

# Towards Complex Object Oriented Analysis and Design

Ludwik Kuźniarz

Maciej Piasecki

Computer Science Department  
Wrocław University of Technology

## 1. Introduction

The *Object Oriented Analysis and Design Methodologies* (OOADM) have been developed for many years and with the emergence of Unified Modelling Language (UML) they are now becoming a matured standard for computer systems development. Advanced tools supporting integrated analysis, designing and development of computer systems are appearing in parallel. Most of them allow easy integration with *Graphical User Interface* (GUI) and impose *object oriented* (OO) construction of *User Interface* (UI). However, in OOADM the problem of proper construction of UI is often limited to technical and presentational issues. In system development OOADM focus on modelling of system structure and behaviour but do not propose any systematic way of describing users' preferences for UI.

The main goal of our work was to construct a complex OO method for system development integrating OOADM and UI development processes as two parallel but interleaving and mutually depended activities. To achieve this the sequence of development phases (together with their goals and notations) is introduced.

## 2. Traditional Approach to UI Design

There are many different methods of UI design. However, most of them cover only some parts of the whole *UI development process* (UIDP) and are incompatible with OOADM. In particular they are often based on early prototyping and do not have explicit links with notations and results of OOADM's sub-

quent steps of development. In the literature a lot of attention is paid to *requirements acquisition* - the first phase of UIDP – where the future users and context of use are described (Newman and Lamming 1995). Many guidelines are also formulated for the last phase concerning the problems of choice of proper GUI standard, application of GUI tools of interaction, layout of the screen, presentation details etc., e.g. (Galitz 1996). However, there is a large gap between the first and the last phase, described only as a “*conceptual design*” which on the other hand is very important to the final shape of UI, because it determines the future semantics of UI i.e. conceptual structures communicated by UI.

### 3. Object Oriented Approach to UI Design

The starting point for the work presented in the paper was an attempt to use OO

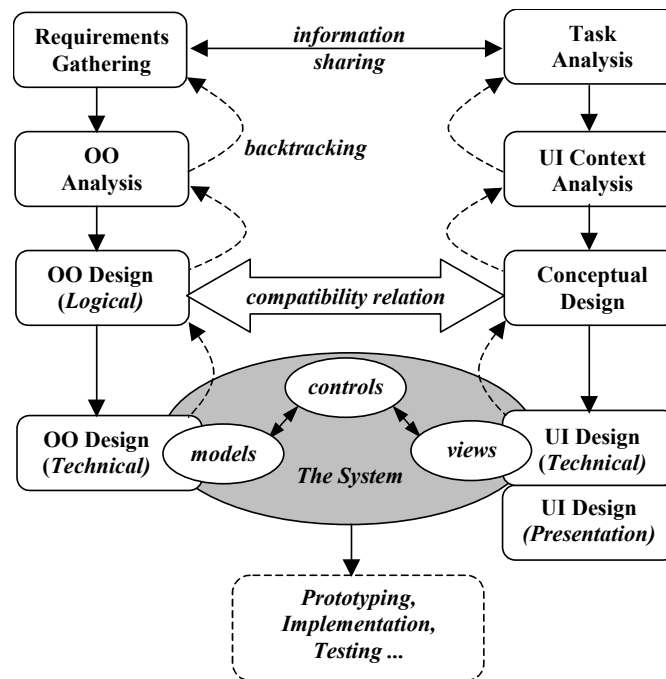


Fig. 1 Scheme of Complex Object Oriented Analysis and Design Method

diagrams to model the conceptual structure of UI and to do this in correlation with OO analysis and design of the system.

The work was inspired by the definition of UI given by L. Barfield (Barfield 1993): “It is composed of those parts of a system that are designed to be apparent to and manipulable by the user and those models and impression that are built up in the mind of the user in response to interacting with these features.”

As well as the emphasis given in the book to the role played by the *user model* (or *user mental model* (Newman and Lamming 1995) i.e. the model of the system possessed by the user) in the interaction between the man and computer. Moreover, the interaction with the system and the presented feedback are identified as the most important factors influencing the construction of user model. It means that UI must ‘explain’ the construction of the system to the user in such a way that the user can understand it and build the user model of the system. To

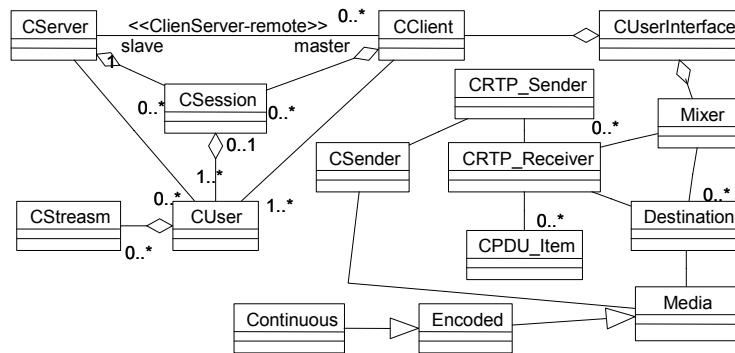


Fig. 2 The system model created during OOAD

achieve this UI does not have to present all details of the system internal construction, but it has to support the user in building a proper picture of the system.

Finally, L. Barfield noticed that the basic function of UI is to allow the user to fulfil his tasks using the system and introduced the four levels of UI description: *tasks, semantics, syntax and presentation levels*.

According to this the proposed Complex OOAD (COOAD) UIDP is split into four phases presented on the Fig. 1.

*The Task Analysis* phase, well described in the literature, e.g. (Hackos and Reish 1998) is parallel to requirements acquisition. *Use cases diagrams* (Rumbaugh J., et al. 1999) can be adjusted to describe the results of this phase.

The main goal of the second phase – *UI Context Analysis* is not only to identify and describe users classes and context of use, as it is proposed in literature (e.g.

ISO 9241), but also to include identification of possible metaphors and users concepts. The metaphors can be described in natural language as well as using simple OO diagrams to capture schemes, structures, patterns of behaviour, which can further be used as the base for some parts of UI. The identified user concepts (and the vocabulary used by the user to express them) should be preserved in further UI design.

The main goal of the third phase - *Conceptual Design* - is to define the conceptual structure of UI i.e. to define what kinds of user models users should build

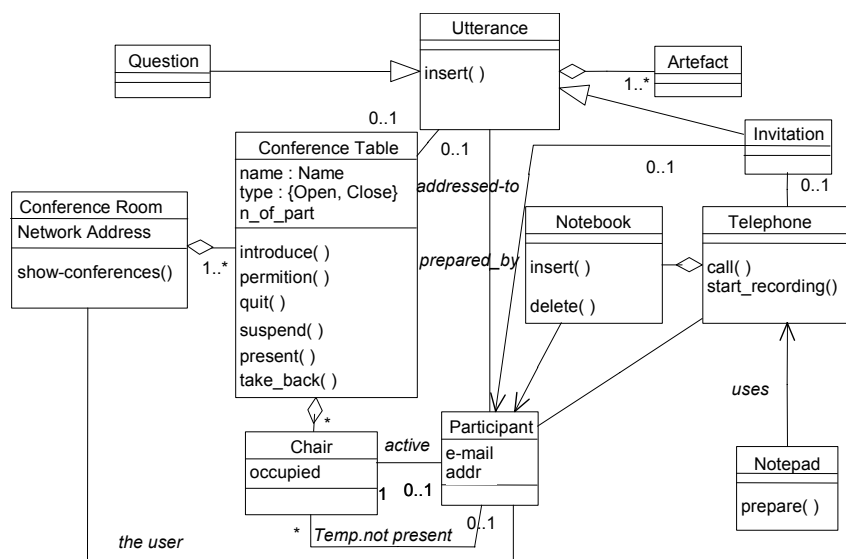


Fig. 3 A user view created during conceptual design phase

as an effect of the interaction with the system. These intended user models are formally described by *user views*. The user view is an OO model describing the structure of intended user model. There is at least one user view for each class of users. Each user view establishes a kind of *perspective* on the system and must be *compatible* with the model of the system created during analysis. The corresponding phase in the system development process is logical design (abstracting from technical details). The starting point for construction of user views is the set of metaphors identified in the second phase. The names of objects and relations come also from concepts collected there. The whole process is a combination of *analysis* done on the information collected earlier and *synthesis* from the structures of the model of the system (simplification) created in parallel.

The next phase – *UI Design* (including presentation design) starts with a choice of GUI standard. Then the user views are transformed into the OO model of UI. The overall architecture of UI is based on the MVC (*models, views, controls*) architecture originally proposed for Smalltalk. The system design model plays the role of MVC *models*, the models created from user views become MVC *views* and the compatibility relation between the system model and user views becomes the OO model of MVC *controls*.

#### **4. Simple Example**

As an example we consider the construction of a teleconferencing system. The system should allow many users to communicate across a computer network using many different media. Users can establish many discussion groups. A particular user can participate in many groups in the same time.

The scheme of the system model created by the independent designer is presented on the Fig. 2. The vocabulary used there is the vocabulary of the system analyst and the diagram represents his view of the system. The user view of the system is presented on the Fig. 3. It shows the different perspective of the system created on the base of simple metaphors: conference room, conference table with a spatial order of participants (chairs) etc.

#### **5. Conclusions**

COOAD is an attempt toward unified method of the system development process based on OO notation. It is not quite matured at the moment. However, it was included into a course delivered to software engineering students and applied to some practical cases realised as student projects. Interesting observation was that designers tried to take different user perspectives during construction of user views. Limitation of the method is assumption of 'a static user' - the natural process of development of the user from 'novice' to 'expert' is ignored at present.

#### **6. References**

- Barfield L. *The User Interface Concepts & Design*. Addison-Wesley 1993.
- Galitz W.O. *Essential Guide to User Interface Design*. Wiley Comp. Pub. 1996.
- Hackos J., Redish J. *User and Task Analysis for Interface Design*, Wiley Comp. Pub. 1998.
- Draft International Standard ISO DIS 9241-11 Ergonomic requirements for office work with visual display terminals (VDTs):- Part 11 Guidance on Usabil-

ity.

Newman W., Lamming M. *Interactive System Design*. Addison-Wesley 1995.

Rumbaugh J., et al *The Unified Modeling Language Reference Manual*, Addison Wesley, 1999