Designing Intelligent User Interfaces: the IUIM Model

DINA GOREN-BAR
Information Systems Engineering Program,
Department of Industrial Engineering and Management,
Ben-Gurion University of the Negev, Beer-Sheva, 84105
ISRAEL

Abstract

Intelligent User Interfaces are human-machine interfaces that aim to improve the efficiency, effectiveness and adaptation of the interaction between the user and the computer by representing, understanding and implementing models.
The Intelligent User Interface Model (IUIM) consists of an architectural model (which defines the components of the model) and a conceptual model (which relates to its contents and behavior). The conceptual model defines three elements: an Adaptive User Model (including components for building and updating the user model), a Task Model (including general and domain-specific knowledge) and an Adaptive Discourse Model (assisted by an intelligent help and a learning module).
We described an application of the IUIM named STIGMA - A STereotypical Intelligent General Matching Agent for improving search results on the Internet.
We compared the new model with others, stating the differences and the advantages of the proposed model.

Key-Words: Intelligent Interfaces, Adaptation, Task Modeling, User Model, Model-Based Interface Design.

1 Introduction

In recent years we have witnessed research and application developments aimed at simplifying the communication between the user and the computer. This is accomplished by adding intelligent components to the interface, such as the "dancing clip" of Microsoft Word, which tries to provide the user with context-sensitive help. However, meaningful progress will involve the understanding of user needs and tasks, to assist, guide, or help users accomplish their work. These systems have to monitor input activities and construct a model of the user. Using this knowledge together with complex tutorial and helping strategies, it should be possible to assist users differentially, according to the task to be performed and the user's experience [3].
In order to develop Intelligent User Interfaces in an orderly path, we need to work within a conceptual framework which will help us to define the needed components, their rules, inter-relationships and possible arrangements. The IUIM provides us with such a framework.
In the next sections we introduce the Intelligent User Interface Model (IUIM), which consists of two sub-models: the Architectural Model (which defines the components of the model), and the Conceptual Model (which relates to its contents and behavior).
Later in this work we will describe one sample application of the IUIM, an intelligent agent for improving search results on the Internet.
We will conclude by comparing the new model with others models, and propose directions for future research.

2 The IUIM's Architectural Model

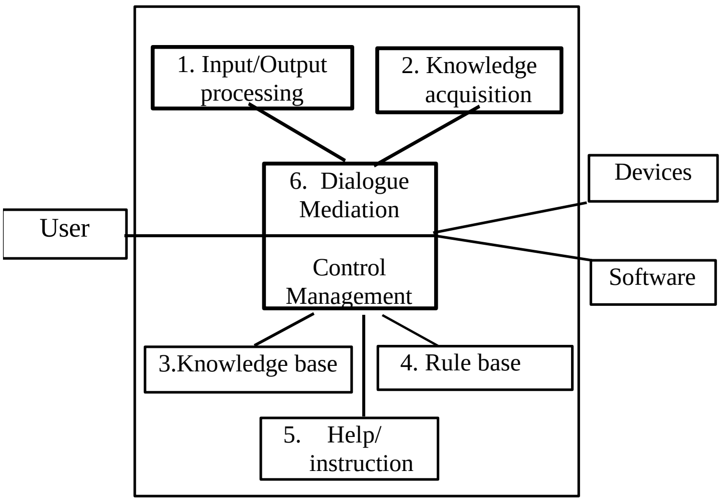
The architectural model describes the various components integrated in the IUIM (see Fig.1).

1. Input/Output Processing:

Intelligent multimodal interfaces incorporate a broad range of input devices (e.g., pads, video and audio recording, eye-head trackers, spoken language, positioning systems, sensors) that support asynchronous, ambiguous and inexact inputs.

Humans are multimodal by nature (may hear, see and touch simultaneously). Similarly, multimodal communication requires the analysis of several input modalities, integration, and, in some cases, reasoning about user intentions, to achieve successful output generation.

Fig. 1 - The IUIM's Architectural Model



2. Knowledge Acquisition and Representation Mechanisms:

IUIM enables the implementation of various strategies for Knowledge Acquisition such as statistical methods (in the case of content analysis), intelligent agents (for building and updating knowledge bases) or direct input from the user by conducting a short interview or filling predesigned forms. Its acquisition of knowledge is based on a default model and its dynamic adaptation through user interaction. Task models are represented using task decomposition techniques (HTA) with plan recognition.

3. Knowledge Bases:

Rissland [8] has defined seven knowledge bases required for an intelligent interface: the user, the tasks, the tools, user's area of occupation, modes of communications, communication routes, interface evaluation. IUIM mainly adopts these knowledge bases, with some modifications. For instance, we added to the areas of occupation the representation of job structure.

4. Rule Bases

Rule bases enable linking various knowledge bases and making inferences.

5. Help/Instruction Mechanisms:

The IUIM supports the system's adaptation to the user's skill levels and preferences represented in the User Model, using smart help and instruction mechanisms. The model supports user-controlled self-adaptation, i.e., the help adaptation process is performed automatically, but the user may intervene at any given moment and modify it.

6. Dialog Mediation and Control Management Rules:

A protocol management module stores interaction data in a database. Emphasis is placed on various measures, which help to estimate the user's level of knowledge in specific subjects and evaluate task performance.

3 The IUIM's Conceptual Model

The conceptual model defines three elements: an Adaptive User Model (including components for building and updating the user model), a Task Model (including general and domain specific knowledge) and a Discourse Model (to be assisted by an intelligent help and a learning module). (Fig. 2)

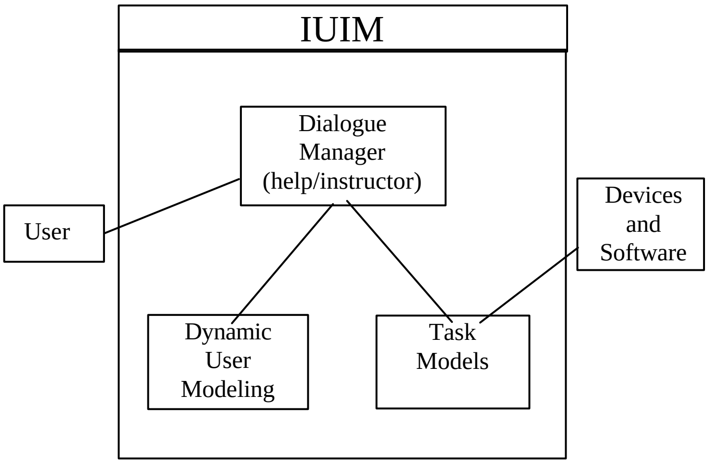
In the IUIM conceptual model we address questions which pertain not the architecture, but to process, such as, how the adaptation occurs, under which circumstances it is preferable to update a stereotype, or how to solve contradictions among stereotypes. It should be noted that identical processes might take place under different architectures.

A. Multi-Stereotype-based user model:

The construction of a stereotype-based user model is the most popular approach in user modeling studies. Stereotypes may be regarded as a pre-programmed model, which fits a group of users. A stereotype system should thus include a classification mechanism to determine which stereotypes are applicable to the present user [1, 2]. The task of such predictions is not to replace any specific information about the individual, but to provide a basis for action until such information is made available.

The user model of the IUIM adds more capabilities to the traditional stereotype based user model by enabling the assignment of several stereotypes to a single user. The user model is flexible, enabling addition and modification of attributes, stereotypes and users, and addressing contradictions and conflicts among stereotypes.

Fig. 2 - IUIM's Conceptual Model



Basic Principles in User Model Characterization

The following are our basic assumptions for the characterization of a stereotypic model:

- A user may belong to a group (stereotype) or several groups in any hierarchical level. For example User A may be an athlete and an engineer, User B: a runner, blue-eyed and a production planning engineer and so.
- User groups may be flat (a single definition), or hierarchical, such as an production planning engineer, which is a subgroup of industrial engineer, which is a son of engineer
- At each level in the tree attributes may be assigned. In addition, it "inherits" attributes from the group above it. The model supports a definition of level of confidence for attributes ranging from 0-1.
- Primary attribute categories are called attribute types, such as education, roles, beliefs, goals, etc.
- Each attribute type may have one or more attributes.
- There is a distinction between direct observation and assumptions. A direct observation assigns the value 1 to an attribute.
- Conflict Resolution: The following are the types of contradictions handled by the system:
  1. Assumptions within the same tree branch: A contradiction resulting from a lower node - for instance, a reasonable clinical psychologist who is a computer illiterate, versus a clinical psychologist engaged in sorting, who is proficient in computerized sorting systems.
     The solution ⇒ using a value of an attribute of a lower hierarchical group.
  2. Contradicting assumptions in parallel branches: a user is both a clerk (assumed to be less physically fit due to long hours in the office), and a runner (top fitness of course).
     The solution ⇒ assigning numerical values according to a level of certainty for each attribute.
  3. Observations contradicting assumptions:
     The solution ⇒ direct observation overrules a stereotypical assumption.
  4. Two conflicting attributes were assigned the same numerical value
     The solution ⇒ asking the user
- There is no exhaustive list of attributes defining the model. During the interviews, several common attributes were identified in user group descriptions, among them experience, education, and nature of usage. One can think of various additional attributes, specific to special populations (such as children, older adults, pilots and others), so that the model must be dynamic, enabling the addition and modification of attributes as needed.
- There is no exhaustive list of populations. New hierarchies may be defined at any given moment.

B. The Task Model

The IUIM enables task description using task decomposition, knowledge-based techniques or entity-relationship based analysis, according to the described application [3].

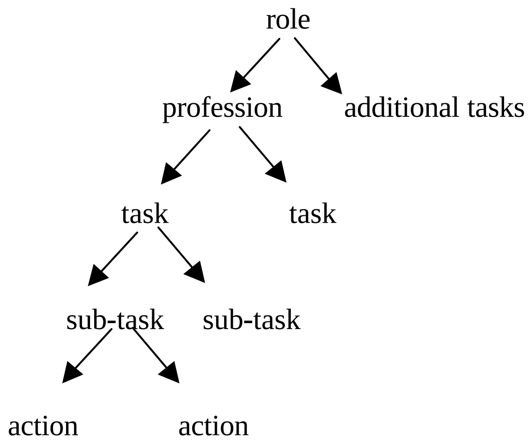
Tasks are actions, which the user considers meaningful. In defining a task one must address a dimension of intent, which does not exist in the description of a function, that is, the task is performed by the user for the purpose of attaining a goal. Therefore, a task model should describe what users do in the course of their work in formal terms and what the user should do in order to attain a certain goal.

Basic Principles of Task Model Characterization

The following are our basic assumption in the characterization of a task model:

- The task analysis approach is user-centered.
- There is a strong relation between the task and the user performing it. The manner in which the task is performed is affected by the user's experience.
- A collection of discreet actions and their control structure are called a task (or a sub-task). (See Fig. 3).
- A collection sub-tasks defines a task.
- A collection of characteristic tasks describes an occupation/profession.
- A collection of tasks performed by the user is called a role.

Fig. 3 - The Linking Structure between actions, tasks and role



- A role may contain profession-characteristic tasks and other special tasks, or any other combination of tasks performed by the same person.
- There are characteristic tasks of various user groups based on their profession or role.
- An action has preconditions and effects
- A plan is an sequenced set of actions, sub-tasks or tasks
- We represent two types of tasks: interface-communicative (shared by applications, i.e., send message, copy) and domain-specific (pertains to a specific application, i.e., present blood-test results)
- Task analysis may be performed under any known methodology using formal tools.
- Various methodologies may be combined for task analysis
- When there is more than one way to perform the same task, all possible ways must be described to provide the user with flexibility and learning ability.

C. Dialog Manager and Adaptation Dynamics

Dialog adaptation is performed as a function of the user model referring to his/her expertise and knowledge level in various areas. As such, it does not require a separate methodology and instead uses the multi-stereotype-based user model.

The dialog manager in the IUIM operates smart adaptation and help mechanisms. The dialog manager connects between the user and his/her representing models: user model and task models. According to the IUIM, dialog adaptation is performed as a function of the user model referring to his/her expertise and knowledge level in various tasks.

Basic Principles of Dialog Adaptation in the IUIM

The model is guided by the following principles:

- The dialog manager connects between the user and the models representing him/her: user model and task models.
- The IUIM defines adaptation as an ordinal variable with varying degrees (high, medium and low, for instance). Systems may be classified according to their degree of adaptation to the user.
- Every computerized system, and therefore every interface, includes some kind of a model of the user and the task.
- The user model may be implicit/hidden (as in a word processing program) or explicit (as in specialized systems pre-defined for a specific user, which enable the user to receive a report about the user's description in the system at any given time).
- The dialog adaptation is performed as a function of the user model, referring to his/her expertise and knowledge level of various tasks. The

adaptation is based on the user's data (goals, preferences or actions).

- The user has control over the adaptation process. Adaptation will not cause changes in the user's actions.
- According to the IUIM concept, the adaptation takes place across four processes: user model definition, default selection, model acquisition and user learning.
- The presence of processes 2-4, all or in part, indicates the existence of a certain degree of adaptation.
- Within the processes of default selection, model acquisition and user learning, there may be varying levels of adaptation to the user.

4 Current Applications of the IUIM

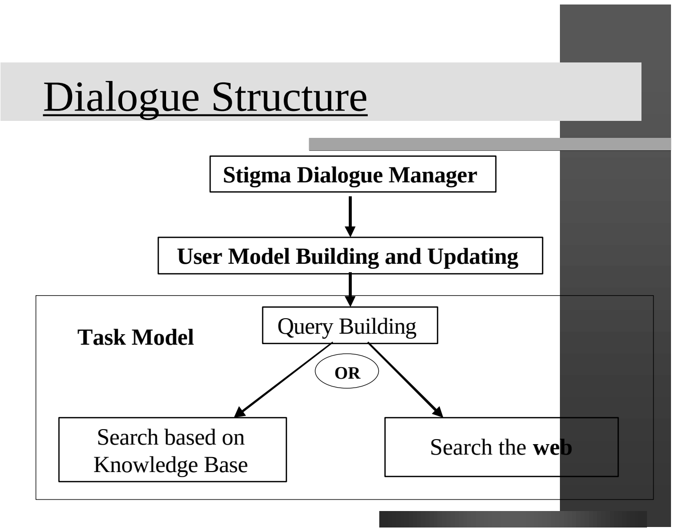
The IUIM is being applied in several domains, such as enhancing the intelligent capabilities of Windows applications (Word, Excel), Web browsing agents, Home Health-Care Telemedicine applications, and intelligent interfaces for specific populations (elderly people, managers).

We will describe one implementation of the IUIM:

STIGMA - ***A STereotypical Intelligent General Matching Agent for Improving Search Results on the Internet***

We have developed the first version of a multiagent general searcher (see Fig.4).

Fig. 4 - The IUIM conceptual Model implemented in STIGMA



The system components are:

- User Model - Multi-Stereotyping User Model, which may represent a user, engaged in a role (student, lecturer, researcher, manager), studying or interested in diverse topics. The user model collects information from several sources: prompting the user for the first time, email analysis, and from the user's organizational role. The knowledge extracted is processed to resolve conflicts between attributes.
- Task Model: STIGMA's Task Model has several components:

1. A query builder, which possesses knowledge about the correct syntax, tips and rules for well-formulated queries.
2. A concept representation of subject areas. By the moment we have a limited representation of engineering subject areas.
3. Knowledge bases, which are fed from the interaction with the user and will be used for future focused search.

- Adaptive Dialogue Manager:

1. Dialog Monitor: Gets all the inputs from the user and transfers the right information to the appropriate module
2. Adaptive Help - Based on the User Model, this component behaves differently towards users from diverse levels of knowledge and professions.
3. Presentation Generator - Gets the answers, checks for site repetition and shows them to the user according to the user's presentation preferences.
4. Learning Module: Analyzes user's responses to the answers received from the search engine and incorporates relevant information into the knowledge bases for future use.

5 Related Work

Previous model-based systems have been developed.

UIDE [4] included a partial interface model by which presentations were generated from data models. It tried to automate the domain-to- presentation mappings. MECANO [6] used domain models to generate form-based interfaces but lacked a user-task model.

The Architectural Model for an Adaptive Intelligent Man-Computer Interface [5] emphasized the interface's adaptability to the user and familiarization over time, as an expression of its intelligence.

HUMANOID [9] applied templates to bridge the domain-to presentation mappings, which proved to be an effective way of building interfaces to display and visualize hierarchical information. Puerta & Eisenstein [7] introduced the idea of a general computational framework for solving the mapping problem in model-based systems: the interface development environment named MOBI-D (Model-Based Interface Designer). It is supported by a set of tools, which help to design a wide variety of user interface types.

6 Discussion

In this paper we have presented the IUIM in its two dimensions - the architectural and the conceptual.

The description of both models is of utmost importance for the sake of clarity. The architectural model of an intelligent user interface describes its components. The IUIM adds a second dimension: the conceptual aspect. The combination of the two models enables us to refer to the model's components on the one hand, and to the nature of its behavior on the other. The arrangement of the components is flexible, so that various architectures may be derived to describe various types of intelligent interfaces We know of interfaces comprising information-rich components and a variety of communication modes, but are not operated in a way that conveys an intelligent behavior (the user must know the interface and how to perform each task, instead of the interface knowing the user and adapting itself to him/her).It seems that the architecture is vital to the existence of an intelligent interface, but the essence of that intelligence is found rather in the conceptual model, on which we focused in this work. That is the main contribution of the IUIM - a systematic and detailed description of the working processes of an intelligent interface's conceptual model.

IUIM doesn't share the deep concern about the automation of the domain to presentation-mappings as UIDE, MECANO and HUMANOID do. Automation seems to be important because it helps to make less mistakes and shorten the development of the implementation's interface, but IUIM achieves the same goals with a different approach, by the development of clear models. From our experience in implementation, the methodology seems to be clear, well defined and easy to convey, enabling reuse of large parts of the models and sharing them in different applications. We intend to test some of these assumptions in a controlled laboratory setting.

We believe that general automated tools will enable, at most, the development of low common denominator predefined simple interface design. Instead, we propose building Task Knowledge Bases encompassing the knowledge and experience of groups of experts from specific domains based on interviews and observations as a starting point, and adaptation and learning modules for incorporating user experiences.

It seems that we are evolving in the same direction that MOBI-D does, i.e., building a general computational framework for interface development environment, supported by a set of tools which help to design a wide variety of user interfaces types, but the framework and the tools are completely different.

References:

[1] Allen R.B. User Models: Theory, Method, and Practice. *International Journal of Man-Machine Studies*, Vol. 32, 1990, pp.511-543.

[2] Chin, D.N. Knome: Modeling What the User Knows in UC. In Wahlster, W. & Kobsa, A. (Eds.) *User Models in Dialog Systems*. Springer-Verlag, Berlin-New York, 1989, pp.74-107.

[3] Dix, A. Finlay, J. Abowd, G. & Beale, R. *Human-Computer Interaction*. 2nd Ed. Prentice-Hall, UK, 1997.

[4] Foley, J. et al., UIDE- An Intelligent User Interface Design Environment in Sullivan, J.W. & Tyler, S.W. (Eds.) *Intelligent User Interfaces*. Reading, Mass., Addison-Wesley. 1991, pp.339-384.

[5] Hefley, W. E. & Murray, D. *Intelligent User Interfaces. In Proceedings of the Intern. Workshop on Intelligent User Interfaces.* Orlando, Florida. ACM Press, 1993, pp.3-10.

[6] Puerta, A.R. The MECANO Project: Comprehensive and Integrated Support for Model-Based Interface Development in *Proc. of CADUI96: Computer Aided Design of User Interfaces*. Namur, Belgium, 1996.

[7] Puerta, A. Eisenstein, J. Towards a General Computational Framework for Model-Based Interface Development Systems. *In Proceedings of the International Conference on Intelligent User Interfaces.* Los Angeles, CA. ACM Press. 1999, pp.171-178.

[8] Rissland, E.L. Ingredients of Intelligent User Interfaces, *International Journal of Man-Machine Studies*, Vol.21, No.4, 1984, pp.377-388.

[9] Szekely, P. Luo, P. & Neches, R. Beyond Interface Builders: Model-Based Interface Tools in *Proceedings of International Conference on Human Interface - InterCHI'93*. ACM Press, 1993.