spiral shape (in solid line). The methodology of MultiCom goes with this cycle (in dotted line). At each step, a set of procedures provides criteria, validation functions, specification attributes and evaluative results or diagnostics. Notations : S(i) ith specification, M(i) ith mock up.

The methodology of MultiCom which goes with the design cycle, consists in a set of procedures depending on the step of the cycle. The procedures provide different functions or criteria for the ergonomic evaluation or for the formal validation.

The specificity of the MultiCom methodology is based on the fact that all the criteria concerning the human factors (sociological and psychological) are taken into account as soon as possible, in the design cycle.

5. DISCUSSION : BENEFIT FOR DEVELOPING COUNTRIES IN USABILITY LABS

A major part of the discussion leaded by Nielsen can be reproduced here.

Usability is playing a steadily more important role in software development. This can be seen in many ways, including the growing budgets for usability engineering. In 1971 Shackel estimated that a reasonable share for usability budgets for non-military systems was about 3% (Shackel 1971). Later, in 1989, a study by Wasserman (1989) of several corporations found that "many leading companies" allocated about 4-6% of their research and development staff to interface design and usability work. In January 1993, Nielsen surveyed 31 development projects and found that the median share of their budgets allocated to usability engineering was 6% (Nielsen 1993). Thus, usability budgets have been steadily growing.

Assuming that a company has decided to improve the usability of its products, what should it do ? Even though this is a special issue on usability laboratories, it is not sure that the answer is to build a usability lab right away. Even though usability laboratories are great resources for usability engineering groups, it is possible to get started with simpler usability methods that can be used immediately on current projects without having to wait for the lab to be built. See Nielsen (1993) for an overview of these methods, which are often referred to as "discount usability engineering." Almost always, the result of applying simple usability methods to current projects is a blinding insight that usability methods improve products substantially, making it hard to believe how anybody could ever develop user interfaces without usability engineering. Unfortunately, many people still do just that, and we need cheap and simple methods to enable them to overcome the initial hurdle of starting to use usability engineering.

Once a company has recognised the benefits of the systematic use of usability methods to improve its products, management often decides to make usability a permanent part of the development process and to establish resources to facilitate the use of usability methods by the various project teams. Two such typical usability resources are the usability group and the usability laboratory.

Usability laboratories are typically used for user testing. There is no doubt that user testing is the main justification for most usability laboratories and that user testing is one of the foundations of usability engineering. Once a usability lab is in place, however, it becomes a convenient resource for many other kinds of usability activities such as focus groups or task analysis. Usability laboratories are sometimes also used to record design sessions, though this is mostly done as part of research projects and not as part of practical development projects. With this design technique, much of the design information is never written down, so capturing the dynamics of the design session on video can provide a valuable record for later reference.

Usability laboratories can also be used for certain variants of heuristic evaluation (Nielsen 1994). Heuristic evaluation is based on having usability specialists or other evaluators inspect an interface to find usability problems that violate a list of established usability principles (the "heuristics") and does not normally require a special laboratory since the evaluators can work anywhere and are normally supposed to do so individually. Sometimes, however, one wants to have an observer present to log the usability problems discovered by the evaluators so that they do not need to spend time and effort on writing up a report. It may be valuable for this observer to have access to a video record of the evaluation session, and it may also sometimes be advantageous to have developers or other project representatives observe a heuristic evaluation session from behind the one-way mirror in a usability lab. In general, though, only a small minority of heuristic evaluation sessions take place in usability laboratories.

A usability laboratory does not necessarily need to be a fixed facility in a given set of rooms constructed for the purpose. It can be used "portable usability labs" with video kits and other data logging equipment that can be brought to the field to allow testing. In addition to the building and equipment of the usability laboratory, an important issue is obviously the actual methods used in running experiments in the lab : much of the data reduction process as possible should be automated. Usability testing generates huge masses of raw data, and any meaningful conclusions have to be based on analyses that summarise the test events in a comprehensible manner.

Much user testing is simply aimed at generating qualitative insights that are communicated through lists of usability problems and highlights videos showing striking cases of user frustration. Such insights are sufficient for most practical usability engineering applications where the goal is the improvement of a user interface through iterative design. Often, there is no real reason to expend resources of gathering statistically significant data on a user interface that is known to contain major usability problems and has to be changed anyway. By using a probabilistic model of the finding of usability problems and an economic model of project management, Nielsen and Landauer (1993) found that one often gets the optimal cost-benefit ratio by using between three and five test users for each round of user testing.

The single-most important decision in usability engineering is simply to do it ! The best intentions of some day building the perfect lab will result in exactly zero improvement in current products, and if the choice is between perfection or doing nothing, nothing will win every time. Luckily, it is possible to start small and then grow a usability effort over time as management discovers the huge benefits one normally gets (Ehrlich and Rohn 1994). It is in the nature of things that this special issue mostly has papers on leading usability laboratories from companies that have a better-than-average approach to usability engineering. This does not mean that people in less fortunate companies should abandon usability, it only means that they have something to strive for as they start small and gradually expand their usability groups and usability labs.

6. <u>CONCLUSION</u>

One of the major advantages of having a usability laboratory as MultiCom, is that the incremental hurdle for user testing of a new product becomes fairly small since all the equipment is already in place in a dedicated room that is available for testing. This effect is important because of the compressed development schedules that often leave little time for delay. Thus, if usability testing can be done now and with minimal overhead, it will get done. Similarly, usability may get left out of a project if there is too much delay or effort involved before results become available. Because of this phenomenon, the support staff form a very important part of a usability laboratory in terms of keeping it up and running, stocked with supplies, and taking care of the practical details of recruiting and scheduling test users.

MultiCom is a consulting and research platform specialised in human factors, usability and safety in product and system design. The following is a listing of service areas provided :
User Research : User research is conducted to support system design specifications to assure optimal interaction between user and system. User research includes user surveys, job and task analyses, human factors profiling of user populations, function allocation and other analyses.
User Interface Design : User interface design includes the design of input devices as well the design of screen layout and control elements. Design services include the functional specification of the user interface as well as control and screen layouts.
Usability Inspections : Usability inspections are conducted on a user interface to assure compliance of the interface with existing design guidelines, standards and practices. Inspections can be conducted throughout the development process from initial specification to working prototype.
Usability Testing : Usability testing is conducted on the product or system with representative end-users engaged in representative tasks. Testing is conducted in the laboratory or the work setting of the end-user.
Safety Reviews and Inspections : Safety reviews and inspections of products and systems are conducted by certified human factors and ergonomics professionals in accordance with national and international standards and practices.
Seminars in User-Centered Design and Evaluation : Seminars in the user-centered design process and evaluation methods as well fundamentals of human factors in product design are provided to assist clients in improving product usability and safety.
Human Factors Research : Ongoing human factors research is conducted to determine the nature and extent of the interactive effects of interface design, user, and task characteristics on overall system performance. The research is both externally and internally funded.

6. REFERENCES

L. Bass & J. Coutaz (1991), *Developing Software for the User Interface*, Addison-Wesley.

A. Bisseret (1995), Représentations et décisions expertes, psychologie cognitive de la décision chez les aiguilleurs du ciel, Octares Editions.

J. Caelen (1996). Nouvelles interfaces homme-machine, OFTA série ARAGO n°18, Lavoisier éd., Paris, (coordinateur de l'ouvrage et rédaction de 2 chapitres “ recommandations ”, “ reconnaissance de la parole ”).

J. Caelen, J. Zeiliger, M. Bessac, J. Siroux, G. Perennou (1997) Les corpus pour l'évaluation du dialogue homme-machine, Actes des JST'97, Avignon, AUPELF-UREF éditions.

R.G. Bias. and D.J. Mayhew (Eds.) 1994, *Cost-Justifying Usability* (Academic Press, Boston, MA).

D. Browne (1994), *STUDIO Structured User-Interface Design for Interaction Optimisation*, Prentice Hall.

B. Catterall (1991). *Three approaches to the input of human factors in IT systems design: DIADEM, The HUFIT Toolset and the MOD/DTI Human Factors Guidelines*, Behaviour & Information Technology, 10(5):359-371.

B. Curtis and B. Hefley (1994). *A WIMP No More - The Maturing of User Interface Engineering*, ACM Interactions, Jan. 1994, pp.22-34.

P. Dard, C. Laumonier, Y. Toussaint, P. Mallein (1992), Minitel et gestion de l'habitat. La domotique en questions, Paris, Plan Construction et Architecture, Ministère de l'Equipement, du Logement, des Transports et de l'Espace, mars 1992, 151 p. + annexes (collection Recherches n°17), synthèse in Cahiers du CSTB, n°2594, livraison 330, 44 p.

P. Dard, C. Laumonier, Y. Toussaint, P. Mallein (1996), *Télésurveillance et gestion de l'habitat public*, synthèse in Cahiers du CSTB, n°2924, livraison 375, 75 p.

P. Dard, C. Laumonier, Y. Toussaint, P. Mallein (1996), *Réseaux de communication et services résidentiels*, synthèse in Cahiers du CSTB, n°2869, livraison 367, 111 p.

K. Ehrlich and J. Rohn. (1994), Cost-justification of usability engineering: A vendor's perspective, in Bias, R.G., and Mayhew, D.J. (Eds.), *Cost-Justifying Usability* (Academic Press, Boston, MA).

J. Froger., P. Mallein (1995), *Concevoir et diffuser une nouvelle technologie d'information*, cahiers du CSTB n°2779, livraison 356, 17 p.

D. Hix and H. R. Hartson (1993). *Developing User Interfaces: Ensuring Usability Through Product and Process*, Wiley, New York.

ISO 9241, *Ergonomic Requirements for office work with Visual Display Terminals*. ISO.

K. Y. Lim and J. Long (1994). *The MUSE Method for Usability Engineering*, Cambridge University Press.

M. Jackson (1983). *Systems Development*, Prentice Hall.

L. Macaulay, C. Fowler, M. Kirby and A. Hutt (1990*). USTM: A New Approach to Requirements Specification*, Interacting with Computers, (2)1:92-118.

M.J. Muller, D.M. Wildman and E.A. White (1993), "Equal opportunity" PD using PICTIVE. *Communications of the ACM* **36**, 4, 64-66.

J. Nielsen (1993), *Usability Engineering* (Academic Press, Boston, MA).

J. Nielsen (1994), Heuristic evaluation, in Nielsen, J., and Mack, R. L. (Eds.), *Usability Inspection Methods*. (John Wiley & Sons, New York, NY), 25-64.

J. Nielsen and T.K. Landauer (1993), A mathematical model of the finding of usability problems, *Proceedings of the ACM INTERCHI'93 Conference* (Amsterdam, the Netherlands, April 24-29), 206-213.

B. Senach (1990), *Evaluation ergonomique des interfaces homme-machine*, in Revue de la littérature, rapport de recherche INRIA n°1180.

B. Shackel (1971), Human factors in the P.L.A. meat handling automation scheme. A case study and some conclusions. *International Journal of Production Research* **9**, 1, 95-121.

A. G. Sutcliffe and M. McDermott (1991). *Integrating methods of human-computer interface design with structured systems development*, Int. J. of Man-Machine Studies, (34):631-655.

A.S. Wasserman, P. Pircher, D. Shewmake and M. Kersten (1986). *Developing Interactive Information Systems with the User Software Engineering Methodology*, in Readings in HCI, Baecker & Buxton [eds], pp.508-527, Morgan Kaufman.

A.S. Wasserman (1989), Redesigning Xerox: A design strategy based on operability. In Klemmer, E. T. (Ed.), *Ergonomics: Harness the Power of Human Factors in Your Business*, Ablex, Norwood, NJ. 7-44.

J. Zeiliger, J. Caelen, J.Y. Antoine (1997), Vers une méthodologie d’évaluation quantitative des systèmes de compréhension et de dialogue oral homme-machine, Actes des JST’97, Avignon, AUPELF-UREF éditions