Karin Coninx
Kris Luyten
Kevin A. Schneider (Eds.)

LNCS 4385

Task Models and Diagrams for Users Interface Design

5th International Workshop, TAMODIA 2006
Hasselt, Belgium, October 2006
Revised Papers

Springer

Karin Coninx Kris Luyten
Kevin A. Schneider (Eds.)

Task Models and Diagrams for Users Interface Design

5th International Workshop, TAMODIA 2006
Hasselt, Belgium, October 23-24, 2006
Revised Papers

Springer

Volume Editors

Karin Coninx
Hasselt University
Wetenschapspark 2
3590 Diepenbeek, Belgium
E-mail: karin.coninx@uhasselt.be

Kris Luyten
Hasselt University
Wetenschapspark 2
3590 Diepenbeek, Belgium
E-mail: kris.luyten@uhasselt.be

Kevin A. Schneider
University of Saskatchewan
Saskatoon SK S7N 5C9, Canada
E-mail: kas@cs.usask.ca

Library of Congress Control Number: 2007920036

CR Subject Classification (1998): H.5.2, H.5, D.2, D.3, F.3, I.6, K.6

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743
ISBN-10 3-540-70815-4 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-70815-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12018130 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 4385

Preface

We are proud to present the TAMODIA 2006 proceedings. In 2006, the TAMODIA workshop celebrated its fifth anniversary. TAMODIA is an obscure acronym that stands for *TAsk MOdels and DIAgrams for user interface design*. The first edition of TAMODIA was organized in Bucharest (Romania) by Costin Pribeanu and Jean Vanderdonckt. The fact that five years later the TAMODIA series of workshops still continues successfully proves the importance of this research area for the human–computer interaction community! The first workshop aimed at examining how multiple forms of task expressions can significantly increase or decrease the quality of user interface design. This is still the scope of the current edition; we tried to assemble papers that discuss how the complexity of HCI design and development can be managed with tasks, models and diagrams. Much like the previous editions, the selection of papers from the 2006 edition reflects the broad scope of this field, which cannot be labeled with a single title or term.

The invited paper is by Joëlle Coutaz and discusses meta-user interfaces for ambient spaces. Finding appropriate ways to design and develop user interfaces for interactive spaces is becoming an important challenge for the creation of future usable applications. This exciting work gives a good feel of the new type of user interfaces and the required new approaches we are evolving toward when we want to realize the vision of ambient intelligent environments and create systems that can be used and controlled by the end-users.

From the papers included in these proceedings, we can clearly see both fields, HCI and software engineering, moving toward each other. Techniques from software engineering are becoming more popular and tuned for HCI design and development. Requirements engineering, model-driven engineering, model-based design and patterns are already familiar in both fields, but differ in notations and approaches. Established notations such as UML and Petrinets are helping us link models used in both fields and integrate approaches. In all papers, the end-user is never forgotten, and the user tasks and context that need to be supported play a central role.

October 2006

Kris Luyten
Belgium

Organization

Workshop Chair

Karin Coninx (Hasselt University, Belgium)

Program Co-chairs

Kris Luyten (Hasselt University, Belgium)
Kevin A. Schneider (University of Saskatchewan, Canada)

Local Organization

Ingrid Konings (Hasselt University, Belgium)
Frederik Winters (Hasselt University, Belgium)
Jo Vermeulen (Hasselt University, Belgium)

Review Committee

Alan Dix, Lancaster University, UK
Peter Forbrig, University of Rostock, Germany
Fabio Paternò, ISTI-CNR, Italy
Philippe Palanque, Université Toulouse 3 (Paul Sabatier), France
Gerrit C. Van der Veer, Vrije Universiteit Amsterdam, The Netherlands
Rémi Bastide, Université Toulouse 1, France
Gaëlle Calvary, University of Grenoble I, France
Quentin Limbourg, Catholic University of Louvain, Belgium
Nuno Jardim Nunes, University of Madeira, Portugal
Costin Pribeanu, ICI Bucuresti, Romania
Matthias Rauterberg, Eindhoven University of Technology, The Netherlands
Anke Dittmar, University of Rostock, Germany
Dave Roberts, Usability Competency Centre, IBM, UK
Ahmed Seffah, Concordia University, Canada
Pavel Slavík, Czech Technical University in Prague, Czech Republic
Karin Coninx, Hasselt University, Belgium
Marco Antonio Alba Winckler, Université Paul Sabatier, France
Corina Sas, Lancaster University, UK
Christian Stary, University of Linz, Austria
Birgit Bomsdorf, University of Hagen, Germany
Carmen Santoro, ISTI-CNR, Italy
Maria-Dolores Lozano, Universidad de Castilla-La Mancha, Spain
Regina Bernhaupt, University of Salzburg, Austria
Erik G. Nilsson, Sinteff, Norway
Mieke Massink, ISTI-CNR, Italy

Kris Luyten, Hasselt University, Belgium
Victor M. López Jaquero, University of Castilla-La Mancha, Spain
Elizabeth Furtado, UNIFOR, Brasil
Peter J. Wild, University of Bath, UK
Zeljko Obrenovic, Centrum voor Wiskunde en Informatica, The Netherlands

Lecture Notes in Computer Science

For information about Vols. 1–4284

please contact your bookseller or Springer

Vol. 4385: K. Coninx, K. Luyten, K.A. Schneider (Eds.), Task Models and Diagrams for Users Interface Design. XI, 355 pages. 2007.

Vol. 4384: T. Washio, K. Satoh, H. Takeda, A. Inokuchi (Eds.), New Frontiers in Artificial Intelligence. IX, 401 pages. 2007. (Sublibrary LNAI).

Vol. 4381: J. Akiyama, W.Y.C. Chen, M. Kano, X. Li, Q. Yu (Eds.), Discrete Geometry, Combinatorics and Graph Theory. XI, 289 pages. 2007.

Vol. 4380: S. Spaccapietra, P. Atzeni, F. Fages, M.-S. Hacid, M. Kifer, J. Mylopoulos, B. Pernici, P. Shvaiko, J. Trujillo, I. Zaihrayeu (Eds.), Journal on Data Semantics VIII. XV, 219 pages. 2007.

Vol. 4377: M. Abe (Ed.), Topics in Cryptology – CT-RSA 2007. XI, 403 pages. 2006.

Vol. 4373: K. Langendoen, T. Voigt (Eds.), Wireless Sensor Networks. XIII, 358 pages. 2007.

Vol. 4371: K. Inoue, K. Satoh, F. Toni (Eds.), Computational Logic in Multi-Agent Systems. X, 315 pages. 2007. (Sublibrary LNAI).

Vol. 4369: M. Umeda, A. Wolf, O. Bartenstein, U. Geske, D. Seipel, O. Takata (Eds.), Declarative Programming for Knowledge Management. X, 229 pages. 2006. (Sublibrary LNAI).

Vol. 4368: T. Erlebach, C. Kaklamanis (Eds.), Approximation and Online Algorithms. X, 345 pages. 2007.

Vol. 4367: K. De Bosschere, D. Kaeli, P. Stenström, D. Whalley, T. Ungerer (Eds.), High Performance Embedded Architectures and Compilers. XI, 307 pages. 2007.

Vol. 4364: T. Kühne (Ed.), Models in Software Engineering. XI, 332 pages. 2007.

Vol. 4362: J. van Leeuwen, G.F. Italiano, W. van der Hoek, C. Meinel, H. Sack, F. Plášil (Eds.), SOFSEM 2007: Theory and Practice of Computer Science. XXI, 937 pages. 2007.

Vol. 4361: H.J. Hoogeboom, G. Păun, G. Rozenberg, A. Salomaa (Eds.), Membrane Computing. IX, 555 pages. 2006.

Vol. 4360: W. Dubitzky, A. Schuster, P.M.A. Sloot, M. Schroeder, M. Romberg (Eds.), Distributed, High-Performance and Grid Computing in Computational Biology. X, 192 pages. 2007. (Sublibrary LNBI).

Vol. 4357: L. Buttyán, V. Gligor, D. Westhoff (Eds.), Security and Privacy in Ad-Hoc and Sensor Networks. X, 193 pages. 2006.

Vol. 4355: J. Julliand, O. Kouchnarenko (Eds.), B 2007: Formal Specification and Development in B. XIII, 293 pages. 2006.

Vol. 4354: M. Hanus (Ed.), Practical Aspects of Declarative Languages. X, 335 pages. 2006.

Vol. 4353: T. Schwentick, D. Suciu (Eds.), Database Theory – ICDT 2007. XI, 419 pages. 2006.

Vol. 4352: T.-J. Cham, J. Cai, C. Dorai, D. Rajan, T.-S. Chua, L.-T. Chia (Eds.), Advances in Multimedia Modeling, Part II. XVIII, 743 pages. 2006.

Vol. 4351: T.-J. Cham, J. Cai, C. Dorai, D. Rajan, T.-S. Chua, L.-T. Chia (Eds.), Advances in Multimedia Modeling, Part I. XIX, 797 pages. 2006.

Vol. 4349: B. Cook, A. Podelski (Eds.), Verification, Model Checking, and Abstract Interpretation. XI, 395 pages. 2007.

Vol. 4348: S.T. Taft, R.A. Duff, R.L. Brukardt, E. Ploedereder, P. Leroy (Eds.), Ada 2005 Reference Manual. XXII, 765 pages. 2006.

Vol. 4347: J. Lopez (Ed.), Critical Information Infrastructures Security. X, 286 pages. 2006.

Vol. 4345: N. Maglaveras, I. Chouvarda, V. Koutkias, R. Brause (Eds.), Biological and Medical Data Analysis. XIII, 496 pages. 2006. (Sublibrary LNBI).

Vol. 4344: V. Gruhn, F. Oquendo (Eds.), Software Architecture. X, 245 pages. 2006.

Vol. 4342: H. de Swart, E. Orłowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments II. X, 373 pages. 2006. (Sublibrary LNAI).

Vol. 4341: P.Q. Nguyen (Ed.), Progress in Cryptology - VIETCRYPT 2006. XI, 385 pages. 2006.

Vol. 4340: R. Prodan, T. Fahringer, Grid Computing. XXIII, 317 pages. 2007.

Vol. 4339: E. Ayguadé, G. Baumgartner, J. Ramanujam, P. Sadayappan (Eds.), Languages and Compilers for Parallel Computing. XI, 476 pages. 2006.

Vol. 4338: P. Kalra, S. Peleg (Eds.), Computer Vision, Graphics and Image Processing. XV, 965 pages. 2006.

Vol. 4337: S. Arun-Kumar, N. Garg (Eds.), FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science. XIII, 430 pages. 2006.

Vol. 4335: S.A. Brueckner, S. Hassas, M. Jelasity, D. Yamins (Eds.), Engineering Self-Organising Systems. XII, 212 pages. 2007. (Sublibrary LNAI).

Vol. 4334: B. Beckert, R. Hähnle, P.H. Schmitt (Eds.), Verification of Object-Oriented Software. XXIX, 658 pages. 2007. (Sublibrary LNAI).

Vol. 4333: U. Reimer, D. Karagiannis (Eds.), Practical Aspects of Knowledge Management. XII, 338 pages. 2006. (Sublibrary LNAI).

Vol. 4332: A. Bagchi, V. Atluri (Eds.), Information Systems Security. XV, 382 pages. 2006.

Vol. 4331: G. Min, B. Di Martino, L.T. Yang, M. Guo, G. Ruenger (Eds.), Frontiers of High Performance Computing and Networking – ISPA 2006 Workshops. XXXVII, 1141 pages. 2006.

Vol. 4330: M. Guo, L.T. Yang, B. Di Martino, H.P. Zima, J. Dongarra, F. Tang (Eds.), Parallel and Distributed Processing and Applications. XVIII, 953 pages. 2006.

Vol. 4329: R. Barua, T. Lange (Eds.), Progress in Cryptology - INDOCRYPT 2006. X, 454 pages. 2006.

Vol. 4328: D. Penkler, M. Reitenspiess, F. Tam (Eds.), Service Availability. X, 289 pages. 2006.

Vol. 4327: M. Baldoni, U. Endriss (Eds.), Declarative Agent Languages and Technologies IV. VIII, 257 pages. 2006. (Sublibrary LNAI).

Vol. 4326: S. Göbel, R. Malkewitz, I. Iurgel (Eds.), Technologies for Interactive Digital Storytelling and Entertainment. X, 384 pages. 2006.

Vol. 4325: J. Cao, I. Stojmenovic, X. Jia, S.K. Das (Eds.), Mobile Ad-hoc and Sensor Networks. XIX, 887 pages. 2006.

Vol. 4323: G. Doherty, A. Blandford (Eds.), Interactive Systems. XI, 269 pages. 2007.

Vol. 4320: R. Gotzhein, R. Reed (Eds.), System Analysis and Modeling: Language Profiles. X, 229 pages. 2006.

Vol. 4319: L.-W. Chang, W.-N. Lie (Eds.), Advances in Image and Video Technology. XXVI, 1347 pages. 2006.

Vol. 4318: H. Lipmaa, M. Yung, D. Lin (Eds.), Information Security and Cryptology. XI, 305 pages. 2006.

Vol. 4317: S.K. Madria, K.T. Claypool, R. Kannan, P. Uppuluri, M.M. Gore (Eds.), Distributed Computing and Internet Technology. XIX, 466 pages. 2006.

Vol. 4316: M.M. Dalkilic, S. Kim, J. Yang (Eds.), Data Mining and Bioinformatics. VIII, 197 pages. 2006. (Sublibrary LNBI).

Vol. 4314: C. Freksa, M. Kohlhase, K. Schill (Eds.), KI 2006: Advances in Artificial Intelligence. XII, 458 pages. 2007. (Sublibrary LNAI).

Vol. 4313: T. Margaria, B. Steffen (Eds.), Leveraging Applications of Formal Methods. IX, 197 pages. 2006.

Vol. 4312: S. Sugimoto, J. Hunter, A. Rauber, A. Morishima (Eds.), Digital Libraries: Achievements, Challenges and Opportunities. XVIII, 571 pages. 2006.

Vol. 4311: K. Cho, P. Jacquet (Eds.), Technologies for Advanced Heterogeneous Networks II. XI, 253 pages. 2006.

Vol. 4309: P. Inverardi, M. Jazayeri (Eds.), Software Engineering Education in the Modern Age. VIII, 207 pages. 2006.

Vol. 4308: S. Chaudhuri, S.R. Das, H.S. Paul, S. Tirthapura (Eds.), Distributed Computing and Networking. XIX, 608 pages. 2006.

Vol. 4307: P. Ning, S. Qing, N. Li (Eds.), Information and Communications Security. XIV, 558 pages. 2006.

Vol. 4306: Y. Avrithis, Y. Kompatsiaris, S. Staab, N.E. O'Connor (Eds.), Semantic Multimedia. XII, 241 pages. 2006.

Vol. 4305: A.A. Shvartsman (Ed.), Principles of Distributed Systems. XIII, 441 pages. 2006.

Vol. 4304: A. Sattar, B.-H. Kang (Eds.), AI 2006: Advances in Artificial Intelligence. XXVII, 1303 pages. 2006. (Sublibrary LNAI).

Vol. 4303: A. Hoffmann, B.-H. Kang, D. Richards, S. Tsumoto (Eds.), Advances in Knowledge Acquisition and Management. XI, 259 pages. 2006. (Sublibrary LNAI).

Vol. 4302: J. Domingo-Ferrer, L. Franconi (Eds.), Privacy in Statistical Databases. XI, 383 pages. 2006.

Vol. 4301: D. Pointcheval, Y. Mu, K. Chen (Eds.), Cryptology and Network Security. XIII, 381 pages. 2006.

Vol. 4300: Y.Q. Shi (Ed.), Transactions on Data Hiding and Multimedia Security I. IX, 139 pages. 2006.

Vol. 4299: S. Renals, S. Bengio, J.G. Fiscus (Eds.), Machine Learning for Multimodal Interaction. XII, 470 pages. 2006.

Vol. 4297: Y. Robert, M. Parashar, R. Badrinath, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2006. XXIV, 642 pages. 2006.

Vol. 4296: M.S. Rhee, B. Lee (Eds.), Information Security and Cryptology – ICISC 2006. XIII, 358 pages. 2006.

Vol. 4295: J.D. Carswell, T. Tezuka (Eds.), Web and Wireless Geographical Information Systems. XI, 269 pages. 2006.

Vol. 4294: A. Dan, W. Lamersdorf (Eds.), Service-Oriented Computing – ICSOC 2006. XIX, 653 pages. 2006.

Vol. 4293: A. Gelbukh, C.A. Reyes-Garcia (Eds.), MICAI 2006: Advances in Artificial Intelligence. XXVIII, 1232 pages. 2006. (Sublibrary LNAI).

Vol. 4292: G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. Nefian, G. Meenakshisundaram, V. Pascucci, J. Zara, J. Molineros, H. Theisel, T. Malzbender (Eds.), Advances in Visual Computing, Part II. XXXII, 906 pages. 2006.

Vol. 4291: G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. Nefian, G. Meenakshisundaram, V. Pascucci, J. Zara, J. Molineros, H. Theisel, T. Malzbender (Eds.), Advances in Visual Computing, Part I. XXXI, 916 pages. 2006.

Vol. 4290: M. van Steen, M. Henning (Eds.), Middleware 2006. XIII, 425 pages. 2006.

Vol. 4289: M. Ackermann, B. Berendt, M. Grobelnik, A. Hotho, D. Mladenič, G. Semeraro, M. Spiliopoulou, G. Stumme, V. Svátek, M. van Someren (Eds.), Semantics, Web and Mining. X, 197 pages. 2006. (Sublibrary LNAI).

Vol. 4288: T. Asano (Ed.), Algorithms and Computation. XX, 766 pages. 2006.

Vol. 4287: C. Mao, T. Yokomori (Eds.), DNA Computing. XII, 440 pages. 2006.

Vol. 4286: P.G. Spirakis, M. Mavronicolas, S.C. Kontogiannis (Eds.), Internet and Network Economics. XI, 401 pages. 2006.

Vol. 4285: Y. Matsumoto, R.W. Sproat, K.-F. Wong, M. Zhang (Eds.), Computer Processing of Oriental Languages. XVII, 544 pages. 2006. (Sublibrary LNAI).

¥562.00元

Table of Contents

Invited Paper

Tool Support

Model-Based Interface Development

User Interface Patterns

Bridging the Gap: Driven by Models

Task-Centered Design

Multi-modal User Interfaces

Reflections on Tasks and Activities in Modeling

Context and Plasticity

Meta-User Interfaces for Ambient Spaces

Joëlle Coutaz

Université Joseph Fourier, LIG,
385 rue de la Bibliothèque, BP 53, 38041 Grenoble Cedex 9, France
joelle.Coutaz@imag.fr

Abstract. In this article, we propose the concept of meta-User Interface (meta-UI) as the set of functions (along with their user interfaces) that are necessary and sufficient to control and evaluate the state of interactive ambient spaces. This set is *meta-*, since it serves as an umbrella beyond the domain-dependent services that support human activities in an ambient interactive space. They are *User Interface*-oriented since their role is to help users to control and evaluate the state of this space. We present a dimension space to classify, compare, and contrast disparate research efforts in the area of meta-UI's. We then exploit the generative power of our design space to suggest directions for future research.

Keywords: Ubiquitous computing, ambient interactive spaces, design space, taxonomy, meta-UI, GUI desktop.

1 Introduction

The capacity for users to control and evaluate system state is fundamental to Computer-Human Interaction [33]. This principle, promoted in the early 1980's by cognitive psychologists and human factors specialists, has actually been applied twenty years earlier by computer scientists who introduced the concept of Job Control Language (JCL). JCL, used for batch processing, has been progressively replaced with the Unix Shell followed by graphical desktops. With the emergence of ambient computing, users are not limited to the system and applications of a single computer. Instead, ambient computing embraces a model in which users, services, and resources discover other users, services and resources, and integrate them into an ambient interactive space.

An *ambient interactive space* is a dynamic assembly of physical entities coupled with computational and communicational entities to support human activities. An ambient interactive space can be as simple as a workstation or a PDA connected to the services of the Internet, or as complex as a computational ecosystem that evolves to adapt to the context of use. Augmented rooms such as FAME [29], iRoom [25], i-LAND [40] and Dynamo [24], are early examples of interactive spaces where users meet in a dedicated place to collaborate. With Jigsaw, users can create domestic services by assembling augmented objects [38]. Coupling two tranSticks makes it possible to extend a local interactive space to that of a distant machine [1]. An ambient interactive space can also be viewed as a computational aura that follows users as they move from place to place [16].

K. Coninx, K. Luyten, and K.A. Schneider (Eds.): TAMODIA 2006, LNCS 4385, pp. 1–15, 2007.

These examples show that, with ambient computing, we are shifting from the control of systems and applications confined to a single workstation to that of a dynamic interactive space where the boundaries between the physical and the digital worlds are progressively disappearing. As a result, the pre-packaged well-understood solutions provided by shells and desktops are not sufficient [5], and many interaction techniques are being developed for ambient computing, although on a case-per-case basis. This ad-hoc approach is adequate for local exploration, but may not provide sufficient insights to the problem.

In this article, we propose the concept of meta-UI (meta-User Interface) to denote a kind of interactive system that allows users to control, mould and understand their interactive ambient spaces. In the following section, we define the notion of meta-UI and propose a taxonomic space to understand its nature more precisely. Then, using this space, we classify, compare, and contrast disparate research efforts in the area of meta-UI's to suggest directions for future research.

2 Definition and Taxonomy

A *meta-UI* is an interactive system whose set of functions is necessary and sufficient to control and evaluate the state of an interactive ambient space. This set is *meta-* because it serves as an umbrella *beyond* the domain-dependent services that support human activities in this space. It is *UI*-oriented because its role is to allow users to control and evaluate the state of the ambient interactive space. In the context of this article, a meta-UI *is not* an abstract model, nor a language description, whose transformation/interpretation would produce a concrete effective UI. It is an over-arching interactive system whose role is to ambient computing what desktops and shells are to conventional workstations.

As shown in Fig. 1, a meta-UI is characterized by its *functional coverage* in terms of *services* and *object types*. In turn, the services and objects are invoked and referenced by the way of an *interaction technique* (or UI) that provides users with some *level of control*. An interaction technique is a language (possibly *extensible*) characterized by the *representation* (vocabulary) used to denote objects and functions as well as by the way users construct sentences (including how they select/*designate* objects and functions). Given the role of a meta-UI, the elements of the interaction technique of the meta-UI cohabit with the UI's of the domain-dependent services that it governs. The last dimension of our taxonomy, the *integration level*, expresses this relationship.

Functional coverage, interaction technique, and quality, are typical issues to be addressed when analyzing and designing an interactive system. We have refined these issues for the particular case of meta-UI's into a set of dimensions that are discussed next in more detail.

2.1 Object Types

Objects involved in the services of a meta-UI may be *digital*, *mixed-by-design* and/or *mixed-by-construction*. Applications and files are typical examples of digital objects

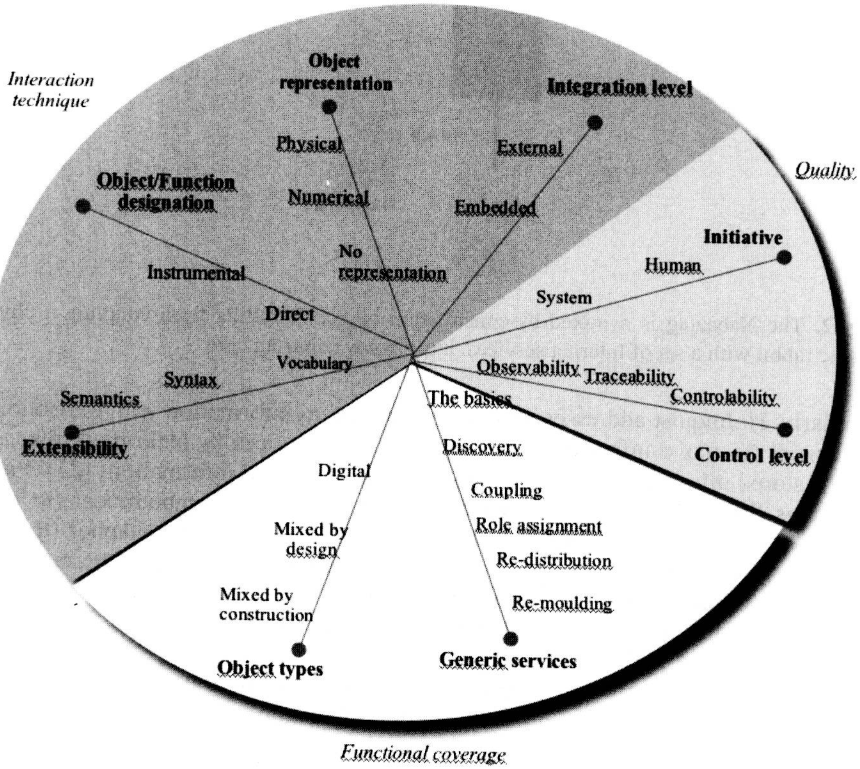

Fig. 1. Dimension space for meta-UI's

manipulated through the services of a meta-UI. Interactors such as windows, pointers, menus and forms, are other examples of digital objects. They are the conceptual units of windowing systems, which, according to our definition, are part of a conventional meta-UI.

A *mixed-by-design object* is an entity that results from the coupling, *by the designer*, of physical entities with digital services. A PDA and a mobile phone are mixed-by-design objects: the assembly of physical components with digital parts has been performed by designers beforehand.

A *mixed-by-construction object* is a mixed object that results from the coupling, *by the end-user*, of physical entities with digital service in order for that object to fulfill its *raison d'être*. For example, to function as a pointing device, the physical object that holds in the hand (called the mouse) must be coupled with the system mouse driver by the end-user. Similarly, the plastic rabbit shown in Fig. 2 must be coupled by the end-user to an Internet service (such as time of the day or weather forecast) to serve as a communicating object.

The distinction between pure digital objects and mixed reality objects is now well understood. For mixed objects, the situation is still unclear. For example, Fitzmaurice's taxonomy applies to a particular type of phicons – the bricks [15].

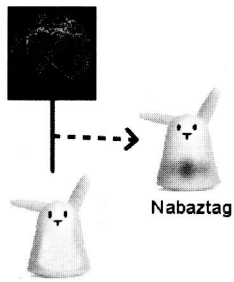

Fig. 2. The Nabaztag is a mixed-by-construction object. It results from coupling a physical plastic rabbit with a set of Internet services. http://www.nabaztag.com

Similarly, Holmquist addresses token-based access to information with the notions of containers (to move information between different devices or platforms), tokens (to access stored information), and tools (to manipulate digital information) [23]. Fishkin structures the problem space of Tangible UI's in terms of embodiment (to reflect spatial relationships between input and output), and in terms of metaphor (to reflect the analogy – or absence of analogy, between the mixed object and the real world) [14]. In particular, a noun metaphor means that an "<X> in the system is like an <X> in the real world", and a verb metaphor, "<X>-ing in the system is like <X>-ing in the real world."

Although these taxonomies can be used to refine the notion of object types, they are limited in scope or serve different purpose. Our notions of mixed-by-design and mixed-by-construction objects are more generic and make it explicit the capacity (or incapacity) for end-users to mould their own interactive space.

2.2 Generic Services

Back in the 1960's, JCL provided end-users with generic services to control jobs execution and to perform files management. In the early 1980's, the Xerox Star introduced additional generic functions such as find, cut and paste, undo and redo. Starting/stopping the execution of a service, moving and renaming files, cutting and pasting data, as well as finding, are the *basics* of conventional meta-UI's. They are conceptually valid in ambient computing, but they need to be extended and refined.

In particular, the notion of finding can be extended with that of *objects discovery*. Objects discovery is key to building a sound mental model of the boundary and of the state of ambient spaces. For example, the Speakeasy browser allows users to explore lists of objects that satisfy some specified search criteria (such as location, object types, availability, etc.) [32].

Because users are not simply consumers, but the designers and architects of their own interactive space, because the system must manage resource allocation dynamically, *coupling objects* becomes key. Coupling is the act of binding objects so that they can operate together to provide a new set of functions that these objects are unable to provide individually [11]. Two ConnecTables can be dynamically coupled by approaching them close to each other to enlarge the screen real estate [41]. With DataTiles, users obtain new services by configuring tagged transparent tiles on a flat panel display [37]. The analysis presented in [11] shows that coupling raises a large number of research issues.

Objects discovery allows users (and the system) to be aware of the objects that can be coupled. By coupling objects, users (and the system) build new constructs whose components play a set of roles (or functions). In conventional computing, roles are generally predefined. Typically, the screen of a laptop plays the role of an interaction resource, and this role is immutable by design. In ambient computing, where serendipity is paramount, *assigning roles to objects* becomes crucial. For example, in the Olympic Café scenario illustrated in Fig. 3, Bob and Jane uses spoons and lumps of sugar to denote the streets and buildings of the city they are planning together [10]. Bob couples a spoon with the table by laying it down on the table. The system can then discover the presence of the spoon and assigns it the role of interaction resource. (The spoon coupled with the system objects tracker and identifier is a mixed-by-construction object.) Then, by uttering the sentence "this spoon is Street Michel-Ange" while pointing at the spoon, Bob couples the interaction resource with a particular digital object known by the system as Street Michel-Ange. By doing so, Bob assigns the role of token[1] to the spoon.

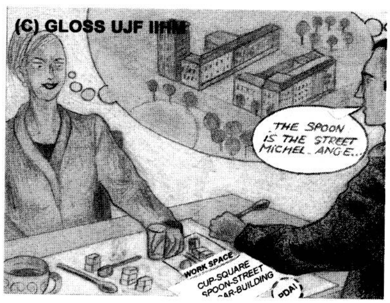

Fig. 3. Bob and Jane use physical objects on the table to illustrate their ideas for the layout of the city they are planning together

One particular role of interest in UI design, is that of input and output interaction resource. In conventional computing, these resources are connected to a single computer. In ambient computing, the platform is a dynamic cluster composed (by the way of coupling) of multiple interconnected computing devices whose interaction resources, all together, form an *habitat* for UI components. Instead of being *centralized*, user interfaces may now be *distributed* across the interaction resources of the cluster.

UI re-distribution, i.e. the application of *objects re-distribution* to UI components, denotes the re-allocation of the UI components of the interactive space to different interaction resources. For example, the Sedan-Bouillon Web site shown in Fig. 4, whose UI is centralized on a single PC screen, is re-distributed in Fig. 5 across the interaction resources of the PC and the PDA. Objects re-distribution and objects coupling may require object re-moulding.

[1] Token, as in Holmquist's taxonomy [23].