Maturation of Usability Evaluation Methods: Retrospect and Prospect

Final Reports of
COST294-MAUSE
Working Groups

(Editors)
Effie Lai-Chong Law
Chair

Dominique Scapin
Working Group 1 Leader

Gilbert Cockton
Working Group 2 Leader

Mark Springett
Working Group 3 Leader

Christian Stary
Working Group 4 Leader

Marco Winckler
Dissemination Leader

http://cost294.org/
Publisher: IRIT Press, Toulouse, France
Maturation of Usability Evaluation Methods: Retrospect and Prospect
(Final Reports of COST294-MAUSE Working Groups)
Effie L-C. Law, Dominique Scapin, Gilbert Cockton, Mark Springett, Christian Stary, Marco Winckler (eds.)
188 pages. 2009


© 2009 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. Re-publication of material on this page requires permission by the copyright owners.
Maturation of Usability Evaluation Methods: 
Retrospect and Prospect
Final Reports of COST294-MAUSE Working Groups

Editors

Effie Lai-Chong Law
Chair

Dominique Scapin
Working Group 1 Leader

Gilbert Cockton
Working Group 2 Leader

Mark Springett
Working Group 3 Leader

Christian Stary
Working Group 4 Leader

Marco Winckler
Dissemination Leader
COST

COST is an intergovernmental framework for European Cooperation in Science and Technology, allowing the coordination of nationally-funded research on a European level. COST contributes to reducing the fragmentation in European research investments and opening the European Research Area to cooperation worldwide.

The goal of COST is to ensure that Europe holds a strong position in the field of scientific and technical research for peaceful purposes, by increasing European cooperation and interaction in this field. This research initiative makes it possible for the various national facilities, institutes, universities and private industry to work jointly on a wide range of Research and Development (R&D) activities.

COST – together with EUREKA and the EU framework programmes – is one of the three pillars of joint European research initiatives. These three complementary structures have differing areas of research.

COST has clearly shown its strength in non-competitive research, pre-normative cooperation, and solving environmental, cross-border and public utility problems. It has been successfully used to maximise European synergy and added value in research cooperation and is a useful tool to further European integration.

Ease of access for institutions from non-member countries also makes COST a very interesting and successful tool for tackling topics of a truly global nature.

As a precursor of advanced multidisciplinary research, COST plays a very important role in building a European Research Area (ERA). It anticipates and complements the activities of the EU Framework Programmes, constituting a “bridge” towards the scientific communities of emerging countries. It also increases the mobility of researchers across Europe and fosters the establishment of scientific excellence in the nine key domains:

- Biomedicine and Molecular Biosciences
- Food and Agriculture
- Forests, their Products and Services
- Materials, Physical and Nanosciences
- Chemistry and Molecular Sciences and Technologies
- Earth System Science and Environmental Management
- Information and Communication Technologies
- Transport and Urban Development
- Individuals, Societies, Cultures and Health

In addition, Trans-Domain Proposals allow for broad, multidisciplinary proposals to strike across the nine scientific domains.

Source: http://www.cost.esf.org/about_cost

COST Action 294 ([http://www.cost294.org](http://www.cost294.org)), which is also known as MAUSE (Towards the Maturation of IT Usability Evaluation), was officially launched in January 2005. The ultimate goal of COST294-MAUSE is to bring more science to bear on Usability Evaluation Methods (UEM) development, evaluation, and comparison, aiming for results that can be transferred to industry and educators, thus leading to increased competitiveness of European industry and benefit to the public.
Acknowledgements

The COST294-MAUSE Closing Conference (19-21 March 2009, Brindisi, Italy) is held under the auspices of COST. As with other past events of COST294-MAUSE, we aim to provide the participants with enlightening environments to further deepen and broaden their expertise and experiences in the area of usability.

First of all, we are much obliged to Professor Maria Francesca Costabile, Dr. Carmelo Ardito, and Dr. Rosa Lanzilotti for their great efforts in hosting the Conference at the University of Bari, Italy.

We would also like to express gratitude to our invited experts:

- **Mark Blythe**, University of York, UK
- **Antonella De Angeli**, University of Manchester, UK
- **Boris De Ruyter**, Philips Research, the Netherlands
- **Peter Forbig**, University of Rostock, Germany
- **Rolf Molich**, DialogDesign, Denmark
- **Virpi Roto**, Nokia Research, Finland
- **Ahmed Seffah**, EHL, Lausanne, Switzerland
- **Antonella Toffetti**, Centro Ricerche Fiat S.C.p.A., Italy

for their generosity of sharing their knowledge and experience about usability and user experience research.

We express our high appreciation towards the leadership of the four Working Group (WG) Leaders, **Dominique Scapin** (WG1), **Gilbert Cockton** (WG2), **Mark Springett** (WG3) and **Christian Stary** (WG4). Their dedicated efforts in coordinating the scientific activities of the four WGs have led to some very interesting results, which serve as valuable inputs for the future research agenda in the domain of usability and user experience. Besides, we are grateful to the efforts of the Dissemination Leaders, **Marco Winckler** and **Philippe Palanque**, in maintaining the Action’s website and collaborative platforms – Digital Library and wikis – and in coordinating the publication of the Action’s workshop proceedings and newsletters, thereby enhancing the visibility of the Action.

Thanks should also go to each of COST294-MAUSE members, whose active participations have contributed to the great success of the Action.

Last but not the least, we are very grateful to the ongoing support of the COST ICT Domain Committee members, especially Dr. **Afonso Ferreira** (Head of Science Operations), Dr. **Gian Mario Maggio** (Senior Science Officer), Ms. **Sophie Beaubron**, (Senior Administrator), Mr. **Graham Worsley** (DC member, the Action’s Rapporteur), and Ms. **Francesca Boscolo** (Science Officer).

Effie Lai-Chong Law  
Chair

Ebba Thora Hvannberg  
Vice-Chair
Preface

COST Action 294-MAUSE is a four-year project (2005-2009) with the mission of bringing more science to bear on the development of usability evaluation methods (UEMs) (http://www.cost294.org). Twenty-one European countries are involved in this Action with a number of young and experienced researchers from the highly interdisciplinary field of Human Computer Interaction (HCI). The Action aims to achieve three major objectives:

- To deepen understanding about strengths and weaknesses of Usability Evaluation Methods;
- To identify reliable and valid methods to compare different UEMs in terms of their effectiveness, efficiency as well as scope of applicability;
- To develop efficacious strategies for extracting useful information from results of usability evaluation to improve the system tested;

They are realised through the scientific activities of the four Working Groups (WGs).

The COST294-MAUSE Closing Conference aims to reflect on the achievements of the four WGs with invited experts from academics and industry. With implications inferred from these reviews, we strive to draw a roadmap for the future usability research. The title of the Conference Proceedings “Maturation of Usability Evaluation Methods: Retrospect and Prospect” captures this idea. The Proceedings consist of four parts with each of them reporting the respective outcomes of the four WGs. With the inputs from the Action’s partners, the four WG Leaders have addressed the corresponding research challenges and concluded with highly relevant results.

During the lifetime of COST294-MAUSE, a series of open workshops addressing the topics closely related to the ongoing work of the four WGs have been organised, including:

- « Towards a Unified View of User Experience », 14th October 2006, Oslo, Norway
- « Review, Report and Refine (R3) UEMs », 5th March 2007, Athens, Greece
- « Towards a UX Manifesto », 3rd September 2007, Lancaster, UK
- « Meaningful Measures: Valid Useful User Experience Measurement (VUUM) », 18th June 2008, Reykjavik, Iceland
- « Interplay between Usability Evaluation and Software Development (I-USED) », 24th September 2008, Pisa, Italy

These workshops were open to the communities of interests, thereby enhancing the visibility of the Action as well as bringing the usability research forward with frontline results.

In addition, Short-Term Scientific Missions (STSMs) have been proved effective in advancing the specific research topics through close collaborations between researchers from different institutions. Besides, the success of the Training School DEVISE (http://www.tik.ee.ethz.ch/~law/DEVISE/) can showcase the necessity and utility of interdisciplinary and trans-sectorial training that is of great need for information and communication technology (ICT) research and practice. In particular, the association of three recently completed PhDs with the COST294-MAUSE project is one of the strong outcomes of this Action, which has provided a unique context for young European usability researchers to develop innovative approaches to understanding usability work and issues.

Last but not the least, the edited book “Maturing Usability: Quality in Software, Interaction and Value” (Law, Hvannberg & Cockton, 2008; London: Springer), which has crystallised some of the important ideas on usability addressed in the Action, is another major achievement.

To conclude, COST294-MAUSE has successfully fulfilled its mission of advancing and integrating the usability research by networking researchers with a series of scientific activities. The paradigmatic shift in the field of HCI (i.e. Third Wave) has led to the new emphasis on user experience (UX); emergent concepts, methodologies and tools pose further challenges to usability researchers. We strive to pursue this new line of scientific inquiry by exploiting the results produced by the Action and sustaining the fruitful collaborative relationships established therein.
# Table of Contents

**Part 1:** .............................................................. pp.1-17
Critical Review and Analysis of Individual Usability Evaluation Methods
(Final Report of COST294-MAUSE Working Group 1)

**Part 2:** .............................................................. pp. 18-82
Comparing Usability Evaluation Methods: Strategies and Implementation
(Final Report of COST294-MAUSE Working Group 2)

**Part 3:** .............................................................. pp.83-118
Understanding Quality Attributes with Repertory Grid Technique
(Final Report of COST294-MAUSE Working Group 3)

**Part 4:** .............................................................. pp.119-188
Review on the Computational and Definitional Approaches in Usability Evaluation
(Final Report of COST294-MAUSE Working Group 4)
Critical Review and Analysis of Individual Usability Evaluation Methods
Final Report of COST 294-MAUSE Working Group 1

Dominique L. Scapin
INRIA, Rocquencourt, France
Leader, COST294-MAUSE Working Group 1

Contributors:

Nigel Bevan
Professional Usability Services, London, UK

Effie Lai-Chong Law
ETH Zürich, Switzerland/
University of Leicester, England

Other WG1 Members
Abstract

This final report of the MAUSE COST 294 WG1: Critical Review and Analysis of Individual UEMs summarizes the activities carried out and outputs delivered during the course of the project. Essentially, WG1, through 10 short physical meetings and offline activities produced firstly several instruments for describing and classifying UEMs “by nature” (a classification scheme of UEMs: two templates (one for «Generic Methods»; one for «Case Studies»); and a guidance document for these templates); secondly a 127-page document “Review, Report and Refine Usability Evaluation Methods (R3 UEMs)” (http://cost294.org/upload/522.pdf) describing a set of 39 UEMs and 8 case studies, with input from 33 authors (from 14 European countries); thirdly contributions to a website (http://www.usabilitybok.org/) with the Usability BoK (Book of Knowledge) description and classification of UEMs from a practitioner’s point of view. This report discusses also the limits of the approaches and identifies further collaborative activities and issues for the selection of UEMs.
Table of Contents

Introduction ............................................................................................................................................ 4

Current State of WG1 Contributions ................................................................................................. 5

Initial Classification and Description ................................................................................................. 5

Classification Instruments .................................................................................................................. 5

The Templates Attributes .................................................................................................................... 5

Classification Scheme ...................................................................................................................... 7

Methods Described ............................................................................................................................. 9

Limits of the “R3 UEMs” Contribution ............................................................................................... 11

Further Dissemination ......................................................................................................................... 11

Nature of BoK Description .................................................................................................................. 11

Current State on BoK Wiki .................................................................................................................. 11

Limits of the BoK Contribution ............................................................................................................ 13

Future Work ....................................................................................................................................... 13

References .......................................................................................................................................... 15

Annex A: List of Contributors ............................................................................................................. 16

Annex B: MAUSE initial plan ............................................................................................................... 17
Introduction

COST294-MAUSE is a usability research community. Its ultimate goal is to bring more science to bear on usability evaluation through scientific activities of four Working Groups; each of them aims to attain specific objectives and outcomes. WG1, one of the Working Groups, focuses on the critical review and analysis of individual UEMs (Usability Evaluation Methods). WG1 has the goal to build a refined, substantiated and consolidated knowledge-pool about usability evaluation based on the expertise, experiences and research work of the participating partners. Selected usability methods are critically reviewed and analyzed on various topics, from bibliographical references, through various methodological attributes, to advantages/ drawbacks.

The expected outcome is to provide a set of records (classifications, reports, templates, case studies, etc.) to be disseminated through the use of a database (MAUSE Digital Library). A critical classification and best practices of existing UEMs, covering operational, organizational and cultural dimensions, can be derived from records of this database and rendered accessible to researchers and practitioners within and outside the community.

Indeed, a variety of usability evaluation methods (UEMs) are employed in a wide spectrum of contexts by people with different backgrounds, goals and needs. Selecting appropriate UEMs to meet contextual requirements and constraints is the foremost and crucial step that leads to useful evaluation outcomes and presumably effective redesign of the system of interest. Furthermore, emerging information technologies (IT) such as ambient intelligence, pervasive computing and universal accessibility have triggered the development of new evaluation methods and tools, which have been adopted or trialed by a local research group but not yet well disseminated to the entire usability community. A refined and consolidated knowledge-pool about established as well as emerging UEMs based on expertise and experiences of usability practitioners and researchers is deemed desirable. It will not only enable the selection of right methods but also serve as valuable resources for informing experienced members of the usability community about new UEMs as well as for training newcomers about the development of UEMs.

With the aim to build such a knowledge-pool, i.e., to perform systematic, critical reviews on a variety of UEMs, to derive or refine best practices of established UEMs, to explore emerging UEMs, identifying their applicability and potentiality, WG1 has worked through 10 short physical meetings and off line activities (individual discussions, remote exchanges (videoconferences, emails, etc.).

The work was launched at the 1st MAUSE WG Meeting: Sunderland, UK, 3-4 May 2005. It was followed by:

- 2nd MAUSE WG Meeting: Rome, Italy, 10-11 September 2005
- 3rd MAUSE WG Meeting: Konstanz, Germany, 20-21 March 2006
- 4th MAUSE WG Meeting: Oslo, Norway, 15-16 October 2006
- 5th MAUSE WG Meeting: Athens, Greece, 6-7 March 2007
- 6th MAUSE WG Meeting: Salzburg, Austria, 7-8 June 2007
- 7th MAUSE WG Meeting: Toulouse, France, 7-8 November 2007
- 8th MAUSE WG Meeting: Bled, Slovenia, 3-4 March 2008
- 9th MAUSE WG Meeting: Reykjavik, Iceland, 18-19 June 2008
- 10th MAUSE WG Meeting: Pisa, Italy, 22-23 September 2008
- 11th MAUSE Closing Conference: Brindisi, Italy, 19-21 march 2009

This final report of COST294-MAUSE WG1 describes the work accomplished and the research avenues to be further followed. It summarizes the activities carried out and outputs delivered during the course of the project. Essentially, WG1 produced firstly several instruments for describing and classifying UEMs “by nature” (a classification scheme of UEMs: two templates (one for «Generic Methods»; one for «Case Studies»); and a guidance document for these templates); secondly a 127-
page document (http://cost294.org/upload/522.pdf) describing a set of 39 UEMs and 8 case studies, with input from 33 authors1 (from 14 European countries); thirdly contributions to a web site (http://www.usabilitybok.org/) with the description and classification of UEMs from a practitioner’s point of view.

This report discusses each one of the contributions as well as the limits of the approaches and identifies further collaborative activities and issues for the selection of UEMs.

**Current State of WG1 Contributions**

**Initial Classification and Description**

In order to provide a common base for usability methods descriptions, WG1 templates have been designed based on a classification scheme developed by Scapin (2005)2. This scheme together with sets of description attributes (including context for case studies) extracted from ISO TR 169823 was further tailored and modified through emails, teleconferences, as well as face-to-face discussions during the following COST294-MAUSE meetings and workshops.

**Classification Instruments**

Specifically, WG1 has developed several instruments:

- A classification scheme of UEMs: Three major categories are DGMM (Data Gathering & Modelling Methods); UIEM (User Interactions Evaluation Methods); CMs (Collaborative Methods), each of which is further divided into sub-categories.

- Two templates:
  - «Generic Methods» - to support descriptions of widely used UEMs at the generic level, i.e. mainly using reference material such as publication, courses, etc.
  - «Case Studies» - to support description of actual cases of UEM implementation, i.e. details on how a specific method was used, with its context, its precise usage of the method.

- A guidance document for these templates

Selected UEMs are categorized, critically reviewed and analyzed on different aspects, from bibliographical references to advantages/disadvantages, through a set of methodological attributes. Individual reviews are documented as a set of records in the MAUSE Digital Library. Best practices of existing UEMs, covering operational, organizational and cultural dimensions, can be derived from these records and rendered accessible to the usability community.

**The Templates Attributes**

After modifications mainly suggested during the WG1 Rome meeting, 3 files have been made available to the COST294-MAUSE participants on the Digital Library (http://www.cost294.org/dl.php : # 342, # 343, # 344):
- A “Comments” file that provides information on the contents of the 2 other files
- A template for “generic methods” file

---

1 See Annex A for the list of contributors
- A template for “case studies” file
  Basically, the “generic methods” template is used to describe in detail a particular UEM, while the “case studies” template is used to describe a UEM in similar terms, with the addition of a section being dedicated to the description of the context and results experienced in applying that particular UEM.

The “generic methods” template is aimed to support descriptions of widely used UEMs at the generic level, i.e., mainly using reference material such as publications, courses, etc.

The “case studies” template is aimed to support descriptions of actual cases of UEM implementation, i.e., details on how a specific method was used, with its context, its precise usage of the method, etc. The case studies can be of course published material, but also current evaluation work and reports within the MAUSE partners’ community.

For the “case studies” template, the method description part is available in case the “generic method” has not yet been described. If the method has been already described, it needs not to be described again, but simply cross-referenced to the applicable “generic methods” template. Also, for the “case studies” template, the various categories to look at for describing the context of the project are described in the “Comments” file.

The main attributes of both templates, besides author of the document, date, version, and cross-reference between “generic methods” and “case studies” are the following:

**Method identification** which consists of:
- Name and category of the method (see next section explaining the classification scheme)
- Origin:
  - Author(s), date, title & full bibliographical reference(s)
  - Theoretical Background/ Reference Framework: design origin (empirical, theoretical, classificatory, etc.); "design rationale", i.e., how the method was built (e.g. from literature survey, theory, scratch, sets of recommendations, etc.) and "focus" and/or "links", i.e., to express its main focus, its relationship with other methods, etc.
  - Abstract: short abstract of the referenced paper/ short method description
  - History: history of the method, version described, main evolutions, etc

**Method characteristics.** This section deals with the known characteristics of the method, what it is intended for, when to use it, its pre-requisites and constraints, its advantages, etc. It consists of:
- Function/ Goal: main goals, main intent in terms of output/ result. What are the questions the method addresses
- Relationship with Software Engineering: type of dimensions the method is dealing with in terms of software engineering (e.g., from ISO 9126).
- Method application scope (platform): all, GUIs only, web only, mobile only, others.
- Method application sector (specific domain(s)).
- Method application to lifecycle stage: Requirements analysis, Architectural design, and Qualification testing are sub-steps of the development process (ISO/IEC 12207); detailed step(s) concerned (e.g., training); detailed function of the method regarding its contribution to the step(s) concerned).
- Procedures: Comments on different steps of the method, its different ways, etc
- Prerequisites/ Constraints: comments on the limitations, constraints of the method (e.g., resources, skills and knowledge required, apparatus required, software required, "learnability of the method", "reliability of the method's output", etc.).
- Research questions: Comments on the unsolved research questions that would need to be addressed.

**Summary of advantages:** description and source (authors statements, possibly documented by empirical evidence, etc.).
Summary of drawbacks: description and source (authors statements, possibly documented by empirical evidence, etc.).

Estimated Impact:
- Extent of Use in Industry: [1: very low, 7: very high]; and qualitative comments.
- Acceptance in the Research Community: [1: very low, 7: very high]; and qualitative comments.
- Extensibility: [1: very low, 7: very high]; and qualitative comments

Other comments: An open section for issues not covered by the templates.

Means of assessment: This section is a place holder for WG2 contributions. It has to do with the various dimensions on which the method has been assessed (validity, reliability, flexibility, etc.) and the means of assessment used (e.g., empirical evaluation, with appropriate bibliographical references).

For the “case study” template, additional information is provided. It allows the description of the systems/projects the methods have been applied to, and the specific, detailed characteristics of the method implementation. (Several case studies can be assigned to the same “generic method”).

In addition, a set of attributes have been proposed to cover some of the case study context. These are extracted from ISO TR 16982 and were initially intended to facilitate the choice of usability methods:
- Very tight time-scale
- Cost/price control
- High quality level
- Need for early assessment
- Highly evolutive
- Users can / cannot be involved
- Significant disabilities/ Accessibility
- Task highly complex
- Severe consequences of errors
- New task
- Wide task spectrum
- Major organizational changes
- High levels of time/ accuracy constraints
- Existing system/product
- Limited and simple product
- High degree of adaptability
- Availability of ergonomic expertise

Classification Scheme

The classification scheme distinguishes three categories of methods: DGMM (Data Gathering & Modelling Methods); UIEM (User Interactions Evaluation Methods); CMs (Collaborative Methods).

A. DGMM

DGMM (Data Gathering & Modelling Methods) are used for gaining knowledge about users and their activities. They can be particularly useful for evaluation for example in usage diagnoses and comparison of products. Two subcategories are distinguished: DGM (Data Gathering Methods) and MM (Modelling Methods).

A.1. DGM

DGM (Data Gathering Methods). These methods focus on the ways to gather knowledge about the relevant characteristics of the users, the tasks and the context in which interactive systems are being used. These individual methods are also often included in large sets of evaluation
methodologies. Examples of such methods are: Observations, Interviews, Questionnaires,
Thinking aloud, Critical (incidents) analysis, Etc.

A.2. MM

MM (Modelling Methods). These methods are often associated with specific data gathering
methods or their combination. Their goal is to provide, at various levels of detail, with variable
underlying models and languages, an accurate representation of the users and their activities.
These models are used in many ways, simply as data repositories, but also to help analyses,
comparisons of interactions, etc. Some of them are associated with software tools. Examples
of such methods are: GOMS (Goals, Operators, Methods, and Selection Rules), HTA
(Hierarchical Task Analysis), MAD (Methode Analytique de Description de tâches – Method
for Activity Description), CTTE (ConcurTaskTreeEnvironment), Etc.

B. UIEM

UIEM (User Interactions Evaluation Methods) are explicitly targeted towards evaluation. They
correspond to two subcategories: (1) KMbM (Knowledge-based and Model-based Methods) are
analytical, and may not necessarily required direct access to users; (2) EM (Empirical Methods) which
can be only used after some form of interaction design is available, and for which direct access to end-
users is mandatory.

B.1. KMbM

KMbM (Knowledge-based and Model-based Methods) are based on available ergonomics
knowledge (experimentation, recommendations, standards, models, etc). These methods are
particularly useful when it is not possible to collect data
directly from the users; but also, they can simply be a useful first step to uncover some major
usability flaws before investigating further more complex problems. They are subdivided into

B.1.1. Expert evaluation
i.e., inspection based solely on the evaluator’s knowledge and experience.

B.1.2. Document-based evaluation
i.e., inspection based on some guiding documents, at various degree of precision.
Examples of such methods are: Standards, Classifications schemes, Guides/
Guidelines / Checklists, Heuristic inspection, Ergonomic Criteria, Etc.

B.1.3. Model-based evaluation
i.e., inspection based on some theoretical model, usually cognitive. Examples of such
methods are: Cognitive Walkthrough, CASSM (Concept-based Analysis of Surface
and Structural Misfits), Etc.

B.2. EM (Empirical Methods)
i.e., user testing. Examples are: User performance testing, User preferences testing,
Etc.

One additional category has been added in order to accommodate descriptions combining different
methods: Mixed Approaches.

C. CM

CMs (Collaborative Methods), also referred to as “creativity methods” require the active participation
of the users in the evaluation or design, using various forms of organization and communication.
These methods are often part of larger methodologies known as participative evaluation (or design). These are methods for bringing, more or less formally, sets of users or designers to exchange ideas, results, designs, from a usability evaluation point of view. Examples of such methods are: Brainstorming, Focus groups, etc.

Methods Described

Below are the various methods that have been described in the document “Review, Report and Refine Usability Evaluation Methods (R3 UEMs)” following the Athens Workshop (http://cost294.org/upload/522.pdf): 39 UEMs are described, accompanied with 8 case studies.

This was made possible by the input of 33 MAUSE consortium authors (from 14 European countries: Austria, Denmark, Finland, Netherlands, Norway, France, Germany, Greece, Iceland, Italy, Poland, Romania, Switzerland, and UK).
Methods described in the document “Review, Report and Refine Usability Evaluation Methods (R3 UEMs)“

DGMM (Data Gathering & Modelling Methods)

**DGMM**

**DGM** (Data Gathering Methods)

- **Surveys - Questionnaires** by Jacek Wachowicz
- **Think-aloud method** by Marta Kristin Larusdottir
- **Card Sorting** by Jacek Wachowicz
- **Context of Use** by Marcin Sikorski
- **Cultural Probing** by Regina Bernhaupt and Marianna Obrist

**MM** (Modelling Methods)

- **K-MADE** by Dominique L. Scapin
- **Personas** by Niina Kantola
- **GOMS** by Costin Pribeanu
- **CPM-GOMS** by Georgios Christou
- **KLM** by Costin Pribeanu
- **NGOMS** by Costin Pribeanu
- **TAG** by Costin Pribeanu
- **HTA** by Costin Pribeanu
- **TKS** by Costin Pribeanu
- **GTA** by Costin Pribeanu
- **CTT** by Fabio Paterno
- **User Scenarios** by Niina Kantola and Timo Jokela
- **User Satisfaction Models** by Marcin Sikorski

UIEM (User Interactions Evaluation Methods)

**KMbM** (Knowledge-based and Model-based Methods)

- **Expert evaluation**
  - **Group-based Expert Walkthrough** by Asbjørn Følstad
  - **Document-based evaluation**
  - **Heuristic Evaluation** by Effie Law
  - **Heuristic Walkthrough** by Gilbert Cockton and Darryn Lavery
  - **Ergonomic Criteria** by Dominique L. Scapin
  - **Structure Expert Evaluation Method: SEEM** by Tilde Bekker
  - **DEPTH** by Petros Georgiakis, Symeon Retalis and Yannis Psaromiligkos

- **Model-based evaluation**
  - **Cognitive Walkthrough** by Ann Blandford
  - **Cognitive Dimensions** by Gilbert Cockton and Darryn Lavery
  - **Abstract Task Inspection** by C. Ardito, M.F. Costabile and R. Lanzilotti

**EM** (Empirical Methods)

- **User performance testing** by Jens Gerken
- **ActA** by Chris Stary and Alex Totter
- **Instant Data Analysis (IDA)** by Jesper Kjeldskov, Mikael Skov and Jan Stage

**Mixed approaches**

- **CASSM : Concept-based Analysis of Surface and Structural Misfits** by Ann Blandford
- **MOT : Metaphors of human thinking** by Erik Frokjer and Kasper Hornbæk
- **User-Action Framework** by Mark Springett
- **Claims Analysis** by Suzette Keith and Mark Springett
- **MultiModalWebRemUsine** by Carmen Santoro

**CM** (Collaborative/ Creativity Methods)

- **CUT : Cooperative usability testing** by Erik Frokjer and Kasper Hornbæk
- **Cooperative User Experience Research** by Sarah Burton-Taylor and Rebecca Whitfield
Limits of the “R3 UEMs” Contribution

Within WG1, in a collaborative manner, a number of goals have been reached with that contribution:
- iterations and agreement on a structure and templates allowing the description of UEMs and case studies
- an extensive version of methods compilation
- delivery of all items to the Dissemination team (Digital Library)

However, there are still some issues that should be further considered:
- still a few methods missing in the methods compilation (e.g., Collaborative methods; UX testing)
- the descriptions vary in their level of precision
- very few case studies; 8 cases for 39 method descriptions
- not much return of experience on the actual use of the document “Review, Report and Refine Usability Evaluation Methods (R3 UEMs)”, as a support, in real life situations of selecting UEMs, but such feedback takes time

There are several avenues that were considered as follow-up activities (testing the use of the method compilation; designing tutorials on usability evaluation methods, and a practical handbook/guidebook, etc.). One additional step of the WG1 activity has been to explore how further dissemination, under a different philosophy of access (and a focus on the population of usability practitioners) has been followed. This effort is described in the next section.

Another issue for further discussion within COST294-MAUSE relates to the consolidation and cross-fertilization of the various WGs is discussed in section Further Work.

Further Dissemination

Nature of BoK Description

The Usability Body of Knowledge, sponsored by the Usability Professionals Association, aims to summarise the state of the art in all areas of usability, and also to summarise the latest research results that could impact on good practice. Draft information is currently being collected on a wiki [http://draft.usabilitybok.org/wiki](http://draft.usabilitybok.org/wiki), for which 430 people have requested a login. After review by experts in the field, material is being transferred to the public site [http://usabilitybok.org/](http://usabilitybok.org/). This will in future be enhanced to provide multiple retrieval mechanisms and different views of the information. The current (draft) criteria for acceptance are:
- Normal entries should summarise good practice (in specified conditions).
- If possible, entries should cite reliable, published sources.
- If there are no appropriate reliable, published sources, the entry will be reviewed to ensure that it summarises current good practice.
- If not, it may be placed in the "research" category described below, placed on hold for insufficient information, or rejected because it does not meet the level of quality for the Body of Knowledge.

Current State on BoK Wiki

So far the largest area is methods, with 42 methods described in detail (including 12 with contributions from COST294-MAUSE). Methods are currently classified from a practitioner perspective, related to the project life cycle. However, it is planned to support additional retrieval methods. The topic headings for description of methods have been developed iteratively with methods authors to provide a comprehensive top-down description from a practitioner perspective. The Usability BoK now lists 109

---

4 Research entries are those that cannot yet be considered good practice. They must be labelled “Research”, and be based on published research results that have potential relevance to the practice of usability. Method descriptions will generally be considered research entries in the BoK if their application in industry has been limited to research projects, singular case studies, or a minor variation on a generally accepted method.
methods (including all the COST294-MAUSE topics). 42 have extensive descriptions, 24 just a definition and references, and 43 have no description yet.

The Usability BoK currently uses the following categories for methods:

<table>
<thead>
<tr>
<th>BoK categories for methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Planning</td>
</tr>
<tr>
<td>1.2 Context of use</td>
</tr>
<tr>
<td>1.3 Requirements</td>
</tr>
<tr>
<td>1.4 Cognitive models</td>
</tr>
<tr>
<td>1.5 Task analysis and modeling</td>
</tr>
<tr>
<td>1.6 Design</td>
</tr>
<tr>
<td>1.7 Usability evaluation</td>
</tr>
<tr>
<td>1.7.1 Usability testing with users</td>
</tr>
<tr>
<td>1.7.2 Remote evaluation</td>
</tr>
<tr>
<td>1.7.3 Evaluation of data from usage of an existing system</td>
</tr>
<tr>
<td>1.7.4 Evaluation using models and simulation</td>
</tr>
<tr>
<td>1.7.5 Expert evaluation</td>
</tr>
<tr>
<td>1.7.6 Evaluation using computational and definitional approaches</td>
</tr>
<tr>
<td>1.7.7 Automated evaluation methods</td>
</tr>
<tr>
<td>1.7.8 Combined methods</td>
</tr>
<tr>
<td>1.7.9 Usability problem classification</td>
</tr>
<tr>
<td>1.8 Accessibility evaluation</td>
</tr>
</tbody>
</table>

The Usability BoK uses the following headings:

<table>
<thead>
<tr>
<th>BoK headings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Brief Description</td>
</tr>
<tr>
<td>2 Where to Find Out More</td>
</tr>
<tr>
<td>3 More Detailed Description and Information</td>
</tr>
<tr>
<td>3.1 Benefits, Advantages and Disadvantages</td>
</tr>
<tr>
<td>3.2 Cost-Effectiveness</td>
</tr>
<tr>
<td>3.3 Appropriate Uses</td>
</tr>
<tr>
<td>3.4 How to Do It</td>
</tr>
<tr>
<td>3.4.1 Procedure</td>
</tr>
<tr>
<td>3.4.2 Participants and Other Stakeholders</td>
</tr>
<tr>
<td>3.4.3 Materials Needed</td>
</tr>
<tr>
<td>3.4.4 Who Can Facilitate</td>
</tr>
<tr>
<td>3.4.5 Common Problems</td>
</tr>
<tr>
<td>3.4.6 Data Analysis Approach</td>
</tr>
<tr>
<td>3.4.7 Next Steps</td>
</tr>
<tr>
<td>3.5 Special Considerations</td>
</tr>
<tr>
<td>3.5.1 Costs and Scalability</td>
</tr>
<tr>
<td>3.5.2 Accessibility Considerations</td>
</tr>
<tr>
<td>3.5.3 International Considerations</td>
</tr>
<tr>
<td>3.5.4 Ethical and Legal Considerations</td>
</tr>
<tr>
<td>3.6 Read More About It</td>
</tr>
<tr>
<td>3.6.1 Originators/Popularizers</td>
</tr>
<tr>
<td>3.6.2 Authoritative References</td>
</tr>
<tr>
<td>3.6.3 Published Studies</td>
</tr>
<tr>
<td>3.6.4 Related Subjects</td>
</tr>
<tr>
<td>4 Contributors</td>
</tr>
</tbody>
</table>

39 MAUSE topics have been considered for incorporation to the BoK. 12 have been copied so far.
Limits of the BoK Contribution

This approach is indeed aiming towards usability practitioners, through their own web site, which should stimulate more reactions from industry. It also has the advantage of relating UEMs to the other issues of usability in practice (information on design, on organizations, on professional development, etc., and a glossary). From the WG1 perspective, it may have a few drawbacks:

- just like previous description/classification efforts, it relies on volunteers to fill up the descriptions, which has been difficult at times; even more so as it relies on a classification slightly different from the initial WG1 one
- some issues may need more clarification and debate within WG1, for instance: the classification structure; the assignment of methods to good practice vs. research; the single assignment of a method to a “lifecycle” step

- On the first issue, it may be interesting to discuss the level of consensus of the classification (and the method used); a number of items do not necessarily correspond to everyone's practice (e.g., context of use vs. needs analysis); others do not correspond to an activity or recognized distinction (e.g., formative vs. summative);
- On the second issue, the assignment can be debatable ... but it is a matter of how to properly establish which methods are really used (with some success) in industry (and predicting which ones may be used in the near future).
- On the third issue, the fact that many methods (e.g. task analyses, some evaluations) can be used in different steps, if not an issue for a "by nature" description", it is definitely an issue for a “by usage” description”.

In any case, this latter dissemination effort towards practitioners and the WG1 discussions should lead to enhanced approaches to the consolidation and delivery of UEM, which is discussed in the next section.

Future Work

Going back to the initial plan\(^5\), which shows topics relationships, and taking into account previous WG1 (and other WGs) contributions and limits, a few issues could be discussed to stimulate further work:

First of all, it has been quite useful to have a method description/classification by nature as it has been done initially in WG1 (and by WG4). This has been a useful instrument for dissemination and teaching. However, if selecting methods from a "nature" point of view has its advantages, it may not be sufficient in the context of selecting UEMs in the field.

Otherwise said, a parallel endeavor should also (it is a different goal) be to provide means of selecting methods from a "usage" point of view. This would also be in line with the latest efforts regarding the UPA BoK wiki. In an effort to lean more towards practitioners, one could envision designing a document (or other electronic tool) that would ideally support:

- a consensus on the system features and context of use for which the UEMs will be used
- an explicit relationship with a software life-cycle model or steps
- an explicit selection of methods with the attributes/criteria selected (this would include the methods described by WG4 (Review on the Computational and Definitional Approaches in Usability Evaluation), not been developed much in WG1: automatic/remote usability evaluation methods.

\(^5\) See Annex B for the initial MAUSE plan
- a consolidation of the methods descriptions with input from UEM assessment and usability problems characteristics. This would require further linking with other MAUSE WGs results, especially:

  o WG2 (Comparing UEMs: Strategies and Implementation) that is in charge of method assessment issues;
  o WG3 (Refining and Validating Classification Schemes for Usability Problems) that is in charge of usability problem attributes issues. A common identification of assessment attributes could be further discussed.

Such a contribution could also have a beneficial influence on usability standards, as being a potential reference for the revision of ISO 16982 (op. cit.) which is a TR supporting ISO 13407 (now ISO 9241-210 « Human-centred design processes for interactive systems»); its purpose is « ... to help project managers make informed decisions about the choice of usability methods to support human-centred design as described in ISO 13407 (with support from human-factors specialists, as appropriate) ».

In addition to its methods classification, and link to ISO 13407, it also provides an explicit reference to a lifecycle process (ISO/IEC 12207, Information technology - Software life cycle processes), and guidelines about which methods should be selected depending on various characteristics of the project being developed.

ISO 16982 being potentially revised, WG1 (and the other WGs, especially WG4) could provide suggestions for improvements particularly in the areas of the reference lifecycle model selected, the integration of more precision, and possibly more methods and links with recent standards (e.g.; CIF: System and software product Quality Requirements and Evaluation (SQuaRE) - Common industry Format for Usability - General framework for usability-related information).

Besides, considering the above new endeavor, one could also think from the start (and with the results of WG4) not only in terms of document delivery, but also in terms of software tools to help:
- selecting methods
- preparing evaluation studies
- analyzing problem situations
- usability reporting
- maintenance and follow up, facilitating long term studies, etc.

To sum up, the suggestion for further work would be to work on converging towards a method for usage classification through:

1. selecting project steps, e.g., analysis, design, evaluation, in some detail (selecting a full lifecycle process might be too high of a goal);
2. selecting which methods should be included (from a practical point of view, not just in theory) in those steps (this should concern results from WG4 as well as WG1)

Of course, this would probably need to consider:

(a) various characteristics of the projects for which the method(s) may be used, for instance application area, complexity, novelty, users availability, etc.;
(b) the results that can be expected (e.g., user/ task knowledge, usability problems types) ... including usability problems characteristics and quality attributes (link with WG3 results);
(c) the limits and levels of confidence of such results (link with WG2 results).
(d) and, obviously, distinguishing between problem diagnoses, problem classification, reporting, and re-design suggestions ...

Finally, as a follow-up on COST294-MAUSE project, in order to better answer industry needs concerning usability methods access, a review of the literature is being conducted on the studies that investigated the selection, use, and impact of usability methods. From this critical review, a set of meta
analyses of the initial data will be performed; also, an interactive questionnaire will be designed and offered on COST294-MAUSE website to cover some of the gaps identified, and to gather more widely (at least through Europe) additional data. Finally, as the issues cannot all be covered with questionnaire items, a set of interview guidelines will be designed in order to launch a set of interviews of senior managers from industry, in charge of human factors involvement in the software lifecycle process.

References

The numerous references relating to the WG1 activities can be found in (http://cost294.org/upload/522.pdf); http://www.usabilitybok.org/); and in http://www.cost294.org/
## Annex A: List of Contributors

<table>
<thead>
<tr>
<th>Name</th>
<th>Affiliation</th>
<th>Country</th>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carmelo Ardito</td>
<td>University of Bari, Italy</td>
<td>Italy</td>
<td><a href="mailto:ardito@di.uniba.it">ardito@di.uniba.it</a></td>
</tr>
<tr>
<td>Asbjørn Følstad</td>
<td>SINTEF, Norway</td>
<td>Norway</td>
<td><a href="mailto:Asbjorn.Folstad@sintef.no">Asbjorn.Folstad@sintef.no</a></td>
</tr>
<tr>
<td>Mathilde M. Bekker</td>
<td>University of Eindhoven, the Neth.</td>
<td>Netherlands</td>
<td><a href="mailto:M.M.Bekker@tue.nl">M.M.Bekker@tue.nl</a></td>
</tr>
<tr>
<td>Regina Bernhaupt</td>
<td>University of Salzburg, Austria</td>
<td>Austria</td>
<td><a href="mailto:Regina.Bernhaupt@sbg.ac.at">Regina.Bernhaupt@sbg.ac.at</a></td>
</tr>
<tr>
<td>Nigel Bevan</td>
<td>Professional Usability Services, UK</td>
<td>UK</td>
<td><a href="mailto:nigel@nigelbevan.com">nigel@nigelbevan.com</a></td>
</tr>
<tr>
<td>Ann Blandford</td>
<td>UCL Interaction Centre, UK</td>
<td>UK</td>
<td><a href="mailto:a.blandford@ucl.ac.uk">a.blandford@ucl.ac.uk</a></td>
</tr>
<tr>
<td>Sarah Burton-Talyor</td>
<td>the Web Usability Partnership Ltd,</td>
<td>UK</td>
<td><a href="mailto:sarah@webusability.co.uk">sarah@webusability.co.uk</a></td>
</tr>
<tr>
<td>Georgios Christou</td>
<td>European University Cyprus, Cyprus</td>
<td>Cyprus</td>
<td><a href="mailto:G.Christou@euc.ac.cy">G.Christou@euc.ac.cy</a></td>
</tr>
<tr>
<td>Gilbert Cockton</td>
<td>University of Sunderland, U.K</td>
<td>UK</td>
<td><a href="mailto:Gilbert.Cockton@sunderland.ac.uk">Gilbert.Cockton@sunderland.ac.uk</a></td>
</tr>
<tr>
<td>Maria Francesca Costabile</td>
<td>University of Bari, Italy</td>
<td>Italy</td>
<td><a href="mailto:costabile@di.uniba.it">costabile@di.uniba.it</a></td>
</tr>
<tr>
<td>Erik Frøkjaer</td>
<td>University of Copenhagen, Denmark</td>
<td>Denmark</td>
<td><a href="mailto:erikf@erikf.dk">erikf@erikf.dk</a></td>
</tr>
<tr>
<td>Petros Georgiakakis</td>
<td>University of Piraeus, Greece</td>
<td>Greece</td>
<td><a href="mailto:geopet@unipi.gr">geopet@unipi.gr</a></td>
</tr>
<tr>
<td>Jens Gerken</td>
<td>University of Konstanz, Germany</td>
<td>Germany</td>
<td><a href="mailto:gerken@inf.uni-konstanz.de">gerken@inf.uni-konstanz.de</a></td>
</tr>
<tr>
<td>Susan Harker</td>
<td>University of Loughborough, UK</td>
<td>UK</td>
<td><a href="mailto:S.D.Harker@lboro.ac.uk">S.D.Harker@lboro.ac.uk</a></td>
</tr>
<tr>
<td>Kasper Hornbæk</td>
<td>University of Copenhagen, Denmark</td>
<td>Denmark</td>
<td><a href="mailto:kash@diku.dk">kash@diku.dk</a></td>
</tr>
<tr>
<td>Timo Jokela</td>
<td>University of Oulu, Finland</td>
<td>Finland</td>
<td><a href="mailto:timo.jokela@oulu.fi">timo.jokela@oulu.fi</a></td>
</tr>
<tr>
<td>Niina Kantola</td>
<td>University of Oulu, Finland</td>
<td>Finland</td>
<td><a href="mailto:Niina.Kantola@oulu.fi">Niina.Kantola@oulu.fi</a></td>
</tr>
<tr>
<td>Suzette Keith</td>
<td>Middlesex University, UK</td>
<td>UK</td>
<td><a href="mailto:s.keith@mdx.ac.uk">s.keith@mdx.ac.uk</a></td>
</tr>
<tr>
<td>Rosa Lanzilotti</td>
<td>University of Bari, Italy</td>
<td>Italy</td>
<td><a href="mailto:lanzilotti@di.uniba.it">lanzilotti@di.uniba.it</a></td>
</tr>
<tr>
<td>Marta Larusdottir</td>
<td>Reykjavik University, Iceland</td>
<td>Iceland</td>
<td><a href="mailto:Marta@ru.is">Marta@ru.is</a></td>
</tr>
<tr>
<td>Darryn Lavery</td>
<td>Microsoft, USA</td>
<td>USA</td>
<td><a href="mailto:darryn@lavery.org">darryn@lavery.org</a></td>
</tr>
<tr>
<td>Effie Law</td>
<td>ETH Zürich, Switzerland</td>
<td>Switzerland</td>
<td><a href="mailto:law@tik.ee.ethz.ch">law@tik.ee.ethz.ch</a></td>
</tr>
<tr>
<td>Marianna Obrist</td>
<td>University of Salzburg, Austria</td>
<td>Austria</td>
<td><a href="mailto:Marianna.Obrist@sbg.ac.at">Marianna.Obrist@sbg.ac.at</a></td>
</tr>
<tr>
<td>Fabio Paterno</td>
<td>ISTI CNR, Pisa, Italy</td>
<td>Italy</td>
<td><a href="mailto:paterno@isti.cnr.it">paterno@isti.cnr.it</a></td>
</tr>
<tr>
<td>Costin Pribeanu</td>
<td>ICI Bucarest, Romania</td>
<td>Romania</td>
<td><a href="mailto:pribeanu@ici.ro">pribeanu@ici.ro</a></td>
</tr>
<tr>
<td>Yannis Psaromiligkos</td>
<td>University of Piraeus, Greece</td>
<td>Greece</td>
<td><a href="mailto:jpsa@teipir.gr">jpsa@teipir.gr</a></td>
</tr>
<tr>
<td>Carmen Santoro</td>
<td>ISTI, Pisa Italy</td>
<td>Italy</td>
<td><a href="mailto:carmen.santoro@isti.cnr.it">carmen.santoro@isti.cnr.it</a></td>
</tr>
<tr>
<td>Dominique Scapin</td>
<td>INRIA, France</td>
<td>France</td>
<td><a href="mailto:Dominique.Scapin@inria.fr">Dominique.Scapin@inria.fr</a></td>
</tr>
<tr>
<td>Marcin Sikorski</td>
<td>Gdańsk University of Technology, Poland</td>
<td><a href="mailto:msik@zie.pg.gda.pl">msik@zie.pg.gda.pl</a></td>
<td></td>
</tr>
<tr>
<td>Mark Springett</td>
<td>Middlesex University, U.K.</td>
<td>UK</td>
<td><a href="mailto:M.Springett@mdx.ac.uk">M.Springett@mdx.ac.uk</a></td>
</tr>
<tr>
<td>Jan Stage</td>
<td>Aalborg University, Denmark</td>
<td>Denmark</td>
<td><a href="mailto:jans@cs.aau.dk">jans@cs.aau.dk</a></td>
</tr>
<tr>
<td>Christian Stary</td>
<td>University of Linz, Austria</td>
<td>Austria</td>
<td><a href="mailto:christian.stary@jku.at">christian.stary@jku.at</a></td>
</tr>
<tr>
<td>Retalis Symeon</td>
<td>University of Piraeus, Greece</td>
<td>Greece</td>
<td><a href="mailto:retal@unipi.gr">retal@unipi.gr</a></td>
</tr>
<tr>
<td>Jacek Wachowicz</td>
<td>Gdańsk University of Technology, Poland</td>
<td><a href="mailto:jwach@zie.pg.gda.pl">jwach@zie.pg.gda.pl</a></td>
<td></td>
</tr>
<tr>
<td>Rebecca Whitfield</td>
<td>the Web Usability Partnership Ltd,</td>
<td>UK</td>
<td><a href="mailto:Rebecca@webusability.co.uk">Rebecca@webusability.co.uk</a></td>
</tr>
</tbody>
</table>
Annex B: COST294-MAUSE Initial Plan

Initial fancy Plan

WG1 Database management Individual UEMs

Meta-data definitions

UEM studies

WG2 Comparative Studies

Meta-analysis Multi-site Experiments

Comparison criteria

UEM studies

MAUSE Database

Quality models

WG3 Defect Classification Scheme (DCS)

DCS studies Multi-site Experiments

UEM studies

WG4 Formalized concepts and models

Project Team A
Project Team B
Project Team C

Project Team D
Project Team E
Project Team F

17
Comparing Usability Evaluation Methods: Strategies and Implementation
Final Report of COST294-MAUSE Working Group 2

Gilbert Cockton
Leader, COST294-MAUSE Working Group 2

Alan Woolrych
COST294-MAUSE Member,
Independent Analyst for CODELIGHTS MSE
MAUSE WG2 Wiki Management

University of Sunderland, England
Abstract

COST Action 294 focused on the Maturity of Information Technology Usability Evaluation (MAUSE). Within the MAUSE project, Working Group (WG) 2 was charged with Comparing UEMs: Strategies and Implementation. Aware of the many methodological challenges in this area, WG2 began by considering how credible multi-site experiments (MSEs) could be designed. To do this, we began with the CODELIGHTS MSE, where members from France to Rumania and from Iceland to Italy contributed evaluation data, reports and coding constructs that were used collaboratively in two workshops during the course of the MAUSE project. The approach taken was to first gather a collection of constructs that were being used or considered to compare usability evaluation methods (UEMs), for example, on the basis of the severity of the problems that they could find, or the value to software development of their discoveries, explanations and recommendations. Such constructs shine light on differences and similarities between UEMs through a process of categorisation, often referred to as coding in qualitative human science methodologies. The WG2 CODELIGHTS MSE thus sought to illuminate UEM comparison through the research activity of coding qualitative data sets for usability evaluations.

A collection of constructs was formed over two meetings, and applied collaboratively in coding exercises to a collection of evaluation problem and summary reports at two further meetings. Insights and issues from these collaborative exercises fed into a final independent analysis by the co-author of this report.

This report explores the motivations behind the work of MAUSE WG2, its relation to other MAUSE WGs (WG1: Critical Review and Analysis of Individual UEMs) and (WG4: Review on the Computational and Definitional Approaches in Usability Evaluation), and the results of group and individual activities (i.e., the implementation aspect of Comparing UEMs). It closes with a discussion of future strategies that would avoid direct comparisons, but instead take a meta-review approach to in depth case studies of usability work. The proposal is that case studies of usability work within full software development contexts focus on the interactions between approaches to usability (A2Us, a looser and broader concept than UEMs). Meta-reviews would then derive models from these case studies. These models of usability work, and the circumstances under which it can be effective, would provide a focused basis for future experimental work that would investigate the impact of specific focused interventions below the level of A2Us. This would greatly reduce the complexity of experiments relative to direct comparisons of UEMs, where the supposedly independent variables (i.e., the UEMs) are too ‘reactive’ and multi-faceted to allow credible comparisons of their role in effective usability work. Future research thus needs to investigate how the ingredients of A2Us (e.g., elements within the broad activity of user testing such as think aloud or participant selection) contribute to valuable outcomes of evaluation and iteration within software development. On this basis, WG2 has made fundamental contributions to advancing the maturity of usability work, rather than just usability evaluation.
Table of Contents

Introduction: Surface Codelights and Deeper Insights ..................................................22
Comparing UEMs ...........................................................................................................24
Constructs and the Coding of Usability Data .................................................................24
Problem Sets ..................................................................................................................28

Approach and Development ..........................................................................................28

Secondary Analysis: Comparability of Problem Sets ..................................................29
   Problem Sets used in Secondary Analysis .................................................................29
   Constructs used in Secondary Analysis ..................................................................31
   Secondary Analysis Results ....................................................................................32

Implications for UEM Comparison ..............................................................................37
Comparing Evaluations .................................................................................................38
Supporting Practice through Future Research ...........................................................39
A New Methodological Framework for Understanding Usability Work and Issues ....42
Conclusions ...................................................................................................................45
References .......................................................................................................................46

Appendix A: Coding Constructs Used in Secondary Analysis ......................................48
   A4 Discoverability ......................................................................................................48
   A7 Distributed Cognitive Resources (DCRs) ............................................................52
   A9a+b Persuasiveness ...............................................................................................53
   A9d Value to Development ......................................................................................54
   S2 Clarity ....................................................................................................................55
   CUP2 Cause ..............................................................................................................55
   CUP3 Context ............................................................................................................55
   CUP6 Frequency .......................................................................................................56
   CUP9 Impact .............................................................................................................56

Appendix B: Other Comparative Coding Constructs ....................................................57
   A1: Coding by User Action Framework ..................................................................57
   A2a Persistency .........................................................................................................58
   A2b Prioritization ......................................................................................................58
   A3: Effort per Problem .............................................................................................58
   A5 Appropriateness ..................................................................................................59
   A8 Evaluator Satisfaction .........................................................................................61
   A9c: Solution proposal ..............................................................................................62
   S1 Filtering .................................................................................................................62
   S3 Complexity of Redesign Proposal ......................................................................62
   CUP1 Actual Phase ...................................................................................................63
   CUP 4 Defect Detection Activity .............................................................................63
   CUP 5 Error Prevention Technique .........................................................................64
   CUP 7 Expected Phase .............................................................................................64
   CUP 8 Failure Qualifier ...........................................................................................64
CUP 10 Trigger ............................................................................................................................................. 65
CUP 11 Type of Fault Removed .................................................................................................................. 65
Appendix C: Two Examples of Problem Sets Used .................................................................................. 66
  List of Unique Usability Problems Identified in Four Language Versions of EducaNext Portal (2004) ........................................................................................................ 66
  Usability Problems for Travel Planner Web Site ..................................................................................... 79
Introduction: Surface Codelights and Deeper Insights

The primary initial objective of COST Action 294 (MAUSE) Working Group (WG) 2 was to identify effective strategies to compare Usability Evaluation Methods (UEMs), including empirical, analytic and model-based UEMs. Two key questions arise from this objective:

1. What criteria do we use to compare UEMs?
2. How do we compare UEMs, rather than the wider circumstances of their use? Can we isolate ‘method effects’ by neutralising confounds due to evaluators, projects, sponsors, developers and other stakeholder and environmental sources?

Note that the above are both questions, and it was not clear when the detailed proposal for MAUSE was written in 2004 what the answers would be. While many options existed as answers to the first question, the second question had two parts, with the first only relevant in the presence of a positive answer to the second part. While the second ‘Can’ question was written in some innocence, in general the answer as to whether such extensive confounds could be neutralised was very likely to be no, and even if extensive control were possible, there is the further practical issue of whether we should exercise such control, since the practical relevance and value of the resulting artificial and over-controlled studies is likely to be minimal.

MAUSE addressed the Maturity of Usability Evaluation via a better understanding of methods. Comparison was chosen as one approach to improving understanding. This has much superficial appeal, but we should also reflect on the purpose of making comparisons, in particular the practical benefits of comparisons to researchers and practitioners. In short, what was the purpose of WG2, and what did we hope to gain from the work here?

One potential benefit was to be able to provide better support for making method choices on the basis of UEM suitability. Here, we would enumerate the qualities and other relevant attributes of UEMs, supporting choice on the basis of the availability of user representatives, usability expertise, time resources, available budgets, stage of development and similar considerations. However, WG1’s efforts largely support such choice through its structured catalogue of UEMs. This clearly supports ‘first fit’, but not necessarily ‘best fit’ (although we know from algorithmic complexity research that first fit often outperforms best fit, especially when costs are offset against benefits). A second potential benefit, and one that brings much more excitement for those attracted by it, is to rank UEMs in terms of their performance. Such ranking pervades modern societies, and has been identified by Foucault as a means of punishment and reward (Foucault, 1991). While the comparison here with a work on the origins of the prison system may seem to be very obscure, it can alert us to the power of intellectual prisons. Not only may objective performance rankings of UEMs prove to be very difficult, but the very demand for them constructs developers and evaluators as uncritical followers of authority and/or fashion, more intent on being seen to choose the right UEM than actually using UEMs to best effect in unique development contexts.

WG2 thus began with objectives that were always open to criticism. However, as with many research endeavours, much can be learned in the attempt to address problems that could be argued to be unsolvable, suspect, or both. Much was indeed learned, through sharing of evaluation data and approaches to making comparisons. While the simple initial objectives have not been fully achieved, in the process of attempting to develop and test a methodology for UEM comparison, particularly in the extremely demanding area of downstream utility (Law et al., 2007), we have moved from high level criticisms of defining, characterising and ranking to a much richer understanding of how we should assess usability evaluation activities within the context of software development. For detailed studies in this area, see the recently completed PhD theses of three MAUSE participants, Dominic Furniss (UCL, London 2008), Mie Nørgaard and Tobias Uldall-Espersen (both DIKU, Copenhagen...
In some ways, the association of these PhDs with the MAUSE project is one of the strong outcomes of COST294, which has provided a unique context for young European usability researchers to develop innovative approaches to understanding usability work and issues that supersede simple minded approaches based on problem counting (Wixon, 2003). In this sense, WG2 has been a success, since the research of key European research groups has re-aligned around understanding usability work practices, and practical ways to improve

WG2 activities began with the proposal of the CODELIGHTS MSE (multi-site experiment) at the MAUSE Konstanz meeting in March 2006. It was called CODELIGHTS because of the ‘light’ that coding constructs ‘shine’ on UEM comparisons. Our ability to compare anything depends on the constructs on which we base comparisons. In the physical world, we base comparisons on volume or extent, weight, colour, form, rigidity and many more taken-for-granted constructs. When comparing UEMs, the constructs are less straightforward, and must be explicitly associated with data and information from usability evaluations (as physical constructs have to be closely associated to sense data). The primary association with data and information in qualitative research is via coding, whereby data is categorised to one or more classes, e.g., the severity of a usability problem. The CODELIGHTS MSE thus took its name from the comparative coding constructs that shone light on the data and information associated with usability evaluations.

During the remainder of 2006, WG2 members proposed a range of constructs for method comparison. Immediately it became apparent that MAUSE members were forming broad and forward looking approaches to the problem of UEM assessment and comparison. In 2007, these were applied to member supplied usability problem sets and evaluation reports. Constructs associated with problem discovery and characterisation were explored at the MAUSE meeting in Salzburg in June 2007. Constructs associated with downstream utility were explored at the MAUSE meeting in Toulouse in November 2007. These group exercises were followed up with selected secondary analyses by Alan Woolrych, with interim reports presented on behalf of WG2 at the MAUSE meetings in Bled (March 2008, by Gilbert Cockton), Reykjavik (June 2008, by Alan Woolrych) and Pisa (September 2008, by Ebba Hvannberg).

This report summarises the intellectual journey of MAUSE from an innocent proposal on comparative scoping and ranking of UEMs, attempting to make as many decisions in advance for inexperienced and/or pressurised usability workers, and thus putting them in a position to rush round the shelves of a usability methods supermarket and fill their shopping trolley with all the best buys (all indicated by clear and unambiguous labelling). During the course of this journey, the strong interest of several MAUSE members in downstream utility forced WG2 to take a much wider view on the data and information relevant to UEM comparison. Interestingly, much of this was already in place in some of the earliest data and information provided in 2006 in the form of consultancy reports.

The next section revisits the nature of UEM comparison in more depth. After that, we introduce a range of coding constructs proposed by MAUSE members for the comparison of UEMs, and then the problem sets and evaluation reports that provided the data and information for the practical group exercises in Salzburg and Toulouse. Following this, the overall approach of WG2 is described, and conclusions from the group exercises are presented. The report continues with a section on the secondary analysis. With these results in place, we move onto a series of closing sections that state the implications for UEM comparison, primarily the need to compare evaluations and not UEMs, due to the difficulties in separating method effects from evaluator and other context of use effects. The answer to the ‘How’ of the second question addressed by WG2 is thus that we cannot, i.e., we cannot isolate method effects, because UEMs themselves are too reactive and incomplete to be isolatable. Usability work is a complex set of interacting factors (Uldall-Espersen, 2007; Uldall-Espersen & Frøkjær, 2007), so we should accept that UEMs are only one highly reactive factor in evaluation outcomes, and their use is more shaped by their usage contexts than vice-versa. We next present considerations on supporting usability practices with alternatives to simple UEM characterisations, scopings and rankings, and close with considerations on feasible and worthwhile research.
methodologies for improving UEMs, as well as the next generation of approaches to user experience, value, values and worth in support of the development of interactive systems.

Comparing UEMs

How can we compare UEMs, and what do we mean by compare? One definition of compare is “to examine the character or qualities of especially in order to discover resemblances or differences” (Merriam-Webster online dictionary, http://www.merriam-webster.com/dictionary/compare). In order to develop strategies for comparison of UEMs, there is a need for criteria to establish and measure resemblances or differences.

Before considering UEMs, let us consider comparing an everyday object like a car. Probably the first comparison could be colour, one red, one blue car for example, other visual similarities could be the number of wheels and steering wheels (most cars have 4 road wheels – with some exceptions) and one steering wheel. But these are superficial comparisons, and do not tell us anything about the car’s performance such as fuel consumption, braking efficiency, speed and road handling. All of these are criteria for comparison. Methods for testing such criteria must be designed, such as rigorous road testing.

To treat UEMs the same way, we need to establish a set of relevant criteria against which we can measure and compare. A popular current philosophy on the role allocated to UEMs is that of the RITE method (Wixon, 2003). Wixon claims that key criteria for UEMs is not ‘how many problems it can find’ but understanding those that are found and fixing them. The latter is contentious, in that, there is a belief that ‘downstream utility’ is beyond the scope of pure evaluation (Cockton, 2005). However, we nevertheless consider these broad criteria for UEM assessment.

Criteria can be related to Wixon’s three questions from the RITE method (Wixon, 2003):

- What problems did we see?
  - Do various methods support sufficient problem description?
  - Is there evidence to show certain problems are within the scope of one method and not in others?
  - Is problem severity addressed?
- Can we explain them?
  - Is it in the scope of the method to attribute causes of the problem?
- Can we fix them?
  - Does the method suggest change, and a rationale for changes to ‘fix’ the problem?

Constructs and the Coding of Usability Data

Having established broad criteria to relate to Wixon’s three questions, the strategy of WG2 was to gather and refine a set of ‘coding’ constructs that could highlight key differences in UEMs. ‘Codelights’ was the collective name for the WG2 activities that developed coding ‘constructs’ and applied these collaboratively to UEM (problem) reports and to shine light on the factors that strongly influence UEM effectiveness.
Constructs are categories to compare, contrast and explain the differences/similarities in UEMs. In all, 28 individual constructs were proposed (details are in Appendices A and B). WG2 members ‘volunteered’ to develop individual constructs appropriate for pertinent UEM comparisons. The final list was developed through discussion in various workshops, and can be found by MAUSE members and others with access to the project wiki (Figure 2.1, access via www.cost294.org).

Each construct was defined and characterised using a common template, which evolved during 2006 and 2007. Below is an example construct for (Problem) Discoverability:

1. **Name of Dimension**: Discoverability
2. **Definition**: The complexity of description (number of steps/systems) of problem discovery (i.e., what did the inspector or test user do to find the problem).
3. **Operationalisation (what does the coder do?)**:
   - Describe the means of discovery as a task method, noting where users have to swap between applications to complete a task (e.g., between web browser and word processor)
   - Count the interaction steps and separate computer applications in this task method
   - Categorise discoverability using the fuzzy set {perceivable (0-1), actionable (1-4ish), constructable (over 3), multi-constructable (more than one application used)}
4. **Evidence requirements**:
   - Discovery narrative in extended report format for UEM problem sets
   - Video recordings or observer notes (user testing)
   - Logs files for routine usage
5. **Sources of bias**:
   - Incorrect self-reporting for UEMs
   - Gaps or inaccuracies in observer notes
   - Gaps in log files

The common template could be further extended, for example, with the knowledge and skill requirements for successful application of a construct. While Discoverability above only involves...
counting of task steps, application switching, and allocation of counts to overlapping fuzzy sets (e.g., 1-4ish), other constructs may require considerable theoretical knowledge and expertise to apply. Although apparently simple, the discoverability construct does require knowledge of the computer applications in a user’s context as well as expertise in task analysis, particularly in the choice of grain size, i.e., the level of abstraction applied to tasks from the keystroke to unit task levels (Card, Moran & Newell, 1983). It further requires good judgement in deciding on fuzzy set sizes (cardinality) and the allocation of problems to one of two overlapping sets. However, the impact of the latter on UEM assessment could be investigated via simple computer programs that computed outcomes for all possible assignments of problems to set categories. For example, a propensity to code problems as actionable for Heuristic Evaluation would confirm its inability to predict problems other than those with very simple causes (Cockton & Woolrych, 2001). Conversely, propensity to code problems as constructable would attribute greater predictive power to Heuristic Evaluation than most existing assessments attribute to it.

Constructs were given identification codes on the basis of their origins. A set of 8 constructs were proposed by MAUSE members at DIKU (Copenhagen), and progressively taken into consideration. An A prefix is used for constructs submitted before and accepted at the Athens Meeting (March 2007), with downstream utility constructs grouped around A9 (reflecting initial reluctance to consider these by the WG2 leader!). The S prefix indicates constructs added at the Salzburg Meeting (June 2007). The CUP prefix indicates constructs from (Vilbergdóttir, Hvannberg & Law 2006).
## Table 2.1: Comparative Coding Constructs Proposed for CODELIGHTS MSE

<table>
<thead>
<tr>
<th>Code</th>
<th>Construct Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>Phase of Interaction</td>
<td>Can be based on UAF or Norman’s Seven Stages of Interaction</td>
</tr>
<tr>
<td>A2a</td>
<td>Persistency</td>
<td>DIKU 5</td>
</tr>
<tr>
<td>A2b</td>
<td>Prioritization</td>
<td>An extended ‘severity’ construct, spanning CUP 6 and 9 below.</td>
</tr>
<tr>
<td>A3</td>
<td>Effort to find the problem</td>
<td>Time and/or other measures</td>
</tr>
<tr>
<td>A4</td>
<td>Discoverability</td>
<td>Used by Woolrych and colleagues (Cockton and Woolrych 2001, Cockton et al. 2003)</td>
</tr>
<tr>
<td>A5</td>
<td>Appropriateness</td>
<td>Used by Woolrych and colleagues (Cockton and Woolrych 2001, Cockton et al. 2003)</td>
</tr>
<tr>
<td>A7</td>
<td>Distributed Cognitive Resources</td>
<td>Used by Alan Woolrych and colleagues (Woolrych et al, 2005)</td>
</tr>
<tr>
<td>A8</td>
<td>Evaluator Satisfaction</td>
<td>Proposed by Rosa Lanzilotti and colleagues</td>
</tr>
<tr>
<td>A9a+b</td>
<td>Persuasiveness</td>
<td>Two versions: DIKU 6 and by Effie Law, related to John and Marks (1997) persuasive power</td>
</tr>
<tr>
<td>A9c</td>
<td>Solution Proposal</td>
<td>DIKU 7</td>
</tr>
<tr>
<td>A9d</td>
<td>Value to Development</td>
<td>Proposed by Effie Law, generalises Design Change Effectiveness (John and Marks 1997)</td>
</tr>
<tr>
<td>S1</td>
<td>Filtering</td>
<td>Proposed by Mark Springett, count of changes to problem report during development</td>
</tr>
<tr>
<td>S2</td>
<td>Clarity</td>
<td>DIKU 1</td>
</tr>
<tr>
<td>S3</td>
<td>Redesign complexity</td>
<td>DIKU 2</td>
</tr>
<tr>
<td>S4</td>
<td>Redesign Holism</td>
<td>DIKU 3</td>
</tr>
<tr>
<td>S5</td>
<td>Observable user action</td>
<td>DIKU 4</td>
</tr>
<tr>
<td>CUP 1</td>
<td>Actual Phase</td>
<td>Software Development Phase</td>
</tr>
<tr>
<td>CUP 2</td>
<td>Cause</td>
<td></td>
</tr>
<tr>
<td>CUP 3</td>
<td>Context</td>
<td></td>
</tr>
<tr>
<td>CUP 4</td>
<td>Defect Detection Activity</td>
<td>Similar to Discovery Method (Cockton et al. 2004)</td>
</tr>
<tr>
<td>CUP 5</td>
<td>Error Prevention Technique</td>
<td></td>
</tr>
<tr>
<td>CUP 6</td>
<td>Frequency</td>
<td></td>
</tr>
<tr>
<td>CUP 7</td>
<td>Expected Phase</td>
<td>Software Development Phase</td>
</tr>
<tr>
<td>CUP 8</td>
<td>Failure Qualifier</td>
<td></td>
</tr>
<tr>
<td>CUP 9</td>
<td>Impact</td>
<td></td>
</tr>
<tr>
<td>CUP 10</td>
<td>Trigger</td>
<td></td>
</tr>
<tr>
<td>CUP 11</td>
<td>Type of Fault Removed</td>
<td></td>
</tr>
</tbody>
</table>

The prefixes remain separate to reflect the main outcome of WG2, i.e., that we cannot compare UEMs, only evaluations, that is, the combined usability work within a development project. The CODELIGHTS constructs were instrumental in reaching this conclusion, and are shown in Table 2.1.

Not all comparative coding constructs made it to the MAUSE wiki with completed characterisations, e.g., five perspectives of usability were seen to have played a significant role in six different software development processes: (1) Interaction Object usability, (2) Task usability, (3) Product usability, (4) Context of Use usability, (5) Enterprise usability (Uldall-Espersen, 2007). These could be used for coding problem types to scope the effectiveness and focus of different UEMs. However, it was not
formally proposed and therefore was not added to the constructs under consideration. As it was, this set of constructs was already large, and thus there was no incentive to carry on adding constructs after first the Salzburg and then the Toulouse meetings.

**Problem Sets**

In order to explore how comparative coding constructs are applied in practice to evaluation data, example problem reports for a variety of methods in a range and languages were supplied by WG2 members. The range of reports covered both analytical methods (expert evaluation and heuristic evaluation, for example) and user testing (empirical). Figure 2.2 below is a screen shot of the MAUSE WG2 group wiki where members were able to submit problem sets. In all, 30 different problem report sets were supplied. Several of these are proprietary and commercially confidential, and were thus only accessible via the wiki exclusively for WG2 member use. Other than the screen shot below, there is no public listing of the problem sets and evaluation reports used within the CODELIGHTS activities.

![Figure 2.2: Listing of Sample Problem Sets (scrolls off) on the MAUSE wiki](image)

The problem sets not only covered a wide range of products, but were also completed in a variety of formats. Whilst some were in the form of a commercial report, others used formal report problem formats, including one developed by Woolrych (Cockton et al, 2003).

**Approach and Development**

As previously mentioned, the approach of WG2 was to apply a set of constructs to a series of problem and evaluation reports. The problem reports were supplied by project partners. The problem reports had a comprehensive coverage of methods including Heuristic Evaluation, Expert Evaluation and User Testing.

After collecting these problem reports, a consistent method for comparison needed to be developing. Constructs were developed by partners of the project as a means to compare the previously supplied...
problem reports. At the Salzburg (2007) workshop, WG2 arranged an exercise where partner groups applied constructs to problem reports.

The outcome was not conclusive, although it did generate some very positive debate. The areas that gave most concern were firstly, the overlap and clarity of the constructs. This led to uncertainty when applying a construct to various report elements. The second problem was that many constructs did not appear to be applicable for some problem reports, especially the simplest and most concise. While the existing templates could be improved, many of the problems in coding confidently were down to the format of the problem and evaluation reports.

It was clear that a secondary analysis was required to clarify these issues. This was undertaken after the Toulouse workshop (2007), which focused on downstream utility. It proved to be even harder to code the available problem sets for downstream utility constructs. Downstream utility refers to John and Marks (1997) constructs of persuasive power and design change effectiveness and to a UEM’s support for RITE Questions 2 and 3: Do we understand it? Can We Fix It? At this point, group work on problem set coding ended, but Alan Woolrych continued the analyses as an expert coder of problem sets and evaluation reports, based on extensive experience over 8 years of usability research (Cockton & Woolrych, 2001; Cockton et al., 2003; Cockton et al., 2004; Woolrych et al., 2005).

Secondary Analysis: Comparability of Problem Sets

In the secondary analysis, bearing in mind the issues from the Salzburg collaborative exercise, both problem reports and constructs were pre-filtered in order to maximise appropriate coverage. The rationale here was to maximise the potential for successful construct coding, and thus UEM comparison.

The issues from the Salzburg exercise specifically relate to the application of the constructs to the available problem sets. These issues were twofold. First of all, several constructs could not be coded against specific problem set formats, which simply lacked the relevant information. For example, the construct (CUP 7) in general terms, is concerned with identifying problems in a part of the development phase that could result in usability problems. From the available problem sets, application of this construct was not possible. We were guided here by experiences from the Salzburg and Toulouse workshops, where task groups reported back on the extent to which they could understand and apply comparative coding constructs.

Secondly, there were issues regarding which part of a report element specific constructs should be applied to. In reality, due to the diverse nature of the problem sets, a single construct could often be applied to several report elements. While this not inherently a problem, and might be overcome by expert coders, it does present difficulties for novice coders.

The secondary analysis built on these and other insights by assessing constructs for clarity and the potential for coverage in the problem sets. Thus a subset of constructs and problem sets/ reports was chosen to give UEM comparison its ‘best chance’. This included selection of two comparative coding constructs that had been developed by the independent analyst.

Problem Sets used in Secondary Analysis

The analysis of the problem sets resulted in eight problem sets for comparison. These were chosen primarily on language (English), but also on clarity of understanding (reducing potential analyst bias regarding report interpretation). Method coverage was also considered in problem set selection. Where possible, it was desirable to have the following information, but this was not available for all problem sets:
Appendix C contains the first two problem sets used in the independent secondary analysis reported here. The comparative coding constructs used in the secondary analysis are in Appendix A, allowing readers to try to apply the coding constructs to some usability problem reports to get a feel for the difficulties involved.

The following reports and problem sets were used in the secondary analysis:

1. **EducaNext System, Set of Usability Problems**
   This problem report set involves the user testing of a service supporting the creation and sharing of knowledge for Higher Education in four different language versions of the portal in 2004. The report contains 95 usability problems, and provides data relevant to possible method comparison on: context (where the problem occurred), problem descriptions, severity, and in some cases evaluator comments with suggestions for possible fixes. Note that the portal has been much extended and revised since 2004.

   **UEM(s) used:** heuristic evaluation and user-based think aloud

   **Number of Evaluators involved in the evaluation:** 2 evaluators for heuristic evaluation; 19 representative end-users were involved in think-aloud

   **Level of experience of the evaluators:** 2 evaluator experts in Heuristic Evaluation; 19 users were mostly novice in usability, but experienced in e-learning.

   **Additional information:** The user tests were multilingual - international tests conducted in four different sites in Europe with their respective local languages (i.e. Slovenian, Iceland, Swiss German, and English) and the results were translated.

2. **Travel Planner, Set of Usability Problems**
   This report set is a heuristic evaluation of www.infotec.be which is a travel planner for (part of the) public traffic in the Wallonian (French speaking) part of Belgium which contains 13 reported usability problems.

3. **e-commerce Application, Usability Problem Report**
   This is a single problem report example from an e-commerce website that sells male and female clothing and fashion accessories. The UEM used is Ergonomic Criteria (Scapin & Bastien, 1997).

4. **jobs.ok-cancel.com, Evaluation Report**
   An informal usability evaluation of a website where users are able to browse job adverts in the HCI community. The report lists 19 Recommendations/Observations of which 6 are positive and 13 are negative (presumably potential usability issues).

   **UEM used:** Expert review (described as an informal analysis in report)

   **Evaluators involved in the evaluation:** 1

   **Level of experience of the evaluator:** Novice at the time of reporting (2004).

   **Additional Information:** The product was a website that displayed HCI jobs. Its intended users were HCI students and professionals.
5. Catering Consultancy Company website, Evaluation Report
An evaluation report that involves a competitor analysis and a Heuristic Evaluation of a catering consultancy website. HE is mostly generalised comments with a few references to specific problems.

**UEM (s) used:** Competitor analysis, Heuristic Evaluation and Expert review (described as an informal analysis in report)

**Evaluators involved in the evaluation:** 1

**Level of experience of evaluators:** Novice at the time of reporting (2004).

**Additional Details:** The product was a website that summarized the profile and services of a catering consultancy company. The intended users of the website were potential customers of the consultancy. This would included a wide range of people with responsibility for catering services that wanted to procure, review, or improve their catering services. Typically these people would have management roles in local government, hospitals, education and businesses.

6. Discovery Space, Evaluation Report
An expert evaluation report that focuses on observations, recommendations and prioritisation of recommendations. For each, screenshots of the application are provided for support.

7. OWL learning management system, Set of Usability Problems
The problem set is the results of user testing on a learning management system for the University of Iceland with 11 participants, five teachers and six students. There are 71 usability problems in the report.

8. Travel NE, Set of Usability Problems
The problem report set is the result of a usability evaluation of a travel web site, local to the North East of England, where visitors to the site can access information about local public transport such as bus and metro timetables.

**UEM used:** Heuristic Evaluation

**Evaluators involved in the evaluation:** 3

**Level of experience of evaluators:** The 3 evaluators were novices in usability however, they did benefit from basic coaching in performing heuristic evaluations prior to the exercise.

**Additional Details:** The evaluations were carried out independently by each of the evaluators who collectively provided a total of 11 usability problem reports. Each problem was reported using an extended problem report format (Cockton et al., 2003).

### Constructs used in Secondary Analysis

As previously mentioned, due to varying degrees of overlap and appropriateness across all of the constructs, some pre-filtering was required before commencing secondary analysis. This was intended to eliminate constructs with potential ambiguous overlap. It was also a goal to identify the constructs used closely matched with the principles of the RITE method (understanding the problem and possible fix, Wixon, 2003).

Secondary analysis involved an attempt to map data from the reports to as many of the previously described ‘constructs’ as possible.
1. A4 Discoverability
2. A7 Distributed Cognitive Resources (DCRs)
3. A9a+b Persuasiveness
4. A9d Value to Development
5. S2 Clarity
6. CUP2 Cause
7. CUP3 Context
8. CUP6 Frequency
9. CUP9 Impact

Definitions and extended characterisations for these comparative coding constructs appear can be found in Appendix A. Note that CUP 2 Cause was only applied in the narrow technical sense of a design decision being the cause of a user difficulty. It was not coded using the broader quality process framework associated with CUP as illustrated in Appendix A.

Secondary Analysis Results

The problems in each of the eight reports were analysed for evidence of each of the constructs. The ‘matrix’ in Table 2.2 is colour coded to show where evidence of each construct exists or is absent, or indeed partially evident. The colour code is:

- Red – No evidence of construct
- Green – Evidence of construct
- Yellow – Partial evidence of construct.

The secondary analysis investigated how each of the problem sets ‘scored’ with regards to how many constructs could be confidently coded. Perhaps unsurprisingly, the four worst scoring problem sets come from analytical methods, whilst the only two empirically-based methods come in the top three (depending on weighting of partially addressed constructs). However, these results must be treated with caution, as a simple count of codable constructs takes no account of their relative priorities in terms of the value of the information that they reveal about UEMs.

Moreover, what must be fully understood is that whilst attempting to compare UEMs, in reality we are actually comparing results from UEM, indeed reports of results. The CODELIGHTS MSE thus explored the extent to which we could base comparisons of UEMs on results data and information from evaluations. The group exercises in Salzburg and Toulouse indicated that most of the problem sets and reports under consideration did not contain sufficient information to allow confident coding, and this was further exacerbated by a lack of familiarity with many of the constructs, most of which had been used by single research groups in specific research contexts. We could not conclude whether the content and context of problem sets and evaluation reports, or the meaningfulness and practicality of construct templates was primarily responsible for these difficulties. The secondary analysis sought to clarify this once it was established that few constructs could be confidently coded against all the problem sets and evaluation reports, bearing in mind that only the most promising subsets of coding constructs and report sets were considered during the initial secondary analysis.

After applying the constructs, each of the chosen problem sets were analysed to identify their basic reporting elements. The purpose of this part of the analysis was to establish the most common elements in the problem reports, as a basis for understanding coding difficulties:
### Table 2.2: Codability of Comparison Constructs against Problem Sets and Evaluation Reports

<table>
<thead>
<tr>
<th>Problem Reports</th>
<th>A4 Discoverability</th>
<th>A7 DCRs</th>
<th>A9 Persuasiveness</th>
<th>A9d Value to Development</th>
<th>S2 Clarity</th>
<th>CUP2 Cause</th>
<th>CUP3 Context</th>
<th>CUP6 Frequency</th>
<th>CUP9 Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>EducaNext System, Set, Effie Law</td>
<td>No – no description of task sets but may exist in any video</td>
<td>N/A</td>
<td>Partial – Not all suggestions adequately cover effect of problem</td>
<td>Yes – Place, description suggested fix</td>
<td>Partial – by some descriptions</td>
<td>Partial – by some descriptions</td>
<td>Yes</td>
<td>Yes – by test participants</td>
<td>Partial – Description of problem but not effect</td>
</tr>
<tr>
<td>Travel Planner, Set, Arnold Vermeeren</td>
<td>No</td>
<td>No - insufficient detail on discovery tactics and analysis</td>
<td>Partial – User difficulties a reported</td>
<td>Yes – by description but not all</td>
<td>Yes – by description and predicted difficulties</td>
<td>No</td>
<td>No – not specific, no examples</td>
<td>No</td>
<td>No – Partial – By description</td>
</tr>
<tr>
<td>e-commerce Application, Problem, Costin Pribeanu</td>
<td>Partial – by description</td>
<td>Partial – by description</td>
<td>No – insufficient detail on discovery tactics and analysis</td>
<td>Partial – by description</td>
<td>Yes – by description and screenshots</td>
<td>Partial – Cause can be understood from description</td>
<td>No</td>
<td>Yes</td>
<td>No – No – Predicts extra action but not impact</td>
</tr>
<tr>
<td>jobs.ok-cancel.com, Report, Dominic Furniss</td>
<td>No – Only reference is to ‘informal report’</td>
<td>No - insufficient detail on discovery tactics and analysis</td>
<td>No – vague descriptions of problems</td>
<td>Partial – by description</td>
<td>No – mostly explained by the vague problem descriptions</td>
<td>Partial – some descriptions refer to possible cause</td>
<td>No</td>
<td>Yes</td>
<td>No – No – Observations do not include impact</td>
</tr>
<tr>
<td>Catering Consultancy Company, Report, Dominic Furniss</td>
<td>No – mainly observation</td>
<td>No - insufficient detail on discovery tactics and analysis</td>
<td>No – Observational, no reporting of user difficulties</td>
<td>Partial – by some description</td>
<td>No – observations are generalised not focused on specific issues</td>
<td>No – due to generalisation</td>
<td>Partial – only a few specific issues</td>
<td>No</td>
<td>No – No – Observations do not include impact</td>
</tr>
<tr>
<td>Discovery Space, Report, Nigel Bevan</td>
<td>No – mainly observation</td>
<td>No - insufficient detail on discovery tactics and analysis</td>
<td>Yes – Observation and recommendation supported by screenshots</td>
<td>Yes – Place, description suggested fix</td>
<td>Yes – By screenshot and description</td>
<td>Yes – but open to evaluator misinterpretation</td>
<td>Yes – by screenshot</td>
<td>No</td>
<td>Partial – by priority recommendation</td>
</tr>
<tr>
<td>OWL learning management system, Set (contains traces of Icelandic), Ebba Þóra Hvannberg</td>
<td>No – no description of task sets but may exist in any video</td>
<td>N/A</td>
<td>Comprehensive detail on user difficulties</td>
<td>Yes – by user difficulty, description, frequency and severity rating</td>
<td>Yes – by description and user difficulties</td>
<td>No – no attempts to describe possible cause</td>
<td>Partial – by test phase</td>
<td>Yes – by participant</td>
<td>Yes – By task efficiency, and time on task criteria</td>
</tr>
<tr>
<td>Travel NE Alan Woolrych</td>
<td>Yes – By description</td>
<td>Partial – By problem impact analysis and discovery method</td>
<td>Yes – by comprehensive reporting of discovery and analysis</td>
<td>Yes – by thoroughness of context, cause and impact</td>
<td>Yes – by thoroughness of context, cause and impact reporting</td>
<td>Partial – cause not always reported</td>
<td>Partial – cause not always reported</td>
<td>No</td>
<td>Partial – By problem description/likely difficulties</td>
</tr>
</tbody>
</table>
1. Problem discovery explanation
   - Is there a coherent explanation of how the problem was discovered that may help explain the problem?

2. Context in which the problem occurs
   - Does the problem only occur in a given set of circumstances, i.e. in a specific context?

3. Problem description
   - Is there a description of the problem?

4. Evidence of problem severity
   - Is there any evidence of problem severity?

5. Predicted problem severity
   - In the absence of evidence of problem severity, is there an attempt by the evaluator(s) to predict problem severity?

6. Actual user difficulties
   - Is there any evidence of user difficulties?

7. Predicted user difficulties
   - In the absence of evidence of user difficulties, is there an attempt by the evaluator(s) to predict possible user difficulties?

8. Frequency of problem
   - Is there any evidence of problem frequency?

9. Suggested fixes for problems
   - Is there evidence of suggested fixes?

The reporting elements were drawn from a variety of sources. Elements 2, 3 and 7 were taken from a structured problem report format (Cockton & Woolrych, 2001) that was developed to aid problem matching for UEM comparison. Element 1 was taken from an extended structured problem report format developed as a research tool to increase understanding of analyst performance in usability evaluation (Cockton et.al., 2004).

Elements 4, 5, 6, 8 and 9 were included to enhance understanding of the problem. Whilst empirical testing may have evidence of problems severity, analytical evaluation can only have assumed problem severity. Element 6 is an empirical variant of Element 7. Following discussions in both Konstanz and Athens workshops, elements 4, 5, 8 and 9 were seen as important elements that were missing from the Sunderland problem report formats (Cockton & Woolrych, 2001; Cockton et al., 2004). Frequency and severity elements are critical to problem prioritization and increasing the value to development of UEMs. Element 9 corresponds to the third RITE question (Can we fix it?). Also, Hornbæk and F rokjær (2006) argued that suggestion for fixes increased developers’ perception of the utility of a problem description. We thus selected a representative set of elements from existing UEM comparison studies.

Moreover, it was important to adopt reporting elements that the chosen constructs can be appropriately ‘mapped onto’. Table 2.3 summarises the constructs that can be coded against the reporting elements that could be found in some (and, very occasionally, the entire problem sets in the secondary analysis).
Table 2.3: Distribution of Elements across Problem and Evaluation Reports

<table>
<thead>
<tr>
<th></th>
<th>Problem Element</th>
<th>Construct</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Problem discovery explanation</td>
<td>A4, A7</td>
</tr>
<tr>
<td>2</td>
<td>Context in which the problem occurs</td>
<td>CUP3</td>
</tr>
<tr>
<td>3</td>
<td>Problem description</td>
<td>A4, A7</td>
</tr>
<tr>
<td>4</td>
<td>Evidence of problem severity</td>
<td>CUP9</td>
</tr>
<tr>
<td>5</td>
<td>Predicted problem severity</td>
<td>A9a+b</td>
</tr>
<tr>
<td>6</td>
<td>Actual user difficulties</td>
<td>CUP2, CUP3</td>
</tr>
<tr>
<td>7</td>
<td>Predicted user difficulties</td>
<td>CUP2, CUP3</td>
</tr>
<tr>
<td>8</td>
<td>Frequency of problem</td>
<td>CUP6</td>
</tr>
<tr>
<td>9</td>
<td>Suggested fixes for problems</td>
<td>A9d, CUP2</td>
</tr>
</tbody>
</table>

It is important to highlight at this point that it is not claimed that the reporting elements or the constructs are in any way are definitive criteria for comparing UEMs. Rather, they were the comparative coding constructs that were deemed to be most appropriate for the secondary analysis in the sense that we expected all of them to be codable against at least one of the chosen problem sets or evaluation reports.

Table 2.4 relates the problem report and UEM(s) used in the evaluation to the presence/absence of the basic reporting elements. Already, it is apparent that there is limited scope for comparison of UEMs on the basis of available problem sets and evaluation reports. Very few codable constructs are common to all problem sets. In fact (referring to Table 2.3), the only element common to all of the problem sets is the third, i.e., the problem description. The problem of comparison is further compounded by potential evaluator bias in all of the analytical methods, and it would also be somewhat generous to ignore potential evaluator bias in the interpretation of problems in user testing. The conclusion thus has to be that there is insufficient ‘common ground’ across our collected problem and evaluation reports to make credible comparisons between UEMs to isolate ‘method effects’. It is on this basis that the second question for WG2 cannot be answered, because we cannot isolate ‘method effects’ by neutralising confounds due to evaluators, projects, sponsors, developers and other stakeholder and environmental sources.

Methodologically, it can be seen that to make credible comparisons between UEMs, we must:

1. Use common reporting formats for problems and evaluation process data, i.e., evaluators using different UEMs in a comparative study must use common reporting formats (Lavery & Cockton 1997).
2. Reporting formats must contain elements that are compatible with the coding constructs on which comparisons of UEMs will be based (Cockton et al. 2004).
3. Comparison constructs must be well enough defined to allow consistent and efficient coding by UEM researchers.
Table 2.4: Distribution of Elements across Problem and Evaluation Reports

<table>
<thead>
<tr>
<th>No</th>
<th>Problem Reports</th>
<th>Method</th>
<th>Discovery</th>
<th>Context</th>
<th>Description</th>
<th>Severity Actual</th>
<th>Severity Predicted</th>
<th>User Difficulties Actual</th>
<th>User Difficulties Predicted</th>
<th>Frequency</th>
<th>Fix Suggestion</th>
<th>Evaluator Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EducaNext System, Set, Effie Law</td>
<td>UT</td>
<td>✘</td>
<td>✓</td>
<td>✓</td>
<td>✘</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>2</td>
<td>Travel Planner, Set, Arnold Vermeeren</td>
<td>HE</td>
<td>✘</td>
<td>✓</td>
<td>✓</td>
<td>✘</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>3</td>
<td>e-commerce Application, Problem, Costin Pribeanu</td>
<td>EC</td>
<td>✘</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>4</td>
<td>jobs.ok-cancel.com, Report, Dominic Furniss</td>
<td>EE</td>
<td>✘</td>
<td>✘</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>5</td>
<td>Catering Consultancy Company, Report, Dominic Furniss</td>
<td>HE</td>
<td>✘</td>
<td>✘</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>6</td>
<td>Discovery Space, Report, Nigel Bevan</td>
<td>EE</td>
<td>✘</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>7</td>
<td>OWL learning management system, Set (contains traces of Icelandic), Ebba Bóra Hvannberg</td>
<td>UT</td>
<td>✘</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>8</td>
<td>Travel NE, Alan Woolrych</td>
<td>HE</td>
<td>✓</td>
<td>✘</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
</tbody>
</table>

**Key**


✓ = Basic Element Present  
✘ = Basic Element Not present
Thus to have worked as WG2 had originally planned, and move from strategies to implementation, the above strategy would have had to be applied when implementing further multi-site experiments on method comparison. This may have been possible by restricting the research to how UEMs on finding problems (RITE Question 1), but MAUSE members were, quite rightly given Wixon’s (2003) position, more interested in RITE Questions 2 and 3, i.e., understanding and fixing problems. This would have required development of a common evaluation report format, i.e., not just for the reporting of individual usability problems, but for the whole evaluation process, especially the ‘fortunes’ of problems as they moved from being discovered to being understood and fixed. Also, since UEMs alone cannot be the sole determining factor of the success of usability work, all other relevant independent variables would have to be captured via a standard comprehensive report format for complete evaluation and iteration processes. Once such a standard report was developed, we could implement UEM comparisons to cover all three RITE questions.

Developing and applying such as comprehensive standard report format for evaluation and iteration was an absolutely unrealistic prospect. Work following the secondary analysis has thus focused on exploring alternative strategies for understanding usability work, rather than simply comparing UEMs, which is, to be blunt, a hopeless endeavour. Even so, it was an exercise that was well suited to a COST action, as we could quickly and efficiently gain insights and move our research field forward through the collaboration of researchers across Europe, who could readily provide the comparative constructs and problems sets and evaluation reports that we needed to be able to demonstrate the effective impossibility of controlled experiments with UEMs as the primary independent variable.

Implications for UEM Comparison

Credible comparison of UEMs is difficult, not least due to the inconsistencies in details across problem reports. Therein lies the problem. We can only compare reports, which often give little or no real insight into how the evaluation was carried out. Moreover, for downstream utility constructs, where we are interested in comparing the persuasive power and design change effectiveness (John and Marks 1997) arising from UEM usage, problem formats alone cannot provide the data and information required for confident construct coding, and thus credible comparison of UEMs. To compare downstream utility, we need information beyond that found even in the most complete and well structured of problem report sets (and these, we must stress are very rare outside of research on problem reporting!). This was the main outcome of the Toulouse WG2 activities, i.e., that to compare the downstream utility of UEMs, we must keep track of far more information than that contained in a set of usability problem reports.

Looking back at the car analogy, the performance of a car is confirmed by strict testing where conditions are constant – on a ‘rolling road’ for example. There is no room for bias, unless of course, if the testing was to be taken out of the ‘clinical’ environment and performed with a human element and associated attributes such as reaction times, and human performance during the test.

Evaluator bias (in both analytical and empirical methods) impacts the validity of all attempts to compare UEMs. As a consequence, direct comparison is of UEMs (as opposed to evaluation episodes) is difficult. For example, Evaluator A on a good day will produce very different results to Evaluator B on a bad day. Furthermore, even on a good day for both evaluators using the same UEM, but in different evaluation contexts, the outcome of each evaluation will obviously be different, specifically since one key difference in evaluation contexts is the system being evaluated. In such circumstances, it is hard to attribute specific aspects of outcomes to evaluators, UEMs and evaluation contexts respectively. We can identify the inputs and review the outputs, but the causal interactions between these cannot be readily unpicked. Note that this situation holds even for the narrow understanding of method comparison and assessment criticised by Wixon (2003). Even when comparisons are limited to the problems ‘found’ by a UEM (rather than ones understood and fixed by it), extensive report formats are needed to let us investigate even simple explanatory models such as the DARe model (Cockton et al., 2004). Once we attempt to compare across all three of the RITE
questions (Wixon, 2003: Is it a problem? Do we understand it? Can we fix it?), we would have to
instrument evaluations in a way that could make viable scientific comparisons impossible. Such
instrumentation could be unbearable in practical settings. In research settings, a lack of real product
and market drivers, plus the use of academic or student evaluators with limited professional
experience, could well invalidate generalisation to real world contexts. This is supported by one report
in the secondary analysis that was intended for commercial use, which scored joint highest for
persuasiveness and clarity, which may well reflect the evaluator’s motivation to communicate their
expertise to the benefit of their client. In the absence of real clients, customers, project sponsors and
other relevant stakeholders, reporting elements may lose strength of motivation.

Comparing Evaluations

We can compare results from UEM usages, but not UEMs in isolation independently of their usage.
Evaluation results reflect not only the UEM(s) used, but evaluator performance and a wide range of
contextual factors. This became clear during the Toulouse workshop. While we were focused, as at
Salzburg, on the sort of problems that a UEM could discover, it remained plausible that, with the right
problem formats (as patchily illustrated by the sample format used in the secondary analysis), well
presented coding constructs, and well trained researchers, we could achieve objective comparisons of
UEMs. Even here however, we would have been really comparing evaluations using problem sets as
our main data. It would only be possible to isolate the effect of compared UEMs by holding all other
factors constant. Given the complexity of problem discovery alone, never mind problem analysis and
redesign suggestions, it is effectively impossible to carry out a scientific experiment here by
manipulating single independent UEM variables while holding all others constant. Worse still, there
may be no such thing as a single independent UEM variable.

As a simple analogy here, consider the task of comparing coffee beans. While a skilled coffee taster
establish can much by biting through a roasted bean, most of us are unable to judge what a cup of
coffee would taste like from a bean alone. Even highly experienced coffee tasters would never rely
wholly on their judgement of beans alone, particularly when blending. At this point a wide range of
variables come into play when tasting a cup of coffee rather than a single bean. Beans need to be
blended, ground, immersed in hot water, dispensed and drunk. At each point, variations in the
preparation of the coffee will impact taste in ways that cannot be attributed to the coffee beans alone.
However, as all the processes here are physical, they can all be systematically and consistently
controlled, e.g., through the use of distilled water, pouring boiled water at a fixed temperature each
time, and use of standard tasting cups. Even so, all this does is let the professional coffee taster
compare blends under fixed artificial conditions. The more the preparation for tasting is controlled,
the less likely it is to accurately predict how the coffee will taste in people’s homes, or in restaurants
or other places where coffee is sold. For some drinks such as wine, close attention is sometimes paid
to the retail contexts (filtering out tartrates from wines sold from brightly lit shelving) or consumer
contexts (e.g., taking great care with the choice of wines for business class airline passengers, since
the taste of wine is adversely affected by cool pressurised cabins). The same applies to comparing
UEMs. We can either only objectively compare them in fixed artificial contexts, as with coffee
tasting, or we must restrict ourselves to seeing how they perform in specific real world contexts (as
with business and first class wine lists for airlines).

We thus have to trade off scientific objectivity against practical relevance. The more we control (if
indeed we fully can), the less likely we are to produce results that will generalise to the real worlds of
usability work. We certainly will not be able to provide the sort of ‘best fit’ or ‘will fit’ advice that
motivated the work of WG2. Given enough resources, we could plan and carry out a very good
scientific experiment, but it would have very limited value beyond its scientific context. Conversely,
if we study UEM use in context, as seen in recent PhD research by MAUSE action members
(Dominic Furniss, Mie Nørgaard and Tobias Uldall-Espersen), then we cannot control factors enough
to establish the relative contributions of UEMs, evaluators or the many other factors that shape how
usability work is carried out in practice. All we see is outcomes, and not the specific causal interactions that led to them. The latter have to be painstakingly inferred from the available data.

## Supporting Practice through Future Research

The initial objectives for WG2 made some implicit assumptions about the nature of advice, guidance and information needed by usability specialists, including software developers with responsibilities for interactive features. It implicitly assumed that they wanted authoritative and definitive knowledge on UEMs, in a form that presumed invariance of UEMs across applications. It further implicitly assumed that usability specialists wanted to be lead by scientific experts, indicating to them which methods are best for which applications, and how methods that are similar in terms of suitability differ in terms of performance. Such implicit assumptions are misguided on two key points.

Firstly, usability work, even when carried out by software generalists, is typically carried out by highly educated individuals, often with postgraduate qualifications. Where individuals have arrived in software development by non-standard routes, their success in systems development indicates a high level of intelligence. Such individuals are used to exercising independent judgement and expect to adapt methods and approaches to fit local circumstances. It is presumptuous to assume that such people would accept that, where both Heuristic Evaluation and Cognitive Walkthrough are appropriate, then Heuristic Evaluation will always be better. It should be clear for anyone with experience of using ‘methods’ that it’s not the UEM alone that has the most impact, but how they are used (Cockton & Woolrych, 2001; Cockton et al. 2003; Hornbæk 2009). Heuristic Evaluation could be used so badly, e.g., as individual prejudice fueled by laziness, that any basic use of Cognitive Walkthrough would outperform it. Similarly, Cognitive Walkthrough, when used by a master of usability practice in ways to remove its weaknesses and to play efficiently to its strengths, could well outperform average usability specialists making unimaginative use of Heuristic Evaluation. To suggest otherwise is like suggesting that coffee beans automatically turn themselves into coffee. Just as good coffee has to be well made, and can be made in different ways, so UEMs, like all methods, have to be applied, and they will be applied by different people in different ways in different circumstances.

The word ‘method’ will seduce the unwary into thinking that human behaviour, in this case highly skilled creative knowledge intensive work, can be completely constrained and programmed by a method. Developers who submit to a method’s discipline are thus guaranteed results. Thus we could be led into thinking that we could objectively compare UEMs in so far as they are being correctly applied. However, this too would be an absolute misconception. While it is clear that methods can be incorrectly applied, this does not automatically lead to poor outcomes. In fact, Cockton and Woolrych (2001) discovered that Heuristic Evaluation is best applied when it is least needed (i.e., to find low frequency and/or severity problems). In contrast, severe high frequency problems are often so obvious that the relevant heuristic is irrelevant, if it actually exists that is.

Pulling the above together, we simply cannot support practice with a raft of misconceptions. Primary amongst them all is the misconception that there is value in comparing UEMs independently of their real world usage. Since this is impossible, it can’t possibly have any value, indeed it has had a negative impact on progress in supporting usability practice.

UEMs are one component of usability practice. They are absolutely parasitical in that they cannot apply themselves, and in a sense have no concrete existence until they are applied. Until applied, they are not so much methods as approaches, a collection of knowledge elements that may combine one of more of knowledge resources, diagrammatic or textual representations, principles, tactics or specific practices, and combine them in ways that suggest possible activities to usability specialists. UEMs are like coffee beans. To make coffee, beans need to be prepared (i.e., ground) and augmented (i.e., with hot water and perhaps milk/cream, sugar and other additives). The same is true of UEMs. Usability specialists have to decide how to use them (i.e., prepare them) and then they need to add sufficient
detail through practical usage to deliver results (i.e., augment them). Just like coffee, once the drink has been made, only the most expert tasters can work out what went into it, and what role each ingredient played. The same is true of evaluations. Once evaluation results are available, only an expert in usability practice can begin to unpick how any UEMs used combined with evaluators and the development context to produce specific outcomes. Such outcomes are not inevitably determined through something so abstract, insubstantial and incomplete as a choice of methods.

The best way to support practice is to stop talking of methods at all, and instead see each approach for what it is, that is, an incomplete set of quite different components (e.g., notations, guidelines, preferences, processes) in different balances that, like the ingredients of a cup of coffee, have to be skillfully prepared and combined to make a wonderful cup of coffee. In the hands of the slovenly and ignorant, the most vile cups of coffee can and do result.

Research in support of usability work thus needs to drop below the level of mythical methods to what is actually proposed by those who develop and advocate specific approaches. Such research needs to be able to identify the ingredients of approaches that are combined uniquely and contextually into methods that, like a river, are never the same when you return to them the next time. Every method application is unique and unrepeatable. Just as you can never enter the same river twice, you can’t use exactly the same method twice either, at least, not without risk of considerable loss of value to software development. Even then, the idea of it being the same method will be an illusion, since choices such as tasks in focus (for testing or inspection), participant briefing, problem prioritization or problem acceptance criteria must differ from one evaluation to the next, even when re-testing or re-inspecting the same application after a remedial design iteration. What are not unique are the ingredients of approaches, and the categories of contextual factors that shape usability. These we can generalise about, and we can also look for patterns of interactions between these ingredients and other factors in usability work. Indeed, Dominic Furniss' PhD (Furniss 2008) attempts to recognise some of these factors through a grounded and systemic view of usability work.

As often noted during the MAUSE action, there were often overlaps between WG1 (Critical Review and Analysis of Individual UEMs) and WG2 (Comparing UEMs: Strategies and Implementation). WG1 (COST294-MAUSE Working Group 1 Final Report, this volume) clearly relies on constructs to support the critical review and analysis of individual UEMs. While in practice it applied these constructs to published scientific reports of UEM usage rather than the primary evidence of evaluation and problem reports, there could and should have been a strong overlap between the constructs used in WG1 and WG2. However, WG1 took a taxonomic approach that largely ignored how existing UEM comparisons had operated, especially the constructs that supported critique and analysis. Much of the templates used in WG1 are descriptive, and do not support productive comparisons of UEMs. However, they do have substantial educational value in communicating a wide range of UEMs in a common format, as demonstrated by the synergies created by MAUSE WG1 with the UPA Body of Knowledge project. However, the more critical elements of the template are essentially ‘headings’, which different UEM reviewers have used in different ways, and these ‘ways’ rarely correspond to the comparative coding constructs used in the CODELIGHTS review.

Had WG1 been able to gather more case studies (as was very much hoped for), a different picture may well have emerged, but even here there is no overlap between the proposed comparative attributes from ISO TR 16982 and the constructs proposed by WG2. However, had WG2 been able to progress to comparing evaluations, then some of the WG1 case study constructs (comparative attributes) may well have become relevant.

Future research thus needs to focus on usability work and issues, and not on usability methods and problems. Usability work needs to be seen as contextual instantiation of idealised approaches. There is a limit to how much we can be specific about prior to the use of a method or approach. One productive technique here is principled scoping, which attempts to reconstruct how a UEM would be able to predict a specific problem (Blandford et al., 2008). This however provides no guarantee that a specific UEM, when used in an arbitrary context would predict a specific category of problem, nor
can it stop problems that are clearly outwith the principled scope of a UEM from being predicted. Thus Heuristic Evaluation has been ‘used’ to predict problems that are clearly out of scope, leaving evaluators struggling to find a heuristic that even has a remote match to a very obvious usability problem (Cockton & Woolrych, 2001).

MAUSE WG4 (Review on the Computational and Definitional Approaches in Usability Evaluation) took an approach similar to Blandford et al’s principled scoping in their analysis of automated UEMs. Such UEMs are more method than approach, but even here tool users may still appropriate automated UEMs through different approaches to tool configuration and results interpretation. Even so, the scoping constructs used by WG4 (COST294-MAUSE Working Group 4 Final Report, this volume) do offer a basis for sensitivities in future case studies of usability work involving not only automated UEMs, but other approaches to usability.

Future research thus needs to combine the approaches of COST294-MAUSE WG1, WG2 and WG4, concentrating primarily on individual UEMs (or more accurately, approaches to usability, or A2Us), but balancing description of an A2U’s ‘ingredients’ with practical advice on how to get the best out of them. Such advice should be supported by empirical evidence, and should extend to how to get the worst out of an approach, that is, include evidence-based cautionary tales on how A2Us can be poorly assembled and applied with disappointing results. This will entail a move away from the idea of method choice as something like picking cans off supermarket shelves. Methodological choices within usability work are far more complex, and the idea of an A2U captures this by making it clear that choosing A2Us refines a context for further unavoidable context-specific choices. Thus usability specialists must decide which ingredients of an A2U to use, missing some out but also adding their own to the mix. Recalling the coffee analogy used earlier, by this stage, usability specialists can only choose the beans for the blend. They still have to make the coffee. As a result, real UEMs, as constituted in real practices in usability work, are always unique and situated. As with coffee, getting from beans to the cup involves a series of actions. Usability work must go through a similar sequence, making further choices at every step, to make project-specific journeys from A2Us to completed project methodologies.

A fundamental error in the idea of choosing methods is the implicit assumption that choices can be specified wholly in terms of options and selections. A good choice is never a good choice solely in relation to unchosen options. There is more to choice than the menu. There is also the diner and their needs, which cannot be reduced to a universal set of comparative coding constructs, allowing us to map deterministically from diners to menu choices. While criteria such as nut allergies have straightforward consequences for choices from menus, wanting something ‘interesting and different’ does not. When choosing and configuring A2Us, the typical diner is more often in the realm of wanting something ‘interesting and different’ than coping with a nut allergy. The goodness of a choice lies in the context of the choice, e.g. Furniss (2008) argues that an appropriate choice and use of a method is functionally coupled to the context, including client biases, practitioner expertise, their relationship, the budget, the problem, the time, auditing potential, persuasiveness. Using comparative coding constructs, scoping principles, UEM analysis templates or similar devices as a proxy for real world contexts for usability work thus carries unmanageable risks of error, misrepresentation, omission and distortion.

We must thus recognise the limits of scoping and ranking UEMs, which don’t actually exist as methods until U2As are assembled and configured and used in real world usability work. Instead, we need new methodological approaches to understanding usability work that are compatible with the realities of complex judgement-based project work.
A New Methodological Framework for Understanding Usability
Work and Issues

The original vision for WG2 assumed that UEMs could be compared through multi-site experiments (MSEs). The expectation was that research methodologies from experimental psychology would be suitable. Here, independent variables are manipulated, while holding potential confounds constant, and the impact on dependent variables is measured. The causal effect of the independent variables can then be established via descriptive or inferential statistics. Given a large enough effect with no overlap in the values of dependent variables for each condition, descriptive statistics alone are enough to establish a causal relationship between the manipulated independent variables and the measured dependent variables. Where there is overlap and thus a less obvious effect, appropriate statistical tests are needed to establish the probability of observed effects being due to chance.

The fundamental error in this thinking is to assume that UEMs can ever be independent variables and that potential associated confounds can be held constant (or even reliably measured for multi-factorial experiments). The true situation is far more fraught than the coffee examples used above, unless we extend the scope of the analogy to the storage of the coffee beans. If the coffee beans are damp, old, sun baked, mouldy or permeated with environmental odours, then the result of the coffee tasting will not be fair to those who grew, harvested, transported and roasted the coffee beans. In the analogies so far, we have assumed that the coffee beans are in good condition. While this is a tractable problem for coffee beans, keeping UEMs fresh, pristine and optimally productive is an impossible challenge.

UEMs are more like alkali metals than coffee beans. They are highly reactive, and like the alkali metals, some are far more reactive than others. No alkali metals are found in elemental form in nature, whereas UEMs are only found in the unnatural elemental forms of text books, tutorial notes and scientific publications, but very rarely in the same form on each appearance! These paper idealisations are similar to the special storage required for all metals, starting with mineral oils and ending with inert or noble gasses. Encased in a soluble vial and dropped into a large vessel of water, rubidium (Rb) and caesium (Cs) will destroy the vessel. This reactivity is due to the presence of only one electron in the outer shell. UEMs can also be thought of as having very incomplete outer ‘shells’. As a result, they will rapidly react with their human environment, instantaneously loosing their text book forms and becoming lost in a compound of unique usability work. The more structure within the UEM, the less reactive they are. Heuristic Evaluation and user testing can be thought of as highly reactive UEMs, while Cognitive Walkthrough is somewhat less reactive, and the automated UEMs surveyed in WG4 less reactive still. However, no UEM is even slightly inert, and if it were, it would be utterly unusable and useless. UEMs must react with their usage environments to be useful and usable.

It is thus simply impossible to design realistic credible experiments with UEMs as the sole independent variable. Instead, we must think of UEMs as U2As, that is, always complex, incomplete and requiring choices to enable real world action. Thus if we can experiment at all, we must experiment with specific configurations of U2As as our independent variables. The combinatorial space here is such as to make any realistic experiment utterly arbitrary in its choice of configurations, e.g., fixed task user testing with concurrent think aloud using an opportunity sample and debriefing compared to Cognitive Walkthrough with supplied keystroke level task sequences, a specific fixed method for determining success or failure cases, no provision of user background information, and a streamlined protocol for skipping answered questions and deferring some unanswered ones. Oh, and the problem extraction for the user testing would be carried out by self-trained psychology undergraduates following a fixed protocol and the Cognitive Walkthrough by human resources staff with responsibility for health and safety in computer-based work. Looking back further to the car example in making comparisons, we may as well run an experiment comparing red shiny fast cars with dirty yellow slow dumper trucks. That would simply be pointless, and so too would a genuine comparison of fully configured UEMs. The laboratory conditions would be no less unique and unreplicable than those that hold in every day usability work. Hence, the latter is what we should be
researching, not supposedly isolatable UEMs. Just as alkali metals are kept pure in mineral oil or noble gases, so UEMs only remain pure on paper. Once placed in a real human context, UEMs rapidly react, leaving few visible traces of their elemental essence.

A new research methodology for comparing U2As should focus on the impact of specific UEM configurations (i.e., a specific combination of the ‘ingredients’ of a U2A plus whatever else is required to fully instantiate the UEM). One of the primary contributions of Furniss (2008) was to develop a positive resonance model of how methods fit into the context of practice. In this conception methods are functionally coupled to the context and their appropriateness and performance is dependent on the ‘push’ it gives to a project. Much like a child learns to apply the right push and timing to resonate with a swing we should understand the resonance that affects a method's push on a project, considering client, practitioner and project contingencies. This impact needs to be studied from the three perspectives of the RITE method, i.e., finding genuine relevant usability problems (GRUPs), understanding them, and fixing them. Where fixing them may also include persuading clients and playing the political game that is sometimes needed to make changes alongside establishing and maintaining relationships in practice.

We also need to avoid the implicit assumptions about UEM choice, scoping and ranking in the initial vision for WG2. The purpose of a new research methodology should not be to hand down idiot-proof positive knowledge to arbitrary unmotivated usability urchins, but to support experienced reflective practitioners in complex creative knowledge work. Given this, a new methodology for understanding A2Us should be able to produce three forms of guidance, each suited to a particular RITE perspective, i.e.,

- To find GRUPs of Type A, use Method B in Manner C with Expectations D.
- To understand GRUPs of Type A, use Method / Theory B in Manner C
- To fix GRUPs of Type A, use Method /Theory B in Manner C

Such guidance can be given four forms of support, with increasingly levels of guarantee (but never absolute ones). The first is advice based on detailed well structured case studies of usability work. Uldall-Espersen et al. (2008) is a recent example of such an in-depth case study. Looking forward, future studies should make careful use of report formats, not only for predicted, persuasive and fixed problems, but also for the overall process of usability work, allowing the influence of a range of factors including even leadership and vision to be explored. Such case studies thus need to be guided and shaped by a common research infrastructure that sets out a basis for credible inferences from adequate comprehensive process data. For example, the Filtering Coding Construct (S1) proposed by Mark Springett would set requirements for tracing the fortunes of usability issues from their identification as a possible problem, through their confirmation as a GRUP, their acceptance as a priority for the current iteration and their associated design change (or other iterative response) to outcomes of these changes during the next set of evaluation activities. Filtering, or translating, augmenting or revising may thus provide a basis for understandings of usability work that span all three questions of the RITE method (Wixon, 2003). Nørgaard and Hornbæk (2006) studied how think aloud tests were conducted in seven Danish companies. Among other things, they showed that immediate analysis of observations made in the think-aloud sessions was done only sporadically, if at all. Also, during testing, evaluators often ask users about their expectations and about hypothetical situations, rather than problems that the user had experienced.

Basing case studies on a common research infrastructure will improve the quality of the second form of support, meta-reviews of case studies of usability work. Individual case studies will always raise issues of generalisation, although many concerns here can be offset by the reflective practices of experienced professionals. All case studies come with a caveat emptor warning, so anyone acting on guidance from such studies needs adequate critical resources to judge what can reasonably be generalised to other settings. Even so, the use of a common research infrastructure can and should reduce concerns about excessive subjectivity and a lack of comparability. Indeed, the quality of such
comparability will be apparent within meta-reviews, which will extend support to less experienced practitioners with less developed critical resources. Thus in time, as more case studies are covered in more meta-reviews, we should witness a steady increase in the reliability and utility of guidance for usability work. However, this critical function would not be the sole role of meta-reviews.

The third form of support for usability work would come in the form of models of the interaction design process spanning complete project lifecycles. A key scientific function of meta-reviews would be to construct such models on the basis of credible generalisations over case studies. Such models would improve the critical resources of usability professionals and specialists at all levels of experience, and could ideally be presented in an accessible form for senior non-specialist project, product or service management. Such models could also provide basis for well grounded and socially realistic capability maturity models for usability work. Furniss (2008) has created cross-case models of methods in usability practice, which have been grounded by practitioner interviews, and demonstrate important functional components in the context of using ‘methods’ in practice. Uldall-Espersen and Frøkjær (2007) report similar empirically grounded models of usability practice.

With models in place, the fourth form of support would be what MAUSE WG2 has originally hoped for, i.e., positive objective knowledge from empirical studies. These can draw on a range of research instruments and procedures, including questionnaires, interviews and even controlled experiments. Such knowledge has remained very elusive in UEM research, and will continue to do so if we remain attached to conceptually inadequate experimentation and research instrument design. Current comparisons, investigations and case studies of UEMs do not gather the extent or depth of data from which plausible models can be built, especially for downstream utility. The field has been riddled with arbitrary experimental combinations of independent and dependent variables. Many of these studies have a ‘headless horseman; feel to them, as warned against in elementary psychology texts on experimental design. Rigorous experimental design depends on well defined plausible theories. Models of interaction design processes should provide a basis for well grounded choices of independent and dependent variables. These models will have to go below the level of UEMs, focusing on specific ingredients of A2Us as independent variables and how they interact (between themselves and other technical, client and management factors) to shape the progression and outcomes of usability work. Dependent variables too will move away from naïve counts of problems, superficial constructs for persuasiveness and condensed constructs for downstream utility. Actually, simple minded causal distinctions between independent and dependent variables will have to be replaced by more sophisticated concepts associated with systems variables, where feedback loops can morph an independent variable into a dependent one. For example, Furniss (2008) resorts to using systemic models to describe methods in usability practice as more linearly causal models didn’t ‘fit’ the data. As such, controlled empirical studies would be focused on understanding the dynamics of specific usability subsystems, and not on some hopeless attempt to isolate the causal power of a single UEM. Examples of system variables in a post-UEM research methodology include:

- Report formats for:
  - problem reporting and downstream processing and editing
  - auditing evaluation and iteration processes
  - clients and other non-specialist stakeholders in usability work
- Problem extraction tools for empirical test data
- Problem merging tools
- Project leadership
- Design purpose and visions
- Client needs and expectations
- Budgetary and other resources
- Alignment of design purpose and evaluation purpose
- User testing protocols
- Format of task specifications
- Think aloud protocols
• Training and tutorial support for A2Us
• Professional/specialist education on general discovery and analysis resources
• Project, product and service specific prioritization criteria
• Experience and competence of usability workers

Note that apparent methodological factors such as report formats and problem extraction are treated as systems variables, recognising the inevitable Hawthorn effects that result from changes to work practice (Cockton et al., 2003). The outdated simple minded scientific view of an objective observer separate from observed phenomena must be replaced by a reflexive recursive understanding within case studies where instrumentation and other research interventions inevitably perturb the system being studied. Indeed, this is the whole point of such research, i.e., not to leave nature unchanged, but to systematically, reliably, valuably and persistently change human work practices through various combinations within A2Us. Hence recently there is more focus on individual elements of usability practice, below the level of the UEM, examining practices and outcomes associated with think aloud (Nørgaard & Hornbæk, 2006), problem merging and matching (Hornbæk & Frøkjær, 2008b), and business impact reporting (Hornbæk & Frøkjær, 2008a).

Conclusions

To return one last time to an overworked analogy, it is time to wake up and smell the coffee! When the proposal for the MAUSE action was being written, the default expectation was that the route to maturing usability was more mature UEMs. This is not the case, firstly because UEMs, as paper-based artefacts, cannot mature. Automated UEM tools can improve of course, but the reality is that few, if any, of them come anywhere near performing as well as human usability specialists. What can mature, and what must mature, is usability work and its competence in dealing with a wide range of usability issues. For research to support this, it too must mature, and let go of the apron strings of factorial experiments and progress to systems approaches grounded in well validated constructs.

None of this, however, should be seen as implying that the CODELIGHTS MSE was a waste of effort. Firstly, it reflected approaches current when the MAUSE proposal was being drafted that valid UEM comparisons would be made possible by methodological improvements (e.g., report formats, problem extraction, problem merging) that would overcome the challenges that had been systematically laid out by Gray and Salzman (1998). As such, the faith in factorial experiments was not then misplaced, even though there were signs that evaluators were far more influential than UEMs in the outcomes of evaluations (e.g., Cockton & Woolrych, 2001; Hertzum & Jacobsen, 2003). However, most UEM researchers still found the evaluator effect more chilling than thrilling. It is, however, time to move on and celebrate the individual differences in evaluation practice and performance, rather than believe that we can eliminate all subjectivity. The issue is not whether evaluators do things differently, but the consequences, both positive and negative, of these differences for software development. We need to identify and promote valuable innovative evaluator practices, and isolate and reduce adverse biases, mistakes and slips.

Secondly, the methodological foci of CODELIGHTS, i.e., comparative coding constructs and problem/evaluation report formats remain key to the new proposed research methodology. Both are essential to meta-reviews and modelling, as well as to study designs for detailed empirical research. CODELIGHTS can thus continue to shine a light on usability research, but not through simple minded comparisons of UEMs. Instead, reflection on comparative coding constructs and data/information reporting formats has and will shine a light on research methodologies, and not just research studies. Maturing usability evaluation requires maturity in our approaches to approaches to usability! Ideally, no-one should ever again be rash enough to make one and only one direct comparison of one UEM to another. Instead, we should compare various blends of the ingredients of A2Us, looking inside the misleading thin and sparse shells of UEMs to the fundamental particles within them. The resulting wisdom will compare very favourably to the naïve innocence underlying the initial plan for MAUSE WG2. We are beginning to grow up as usability researchers.
References


Science, University of Glasgow, Scotland. Available via http://www.dcs.gla.ac.uk/asp/materials/, last accessed 8.2.09


Meriam-Webster On-line (ND), last accessed 5/2/09 at http://www.merriam-webster.com/dictionary/compare


Appendix A:
Coding Constructs Used in Secondary Analysis

This appendix presents the comparative coding constructs used in the secondary analysis. Each was developed by one or more MAUSE WG2 members, uploaded to the MAUSE wiki, and edited as the construct evolved. Alan Woolrych steered the constructs through to the current state below.

Note that some construct presentations are more extensive than others, reflecting their maturity in specific research contexts. The two most complete are presented first, and are drawn from research by Alan Woolrych and colleagues, and were known to be codable against the last selected problem set for the north east England travel system web site.

All references in this appendix can be found in the references list above

A4 Discoverability

1. **Name:** Discoverability
2. **Definition**
   The complexity of description (number of steps/systems) of problem discovery (i.e., what did the inspector or test user do to find the problem)
3. **Operationalisation** (what does the coder do?)
   a. Describe the steps to problem discovery as an interaction sequence
   b. Count the interaction steps and separate applications in this task method
   c. Categorise discoverability using the fuzzy set \{perceivable (0-1), actionable (1-4ish), **constructable** (over 3), multi-constructable (more than one application used)\}
4. **Evidence requirements**
   a. Discovery narrative in extended report format for UIM prediction sets
   b. Video recordings or observer notes (user testing)
   c. Logs files from routine usage
5. **Sources of bias**
   a. Incorrect self-reporting for UIMS
   b. Gaps or inaccuracies in observer notes
   c. gaps in log files
6. **Motivation and Relevance**
   Finding some problems is more straightforward than others. This measure is evaluator independent, and complements evaluator reported effort on problem.
7. **Used in:** Cockton and Woolrych, HCI 2001
8. **Examples** follow from Alan Woolrych’s MPhil thesis follow (overview in Cockton and Woolrych, HCI 2001)

**Example Problem 13. Differences Rotating Grouped and Co-Selected Objects**

*Not predicted by evaluators using Heuristic Evaluation, constructable problem due to complexity of task required to discover it*

When a series of objects are selected they are co-selected (but not grouped). When a series of objects are ‘grouped’ they become a ‘single’ object. Figure AG6 shows identical objects. The upper row of objects are ‘co-selected’, the lower row of objects are grouped.
Figure AG6 Co-Selecting and Grouping Objects

Figure AG7 Rotating Co-Selected Grouped and Objects

Figure AG7 shows the totally different results of attempting to rotate either co-selected or grouped objects. The vertical boxes are the rotated grouped objects, and the horizontal row shows the rotated ‘co-selected’ objects, where each individual box has independently rotated on its own axis.

Example Problem 15. **Undo can only be used for a single previous action.**
*Predicted by 9 evaluator (group)s using Heuristic Evaluation, actionable problem due to simplicity of interaction required to discover it*

The particular application only supports a single ‘undo’ action, resulting in the user having limited control for reversing actions. It was predicted that users are reliant upon ‘multiple undoes’ and task efficiency would be reduced in the absence of such a feature.

Appropriate heuristics judged to have been breached: **3: User Control and Freedom** (x6)

Heuristic 3 is the single most appropriate heuristic relevant to this problem. Despite the provision of an undo feature, evaluators clearly felt it was insufficient and multiple ‘undoes’ was more appropriate. Heuristics cited by various analysts deemed inappropriate to the problem;
None of the above are relevant to this problem. For example, Heuristic 5: an alternative design (multiple undos) would not prevent errors, but instead assist in error recovery.

### User Testing Results

<table>
<thead>
<tr>
<th>Frequency</th>
<th>High</th>
<th>Medium</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Severe</td>
<td>Nuisance</td>
<td>Minor</td>
</tr>
<tr>
<td>P11</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Despite the high frequency of prediction for this element, very few of the users showed a dependency on the feature. It was clear to the observer that a ‘multiple undo’ feature would have prevented all of the task failures regarding Problem 15. However no participant who suffered task failure as a consequence of Problem 15, were thwarted as a consequence of just a single undo feature.

Despite the presence of features which support almost unlimited ‘undo’ operation in current applications, user testing in this study did not provide evidence to support the dependence of such features. However, it was clear to the observer that should such a feature have been available, and the user had been aware of the feature, many task failures could have been avoided.

The exception was P11 who did in fact express a wish for further options for undoing work, however in this participant’s case the problem was minimized because they had saved work previous to this task and simply reverted to the saved version.
Example Problem 35. Default colour for objects is not displayed. 
Predicted by 1 evaluator using Heuristic Evaluation, perceivable problem due to lack of interaction required to discover it

The chosen/default fill colour for drawn objects is not displayed anywhere on the interface (e.g., as part of the fill tool icon). It was predicted this element posed a usability problem as it breached visibility of system status.

Appropriate heuristics judged to have been breached: 1: Visibility of System Status

Once again, despite the lack of evidence to prove this prediction to be a usability problem, the correct heuristic has been considered.

User Testing Results
There was no user testing evidence to prove this element a problem.
A7 Distributed Cognitive Resources (DCRs)

1. **Name:** Distributed Cognitive Resources (DCR’s)
2. **Definition**
   Those cognitive resources available to analysts as information, knowledge or beliefs, distributed in the project environment as documents, expertise or opinions that influence analysts in problem discovery and analysis during inspection.
3. **Operationalisation** (what does the coder do?)
   a. Analyse extended problem report format narratives for evidence of specific DCRs (discovery and confirmation/elimination narratives)
   b. Code as instances of these current categories:
      i. User
      ii. Task
      iii. Domain
      iv. Design
      v. Interaction
      vi. Technical
      vii. Product
4. **Evidence requirements**
   a. Discovery and analysis narratives in extended report format for UIM prediction sets
5. **Sources of bias**
   a. Incorrect self-reporting for UIMS
   b. Gaps in self-reporting for UIMS
   c. Limited confidence in accurate coding of DCR’s
6. **Motivation and Relevance**
   Finding problems requires different cognitive resources. Whilst some may require an understanding of user tasks (interaction knowledge), other problems for example may require technical or domain knowledge. A lack of, or ‘perceived’ knowledge (misuse) can result in missed problems and/or misanalysis of discovered problems (false positives/negatives)
7. **Used in:** Woolrych et al., HCI 2005 – extracts follow
8. **Examples** (See six problem sets on MAUSE wiki: College Web Site, Alan Woolrych and Professional Resource Web Site (URL in some problem reports), Alan Woolrych

Further insights from Woolrych et al. (2005)

1. **Perceived user knowledge and false positives**
   What appears to be significant correlation between user knowledge and false positives suggest that “knowledge” of users is often inappropriate beliefs, and its absence thus improves prediction outcomes. In the pilot there are several examples of misconceptions of user abilities resulting in false positive reporting due to underestimating the abilities of users. In the first example, (Pilot 1, Group 1, Master Problem 4), the analyst group reported probable usability problems due to the lack of a “general back button integrated throughout the site”. The group justified their confirmation of this prediction by stating “users could end up lost and confused as they won’t be able to get back to the main sections of the site”. In user testing, users were able to navigate throughout the site without any difficulties associated with the lack of such a feature, resulting in the reported problem being a false positive.

2. **Resources usage frequency**
   Product knowledge is not included in the following tables. This is explained by the fact that some degree of product knowledge is implied by the ability to carry out inspections by the analysts. As could be expected there was not an even distribution of resources usage in the pilot study. Table 1 shows the frequency of resource usage in both problem discovery and analysis.

**TABLE 1:** Resource usage in problem discovery and analysis
TABLE 2: Resource usage in problem discovery

<table>
<thead>
<tr>
<th>Resources</th>
<th>User</th>
<th>Task</th>
<th>Domain</th>
<th>Design</th>
<th>Interaction</th>
<th>Technical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>87</td>
<td>4</td>
<td>11</td>
<td>49</td>
<td>19</td>
<td>20</td>
</tr>
</tbody>
</table>

TABLE 3: Resource usage in problem analysis

<table>
<thead>
<tr>
<th>Resources</th>
<th>User</th>
<th>Task</th>
<th>Domain</th>
<th>Design</th>
<th>Interaction</th>
<th>Technical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>43</td>
<td>1</td>
<td>3</td>
<td>20</td>
<td>8</td>
<td>7</td>
</tr>
</tbody>
</table>

User knowledge was the most frequently used resource, with an even distribution across both discovery and analysis (Tables 1 and 2). Task knowledge was the most infrequently used knowledge resource, whilst domain knowledge although relatively limited in use, was predominantly used as a discovery resource.

**A9a+b Persuasiveness**

This construct was included in the secondary analysis to bring downstream utility into scope in a focused manner

1. **Name:** Persuasiveness\(^1\) (alias Feedback’s appeal to stakeholders)
2. **Definition:** The extent to which relevant stakeholders (incl. designers, developers, managers) are persuaded about the realness of the problems reported, and the necessity as well as the urgency to fix them. It is a composite construct being determined by several attributes. This construct plays an important role in so-called developer effect, which refers to systemic biases of developers to fix usability problems with particular characteristics.
3. **Operationalisation:**
   (a) Check the availability and characteristics of the following attributes in individual problem descriptions:
      - Severity – provide stakeholders with information how serious a problem is (NB: unreliability of severity nominal/ordinal scales)
      - Frequency – provide stakeholders with information how many users have experienced the same problem (NB: objective data, more convincing)
      - Context – provide stakeholders with sufficiently detailed, clear, and concrete information to understand a problem
      - Value – provide stakeholders, especially business representatives and project sponsors, with the information how a problem would adversely influence economical value and long-term business development
   (b) Check the availability and characteristics of the following attributes in individual redesign proposal:
      - Rationale – explain to stakeholders why the proposal can be a resolution to the problem or a better alternative of the current design
      - Scope – provide stakeholders with information whether the proposal deals with a specific instance of a problem or a cluster of similar problems
      - Complexity/Difficulty – provide stakeholders (especially managers) with information how much work is estimated to be required to implement the redesign proposal (NB: This attribute is better evaluated by developers)

Note: Both (a) and (b) should be presented in a clear and comprehensive way; some balanced views of the existing system are deemed important.

4. **Evidence Requirements:**
   a. Think-aloud protocols and video recordings from user-based evaluation sessions
   b. Structured evaluation report, including problem description and redesign solution
   c. Interviews with developers

---

\(^{1}\) Integrated the construct definition prepared by DIKU
5. **Sources of Bias:**
   a. Incorrect interpretations of users' behaviours and verbal comments
   b. Wrong severity ratings; inaccurate estimation of economic impacts
   c. Incomprehensible problem descriptions or redesign solutions

6. **Motivation and Relevance:**
   Usability evaluation outcomes should be balanced and addressed to relevant stakeholders, including developers, designers, managers, business representative, project sponsors, etc. Doing so can enhance the impact of evaluation on software product design because motivation and expertise is propagated in the transformation processes when stakeholders recognize the problems and are able to contribute to solutions (NB: adapted from DIKU).

7. **Used in:** Some attributes are mentioned in Law (2006), otherwise much of this is speculative.

---

**A9d Value to Development**

This construct was included in the secondary analysis to bring downstream utility into scope in a broad manner

1. **Name:** Downstream Utility (DU)/ Value to Development (VtD)

2. **Definition:**
   Downstream utility is defined as the effectiveness with which the resolution to a usability problem is implemented. DU manifests in terms of value to development – the extent to which a usability evaluation report contributes to the future development of the system under scrutiny (NB: This construct is better coded by developers themselves or by usability professionals interviewing developers). DU is influenced by developer effect that is strongly related to the construct persuasiveness.

3. **Operationalisation:**
   a. Check the value of a problem description to enable a development team to understand, prioritise, and fix the problem:
      i. Understandability: Does the description tell about “what”, “where”, “when”, and “how” the problem occurred (i.e. the contextual information) in sufficient details?
      ii. Prioritization: Does the description provide data (e.g. severity, frequency) that may enable a development team to decide which problems should receive fix first (the question of “which”)?
      iii. Fixing: Does the description provide data about the origin of the problem (i.e. the question of “why”)?
   b. Check the value of a redesign solution to enable a development team to improve the overall quality of a system:
      i. Abstraction – how detailed and concrete the solution is
      ii. Complexity – how many code changes are required
      iii. Innovativeness – how creative the solution is
      iv. Feasibility – how practical the solution is in terms of resources required (i.e. associated with complexity)

4. **Evidence Requirements:**
   a. Think-aloud protocols and video recordings from user-based evaluation sessions
   b. Structured evaluation report with the attribute “redesign solution”
   c. Interviews with developers

5. **Sources of Bias:**
   a. Incorrect interpretations of users’ behaviours and verbal comments
   b. Wrong severity ratings
   c. Incomprehensible problem descriptions or redesign solutions

6. **Motivation and Relevance:**
   It is important to enhance the interplay between evaluation and redesign. Usability evaluation reports should provide developers with not only detailed information about problems but also plausible resolutions

7. **Used in:**
S2 Clarity
This construct definition and characterisation is in an older format than the ones above. It was included in the secondary analysis as it appeared to be a very inclusive construct that should apply regardless of reporting formats.

1. **Name:** Clarity
2. **Definition:**
   Whether or not the reader of the problem description is left with a relatively sure understanding of what is intended on part of the evaluator.
3. **Motivation and Relevance:**
   Usability problems should be easily understandable for the receiver of the usability report. Dumas et al. (2004), for example, suggested being careful to avoid usability jargon in description of usability problems.
4. **Data requirements:** Any kind of problem description
5. **Coding Scale:** nominal – clear / not clear
6. **Coding Rules:**
   Is the reader left with a relatively sure understanding of what is intended on part of the evaluator? (Yes/No)
7. **Used In:**

CUP2 Cause
A subset of CUP constructs (Vilbergsdóttir et al. 2006, which should be consulted for examples) was chosen to reflect experiences and preferences when forming the wider group of comparative coding constructs, as well as experiences at the Salzburg and Toulouse workshops.

1. **Name:** Cause,
2. **Definition:**
   The cause of the fault that lead to the failure.
3. **Operationalisation:**
   The coder indicates the values for the causes. There are five attribute values for the causes: Personal, Communication, Technical, Methodological, Managerial and Review. The value can also be free text if none of the values are appropriate.
4. **Evidence requirements:** Interviews with developers team
5. **Sources of bias:**
   a. reluctance of developer team to discuss causes openly
6. **Motivation and Relevance:**
   With the aim to prevent future errors by improving the development processes, it is useful to understand why developers have committed the error. The cause attribute can guide quality engineers when they determine what to improve or whether the personnel need more training.

CUP3 Context

1. **Name:** Context
2. **Definition:**
   Describes in what part of the user interface the evaluator was when the UP occurred.
3. **Operationalisation:**
   Describe on which screen the UP occurred, identified by title and where within the screen, e.g. screen divided into numbered areas before coding.
4. **Evidence requirements:**
   a. A screenshot from the user interface can be used for additional information and clarification.
   b. Problem reports for each usability problem.
5. **Sources of bias:**
   a. Depends on the accuracy of the data source, e.g. video or screen capture.

6. **Motivation and Relevance:**
   If there are many UPs that have the same value for this attribute it might imply that this particular part of the user interface has some property that is causing the users problems.

### CUP6 Frequency

1. **Name:** Frequency
2. **Definition:**
   Frequency is the number of users/experts that experience/predict a particular usability problem.
3. **Operationalisation:**
   Count how many users/experts experience/predict a particular usability problem.
4. **Evidence requirements:**
   Problem reports for each usability problem.
5. **Sources of bias:**
6. **Motivation and Relevance:**
   Provides the developer with an idea of how much different users/experts agree on that particular usability problem.
   A UP with a high frequency is obviously causing many users problems/delays and may increase the importance of fixing it.

### CUP9 Impact

1. **Name:** Impact
2. **Definition:**
   Indicate what effect the UP can have on the user in case of analytic usability evaluation such as expert inspection or is observed to have in case of empirical usability evaluation such as user test.
3. **Operationalisation:**
   The attribute values of impact are: severe, moderate, and minor.
   - **Severe** usability problems are those that prevent the test participant from completing a task or result in catastrophic loss of data or time.
   - **Moderate** usability problems are those that significantly hinder task completion but for which the test participant can find a way to work-around.
   - **Minor** usability problems are those that are irritating to the test participant but do not significantly hinder task completion.
   In addition, for user tests, the attribute values of impact are: effectiveness, efficiency, frustration expressed and help sought.
4. **Evidence requirements:**
   a. Problem reports for each usability problem.
   b. Video recordings or observer notes (user testing)
5. **Sources of bias:**
   a. Coder’s skills
   b. User’s different expression of experience/emotion, e.g. irritation or perserverance.
6. **Motivation and Relevance:**
   Collecting measurements pertinent to how the UP affects the test participant can help developers decide how serious the UP is. The reported impact can influence how much effort developers intend to use in correcting the fault or how much they intend to change the user interface.
Appendix B: Other Comparative Coding Constructs

The following comparative coding constructs were all considered to some extent at the Salzburg and Toulouse workshops, but following experiences there and/or the judgement of the independent analyst, they were not applied as part of the secondary analysis.

A1: Coding by User Action Framework

1. **Name of Construct:** User Action Framework (UAF)
2. **Definition:** The UAF is a taxonomy of the problems users may experience based on the notion of an interaction cycle, related to Norman’s (1986) model of stages of action. The UAF separates in its top categories issues of planning, translation, physical actions, outcome and system functionality, assessment, and problems independent of the interaction cycle. For each of these issues, usability problems may be classified to subcategories that aim at describing more closely the nature of the problem. This constitutes the UAF problem classification tree.
3. **Operationalisation:**
   a. Interpret the description of each problem report in the light of the definitions of the nodes of the UAF classification tree (and decide which node applies to the problem, i.e., code it using the tree). Decide how many levels in the UAF tree are used (e.g., only the top two levels).
   b. Count the number of problem reports for each node.
   c. Calculate the proportion of problem reports for each node (based on the total number of problem reports found in the study).
4. **Evidence Requirements:** The UAF tree contains descriptions of the meaning of each of the nodes. These descriptions are used as definitions for interpreting whether a problem report describes a problem in relation to that node. There is a public UAF viewer available from Virginia Tech as well as a classification tool called DCART. The coding rules are more-or-less contained in those tools. In Copenhagen, we have used classification both at the first and second level of the UAF tree, some 10 and 50 nodes respectively, to minimize the time it takes to classify a problem.
5. **Sources of Bias:** The matching of node descriptions and problem reports requires (qualitative) language interpretation of both the UAF tree as well as of the problem reports.
6. **Motivation and Relevance:** One motivation behind using UAF to code usability is to profile the kind of problems a particular UEM helps identifying in relation to the interaction cycle. Another motivation is that matching of usability problems may be supported through the use of systematic classification of problems.
7. **Used in/Examples in:** Andre et al. (2001) and Hornbæk and Frøkjær (2008b).
A2a Persistency

This construct definition and characterisation is in an older format, as S2 in Appendix A.

1. **Name:** Persistency (*alias:* Expert / Novice problem)
2. **Definition:**
   A problem is persistent if users proficient with the system would experience the problem. (Is it possible that proficient users experience problems that novices do not? This could indicate that the definition should be adjusted).
3. **Motivation and Relevance:**
   Typically, UEMs are supposed to uncover problems experienced by novice users. For example, think aloud testing is usually conducted with inexperienced users and in heuristic evaluation only one heuristic address experienced users directly. This is reasonable for systems like e-commerce sites or cell phones where first time task completion is essential for success. However, it may not be reasonable for systems used repeatedly by the same user. It has been suggested that persistent problems are valued more by developers than non-persistent (Hornbæk and Frøkjær, 2005)
4. **Data requirements:** Any kind of problem description. If more and less experienced users are tested it should be possible to trace the identified problem back to the user. Otherwise it should be possible to establish by judgment, whether a problem is persistent or not. Missing functionality is always considered persistent.
5. **Coding Scale:** nominal – persistent / not persistent
6. **Coding Rules:**
   Will users proficient with the system experience the problem? (Yes/No)
7. **Used In:**

A2b Prioritization

1. **Name:** Prioritization (Hertzum 2006)
2. **Definition:**
   Viewed from inside a usability evaluation, severity assessments are the major device for influencing problem prioritization and thereby guiding the utilization of design resources.
3. **Operationalisation:**
   The severity of a usability problem is generally considered to be a combination of three factors (Nielsen, 1994):
   - **Impact:** How much trouble will affected users experience?
   - **Persistence:** How many times will a user experience the problem?
   - **Frequency:** How many users will be affected by the problem?
   These factors can be measured individually, but they are frequently collapsed into a single severity rating.
   Often, rating of individual factors is bypassed, and evaluators simply make a single rating of severity on a scale such as 1 (*not a usability problem*), 2 (*minor usability problem*), 3 (*major usability problem*), and 4 (*usability disaster*).
4. **Evidence requirements:** Problem reports describing each Usability Problem.
5. **Sources of bias:** Evaluator skills
6. **Motivation and Relevance:**
   Severity assessments enable prioritization of problems encountered during usability evaluations. However, designers’ response to usability evaluations is also influenced by other factors (i.e skills). If the developer doesn’t know how to fix the UP he is more likely to ignore the UPs priority of correcting.
7. **Used in:** Hertzum (2006)
A3: Effort per Problem
1. **Name**: Effort per Problem
2. **Definition**: during inspection or testing the time needed to find a problem
3. **Operationalisation**:
   a. Evaluator (expert or participant) uses software
   b. Measure time needed for evaluation for each evaluator (session duration)
   c. Count total time needed for all evaluation sessions
   d. Count usability problems for each evaluator
   e. Count total distinct usability problems
   f. Divide total time by number of distinct usability problems = time needed per problem
4. **Evidence requirements**:
   a. Session durations
   b. Problem reports describing each usability problem for each session (needed for matching usability problems)
   c. Matching scheme to find similar usability problems across sessions in order to identify distinct usability problems.
5. **Source of bias**:
   a. Every problem is treated the same. A method finding lots of simple/non-severe problems needs little effort per problem. A method finding mainly severe (and therefore probably less) problems needs a high effort per problem. A severity factor could be included, meaning that e.g. a severe problem gets multiplied by 2.
   b. Outliers have a high impact, e.g. if only one or two problems are responsible for longer session durations, these problems might distort the effort per problem calculation.
   c. Matching problems across evaluators is difficult and often subjective and error-prone.
6. **Motivation and relevance**:
   a. Time is money. If a method finds more problems in a shorter period of time, it is more cost-efficient.

A5 Appropriateness
1. **Name**: Appropriateness
2. **Definition**
The accuracy to which analysts apply heuristics to candidate usability problems (i.e., does the heuristic match the problem?)
3. **Operationalisation** (what does the coder do?)
   a. Analyse the problem description(s) for appropriate breaches of the nominated heuristic relative to explicit conformance criteria (e.g., as in Lavery et al. 1996)
4. **Evidence requirements**
   a. Nominate heuristic(s) for each usability problem.
   b. Provide evidence of non-conformance of heuristic in extended report format for UIM prediction sets
5. **Sources of bias**
   a. Insufficient preparation in the UIM
   b. Incorrect self-reporting for UIMS
6. **Motivation and Relevance**
   It can be misleading to credit heuristics for accurate problem discovery and analysis. Downstream concerns include inappropriate heuristic applications that could result in inappropriate recommendations for design changes.
8. **Examples**: Analysis for Problem set for PowerPoint 95 follows (Woolrych MPhil)

“If a heuristic cannot be associated with a particular usability problem, then Heuristic Evaluation certainly cannot be accredited with the problem discovery."
Figure 5.3 shows the overall use of heuristics by all analysts when reporting what were confirmed as actual problems, and the breakdown of Heuristic usage on severe, nuisance and minor problems.

**Figure 5.3 Heuristic Usage**

Of the 14 correctly predicted problems, 61% of heuristics quoted were considered inappropriate to the specific problem, with 39% of quoted heuristics considered appropriate to the specific problem. The high proportion of inappropriate use of heuristics would indicate that heuristics play a limited role in candidate problem discovery. This is based on the argument that if heuristics played a significant role in candidate problem discovery, the incidence of inappropriate heuristic application would be very much lower.

If heuristics do not play a role in problem discovery, then the only alternative is the application of heuristics ‘after’ candidate problem discovery, i.e. to confirm or eliminate possible (candidate) problems as highly probable ones.

Analysis of heuristic ‘usage’ in the severity categories (severe, nuisance, minor), for problems predicted, shows an interesting trend. The application of heuristics on severe problems (Figure 5.3), shows inappropriate application of heuristics higher (79%) than the overall average (61%), with only 21% of heuristics cited appropriate to severe problems.

Figure 5.3 also shows the application of appropriate and inappropriate heuristics to predicted problems with a nuisance impact. In this category of problem, the application of appropriate heuristics has improved from 21% to 37%. The trend of improving appropriate heuristic association as problem severity decreases continues with those problems that had minor impact. For predicted problems with minor impact on the user, the heuristic usage was the most accurate. Heuristics appropriate to the individual problems now account for 62% of all heuristic applications, with only 38% inappropriate.

Were heuristics playing a prominent role in the discovery of problem candidates, one could expect a much more accurate association of appropriate heuristics to individual problem candidates in general. If heuristics played any role at all in candidate problem discovery, then the heuristic ‘usage’ would be similar across all severity types.

The evidence so far shows the analysts are not influenced by heuristics in the discovery of candidate problems. This is demonstrated by the high proportion of inappropriate heuristics associated with
predictions. The analysis of heuristic usage shows the evaluator is less likely to apply the appropriate heuristic for more obvious problems that turned out to be severe. Conversely, those problems which actually had less impact (nuisance and minor impact), the evaluator is more likely to associate the appropriate heuristic to candidate problems.

The data also suggests that the heuristics are applied after the candidate problem is discovered and that the accuracy of heuristic application is dependant on the evaluator’s need to justify candidate problem selection after problem discovery. The association of appropriate heuristics with only low frequency and low severity problems suggests that analysts only carefully consider heuristics for those candidate problems that are least clear cut. The more probable the problem, the less effort appears to be applied to heuristics in the confirmation of a problem.

In summary, heuristics do not assist the analyst in the discovery of candidate problems, but are used more effectively to confirm some candidate problems, which are low severity/frequency problems.”

Woolrych, A. MPhil Thesis (2001)

“Appropriateness of heuristic application rose to 57% (from 31% for all student predictions in first study), a 26% practical improvement with the introduction of Extended Structured Problem Report Formats (ESPRFs).”

Cockton et al. (2003)

A8 Evaluator Satisfaction

1. **Name:** Evaluator Satisfaction

2. **Definition:** The extent to which evaluators perceive usefulness, difficulty, acceptability of the evaluation technique and confidence with the achieved results, i.e. the discovered problem set. Being determined by several attributes, evaluator satisfaction is a composite construct.

3. **Operationalisation:**
   - It is difficult to get evidence about evaluator satisfaction with the achieved results only from the problem set; feedback is needed from the evaluators. It is suggested to assess evaluator satisfaction in both a direct and an indirect way: directly, by asking evaluators to express their gratification, e.g. through Likert-type questions; indirectly, by the number of problems found, since, usually, the more problems an inspector thinks he has found, the more satisfied he is with his performance.
   - Evaluator satisfaction with the inspection technique can be assessed from evaluators through questions with a semantic-differential scale that require evaluators to judge the technique on pairs of adjectives addressing usefulness, difficulty of learn, difficulty of use, etc.

4. **Evidence Requirements:**
   - a. Questionnaires and/or interviews with evaluators
   - b. Number of discovered problems

5. **Sources of Bias:**
   - c. In some cases, the number of the problem found may not be related to evaluator satisfaction

6. **Motivation and Relevance:**
   - The evaluators’ perception of usefulness, difficulty, acceptability with respect to the evaluation technique has an impact on the evaluation process, since it augments the evaluators’ control on the overall process.
   - Evaluator satisfaction is important in comparing evaluation methods.

7. **Used in/For Examples see:**
   - This construct has been used as a validation dimension in controlled experiments comparing different evaluation methods, e.g. comparing heuristic evaluation and AT inspection in (a) and comparing heuristic evaluation, AT inspection and user testing in (b).
A9c: Solution proposal

This construct definition and characterisation is in an older format, as S2 in Appendix A.

1. **Name:** Solution proposal
2. **Definition:**
   Is there a proposed solution to the problem as well as a description of it?
3. **Motivation and Relevance:**
   It has been suggested that developers appreciate solutions as an outcome of usability evaluations (Hornbæk & Frøkjær, 2005)
4. **Data requirements:** Any kind of problem description.
5. **Coding Scale:** nominal – solution / no solution
6. **Coding Rules:** Does the description give one or more recommendations for how to fix a problem? (Yes/No)
7. **Used In:** Hornbæk & Frøkjær (2006)

S1 Filtering

1. **Name:** Filtering
2. **Definition:**
   The journey that a Usability Problem takes from point of discovery to its destination, in terms of changes of recorded form, translations into formats, taxonomies etc and passing between individuals Is there a proposed solution to the problem as well as a description of it?
3. **Operationalisation:**
   Count transfers of evaluation information (e.g. descriptions of the phenomena, categorization, interpretation).
   a. Number of transfers in journey
   b. Type/nature of transfer (taxonomy classification, event description etc)
   c. Estimate inference margin at each transfer (e.g. inferred meanings of behavioral data, uncertain causal reasoning, interpretations of terminology).
   d. Actors involved
   e. physical form: observed behaviour (tacit), introspection (UI), semi-tacit articulation (e.g. protocol utterance), taxonomy selection, textual description, visual description
4. **Evidence Requirements:** all evaluation materials, procedural description of methods
5. **Sources of Bias:** poorly filled forms, misinterpreted behaviours, mismatched interpretation of descriptions.
6. **Motivation and Relevance:** Method efficacy is affected by the number of times that information is interpreted, described and passed on, and the form and nature of the translation. This applies both to the way elicitation is set up to prompt observable phenomena, and the way in which descriptions and diagnostic judgements are formed, classified, shared and passed on. Method quality is affected by the reliability of inferences and the efficiency of these events
7. **Used In:** Speculative, Mark Springett

S3 Complexity of Redesign Proposal

This construct definition and characterisation is in an older format, as S2 in Appendix A.

1. **Name:** Complexity of re-design proposals
2. **Definition:**
   This construct allows for an estimation of the complexity of a re-design proposal
3. **Motivation and Relevance:**
   Proposals for re-design have varying degrees of complexity; a complex proposal may involve many changes (such as changing the entire workflow of how to buy an e-ticket), a less complex proposal may involve simple directions for correction (such as adding tool-tips to explain how to buy an e-ticket). Depending on when the evaluation takes place in the product cycle complex re-design proposals (however well considered) may be difficult to implement and thus face the fate of being ignored.
4. **Data requirements:** Any kind of problem description.
5. **Coding Scale**: ordinal: low-medium-high complexity, quick-and-dirty vs. time consuming/expensive nominal

6. **Coding Rules**: The complexity may be rated according to how much work is required to implement the re-design proposal. The evaluator estimates (roughly) how many issues the proposal touches upon and how much time it will take to implement the re-design proposal. Low-complexity proposals touch on ‘few/minor’ issues, medium-complexity proposals touch on ‘some’ issues and high-complexity proposals touch upon ‘many’ issues

7. **Used In**: Speculative, DIKU

**CUP1 Actual Phase**

The remaining CUP constructs (Vilbergsdóttir et al. 2006, which should be consulted for examples) that were not used in the secondary analysis.

1. **Name**: Actual Phase
2. **Definition**: Indicates in which of the eight phases of software development lifecycle the developer corrects the faults that lead to the failure. In the case of user test, the expected phase does not have to be specified right after the test but before feedback is given to developers.
3. **Operationalisation**: The coder indicates where the developer corrected the fault that lead to failure:
   - Task analysis and context of use (TAN)
   - Functional requirements (FUR)
   - Quality attribute analysis (QAN)
   - Conceptual modelling (COM)
   - Dialogue design (DIA)
   - Navigational design (NAV)
   - Presentation design (PRE)
   - Implementation (IMP)
4. **Evidence requirements**: Fix reports for each usability problem.
5. **Sources of bias**: a. Codes could be overlapping  
   b. Coder’s skill
6. **Motivation and Relevance**: This construct is to be collected after the fault has been fixed. The main reason for collecting it is to see whether evaluators are right in predicting the phase where the fault leading to UP actually lies (cf. Expected Phase). The developer should attempt to fill out this construct using the eight phases. The values Requirements Analysis and Design can be used in cases where the developer can not distinguish between the sub-phases of these phases. In addition to the eight phases listed above N/A (not applicable) is a possible value when developers do not fix the UP identified, because they are not convinced about the necessity or urgency of fixing it. This construct can also give developers some ideas which phases carry most faults and hence improve those processes.

**CUP 4 Defect Detection Activity**

1. **Name**: Defect Detection Activity (Alias: Defect Removal Activity)
2. **Definition**: Describes the type of usability testing methods, i.e. user test, heuristic evaluation or cognitive walkthrough.
3. **Operationalisation**: Describe what method was used to detect a particular defect. (i.e. Usability Evaluation Method)
4. **Evidence requirements**: Evaluation protocol
5. **Sources of bias**: None
6. **Motivation and Relevance**: to be able to compare attributes between defect removal activities.
CUP 5 Error Prevention Technique

1. **Name:** Error Prevention Technique.
2. **Definition:** Ideas are collected on what can be done in the future to prevent the fault.
3. **Operationalisation:** Ideas on what can be done in the future to prevent the fault.
4. No categories of attribute values are designed for this construct. What can be obtained for this attribute are various guidelines for error prevention.
5. **Evidence requirements:** Interview with developer team
6. **Sources of bias:** Inexperience with process improvements
7. **Motivation and Relevance:** When more data of this sort is systematically collected, categories representing different types of attribute values can be developed. To qualify this construct, reviewers can examine the cause and see how the same problem can be prevented from happening again.

CUP 7 Expected Phase

1. **Name:** Expected Phase
2. **Definition:** Indicates in which phase of a software development lifecycle the developer is expected to be able to fix the Usability Problem.
3. **Operationalisation:** The coder indicates where the developer is expected to be able to fix the UP
   - Task analysis and context of use (TAN)
   - Functional requirements (FUR)
   - Quality attribute analysis (QAN)
   - Conceptual modelling (COM)
   - Dialogue design (DIA)
   - Navigational design (NAV)
   - Presentation design (PRE)
   - Implementation (IMP)
4. **Evidence requirements:** Problem reports for each usability problem.
5. **Sources of bias:**
   a. Detailed software lifecycle not used in project.
   b. Coder’s familiarity with software lifecycle
6. **Motivation and Relevance:**
   - Helps to analyze if something is wrong in the development process, e.g. if many problems are related to conceptual modeling then there is something wrong with that particular phase.
   - Helps to see how to find a solution of a problem.

CUP 8 Failure Qualifier

1. **Name:** Failure Qualifier
2. **Definition:** To capture more meaning about the UP. A usability specialist records the value based on the verbal protocol of the test participant. In case of heuristic evaluation and cognitive walkthrough, the usability specialist records his or her experience.
3. **Operationalisation:** In the context of user test, the qualifier further explains whether the test participant (i.e. the user) experiences the UP as something being Missing (M), Incongruent mental model (I), Irrelevant (IR), Wrong (W), or whether there could be a Better way (B) for doing a task or the test participant Overlooked (O) something
4. **Evidence requirements:** Problem reports for each usability problem.
   - Video recordings or observer notes. (user testing)
5. **Sources of bias:**
   a. Codes could overlap
   b. Coder tries to estimate/guess user’s experience, e.g. incongruent mental model.
6. **Motivation and Relevance:**
   - Useful to understand the description of a problem better. Should be easier to understand a problem and subsequently find a solution.
CUP 10 Trigger

1. **Name:** Trigger
2. **Definition:** This attribute describes what an evaluator is doing when he or she discovers the usability problem.
3. **Operationalisation:**

<table>
<thead>
<tr>
<th>Usability Evaluation Method</th>
<th>Trigger</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Test</td>
<td>Task scenarios that evaluators are asked to perform on a user interface.</td>
</tr>
<tr>
<td>Heuristic Evaluation</td>
<td>Heuristics that an evaluator applies to examine the interface and it is necessary to register where in the interface the UP is found.</td>
</tr>
<tr>
<td>Cognitive Walkthrough</td>
<td>Reflective questions that an evaluator attempts to answer when analyzing a specific action of a selected.</td>
</tr>
</tbody>
</table>

4. **Evidence requirements:** Problem reports for each usability problem.
5. **Sources of bias:**
6. **Motivation and Relevance:** If there are many UPs that have the same trigger value that might imply that the task being performed is causing the users different types of problems. It might imply that how the task is carried out in the system needs reconsideration. The trigger value also aids understanding of the UP because it provides information about what the user was intending to do, i.e. his goal.

CUP 11 Type of Fault Removed

1. **Name:** Type of Fault Removed.
2. **Definition:** This construct is to be collected after the fault has been fixed. It is meant to describe what in the user interface was changed to fix the fault. No categories of construct values are designed for this attribute.
3. **Operationalisation:** Describe what in the user interface was changed to fix the problem.
4. **Evidence requirements:** Fix report
5. **Sources of bias:**
6. **Motivation and Relevance:** The type of information developers keep track of themselves.
### First Problem Set used in Secondary Analysis

**List of Unique Usability Problems Identified in Four Language Versions of EducaNext Portal (2004)**

<table>
<thead>
<tr>
<th>Serial ID</th>
<th>Task ID</th>
<th>Who</th>
<th>Context</th>
<th>Descriptions of Usability Problem</th>
<th>Severity (Impact)</th>
<th>Tester’s Suggestions/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UP1</strong></td>
<td>1.1</td>
<td>E3</td>
<td>Login page</td>
<td>The color of the hyperlink (blue) is hardly distinguishable from the rest of the text</td>
<td>Minor</td>
<td>Brighter color with block letters should be used.</td>
</tr>
<tr>
<td><strong>UP2</strong></td>
<td>1.2</td>
<td>E3</td>
<td>Step 1: Apply for a user account</td>
<td>Terms of Use is not translated into German</td>
<td>Minor</td>
<td>What is the reason of not translating the document?</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Terms of Use)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>UP3</strong></td>
<td>1.3</td>
<td>E1, E2, E6, S3, I3</td>
<td>Step 3: Apply for a User Account (password)</td>
<td>5 instead of the required 6 characters for the password. Overlooked the error message, which was regarded as inconspicuous.</td>
<td>Moderate</td>
<td>Explicit statement that the field password requires minimum 6 characters. Error messages should be presented in the conspicuous color or, better, on a separate popped up window.</td>
</tr>
<tr>
<td><strong>UP4</strong></td>
<td>1.4</td>
<td>E3, E4</td>
<td>Step 3: Apply for a user account (personal details)</td>
<td>Surprised to find the default address (that of the LRA?). Some attributes such as fax number, telephone number are regarded unnecessary</td>
<td>Minor</td>
<td>Reset the attributes</td>
</tr>
<tr>
<td><strong>UP5</strong></td>
<td>1.5</td>
<td>E4</td>
<td>Step 3: Apply for a user account (role)</td>
<td>The attribute ‘Role’ is not completely clear, the helpertext cannot well explain it.</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td><strong>UP6</strong></td>
<td>1.6</td>
<td>I1, I4</td>
<td>Apply for an account</td>
<td>The tester’s institution cannot be found easily</td>
<td>Moderate</td>
<td>Users have to scroll through a long list of institutions in the drop-down menu. It is better to add search function here.</td>
</tr>
<tr>
<td>UP7</td>
<td>1.7 (2)</td>
<td>L1, L2</td>
<td>Retrieving forgotten password</td>
<td>The facility for password reminder transmission via email did not work. The request was submitted successfully, a confirmation was displayed, but no email with the password was received.</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP8</td>
<td>1.8 (1)</td>
<td>L2</td>
<td>Retrieving forgotten password</td>
<td>If an invalid username is used, the system accepts it as a legitimate one and acknowledges a password reminder request.</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP9</td>
<td>2.1 (9)</td>
<td>E1, E2, E3, E5, E6, E7, I3, I5, L2</td>
<td>Start page of “My Contribution”</td>
<td>Had difficulty in locating where to provide LR. The label “My Contribution” (Meine Lehrinhalten) is not suggestive enough, no indication about uploading or managing content. The drop down menu of “My Contribution” does not include all the items on the left-hand-side navigation bar on its front page.</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP10</td>
<td>2.2 (6)</td>
<td>E2, E3, E6, I1, I4, L1</td>
<td>Help &amp; Support</td>
<td>The user was unable to find the menu for providing LR, then attempted to locate the relevant information from “User Manual”. Too much text to read and could not find the required information. Another user was not aware of the presence of helpertext; for her, the icon (?) was not meaningful.</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP11</td>
<td>2.3 (5)</td>
<td>E3, E4, E5, E6, L1</td>
<td>Help &amp; Support</td>
<td>The PDF version of User Manual could not be opened (An empty page is shown);</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP12</td>
<td>2.4 (2)</td>
<td>E3, E5</td>
<td>Help &amp; Support</td>
<td>FAQ is not translated into German.</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP13</td>
<td>2.5</td>
<td>E2, I3</td>
<td>Online information</td>
<td>No clear indication of the starting</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP</td>
<td>(2)</td>
<td>E1, E5, S1, S2, I3, I5</td>
<td>Step 1: Provide a new Ed Material</td>
<td>Omitted the mandatory field; overlooked the error message and needed to go through all the pages to locate the error.</td>
<td>Severe</td>
<td>Error message should NOT be embedded in the field, but be a popped up window displaying a clear message. Clicking ‘ok’ on this window should lead back to the page where the error occurs.</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>------------------------</td>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UP15</td>
<td>2.7</td>
<td>E3, E4, S1, S2, S3, L2</td>
<td>Step 1: Provide new LR (Defining discipline)</td>
<td>The participant hardly found the link to the discipline setting.</td>
<td>Minor</td>
<td>It should be more visible.</td>
</tr>
<tr>
<td>UP21</td>
<td>2.13</td>
<td>E1, E3, I5</td>
<td>Provide Ed Material – Step 2 (Select author)</td>
<td>It was like (the blank next to the field author) that the user forgot to pick up an author. The link (referring to author selection) was not underlined. In fact, there was inconsistency: “Select author” by clicking the button; Select discipline by clicking the hyperlink text “Click here”</td>
<td>Moderate</td>
<td>The user expected the same design as for inputting discipline in Step 1. Reverse the order of the hyperlink text “Click here” and the empty blanks. It appears more logical and makes it similar to “select author”. Or, to modify the layout to make the two input methods consistent.</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------------</td>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UP22</td>
<td>2.14</td>
<td>E2, E6, I5</td>
<td>Step 2 of Provide a new Ed (Create a new contributor)</td>
<td>Users had the impression that they’re required to fill in too many forms!</td>
<td>Moderate</td>
<td>Eliminate some optional fields in the form so as to make the form less intimidating.</td>
</tr>
<tr>
<td>UP23</td>
<td>2.15</td>
<td>S1, S3</td>
<td>Provide new LR – Step 2</td>
<td>The test user (first time provider) did not realize that he/she had already been included in the list of contributors. After choosing the ‘Select author’ button, she created a new contributor with her personal information.</td>
<td>Moderate</td>
<td>The process should perhaps be divided into two steps. The user first searches for an existing contributor. Only in the case, he/she doesn’t find the contributor, the EducaNext offers him a possibility to create a new one.</td>
</tr>
<tr>
<td>UP24</td>
<td>2.16</td>
<td>E1, E2, E6, S2</td>
<td>Step 1 and 2 of Provide Ed Material</td>
<td>Tended to enter text in the empty spaces, which are actually to be filled by selecting one of the given options. The ineffective clicks on the empty space led to frustration in the user</td>
<td>Moderate</td>
<td>&quot;Hide&quot; the blank spaces. The values of to-be-filled fields will be shown after the options are selected. Or allow users to enter text in these blanks as it can grant them stronger sense of control.</td>
</tr>
<tr>
<td>UP25</td>
<td>2.17</td>
<td>E6, I5</td>
<td>Step 2: URL</td>
<td>The user did not bother to change URL, which was reported to be invalid but actually correct!!</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP26</td>
<td>2.18</td>
<td>E6, L1</td>
<td>Step 2: Upload LR</td>
<td>The user attempted to upload the file, but was confused by the term UNIVERSAL Hosted Server, then left this option and entered another</td>
<td>Moderate</td>
<td>The option “UNIVERSAL hosted server” sounds somewhat intimidating for non-technical users. Choice of server could be better explained</td>
</tr>
<tr>
<td>UP</td>
<td>2.19</td>
<td>S1, S2</td>
<td>Provide new LR – Step 2</td>
<td>When a provider chose her delivery system, but did not upload her LR, the EducaNext did not complain. LR provision was successfully finished, without LR being uploaded.</td>
<td>Severe</td>
<td>The procedure of LR provision should not be regarded as finished successfully before a provider correctly uploads LR or provides URL. A clear warning message should be displayed if an empty LR is provided.</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>--------</td>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>--------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UP28</td>
<td>2.20</td>
<td>S1</td>
<td>Provide new LR – Step 2</td>
<td>The user attempted to offer the LR several times, because the title of LR was not shown on the homepage (&quot;Recently added Ed Material&quot;). In the first run, she forgot to upload the LR, but she did so in the second run. However, the UBP somehow still could not display her LR.</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP29</td>
<td>2.21</td>
<td>E2, E6, E7</td>
<td>Step 2/3: Provide Ed Material</td>
<td>Clicked the back button of the browser and lost all the data, very frustrating (Alles verloren. Furchtbar!).</td>
<td>Severe</td>
<td>Disable the back button of the browser. The back buttons of the browser and of the platform are very confusing! The constraint for not allowing users to click the back button of the browser should be removed.</td>
</tr>
<tr>
<td>UP30</td>
<td>2.22</td>
<td>E3, E4</td>
<td>Step 4 of Provide Ed Material</td>
<td>Awful terms for “Lernmethode” – Fremdgesteuerte Lernen</td>
<td>Moderate</td>
<td>Help text should be given here.</td>
</tr>
<tr>
<td>UP31</td>
<td>2.23</td>
<td>E3</td>
<td>Step 4 of Provide Ed Material</td>
<td>Decimal number for ECTS. The error message was not specific enough to explain why 0.5 could not be accepted for ECTS</td>
<td>Minor</td>
<td>Improve the error message or help-text to emphasize the fact that only positive integers are accepted as ECTS</td>
</tr>
<tr>
<td>UP32</td>
<td>2.24</td>
<td>E3, E4, S3, L3</td>
<td>Offer page</td>
<td>Eingeschränkte UNIVERSAL Lizenzvereinbarung (Lizenz Details ansehen). When the hyperlink was clicked, the error message “Page not found” was displayed.</td>
<td>Moderate</td>
<td>A program bug!</td>
</tr>
<tr>
<td>UP33</td>
<td>2.25</td>
<td>E3, E4</td>
<td>Navigation Bar on the left-hand-side</td>
<td>When offering function was invoked, some “Umlaut” characters on some</td>
<td>Minor</td>
<td>A program bug!</td>
</tr>
<tr>
<td>UP34</td>
<td>2.26 (3)</td>
<td>S1, S2, S3</td>
<td>Provide new LR – Step 2</td>
<td>The provider thought that besides the author, additional contributors are also mandatory</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP35</td>
<td>2.27 (1)</td>
<td>S2</td>
<td></td>
<td>The participant did not know if the LR was uploaded correctly or not.</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP36</td>
<td>2.28 (1)</td>
<td>L1</td>
<td>Offer new LR</td>
<td>What the offer was actually for was not clearly explained</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP37</td>
<td>3.1 (2)</td>
<td>E1, E6</td>
<td>Front page of My Contribution</td>
<td>Forgot to click the select button before attempting to modify the LR</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP38</td>
<td>3.2 (2)</td>
<td>L1, L3</td>
<td>Modifying discipline of the Ed. Material</td>
<td>One has to go through all the “LR Provisions” screens, even what is needed is to change just one item. Time wasted in verifying all other irrelevant screens.</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP39</td>
<td>3.3 (1)</td>
<td>E2</td>
<td>Summary page of LR – confirm the update</td>
<td>Application error – but the user had no interest to fill in the error report form</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP40</td>
<td>3.4 (4)</td>
<td>E5, E7, S3, I5</td>
<td>Modify the discipline</td>
<td>The user deleted the old discipline but not entered the new one. It is not so intuitive that for modifying one needs to delete the old discipline and add a new one</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP41</td>
<td>3.5 (1)</td>
<td>E6</td>
<td>Modify the discipline</td>
<td>There was a lack of immediate feedback that the modification process was completed and what should be done next</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP42</td>
<td>3.6 (1)</td>
<td>E7</td>
<td>Update the LR</td>
<td>The user complained that the upload time of the modified file was too long</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP43</td>
<td>4.1</td>
<td>E5, E6, E7, S2, I1, L1</td>
<td>Step 1 of Provide Ed Activity (LR type)</td>
<td>Users had difficulty in finding out where to input a new Ed Activity. They had no clear concept about the relationship between Ed Material and Ed Activity. Some users expected separate interfaces for EA/EM.</td>
<td>Moderate</td>
<td>Users did not read the online text carefully. They tended to scan it. Or is there too much text for them to read?</td>
</tr>
<tr>
<td>UP44</td>
<td>4.2</td>
<td>E3, E4, E5, S2</td>
<td>Step 2: Zurück (Back) button</td>
<td>Everything entered had been deleted after the “zurück” button above the input window (not the “zurück” button of the browser) was clicked – a very frustrating experience</td>
<td>Severe</td>
<td>The two “zurück” buttons correspond to “Return” and “Previous” in English. Clicking “Return” will navigate back to the front page of “My Contribution” and everything previously entered will be deleted. Clicking “Previous” simply will navigate back to the step earlier. When the top “zurück” (i.e. Return) is visible and the bottom one is hidden, users tend to click the former. Then they lose everything just entered and become frustrated. Actually, the top “zurück” is not necessary at all. If I click “Abbrechen”, it will navigate back to the very front page of “My Contribution”.</td>
</tr>
<tr>
<td>UP45</td>
<td>4.3</td>
<td>E3</td>
<td>Step 2 of Provide Ed Activity (Bildungsaktivitätstyp)</td>
<td>After selecting the values for the attributes “Medientyp” and “Bildungsaktivitätstyp”, the user clicked “Hilfe &amp; Support”, scanned the document “Qualitatsrichtlinien Metadata” and then clicked “zurück”, the previously two selected values were gone.</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP46</td>
<td>4.4</td>
<td>E3, E4, E7, I1, I3, I4, I5</td>
<td>Step 2: Provide Ed Activity (Bildungsaktivitätstyp)</td>
<td>Lack of explanation for different learning resource types (Bildungsaktivitätstyp). For instance, the definition of “Kurs” varies with</td>
<td>Moderate</td>
<td>Help text or hyperlink to some relevant web-pages should be added here</td>
</tr>
<tr>
<td>UP47</td>
<td>4.5 (5)</td>
<td>E4, E7, S1, S2, L2</td>
<td>Step 2: Provide Ed Activity (Additional Contributors)</td>
<td>The attribute is optional, but is placed within the cell described as mandatory. Misleading!</td>
<td>Moderate</td>
<td>Clear separation lines should be presented. Or put the word “optional” after the attribute.</td>
</tr>
<tr>
<td>UP48</td>
<td>4.6 (3)</td>
<td>E1, S2, L1</td>
<td>Step 2: Provide a new Ed Activity</td>
<td>Overlooked the mandatory fields.</td>
<td>Minor</td>
<td>The indicator for mandatory fields should be more conspicuous</td>
</tr>
<tr>
<td>UP49</td>
<td>4.7 (1)</td>
<td>E1</td>
<td>Step 2 of Provide a New Ed Activity (helptext)</td>
<td>Helptext of the attribute &quot;Information&quot; is the same as that of &quot;URL&quot;.</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP50</td>
<td>4.8 (1)</td>
<td>E6</td>
<td>Step 2: Provide Ed Activity (URL/Access Info)</td>
<td>The user attempted to enter both URL and Access Information, because she perceived that both fields were mandatory.</td>
<td>Moderate</td>
<td>The layout is confusing. The chosen one is highlighted whereas the other is dimmed.</td>
</tr>
<tr>
<td>UP51</td>
<td>4.9 (1)</td>
<td>E3</td>
<td>Step 3: Provide Ed Activity (Technical requirements)</td>
<td>Lack of the option for videoconferencing in “Technical Requirements”, e.g. H323, H320, Codec</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP52</td>
<td>4.10 (5)</td>
<td>E1, E4, S2, S3, I5</td>
<td>Step 5: Provide a New Ed Activity (frequency)</td>
<td>Confused by the given options of scheduling, because none of them fits one-shot event</td>
<td>Severe</td>
<td>The option of one-shot event should be given</td>
</tr>
<tr>
<td>UP53</td>
<td>4.11 (2)</td>
<td>E1, L1</td>
<td>Offering the provided Ed Activity</td>
<td>Error message about the wrong URL was displayed. But after clicking &quot;OK&quot;, the front page of “My Contribution” instead of the corresponding page where URL was inputted was displayed. The user re-entered the metadata and did not bother to correct the URL.</td>
<td>Severe</td>
<td>Navigation error: the appropriate page where the error lies (I.e. URL) should be displayed. Alternatively, instruction of how to update the URL should be provided</td>
</tr>
<tr>
<td>UP54</td>
<td>4.12 (1)</td>
<td>E3</td>
<td>Step 5: Time zone</td>
<td>The helptext for time zone is wrong!</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP55</td>
<td>4.13 (1)</td>
<td>E3</td>
<td>Offer –defining usage</td>
<td>Users were impatient by asking to fill in too many online forms!</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP56</td>
<td>4.14 (1)</td>
<td>I4</td>
<td>All pages: Provide Ed Activity</td>
<td>Small font</td>
<td>Minor</td>
<td>There should be a link to make characters smaller or bigger</td>
</tr>
<tr>
<td>UP57</td>
<td>4.15</td>
<td>I4, I5</td>
<td>All pages: Provide Ed Activity</td>
<td>Mix of English and Icelandic</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP58</td>
<td>4.16 (2)</td>
<td>I5, E7</td>
<td>Offer: License Agreements</td>
<td>Too long and too detailed. The long text discouraged users to read them seriously.</td>
<td>Minor</td>
<td>A precise of the License Agreements should be first presented. Interested readers can click links to read full text.</td>
</tr>
<tr>
<td>UP59</td>
<td>4.17 (2)</td>
<td>E5, E7</td>
<td>Search for Contributors</td>
<td>Misunderstood that all the criteria were compulsory</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP60</td>
<td>4.18 (1)</td>
<td>E3</td>
<td>Step 2/3 Provide a new Ed Activity</td>
<td>The thin white lines separating different segments are not conspicuous enough. The terms like Discipline should be on the top of the cell but not in the middle</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP61</td>
<td>4.19</td>
<td>E4</td>
<td>All pages: Provide Ed Activity</td>
<td>Allow for free text entry for those blanks that are designed to be filled in by making a choice out of the set of predefined options</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP62</td>
<td>4.20 (1)</td>
<td>I3</td>
<td>Provide Ed Activity – Front page</td>
<td>The user had difficulty in reading the menu items, because of the floating text that covers menu items</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP63</td>
<td>4.21 (1)</td>
<td>I5</td>
<td>Provide Ed Activity – Step 2</td>
<td>Instead of entering URL, the user had to provide additional information, which was perceived as excessive demand</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP64</td>
<td>5.1 (2)</td>
<td>S1, S2</td>
<td>Modifying the schedule</td>
<td>The user thought that it was necessary to delete old appointments first and then add the new ones (negative transfer from ‘discipline’)</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP65</td>
<td>5.2 (3)</td>
<td>E3, E7, I5</td>
<td>Modify the Schedule of Ed Activity</td>
<td>For the newly input LR, by default the start and end dates are “heute”. However, for updating the schedule of an existing LR, the start and end dates are 01 Jänner (which language?) 2000 are presented.</td>
<td>Minor</td>
<td>The old schedule instead of 01 Jan 2000 should be displayed.</td>
</tr>
<tr>
<td>UP66</td>
<td>5.3 (1)</td>
<td>I3</td>
<td>Delete old appointment</td>
<td>Application error occurred</td>
<td>Severe</td>
<td>The system appeared to be rather unstable</td>
</tr>
<tr>
<td>UP67</td>
<td>5.4 (1)</td>
<td>I3</td>
<td>Update Scheduling</td>
<td>Could not find where to delete the specific appointment of an educational activity</td>
<td>Moderate</td>
<td>The icon for delete does not have universal connotation</td>
</tr>
<tr>
<td>UP68</td>
<td>5.5 (1)</td>
<td>I5</td>
<td>Modify Scheduling</td>
<td>The icon for modifying was not obvious</td>
<td>Moderate</td>
<td>Avoid using icon that is not intuitive or lack of universal interpretation</td>
</tr>
<tr>
<td>UP69</td>
<td>5.6 (4)</td>
<td>E5, E6, S3, L2</td>
<td>Modifying the schedule of the Ed Act.</td>
<td>After time scheduling, the provider of the educational activity did not press the ‘Add appointments’ button, but the system did not complain. EA provision procedure was successfully finished.</td>
<td>Severe</td>
<td>The EducaNext has to warn the provider when he/she forgets pressing the ‘Add appointments’ button after time scheduling.</td>
</tr>
<tr>
<td>UP70</td>
<td>5.7 (1)</td>
<td>E5</td>
<td>Scheduling</td>
<td>The concept that setting an appointment was different from making an offer was not intuitive to the user</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP71</td>
<td>6.1 (8)</td>
<td>E1, E2, E3, E4, E6, E7, S2, S3</td>
<td>Front page of My Contribution</td>
<td>Had great difficulty in locating the right menu, the users adopted trial-and-error approach. Usually, the users clicked “Ändern” (which only for changing the metadata) several times. Then they tried out different menus on the left hand side navigation bars (e.g., LR freigegebenen, LR eingreifen, Lizenz, etc). The users complained that the names of the tags were ambiguous and not suggestive of the function (e.g., Vorlage for &quot;template&quot;, odd translation?)</td>
<td>Severe</td>
<td>There are at least three different ways to enter the offer-changing processes: - “Angebot zeigen“ (view offer) - “Details” (details) - „Ausgewählten Lehrinhalt anbieten (Offer selected LR, left hand side bar) Interestingly, it took most users quite a long time to hit one of the three paths. Re-labeling the button &quot;view offer&quot; on the top of the list of LR as &quot;view/modify offer&quot;.</td>
</tr>
<tr>
<td>UP72</td>
<td>6.2 (1)</td>
<td>E1</td>
<td>Front page of My Contribution</td>
<td>The users complained that the fonts were too small</td>
<td>Minor</td>
<td>Enlarge the fonts of left-hand-side menus.</td>
</tr>
<tr>
<td>UP73</td>
<td>6.3 (1)</td>
<td>E2</td>
<td>Customized offer</td>
<td>Appeared too complex. The user was intimidated by a bundle of forms to be filled</td>
<td>Moderate</td>
<td>Minimize the number of attributes in the form</td>
</tr>
<tr>
<td>UP74</td>
<td>6.4 (4)</td>
<td>E3, E5, S2, S3</td>
<td>Offer</td>
<td>After clicking the button “Angebot aktualisiert”, the summary page showing the updated offer with three buttons “ändern”, ‘löschen’, “erstellen” below was displayed. Only by clicking “zurück” can one leave the task. Sort of confusing!</td>
<td>Moderate</td>
<td>A summary page confirming the updating with a clickable ‘ok’ button should be presented instead, just like the message for thanking an offer when it is first defined.</td>
</tr>
<tr>
<td>UP75</td>
<td>6.5 (1)</td>
<td>E3</td>
<td>Offer</td>
<td>The warning message that the user should not click the back button of the browser was issued, though the user did not touch that button at all</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP76</td>
<td>6.6 (2)</td>
<td>E4, E7</td>
<td>Offer</td>
<td>The structure of making offer is not intuitive enough. It is not clear to users that an LR can have more than one type of offers</td>
<td>Severe</td>
<td>Is it really logical that an LR can have more than one type of offer? Relationship between provision and offer should be clearer.</td>
</tr>
<tr>
<td>UP77</td>
<td>8.1 (2)</td>
<td>E1, E3</td>
<td>Catalogue on the homepage</td>
<td>The number of actual LRs in a certain discipline in the UBP catalogue does not match with the figure given in the brackets. It was confusing and disappointing to find the mismatch.</td>
<td>Severe</td>
<td>Is the algorithm for automating the numbers of LRs in different disciplines as displayed in the catalogue wrong? It is strange that the discipline of LR will not be displayed in the catalogue if it is classified with a sub-discipline. The structure of the catalogue is not intuitive to users, especially the total number of LRs in a certain discipline does not equate with the sum of the LRs distributing in its sub-disciplines. The term “LR not further categorized: XX” should be put under each discipline to make the two numbers match.</td>
</tr>
<tr>
<td>UP78</td>
<td>8.2 (2)</td>
<td>E2, E7</td>
<td>Catalogue browsing page</td>
<td>Confused why LRs were not listed. Overlooked the combination of the filter</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP79</td>
<td>8.3</td>
<td>E4</td>
<td>Catalogue browsing</td>
<td>The filter was not flexible, topical</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP80</td>
<td>8.4</td>
<td>E4, I1</td>
<td>Viewing LR metadata</td>
<td>More information about LRs prior to booking would be useful. By looking at the metadata, one has no idea what an LR is about.</td>
<td>Severe</td>
<td>Quality control of metadata. Some strategies to enforce or encourage providers to describe LRs as thoroughly as possible.</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>--------</td>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------</td>
<td>--------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UP81</td>
<td>8.5</td>
<td>E3</td>
<td>View LR</td>
<td>Logo of the institutions and authors of some LRs were missing</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP82</td>
<td>8.6</td>
<td>L2</td>
<td>Browsing catalogue</td>
<td>The user was not aware of the personalization effect and complained that the browser catalogue was stuck to “Computer Science”, which was actually his preferred discipline.</td>
<td>Minor</td>
<td>It’s paradoxical that the user perceived the personalization effect as a nuisance</td>
</tr>
<tr>
<td>UP83</td>
<td>8.7</td>
<td>I5</td>
<td>Browsing the catalogue</td>
<td>A couple of spelling errors in the descriptions of learning resources</td>
<td>Minor</td>
<td></td>
</tr>
<tr>
<td>UP84</td>
<td>9.1</td>
<td>E4</td>
<td>Advanced Search page</td>
<td>AND/OR logical combinations, wildcard search etc would be helpful</td>
<td>Moderate</td>
<td>Copy the help text from LRP pages to here</td>
</tr>
<tr>
<td>UP85</td>
<td>9.2</td>
<td>E2</td>
<td>Advanced Search</td>
<td>No help text for the attributes</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP86</td>
<td>9.3</td>
<td>I5</td>
<td>Search</td>
<td>The search did not respond when the Icelandic characters were entered as keyword.</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP87</td>
<td>9.4</td>
<td>E3</td>
<td>Catalogue</td>
<td>A user chose a sub-discipline for the discipline ‘Andralogy’, the corresponding discipline was not shown in the catalogue. He deleted the sub-discipline, then ‘Andralogy’ appeared in the catalogue.</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP88</td>
<td>10.1</td>
<td>E2, E6, S1, S2, S3, L2</td>
<td>Access LRs</td>
<td>Expired LRs. The users did not know exactly why some LRs were not accessible.</td>
<td>Severe</td>
<td>Catalogue management. How should we ‘upgrade’ the expired LRs - by re-offering? The fact that there are so many expired Ed Material and Ed Activity will hurt the image of the platform. Alternatively, disable/dim the button</td>
</tr>
<tr>
<td>UP</td>
<td>10.x</td>
<td>E</td>
<td>Issue/Feature</td>
<td>Description</td>
<td>Severity</td>
<td>Note</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>UP89</td>
<td>10.2</td>
<td>E3</td>
<td>Access LR</td>
<td>The license terms were shown twice</td>
<td>Minor</td>
<td>This problem had been identified in the earlier usability tests and was reportedly eliminated.</td>
</tr>
<tr>
<td>UP90</td>
<td>10.3</td>
<td>E3</td>
<td>Summary page of LRs</td>
<td>Technical information should be compulsory, which is important for users to select LR</td>
<td>Moderate</td>
<td></td>
</tr>
<tr>
<td>UP91</td>
<td>10.4</td>
<td>E1</td>
<td>Access LR page</td>
<td>Frustrated when asked to fill in the form for the information related to the usage of the LR</td>
<td>Moderate</td>
<td>Reduce the ‘load’ on the user by minimizing the need to fill in the form.</td>
</tr>
<tr>
<td>UP92</td>
<td>10.5</td>
<td>E1</td>
<td>Access LR page</td>
<td>Had no idea what to input for “username” and “password” on the Login page for accessing a restricted LR. The error message “You have no right accessing this LR” was displayed. The message is not polite enough</td>
<td>Moderate</td>
<td>Clear explanations should be given how to access restricted LRs! Change the message “You have no right accessing this LR” to something less aggressive.</td>
</tr>
<tr>
<td>UP93</td>
<td>10.6</td>
<td>E1, I1</td>
<td>Accessing LR</td>
<td>System failure. Could not access the selected LR. Frustrated</td>
<td>Severe</td>
<td></td>
</tr>
<tr>
<td>UP94</td>
<td>General</td>
<td>I1, I3, I4, I5</td>
<td>All pages</td>
<td>There are errors in the Icelandic translation. The helptext on the pages does not make sense. The learning resource types given are confusing because of the translation problem. The Icelandic text was not well written</td>
<td>Severe</td>
<td>Thorough proofread the translated version</td>
</tr>
<tr>
<td>UP95</td>
<td>General</td>
<td>L2, L3, I3</td>
<td>All pages</td>
<td>The system crashed without any warning or error message; the reliability of the system must be improved</td>
<td>Severe</td>
<td></td>
</tr>
</tbody>
</table>

Note: the shadowed 7 UPs are culture-related. Last updated: 22.03.04
### Second Problem Set used in Secondary Analysis

#### Usability Problems for Travel Planner Web Site

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>Specify part of the site</th>
<th>Screen for entering starting point and end points for routes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td>The term Entity is unclear. What is meant is City.</td>
</tr>
<tr>
<td></td>
<td>Violates heuristic number:</td>
<td><strong>2, entity is not a real world term in traveling.</strong></td>
</tr>
<tr>
<td></td>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>People will be confused about what should be entered here, but in the end they will probably find out.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>Specify part of the site</th>
<th>Screen for entering starting point and end points for routes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td>The field for entering an entity (for example) does not automatically quickly come up with a list the first time you use it. At subsequent times during the same session it does come up quicker.</td>
</tr>
<tr>
<td></td>
<td>Violates heuristic number:</td>
<td><strong>1, 4. Not consistent with windows.</strong></td>
</tr>
<tr>
<td></td>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>People will hesitate about what to enter, and will then enter a complete word. Often no matches will be found then. If the list would appear quick enough during typing people would realize it is better to select something after a few letters.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>Specify part of the site</th>
<th>Screen for entering starting point and end points for routes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td>Entering a date can not be done by typing, it has to be selected.</td>
</tr>
<tr>
<td></td>
<td>Violates heuristic number:</td>
<td><strong>4. Inconsistent with the other fields to be entered.</strong></td>
</tr>
<tr>
<td></td>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>People will start typing in the field. That does not work, also no list will show up: one has to click on change date, which looks like something not related to this field; usually these things look differently in windows.</td>
</tr>
</tbody>
</table>
Problem no. 4
Specify part of the site
Screen for entering starting point en end points for routes.

Problematic design aspect and heuristic that is violated
Entering a date. The default date shown is in English, however, once you have to select another date, only French names of days are shown.
Violates heuristic number:
2. and 4.
Likely user difficulties
Inconvenient to select a day, because if you don't know French you will have to count to find out what day it will be.

Problem no. 5
Specify part of the site
Showing the results of a search.

Problematic design aspect and heuristic that is violated
Unclear what the line number in the right column will give. It seems to be in the column “subsequent itinerary”
Violates heuristic number:
6
Likely user difficulties
People may not tend to use the link to the line number, because it is not clear what to get there.

Problem no. 6
Specify part of the site
More specific info about a line number after getting the results of an itinerary.

Problematic design aspect and heuristic that is violated
A pop up is blocked. Whereas at the results screen there was a message saying that a popup was blocked, here no such message appears.
Violates heuristic number:
4.
Likely user difficulties
Users may wonder why nothing appears and re-click the line number.

Problem no. 7
Specify part of the site
More specific info about a line number after getting the results of an itinerary

Problematic design aspect and heuristic that is violated
The extra information consists of the full time table for that line. Something like that could be useful in case one wants to be flexible in their trip. But in that case one would want to have alternatives for the other modes of traffic as well.
Violates heuristic number:
7
Likely user difficulties
Users may get frustrated to find out that they can only find alternatives for bus times and not for other traffic.
<table>
<thead>
<tr>
<th>Problem no.</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify part of the site</td>
<td></td>
</tr>
<tr>
<td>More specific info about a line number after getting the results of an itinerary</td>
<td></td>
</tr>
<tr>
<td>The heading is in French, whereas the rest of the text is in English</td>
<td></td>
</tr>
<tr>
<td>Violates heuristic number:</td>
<td>4</td>
</tr>
<tr>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>No serious difficulty expected.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify part of the site</td>
<td></td>
</tr>
<tr>
<td>More specific info about a line number after getting the results of an itinerary, asking for a pdf</td>
<td></td>
</tr>
<tr>
<td>The pdf does not appear, only a message mentioning: No Timetable available for the line chosen</td>
<td></td>
</tr>
<tr>
<td>Violates heuristic number:</td>
<td>5</td>
</tr>
<tr>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>Users may want to click to get the full time schedule as pdf in order to print it. However, nothing appears. This will frustrate the user, and causes extra unnecessary actions.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify part of the site</td>
<td>Results of itinerary, price information</td>
</tr>
<tr>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td></td>
</tr>
<tr>
<td>On the results popup there is no possibility to find price information. People would expect it there. However, it can only be found from the page in which the route is specified (and then only partly).</td>
<td></td>
</tr>
<tr>
<td>Violates heuristic number:</td>
<td>7</td>
</tr>
<tr>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>People may not easily find where price information is to be found. They will not tend to search for it on the page for specifying the route.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify part of the site</td>
<td>Price information, after receiving the results of a search for a route</td>
</tr>
<tr>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td></td>
</tr>
<tr>
<td>If one finds the place for finding price information one has to specify entities and stops again. One would expect that there would be a link between the results just found. Now one has to remember the stops or look them up again.</td>
<td></td>
</tr>
<tr>
<td>Violates heuristic number:</td>
<td>6</td>
</tr>
<tr>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>Most likely many people will give up trying to find the exact price.</td>
</tr>
<tr>
<td>Problem no.</td>
<td>12</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Specify part of the site</td>
<td>Price information, after receiving the results of a search for a route</td>
</tr>
<tr>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td>One has to search for number of zones and then later for price per zone. This is incomprehensible and inefficient. One would expect that this would be possible at one go. Violates heuristic number: 7.</td>
</tr>
<tr>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>People will be frustrated for having to re-enter information they already entered.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify part of the site</td>
<td>Price information, after receiving the results of a search for a route</td>
</tr>
<tr>
<td>Problematic design aspect and heuristic that is violated (what element or behavior of the design will lead to difficulties?)</td>
<td>Only after trying to find the price information one gradually may find out that only prices for buses can be found. As this seems to be a general public traffic planner, one would expect to be able to find the complete price for a route. Violates heuristic number: 5.</td>
</tr>
<tr>
<td>Likely user difficulties (to what difficulties in use will the problematic design aspect lead, how does it affect website use)</td>
<td>People will spend quite some time trying to find out whether they did something wrong, before they find out that only price information for buses can be found if they find out at all.</td>
</tr>
</tbody>
</table>
Understanding Quality Attributes with Repertory Grid Technique
Final Report of COST294-MAUSE Working Group 3

Mark Springett
Middlesex University, U.K.
Leader, COST294-MAUSE Working Group 3

Contributors:

Effie L-C. Law
ETH Zürich, Switzerland/
University of Leicester, England

Marc Hassenzahl
Folkwang University, Germany

Jari-Jukka Heikkilä
University of Oulu, Finland

Jan Stage
Aalborg University, Denmark
Abstract

COST294-MAUSE Working Group 3 investigates the nature of usability classifications and terminology. Its main mission is to properly define and understand the relationship between traditionally defined metrics and quality attribute terminology, to address the need for a thoroughly validated defect classification theme and definitions of quality in support of design and evaluation. Its principle work involves empirical studies in which HCI experts and Design/Development specialists are interviewed to understand their perceptions of quality attribute terminology and in turn assess their prioritization of significant design concepts and their assessment of how this is represented by quality attribute terms. In doing this we investigate perceptions of the relationship between usability and user experience concepts, a key emergent theme in the lifetime of this COST Action. The investigation involves both a macro-level comparison of HCI and Design professionals statements made in Repertory Grid interviews and more detailed analysis of their accounts of design and evaluation practice in their specialised fields.
# Table of Contents

- Overview .......................................................................................................................... 86
- Analysis of Quality Attribute Interpretations ................................................................. 87
  - Rationale for the Empirical Work ................................................................................ 87
- The Repertory Grid Technique ....................................................................................... 87
- Analysis of Results .......................................................................................................... 89
  - Procedure .................................................................................................................... 89
  - Findings ....................................................................................................................... 89
    - Experience from Data Collection ........................................................................... 89
    - Macro Analysis from February 2007 ................................................................. 89
    - Macro Analysis from February 2009 ................................................................. 90
- Analysis of Designers ..................................................................................................... 91
  - Profiles of the Respondents ....................................................................................... 91
  - Descriptive Statistics ............................................................................................... 92
  - Individual Grid Analysis: Des1 .............................................................................. 97
  - Individual Grid Analysis: Des5 .............................................................................. 99
- Analysis of HCI Professionals ....................................................................................... 102
  - Profiles of the Respondents ..................................................................................... 102
  - Descriptive Statistics ............................................................................................. 102
  - Individual Grid Analysis: HCI-5 ............................................................................ 107
- Analysis of Combined Datasets ..................................................................................... 109
- Low Level Analysis ...................................................................................................... 111
  - Background ............................................................................................................... 111
  - Measurability and Desirability ............................................................................... 111
  - Further Analysis of Measurability ......................................................................... 114
  - Definitional Variation ............................................................................................... 114
    - Complexity – Three Definitions ......................................................................... 115
- Ongoing Work .............................................................................................................. 115
- References .................................................................................................................... 116
- Appendix A: Initial Euclidian Distance Model .............................................................. 117
Overview

The objectives of Working Group (WG) 3 embraces principally three key issues defined in the main mission statement. These are to properly define and understand the relationship between traditionally defined metrics and quality attribute terminology, to address the need for a thoroughly validated defect classification theme and definitions of quality in support of design and evaluation. These are described in the initial MAUSE documentation:

- There is a lack of systematic review on the suitability of the three traditionally defined usability metrics – effectiveness, efficiency and satisfaction – for revealing the extent to which a system is useable or useful, especially when the complexity of the system evaluated is high. Similarly, there is also a lack of empirical research to investigate the relationships among a cluster of quality attributes, all of which should ideally be incorporated into a system design. Specifically, whether these attributes are truly orthogonal as traditionally assumed or they are connected in a way that can be represented with a mathematical model is still an empirical question to be explored.

- There is a lack of a thoroughly validated defect classification scheme specifically designed for analysing usability problems based on the integration of basic concepts of HCI and SE. Such a scheme is crucial for extracting useful information for developers to improve the systems evaluated. Indeed, a defect classification scheme is a method to enhance communication between usability engineers and software developers.

- There is a lack of a global approach to defining quality models for the Web. Some UEMs do not explicitly have such an underlying quality model, whereas some others do. Given this condition, quality models are hard to compare to each other as they are expressed in various formalisms (e.g., algebraic notation, first-order predicate logic, graph theory, graph grammars) that prevent a true benchmarking. Consequently, it is desirable to come up with a benchmarking of quality models of the Web that effectively and efficiently enable people to assess the usability of a website according to used UEMs.

After initial consideration and some re-scoping of the groups activities, the work focussed principally on the definition of quality attributes, their associated metrics and implications for method selection. The motivation for this work has a number of component aspects.

- Investigating the relationship between usefulness and usability:

One of the emergent themes during the lifetime of MAUSE is the need to understand the relationship between traditional notions of usability and the more recent notion of user experience. In particular, we investigated the prima facie issue of whether subjective elements of system use can adequately be defined or measured. Following on from this we considered the role of relatively ‘soft’ user experience attributes in design and evaluation.

- Investigating semantic and definitional ambiguities in quality attribute definitions:

The group investigated possible ambiguities in definition of quality attributes, partly as these definitions may have significantly different interpretations dependent on domain, and partly due simply to semantic ambiguities. The concern addressed is that this terminology is intended as an aid to understanding design problems, and communicating such concepts within design teams.

- Targets are set and used on design projects:

Efficiency, effectiveness and satisfaction are cited with definitions in quality standards documents. Other attributes have known definitions associated with them. More modern definitions of user experience quality are proposed in the academic literature, but orthodoxy has yet to emerge.
Success of a design against such quality criteria varies in tangibility. Some attributes such as efficiency can be linked to measurable targets in a relatively precise way. Therefore the setting of these targets implicitly prescribe certain methods and instruments for evaluation against specific criteria. However other attributes are less tangibly linked to evaluation approaches.

The nature of Working Group 3’s multi-partner mission emerged essentially as research, distinguishing it from the other working groups in the action. This was deemed necessary for two reasons. First was the need to analyse attribute definitions in a systematic way. The second was the analysis of relatively new user-experience definitions, which were not yet established in the literature or in standards.

Analysis of Quality Attribute Interpretations

Rationale for the Empirical Work

The empirical work undertaken by the group involved structured interviews with experts in two cells, namely HCI researchers and designers/developers. The study used the Repertory Grid technique (RGT), described in the selection below. The aims of the work are summarised as follows:

• To develop meta-categories for analysing quality attributes
• To identify areas of dispute in definitions of quality terms and conceptual ambiguities in attribute definitions
• To gauge the influence of specialism or background on expert perceptions or interpretations of HCI concepts
• To assess the repertory grid technique as a tool for defining quality requirements on design projects.
• To investigate the possible development of tighter, more specific expressions of quality in future standards publications e.g. setting targets as a process of meaning-making rather than a selection from a list

A suite of established attribute terms and definitions appear in standards documents and HCI evaluation checklists. Their presence in these lists suggests that there are orthodox interpretations on such terms. However there is some doubt that such declared orthodoxy sufficiently reflects the complexity of the real-world phenomena that these terms refer to. Equally, an emerging set of user experience attributes are ubiquitously proposed in the academic literature, suggesting the need for a close analysis of their nature, role and relationship to more traditional usability attributes.

The Repertory Grid Technique

The technique emanated from work in the 1950’s on personal construct psychology (Kelly, 1955). The technique was originally designed to help individuals to articulate knowledge and attitudes in an indirect fashion that could not easily be expressed by direct probing. The technique has been used in knowledge acquisition for expert systems as a way of probing tacit expertise in domain professionals (Ford & Petry, 1989). More recently it has been deployed in user experience evaluation, to compare user experiences with alternative designs (Hassenzahl & Wessler, 2002). The technique was selected as a way of facilitating elicitation and comparison of expert interpretations of attribute meanings and application to real world phenomena. The technique is now described in detail:

• Preliminary phase
The subjects were asked about their experience, roles and domains in which they have worked.

• Introduction phase
The subjects were briefed on the repertory grid technique. A small ‘priming’ study was given to them at the beginning on an off-topic subject area. This was intended to ensure that they were clear about the procedure.

- **Elicitation phase**
  In the first phase of the empirical work a total of thirteen attributes were used as elements. These were:
  - Learnability
  - Understandability
  - Responsiveness
  - Safety
  - Adaptability
  - Trustworthiness
  - Accessibility
  - Excitement
  - Challenge
  - Effectiveness
  - Efficiency
  - Satisfaction
  - Complexity

The attributes were selected from a provisional list of 25 to give a spread between terminology that has appeared in standards documents, and others that had emerged from the user experience literature. The list was reduced to ten after 11 initial study sessions were carried out, with safety, effectiveness and responsiveness removed. The shortening of the list was to reduce the workload for subjects. The selection of these attributes for removal was made after the initial analysis shown in Appendix A below.

Cards containing the selected quality attributes were presented to the subject in randomly drawn triads. The subjects were asked to think of a dimension (a personal construct). Two of the three terms were grouped together with reference to the personal construct declared. The grouping was be labelled citing the similarity identified for the two (the inclusive construct-pole) and the contrasting nature of the third (the exclusive construct pole). The extraction phase was completed at the point where it is clear that no new constructs will be generated.

- **Assessment (Comparison) Phase**
  A ‘repertory grid questionnaire’ was constructed to further examine relationships between the bi-polar constructs and the set of elements. This gave each subject the chance to cross-reference the generated constructs with the entire element set. Each element was cross-referenced with the constructs generated on a scale of one to seven. A score of one represents the strongest possible association between the ‘pole 1’ and the element. A score of seven represents the strongest association between ‘pole 7’ and an element. A score of 4 is neutral. After the initial 11 sessions were analysed the scale was reduced to 5 points from 7 points. The subjects were asked to identify which of the construct poles can be classed as positive/negative. This is likely to be a subset of the generated construct poles.

- **Follow-up Elicitation**
  The constructs produced were examined further using a questioning strategy inspired by the Laddering technique (Rugg & McGeorge, 1995). This is aimed at eliciting richer information about the way in which the subjects specify meaning in particular contexts. Laddering prompts the subject to elaborate on the bi-polar construct by eliciting examples to which the construct refers. Similarly, Pyramiding prompts subjects to elaborate on the definitions expressed in constructs. Thus we elicit full explanation of the relationship between the quality terminology, its domain specific interpretation, and design particulars.
Experience in the early sessions resulted in a modification of the data collectors’ approach. It was found that subjects were typically offering an elaborate description of their reaction to the randomly draw triads. Rather than curtail or ignore these, they were recorded. The clipped construct descriptions were in most cases summaries of longer descriptions negotiated and agreed with the subjects. The follow up interviews tended to be less structured in the main group of subjects.

**Analysis of Results**

The analysis of results is scheduled for completion by the end of the COST294-MAUSE project in May 2009. What is reported here is the state of the analysis at the time of going to press (February 2009). Therefore a full statistical analysis is not reported.

**Procedure**

1. Semantic comparison: Generated constructs are assessed for semantic equivalence, and placed into groups based on subjective assessment of similarity. They are then given a topic heading.

2. Each element is profiled demonstrating topics and constructs associated with them.

3. Pathfinder Network Analyses are used to create a map showing links between the elements. These are generated for each of the parallel studies for comparison.

4. The construct lists for each participant are compared, along with their comparative ratings to gauge the degree of intra-agreement, and provide a general profile for that cell (the specific domain). Once this has been done for all the parallel studies, the profiles for each of the cells can be compared to assess the contrasts.

5. The follow-up evidence is subjected to qualitative analysis to extract anecdotal evidence of similarities, contrasts and the general ‘character’ of subjects’ constructs, and descriptions.

**Findings**

**Experience from Data Collection**

There was some degree of variation in the approach taken by subjects when relating to the attributes, in both cells. Some subjects described the attributes and generated constructs from the perspective of the analyst or designer. Other subjects tended to relate to the concepts from the perspective of an end user.

Some data collectors found difficulty in applying the Laddering technique as the imposition of structure on subjects seemed to have the effect of compromising their expansive descriptions of their constructs. Therefore the alternative of allowing less structured descriptions and analysing transcripts of subjects’ responses was favoured.

**Macro Analysis from February 2007**

The collected grid data has been subject to a Macro-level analysis in order to analyse the degree of consensus or diversity in the experts’ interpretations. The first of these was produced in February 2007, using an initial set of data from 11 subjects. The data were all from HCI experts. A Euclidian Distance model was generated. This is shown in Appendix A.

The model showed two Dimensions. Dimension 1 was the more complex of the two. The example of complexity versus efficiency shown in Appendix A showed efficiency as mainly positive over
negative complexity. Whilst at this stage it was not easy to assert a definitive explanation the scale from complexity (negative) to simplicity (positive) seemed a plausible explanation.

Dimension 2 appeared to be something like subjective/fun in a positive sense versus objective/soberness. The attribute Satisfaction was the closest to the centre of the graph, perhaps suggesting a unifying character between the more hedonic attributes of a system and those associated with efficiency and reliability.

**Macro Analysis from February 2009**

**Background**

There are altogether 19 datasets or grids: Seven from a group of Designers (Des) and eleven from a group of HCI researchers (HCI). Ten elements (i.e. quality attributes) are supplied and common across all the 19 grids whereas the constructs are generated by the respondents. The major goal of the following analyses is to observe the differences between these two groups of respondents with respect to their interpretations of the ten supplied elements.

Our assumptions are that ten quality attributes can be interpreted differently and that the degree of similarity in interpreting these attributes is higher within a group (designers or HCI researchers) than between the groups. Concisely speaking, we aim to answer the question “Do the different grids show similar or different patterns of relationships among the ten common elements?” Advanced statistical analysis techniques are employed to answer this question.

**Preliminary Descriptive Statistics**

As shown in Table 3.1 below, the total numbers of constructs elicited from the two groups differ, though statistically not significant. There are 90 and 114 constructs for Designers ($M = 11.2, SD = 2.9$) and HCI researchers ($M = 10.4, SD = 4.1$), respectively. Nonetheless, the value of this parameter is not a reliable or valid indicator for comparing the two groups as it is highly dependent on the setting of the interviews, such as interviewers’ skills (Note: four different interviews were involved in the study), time pressure, interviewees’ articulative ability, etc. Nonetheless, the lists are presented for references. More relevant is the meticulous content analysis of the constructs elicited, which will be performed at the later stage of this empirical work.

**Table 3.1: Number of constructs elicited per respondent**

<table>
<thead>
<tr>
<th>ID</th>
<th>No. of Constructs elicited</th>
<th>ID</th>
<th>No. of Constructs elicited</th>
</tr>
</thead>
<tbody>
<tr>
<td>Des1</td>
<td>10</td>
<td>HCI1</td>
<td>10</td>
</tr>
<tr>
<td>Des2</td>
<td>16</td>
<td>HCI2</td>
<td>18</td>
</tr>
<tr>
<td>Des3</td>
<td>13</td>
<td>HCI3</td>
<td>15</td>
</tr>
<tr>
<td>Des4</td>
<td>6</td>
<td>HCI4</td>
<td>8</td>
</tr>
<tr>
<td>Des5</td>
<td>12</td>
<td>HCI5</td>
<td>8</td>
</tr>
<tr>
<td>Des6</td>
<td>10</td>
<td>HCI6</td>
<td>10</td>
</tr>
<tr>
<td>Des7</td>
<td>11</td>
<td>HCI9</td>
<td>15</td>
</tr>
<tr>
<td>Des8</td>
<td>12</td>
<td>HCl</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HClb</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HClc</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HClD</td>
<td>7</td>
</tr>
</tbody>
</table>

**Advanced Statistical Analysis**

Two facets of a repertory grid are elements and constructs (Table 3.2; Bell, 1997). Each of these facets can be fixed or varied (i.e. under the control of the respondents), depending on the question of interest.
Our empirical work falls in the category of Case II, which is equivalent to Case III with one of the facets being replicated across the grids.

**Software Applications**

To analyse individual grids and compare across them, advanced statistical techniques are applied. Three common ones are cluster analysis, multidimensional scaling, and principal component analysis are adopted. To facilitate the process, we have employed different software applications, both proprietary like SPSS 16.0, and free ones specifically developed for grid analysis, viz. WebGrid III (Gaines & Shaw, 2008) and GRIDSCAL (Bell, 2002).

As suggested by its name, WebGrid III is a web-based application. It supports FOCUS cluster analysis (outputs as ‘clusters’) and multidimensional analysis (ALSCAL) (outputs as ‘maps’). FOCUS cluster analysis is a sound tool for deriving a hierarchical structure underlying a flat grid (Shaw & Gaines, n.d.). Furthermore, WebGrid III is a powerful tool to visualize the results of clustering and MDS, including multidimensional unfolding. However, it does not support multiple grid analysis. Another drawback is that individual data points have to be entered manually via the user interface, and such a data entry procedure is pretty tedious. In comparison, GRIDSCAL, which is specifically designed for multiple grid analysis, is more usable by enabling the data to be read directly from text files. Another advantage of GRIDSCAL is its ability to deal with the issue of confounding sources of variation between grids and either elements or constructs.

**Group-specific Analyses**

A five-point Likert scale has been employed for all the 19 cases except one (HCI2 where a seven-point scale was used). No re-scaling is performed because it is appropriate only when the scales of measurement in each of the matrices differ dramatically. Hence, it is justified to apply the common space analysis of multidimensional scaling (MDS). Re-scaling may make it impossible to compare the weights on separate dimensions. Euclidean distances but not correlations are used as measures of association. MDS is particularly useful in repertory grid analysis since it can provide analysis of data other than correlations. It is shown that correlations among elements are dependent on the orientation of constructs while Euclidean distances are not.

Subsequently we first illustrate the results for the eight datasets of designers (Des) then the eleven datasets of HCI researchers (HCI).

## Analysis of Designers

### Profiles of the Respondents

The profiles of the eight respondents in the group of Designers vary in terms of their academic degree and working experience whilst all of them have some pre-knowledge of the ten quality attributes investigated.
Table 3.3: Profiles of the Designers

<table>
<thead>
<tr>
<th>DESIGNERS/DEVELOPERS</th>
<th>ID</th>
<th>Academic Degree</th>
<th>Working Experience</th>
<th>Pre-Knowledge of the QA?</th>
<th>Application contexts of the Quality Attributes (QA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Des1</td>
<td></td>
<td>Collaborative Design Learning</td>
<td>4 years of practitioners in multimedia and web design, graphic design; 4 years of researchers in explorative design</td>
<td>Yes</td>
<td>Prototyping components for community website Observation and analysis of human interaction with computer systems</td>
</tr>
<tr>
<td>Des2</td>
<td></td>
<td>Textile Design</td>
<td>12 years teaching, research</td>
<td>Yes, not explicitly used</td>
<td>Digital media, digital art, design education, location-based media, design and emotion</td>
</tr>
<tr>
<td>Des3</td>
<td></td>
<td>Game design</td>
<td>4 years design</td>
<td>Yes</td>
<td>game design/graphics for mobile applications</td>
</tr>
<tr>
<td>Des4</td>
<td></td>
<td>Mechanical Engineering</td>
<td>14 years design and development</td>
<td>Yes, knowledge except excitement, challenge</td>
<td>Robotics, Mechatronics, Autonomous Land Systems, Military Technology, Unmanned Aerial Vehicles, Cross-Cultural Design</td>
</tr>
<tr>
<td>Des5</td>
<td></td>
<td>Product Design</td>
<td>4 years design, teaching</td>
<td>Yes, some knowledge</td>
<td>Product design, product customization</td>
</tr>
<tr>
<td>Des6</td>
<td></td>
<td>Product Design</td>
<td>3 years design, consultancy</td>
<td>Yes, some knowledge</td>
<td>Product design, car cockpit design</td>
</tr>
<tr>
<td>Des7</td>
<td></td>
<td>n/a</td>
<td>7 years design, design research</td>
<td>Yes, knowledge except excitement, challenge</td>
<td>Car cockpit design Automotive Ergonomics, Inclusive Design</td>
</tr>
<tr>
<td>Des8</td>
<td></td>
<td>Master in Computer science + communication</td>
<td>4 years as web developers and UI designer</td>
<td>Yes</td>
<td>User interface designer of human resource and salary systems</td>
</tr>
</tbody>
</table>

Descriptive Statistics

A summary of the eight matrices of the group Designers is shown below.

<table>
<thead>
<tr>
<th>Grid</th>
<th>Constructs</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Des1</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Des2</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>Des3</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>Des4</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>Des5</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>Des6</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Des7</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>Des8</td>
<td>12</td>
<td>10</td>
</tr>
</tbody>
</table>

A simple way to evaluate the source of variation in grid data is through the analysis of variance. Such an analysis also enables the calculation of the intra-class correlation. The intra-class correlation for constructs (or elements) is equivalent to the average or pooled correlation among all pairs of constructs (or elements). Bell & Keen (1981) suggest that this measure is a useful operationalization of the concept “cognitive complexity”. GRIDSCAL is used to compute these statistics for each grid separately and thus do not depend on both constructs and elements being common.
The output above shows clearly that this is a dataset where the constructs vary because the degrees of freedom (df) under ‘Construct’ change whereas those under ‘Element’ remain constant (i.e. df = 9). The ‘MS’ (Mean Squares) columns show the variances attributable to constructs and elements in each grid, and the variation in general is small for all the grids. The ICC correlation coefficients for both constructs and elements are small, implying that they are interpreted differently by the respondents.

The above statistics seem to imply the respondents tend to rate the ten elements moderately with respect to the constructs elicited. This observation is corroborated by the grid statistics illustrated below.

Furthermore, the table below shows the results of principal component analysis of the Designers datasets. If the relationships between elements are measured by correlation coefficients (which is the case for principal component analysis), then a simple way to assess whether the correlations are uniformly high, is via the first eigenvalue. If the first eigenvalue accounts for a substantial amount of variance, then it can be argued that the respondents are interpreting the elements in the same way. Here we can observe that the first eigenvalues of most grids (except grid#2 and grid#3) are fairly large.
Eigenvalues (as % of Trace) from Separate Scalings
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Sets defined by Grids

<table>
<thead>
<tr>
<th>Set</th>
<th>Trace</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10.00</td>
<td>75.85</td>
<td>13.49</td>
<td>4.91</td>
<td>0.98</td>
<td>0.73</td>
<td>0.58</td>
<td>0.27</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10.00</td>
<td>25.90</td>
<td>23.21</td>
<td>15.13</td>
<td>13.57</td>
<td>9.77</td>
<td>4.13</td>
<td>3.32</td>
<td>2.29</td>
<td>1.58</td>
</tr>
<tr>
<td>3</td>
<td>10.00</td>
<td>37.04</td>
<td>26.76</td>
<td>14.17</td>
<td>8.50</td>
<td>6.52</td>
<td>3.29</td>
<td>2.02</td>
<td>0.80</td>
<td>0.68</td>
</tr>
<tr>
<td>4</td>
<td>10.00</td>
<td>80.28</td>
<td>13.92</td>
<td>4.04</td>
<td>1.13</td>
<td>0.47</td>
<td>0.16</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>5</td>
<td>10.00</td>
<td>65.86</td>
<td>19.80</td>
<td>5.65</td>
<td>3.43</td>
<td>3.26</td>
<td>1.01</td>
<td>0.61</td>
<td>0.27</td>
<td>0.08</td>
</tr>
<tr>
<td>6</td>
<td>10.00</td>
<td>64.94</td>
<td>9.83</td>
<td>8.80</td>
<td>5.77</td>
<td>4.69</td>
<td>2.71</td>
<td>1.80</td>
<td>0.93</td>
<td>0.51</td>
</tr>
<tr>
<td>7</td>
<td>10.00</td>
<td>56.06</td>
<td>27.97</td>
<td>5.66</td>
<td>4.01</td>
<td>2.54</td>
<td>1.70</td>
<td>1.10</td>
<td>0.68</td>
<td>0.28</td>
</tr>
<tr>
<td>8</td>
<td>10.00</td>
<td>57.62</td>
<td>18.56</td>
<td>13.59</td>
<td>4.63</td>
<td>3.00</td>
<td>1.81</td>
<td>0.34</td>
<td>0.26</td>
<td>0.17</td>
</tr>
</tbody>
</table>

Another way of examining the distinction between the first and subsequent eigenvalues is to compare the differences between adjacent eigenvalues. A simpler index of this is the ratio of the difference between the first and second factors, over the difference between the first and third factors. Such a value will have a value of 1.0 when the first and second factors differ but not the second and third; and a value of 0.5 when the second and third factors differ as much as the first and second factors. It is computed as First Factor Differentiator (FFD Index) in GRIDSCAL. Here we can see that for several grids (#2, #4, #6, #8) the first eigenvalues are very clearly differentiated from the following two, indicating that there is a major source of agreement about these particular grids.

First Factor Differentiation
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

<table>
<thead>
<tr>
<th>Set</th>
<th>FFD Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.88</td>
</tr>
<tr>
<td>2</td>
<td>0.25</td>
</tr>
<tr>
<td>3</td>
<td>0.45</td>
</tr>
<tr>
<td>4</td>
<td>0.87</td>
</tr>
<tr>
<td>5</td>
<td>0.76</td>
</tr>
<tr>
<td>6</td>
<td>0.98</td>
</tr>
<tr>
<td>7</td>
<td>0.56</td>
</tr>
<tr>
<td>8</td>
<td>0.89</td>
</tr>
</tbody>
</table>

The importance of the results of first factor loadings is to observe the extent to which they vary across elements within each set. To the extent that an element factor loading is high, it implies that the empirically derived first principal component is important for all respondents with respect to this element. If it is low, it suggests that respondents have different ideas about the importance of this common underlying component for their ratings of this element. Furthermore, the low values of standard deviation (Std Dev) in the cases of Des1 (= 0.11) and Des6 (= 0.08) imply that they tend to interpret the ten elements in a similar way. These observations are consistent with the FFD Index discussed above.

First Factor Loadings from Separate Scalings
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Sets defined by Grids

<table>
<thead>
<tr>
<th>Set</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.88 0.88 0.76 0.83 0.77 0.97 0.85 0.90 0.94 0.90</td>
</tr>
<tr>
<td>2</td>
<td>-0.31 0.03 0.80 -0.64 0.77 -0.68 0.23 -0.49 -0.06 0.31</td>
</tr>
<tr>
<td>3</td>
<td>0.51 0.39 0.61 0.59 0.67 0.55 0.68 0.70 0.70 0.62</td>
</tr>
<tr>
<td>4</td>
<td>0.91 0.96 0.81 0.79 0.98 0.78 0.82 1.00 0.82</td>
</tr>
<tr>
<td>5</td>
<td>0.83 0.90 0.92 0.79 0.67 0.75 0.78 0.85 0.72 0.87</td>
</tr>
<tr>
<td>6</td>
<td>0.91 0.83 0.79 0.76 0.82 0.72 0.82 0.77 0.79 0.82</td>
</tr>
<tr>
<td>7</td>
<td>0.72 0.83 0.79 0.75 0.56 0.88 0.75 0.75 0.59 0.81</td>
</tr>
<tr>
<td>8</td>
<td>0.88 0.89 0.11 0.81 0.61 0.93 0.86 0.76 0.94 0.33</td>
</tr>
</tbody>
</table>
The latent root (eigenvalue) displayed below shows that two most important factors explain 48.5% of the variance. Factor one explains 28.6% and factor two 20%.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Root</th>
<th>Percent of Trace</th>
<th>Cum. Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>76.505</td>
<td>28.577</td>
<td>28.577</td>
</tr>
<tr>
<td>2</td>
<td>53.337</td>
<td>19.923</td>
<td>48.501</td>
</tr>
<tr>
<td>3</td>
<td>34.257</td>
<td>12.796</td>
<td>61.297</td>
</tr>
<tr>
<td>4</td>
<td>28.433</td>
<td>10.621</td>
<td>71.918</td>
</tr>
<tr>
<td>5</td>
<td>18.684</td>
<td>6.979</td>
<td>78.897</td>
</tr>
<tr>
<td>6</td>
<td>17.604</td>
<td>5.085</td>
<td>85.084</td>
</tr>
<tr>
<td>7</td>
<td>14.823</td>
<td>5.085</td>
<td>91.009</td>
</tr>
<tr>
<td>8</td>
<td>13.613</td>
<td>3.906</td>
<td>96.094</td>
</tr>
<tr>
<td>9</td>
<td>10.457</td>
<td>3.906</td>
<td>100.000</td>
</tr>
<tr>
<td>10</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
</tr>
</tbody>
</table>

Common Coordinate Matrix for Elements

<table>
<thead>
<tr>
<th>Label</th>
<th>Dim1</th>
<th>Dim2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Adaptability</td>
<td>-3.00</td>
<td>-0.24</td>
</tr>
<tr>
<td>2 Responsiveness</td>
<td>-2.49</td>
<td>-0.53</td>
</tr>
<tr>
<td>3 Challenge</td>
<td>2.83</td>
<td>4.17</td>
</tr>
<tr>
<td>4 Accessibility</td>
<td>-2.49</td>
<td>-2.49</td>
</tr>
<tr>
<td>5 Excitement</td>
<td>3.61</td>
<td>-1.47</td>
</tr>
<tr>
<td>6 Efficiency</td>
<td>-2.62</td>
<td>-0.28</td>
</tr>
<tr>
<td>7 Trustworthiness</td>
<td>2.24</td>
<td>-1.49</td>
</tr>
<tr>
<td>8 Learnability</td>
<td>-1.79</td>
<td>-0.85</td>
</tr>
<tr>
<td>9 Satisfaction</td>
<td>2.93</td>
<td>-2.21</td>
</tr>
<tr>
<td>10 Complexity</td>
<td>0.78</td>
<td>5.39</td>
</tr>
</tbody>
</table>

A first analytical task is to label these factors. Factor one strongly loads *Excitement—Adaptability*. Other elements such as *Satisfaction*, *Efficiency* and *Challenge* are also strongly correlated with this factor. Factor 2 strongly loads on *Complexity—Accessibility*. The diagram below suggests that the Designers tend to interpret the attributes “Adaptability”, “Efficiency”, “Responsiveness” and “Learnability” very similarly. In contrast, they construe the attributes “Complexity”, “Challenge”, and “Excitement” very differently. Further detailed analysis of these two factors with their associated constructs will be conducted later.
In the graph below we illustrate the association (commonality) between the designers with respect to the two dimensions. Apparently, three of the Designers (i.e. Des4, Des1 and Des6) tend to interpret the ten elements in a similar way, whereas the others are differentiated from each other.

Figure 3.1: Multidimensional scaling analysis on the Designers’ construing of the ten quality attributes

Figure 3.2: Distributions of Designers’ grids with respect to the two principal dimensions
Besides, we look at other indicators such as variance and Fit index. The column “Variance explained” is somewhat self-explanatory. It shows the proportion of variance (in the matrix of associations for the chosen facet – the ten quality attributes in our case, not the grids themselves) that the common solution accounts for. The Fit index is a measure of the degree to which the assumptions of the model (i.e. the commonality of construing) are violated. Low values indicate the assumptions of the model are met.

Fit Measures for Common Space Solution
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
<table>
<thead>
<tr>
<th>Grids</th>
<th>Variance Explained</th>
<th>Fit</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.36</td>
<td>0.02</td>
<td>Des1</td>
</tr>
<tr>
<td>2</td>
<td>0.40</td>
<td>0.33</td>
<td>Des2</td>
</tr>
<tr>
<td>3</td>
<td>0.74</td>
<td>0.24</td>
<td>Des3</td>
</tr>
<tr>
<td>4</td>
<td>0.53</td>
<td>0.45</td>
<td>Des4</td>
</tr>
<tr>
<td>5</td>
<td>0.57</td>
<td>0.69</td>
<td>Des5</td>
</tr>
<tr>
<td>6</td>
<td>0.11</td>
<td>0.28</td>
<td>Des6</td>
</tr>
<tr>
<td>7</td>
<td>0.88</td>
<td>0.02</td>
<td>Des7</td>
</tr>
<tr>
<td>8</td>
<td>0.71</td>
<td>0.67</td>
<td>Des8</td>
</tr>
</tbody>
</table>

We look closely at the cases with the highest Fit index (Des5) and lowest Fit index (Des1). First, we show the results of cluster analysis obtained from WebGridIII (correlations are the measure of association) and then the ‘map’ derived from the principal component analysis, which is comparable to multidimensional analysis except it is based on correlations rather than Euclidean distances.

Individual Grid Analysis: Des1

**FOCUS cluster analysis: Hierarchy structure underlying the grid Des1**

The values of the above clusters are expressed as “element matches”, which are expressed in terms of percent; the lower these values, the more dissimilar the elements are. This is useful because it provides more precise information than the links given in the "red" dendrogram above.
The above clusters and values are computed using the FOCUS algorithm (Shaw, 1980). They correspond to those derived from the Hierarchical Cluster Analysis, as shown in the dendrogram below.

**Dendrogram using Single Linkage (SPSS 16.0)**

**Rescaled Distance Cluster Combine**

<table>
<thead>
<tr>
<th>CASE</th>
<th>Label</th>
<th>0</th>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Responsi</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Trust</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Access</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Efficien</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Satisfac</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adapt</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Challeng</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Learn</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Complexi</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Exciteme</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Principal Component Analysis of Des1 (based on correlations)**

For the sake of illustration, the ten constructs elicited from Des1 and their correlations are listed as follows:
Correlations across the ten constructs elicited

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.246</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-0.801</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.889</td>
<td>-0.791</td>
<td>0.535</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.777</td>
<td>-0.875</td>
<td>0.146</td>
<td>0.855</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-0.525</td>
<td>0.586</td>
<td>0.140</td>
<td>-0.391</td>
<td>-0.382</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.669</td>
<td>-0.634</td>
<td>0.537</td>
<td>0.817</td>
<td>0.644</td>
<td>-0.106</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>-0.067</td>
<td>0.000</td>
<td>-0.408</td>
<td>0.243</td>
<td>0.060</td>
<td>0.401</td>
<td>-0.146</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.460</td>
<td>-0.475</td>
<td>0.688</td>
<td>0.707</td>
<td>0.459</td>
<td>0.046</td>
<td>0.606</td>
<td>-0.281</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.368</td>
<td>-0.102</td>
<td>-0.220</td>
<td>0.094</td>
<td>0.077</td>
<td>-0.657</td>
<td>-0.047</td>
<td>-0.179</td>
<td>-0.472</td>
</tr>
</tbody>
</table>

**Indivdual Grid Analysis: Des5**

**FOCUS cluster analysis: Hierarchy structure underlying the grid Des5**

- **FOCUS OA, Domain: HCI**
- **Context:** Quality, 10 Qualities, 12 Properties

![Diagram showing hierarchy structure with numbered nodes and edges representing correlations and constructs.](image-url)
Element matches of Des5

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>*</td>
<td>60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>65</td>
<td>92</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>*</td>
<td>81</td>
<td>67</td>
<td>67</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>46</td>
<td>77</td>
<td>73</td>
<td>44</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>*</td>
<td>88</td>
<td>56</td>
<td>60</td>
<td>73</td>
<td>46</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>50</td>
<td>73</td>
<td>73</td>
<td>48</td>
<td>83</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>*</td>
<td>79</td>
<td>69</td>
<td>69</td>
<td>77</td>
<td>50</td>
<td>75</td>
<td>54</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>44</td>
<td>67</td>
<td>58</td>
<td>42</td>
<td>73</td>
<td>44</td>
<td>73</td>
<td>52</td>
</tr>
<tr>
<td>10</td>
<td>*</td>
<td>73</td>
<td>71</td>
<td>75</td>
<td>75</td>
<td>52</td>
<td>77</td>
<td>60</td>
<td>65</td>
</tr>
</tbody>
</table>

The above clusters and values are computed using the FOCUS algorithm (Shaw, 1980). They correspond to those derived from the Hierarchical Cluster Analysis, as shown in the dendrogram below.

Dendrogram using Single Linkage
Principal Component Analysis of Des5

For the sake of illustration, the twelve constructs elicited from Des5 and their correlations are listed as follows:

<table>
<thead>
<tr>
<th>No.</th>
<th>Pole1</th>
<th>Pole5</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>about designing for use</td>
<td>about feedback from the system</td>
</tr>
<tr>
<td>#2</td>
<td>about perceived reliability/quality</td>
<td>could be positive or negative dependent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>on context</td>
</tr>
<tr>
<td>#3</td>
<td>about functionality and usability</td>
<td>about user perception and reaction</td>
</tr>
<tr>
<td>#4</td>
<td>reaction to use of a product</td>
<td>contemplation of a product</td>
</tr>
<tr>
<td>#5</td>
<td>conceptual quality of product</td>
<td>measurable quality of a product</td>
</tr>
<tr>
<td>#6</td>
<td>ux design target</td>
<td>emotional feedback from user</td>
</tr>
<tr>
<td>#7</td>
<td>functional design criteria</td>
<td>experience within use</td>
</tr>
<tr>
<td>#8</td>
<td>tangible design criteria</td>
<td>perceptions of a design</td>
</tr>
<tr>
<td>#9</td>
<td>to do with interaction</td>
<td>emotional reaction</td>
</tr>
<tr>
<td>#10</td>
<td>about understanding the user</td>
<td>tangible interaction criteria</td>
</tr>
<tr>
<td>#11</td>
<td>measurable design criteria</td>
<td>intangible user reaction</td>
</tr>
<tr>
<td>#12</td>
<td>product interaction with user</td>
<td>product functionality in itself</td>
</tr>
</tbody>
</table>

Correlations across the twelve constructs elicited, ranging from 0.0 (the construct pair #8-#10: tangible design criteria-perceptions of a design and about understanding the user-tangible interaction criteria) to 0.931 (the construct pair #4-#5: reaction to use a product – contemplation of a product and conceptual quality of product – measurable quality of a product)
## Analysis of HCI Professionals

### Profiles of the Respondents

Note that the background information of the four of the respondents, viz. HCIa, HCIb, HCIc and HCId, is missing.

<table>
<thead>
<tr>
<th>HCI SPECIALISTS</th>
<th>ID</th>
<th>Academic Degree</th>
<th>Experience</th>
<th>Pre-Knowledge of the QA?</th>
<th>Application contexts of the Quality Attributes (QA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCI1</td>
<td>HCI</td>
<td>Physics</td>
<td>14 years researcher, teacher HCI, consultancy</td>
<td>Yes</td>
<td>Digital libraries, safety critical processes, educational multimedia</td>
</tr>
<tr>
<td>HCI2</td>
<td>HCI</td>
<td>HCI and user experience with mobiles</td>
<td>4 years HCI research, teaching, consultancy</td>
<td>Yes, but not perfect on definitions</td>
<td>Mobile phones, social software, educational multimedia</td>
</tr>
<tr>
<td>HCI3</td>
<td>HCI</td>
<td>Software Engineering: Safety critical systems</td>
<td>16 years HCI research</td>
<td>Yes, from standards/literature except excitement, challenge</td>
<td>Power control systems, emergency ambulance dispatch, air-traffic control, public information systems</td>
</tr>
<tr>
<td>HCI4</td>
<td>HCI</td>
<td>Computer ethics</td>
<td>11 years, end-user HCI</td>
<td>Yes, aware, no explicit knowledge</td>
<td>End-user systems, secure systems</td>
</tr>
<tr>
<td>HCI5</td>
<td>HCI</td>
<td>Ergonomics</td>
<td>18 years HCI/ergonomics researcher</td>
<td>Yes</td>
<td>Digital libraries, industrial ergonomics, design for universal access to end-user systems and ubiquitous applications, car cockpit design</td>
</tr>
<tr>
<td>HCI7</td>
<td>HCI</td>
<td>Cognitive Psychology</td>
<td>12 years research, consultancy and teaching</td>
<td>Yes, sometimes use them</td>
<td>Digital libraries</td>
</tr>
<tr>
<td>HCI9</td>
<td>HCI</td>
<td>Collaborative working</td>
<td>22 years HCI research, teaching</td>
<td>Yes</td>
<td>Collaborative user support, computer-supported collaborative learning and computer-supported cooperative work</td>
</tr>
</tbody>
</table>

### Descriptive Statistics

The grid numbers correspond to the order of the respondents coded above, i.e. Grid 1 = HCI1, Grid2 = HCI2, so and so forth.

<table>
<thead>
<tr>
<th>Grid Constructs Elements</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>18</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>15</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>7</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>
The ‘MS’ (Mean Squares) columns show the variances attributable to constructs and elements in each grid, and the variation in general is small for all the grids except grid#10 (i.e. HCIc) in the case of construct and grid#9 (i.e. HCIb). The ICC correlation coefficients for both constructs and elements are small, implying that they are interpreted differently by the respondents.

**Element Statistics**

<table>
<thead>
<tr>
<th>Element No.</th>
<th>Mean</th>
<th>Std Dev</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.5</td>
<td>1.40</td>
<td>Adaptability</td>
</tr>
<tr>
<td>2</td>
<td>2.5</td>
<td>1.43</td>
<td>Responsiveness</td>
</tr>
<tr>
<td>3</td>
<td>3.6</td>
<td>1.55</td>
<td>Challenge</td>
</tr>
<tr>
<td>4</td>
<td>2.5</td>
<td>1.42</td>
<td>Accessibility</td>
</tr>
<tr>
<td>5</td>
<td>3.5</td>
<td>1.53</td>
<td>Excitement</td>
</tr>
<tr>
<td>6</td>
<td>2.8</td>
<td>1.75</td>
<td>Efficiency</td>
</tr>
<tr>
<td>7</td>
<td>3.0</td>
<td>1.50</td>
<td>Trustworthiness</td>
</tr>
<tr>
<td>8</td>
<td>2.5</td>
<td>1.45</td>
<td>Learnability</td>
</tr>
<tr>
<td>9</td>
<td>3.0</td>
<td>1.40</td>
<td>Satisfaction</td>
</tr>
<tr>
<td>10</td>
<td>3.5</td>
<td>1.65</td>
<td>Complexity</td>
</tr>
</tbody>
</table>

The above statistics seem to imply the respondents tend to rate the ten elements moderately with respect to the constructs elicited. This observation is corroborated by the grid statistics illustrated below.

**Grid Statistics**

<table>
<thead>
<tr>
<th>Grid No.</th>
<th>Mean</th>
<th>Std Dev</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.6</td>
<td>1.34</td>
<td>HCI1</td>
</tr>
<tr>
<td>2</td>
<td>2.7</td>
<td>1.30</td>
<td>HCI2</td>
</tr>
<tr>
<td>3</td>
<td>3.0</td>
<td>1.50</td>
<td>HCI3</td>
</tr>
<tr>
<td>4</td>
<td>2.9</td>
<td>1.57</td>
<td>HCI4</td>
</tr>
<tr>
<td>5</td>
<td>3.1</td>
<td>1.69</td>
<td>HCI5</td>
</tr>
<tr>
<td>6</td>
<td>2.4</td>
<td>1.54</td>
<td>HCI7</td>
</tr>
<tr>
<td>7</td>
<td>3.0</td>
<td>1.50</td>
<td>HCI9</td>
</tr>
<tr>
<td>8</td>
<td>3.3</td>
<td>1.25</td>
<td>HCIa</td>
</tr>
<tr>
<td>9</td>
<td>3.0</td>
<td>1.72</td>
<td>HCIb</td>
</tr>
<tr>
<td>10</td>
<td>4.1</td>
<td>2.48</td>
<td>HCIc</td>
</tr>
<tr>
<td>11</td>
<td>3.1</td>
<td>1.67</td>
<td>HCId</td>
</tr>
</tbody>
</table>

Furthermore, the table below shows the results of principal component analysis of the Designers datasets. If the relationships between elements are measured by correlation coefficients (which is the case for principal component analysis), then a simple way to assess whether the correlations are uniformly high, is via the first eigenvalue. If the first eigenvalue accounts for a substantial amount of variance, then it can be argued that the respondents are interpreting the elements in the same way. Here we can observe that the first eigenvalues of most grids (except grid#2, #3 and #7) are fairly large.
Eigenvalues (as % of Trace) from Separate Scalings (Set defined by Grids)

<table>
<thead>
<tr>
<th>Eigenvalue No.</th>
<th>Set</th>
<th>Trace</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10.00</td>
<td>68.94</td>
<td>11.98</td>
<td>7.91</td>
<td>4.68</td>
<td>3.30</td>
<td>2.44</td>
<td>0.35</td>
<td>0.23</td>
<td>0.14</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10.00</td>
<td>33.75</td>
<td>28.96</td>
<td>13.92</td>
<td>9.48</td>
<td>6.08</td>
<td>4.25</td>
<td>1.55</td>
<td>1.13</td>
<td>0.65</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10.00</td>
<td>46.53</td>
<td>23.31</td>
<td>11.80</td>
<td>7.57</td>
<td>4.80</td>
<td>2.54</td>
<td>1.28</td>
<td>1.11</td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10.00</td>
<td>72.44</td>
<td>11.19</td>
<td>6.58</td>
<td>5.34</td>
<td>2.14</td>
<td>1.06</td>
<td>1.05</td>
<td>0.20</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10.00</td>
<td>74.30</td>
<td>10.54</td>
<td>7.25</td>
<td>4.24</td>
<td>1.99</td>
<td>1.30</td>
<td>0.30</td>
<td>0.08</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10.00</td>
<td>70.02</td>
<td>13.13</td>
<td>5.77</td>
<td>4.14</td>
<td>2.36</td>
<td>2.17</td>
<td>1.50</td>
<td>0.57</td>
<td>0.22</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>10.00</td>
<td>46.53</td>
<td>23.31</td>
<td>11.80</td>
<td>7.57</td>
<td>4.80</td>
<td>2.54</td>
<td>1.28</td>
<td>1.11</td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>10.00</td>
<td>87.27</td>
<td>5.60</td>
<td>3.15</td>
<td>2.03</td>
<td>0.85</td>
<td>0.61</td>
<td>0.26</td>
<td>0.22</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10.00</td>
<td>89.54</td>
<td>4.43</td>
<td>2.79</td>
<td>1.44</td>
<td>1.02</td>
<td>0.31</td>
<td>0.27</td>
<td>0.15</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10.00</td>
<td>79.92</td>
<td>13.73</td>
<td>4.50</td>
<td>1.84</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>10.00</td>
<td>83.44</td>
<td>7.89</td>
<td>4.59</td>
<td>1.98</td>
<td>1.81</td>
<td>0.26</td>
<td>0.05</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
</tbody>
</table>

Another way of examining the distinction between the first and subsequent eigenvalues is to compare the differences between adjacent eigenvalues. A simpler index of this is the ratio of the difference between the first and second factors, over the different between the first and third factors. Such a value will have a value of 1.0 when the first and second factors differ but not the second and third; and a value of 0.5 when the second and third factors differ as much as the first and second factors. It is computed as First Factor Differentiator (FFD Index) in GRIDSCAL. Here we can see that for most of the grids except #2 (HCI2), #3 (HCI3) and #7 (HCI9) the first eigenvalues are very clearly differentiated from the following two, indicating that there is a major source of agreement about these particular grids.

First Factor Differentiation

<table>
<thead>
<tr>
<th>Set</th>
<th>FFD Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.93</td>
</tr>
<tr>
<td>2</td>
<td>0.24</td>
</tr>
<tr>
<td>3</td>
<td>0.67</td>
</tr>
<tr>
<td>4</td>
<td>0.93</td>
</tr>
<tr>
<td>5</td>
<td>0.95</td>
</tr>
<tr>
<td>6</td>
<td>0.89</td>
</tr>
<tr>
<td>7</td>
<td>0.67</td>
</tr>
<tr>
<td>8</td>
<td>0.97</td>
</tr>
<tr>
<td>9</td>
<td>0.98</td>
</tr>
<tr>
<td>10</td>
<td>0.88</td>
</tr>
<tr>
<td>11</td>
<td>0.96</td>
</tr>
</tbody>
</table>

The importance of the results of first factor loadings is to observe the extent to which they vary across elements within each set. To the extent that an element factor loading is high, it implies that the empirically derived first principal component is important for all respondents with respect to this element. If it is low, it suggests that respondents have different ideas about the importance of this common underlying component for their ratings of this element.

First Factor Loadings from Separate Scalings (Set defined by Grids)

<table>
<thead>
<tr>
<th>Elements</th>
<th>Set</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10.00</td>
<td>0.86</td>
<td>0.80</td>
<td>0.84</td>
<td>0.80</td>
<td>0.84</td>
<td>0.84</td>
<td>0.92</td>
<td>0.85</td>
<td>0.86</td>
<td>0.68</td>
</tr>
<tr>
<td>2</td>
<td>10.00</td>
<td>0.61</td>
<td>0.39</td>
<td>0.93</td>
<td>-0.19</td>
<td>-0.14</td>
<td>0.83</td>
<td>-0.82</td>
<td>0.10</td>
<td>-0.22</td>
<td>0.70</td>
</tr>
<tr>
<td>3</td>
<td>10.00</td>
<td>0.66</td>
<td>0.67</td>
<td>0.97</td>
<td>0.75</td>
<td>0.65</td>
<td>0.66</td>
<td>0.78</td>
<td>0.74</td>
<td>0.79</td>
<td>0.53</td>
</tr>
<tr>
<td>4</td>
<td>10.00</td>
<td>0.71</td>
<td>0.72</td>
<td>0.91</td>
<td>0.90</td>
<td>0.93</td>
<td>0.77</td>
<td>0.88</td>
<td>0.86</td>
<td>0.92</td>
<td>0.87</td>
</tr>
<tr>
<td>5</td>
<td>10.00</td>
<td>0.85</td>
<td>0.84</td>
<td>0.80</td>
<td>0.93</td>
<td>0.81</td>
<td>0.87</td>
<td>0.90</td>
<td>0.90</td>
<td>0.75</td>
<td>0.95</td>
</tr>
<tr>
<td>6</td>
<td>10.00</td>
<td>0.86</td>
<td>0.86</td>
<td>0.91</td>
<td>0.83</td>
<td>0.95</td>
<td>0.76</td>
<td>0.84</td>
<td>0.83</td>
<td>0.79</td>
<td>0.71</td>
</tr>
<tr>
<td>7</td>
<td>10.00</td>
<td>0.66</td>
<td>0.67</td>
<td>0.54</td>
<td>0.75</td>
<td>0.65</td>
<td>0.66</td>
<td>0.78</td>
<td>0.74</td>
<td>0.79</td>
<td>0.53</td>
</tr>
<tr>
<td>8</td>
<td>10.00</td>
<td>0.94</td>
<td>0.95</td>
<td>0.97</td>
<td>0.86</td>
<td>0.93</td>
<td>0.89</td>
<td>0.95</td>
<td>0.93</td>
<td>0.96</td>
<td>0.96</td>
</tr>
<tr>
<td>9</td>
<td>10.00</td>
<td>0.94</td>
<td>0.96</td>
<td>0.95</td>
<td>0.94</td>
<td>0.98</td>
<td>0.99</td>
<td>0.92</td>
<td>0.91</td>
<td>0.93</td>
<td>0.95</td>
</tr>
<tr>
<td>10</td>
<td>10.00</td>
<td>0.90</td>
<td>0.78</td>
<td>0.92</td>
<td>0.73</td>
<td>0.96</td>
<td>0.79</td>
<td>0.95</td>
<td>0.98</td>
<td>0.93</td>
<td>0.94</td>
</tr>
<tr>
<td>11</td>
<td>10.00</td>
<td>0.95</td>
<td>0.92</td>
<td>0.88</td>
<td>0.88</td>
<td>0.96</td>
<td>0.95</td>
<td>0.91</td>
<td>0.95</td>
<td>0.95</td>
<td>0.77</td>
</tr>
</tbody>
</table>
Furthermore, the low values of standard deviation (Std Dev) in most of the cases imply that most respondents tend to interpret the ten elements in a similar way. These observations are consistent with the FFD Index discussed above.

### Standard Deviation of 1st Factor Loadings by Set
<table>
<thead>
<tr>
<th>Set</th>
<th>Std Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.09</td>
</tr>
<tr>
<td>2</td>
<td>0.31</td>
</tr>
<tr>
<td>3</td>
<td>0.12</td>
</tr>
<tr>
<td>4</td>
<td>0.13</td>
</tr>
<tr>
<td>5</td>
<td>0.10</td>
</tr>
<tr>
<td>6</td>
<td>0.11</td>
</tr>
<tr>
<td>7</td>
<td>0.12</td>
</tr>
<tr>
<td>8</td>
<td>0.06</td>
</tr>
<tr>
<td>9</td>
<td>0.04</td>
</tr>
<tr>
<td>10</td>
<td>0.14</td>
</tr>
<tr>
<td>11</td>
<td>0.09</td>
</tr>
</tbody>
</table>

The latent root (eigenvalue) displayed below shows that two most important factors explain 52.1% of the variance. Factor one explains 36.4% and factor two 15.7%.

<table>
<thead>
<tr>
<th>Factor Root</th>
<th>Percent of Trace</th>
<th>Cum. Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>77.686</td>
<td>36.378</td>
</tr>
<tr>
<td>2</td>
<td>33.594</td>
<td>52.108</td>
</tr>
<tr>
<td>3</td>
<td>24.323</td>
<td>63.498</td>
</tr>
<tr>
<td>4</td>
<td>18.926</td>
<td>72.360</td>
</tr>
<tr>
<td>5</td>
<td>16.300</td>
<td>79.993</td>
</tr>
<tr>
<td>6</td>
<td>13.717</td>
<td>86.416</td>
</tr>
<tr>
<td>7</td>
<td>12.135</td>
<td>92.099</td>
</tr>
<tr>
<td>8</td>
<td>10.099</td>
<td>96.828</td>
</tr>
<tr>
<td>9</td>
<td>6.775</td>
<td>100.000</td>
</tr>
</tbody>
</table>

A first analytical task is to label these factors. Factor one strongly loads **Challenge—Accessibility**. Factor 2 strongly loads on **Trustworthiness—Efficiency**. Detailed analysis of these two factors with their associated constructs will be conducted later. As shown in Figure 3.3 below, the scattered distribution of the attributes along the two dimensions suggests that the HCI professionals interpret them relatively divergently.

<table>
<thead>
<tr>
<th>Elements</th>
<th>Factor1</th>
<th>Factor2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Adaptability</td>
<td>-1.86</td>
<td>-0.41</td>
</tr>
<tr>
<td>2 Responsiveness</td>
<td>-2.14</td>
<td>-1.97</td>
</tr>
<tr>
<td>3 Challenge</td>
<td>4.38</td>
<td>0.07</td>
</tr>
<tr>
<td>4 Accessibility</td>
<td>-3.01</td>
<td>0.29</td>
</tr>
<tr>
<td>5 Excitement</td>
<td>3.63</td>
<td>1.59</td>
</tr>
<tr>
<td>6 Efficiency</td>
<td>-2.34</td>
<td>-2.81</td>
</tr>
<tr>
<td>7 Trustworthiness</td>
<td>-0.77</td>
<td>2.69</td>
</tr>
<tr>
<td>8 Learnability</td>
<td>-2.23</td>
<td>0.33</td>
</tr>
<tr>
<td>9 Satisfaction</td>
<td>0.26</td>
<td>2.58</td>
</tr>
<tr>
<td>10 Complexity</td>
<td>4.07</td>
<td>-2.36</td>
</tr>
</tbody>
</table>
Figure 3.3: Multidimensional analysis on the HCI Professionals’ construing of the ten quality-attributes

Figure 3.4: Distributions of HCI Professionals’ Grids with respect to the two principal dimensions

In the above Figure 3.4 we illustrate the association (commonality) between the designers with respect to the two dimensions. Apparently, there are several subgroups of HCI professionals (e.g. HCI3 and HCI9; HCI1 and HCI4) tend to interpret the ten elements in a similar way, whereas the others are differentiated from each other.
Besides, we look at other indicators such as variance and Fit index. The column “Variance explained” is somewhat self-explanatory. It shows the proportion of variance (in the matrix of associations for the chosen facet – the ten quality attributes in our case, not the grids themselves) that the common solution accounts for. The Fit index is a measure of the degree to which the assumptions of the model (i.e. the commonality of construing) are violated. Low values indicate the assumptions of the model are met.

### Fit Measures for Common Space Solution

<table>
<thead>
<tr>
<th>Grids</th>
<th>Variance Explained</th>
<th>Fit</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.44</td>
<td>0.15</td>
<td>HCI-1</td>
</tr>
<tr>
<td>2</td>
<td>0.36</td>
<td>0.67</td>
<td>HCI-2</td>
</tr>
<tr>
<td>3</td>
<td>0.73</td>
<td>0.10</td>
<td>HCI-3</td>
</tr>
<tr>
<td>4</td>
<td>0.37</td>
<td>0.23</td>
<td>HCI-4</td>
</tr>
<tr>
<td>5</td>
<td>0.16</td>
<td>0.06</td>
<td>HCI-5</td>
</tr>
<tr>
<td>6</td>
<td>0.69</td>
<td>0.27</td>
<td>HCI-7</td>
</tr>
<tr>
<td>7</td>
<td>0.73</td>
<td>0.10</td>
<td>HCI-9</td>
</tr>
<tr>
<td>8</td>
<td>0.08</td>
<td>0.42</td>
<td>HCI-a</td>
</tr>
<tr>
<td>9</td>
<td>0.71</td>
<td>0.51</td>
<td>HCI-b</td>
</tr>
<tr>
<td>10</td>
<td>0.68</td>
<td>0.56</td>
<td>HCI-c</td>
</tr>
<tr>
<td>11</td>
<td>0.66</td>
<td>0.87</td>
<td>HCI-d</td>
</tr>
</tbody>
</table>

We look closely at the cases with the lowest Fit index (HCI5), which is an interesting case. First, we show the results of cluster analysis obtained from WebGridIII (correlations are the measure of association) and then the ‘map’ derived from the principal component analysis, which is comparable to multidimensional analysis except it is based on correlations rather than Euclidean distances.

### Individual Grid Analysis: HCI-5

**FOCUS cluster analysis: Hierarchy structure underlying the grid HCI-5**

Interesting, this respondent only gives the ratings ‘1’, ‘3’ or ‘5’, without any ‘2’ or ‘4’. For the first construct “perceived ease of use – appropriateness of demands on users”, he consistently gives ‘5’ to all the ten quality attributes.
Element matches of HCI5

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Principal Component Analysis of HCI5

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
For the sake of illustration, the ten constructs elicited from HCI5 and their correlations are listed as follows:

<table>
<thead>
<tr>
<th>No.</th>
<th>Pole1</th>
<th>Pole5</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1.</td>
<td>Perceived ease of use</td>
<td>Appropriateness of demands on users</td>
</tr>
<tr>
<td>#2.</td>
<td>Desired quality of interaction</td>
<td>Desired end state</td>
</tr>
<tr>
<td>#3.</td>
<td>Ability to perform action</td>
<td>Quality of interaction</td>
</tr>
<tr>
<td>#4.</td>
<td>Feeling on completing task</td>
<td>Feeling performing task</td>
</tr>
<tr>
<td>#5.</td>
<td>Components of experience</td>
<td>Outcome of experience</td>
</tr>
<tr>
<td>#6.</td>
<td>Crucial to performance</td>
<td>Measure of performance</td>
</tr>
<tr>
<td>#7.</td>
<td>Necessary feeling for interaction to happen</td>
<td>A feeling during interaction</td>
</tr>
<tr>
<td>#8.</td>
<td>Components of experience</td>
<td>Outcome of experience</td>
</tr>
</tbody>
</table>

Correlations among the constructs elicited by HCI5

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
</table>
| 2     | 0.000
| 3     | 0.000 -0.398
| 4     | 0.000 -0.192 0.173
| 5     | 0.000 -0.497 -0.089 0.215
| 6     | 0.000 0.237 -0.397 -0.547 0.389
| 7     | 0.000 0.447 0.000 -0.516 -0.667 0.106
| 8     | 0.000 -0.532 0.229 -0.184 0.524 -0.015 -0.429

The above table show that Construct #1 (Perceived ease of use – Appropriateness of demands on users) does not correlate with any of the other seven constructs, because the respondent has given the rating ‘5’ for all the ten elements. These non-discriminatory ratings indicate that HCI5 assumes both poles of this construct are relevant to the ten elements.

**Analysis of Combined Datasets**

The 19 datasets have been pooled together to form a “supergrid”, which has then been subjected to the similar analysis procedure described above. Here below we highlight the more interesting results.

As indicated by the following statistical outputs, the first two factors are more salient, accounting for 32% and 17% of the variance, respectively.

<table>
<thead>
<tr>
<th></th>
<th>Percent of Trace</th>
<th>Cum. Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>76.007</td>
<td>32.158</td>
</tr>
<tr>
<td>2</td>
<td>40.755</td>
<td>49.400</td>
</tr>
<tr>
<td>3</td>
<td>23.810</td>
<td>59.474</td>
</tr>
<tr>
<td>4</td>
<td>23.067</td>
<td>69.233</td>
</tr>
<tr>
<td>5</td>
<td>19.661</td>
<td>77.551</td>
</tr>
<tr>
<td>6</td>
<td>16.784</td>
<td>84.653</td>
</tr>
<tr>
<td>7</td>
<td>13.270</td>
<td>90.267</td>
</tr>
<tr>
<td>8</td>
<td>11.978</td>
<td>95.335</td>
</tr>
<tr>
<td>9</td>
<td>11.026</td>
<td>100.000</td>
</tr>
</tbody>
</table>

The table and the corresponding diagram below show the quality attributes - “Efficiency”, “Responsiveness” and “Adaptability” - are construed in a similar way by the respondents, whereas the interpretations of “Complexity”, “Challenge” and “Trustworthiness” seem relatively divergent. This pattern is similar to that demonstrated by the Designers.
Figure 3.5: Multidimensional analysis on the ALL respondents’ construing of the ten quality-attributes

Furthermore, the graph below seems to suggest that there seem no major differences between the two groups of respondents with respect to the two empirically derived dimensions. Two ‘pairs’ of the Designer-HCI professional, viz. Des5: HCI-c and Des2: HCI-9, can be identified. Some Designers (Des4, Des1, and Des6) form a subgroup. So do HCI-4, HCI-1 and HCI-5. The others, however, are scattered along the two dimensions.
In summary, the statistical analysis results presented above suggest that the Designers interviewed construe the ten given quality attributes somewhat differently from their HCI professionals counterparts. It may be attributed to their background training and to the context of their practice and research. More empirical work is called for to identify such underlying causes.

**Low Level Analysis**

**Background**

The following sections describe useful insights emanating from a more detailed low-level analysis. The data for this emanated from analysis of individual constructs, the expanded descriptions of generated constructs and the fourth stage of follow-up interviews. What follows are descriptions of emergent themes and work towards the meta-categorisation of attributes.

**Measurability and Desirability**

Some controversy emerged over the value of measurability in relation to system attributes. The figures below show a sharp contrast in the reaction of the subjects.
This example is from a designer with vast experience in working on interactive end-user systems such as interactive art, education, and therapy. The subject identifies a negative correlation between measurability and usefulness to designers. A further example suggests a trade off between the rigorous specification of requirements and satisfactory exploration of the design space.

A further construct analysis again suggests scepticism about the role of attributes that define functional or ‘machine-like’ qualities of systems. In the example cited by the subject the attribute of efficiency is assigned to the design of an artefact without accounting for the human or environmental factors associated with its use. This is contrast with the more human-like attribute of challenge.
In a contrasting example from the HCI cell, the desirability of targets to measure against is made clear.

Another subject from the HCI cell makes a similar claim, in stark contrast to the assessment made by subjects in the designer cell above. Measurability is cited as a near-essential link to design improvement.
Further Analysis of Measurability

The data suggests a significant contrast between at least some of the HCI analysts and designers, in the value of measurable attribute values. The examples from HCI suggest a positive correlation between measurability and usefulness in design iteration.

The measurability issue also seems to give an insight into the relationship between user experience and usability quality factors. It seems that most of what are identified or identifiable as user experience attributes are actually outcomes of some process. It may seem counterintuitive that something that takes place during an interactive session, perhaps the experience of joy, challenge or excitement can be referred to as an outcome when it takes place during interaction. However, it can be seen as the outcome of an atomic interactive encounter. This is witnessed by examples in which those attributes cited as experience attributes are referred to as outcomes by some, but in other examples as intrinsic in system use.

Definitional Variation

Whilst many of the attributes in common usage have standard definitions in the literature, there is nonetheless a space of possible interpretation that is conditioned both by semantic and domain-specific factors. It may be impossible to ultimately have stable agreed definitions but useful to consider semantic issues of ambiguity, possible definitional vagueness and variability of interpretation particularly across domains.

Attributes have typically been used to set quality targets for system usage. Traditionally this sets usability targets but increasingly user-experience targets. These attributes are usually difficult to associate with precise measures, and this can vary dependent on the nature of the project. The example below describes three contrasting interpretations of complexity, as described by subjects. Definition one is the most commonly occurring definition that appears to be supported by the majority of subjects. The second definition was contributed by an HCI subject with experience in design and evaluation of complex and safety critical systems. The third definition was by a designer with experience of games and entertainment systems.
Complexity – Three Definitions

Definition 1:
Complexity is a scale of difficulty in task performance. The desirable end of the scale is lowest complexity where the degree of difficulty in achieving a goal is minimized.

Definitional variants: task complexity, cognitive complexity

Referent: artefact, artefact-in-use

Source of measurement/Assessment:
- Number of input actions per task
- Input actions relative to task-model steps – elicited user model of task

Problem Characteristics: Excessive number of task steps, task steps not yoked to task-model elements (Keiras & Polson, 1985)

System type: All

Definition 2:
Complexity is a two-axis concept. On one axis is the complexity represented at the user interface, in the level of detail that is present and the interactivity. The other axis is the degree of complexity in the underlying domain or process being represented. The desirable state is where the complexity of the interface reflects the complexity of the underlying process. This is cited in reference to process control interfaces.

Referent: artefact, dependent process

Source of Measurement/Assessment:
- Actions relative to task/process model,
- process visibility at interface level

Problem Characteristics: Incomplete or inaccurate representation of process requirements at the functional and interface level

System type: Process control

Definition 3:
Complexity is an incrementally increasingly level of challenge aimed at engaging and entertaining the user.

Referent: artefact, artefact-in-use, experience of use

Source of measurement/Assessment: Subjective user response, other unspecified

Problem characteristics: User dissatisfaction, over-competent performance, inability to reach competitive level of performance

System type: games

Ongoing Work

Results of the empirical studies presented in this report are parts of the ongoing analysis work, which will be continued till May 2009 and disseminated via the academic literature.
References


Appendix A: Initial Euclidian Distance Model

Milestone one:
A hierarchical cluster analysis based on 12 data sets. February 2007.

Euclidian Distance Model
|   | 2   | 2  | Slow n | Fast p |
|   | 2   | 2  | Difficult n | Easy p |
|   | 2   | 2  | non-measurable n | Measurable p |
|   | 3   | 3  | Subjective p | Objective n |
|   | 2   | 3  | Inflexible n | Flexible p |
|   | 2   | 3  | Qualitative n | Useable p |
|   | 2   | 3  | Negative n | Positive p |
|   | 4   | 4  | Secure p | Effective p |
|   | 2   | 4  | Difficult n | Suitable p |
|   | 2   | 4  | Slow n | Effective p |
|   | 2   | 4  | Effective n | Flexible p |
|   | 2   | 4  | Slowness n | Flexible p |
|   | 2   | 4  | Thorough n | Manageable p |

**Dimension 2:**

<table>
<thead>
<tr>
<th></th>
<th>safety</th>
<th>adaptability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>well-defined design targets</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>intrinsic system properties</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>something you can evaluate against</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>Possible usability targets</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>About certainty</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>Responsiveness to environment</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>General p</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>Universal view p</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>Secure p</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>constrained capabilities</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>Outcome</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>About outcome</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>Reliable</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>unpredictability n</td>
</tr>
</tbody>
</table>
| 2 | 2  | Unpredictability n | Effective p
"Ability to see how things work", |
| 2 | 2  | Unseen n | Flexibility p                      |
| 2 | 2  | invisible n | Visible p                          |
| 2 | 2  | Non-available n | Availability p                    |
| 2 | 2  | Hidden n | On view p                          |
| 2 | 2  | Abstract n | Concrete p                         |
| 2 | 2  | Open to interpretations n | Measurable p                      |
| 2 | 2  | Subjective n | Visible p                         |
| 2 | 3  | Objective n | Subjective p                       |
| 2 | 4  | Complex n | easy to learn p                    |
| 2 | 4  | Slowness n | Challenging p                      |
| 2 | 0  | Efficiency n | Pleasing p                        |
Review on the Computational and Definitional Approaches in Usability Evaluation
Final Report of COST294-MAUSE Working Group 4

Christian Stary
University of Linz, Austria
Leader, COST294-MAUSE Working Group 4

Claudia Maderthaner
University of Linz, Austria
Co-editor, COST294-MAUSE Working Group 4 Reports

Peter Eberle
University of Linz, Austria
Co-editor, COST294-MAUSE Working Group 4 Reports

Contributors: Other WG4 Members
Abstract

The primary objective of Working Group 4 was to review systematically the existing models and procedures for estimating certain key usability test parameters and traditionally defined usability quality metrics in Human Computer Interaction (HCI).

In order to understand inherent strengths and weaknesses of automated usability evaluation methods (A-UEMs) the members of Working Group 4 have developed a methodology to review and analyze those methods. A systematic and structured review of well documented methods forms a reliable basis for profound cross checking and developing selection mechanism when it comes to the applicability of methods.

The work of WG 4 started with a preparation phase leading to the specification of selection criteria and the selection of A-UEMs to be analyzed. In the second phase the selected methods have been documented using a structured representation schema, processing original documents. In the course of fine-grained analyses the findings could be consolidated, and structured for further research increasing the methods’ maturity.

To ensure the quality of the review six strategic principles have been outlined during the preparation phase. They have guided the review process. Based on these principles a guide for performing documentation and analysis as well as a review template could be developed.

In the explication phase eight selected A-UEMs have been documented by invited experts. Each review has been conducted according to the review template, in order to ensure that the results can be cross-checked.

In the course of consolidation the collected data have been structured using so-called “usability cards”. In this process four different clusters of A-UEMs could be identified in respect to the targeted usability principles.

The cluster analysis of the results of the review showed that some usability principles are addressed by more than one A-UEM. A closer examination of the methods and implemented algorithms has revealed significant differences in the way the principles are utilized for evaluation.

From the results we can conclude that there is still a considerable potential for improvement with respect to the maturity of automated usability evaluation methods. Concepts related to usability evaluation stemming from Working Group 2 and 3 could be used to improve the applicability of Working Group 4 results, as additional aspects of usability engineering could be addressed. Overall, automated usability evaluation methods could be integrated in a more effective in interactive product development lifecycles.
Table of Contents

Introduction........................................................................................................................................ 122
Objectives .......................................................................................................................................... 122
Methodology ...................................................................................................................................... 122
  Procedure ........................................................................................................................................ 122
    Identification of Targets.................................................................................................................. 123
    Explication .................................................................................................................................... 123
    Consolidation ................................................................................................................................. 123
Strategic Guide................................................................................................................................... 123
  Principles......................................................................................................................................... 123
  Relevance of Strategic Principles for the Proposed WG4-Work...................................................... 125
Template .............................................................................................................................................. 127
Results ................................................................................................................................................. 129
Reviews............................................................................................................................................... 129
  The Bloodhound Project.................................................................................................................. 130
  IRC Framework for Notification Systems........................................................................................ 134
  DESTINE....................................................................................................................................... 139
  Cognitive Walkthrough for the Web................................................................................................. 144
  Keystroke Level Model .................................................................................................................... 149
  ProcessLens....................................................................................................................................... 153
  WebRemUSINE................................................................................................................................. 161
  WebTango......................................................................................................................................... 168
Consolidation and Interpretation......................................................................................................... 172
  Usability Cards................................................................................................................................. 172
  Clustering ......................................................................................................................................... 172
  Cross Checks.................................................................................................................................... 175
Integration of Results of other Working Groups................................................................................ 178
  Integrating Working Group 1 Results.............................................................................................. 178
  Integrating Working Group 2 Result................................................................................................ 180
  Integrating Working Group 3 Results.............................................................................................. 180
Conclusion .......................................................................................................................................... 180
Appendix ............................................................................................................................................. 182
  Additional Reviews............................................................................................................................ 182
    AutoCardSorter - Automated Card Sorting Tool ........................................................................... 182
Introduction

Working Group 4 (WG4) is concerned with a systematic review on the computational and definitional approaches in usability evaluation. Our primary target is a systematic review of existing models and procedures for estimating or calculating certain key usability test parameters and traditionally defined usability quality metrics. Of primary interest are the capabilities of formal representations and corresponding computational schemes.

Computational approaches in usability evaluation target towards the automated evaluation of user interfaces, user characteristics, scenarios of use, or specifications. Since the proposed variety of techniques show different backgrounds and various conceptual and operational assumptions, it is difficult to find out their specific capabilities which determine their quality.

Objectives

The main objective of MAUSE is threefold:

- to deepen understanding about the inherent strengths and weaknesses of individual Usability Evaluation Methods (UEMs),
- to identify reliable and valid methods for comparing different UEMs in terms of their effectiveness, efficiency and scope of applicability, and
- to develop efficacious strategies for extracting useful information from the results of UEMs in order to improve the systems tested.

The primary objective of Working Group 4 is to review systematically the existing models and procedures for estimating certain key usability test parameters and traditionally defined usability quality metrics in Human Computer Interaction (HCI).

Based on the insights gained from the comprehensive review, we should find out how to compute accurately the problem-discovery rate (p), inter-/intra-evaluator concordance and appropriate sample size of test participants.

Furthermore, approaches to evaluating the three classic usability metrics will be examined. Specifically, “effectiveness” and “efficiency” as the function of unassisted task completion rates and task completion times, respectively, and “satisfaction” as the function of users’ subjective and often retrospective ratings may not be enough to capture the multifaceted features of usability. Hence, these definitions could be ameliorated and augmented to accommodate the increasing complexity of user interface designs. In WG4, the relationships between internal, external, and in-use metrics are studied that pertain to usability, or similarly quality characteristics that may affect usability, with the use of sophisticated statistical models or other computational techniques. Furthermore, the definitional issues pertinent to usability problems are investigated.

Methodology

In order to collect the information necessary for further analysis Working Group 4 has developed a systematic review procedure. A comprehensive review template fosters the collection of data which could later be compared in respect to the aspects relevant and important for stakeholders in a usability evaluation process.

Procedure

The action plan for WG 4 has led to a three-step procedure. It started with a preparation phase focussing on the selection of A-UEMs to be analysed. We proceeded with an explication phase, detailing the information about the selected A-UEMs, and continued with a consolidation phase integrating the findings for further processing.
Identification of Targets

The first phase of the work of WG4 consisted of three main tasks:

- the compilation of a strategic guide for the reviewers in order to guide them in the review process
- the development of a review template for the evaluation of the UEMs
- the selection of UEMs meeting the criteria for Computational Relevance

Results

- strategic guide
- review template (including operation guide)
- list of selected A-UEMs

Explication

The UEM selected in the previous phase were analyzed and then structured according to the review template. The information gathered in this analysis was the base for an interpretation of the content.

Results

- completed review templates (one per selected UEM)

Consolidation

Finally, the results of the explication phase have been consolidated in terms of encountered usability problems as well as the quality of the A-UEMs.

Strategic Guide

To facilitate the review process a layered scheme for the structured evaluation of existing automated usability evaluation methods (A-UEMs) has been developed. In contrast to existing schemes it reveals relationships between usability principles and computable measurements. The scheme can be seen as an implementation of some comparative analysis principles which are shortly described in the following.

Principles

The primary target of our work is a systematic review of existing models and procedures for estimating or calculating certain key usability test parameters and traditionally defined usability quality metrics. Of primary interest are the capabilities of formal representations and corresponding computational schemes.
The work has been guided by several principles, in order to facilitate a systematic review. The following list contains the strategic guidelines ordered by their anticipated priority:

(1) Computational UEM Relevance
We have to agree on the computational aspects of UEMs. A meta-model of computing usability parameters facilitates cross-checks. It has to contain indicators and prospective indices for human behaviour and system performance. It might contain calculi structures and calculi-specific constraints, e.g., for FOL.

(2) Definitional Primacy
Each usability parameter requires an explanatory definition for consistent reviewing and sharing knowledge about UEMs. Such definitions are also required to develop a commonly accepted concept of usability-problem (UP) specification.

(3) Fine-Grain UEM Specification
A structured review requires a high granularity when describing UEM. We do not only consider a computational or/and a definitional commitment of an UEM to be crucial for our review, but also a comprehensive list of UEM attributes, such as authors, target of evaluation, addressed usability parameters, derived metrics, application procedure. Since such a catalogue of attributes is of relevance for all WGs in MAUSE, its development should be a mutually tuned approach.

(4) Base(line) Specificity Exploitation
The specificity of an UEM denotes the (required) universe of discourse for evaluation. What do users and evaluators need to know when applying an UEM? It may address a certain stage of development, such as design or a particular dimension, such as the organization of work or the mapping of business objects to interactive data forms. It may also address a certain level of user qualification, such as the capability to handle a virtual desktop. In this case, the UEM will focus on higher-level interaction tasks. The review needs to take into account those items, since we have to expect a diverse variety of UEMs, including different granularity in terms of parameters, models or procedures. Guiding the review by explicating the base(line) facilitates analyses and cross-checks.

(5) Explication of Semantic Encoding(s)
Since usability parameters, such as task conformance, convey a certain meaning, it is of crucial interest for the review (i) how this meaning is specified for the different UEMs, and (ii) how it is propagated to measurements, models, and procedures. Of particular interest is the mapping of a definitional item to a calculus or computational scheme. Last, but not least, the semantic relationship(s) to the traditional high level usability parameters efficiency, effectiveness, and satisfaction has (have) to be made transparent.

(6) UEM Accuracy Determination
Since the result of our studies is expected to be a sound estimate on the quality of UEMs, their accuracy plays a major role in the review. Do UEMs use adequate means for capturing interactive software quality and measurement? Or do they generate overhead due to computational requirements or definitional constraints? The accuracy of an UEM indicates in how far the intended objectives can be met through its application and usage.

Some of the introduced strategic principles are complementary. For instance, the definitional primacy (principle 2) ensures transparency for semantic encoding (principle 5). Another example is the Base(line) Specificity Explication (principle 4) that requires implementing the Fine-Grain Specification principle (3). However, the differentiation into dedicated (sub)principles facilitates focussing on crucial review issues which should drive our research.
Relevance of Strategic Principles for the Proposed WG4-Work

Examination of Classic Usability Metrics

The Explication of Semantic Encoding(s) principle (5) helps to reveal whether “effectiveness” and “efficiency” can be seen as a function of unassisted task completion rates and task completion times, respectively. It will also help to reveal whether “satisfaction” as the function of users’ subjective and often retrospective ratings are sufficient to capture the multifaceted features of usability. In this way, the three conventionally defined usability metrics – effectiveness, efficiency, and satisfaction – are put on trial, in particular when evaluating highly complex user interfaces.

The Definitional Primacy principle (2) should primarily drive the study of definitional issues pertinent to usability problems, and help to ameliorate and augment existing usability-parameter definitions, in order to accommodate the increasing complexity of user interface designs. Led by both principles, the relationships between internal, external, and in-use metrics that pertain to usability, or similarly quality characteristics that may affect usability, can be explored. In case sophisticated statistical models or other computational techniques are used, the Computational UEM Relevance principle (1) and the UEM Accuracy Determination principle (6) direct the review.

The Fine-Grain UEM Specification principle (3) helps to overcome the observed lack of reliable and valid means to compare UEMs in terms of their relative effectiveness, efficiency and scope. The scope of different UEMs in terms of who, when, how, and where they are used, is addressed by the Base(line) Specificity Exploitation principle (4). The scope as well as the underlying framework, model(s), and theories with respect to usability have to become part of the review catalogue. The latter principle should also help to overcome the observed lack of systematic review on the suitability of the three traditionally defined usability metrics – effectiveness, efficiency and satisfaction – for revealing the extent to which a system is usable or useful, especially when the complexity of the system evaluated is high.

Finally, the Explication of Semantic Encoding(s) principle (5) should foster empirical research to investigate the relationships among a cluster of quality attributes, all of which should ideally be incorporated into a system design. Specifically, the orthogonality issue should be resolved in that context when connected parameters in a way that can be represented with a computational model.

Capturing Empirical and Analytical UEMs

The review principles allow both types of UEMs - empirical and analytical – to be investigated. The Base(line) Exploitation principle (4) and the Fine-Grain UEM Specification principle (3) should set the stage for empirical UEMs involving the participation of users, test administrators, observers as well as evaluators, and the employment of a set of scenario-based tasks and thinking aloud techniques. Since analytical UEMs have no direct access to users, these UEMs have to meet the Computational UEM Relevance principle (1) or the Definitional Primacy principle (2) employing a set of guidelines and/or heuristics for streamlining and structuring the inspection process. The principles allow capturing tests conducted on prototypes or operational products but also on other artefacts of the development process such as models, performed manually or automatically.

Analysis of Usability Problems

The Definitional Primacy principle (2) facilitates the extraction of Usability Problems (UPs) from a system or its usage. Since UPs should be analysed systematically so as to derive information about their possible causes, applying the Base(line) Specificity Exploitation principle (4) is required to capture context-sensitive parameters for analysis. In addition, the UEM Accuracy Determination principle should facilitate the development of a commonly acceptable definition of a UP clarifying the appropriate granularity level at which a usability problem should be registered. The Fine-grain UEM Specification principle (3) together with principle 2 and 4 should direct the review to examine the specification of intended values of a product or service, and to come up with threshold values for usability problems.
Integration of UEMs into Systems Development Lifecycles

As mentioned in the proposal, one significant idea emphatically pointed out by usability professionals is that usability should be integrated into the whole systems development lifecycle, from the earliest phase of requirement analysis to the latest phase of post-launch evaluation. Apparently, the later in the lifecycle a problem is detected, the more expensive it is to fix. Hence, the review targets the issue whether usability concepts can be integrated successfully into all phases of the development lifecycle. Applying the Fine-Grain UEM Specification principle (3) as well as the Base(line) Specificity Exploitation principle (4) details upon the collaboration between usability specialists and software engineers can be analysed. However, in the context of WG4 the analysis has to be performed under the auspices of the Computational Relevance and the Definitional Primacy principle (1 and 2). They have to lead the investigations of usability attributes or compositions of those comprising basic components, such as utility (or effectiveness), learnability, efficiency, retainability (or memorability), errors, and satisfaction. Finally, the UEM Accuracy Determination principle (6) directs the review to cross-check the claims of UEMs against their actual achievements. The Semantic Encoding(s) Explication principle (6) helps to find out the reasons for mismatches between UEM claims and achievements.

Automation of UEMs

Automating usability evaluation and tool development for guideline implementation is a crucial review issue (applying the Computational UEM Relevance principle (1)), targeting guideline-definition languages and user-interface generators. Since there is still a number of problems to be resolved, caused by ineffective algorithms or representation schemes, several principles have to be taken into account: The Fine-Grain UEM Specification principle (3) as well as the Base(line) Specificity Exploitation principle (4) have to be applied to achieve the required level of detail and comprehensiveness for review. Once we have achieved the accurate level of detail and granularity the Definitional Primacy principle (2) for UP specification has to be implemented, since the accurate representation of UP seems to be crucial for improving the usability of a system. Finally, the UEM Accuracy Determination principle (6) directs the review to cross-check the possibilities of UEM automation against their intentions. The Semantic Encoding(s) Explication principle (5) again helps to find out the reasons for possible mismatches between claims and achievements.

Applicability of UEM for Every Possible Action

As a lack of approaches that enable usability professionals to design experiments by modelling consequences of every possible action has been observed, the development of such approaches need to be explored and critically reviewed. For analysing whether the two major requirements are met - a probabilistic model for the testing process and the outcomes when the software system tested, and the utility of the possible outcomes – the Computational UEM Relevance principle (1) is useful. Depending on the semantic encoding(s) (principle 5) and the base specificity (principle 4) the quality of approaches combining probabilities and utilities into a coherent decision-making process with the use of a Bayesian decision-theoretic approach can be checked for accuracy (principle 6 – UEM Accuracy Determination).

True Benchmarking

Due to the lack of a global approach to defining quality models for the Web, gathering those data is led by the Fine-Grain UEM Specification principle (3). Such guidance leads to UEM foundations through explicit underlying quality models as well as to the comparability of quality models for Web applications, even when they are expressed in various formalisms (e.g., algebraic notation, first-order predicate logic, graph theory, graph grammars) that initially prevent true benchmarking. Applying the Computational UEM Relevance principle (1) leads to a meta-model for computation that allows us to come up with a benchmarking of Web-quality models enabling evaluators to assess the usability of a Web application according to the used UEMs.

Once we are able to improve models and algorithms, more accurate estimations of key usability test parameters can be expected. Practitioners might re-arrange usability evaluation accordingly, based on
revised usability metrics, operational criteria for usability problems, and in line with other quality attributes upheld by other professionals, such as quality managers.

Template

The strategic guide described above was the base for the creation of a review template. The review template ensures that the reviews of the different selected UEMs are comparable and therefore suitable for further consolidation and interpretation.

Together with the template an operation guide has been created. This document works as a guideline for reviewers in order to support them during the reviewing process and the documentation of the results in the review template.

The preparation of the template was an iterative process. Every draft of the template was tested and the experience gathered during the tests was incorporated in the next version of the template as well as of the operation guide. During the process four version of the template have been tested.

The following pages show the last version of the review template which was (together with the operation guide) the base for the review of the selected UEMs.

**IMPORTANT:**
The structured representation should be based on raw data that can be found directly in the UEM text and material. Interpretations and citations have to be marked as such, e.g., using dedicated [own text] or >interpretation< brackets, otherwise the results are hard to trace.

(Principle #) refers to Principle (1)-(4) mentioned in the operation guide.

Principle (5) and (6) can only be applied after all UEMs have been analyzed, since they are used for cross-checking the findings. Applying the Explication of Semantic Encoding (principle 5) should help to reveal different meanings for identical or similar terms denoting UEM-parameters or -principles. Applying UEM Accuracy Determination (principle 6) should help to develop an understanding how well targeted UEMs have been developed, and consequently, can be applied. It should allow a final judgment on the quality of UEMs.

In the following all categories for the structured description are underlined, heading the actual data. If applicable, a set of references to the principles (1) – (6) follows to demonstrate the applicability of the acquired information according to this category. In case specific items of a principle are addressed, they are referred through the corresponding number in (), such as 1(1.), referring to the first enumerated item of principle 1 (Computational UEM Relevance).

**1 Method Identification**

**1.1 UEM Name:**
> Full bibliographical reference and availability:

**1.2 Abstract**
> Abstract of the reference paper, short method description

**1.3 UEM- History**
> History of the method, version described, main evaluations, etc.

**2 Method Characteristics**

**2.1 Objectives**
> Main goals and motivation, expected output/results. What are the questions that the method addresses and why?
2.2 Usability Principle(s) Addressed by the UEM

(Principle 1(1.), 2, 3)

**UE-Principle 1**
- Name:
- Explanatory definition (incl. ref. to standard):
- Dimension/target/focus (organisation of work/technology/cognitive system):
- Addressed (list of) parameters:
- Relationships to other Principles (give both, type of relationship and principle):

[...]

**UE-Principle n**
- Name:
- Explanatory definition:
- Dimension / target / focus:
- Addressed list of parameter list:
- Relationships to other Principles:

2.3 UE-Parameters:

(Principle 1(1.), 2, 3)

**UE-Parameter 1**
- Explanatory Definition:
- Relationship to other Parameters (give both, type of relationship & parameter):
- Formal Encoding (incl. type):
- Elicitation/Acquisition Method when measured:
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme):
- Involvement in Computation (input / output to, side effect of calculation):

[...]

**UE-Parameter n**
- Explanatory Definition:
- Relationship to other Parameters:
- Formal Encoding (incl. scale, value set):
- Elicitation/Acquisition Method:
- Mapping to Computation Scheme:
- Involvement in Computation:

2.4 UE-Principle/Parameter Relationship Matrix

(Principle 3)

<table>
<thead>
<tr>
<th></th>
<th>UE-principle 1</th>
<th>UE-principle n</th>
</tr>
</thead>
<tbody>
<tr>
<td>UE-Parameter 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UE-Parameter 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UE-Parameter n</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.5 Computation Scheme

(Principle 1(2.,3.))
- Calculus:
- Representational Elements (facts, rules or data structures):
- Algorithm / Computation:
- Result(s) incl. Rules for Interpretation:

2.6 Best Practice

(Principle 4)
Results

In this chapter we detail the findings of the WG4 activities. We give the results of the conducted reviews as well as the outcome of the consolidation and interpretation of the review data.

Reviews

We have analysed eight CUEMs using our review template:

1. The **Bloudhound Project**: Chi et al. (2003)
An additional review was conducted after the consolidation and interpretation phase. Consequently, it could not be included due to time reasons. The respective results can be found in the appendix of this document.

The Bloodhound Project

Reviewer: Chris Stary

Method Identification

**UEM Name:**
Bloodhound Project

**Reference(s) incl. authors and availability:**

**Usability Principle(s) Addressed by the UEM**
There is no explicit reference to any UE-principle in the text.

**UE-Principle 1**
- **Name:** [controllability]
- **Explanatory definition (incl. ref. to standard):** [not given, since interpreted]
- **Dimension / target / focus:** [technology]
- **Addressed (list of) parameters:** navigability
- **Relationships to other Principles (give both, type of relationship and principle):** [not given, since interpreted]

**UE-Parameters:**

**UE-Parameter 1**
Navigability of a web site
- **Explanatory Definition:** As users discover information items their needs have been satisfied. Target documents are assumed to satisfy user needs. „The rate in which people finish is a measurement of the navigability of a site“. (adopted from the ISAR part, p. 508)
- **Relationship to other Parameters (give both):** [none]
- **Formal Encoding (incl. scale, value set):** probability of success in %
- **Elicitation/Acquisition Method:** information needs (and links)
- **Mapping to Computation Scheme:**
  - information needs into vector
  - links
  - proximal cues
- **Involvement in Computation:** representation is not changed, input to calculation

**UE-Principle/Parameter Relationship Matrix**

<table>
<thead>
<tr>
<th>UE-Parameter</th>
<th>UE-Principle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigability</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>[controllability]</td>
</tr>
</tbody>
</table>

**Computation Scheme**

- **Calculus:** [not named]
- **Representational Elements (facts, rules o.t.l.):** matrix, vector
- **Algorithm / Computation:**

**INFOSCENT-SIMULATION**
[Background that is required for understanding the algorithm:]

- The simulation is based on the theoretical notion of information scent developed in the context of information foraging theory (how users optimize behaviour to seek information both in directed structured and opportunistic unstructured ways)
- [Authors have found out] users commonly have some information goal – some specific information they are seeking – when they visit a Web site.
- Users typically forage for information by navigating from page to page along hyperlinks.
- The content of pages associated with these links is presented to the user by some snippets of text or graphic called proximal cues.
- Foragers use these browsing proximal cues to access the distal content: the page at the other end of the link.
- Information Scent is the imperfect, subjective perception of the value, cost, or access path of information sources obtained from proximal cues.
- During information seeking, when choosing from a set of outgoing links on a page, the user examines some of the links and compares the cue (i.e., link anchor and/or surrounding text) with her information goal.
- The user takes the degree of similarity as an approximation to how much the content reachable via that link coincides with the information goal.
- Using this concept, Web User Flow by Information Scent (WUFIS) is a predictive simulation technique based on a combination of information retrieval techniques and spreading activation.
- The prediction model employs a simulation of an arbitrary number of users traversing the links and content of a Web site.

**General description**

- The users have information goals that are represented by vectors of content words. At each page visit, the model assesses the information scent associated with each hyperlink.
- The scent of a link is calculated as a degree of similarity between the proximal cues and the information need. It then computes a probabilistic network of the likelihood of one user moving from one page to another page along hyperlinks that may or may not match the user’s information goal.
- This probabilistic network is then used to simulate the user flow throughout the site based on that information goal.

**Simulation algorithm**

1. Extraction and Representation
   - content and linkages of a Web site are extracted
   - hyperlink topology is represented as an adjacency matrix $T$
   - content (word x document) is obtained as $W$ matrix – an entry in the $W$ matrix specifies how important a word is in that document according to TF.IDF, an information retrieval algorithm
   - A user’s information need is expressed as a keyword query vector $Q$

2. For each link $E(i,j)$, the proximal cue words are obtained that are associated with that link, and insert this information into a matrix $K$ - a three dimensional matrix, with an entry $K(i,j,k)$ specifying that link $E(i,j)$ contains the keyword $k$

3. Proximal cues are obtained by looking up the weighting of each keyword in $K$ in the matrix $W$ to measure the importance of each keyword

4. The link cues are multiplied in $K$ with $Q$ to obtain the Proximal Scent matrix $PS$. 
Thus, for each link $E(i,j)$, the corresponding proximal cue words from $K$ are found, obtaining a vector $K(i,j,*)$. $PS(i,j) = K(i,j,*)^TQ$. $PS$ is then normalized so that each column sums to 1.0. This Proximal Scent matrix specifies the probabilities of users following each particular link.

5. At this point, the scent matrix is used to simulate users flowing through various links of a site, giving each link a different proportion of the users relative to the strength of the scent. The probability associated with each link essentially specifies the proportion of users that will flow down various link choices. A network flow algorithm called spreading activation is used. For an entry page an entry vector $E$ is constructed. The initial activation vector $A(1) = E$. The algorithm goes through $t=1...n$ number of clicks: $A(t) = (\alpha)S A(t-1) + E$. The parameter $\alpha$ simulates the proportion of users that do not go from step $t$ to step $t$ (e.g. Law of Surfing estimation could be used here). This process generates a predicted user flow, which can be used to extract simulated user paths and infer the usability of a web site.

**Note by the authors:**

'By itself, the simulation algorithm reported earlier is not enough to measure this navigability. We need a method to directly measure how easily can users reach the targeted destinations using the information goal given. To do this, we developed the following new method called Information Scent Absorption Rate (ISAR).’ (p.508)

**INFORMATION SCENT ABSORPTION RATE METHOD**

[Background that is required for understanding the algorithm:]

'The intuition is that as users discover information items their needs have been satisfied and the simulated users should settle and terminate at a set of documents. These target documents are the documents that satisfy their information needs. The rate in which people finish is a measurement of the navigability of a site.’ (p.508)

**Algorithm**

1. The Scent matrix is computed as specified above. 'Each entry $S(i,j)$ in the Scent matrix is the calculated probability that a user will surf from page $i$ to page $j$, given that this user has the given information goal.

2. However, the scent matrix that describes the surfing graph has leaves (nodes without connections to further nodes). The spreading activation algorithm does no backtracking, as simulations only move forward on the network. One way to fix this is to tie leaf nodes back to the starting point.’ (p. 508) If node $j$ is a leaf, $S(j, \text{starting page}) = 1.0$. Any user reaching node $j$ would start over at the initial page.

3. The actual destination pages are made the absorption states. Users reaching these absorption states do not leave these documents. To do this, the destinations are turned into nodes that do not have any children (that is, turn them into leaf nodes of the graph.) The Scent matrix $S$ is taken and the entire column of the target documents is zeroed out. If target document is $t$, then the $t$-th column of the $S$ matrix should be zeroed out (this new scent matrix is termed $S'$).

4. The spreading activation user flow simulation is done using this updated $S'$ scent matrix and the amount of activation still left in the activation vector at the last click of the simulation is summed up (this value is beta - it can be thought of as
the probability that someone would still be searching for the destination page). Then the probability of success is (1-beta).

- Result(s) incl. Rules for Interpretation:
  - A probability of success of about 35% is considered to be a ‘fair’ ranking (cf. p. 508)
  - Differences between success rates indicate usability problems (cf. p.509)

**Best Practice**
- Setting (users, location etc.):
  - [web site has to be available]
  - [analyst has to be available]
  - [information goals and/or user tasks have to be specified]
- Procedure:
  'The application scenario is that a customer of the service would specify the site to be analyzed, and the information goal to be simulated in the analysis. Then the Bloodhound service would return usability metrics that tell the customer how easy it is to accomplish the information goal that was given.’ (p. 509)
  1. Input for analysts to specify the site to be analyzed and a set of user tasks (specified using information keywords related to the goal) and the associated destinations to retrieve the correct information.
  2. The results are calculated showing average success rate of the tasks.
  3. Display of several high traffic pages are used as intermediate navigational pages, including pages that may be bottleneck pages.
- Documentation: automated usability reporting
- Particularities: [not reported]

**Encountered Usability Problems when Applying UEM**
- Problem Description: There are navigational hindrances.
- Reasons: [are not provided, have to be elicited from elsewhere or are indicated through direct comparisons]
- Suggestions for Improvement: [not provided]
IRC Framework for Notification Systems

Reviewer: Ebba Thora Hvannberg

Method Identification

**UEM Name**
IRC Framework for Notification Systems

**UEM Name**

**Abstract**
Critical parameters are used to characterise user goals and usability artifacts. The proposal is specifically targeted to address interface design for notification systems, but its themes can be generalized to any constrained and well-defined genre of interactive system design. A slight variation on the concept of critical parameters is presented with abstract and concrete knowledge representations. The critical parameters are Interruption, Reaction and Comprehension (IRC). With this concept, the authors demonstrate a feasible approach by introducing equations that elaborate and allow evolution of notification system critical parameters, which is made operational with a variety of usability evaluation instruments.

**UEM- History**
Newman (1997) introduced the concept of critical parameters. Scott Mccrickard has published papers on critical parameters for notification systems since 2003 but the computational aspects are new in this paper.

**UEM- Type**
Can either be analytical or empirical. [We emphasize analytical in this report]

**Method Characteristics**

**Objectives**
...[critical parameters] would help interface designers recognize the broader intentions of the technology, shifting focus away from interfacespecific details to qualities that could be directly measured, compared to benchmarks, and reengineered to better serve a user’s purpose. Critical parameters have three essential characteristics: their satisfaction is critical to the success of the system, they are persistent across successive systems, and must be manipulable by designers [21]. Critical parameters support the organization of systems by scenario families, collections of systems and the context of their use grouped by critical parameter value. Expressing new design problems in terms of critical parameter values allows efficient association with theories and guidelines from psychology, sociology, and human factors—information that is otherwise difficult to obtain. Designers are, in effect, using critical parameters as an index into a vast store of knowledge.

Assessment of critical parameter values through mediated evaluation can allow systems to be compared in formative phases with other systems, benchmarks, and standards. The value of the IRC computed variables make inferences about usability issues and possible directions of redesigns. While the IRC parameters were useful in assessing each design individually, the broader benefits of using critical parameters are recognized in activities such as system comparison and design knowledge reuse.

Values of critical parameters collected in either empirical or analytical usability evaluations are used to suggest redesigns.
Usability Principle(s) Addressed by the UEM
[None given, but interpreted. I need to see further explanations of the UE-Principles to be able to make the final interpretations.]

UE-Principle 1
- Name: [Controllability] <is it better to skip the interpretation?? >
- Explanatory definition (incl. ref. to standard):
- Dimension / target / focus (organisation of work / technology / cognitive system) : Cognitive system
- Addressed (list of) parameters: Interruption
- Relationships to other Principles (give both, type of relationship and principle):

UE-Principle 2
- Name: [Suitability for the task ]
- Explanatory definition: :
- Dimension / target / focus:
- Addressed list of parameter list: Responsiveness
- Relationships to other Principles:

UE-Principle 3
- Name: [Explainability]]
- Explanatory definition: It is not mentioned explicitly, but interpreted.
- Dimension / target / focus:
- Addressed list of parameter list: Comprehension
- Relationships to other Principles:

UE-Parameters

UE parameter 1: Input parameters to I
- $S$ = sustainment
- Actual interruptive effect can be gauged by primary task sustainment – a metric used to quantify the change in the primary task performance from solo-task to dual-tasks performance.
- Compared to the primary task performance before the notification delivery, how much does the primary task performance reduce when the notification is present
- Floating number from 0=completely stops to 1=not at all

UE parameter 2: Input parameters to I
- $COI$ = cost of interruption
- Appropriateness of an interruption is represented by COI, characterising the user’s willingness to accept an interruption, and thus the urgency of the primary task can be inferred.
- Given the nature and importance of the user’s primary task at the receipt of the notification, how costly would an interruption be?
- Floating number from 0=not at all to 1=extremely

UE-Parameter 3: I (Interruption)
- Explanatory Definition:”I” describes both the appropriateness of an interruption, as well as the actual interruptive effect of the notification artifact (distraction to the primary task).
- Relationship to other Parameters (give both, type of relationship & parameter):
- Formal Encoding (incl. type): Floating number less than 1 and greater than equal to 1
- Elicitation/Acquisition Method when measured: Through survey
- Mapping to Computation Scheme (which element of the calculation scheme
corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme):

- \( I = 1 - S \cdot 3 \cdot \text{COI} \)
- Involvement in Computation (input / output to, side effect of calculation) : Output

#### UE parameters 4-5: Input parameters to R (Reaction)
- In addition to COI being an input parameter, there are two parameters \( h \) and \( t \):
- \( h \) = hit rate refers to the concept from signal detection theory where a user correctly detects and responds to a signal (a notification).
- How often will users actually notice important changes in the notification, as opposed to not noticing them
- Always=1, more than half=0.75, about half=0.5, less than half=0.25, never =0.0001
- \( t \) = relative response time is a ratio between actual and expected response times.
- expected response times are obtained in requirements gathering.
- In cases where a notification suggests an action for a user to take, how does the user’s response time compare to the reasonably desired response time?
- Better or as good as expected=1 slightly slower=0.75, about twice as slow as expected=0.5, much slower=0.25 extremely slow or action never taken [missing value ]; no action ever required=0

#### UE-Parameter 6: R (Reaction)
- Explanatory Definition: Describes a user goal that can be generalized as an immediate response to a new notification.
- Relationship to other Parameters:
- Formal Encoding (incl. scale, value set): Between 0 and 1.
- Elicitation/Acquisition Method: Survey
- Mapping to Computation Scheme:

\[
R = \frac{(t - h)^{3 - \text{COI}}}{2} + C \cdot \left(0.5 + \text{COI}\right)
\]

- Involvement in Computation: Output

#### UE-Parameter 7-9: Input parameters to Comprehension
- \( f \) = projection of future status
- Based on the notification content, how successful will the user be in making projections or predictions about future trends or the long-term state of the system being monitored?
- \{extremely successful=1, very successful=0.75, somewhat successful=0.5, not very successful \}
- \( c \) = base comprehension of the current situation
- How much of the notification content will the user want to remember and be able to remember several minutes after the notification is delivered?
- all content=1, more than half=0.75, about half=0.5, less than half=0.25, none at all=0
- \( p \) = perception rate of the elements of the system.
- When considering the total number of times a user interacts with the notification system, what is the ratio of the interactions in response to an important notification vs. total interactions (including those when no actual notification was being delivered, i.e. user checking on their own or thinking
there was a notification.

- 1 to 1 = 1: 2 to 3 = 0.75; 1 to 2 = 0.5; 1 to 4 = 0.25, more than 1 to 4 = 0

**UE-Parameter 10**: C (Comprehension)

- Explanatory Definition: Our abstract parameter of comprehension is based on the concept of situation awareness in which a user accumulates Perception, Comprehension and the Projection.
- Relationship to other Parameters:
- Formal Encoding (incl. scale, value set): Floating number between 0 and 1.
- Elicitation/Acquisition Method: through surveys
- Mapping to Computation Scheme:

\[ C = f + \frac{(1 - f)(p + 2c - cp)}{3} \]

- Involvement in Computation: output

**UE-Principle/Parameter Relationship Matrix**

[not given, since even UE-principles are not given and have only been interpreted]

**Computation Scheme**

- Calculus: 
- Representational Elements (facts, rules or data structures): formulas
- Algorithm / Computation: 3 formulas
- Result(s) incl. Rules for Interpretation: Comparison of results with target values. Claims expressed while giving ratings are used for redesigns.

**Best Practice**

- Setting or Context of Use (users, location etc.): ..., the instrument should be used by interface experts or at least experienced notification systems designers familiar with applicable challenges.
- Procedure (step 1 to n to follow when using of UEM): Testing and analysis procedure:
  1. Collect design model intentions in the form of targeted IRC values from each system’s design team. This was accomplished with a survey tool that has been validated to produce accurate and consistent design model IRC values [8].
  2. Presenting the interface prototypes for analytical evaluation. Ratings were collected and interactive discussion among evaluators about design decisions were carried out.
  3. Data analysis. Calculate the IRC values and compare to targets. [Index claims with IRC values to store reusable knowledge]
- Documentation (how is the process, how are the results documented): While response selections provide feedback in the form of critical parameter values, perhaps of equal or greater value are the specific comments and rationale behind each rating, which can be expressed as claim upsides and downsides. We envision this analytical instrument to be used in a moderated evaluator discussion session that may or may not include the system designer, although each evaluator would provide individual assessment of each question.
- Particularities (is there anything to be considered when using the UEM): [how resource demanding is the UEM?]; Evaluators are grouped into clusters according to the values that they give for the IRCs.

**Encountered Usability Problems when Using UEM**

- Problem Description: [non given]
- Reasons:
- Suggestions for Improvement:
Other encountered Problems when Using UEM

- Problem Description (e.g., reliability): [non given]
- Reasons:
- Suggestions for Improvement:

Summary of Advantages

- > Contributions:
- „„ critical parameter, which would allow benefits related to both abstract and concrete knowledge representations
- Equations and usability evaluation support to elaborate and allow evolution of notification system critical parameters.

Estimated impact

> [The method is very young, 2004, and therefore too early to speculate. Admittedly it is still under development]

Open research questions

- .. by articulating the concrete terms we rely on to form our abstract notion of notification systems comprehension, we open this issue and others for debate within the research community.
- We have intentionally developed our proposal to serve as an open, corrigable record of issues and possibilities, rather than a final solution
- the difficulty still arises in that we may not have a key, manageable set of critical parameters.
- [How applicable is the UEM to other systems than notification systems. How can critical parameters be developed for other application domains?]

Other comments

> Criticism: „„ the notion of setting a user goal and psychological effect like reaction to a linear axis often evokes resistance. However, we suggest that the notion can be embraced as a conceptual metaphor and tool for dialog.

Description of Case Studies

- Name and reference: Three different Prototypes A, B and C of the Scope, a small display that resides in the corner of a user’s desktop, depicting new and existing notifications in quadrants for email, calendar, task and alert items.
- Abstract
- Method: 3 to 6 evaluators were organized into sessions in which one interface was analyzed with the analytic instrument. Sessions were moderated, but evaluators gave ratings.
- Participants: 34 experienced notification systems designers as evaluators
- Setting: Moderated group sessions, but individual ratings.
- Results: A significant difference in support of our hypothesis –the instrument helps evaluators achieve consistency that is meaningful according to system.
DESTINE

Reviewer: Effie Law

Method Identification

**UEM Name:**
DESTINE Evaluation Tool


**Abstract**
A system for automatically evaluating the usability and accessibility (U&A) of websites by checking their HTML code against guidelines has been developed. All U&A guidelines are formally expressed in a XML-compliant specification language called Guideline Definition Language (GDL) so as to separate the evaluation engine from the evaluation logics (the guidelines). This separation enables managing guidelines (i.e. create, retrieve, update and delete) without affecting the code of the evaluation engine. The evaluation engine is coupled to a reporting system that automatically generates one or many evaluation reports in a flexible way: adaptation for screen reading or for a printed report, sorting by page, by object, by guideline, by priority, or by severity of the detected problems.

**UEM- Type & History**
Analytic UEM; Tool for Working with Guidelines – initiated as a SIG in CHI’94

**METHOD CHARACTERISTICS**

**Objectives**
Four major objectives:

- To reduce the resources required for checking websites against U&A guidelines thoroughly and successfully
- To remedy the lack of experts who are able to conduct U&A evaluation techniques and analyse results
- To eliminate biases caused by subjective interpretations of guidelines
- To enhance the impact of evaluation results on the redesign of websites

**Usability Principle(s) Addressed by the UEM**

**UE-Principle 1**

- Name: Effectiveness
- Explanatory definition (incl. ref. to standard): Checking a webpage against U&A guidelines thoroughly and successfully
- Dimension / target / focus (organisation of work / technology / cognitive system): Rendering the resource-demanding checking process manageable and economical; Evaluation results can automatically be generated in a selected format
- Addressed (list of) parameters: Number/source of guidelines; Number/type of websites evaluated
- Relationships to other Principles (give both, type of relationship and principle): Flexibility (see below)
**UE-Principle 2**

- Name: Flexibility
- Explanatory definition (incl. ref. to standard): Different navigations correspond to various evaluation intellectual paths; Different reporting formats (HTML or printed hardcopy)
- Dimension / target / focus (organisation of work / technology / cognitive system): Structured documentation; Ease of search for relevant information
- Addressed (list of) parameters: Usability errors sorted by different criteria and presented in different locations; Levels of details of usability error reported; Delivery mode of usability report
- Relationships to other Principles (give both, type of relationship and principle): Effectiveness, Controllability, Self-descriptiveness

**UE-Principle 3**

- Name: Self-descriptiveness
- Explanatory definition (incl. ref. to standard): Information can be found in an intuitive way
- Dimension / target / focus (organisation of work / technology / cognitive system): Reduce searching time [and lessen user cognitive workload]
- Addressed (list of) parameters:
- Relationships to other Principles (give both, type of relationship and principle): Flexibility (see above)

**UE-Principle 4**

- Name: Controllability
- Explanatory definition (incl. ref. to standard): Users are able to read the evaluation results at a selected focus and level of details
- Dimension / target / focus (organisation of work / technology / cognitive system): Contents of the usability report
- Addressed (list of) parameters: Selected source/number of guidelines to be checked; Usability errors sorted by different criteria and presented in different locations; Levels of details of usability error reported; Delivery mode of usability report
- Relationships to other Principles (give both, type of relationship and principle): Flexibility; Suitability for individualization

**UE-Principle 5**

- Name: Suitability for Individualization
- Explanatory definition (incl. ref. to standard): The visual aspect of the report can be customized by the user, enhancing user-friendliness
- Dimension / target / focus (organisation of work / technology / cognitive system): Personalization of the contents [increase user satisfaction]
- Addressed (list of) parameters: Colours, type of graphics, font
- Relationships to other Principles (give both, type of relationship and principle): Controllability, Flexibility

**UE-Principle 6**

- Name: Suitability for the Task
- Explanatory definition (incl. ref. to standard): The presentation of the evaluation report is compliant with U&A guidelines as the report in itself consists of web pages
- Dimension / target / focus (organisation of work / technology / cognitive system): Compliance with standards; platform-independence; adaptation to
users’ specific needs
– Addressed (list of) parameters: Readability of HTML documents across platforms;
– Relationships to other Principles (give both, type of relationship and principle):
  Effectiveness; Flexibility

**UE-Parameters**

**UE-Input Parameter 1 – Formalized U&A Guidelines**
– Explanatory Definition: Respect or Violation of U&A Guidelines
– Relationship to other Parameters (give both, type of relationship & parameter):
  HTML elements of a web page (which and where – scope)
– Formal Encoding (incl. type): Guidelines Definition Language (GDL): formalizing the natural language U&A Guidelines into a GDL-compliant form and store in an XML file for parsing
– Elicitation/Acquisition Method when measured: Collecting U&A data with their corresponding metrics; Interpretation or re-expression of guidelines
– Mapping to Computation Scheme: Evaluation structure (i.e. HTML tags) and evaluation logic (i.e. the full declarative definition of the checkpoints to be processed and the actions to be taken when deviation with respect to any checkpoint is detected)
– Involvement in Computation (input / output to, side effect of calculation): input

**UE- Input Parameter 2 – Selection of U&A Guidelines with respect to Webpage Characteristics**
– Explanatory Definition: A selected set of guidelines (on-demand evaluation) that are considered to be relevant for the webpage being checked
– Relationship to other Parameters: [User profile]
– Formal Encoding (incl. scale, value set): GDL-compliant form
– Elicitation/Acquisition Method: The user of the evaluation engine can manually select guidelines by looking up the tables of content
– Mapping to Computation Scheme: [not known]
– Involvement in Computation: Input

**UE-Output Parameter 1 – User Profile**
– Explanatory Definition: User profile - Expert profile (i.e. human factors expert who wants to know which guidelines are violated and how serious and frequent the violation) and Designer Profile (i.e. a website designer needs to know where the errors are located in the HTML code and how to correct them)
– Relationship to other Parameters: Navigation structure
– Formal Encoding (incl. scale, value set): Expert vs. designer
– Elicitation/Acquisition Method: [user input]
– Mapping to Computation Scheme: [not known]
– Involvement in Computation: output

**UE-Output Parameter 2 - Menu**
– Explanatory Definition: Dynamic and modifiable according to user preference
– Relationship to other Parameters: [User profile]
– Formal Encoding (incl. scale, value set): Guidelines can be sorted by “Source”, “Ergonomic Aspect” and “HTML Object”
– Elicitation/Acquisition Method: Selection of item in a combination box
– Mapping to Computation Scheme: [not known]
– Involvement in Computation: output

**UE-Output Parameter 3 – The Main Frame**
- Explanatory Definition: Three different page types - global statistics of the evaluation; selected sorting criteria; webpage under evaluation
- Relationship to other Parameters: Descriptive statistics by pages and by source of guidelines
- Formal Encoding (incl. scale, value set): Number of pages having errors at different priority levels; Number of pages having violated guidelines from different sources; Criteria most/least respected; For each evaluated page some local statistical graphics and theory about the selected criteria; List of wrong instances and their location in the HTML code
- Elicitation/Acquisition Method: Navigational design for the web pages of evaluation results
- Mapping to Computation Scheme: [not known]
- Involvement in Computation: output

**UE-Principle/Parameter Relationship Matrix**

<table>
<thead>
<tr>
<th></th>
<th>Effectiveness</th>
<th>Flexibility</th>
<th>Self-descriptiveness</th>
<th>Controllability</th>
<th>Suitability for Individualization</th>
<th>Suitability for Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected Guidelines</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>GDL-compliant form</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>HTML elements</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>User profile</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Navigation structure</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Statistical data</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Level of details of items in an evaluation report</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Report delivery mode</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

**Computation Scheme**

[not given]

**Best Practice**

- Method Application Sector: Basically all sorts of websites and all types of guidelines
- Setting or Context of Use (users, location etc.): Two main groups of users – usability practitioners and website designers [no restriction on location]
- Procedure (step 1 to n to follow when using of UEM):
  i. Collecting U&A data with their corresponding metrics such as task completion time, errors, guideline violation and subjective ratings
  ii. Analyzing collected U&A data to detect U&A problems in the website
  iii. Reporting analysis results to the end user(e.g. website designer, owner and
iv. Suggesting solutions or improvements to repair the previously detected problems
   – Documentation (how is the process, how are the results documented): Automatically generated evaluation reports of which the granularity of the content, the presentation of the content as well as the delivery mode can be regulated by the user of the evaluation tool
   – Particularities (is there anything to be considered when using the UEM): [not given]

Encountered Usability Problems when Using UEM
[not given; the related work on evaluating the usability of the DESTINE evaluation tool is ongoing]

Other encountered Problems when Using UEM
[not given; the related work on evaluating the usability of the DESTINE evaluation tool is ongoing]

Summary of Advantages
It is possible to generate usability reports with an unprecedented level of flexibility, and the reports can be delivered through various modes [no user-based empirical evidence about the usability/perceived usefulness of the tool is yet available]

Estimated impact
[it sounds to be a useful tool; may have some significant impact on the industry]

Open research questions
[User acceptance towards the evaluation tool; the reliability and validity of the evaluation tool; benchmarking with other automated evaluation tools]

Other comments
[An interesting applied research]

Description of Case Studies
[no details are given]
Cognitive Walkthrough for the Web

Reviewer: Peter Eberle

Method Identification

**UEM Name**
CWW – Cognitive Walkthrough for the Web


**Abstract**

The new Cognitive Walkthrough for the Web (CWW) is superior for evaluating how well websites support users' navigation and information search tasks. The CWW uses Latent Semantic Analysis to objectively estimate the degree of semantic similarity (information scent) between representative user goal statements (100-200 words) and heading/link texts on each web page. Using an actual website, the paper shows how the CWW identifies three types of problems in web page designs. Three experiments test CWW predictions of users' success rates in accomplishing goals, verifying the value of CWW for identifying these usability problems.

**UEM – Type**

analytical UEM.

**UEM- History**

Theoretical foundations are:

- CoLiDeS Simulation Model of Website Navigation: CoLiDeS, an acronym for Comprehension-based Linked model of Deliberate Search, extends a series of earlier models of performing by exploration based on Kintsch’s construction-integration theory of text comprehension and problem solving processes.

- Latent Semantic Analysis, (LSA) LSA is a mathematical technique that estimates the semantic relatedness of texts, based on a statistical analysis of a large corpus. (http://lsa.colorado.edu/)

**Method Characteristics**

**Objectives**

- The method is superior for evaluating how well websites support users’ navigation and information search tasks.

- The method should identify usability problems that hinder users to accomplish their information search tasks.

**Usability Principle(s) Addressed by the UEM**

[since the usability principles are not given they have to be interpreted from the evaluation questions:]

Q1) Will the correct action be made sufficiently evident to the user?

→ [self descriptiveness 1]

Q2) Will the user connect the correct action’s description with what he or she is trying to
do?
→ [self descriptiveness 2]
Expressed by the cosine of the angle of two high dimensional vectors. Vector 1 is calculated from the 100-200 word statement of the user goals and vector B is calculated from the link texts.
Q3) Will the user interpret the system’s response to the chosen action correctly?
→ [self descriptiveness 3]

**UE-Parameters**

**Input Parameter 1: user goals**
- Explanatory Definition: Representative set of user [information seeking] goals
- Relationship to other Parameters (give both, type of relationship & parameter): [none]
- Formal Encoding (incl. type): 100-200 word statements
- Elicitation/Acquisition Method when measured: [Compiled by the analyst]
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Input parameter for the calculation of the semantic similarity between the goal and each of the headings and links of the page
- Involvement in Computation (input / output to, side effect of calculation): Input for the calculation of self descriptiveness1

**Input Parameter 2: headings and link labels**
- Explanatory Definition: The analyst has to submit headings and link labels respective one and two word extractions of the headings and link label to the LSA
- Relationship to other parameters (give both, type of relationship & parameter): [none]
- Formal encoding (incl. type): one or two word extractions of the headings and links
- Elicitation/Acquisition Method when measured: Compiled by the analyst
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Input parameter for calculating the similarity of different headings/link labels
- Involvement in Computation (input / output to, side effect of calculation): Input

**Input Parameter 3: semantic spaces**
- Explanatory Definition: semantic spaces [should] represent the level of reading comprehension ability and background knowledge of specific user groups.
- Relationship to other Parameters (give both, type of relationship & parameter): [none]
- Formal Encoding (incl. type): [not given]
- Elicitation/Acquisition Method when measured: [not given]
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Input parameter for calculating the familiarity/unfamiliarity of the headings/link labels
- Involvement in Computation (input / output to, side effect of calculation): Input for selfdescriptiveness1 -3
Output Parameter 1: Semantic similarity between the goals, headings and link labels = self descriptiveness 1
- Explanatory Definition: Estimations of the semantic similarity between the 100-200 word goal description and each of the headings and link labels of the page for a specific semantic space.
- Relationship to other Parameters (give both, type of relationship & parameter): none
- Formal Encoding (incl. type): numeric table of cosines which represents the angels between the high dimensional term vectors calculated through the LSA.
- Elicitation/Acquisition Method when measured: Analysts submits each detailed user goal statement, each heading and each link label on the page to the LSA. The LSA computes high lever term vectors and give as result a matrix of the cosines of the angels of the single vectors.
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): the user will select the headings/links with the top ranking cosines. Pairs yielding a cosine of 0.6 or more are likely to be confuseable.
- Involvement in Computation (input / output to, side effect of calculation) : Output

Output Parameter 2: Unfamiliar headings/ link labels = self descriptiveness 2
- Explanatory Definition: the user will not know what the heading/mean because it is unfamiliar according to his expected semantic space.
- Relationship to other Parameters (give both, type of relationship & parameter):
- Formal Encoding (incl. type): numeric (term vector length of the link labels; if the link labels are longer than 2 words, the two most important words are selected [criteria not given, selection is presumably conducted by the analyst]). The length of the term vector of the LSA estimates the how much knowledge of the element is embedded in the designated semantic space.
- Elicitation/Acquisition Method when measured: Result of the LSA (“term to term” and “one to many comparison”).
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): vector length of less than 0.8 is likely to be unfamiliar.
- Involvement in Computation (input / output to, side effect of calculation) : Output

UE-Parameter 3: Confusable headings/link labels
- Explanatory Definition: how body text (i.e., text that is not headings or links) Similar headings and link labels [for different content] is confusing
- Relationship to other Parameters (give both, type of relationship & parameter): none
- Formal Encoding (incl. type): numeric (cosines)
- Elicitation/Acquisition Method when measured: Result of the LSA.
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): cosines higher than 0.6 is likely to be confuseable.
- Involvement in Computation (input / output to, side effect of calculation) :
Output

**UE-Principle/Parameter Relationship Matrix**

<table>
<thead>
<tr>
<th>UE-Parameter 1: User goals</th>
<th>UE-Principle “self-descriptiveness”</th>
</tr>
</thead>
<tbody>
<tr>
<td>UE-Parameter 2: Semantic similarity of goals, headings and link labels</td>
<td>x</td>
</tr>
<tr>
<td>UE-Parameter 3: Unfamiliar headings/ link labels</td>
<td>x</td>
</tr>
<tr>
<td>UE-Parameter 4: Confusable headings/link labels</td>
<td>x</td>
</tr>
</tbody>
</table>

**Computation Scheme**


**Best Practice of UEM Application**

- Setting or Context of Use (users, location etc.): a computer with access to the internet is necessary. Access to the LSA.
- Method application to software lifecycle stage: design and evaluation
- Prerequisites / Constraints (limitations of the method, e.g. resources; skills and knowledge of users and operator; apparatus required; software required; learnability of the method: Analyst has to be a method expert, otherwise the results may not be valid.
- Procedure (step 1 to n to follow when using of UEM):
  1. Compiling a Set of Realistic User Goals and Intended Selections
  2. Using LSA to Estimate Semantic Similarity of Goals, Headings, and Link Labels
  3. Identifying Problematic Heading/Link Labels
  4. Finding Goal-Specific Problems

**Encountered usability problems when using UEM**

User cannot achieve their navigation and information search tasks due to:

- Semantic similarity of goals, headings and link labels
- Unfamiliar headings/ link labels
- Confusable headings/link labels

These problems can be characterised as global or goal specific problems.

**Other encountered drawbacks when using UEM**

LSA is likely to overestimate the similarity of items

The analyst must decide what portion of each link label text she submits to the LSA.

The semantic spaces have not been tested, [and only less are available]

**Summary of Advantages**

Analytic goal and user group specific investigation of navigational problems.

**Estimated impact**

[?]  

**Open research questions**
[how to construct the semantic spaces]
[how to increase the objectivity of the method]

**Other comments**
[questionable availability of the needed “ontologies” i.e. semantic spaces]

**Description of Case Studies**
A small case study is published in:
Keystroke Level Model

Reviewer: Peter Eberle

<table>
<thead>
<tr>
<th>Method Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UEM Name</strong></td>
</tr>
<tr>
<td>Keystroke- Level Model (KLM)</td>
</tr>
</tbody>
</table>

**Abstract**
The Keystroke-Level Model (KLM), proposed by Card, Moran, & Newell (1983), predicts task execution time from a specified design and specific task scenario. Basically, you list the sequence of keystroke-level actions the user must perform to accomplish a task, and then add up the times required by the actions. It is not necessary to have an implemented or mocked-up design; the KLM requires only that the user interface be specified in enough detail to dictate the sequence of actions required to perform the tasks of interest. The actions are termed keystroke level if they are at the level of actions like pressing keys, moving the mouse, pressing buttons, and so forth, as opposed to actions like "log onto system" which is much more abstract. The KLM requires that you describe how the user would do the task in terms of actions at this keystroke level. The basic actions are called operators, in the sense of operators in the Model Human Processor discussion. There is a standard set of operators for use in the KLM, whose execution times have been estimated from experimental data. This presentation is based on the Card, Moran, & Newell(1983), but has several extensions, especially concerning the placement of mental operators.

**UEM — Type**
analytical UEM.

**UEM - History**
KLM is part of the GOMS family:
- CMN-GOMS: The original formulation proposed in CMN was a loosely defined demonstration of how to express a goal and subgoals in a hierarchy, methods and operators, and how to formulate selection rules.
- KLM: A simplified version of CMN called the Keystroke-Level Model uses only keystroke-level operators—no goals, methods, or selection rules. The analyst simply lists the keystrokes and mouse movements a user must perform to accomplish a task and then uses a few simple heuristics to place “mental operators.”
- NGOMSL: A more rigorously defined version called NGOMSL presents a procedure for identifying all the - GOMS components, expressed in a form similar to an ordinary computer programming language. NGOMSL includes rules-of-thumb about how many steps can be in a method, how goals are set and terminated, and what information needs to be remembered by the user while doing the task.
- CPM-GOMS: A parallel-activity version called CPM-GOMS uses cognitive,
perceptual, and motor operators in a critical-path method schedule chart (PERT chart) to show how activities can be performed in parallel.

**Method Characteristics**

**Objectives**
The method shall acquire average task completion times of standardized tasks which have to been modelled with the KLM standard operators to compare different system designs

**Usability Principle(s) Addressed by the UEM**

[not given]

**UE-Parameters**

**UE-Input/Output Parameters: averaged completion times for the KLM standard operators:**

- Explanatory Definition:
  - K - Keystroke. This operator is pressing a key or button on the keyboard. Pressing the SHIFT or CONTROL key counts as a separate keystroke.
  - T(n) - Type a sequence of n characters on a keyboard (n x K sec). This operator is simply a short hard for a series of K operators, and would normally be used only when the user is typing a string of characters that is a single "chunk," such as a filename.
  - P - Point with mouse to a target on the display. This operator represents the action of moving the mouse to point the cursor to a desired place on the screen.
  - B - Press or release mouse button. This is a highly practiced, very rapid movement.
  - BB - Click mouse button. Pushing and releasing the mouse button rapidly.
  - H - Home hands to keyboard or mouse. The well practiced movement, between keyboard and mouse, and vice-versa.
  - M - Mental act of routine thinking or perception. This operator is based on the fact that when reasonably experienced users are engaged in routine operation of a computer, there are pauses in the stream of actions and that are associated with routine acts such as remembering a filename or finding something on the screen. The M operator is not intended to represent complex, lengthy, problem-solving, racking the brain, or creative meditations.
  - W(t) - Waiting for the system to respond. This is the time that the user must wait on the system before he or she can proceed.

- Relationship to other Parameters (give both, type of relationship & parameter):
  [not given]
- Formal Encoding (incl. type): seconds or milliseconds (number)
- Elicitation/Acquisition Method when measured: through empirical investigations
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): All execution times are summarized
- Involvement in Computation (input / output to, side effect of calculation) : Input/Output

**UE-Principle/Parameter Relationship Matrix**

[not given]

**Computation Scheme**

the operators are substituted through their completion times and summarized
Best Practice of UEM Application

- Method Application sector – specific domains [not given]
- Setting or Context of Use (users, location etc.): task scenarios must be known
- Method application to software lifecycle stage: design, [testing, maintenance]
- Prerequisites / Constraints (limitations of the method, e.g. resources; skills and knowledge of users and operator; apparatus required; software required; learnability of the method: Method is only appropriate for
  - tasks, accomplished by experienced users, routine cognitive skills
  - sequential (not parallel) tasks
- list of high-level user goals must exist, which is not provided by GOMS analyses and techniques, and had to come from external sources
- GOMS provide no simulations of mental models of the users about the system
- The only result parameter is the total completion time, no measurements closer to usability principles are given.
- Procedure (step 1 to n to follow when using of UEM):
  1. Choose one or more representative task scenarios.
  2. Have the design specified to the point that keystroke-level actions can be listed for the specific task scenarios.
  3. For each task scenario, figure out the best way to do the task, or the way that you assume users will do it.
  4. List the keystroke-level actions and the corresponding physical operators involved in doing the task.
  5. If necessary, include operators for when the user must wait for the system to respond
  6. Insert mental operators for when user has to stop and think.
  7. Look up the standard execution time to each operator.
  8. Add up the execution times for the operators.
  9. The total of the operator times is the estimated time to complete the task.
- Documentation (how is the process, how are the results documented): list of the keystroke level actions is created. The keystroke level actions are added up with the provided standard execution time.
- Particularities (is there anything to be considered when using the UEM): [ ]

Encountered usability problems when using UEM
[whether global long task completion times caused by long winded designs]

Other encountered drawbacks when using UEM
heuristics exists only for the insertion of the mental operator
[no relationships to usability principles such as task conformance, rather global, e.g. efficiency ≠ time]

Summary of Advantages
fast method (compared with the other GOMS techniques)

Estimated impact
[not given]

Open research questions
[value of the results? → what does it predict?

Other comments
[missing context (task, criteria)]

Description of Case Studies
Case studies are e.g. published in:
ProcessLens

Reviewer: Chris Stary

Method Identification

**UEM Name**
ProcessLens


**Abstract**
This chapter shows how specifications of mobile multimedia applications can be checked against usability principles very early in software development through an analytic approach. A model-based representation scheme keeps transparent both, the multiple components of design knowledge as well as their conceptual integration for implementation. The characteristics of mobile multimedia interaction are captured through accommodating multiple styles and devices at a generic layer of abstraction in an interaction model. This model is related to context representations in terms of work tasks, user roles and preferences, and problem-domain data at an implementation-independent layer. Tuning the notations of the context representation and the interaction model enables, prior to implementation, to check any design against fundamental usability-engineering principles, such as task conformance and adaptability. In this way also alternative design proposals can be compared conceptually. As a consequence, not only the usability of products becomes measurable at design time, but also less effort has to be spent on user-based ex-post evaluation requiring re-design.

**UEM- Type**
Model-based UEM

**UEM- History**
The method is based on the model-based TADEUS approach to generate work-task-based user-interface design (Stary, 2000). It has been re-implemented by Heftberger and elaborated in the context of organisational learning (cf. Heftberger et al., 2005) Heftberger, S.; Stary, Ch.: Participative Organizational Learning. A Process-based Approach (in German), Deutscher Universitätsverlag, Wiesbaden, 2005.


**METHOD CHARACTERISTICS**

**Objectives**
The UEM aims to ensure task conformance and flexibility of user interfaces at the specification level, according to the ISO standard 9241. Both principles are of major relevance, since (i) most interactive systems are part of work systems, and are supposed to support work tasks, (ii) individualisation, both, in terms of device adaptation, and user customisation, influences user acceptance directly. In case specifications can be checked against those principles, development effort can be reduced, and user acceptance can be increased. As an output of the UEM the developers receive a status report whether task conformance and/or suitability for individualisation are met in principle, given a specific design representation.

**Usability Principle(s) Addressed by the UEM**

**UE-Principle 1**
- **Name:** task conformance
- **Explanatory definition (incl. ref. to standard):** “A dialogue supports task conformance, if it supports the user in the effective and efficient completion of the task. The dialogue presents the user only those concepts which are related to the task” (ISO 9241 Part 10).
- **Dimension / target / focus:** [technology, human]
- **Addressed (list of) parameters:** It can be measured in an analytical way through two metrics: completeness with respect to task accomplishment (effectiveness), and efficiency. The first metric means that at the system level there have to be both, a path to be followed for user control (i.e. navigation procedures), and mechanisms for data presentation and manipulation, in order to accomplish a given task. The second metric is required to meet the principle in an optimised way. High efficiency in task accomplishment can only be achieved, in case the users are not burdened with navigation tasks and are able to concentrate on the content to be handled in the course of task accomplishment.
- **Relationships to other Principles (give both, type of relationship and principle):** effectiveness, efficiency

**UE-Principle 2**
- **Name:** suitability for individualization
- **Explanatory definition (incl. ref. to standard):** In ISO 9241, Part 10 'Dialogue systems are said to support suitability for individualization if the system is constructed to allow for adaptation to the user’s individual needs and skills for a given task.’
- **Dimension / target / focus:** [technology, human]
- **Addressed (list of) parameters:** [Flexibility]

...in general, interactive applications have to be adaptable in a variety of ways: (i) to the organisation of tasks; (ii) to different user roles; (iii) to various interaction styles, and (iv) to assignments. Adaptable skills means to provide more than a single option, and to be able to switch between the various options for each of the issues (i) – (iv).
- **Relationships to other Principles (give both, type of relationship and principle):**
  - **Task conformance**

**UE-Parameters**

**UE-Parameter 1**

[Effective navigability]:
- **Explanatory Definition:** According to the ISO-definition above task conformance requires (a) the provision of an accurate functionality by the software system to accomplish work tasks ...
- **Relationship to other Parameters (give both):** [efficient navigability]
- **Formal Encoding (incl. scale, value set):** In terms of specifications this understanding of task conformance means to provide task specifications (task models) and refinements to procedures both, at the problem domain level (data and user models) and at the user interface level (interaction and application models).
- **Elicitation/Acquisition Method:** [inputs provided by the design representation]
- **Mapping to Computation Scheme:** [Setting of causal and behaviour relationships between task elements and task- or domain-specific presentation elements, i.e. links between nodes (model elements).]

The corresponding algorithms are designed in such way that they scan the entire set of design representations. Basically, the meaning of constructs is mapped to
Constraints that concern ORDs and OBDs of particular models, as well as OIDs when OBDs are synchronised.

- Involvement in Computation: [element of constraint – there needs to be at least a chain of links starting with a task node, ending in an interaction node in the application model]

**UE-Parameter 2**
[Efficient navigability]:

- Explanatory Definition: According to the ISO-definition above task conformance requires (a) ... , and (b) minimal mental load for performing interaction tasks at the user interface to accomplish these tasks.
- Relationship to other Parameters (give both): [effective navigability]
- Formal Encoding (incl. scale, value set): length of navigation path [i.e. number of links between nodes]
- Elicitation/Acquisition Method: [inputs provided by the design representation]
- Mapping to Computation Scheme: At the specification level an algorithm has to check whether the shortest path in the application model for a given task, device and style of interaction has been selected for a particular solution, in order to accomplish the control and data-manipulation tasks. In order to achieve (b) [see efficiency], based on an adequate modality the number of user inputs for navigation and manipulation of data has to be minimised, whereas the software output has to be adjusted to the task and user characteristics. The latter is performed at the user, interaction and application-model level. As a result, the task-specific path of an OBD or OID should not contain any procedure or step that is not required for task accomplishment. It has to be minimal in the sense, that only those navigation and manipulation tasks are activated that directly contribute to the accomplishment of the addressed work task.
- Involvement in Computation: [minimal number of links between task node and interaction node in the application model]

**UE-Parameter 3**
[Flexibility]:

- Explanatory Definition: Adaptability with respect to the organisation of tasks means that a particular task might be accomplished in several ways. Hence, a specification enabling flexibility with respect to tasks contains more than a single path in an OBD or ORD within a task (implemented through ‘before’-relationships) to deliver a certain output given a certain input. Flexibility with respect to user roles means that a user might have several roles and even switch between them, eventually leading to different perspectives on the same task and data. Adaptability with respect to interaction devices and styles, i.e. the major concern for polymorph multimedia application development, does not only require the provision of several devices or styles based on a common set of dialog elements, as shown in Figure 4.2 for Graphical User Interfaces and browser-based interaction, but also the availability of more than a single way to accomplish interaction tasks at the user interface, for instance direct manipulation (drag & drop) and menu-based window management. The latter can be checked again at the OBD/OID-level, namely through checking whether a particular operation, such as closing a window, can be performed along different state transitions. For instance, closing a window can be enabled through a menu entry or a button located in the window bar.
Relationship to other Parameters (give both): [none, not given]

Formal Encoding (incl. scale, value set): number of paths [i.e. number of links between nodes (model elements)]

Elicitation/Acquisition Method: [not given]

Mapping to Computation Scheme: See Explanatory Definition.

Involvement in Computation: [counter]

### UE-Principle/Parameter Relationship Matrix

<table>
<thead>
<tr>
<th>Task conformance</th>
<th>Suitability for individualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Effective navigability]</td>
<td>X</td>
</tr>
<tr>
<td>[Efficient navigability]</td>
<td>X</td>
</tr>
<tr>
<td>[Flexibility]</td>
<td>X</td>
</tr>
</tbody>
</table>

### Computation Schemes

**TASK CONFORMANCE:**

- Calculus: [not named]
- Representational Elements (facts, rules o.t.l.): UML elements

Following the tradition of model-based design representations (cf. Szekely, 1995; Puerta, 1996; Stary, 2000) several models are required to define (executable) specifications of interactive software:

- a task model - that part of the situational setting (organisation, public domain etc.) the interactive computer system is developed for
- the user model – the set of user roles to be supported, both, with respect to functional roles (e.g. search for information), and with respect to individual interaction preferences and skills (e.g., left-handed pointing)
- a problem domain (data) model – the data and procedures required for task accomplishment
- the interaction (domain) model – providing those interaction styles that might be used for interactive task accomplishment.

For processing, each model is described in an object-oriented way, namely through a structure diagram, i.e. an ORD (Object Relationship Diagram), and a behaviour (state/transition) diagram, i.e. an OBD (Object Behaviour Diagram). Structural relationships are expressed through relationships, linking elements either within a model or stemming from different models. Behaviour relationships are expressed through linking OBDs at the state/transition level, either within a model or between models. Both cases result in OIDs (Object Interaction Diagrams).

- Algorithm / Computation: ... in order to ensure task conformance, two different types of checks have to be performed: (i) completeness of specification, and (ii) minimal length of paths provided for interactive task accomplishment (only in case check (i) has been successful). Check (i) ensures that the task is actually an interactive task (meaning that there exists a presentation of the task at the user interface) as well as there exists an interaction modality to perform the required interactive data manipulation to accomplish the addressed task.

For (ii), initially, the checker determines whether the task in the task-model OBD is linked to an element of an ORD of the interaction model to present the task at the user interface (through the relationship “is-presented”). Then it checks links throughout the entire specification whether the task is linked to data that have to be manipulated interactively to accomplish a task. In order to represent that relationship between tasks and problem-domain data structures, the ‘is-based-on’-relationship has to be used. Each interactive task also requires a link to
interaction-model elements which present the work-task data for interactive manipulation to the user. Finally, each task requires a link to a user role (expressed through ‘handles’). Once the completeness of a task specification has been checked, each path related to that task in an OBD is checked, whether there exists a path that is not related to the data manipulation of that task. This way, the specification is minimal in the sense, that the user is provided with only those dialogues that are necessary to complete the addressed task through the assigned role. Note, that there might be different paths for a task when handled through different roles. This property relates to adaptability which is discussed after the implementation sample for an algorithm.

The persistence algorithms have been implemented as inherent part of the ProcessLens environment, namely in the set of model editors for designers and the prototyping engine. In the following the algorithm for the check of consistent task model hierarchies (see i.) task model check – static view) is exemplified in Java pseudo code (cf. Eybl, 2003; Heftberger et al., 2004; p. 137f)

```java
checkTaskModelConsistency () {
    taskModel = get TaskModel();
    allBeforeRelations = new Vector();
    enum = taskModel.getElements();
    while (enum.hasMoreElements()) {
        element = enum.nextElement();
        enum2 = bmod.getRelations();
        error = false;
        while (enum2.hasMoreElements()) {
            relation = enum2.nextElement();
            if (relation instanceof BeforeRelation) {
                if (!allBevorRelations.contains(relation))
                    allBevorRelations.addElement(relation);
            }
        }
        enum3 = allBevorRelations.elements();
        while (enum3.hasMoreElements()) {
            rel = enum3.nextElement();
            if (rel instanceof BevorRelation) {
                checkConsistency2(rel.getStartElement(), rel.getEndElement());
                checkConsistency3();
            }
        }
    }
    if (error) println("inconsistent ‘before’-relationship in task model");
}
checkConsistency2 (Element start, Element end) {
    startVector = new Vector();
    endVector = new Vector();
    startVector.addElement(start);
    endVector.addElement(end);
    fill startVector with all elements that have direct and indirect is_part_of relation with element ‘start’
    fill endVector with all elements that have direct and indirect is_part_of relation with element ‘end’
}
checkConsistency3 () {
    enum = startVector.elements();
    ...}
```
while (enum.hasMoreElements()) {
    source = enum.nextElement();
    enum2 = endVector.elements();
    while (enum2.hasMoreElements()) {
        destination = enum2.nextElement();
        checkBeforeCycles(source, destination);
    }
}

checkBeforeCycles(Element predecessor, Element successor) {
    enum = successor.getAssociationRelations();
    while (enum.hasMoreElements()) {
        association = enum.nextElement();
        if (association instanceof BeforeRelation) {
            beforeRelation = association;
            next = beforeRelation.getEndElement();
            if (!(next.equals(successor))) {
                if (next.equals(predecessor)) {
                    error = true;
                    return;
                }
                checkBeforeCycles(predecessor, next);
            }
        }
    }
    checkBeforeCycles(predecessor, next);
}

In CheckConsistency() those elements of the task model that are not leave nodes in the task hierarchy are processed. When being part of a before-relation the method CheckConsistency2(start, end) is executed. 'Start' denotes the start element of the 'before'-relation, and 'end' the end element of the 'before'-relation. In CheckConsistency2(start, end) all elements are stored in a vector which have a direct or indirect is_part_of-relation with the 'start' element (startVector). Another vector has to be set up for the 'end' element, i.e. the endVector. In checkConsistency3() it is checked whether an element of endVector is part of a before-relation involving an element of startVector. If such an element can be found, the task model has inconsistent before-relations and an error is reported.

The checker indicates an exception, as long as the specifications do not completely comply to the defined semantics. For instance, the correct use of the 'before' relationship in the task model requires to meet the following constraints: (i) at the structure layer: 'Before' can only be put between task specifications, (ii) at the behaviour layer: The corresponding OBDs are traced whether all activities of task 1 (assuming task 1 ‘before’ task 2) have been completed before the first activity of task 2 is started. Hence, there should not exist synchronisation links of OIDs interfering the completion of task 1 before task 2 is evoked.

- Result(s) incl. Rules for Interpretation: [consistency error remove inconsistency, (minimal) path detection task conform behaviour]

**Best Practice – TASK CONFORMANCE CALCULATION**
- Method Application Sector: [not specified, but can be applied at designSpecification phase of any type of interactive software]
- Setting or Context of Use (users, location etc.): [design representations have to
be available, at least one work tasks has to be decomposed and refined to
interactions at the user interface]

– Method Application to Software Lifecycle: Design, Specification
– Prerequisites, Constraints: see also setting; limitations: none ; apparatus
required: UML structure and behaviour specification, software required:
ProcessLens editor; learnability: [as for UML].
– Procedure: [walk through specification / push the button; interpret results]
– Particularities: [not reported]

**Encountered Usability Problems (as Outcome) Using UEM – TASK CONFORMANCE**

– Problem Description: [not reported]
– Reasons: n.a.
– Suggestions for Improvement: n.a.

**SUITABILITY FOR INDIVIDUALIZATION**

– Calculus: [not named]
– Representational Elements (facts, rules o.t.l.): UML – See Computation Scheme
  for Task Conformance
– Algorithm / Computation:
  – Actually, to ensure adaptability (i) the designer has to set up the proper design
    space, and (ii) modifications have to occur in line with the semantics of the
definition space. The first objective can be met through providing relationships
between design items, both, within a model (e.g., ‘before’), and between
models (e.g., ‘handles’). The first relationship enables flexibility with each of the
perspectives mentioned above, e.g., the organisation of tasks, whereas the
second relationship allows for flexible tuning of components, in particular the
assignment of different styles of interaction to tasks, roles, and problem domain
data.
The second objective can be met allowing to manipulate relationships according
to the syntax and semantics of the representation scheme (i.e. specification
language), and providing algorithms to check the correct use of the language as
well as the changes related to the manipulation of relationships. For instance,
the ‘before’ relationship can only be set between sub tasks in the task model,
since the algorithm for ‘before’ described previously processes the design
representation: In case the relationship is modified, e.g., ‘before’ set between
other sub tasks, the restrictions that applied to the OBDs are lifted, and new
ones, according to the semantics of the relationship are enforced, however,
using the same algorithm.

Adaptability of assignments involves (i) – (iii) as follows: In case a user is
assigned to different tasks or changes roles (in order to accomplish a certain
task) assignments of and between tasks and roles have to be flexible. In case a user
wants to switch between modalities or to change interaction styles (e.g., when
leaving the office and switching to mobile interaction, the assignment of
interaction styles to data and/or tasks have to be modified accordingly.
Changing assignments requires links between the different entities that are
activated or de-activated at a certain point of time. It requires the existence of
assignment links as well as their dynamic manipulation. Both has been provided
in the environment for modeling user interaction, either through additional
semantic relationships, e.g., ‘handles’ between roles and tasks (linking different
models), or the runtime environment of the prototyping engine.
- Result(s) incl. Rules for Interpretation: [single link (i) – (iv) re-design required, more than one link - suitability for individualization provided]

**Best Practice – SUITABILITY FOR INDIVIDUALIZATION CALCULATION**
- Method Application Sector: [not specified, but can be applied at design/specification phase of any type of interactive software]
- Setting or Context of Use (users, location etc.): [design representations have to be available, at least one work tasks has to be decomposed and refined to interactions at the user interface]
- Method Application to Software Lifecycle: Design, Specification
- Prerequisites, Constraints: see also setting; limitations: none; apparatus required: UML structure and behaviour specification, software required: ProcessLens editor; learnability: [as for UML].
- Procedure: [walk through specification / push the button; interpret results]
- Documentation: manual report / report on the screen
- Particularities: [not reported]

**Encountered Usability Problems (as Outcome) Using UEM – SUITABILITY FOR INDIVIDUALIZATION**
- Problem Description: [not reported]
- Reasons: n.a.
- Suggestions for Improvement: n.a.

**Encountered Problems of Using UEM**
- Problem Description (e.g., reliability): [not given]
- Reasons: n.a.
- Suggestions for Improvement: n.a.

**Summary of Advantages**
[Although no case study data have been provided, the sample studies in Heftberger et al., 2005 reveal the potential, given by implementation-independent usability checks in terms of task conformance and suitability for individualization: less development effort, increased user acceptance.]

**Estimated impact**
[High, both to research and development.]

**Open research questions**
Can all ISO principles be addressed in this way?]

**Other comments**
> none

**Description of Case Studies – not given**
WebRemUSINE

Reviewer: Peter Eberle

Method Identification

**UEM Name**
WebRemUSINE


**Abstract**
The main goal of this work is to propose a method to evaluate user interfaces using task models and logs generated from a user test of an application. The method can be incorporated into an automatic tool which gives the designer information useful to evaluate and improve the user interface. These results include an analysis of the tasks which have been accomplished, those which failed and those never tried, user errors and their type, time-related information, task patterns among the accomplished tasks, and the available tasks from the current state of the user session. This information is also useful to an evaluator checking whether the specified usability goals have been accomplished.

**UEM – Type**
empirical / analytical UEM.

**UEM- History**
[not given]

Method Characteristics

**Objectives**
The method shall evaluate user interfaces combining task models and logs, generated from a user test of an application, calculate measures to enable a judgement about the interface quality or usability prediction.
The method should facilitate and give automatic support to the evaluation process, show differences and similarities to the highly rated designs [these are rated with expert ratings].
The method wants to decrease the involvement of the usability specialist during the evaluation process.
The method should give relevant information for improvements.

**Usability Principle(s) Addressed by the UEM**
Efficiency: effectiveness (how correctly and completely the goals are achieved) divided by the time.
Learnability (a certain degree of learnability): measured by the time completion rate and the number of errors.
Relevance (a certain degree of relevance): success of the user in completing the desired tasks.

**UE-Parameters:**

**UE-Input Parameter 1: User logs**
- Explanatory Definition: Logged user interactions, while using the interface to accomplish some tasks.
- Relationship to other Parameters (give both, type of relationship & parameter): [not given]
- Formal Encoding (incl. type): textfile of actions with time stamp:
  - The type of user action such as a mouse click
  - The name of the widget affected, the target of the action, e.g., a list
– The mouse button used
– The content of the widget affected, such as an item in a list
– The coordinates of the mouse click
– The time elapsed from the previous log
– Elicitation/Acquisition Method when measured: logged with the Replay Tool
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): on part for the creation of the log-task table. Involvement in Computation (input / output to, side effect of calculation) : Input

**UE-Input Parameter 2: Task Model**
– Explanatory Definition: Describing the desired task in a hierarchical task tree using the ConCurTask editor. Both the task and the temporal relationship between the tasks are defined.
– Relationship to other Parameters (give both, type of relationship & parameter): The precondition table is automatically created from the task model
– Formal Encoding (incl. type): graphical and/or verbal defines task model
– Elicitation/Acquisition Method when measured: Created by the evaluator
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): the second part for the creation of the log-task table. Involvement in Computation (input / output to, side effect of calculation) : Input

**UE-Input Parameter 3: Preconditions**
– Explanatory Definition: Describing the desired task in a hierarchical task tree using the ConCurTask editor. Both the task and the temporal relationship between the tasks are defined.
– Relationship to other Parameters (give both, type of relationship & parameter): [not given]
– Formal Encoding (incl. type): [?] 
– Elicitation/Acquisition Method when measured: Automatically generated from the Task Model
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme):
– Involvement in Computation (input / output to, side effect of calculation) : Input

**UE-Output Parameter 1: Accomplished Tasks**
– Explanatory Definition: Tasks with and without preconditions that the user accomplishes
– Relationship to other Parameters (give both, type of relationship & parameter): [not given]
– Formal Encoding (incl. type): alphanumeric list, numeric quantity
– Elicitation/Acquisition Method when measured: comparision between task-log and task model.
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, and the user logs, the log-task table, and the precondition table
– Involvement in Computation (input / output to, side effect of calculation) : Output
**UE-Output Parameter 2: Failed Tasks**
- Explanatory Definition: Tasks which failed because the preconditions were not satisfied.
- Relationship to other Parameters (give both, type of relationship & parameter): [not given]
- Formal Encoding (incl. type): alphanumeric list [if they failed by reason of precondition errors], numeric quantity
- Elicitation/Acquisition Method when measured: none, is calculated
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, and the user logs, the log-task table, and the precondition table
- Involvement in Computation (input / output to, side effect of calculation) : Output

**UE-Output Parameter 3: Tasks never tried**
- Explanatory Definition: Tasks with and without preconditions that the user accomplishes
- Relationship to other Parameters (give both, type of relationship & parameter): [not given]
- Formal Encoding (incl. type): [alphanumeric list]
- Elicitation/Acquisition Method when measured: none, is calculated
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-task table, precondition table
- Involvement in Computation (input / output to, side effect of calculation) : Output

**UE-Output Parameter 4: Precondition errors**
- Explanatory Definition: Preconditions which are needed to perform a task were not fulfilled
- Relationship to other Parameters (give both, type of relationship & parameter): [not given]
- Formal Encoding (incl. type): numeric (quantity and time of occurrence)
- Elicitation/Acquisition Method when measured: is calculated
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-task table, precondition table
- Involvement in Computation (input / output to, side effect of calculation) : Output

**UE-Output Parameter 5: Unneeded actions**
- Explanatory Definition: Actions which are not needed to perform any tasks. Relationship to other Parameters (give both, type of relationship & parameter): [not given]
- Formal Encoding (incl. type): numeric (quantity and time of occurrence)
- Elicitation/Acquisition Method when measured: none, is calculated
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-
task table
  – Involvement in Computation (input / output to, side effect of calculation): Output

**UE-Output Parameter 6: Tasks completion times**
  – Explanatory Definition: How long each task took to complete
  – Relationship to other Parameters (give both, type of relationship & parameter): [not given]
  – Formal Encoding (incl. type): [alphanumeric task list, with task completion times]
  – Elicitation/Acqisition Method when measured: none, is calculated
  – Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-task table
  – Involvement in Computation (input / output to, side effect of calculation): Output

**UE-Output Parameter 6: Tasks completion times for abstract tasks**
  – Explanatory Definition: How long abstract tasks (tasks which require complex actions whose performance allocation has not yet been decided e.g., a user session with a system) took to complete.
  – Relationship to other Parameters (give both, type of relationship & parameter): [not given]
  – Formal Encoding (incl. type): [alphanumeric task list, with task completion times]
  – Elicitation/Acqisition Method when measured: none, is calculated
  – Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-task table
  – Involvement in Computation (input / output to, side effect of calculation): Output, useful for measuring efficiency

**UE-Output Parameter 7: The particular time, when the errors occurred**
  – Explanatory Definition: %
  – Relationship to other Parameters (give both, type of relationship & parameter): [not given]
  – Formal Encoding (incl. type): numeric
  – Elicitation/Acqisition Method when measured: none, is calculated
  – Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-task table, precondition table
  – Involvement in Computation (input / output to, side effect of calculation): Output. If the errors only occurred at the beginning it could mean the user should have more support and help at the beginning of the interaction.

**UE-Output-Parameter 8: Task patterns**
  – Explanatory Definition: Sequences of accomplished tasks that accomplished tasks that occur more than once.
  – Relationship to other Parameters (give both, type of relationship & parameter): [not given]
  – Formal Encoding (incl. type): [not given]
  – Elicitation/Acqisition Method when measured: none, is calculated
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): calculated using the task model, user logs, log-task table, precondition table
– Involvement in Computation (input / output to, side effect of calculation): Output. This can be used to identify where macros should be implemented.

**UE-Output-Parameter 9: Test time:**
– Explanatory Definition: Overall test time.
– Relationship to other Parameters (give both, type of relationship & parameter): [not given]
– Formal Encoding (incl. type): numeric
– Elicitation/Aquisition Method when measured: none, is calculated
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): [calculated using the task model, user logs, log-task table]
– Involvement in Computation (input / output to, side effect of calculation): Output

**UE-Output-Parameter 10: Frequency of scrollbar use:**
– Explanatory Definition: %
– Relationship to other Parameters (give both, type of relationship & parameter): [not given]
– Formal Encoding (incl. type): numeric
– Elicitation/Aquisition Method when measured: interaction log
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): [calculated using user logs]
– Involvement in Computation (input / output to, side effect of calculation): Output

**UE-Output-Parameter 11: Frequency of resizing windows:**
– Explanatory Definition: %
– Relationship to other Parameters (give both, type of relationship & parameter): [not given]
– Formal Encoding (incl. type): numeric
– Elicitation/Aquisition Method when measured: interaction log
– Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): [calculated using user logs]
– Involvement in Computation (input / output to, side effect of calculation): Output

**UE-Principle/Parameter Relationship Matrix**
[interpreted since not given]

<table>
<thead>
<tr>
<th></th>
<th>Efficiency</th>
<th>Learnability</th>
<th>Relevance</th>
<th>users attitude</th>
<th>safety</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accomplished tasks</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failed tasks</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tasks never tried</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Precondition errors</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----</td>
<td>-----</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unneeded actions</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task completion times</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task completion times for abstract tasks</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The particular time, when the errors occurred</td>
<td>[x]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task patterns</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test time</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frequency of scrollbar use</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frequency of resizing windows</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Computation Scheme**  
[not given]

**Best Practice of UEM Application**
- Method Application sector – specific domains [not given]
- Setting or Context of Use (users, location etc.): a computer with access to the internet is necessary. The pages to test must be able to be invoked. Web Page must be
- Method application to software lifecycle stage: testing, maintenance
- Prerequisites / Constraints (limitations of the method, e.g. resources; skills and knowledge of users and operator; apparatus required; software required; learnability of the method: task models must exist (modeled with the ConcurTaskTrees notation), Replay tool must be installed. Log Task table must be build manually (but system aided)
- Procedure (step 1 to n to follow when using of UEM):  
  1a. Logging the user actions with the Replay tool  
  1b. Creating the task models using the ConcurTaskTrees notation  
  2. Preparation Part: The log-task table has to be created (docomposition of the task model)  
  3. Result Part: The parameters are calculated.
- Documentation (how is the process, how are the results documented): [the analysis tool shows the evaluation results [above parameters]. [The results have to be interpreted by the expert]
- Particularities (is there anything to be considered when using the UEM):  

**Encountered usability problems when using UEM**

- tasks missed because of precondition errors, tasks never tried, unneeded actions, task patterns.

**Other encountered drawbacks when using UEM**
[not given]

**Summary of Advantages**
System aided analytical UEM based on task models.

**Estimated impact**

**Open research questions**

**Other comments**

**Description of Case Studies**
A small case study is published in:
WebTango

Reviewer: Peter Eberle

**Method Identification**

**UEM Name**
WebTango


Website http://webtango.berkeley.edu/

**Abstract**

The Web enables broad dissemination of information and services, yet most sites have in‐adequate usability and accessibility Numerous automated evaluation methodologies and tools have been developed to help designers to improve their sites. We describe the state-of-the-art in automated web site evaluation and then elaborate on our WebTango approach, which entails deriving design guidelines by mining empirical data. We compute over 157 measures, which assess many web interface aspects, and then use these measures along with expert ratings from Internet professionals as input to data mining algorithms. This mining process enables us to derive statistical models of highly rated web interfaces, such that the models reflect effective design patterns that are used on them in the automated analysis of their sites.

**UEM – Type**

analytical UEM.

**UEM- History**

[not given]

**Method Characteristics**

**Objectives**

– The method shall calculate measures to enable a judgement about the interface quality or usability prediction

– The method should show differences and similarities to the highly rated designs [these are rated with expert ratings]

– The method should give specific suggestions for improvements

**Usability Principle(s) Addressed by the UEM**

[none given]

**UE-Parameters:**

A set of 157 page-level and site-level measures have been developed. These are based on an extensive survey of design recommendations from recognized experts and usability studies:

**UE-Parameter 1: Link Elements (6)**

– Explanatory Definition: the number and type of links (e.g., graphic and text links) on a page.

– Relationship to other Parameters (give both, type of relationship & parameter): [not given]
− Formal Encoding (incl. type): numeric
− Elicitation/Acquisition Method when measured: crawler tool/software computation
− Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme):
− Involvement in Computation (input / output to, side effect of calculation) : input

**UE-Parameter 2: Graphic elements (6)**
− Explanatory Definition: the number and type of images (e.g., animated and link images) on a page.
→ further categories as UE Parameter 1

**UE-Parameter 3: Text Formatting: (24)**
− Explanatory Definition: how body text (i.e., text that is not headings or links) is emphasized; whether there is underlined text that is not in text links on the page; font styles and sizes; the number of text colors; the number of times text is re-positioned on the page; and how text areas are highlighted
→ further categories as UE Parameter 1

**UE-Parameter 4: Link Formatting: (3)**
− Explanatory Definition: whether there are text links that are not underlined and colors used for links.
→ further categories as UE Parameter 1

**UE-Parameter 5: Graphic Formatting: (7)**
− Explanatory Definition: the minimum, maximum, and average width and height of images as well as the amount of page area covered by them.
→ further categories as UE Parameter 1

**UE-Parameter 6: Page Formatting: (27)**
− Explanatory Definition: colour usage, fonts, use of interactive elements, how the page style is controlled, and other page characteristics. Key measures include evaluating the quality of colour combinations (for text and panels).
→ further categories as UE Parameter 1

**UE-Parameter 7: Page Performance: (37)**
− Explanatory Definition: page size, page download speed; accessibility of the page for people with disabilities; whether there are HTML errors on the page; and whether there is strong “scent” to the page. A model for predicting download speed that has 86% accuracy was developed; the model considers the number and size of HTML, graphic, script, and object (e.g., applet) files along with the number of tables on the page.
→ further categories as UE Parameter 1

**UE-Parameter 8: Site Architecture: (16)**
− Explanatory Definition: the consistency of page elements (i.e., text, link, and graphic elements), element formatting, page formatting and performance as well as the depth, breadth, and size of the site (i.e., the number of pages or documents). The site architecture measures only reflect the portion traversed by the crawler (i.e., the total number of pages crawled as well as the crawling breadth and depth).
→ further categories as UE Parameter 1

**UE-Principle/Parameter Relationship Matrix**
[not given]

**Computation Scheme**
1. For page level analysis:
   - Calculus1: Analyses across pages
   - Representational Elements 1(facts, rules or data structures): 157 measurements, the top ten predictors are minimum font size, minimum color use, italicized body word count, Weblint errors, graphic ad count, link text cluster count, interactive object count, Bobby priority 2 errors, text link count, and good link word count.
   - Result(s) incl. Rules for Interpretation 1: Classification of the web pages into three groups (good, average poor)
   - Calculus2: Analyses for context subgroups of pages
   - Algorithm / Computation 2: Discriminant Analyses for subgroups of good pages
   - Representational Elements 2(facts, rules or data structures): the 157 measurements, 157 measurements were extended with 6 content categories: community, education, finance, health, living, and services
   - Result(s) incl. Rules for Interpretation 2: The top 10 predictor variables varied across the contexts. 91% of the pages were identified correctly.
   - Calculus3: Analyses for pages within the subgroups
   - Representational Elements 3 (facts, rules or data structures): the 157 measurements, were extended with 6 content categories: community, education, finance, health, living, and services
   - Algorithm / Computation 3: Discriminant Analyses within context subgroups
   - Result(s) incl. Rules for Interpretation 3: The top 10 predictor variables varied across the contexts. 82% of the pages were identified correctly.

2. For site level analysis:
   - Calculus1: Consistency of pages in the site
   - Representational Elements 1(facts, rules or data structures): Variation of the 157 measurements,
   - Algorithm / Computation 1: Coefficients of Variation for all measures of at least 5 pages within a site.
   - Result(s) incl. Rules for Interpretation 1: Classification of the web sites into three groups (good, average poor)
   - Calculus2: Site structure
   - Representational Elements 2(facts, rules or data structures): accessable pages within a site through the web crawler tool
   - Algorithm / Computation 2: [not given]
   - Result(s) incl. Rules for Interpretation 2: [not given]
   - Calculus3: Analysis across sites (deepness and broadness)
   - Result(s) incl. Rules for Interpretation 3: sites only differed significantly on the maximum depth measure.
Best Practice of UEM Application

- Method Application sector – specific domains: [not given]
- Setting or Context of Use (users, location etc.): a computer with access to the
  internet is necessary.
- Method application to software lifecycle stage: testing, maintenance
- Prerequisites / Constraints (limitations of the method, e.g. resources; skills and
  knowledge of users and operator; apparatus required; software required;
  learnability of the method:
- Procedure (step 1 to n to follow when using of UEM):
  1. Download a representative set of pages from the particular site
  2. Use the analyses tool to compute site level and page level measures
  3. Analyse the calculated quality predictions
  4. Determine how the pages and sites are consistent which or derivate from
     good es.
  5. Interpret the model predictions to determine appropriate design changes
  6. Chance the pages and sites accordingly (manul)
  7. Go to step 1
- Documentation ( how is the process, how are the results documented): [the
  analysis tool prints out the results of the measures, the resulting improvements
  of the pages and sites have to be interpreted by the user]
- Particularities (is there anything to be considered when using the UEM):

Encountered usability problems when using UEM
UEM problems (problem description, reasons, suggestions for improvement) have to be
interpreted from the given measures.

Other encountered drawbacks when using UEM
[not given]

Summary of Advantages
Webtango can be used for analysis of sites and pages and assist the systematic
improvement of sites and pages

Estimated impact
The Webtango project seems to have finished since the last publication was 2002.

Open research questions
[Cross validation of the data mining algorithm. Correct classification of pages and sites
based on new data, which were not used for parameterization of the data mining
algorithm to evaluate the validity of the prediction.]

Other comments
Description of Case Studies
A small case study is published in:
Interface Design: Exploring Data Generated by the WebTango Tools, Deep Debroy. In
http://www.ocf.berkeley.edu/~berj/BERJ_sp02.pdf
Consolidation and Interpretation

The well structured form of the data gathered in the review process facilitates the consolidation and interpretation of the results. In a first step we tried to design an alternative approach of presenting the review data which we called “usability cards”. This graphical representation of the results helps to quickly grasp the essence of a method. In a next step we identified the clusters which are determined by the usability principles evaluated by the different UEMs. Finally we show some comparisons between UEMs evaluating the same principles.

Usability Cards

In order to depict the results of the single reviews we condensed them into so called “usability cards” using the following structure:

- **Name of the Computational Usability Evaluation Method (CUEM):** the name distinctly identifies the method
- **Addressed usability principle:** the name of the usability principle(s) addressed by the CUEM
- **Area of application:** this defines the scope of the method application
- **Object of Investigation:** which elements of the application are studied during evaluation
- **Verbal description:** a short verbal description on what is done
- **Calculation, Interpretation:** a short description of the calculation scheme and how its result are to be interpreted

![Figure 4.2: The "usability card" schema](image)

Clustering

According to the addressed usability principles we identified three concrete clusters of CUEMs, and one undefined/open cluster.

**Cluster 1: Controllability and self-descriptiveness**

The CUEMs "Bloodhound" and "Cognitive Walkthrough for the Web" address the usability principles **controllability** and **self-descriptiveness**.
Both CUEMs are concerned with the evaluation of the navigation of hypermedia systems. Based on the case that users have specific information seeking goals when they interact with a web site, the methods try to derive whether they will end up at the right path so that they find the relevant information of the web site in relation to the available information of the web site.

Therefore the Cognitive Walkthrough for the Web is using the Latent Semantic Analysis Method (LSA) Landauer, Dumais, (1997) to calculate different parameters.

The Bloodhound Project developed a new algorithm to derive the probability that users who are following the "information scent" of the links will find the right information according to their information goal.

Cluster 2: Suitability for the task/task conformance

The usability principle task conformance is addressed by the CUEMs "KLM", "WebRemUSINE" and "Evaluation of notification systems with the IRC Framework".

The concrete semantic behind the (from the reviewer interpreted) principle "self descriptiveness" varies from CUEM to CUEM:

Both KLM and WebRemUSINE measure the task completion times. WebRemUSINE offers a second dimension, namely the number of accomplished tasks in relation to task never tried and failed tasks. Presumably because of the different area of application (notification systems), the IRC framework measures the task conformance with the immediate response of the user to a new notification (reaction), the amount of reallocated attention from the primary task (interruption), and the situated awareness of the user (comprehension). Situated awareness depends on whether the user is able to identify the current state, the perception of the relevant elements and the possibility of the projection of future states.

In this cluster the CUEMs KLM and WebRemUSINE can be used for a wide application area, while the IRC Framework is only applicable to notification systems.

Whereas KLM and WebRemUSINE use task models of different complexity (under the assumption that the keystroke level model is a kind of a simple task model) to calculate their parameters. WebRemUSINE uses a (from experts) predefined task model and the usage logs to compute the relevant parameters for the usability principle "suitability for the task", in the IRC framework some input parameters must be set by experts before the interruption, reaction and comprehension values can be calculated.
Cluster 3: Suitability for learning

Figure 4.3: Usability cards concerning suitability for the task/task conformance

We identified only the CUEM WebRemUsine to address the usability principle *suitability for learning*. WebRemUsine is able to compute the user errors over the task completion time which makes
it possible to make statements about the learnability of the investigated system. The computation of the user errors is based on the predefined task model and the user logs.

**Cluster 4: Other**

The following tables show measurements which we were not able to relate to a dedicated usability principle. In case of DESTINE the reason is, that in this open CUEM the addressed principle is related to the chosen guidelines. Because the evaluator is free to choose the guidelines (as long as he/she is able to express the guideline in GDL) it is not possible to cluster DESTINE in respect of the usability principles.

![DESTINE Usability Card](image1)

**Figure 4.5: Usability card for DESTINE**

The situation is a little bit different with Webtango. Due this CUEM calculates very general parameters as the "overall page quality" and provides support for increasing the overall page quality based on information about 10 consistent and inconsistent parameters (out of a parameter set of 157 parameters) it was not possible to establish a relation to a certain usability principle.

![Webtango Usability Cards](image2)

**Figure 4.6: Usability cards for Webtango**

**Cross Checks**

The cluster analysis of the results of the review showed that some usability principles are addressed by more than one UEM. Closer examination of the methods and algorithms implemented show that there are differences in the way the principles are evaluated.

The next pages show some examples how these methods can be compared on the basis of the data gathered by the reviews.

**Bloodhound vs. ProcessLens**

The assignment of values (cf. review of Bloodhound) to properties of user behaviour or technical items seems not to be reflected very well. It might stem from empirical studies that are not detailed in the scanned data. The mapping of quantitative data to qualitative statements with respect to usability seems to need further research or at least detailing empirical work using a technique.

Although being crucial for transparent interpretation of computed results, this mapping or value transformation might not be a straightforward task. For instance, the percentages of **Bloodhound**
approach might need methodologically sound investigations, whereas ProcessLens delivers quantitative data that can be interpreted in a straightforward way.

The relationship to usability-engineering principles requires interpretation, since Bloodhound does not address standardised usability principles, such as task conformance directly. It requires a semantic content analysis, whether ProcessLens addresses navigability at all. The same holds for the dimension (organization of work, technology, cognitive systems), since no references are provided to traditional explanatory definitions. Finally, the parameter used for calculation has a limited scope, mostly the technique using them, e.g., information scent in case of Bloodhound.

The specification of context as well as the scope depends on one hand on the contextual specification of each technique, and on the other hand on the derived information from the review. A typical example for contextual specification is the assignment of text to the processing scheme as done in the course of the Bloodhound review – see section 4, first paragraph of each algorithm. In addition, the scope of the technique becomes transparent – Bloodhound: Users are assumed to have information goals. When users discover information items (according to their goals) their needs have been satisfied. The rate in which users are satisfied is considered as a measurement of the navigability of a web site.

The semantic encoding(s) can be considered at the overall level, as well as for cross-checking selected techniques, since it helps to reveal different meanings for usability-engineering parameters or -principles. For instance, both, Bloodhound and ProcessLens refer to navigability. In Bloodhound the context of using navigation is not considered. It may be used for task-specific navigation or hypermedia-links in general, whereas the framework of ProcessLens requires task- and domain context for navigation.

With respect to accuracy both, Bloodhound and ProcessLens are well-targeted automated usability-evaluation techniques. Although both of them do not provide an explicit understanding of the parameters and the algorithms, the processing schemes use minimal input to produce the desired results. The output can be interpreted by the users of each technique, however, might require additional information in the case of Bloodhound.

The specificity of the results differs according the scope of the technique. In case of Bloodhound the semantic utility of the found pages is not addressed at all, whereas ProcessLens implicitly refers to the task context of user activities. The latter might be represented in the navigation structure of the web site. However, it is not addressed by the Bloodhound goals, target documents, or navigation structure explicitly.

Finally, consistency can be addressed at the individual and group level: Each of the compared techniques can be considered to be consistent per se, since the terms, concepts, and parameters are used in the same way throughout the application of the technique. ProcessLens is not only consistent with respect to concepts and terms, but also with respect to the processing scheme: The same logic is used for checking different usability principles, namely suitability for the task, and suitability for individualization.

The consistency in mutual relation or with respect to contextual items differs, as already indicated in section 1. For instance, ProcessLens is only partially consistent to Bloodhound, since navigability is bound to task structures, whereas the Bloodhound algorithms operate on navigation patterns without having to specify their context of use or design. This fact is reflected through the assumption in Bloodhound that users have information needs and search goals that can be met. This understanding might be considered as some abstract task context.

The inconsistency with respect to navigability is also reflected by the review results. According to the template, the reviewers of ProcessLens consider navigation in the context of task conformance (the explicit relationship has been provided by the authors), whereas the reviewers of Bloodhound frame navigability by controllability, since it addresses one aspect of user control.
**KLM vs. WebRemUSINE**

**KLM** does not refer to a usability principle explicitly but the output parameter (*task completion time*) can be used as an indicator for *suitability for the task*. It is assumed that a lower task completion time is indicating a better suitability for the task. But the values generated can only be evaluated in references to other results and not as absolute numbers. As it is measuring execution times the method has to be applied to sequential action flows. **WebRemUSINE** is related to the principles *suitability for learning*, *suitability for the task* as well as *controllability*. Therefore it interprets the output parameters among which are the *number of the accomplished and failed tasks*. This output parameters can be interpreted as indicators for the suitability for the task as a high number of accomplished task and a low number of failed tasks let assume that the use of the system is efficient. But they can also work as indicators for learnability as a system suitable for learning will probable lead to a good ratio between accomplished and failed tasks. It is targeting web presences and facilitates remote usability evaluation.

The objective of **KLM** is to create values which can be used to compare different system designs with specific task scenarios. The user tasks have to be modelled in advance with **KLM** standard operators in order to assure comparability of the different systems.

Both methods need some implementation of the application (a completed application or a prototype). They extract the data necessary for analysis from existing artefacts. Hence these methods are suitable for usability evaluation during the development and the maintenance of applications. **KLM** is used for static testing (tests of the source code). A **KLM** model is created and used for calculation. Each action like keystrokes, typing or mouse pointing in the model is assigned with a certain execution time. All this execution times are then added up in order to calculate the task completion time. Also the “mental act of routine thinking or perception” is valued with an estimated time.

**WebRemUSINE** on the other hand needs user tests. Therefore this method is able to combine the advantages of traditional user testing and automated evaluation. During the test all user actions are logged without human intervention. These logs are then are then processed in the automatically evaluation. The values in the logs are compared to the task models of the application. The task model has to be compiled by an expert in advance with a special editor, the ConCurTask editor. Both methods are using task models, but the task models are entirely different. The KLM task model is very detailed documenting the task at the level of single keystrokes. WebRemUSINE uses a hierarchical task tree which focuses rather on what to do than how to do it.

Although **KLM** and **WebRemUSINE** are both appropriate for the evaluation of the suitability for the task their output parameters are not easily comparable. Whereas **KLM** focuses on task completion time as an indicator for suitability for the task **WebRemUSINE** uses this parameter among several others. **KLM** can be used for quantitative comparison of different system designs which pursue similar goals. The results of **WebRemUSINE** need expert interpretation, but this interpretation can deliver findings on its own without the need of comparison to other systems.

**WebRemUSINE vs. IRC Framework**

The **IRC** framework is a method which was originally designed for the evaluation of notification systems but it is possible to adapt the method in order to use it with other kinds of interactive systems. Even so most of the descriptions are targeting notifications systems explicitly. Further investigation concerning the application of this method in other application areas is advisable. The adoption and development of new critical parameters is necessary.

**IRC** does not explicitly address usability principles but the usability parameters used for calculation do allow inferences on the *controllability*, the *explainability* as well as the *suitability for the task*. Because of the original scope of the method (notification systems) one has to take into account, that the focus of the results is connected with the original area of application.

In contrast to **WebRemUSINE** which needs a task model and usage data in order to calculate the results **IRC** needs analysts or experts in order to perform a mediated evaluation of the system. As the
later does not rely on user testing it is possible to start the evaluation with the IRC framework during the design phase.

For the interpretation the result of IRC are compared with target values. This is different to WebRemUSINE which needs an interpretation of the results by experts.

**Integration of Results of other Working Groups**

**Integrating Working Group 1 Results**

The aim of Working Group 1 was to build a refined, substantiated and consolidated knowledge-pool about usability evaluation, based on the expertise, experiences, and research works of the participating project partners. They built a categorization scheme to facilitate the gathering of the relevant data.

The categorization scheme of WG1 shows certain similarities to the review template designed for the review process in WG 4. Therefore the review results can be merged within the knowledge pool created by WG 1.

The following Table 4.1 shows how the two schemes correspond to each other.

**Table 4.1: Integration of WG4 and WG1 categorisation schemes**

<table>
<thead>
<tr>
<th>WG4 Categorization Scheme v4</th>
<th>WG1 Categorization Scheme</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Method Identification</td>
<td>Origin + Theoretical Background / Reference Framework</td>
<td></td>
</tr>
<tr>
<td>1.1 UEM name</td>
<td>Name of the Method, + Author(s), date title &amp; full bibliographical reference(s)</td>
<td>WG4 UEM Name includes also the bibliographical references</td>
</tr>
<tr>
<td>1.2 Abstract</td>
<td>Abstract</td>
<td></td>
</tr>
<tr>
<td>1.3 UEM type</td>
<td>Category of the method</td>
<td>in WG4 UEM type is due to the focus always computational UEM, this point has been added to version 5 of the template</td>
</tr>
<tr>
<td>1.4 UEM history</td>
<td>History</td>
<td>in WG4 here can also be found related work</td>
</tr>
<tr>
<td>2. Method Characteristics</td>
<td>Method Characteristics</td>
<td></td>
</tr>
<tr>
<td>2.1 Objectives</td>
<td>Function / Goal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Relationship with Software Engineering</td>
<td></td>
</tr>
</tbody>
</table>
| 2.2 Adressed usability principles | - Name  
  - Explanatory definition  
  - Dimension / target / focus  
  - Addressed list of parameters  
  - Relationship to other principles | given for each principle |
<table>
<thead>
<tr>
<th>WG4 Categorization Scheme v4</th>
<th>WG1 Categorization Scheme</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3 UE-Input/Output-Parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Explanatory definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Relationship to other parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Formal encoding (incl. type)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Elicitation / Acquisition method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Mapping to computation scheme</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Involvement in computation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>given for each parameter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.4 UE-Principle/ Parameter Relationships</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5 Computation Scheme</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Calculus</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Representational elements</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Algorithm / Computation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Result(s) incl. rules for interpretation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.6 Best Practice of UEM Application</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Method application scope</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Method application sector</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Method application to lifecycle stage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Prerequisites, Constraints</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Procedures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Method Application Sector; Method Application to Software Lifecycle Stage; Prerequisites, Constraints had been added to version 5 of the template.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.7 Encountered usability problems when using UEM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Problem description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Reasons</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Suggestions for improvement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.8 Other encountered problems when using UEM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Summary Drawbacks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.9 Advantages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Summary Advantages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.10 Estimated Impact</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Industry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Research community</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estimated Impact</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Industry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Academia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Extensibility</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.11 Open research questions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Research questions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.12 Other comments</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Means of Assessment</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Integrating Working Group 2 Results

Working Group 2 has identified empirical usability constructs relevant for usability evaluation. These constructs complement and enhance the standard set of usability principles given in ISO 9241. This new set of constructs relevant for usability evaluation is a valuable input for future work. The constructs could help to build a wider set of usability principles CUEMs might address.

Integrating Working Group 3 Results

Working Group 3 focused on quality attributes which are important for valid classifications in usability evaluations. Using repertory grid interviews with HCI experts, designers and developers, WG3 have tried to link the classification of problems to quality targets. The interviews helped to identify ambiguities within the terminology used in usability evaluation as well as domain-specific differences. These findings could help to improve the strategic guide and the review template of WG4.

Conclusion

In order to deepen the understanding of inherent strengths and weaknesses of individual usability evaluation methods Working Group 4 developed a structured approach to review and analyze automated usability evaluation methods. A systematic review of existing and well documented methods forms a reliable basis for a profound analysis of the methods as well as for a selection mechanism for the practical use of the methods.

To ensure the quality of the review six strategic principles have been developed in order to guide the review process. Each review was conducted according to a review template in order to ensure that the results can be compared to each other.

As the level of detail varied significantly, not all methods currently known could be included in the review. During the main review phase eight usability methods have been identified and reviewed.

The most important aspects in this review were the usability principles addressed by the specific methods. The analysis of the conducted reviews shows that by far not all principles are covered by automated usability evaluation methods. The findings indicate that there are three clusters of methods, with more than fifty percent of the ISO usability principles not addressed.
However, additional concepts related to usability evaluation have been discovered during the work of working groups 2 and 3. Future reviews should try and include the usability constructs of WG2 and the quality attributes of WG3 in order to improve the applicability of the review technique, as they deal with different aspects of usability, taking into account the individual perspectives of stakeholders.

In addition usability principles targeted are often not explicitly mentioned in literature dealing with the respective methods. In order to identify the appropriate principles a semantic context analysis is necessary. Considering this approach we have to be carefully when interpreting the results as we do not know exactly whether the methods deal appropriately with the usability principle.

Cross checks of methods within a cluster indicate that the evaluation procedures targeting the same usability principle differ considerably. Therefore the results compiled through these procedures are difficult to compare.

Moreover, the different scopes of application within the product development lifecycle are not thoroughly covered. If usability evaluation is to be implemented along a products life cycle, evaluation methods should cover all steps of product development.

Overall we can see that there is still potential to improve the maturity of automated usability evaluation methods. In order to meet the expectations of different groups of HCI experts and practitioners the well known usability principles as well as the usability constructs and attributes identified during the MAUSE project need to be addressed through usability evaluation methods.

The systematic review procedure is a reliable approach to observe and study future developments in automated usability evaluation, since it allows tacking the progress in this field in a structured way. It could also serve as a knowledge base for HCI experts and practitioners interested in these methods. Finally, it helps to search and compare UEMs suitable for the evaluation task at hand.
Appendix

Additional Reviews

Unfortunately not all reviews could be included to the consolidation and interpretation as they were conducted after the main review phase. Nonetheless they contain valuable data which we want to publish here.

AutoCardSorter - Automated Card Sorting Tool

Reviewer: Christos Katsanos

<table>
<thead>
<tr>
<th>Method Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UEM Name</strong></td>
</tr>
<tr>
<td>AutoCardSorter - Automated Card Sorting Tool.</td>
</tr>
<tr>
<td><strong>Abstract</strong></td>
</tr>
<tr>
<td>Structuring of the content is an important step in web site design, affecting greatly navigability and the overall user experience. Automated support of this task is the object of this proposal. AutoCardSorter, a computational tool that supports the design and evaluation of the information architecture of a Web site, is introduced. The tool combines semantic similarity measures (e.g. Latent Semantic Analysis), hierarchical clustering algorithms and mathematical heuristics (e.g. eigenvalue-one criterion) in order to suggest suitable information navigation schemes in an automated manner. This is achieved by creating a matrix of semantic similarities among [text] descriptions of pages using LSA as a similarity metric. Subsequently, clustering algorithms are used to construct the information space. The output is presented as an interactive dendrogram. Furthermore, AutoCardSorter implements additional automated analyses to support the designer towards creating a suitable navigation scheme: (a) two complementary ways to determine the optimal number of categories in terms of variance explained and (b) a method to identify associative links in the adopted classification. Three independent studies were conducted in order to investigate the validity and efficiency of the proposed tool-based methodology. The studies compared the proposed method against the established card-sorting approach in different domains. It was found that substantial gain in efficiency was achieved without expense in the quality of results. Authors argue that such an approach could facilitate information-rich applications design, like most Web sites, by reducing time and resources required.</td>
</tr>
<tr>
<td><strong>UEM – Type</strong></td>
</tr>
<tr>
<td>Analytical UEM.</td>
</tr>
<tr>
<td><strong>UEM- History</strong></td>
</tr>
<tr>
<td>Inspired by the Card Sorting technique.</td>
</tr>
</tbody>
</table>
Method Characteristics

Objectives
- Automate the design/evaluation of the information architecture of websites. The method addresses the problem of content structuring (structural navigation) and helps in creating semantic relationships between related pieces of content across levels of a hierarchy (associative navigation).
- Aims at providing the necessary flexibility and efficiency to usability practitioners and web site designers and can be used for both the initial design and redesign of web sites.

Usability Principle(s) Addressed by the UEM

UE-Principle 1
- Name: [Suitability for the task].
- Explanatory definition (incl. ref. to standard): [Support efficient and effective information seeking tasks on the web]. (“A dialogue is suitable for a task when it supports the user in the effective and efficient completion of the task” - ISO 9241 Part 10).
- Dimension/target/focus (organisation of work/technology/cognitive system): [technology, cognitive system].
- Addressed (list of) parameters: [Semantic similarity-matrix; Interactive dendrogram].
- Relationships to other Principles (give both, type of relationship and principle): [Effectiveness, efficiency, self-descriptiveness].

UE-Principle 2
- Name: [Self-descriptiveness].
- Explanatory definition (incl. ref. to standard): [Support intuitive access and findability of information in web sites]. (“A dialogue is self-descriptive when each dialogue step is immediately comprehensible through feedback from the system or is explained to the user on request” - ISO 9241 Part 10).
- Dimension/target/focus (organisation of work/technology/cognitive system): [technology, cognitive system].
- Addressed (list of) parameters: [Semantic similarity-matrix; Interactive dendrogram].
- Relationships to other Principles (give both, type of relationship and principle): [Effectiveness, suitability for the task].

UE-Parameters:

UE-Input Parameter 1: Content items descriptions
- Explanatory Definition: Text descriptions of the content items to cluster (e.g. pages).
- Relationship to other Parameters (give both, type of relationship & parameter): Required input for both semantic similarity-matrix and interactive dendrogram production.
- Formal Encoding (incl. type): Free text [length of text is not specified but it is not specified either in the traditional card sorting technique].
- Elicitation/Acquisition Method when measured: Compiled by the analyst.
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Input to calculate the semantic similarity between the content items.
- Involvement in Computation (input / output to, side effect of calculation): Input.
UE-Input Parameter 2: Semantic spaces
- Explanatory Definition: Semantic spaces represent the level of reading comprehension ability and background knowledge of specific user groups.
- Relationship to other Parameters (give both, type of relationship & parameter): Required input for both semantic similarity-matrix and interactive dendrogram production.
- Elicitation/Acquisition Method when measured: Provided by lsa.colorado.edu [what if none of the available is representative of the typical users?, how easy and practical it is to create mine?].
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Input to calculate the semantic similarity between the content items.
- Involvement in Computation (input / output to, side effect of calculation): Input.

UE-Output Parameter 1: Semantic similarity-matrix
- Explanatory Definition: Symmetric NxN matrix of the semantic similarities among descriptions of all pairs of content items using a similarity metric (e.g. LSA).
- Relationship to other Parameters (give both, type of relationship & parameter): The values of the matrix’s cells depend on the content item descriptions and the selected semantic space.
- Formal Encoding (incl. type): Numeric (cosine).
- Elicitation/Acquisition Method when measured: Automatically produced.
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Each cell is the result of the LSA algorithm (cosine value) for each pair of content descriptions given the selected semantic space.
- Involvement in Computation (input / output to, side effect of calculation): Output.

UE-Output Parameter 2: Interactive dendrogram
- Explanatory Definition: Interactive tree diagram presenting the recommended clustering of pages.
- Relationship to other Parameters (give both, type of relationship & parameter): Graphical result of cluster analysis on a transformation of the semantic similarity-matrix (called semantic dissimilarity-matrix).
- Formal Encoding (incl. type): Graphical (nodes, lines); [Nodes include semantic dissimilarity values where groupings take place (numeric)].
- Elicitation/Acquisition Method when measured: Automatically produced; analyst can differentiate the number of the desired groups in a visual way or by specifying explicitly the desired number. In both cases, the tool reorganizes the results, showing the most effective item-clustering in real time.
- Mapping to Computation Scheme (which element of the calculation scheme corresponds in which way to the parameter, or, how is the parameter used as element of a calculation scheme): Output; structural navigation of the website; result of cluster analysis on a transformation of the semantic similarity-matrix (called semantic dissimilarity-matrix).
Involvement in Computation (input / output to, side effect of calculation): Output.

**UE-Principle/Parameter Relationship Matrix**

<table>
<thead>
<tr>
<th></th>
<th>Suitability for the task</th>
<th>Self-descriptiveness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semantic similarity-matrix</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Interactive dendrogram</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

**Computation Scheme**
- Calculus: [Not named].
- Representational Elements (facts, rules or data structures): Matrix, vector, diagram.
- Algorithm / Computation:
  [Background required in order to understand the algorithm: Latent Semantic Analysis]
  First, LSA parses suitable, large text corpora that represent a given user population’s understanding of words and produces a term-document matrix of each word’s frequency of occurrence. Next, each cell is weighted by a function that expresses both the word’s importance in the particular document and the degree to which the word type carries information in the domain of discourse in general. Alternatively, the term-document matrix can be viewed as a huge, multidimensional space where each term is considered as a separate dimension and each document is a vector in this term-space. Subsequently, LSA applies Singular Value Decomposition (SVD), the mathematical generalization of factor analysis, to project this large, multidimensional space down into the least-squares best fit number of dimensions. This is a key step in the LSA method since the SVD algorithm reveals similarities that are latent in the document collection. LSA is, actually, exploiting the property of natural language that words with similar meaning tend to occur together. In this way, the initial term-space is transformed into a semantic space, where the degree of semantic similarity between any pair of texts, such as the descriptions of two web pages, is measured by the cosine of the corresponding two vectors. Each cosine value lies between +1 (identical) and -1 (opposite). Near-zero values represent unrelated texts.

For more information on LSA the article references to:

**AutoCardSorter Algorithm:**
1. Begin with the text descriptions of the N web pages.
2. Create a similarity matrix S using a semantic similarity measure (e.g. LSA) to calculate the semantic similarity for each pair (i), (j) of pages.
   - $S \begin{bmatrix} (i), (j) \end{bmatrix} = \text{LSA} (i, j)$, where S is a symmetric matrix of NxN dimensions.
3. Convert the similarity matrix S to a normalized dissimilarity (or distance) matrix D:
   - $D \begin{bmatrix} (i), (j) \end{bmatrix} = d \{ S \begin{bmatrix} (i), (j) \end{bmatrix} \}$, with d defined as:
     - $d (i, j) = 50 (1 - \text{LSA} (i, j))$, where $d (i, j) \in [0,100]
4. Consider each web page as a separate cluster.
5. Find the less dissimilar pair of clusters in the current clustering step, say pair \((r), (s)\), according to:
   - \(D \[(r),(s)\] = \{ D \[(i),(j)\] \}
6. Merge clusters \((r)\) and \((s)\) into a single cluster.
7. Update the distance matrix, \(D\), by deleting the rows and columns corresponding to clusters \((r)\) and \((s)\) and adding a row and column corresponding to the newly formed cluster. The distance \(d\) between the new cluster, denoted \((r,s)\) and old cluster \((k)\) is defined based on the selected type of hierarchical clustering algorithm:
   - If Single-Linkage selected then \(d((k),(r,s)) = \min \{ D((k),(r)), D((k),(s)) \}\).
   - If Complete-Linkage selected then \(d((k),(r,s)) = \max \{ D((k),(r)), D((k),(s)) \}\).
   - If Average-Linkage selected then \(d((k),(r,s)) = \frac{D((k),(r)) + D((k),(s))}{2}\).
8. If all web pages are in one cluster, stop. Else, go to step 5.

- Result(s) incl. Rules for Interpretation: Tool provides an interactive tree dendrogram presenting the recommended clustering of pages and the analyst decides on the final groupings. Additional support is provided to the analyst to identify the optimal number of top-level categories in terms of total variance explained and associative links in the adopted classification.

**Best Practice of UEM Application**

- Method Application Sector: Web sites [but can be extended to any type of application that requires information structuring e.g. menus].
- Setting or Context of Use (users, location etc.): Typical users’ profile [and a rough idea of the content of the pages needs to be available]; requires computer with access to internet and LSA; targets two main groups of users: (a) usability practitioners and (b) website designers.
- Method Application to Software Lifecycle: Design and evaluation.
- Procedure (step 1 to n to follow when using of UEM):
  1. Provide a set of text descriptions of the pages that the web site will contain [assumes that the analyst has a rough idea or draft version of the pages’ content].
  2. Select an appropriate LSA semantic space to represent typical users of the site [assumes that typical user profiles are already available].
  3. Select hierarchical clustering algorithm type. Available types are: single, complete or average linkage.
  4. Optional: Provide a set of text labels of sections to be created (simulates a closed-card sorting).
  5. Run automated analysis.
  6. Output: interactive tree diagram (known as dendrogram), presenting the recommended clustering of pages. Analyst can differentiate the number of the desired groups in a visual way or by specifying explicitly the desired number. In both cases, the tool reorganizes the results, showing the most effective item-clustering in real time.
  7. Additional automated analyses: (a) two complementary ways to determine the optimal number of categories in terms of variance explained and (b) a method to identify associative links in the adopted classification.
8. Analyst decides on the final navigation scheme to adopt.
   - Documentation (how is the process, how are the results documented): Matrix with the semantic similarities of all pairs of pages is automatically produced; interactive dendrogram of site structure is automatically produced; additional automated analyses to identify number of categories and associative links are provided; [results on-screen only].
   - Particularities (is there anything to be considered when using the UEM): selection of a representative semantic space of the typical users; [how to construct a semantic space if the available are not appropriate]; production of page descriptions enriched with the contextual information that they are referring to (e.g. “replace pronouns with their implicitly related nouns or noun-phrases”).

Encountered usability problems when using UEM
[Not given]
Other encountered drawbacks when using UEM
[Not given]

Summary of Advantages
- Reduces resources required through analytical elicitation of information architecture.
- Accelerates the design and evaluation lifecycle by providing the necessary efficiency and flexibility.
- Remains an efficient solution when designing or evaluating large sites (>100 pages) where traditional techniques (e.g. card-sorting) are hard to apply.
- Increases the possibility to explore alternative designs and, therefore, can lead to better solutions for web sites’ structures.
- [Seems intuitive and simple to learn and apply, therefore, increasing the chance for wider adoption.]

[The validity and efficiency of the approach has been demonstrated in three studies comparing the approach to the traditional card sorting. However, no practitioners-based empirical evidence about the usability/perceived usefulness of the tool is yet available].

Estimated impact
[It seems a useful tool and may have some significant impact in both research and development. However, the method is quite young (2008) and is still under development]

Open research questions
- How to help the analyst choose labels that for the produced categories. Authors plan to investigate ways to address this issue, such as automatically identifying the most frequent words in the categories created and providing a set of ‘near-neighbor’ words (i.e. semantically close) from which the designer could choose to form a valid label
- Issues related to semantic similarity algorithms: (a) poor performance of LSA when the words are relatively rare in the corpora it has been trained on, (b) [how to construct domain-specific semantic spaces]. (c) [outdated semantic spaces]. As a possible solution the investigation of dynamic approaches for training of corpora (i.e. web search results) to induce semantic similarities between terms/passages is proposed (e.g. LSA-IR).
- Explore alternatives to hierarchical cluster analysis (e.g. factor analysis, multidimensional scaling).
- [Practitioners’ acceptance towards the tool].
Other comments
[An interesting applied research]

Description of Case Studies

Case study 1
- Abstract: Presents a study that investigates the validity and efficiency of the proposed tool-based methodology against the widely-used card-sorting method.
- Method: The proposed tool-based approach was used to design the information architecture of a web site in the health domain and the results were compared with those of an open card-sorting study. Validity of the approach was investigated by two types of analyses: (a) similarity-matrices correlation analysis and (b) elbow-based navigation schemes comparison. Efficiency of the approach was measured as the ratio of the total time required for the card-sorting experiment to the total time required to use the presented approach.
- Participants: 18 participants (10 male, 8 female, aged 22-55 with a mean of 28). Participants shared a high level of education and reported high internet experience
- Setting: Open card-sorting sessions for each participant. Traditional index cards were used.
- Results: The study demonstrated substantial efficiency gain in favor of the automated approach – proved approximately 27 times faster - without expense in the quality of results (similarity-matrices r=0.80, p<0.01; identical elbow-based navigation schemes).

Case study 2
- Abstract: Two more replication studies demonstrate the validity and efficiency of the proposed approach by comparing it to the card-sorting technique in the design or redesign of the information architecture of web sites for various domains (health, education, tourism) and sizes.
- Method: Same as the study 1. Only difference is that an additional type of validation (base-clusters comparison) was added to the initial two.
- Participants: 26 (17 male, 9 female, aged 22-25 with a mean of 23) and 38 (25 male, 9 female, aged 21-26 with a mean of 22) participants respectively. Participants shared a high level of education and reported high internet experience.
- Setting: Open card-sorting sessions for each participant. A card-sorting tool, USort (Dong et al., 2001), was used to facilitate the card-sorting exercises.
- Results: AutoCardSorter provided semantically similar groupings and overall information structures to the ones derived by representative users involved in the card-sorting sessions. AutoCardSorter was found to be more efficient than the established card-sorting technique, as it was 11 to 27 times faster.
European Science Foundation provides and manages the scientific and technical secretariat for COST

COST is supported by the EU RTD Framework Programme