© Pete Saloutos/The Stock Market

Model-Based Design of Interactive Applications

Model-based approaches aim to find declarative models that allow designers and developers to concentrate on relevant aspects of their work without being immediately immersed in details of implementation. Such approaches have developed mainly in academic environments, with limited use in developing real applications. The advent of the World Wide Web has stimulated a diffuse interest in developing and using hypermedia and consequently a strong need for structured frameworks to support design of usable hypermedia (Schneiderman 1997). Often such hypermedia must support access to the same information from different types of users.

Fabio Paternò and
Cristiano Mancini
CNUCE-C.N.R.
Via S. Maria 36
56126 Pisa, Italy
F.Paterno@cnuce.cnr.it
and
C.Mancini@cnuce.cnr.it
giove.cnuce.cnr.it/

Consequently, we developed our work with the goals of

- Investigating how a model-based approach can support the design of hypermedia that can be used by different types of users, and
- Applying the criteria identified to the development of a real application that has a wide variety of users.

For this purpose we needed a method for supporting the design and implementation of easy-to-use and adaptable hypermedia to overcome the main limitations that can often be found in current approaches:

- *Design completely based on ad hoc solu-*

tions and the intuition of the designer can be successful in a few cases, but generally designers will have to solve problems without having methods supporting elements to provide effective solutions. Moreover, they may propose inconsistent design, providing different solutions in similar situations and confusing the end user;

- *System-oriented models used in the design of hypermedia* are oriented to the systematic structure of the data and the software architecture. They are useful in solving technical and implementation problems but contribute only partly to introducing user-oriented elements, thus allowing designers to implement applications that are engineered from an implementation point of view but are not usable enough.

We believe overcoming these limitations requires strong user involvement. However, users are not designers. They can supply useful information to designers, who should then use it in developing the presentation and the navigation of hypermedia. Design is a complex activity. It needs to be supported by principles that incorporate the designer's skill and experience. It also must be able to support the management of the increasing complexity of the information contained in a hypermedium in order to obtain applications usable for the end user. To this end we have developed a structured method based on the following elements:

- *User involvement.* Involve users from the early phases of the design cycle in order to achieve task models that are meaningful by incorporating the intentions of the users into the application.
- *Task models for the various types of use foreseen.* A wide variety of users can access a hypermedium with different purposes; thus, it is important to have different task models for different types of user.
- *Rules for structuring a hypermedium according to the task model.* Nonprescriptive and flexible descriptions of the possible tasks and their performance can give useful information during various phases of the hypermedium design.

Task models allow designers to address the specification of functional and interaction aspects within an integrated framework. Interest in their use (Bodart et al. 1994, Paternò 1999, Szekely et al. 1995) for designing interactive applications is becoming greater and greater. They are an important topic that belongs to the intersection of human–computer interaction and artificial intelligence. Our contribution compared with other approaches to task modeling is that we provide a richer set of temporal operators capable of supporting precise descriptions of concurrent, interactive, and interrupting tasks. Previous works have focused mainly on the design of the user interface, in some cases including its underlying software. We have not found proposals that also introduce user- and task-related aspects in structuring hypermedia, which have specific navigation and presentation styles. Hypermedia are characterized by supporting access to a large amount of information organized into numerous nodes that are linked to each other, with the ability to display to the user only one node at any time.

We recently developed a proposal for applying a comprehension model in the design of hypermedia manuals (Paternò and Mancini 2000). Garzotto et al. (1995) analyze and evaluate a museum application using the hypertext design model (HDM) for hypermedia structures. This model has been influenced chiefly by models for the design of databases, so it is useful for identifying inconsistencies in the structure of the data and the related presentation and navigation. However, we believe that task models can provide further information in designing usable hypermedia. For this purpose task models have to be developed with the involvement of users so that they incorporate users' requirements for navigation and presentation design. An interesting approach in this direction is the Object-Actions Interface model (Schneiderman

1997), which provides a helpful guide to Web site designers in breaking down a complex information problem and fashioning a comprehensible and effective Web site. Our purpose is similar, but we provide a more formal approach, considering a wide set of temporal relationships and aiming at obtaining more structured task models and systematic indications.

When a hypermedium is used by users with different goals and levels of knowledge another important aspect is to support adaptation: different users may be interested in different parts of the content and they may want to use different links for navigation. Systems that allow you to modify some parameters of the system and adapt their behavior accordingly are called adaptable. We believe that *to design adaptable hypermedia we need to take into account that different types of users have different task models associated with them.* This means that if task-based design is applied, the hypermedium should be able to adapt itself to this diversity of possible tasks depending on the type of user.

In this paper we discuss task-based design of adaptable hypermedia and how we have applied it to a case study: the design of hypermedia containing information on the Museum of Marble in Carrara (Italy). We highlight aspects that may be useful for other designers or researchers working on similar problems.

Method

The method we use starts with an informal task analysis. Users were strongly involved, through interviews and questionnaires, and we analyze which tasks are supported by similar applications and how. The purpose is to identify the tasks that the new application has to support and the current problems in their performance in order to understand how to improve it. Once we have identified an informal list of tasks to support and further requirements to satisfy, we begin building a task model that describes how the tasks should optimally be performed.

We use the ConcurTaskTrees notation (Puerta et al. 1999) to describe the task model. It is an automatic tool-supported notation (giove.cnuce.cnr.it/ctte.html) for specifying task models in a hierarchical way. ConcurTaskTrees has a rich set of operators, allowing designers to give a compact specification of many possible types of temporal relationships. This capability allows them to describe flexible, nonprescriptive task models.

In building the task model we took into account the data available to us to support the tasks.

The development of the task model is the result of an interdisciplinary discussion involving many actors: designers, developers, managers, application domain experts, and end users. Once we have obtained task models for the classes of users identified, we use them to drive the design of the hypermedium because they contain many useful suggestions for a user-oriented design. The task model can give information about how to structure the multimedia presentation and the links supporting the navigation. The resulting hypermedium contains the available data, and its structure can be described by models for hypermedia.

This initial hypermedium design can be evaluated by metrics for hypermedia and presentation design (Paternò and Mancini 1999), which have been shown to include valid indications for obtaining more usable applications. The evaluation based on metrics is not considered in this paper.

We performed empirical evaluation of the hypermedia in order to validate the usability of the hypermedium obtained by a task-based design. Empirical testing can be expensive because it can take a lot of time before meaningful results are produced. However, the use of our method, which includes many user requirements by using task models developed with strong user involvement, allows designers to reduce the amount of user testing. This type of evaluation is always important. Sometimes it does not change the overall structure of the hypermedium but it always helps to improve small aspects of the design. We can thus obtain the final hypermedium

and the related task model.

Informal Task Analysis

We applied our method to the design and development of a hypermedium for accessing and navigating information on the Museum of Marble in Carrara, Italy. Carrara has a large concentration of white marble quarries. Marble has been used for artistic works since the Roman period. The museum is interdisciplinary, containing modern sculptures made from white marble or other materials, pieces of industrial archaeology that were used to quarry the marble in the past, a gallery of marbles (samples of many types of marbles and granites), technical applications of marble in architecture, artisanship, and so on. The town can be considered a natural extension of the museum, housing many works of art (sculptures, monuments, marble icons).

While developing task models for our application we decided first to interview experts in the field and possible end users. We soon recognized that we had to design slightly different task models for different types of users because people with different backgrounds interact with the same information for different purposes and in different ways. We decided to give users the ability to dynamically change their profiles during the application session, so that they could navigate through the museum information and have it presented differently.

This work considers three types of users: tourists, students in artistic heritage, and experts. They differ mainly in the knowledge of the information contained in the application and in the preferred style of navigation. When there is a classification of possible users it is always possible to find a few cases that do not fit exactly into any of them; however, we find this grouping suitable for most users. In the next sections we describe the requirements highlighted by interviews and meetings that we had with possible end users, application domain experts, and employers of the museum.

The main requirements for the different types of users identified are as follows:

- *Tourists* are characterized by a low average knowledge of the topics considered. Usually they prefer to have guided tours through the rooms of the museum and the town with pictures and information about the works of art. However linear predefined tours alone would be too restrictive, so some degree of navigational freedom is important. Tourists want general information on the artistic works, and this information has to be presented clearly and in a limited amount because it has to be interpreted easily. Thus, a work can be presented by an image, the title, a short description, the name of the author, the material and technique used for its creation, and when it was made.
- *Students in the artistic field* who visit a museum already have some basic knowledge. They prefer to have information on a wider variety of topics than tourists. The presentation of some images and related texts often stimulates a request for more detailed information. In this case providing predefined tours is not the right answer. Instead, we let students choose from different types of information about not only the works of art but also a wider spectrum of topics, so that students can improve their knowledge. They also appreciate the availability of a technical glossary explaining specific terms. Examples of information that they may be interested in are the life of an artist, in-depth information on the theme of the work, its relationship to other works, the state of conservation, and so on.
- *Expert users* generally know exactly what information they want. Thus, they should be allowed, right from the beginning of the session, to make increasingly precise requests, for example, directly specifying the name of the artist of interest or the historical period or title of the work (see Figure 1). In

this case the information they require may relate to:

- An artist. Initially a critique may be useful (not a simple biography, which an expert is presumed to already know), from which it should be possible to get information on the artist's work and texts or Internet sites that discuss them, plus a short summary.
- Artists who lived in the same period or area, or both.
- Works of art. Presentation of the work should yield information about the artist who created it and in-depth descriptions.
- A specific historical period that gives the expert user the chance to get detailed information such as how to access—through the Web—the works characterizing it.

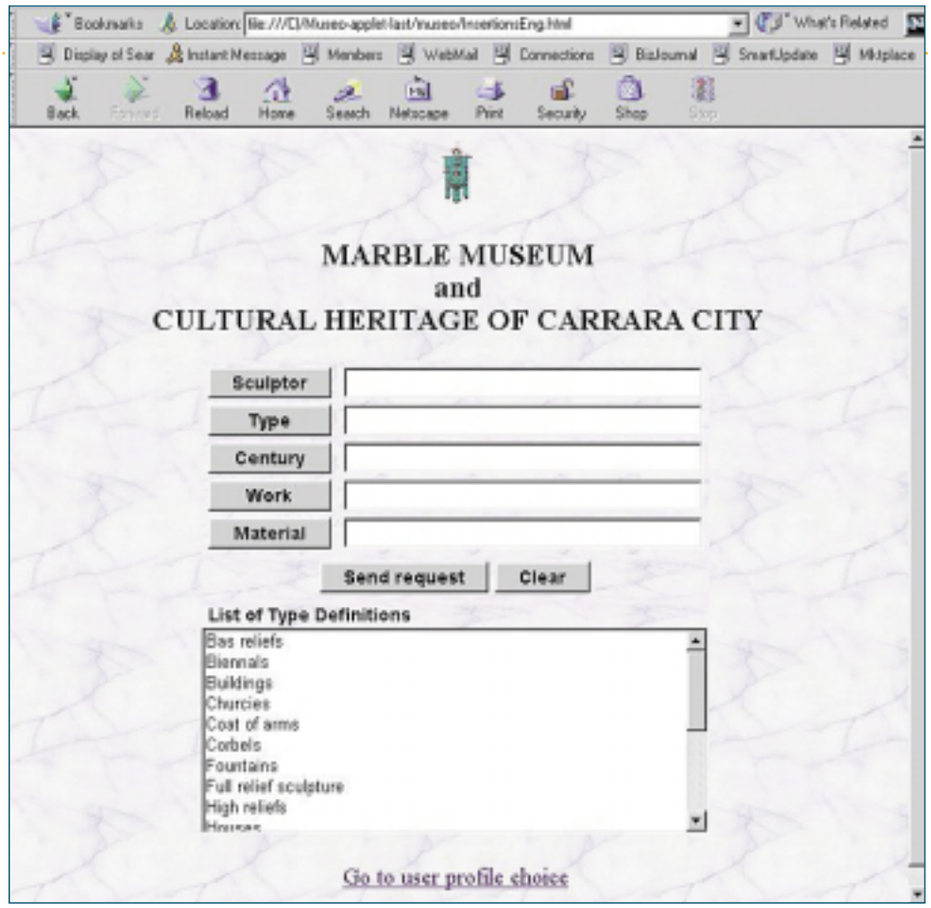

Figure 1: Initial choice of information for expert users.

Task Models

Task models describe how to perform possible activities to reach the users' goals without violating temporal or semantic constraints. Task models should not be prescriptive; therefore, it is important that they provide, as in our approach, the ability to describe a rich set of temporal relationships so that parallel tasks, alternative ways to reach goals, and dynamic enabling and disabling of tasks can be described.

A task model records the results of the discussion among the different actors involved in the design. The specification should be used by the designer in order to remove ambiguities, to evaluate design options, and to check the completeness of the design. Task models can be used to support the development of hypermedia. They are also useful for supporting the ability to modify related hypermedia. When new tasks have to be supported or users require the capability to perform existing tasks in different ways, task models immediately indicate what part of the application has to be modified and how it relates to the other parts.

ConcurTaskTrees is the notation that we have developed to specify task models. It is a graphical notation that can be used to describe tasks at different levels of abstractions. We can distinguish categories of tasks depending on the allocation of their performance (to the user, to the application, to their interaction—otherwise, we have abstract tasks that are not univocally allocated for the performance). In the task specification unique icons indicate the category of the task. In addition, we have a set of operators describing the possible temporal relationships (see Figure 2); in this paper we will not describe all the details of the specification.

Figure 2 describes the first levels of the task model structure for the tourist. At the first level we distinguish between the activities that can be performed during the session (*AccessWorksOfArt* task) until they are disabled ([> operator) by the CloseSession task. Iterative tasks are indicated by the * operator. At the beginning is a task (*AccessGenInfo*). Its subtasks allow users to access various information: general information about the museum (fares, opening time, how to get there), the ability to change the language, the ability to activate some music. Then the user can explicitly start the session (*StartSession* task). Once this task is terminated it enables (>> operator) the task performed by the application to present the possible user profiles. An optional task allows the user to have more information about the user profiles available (*AccessInfoProfile* task). Optional tasks are indicated by square brackets. For the tourist profile, when the user selects the user profile, he or she can choose ([ ] operator) from four types of information: general information on the town, general information on the museum, activation of the interactive map of the museum, and activation of the interactive map of the town. Depending on the selection it is possible to access different information ([ ]>> means sequential tasks in which the former task provides some information to the latter task). If, for example, activation of the interactive map of the museum is selected (*SelMuseumMap* task), after its presentation (*ShowMusumMap*) the user can select a room and navigate through the related information (*AccessInfoRoom* task). This task is then broken down into other activities (an abstract task is indicated by the cloud icon) that are not shown in the figure. The task of accessing information about a room can be disabled by allowing the user to go to the museum map (*GoMuseumMap* task) again or by the task that allows the user to go to the initial choice (*GoTouristOpeningPage* task) of information for the tourist user or by the task that ends the session.

Using Task Models to Design Hypermedia

The task model can be used to support the design of the hypermedia structure underlying

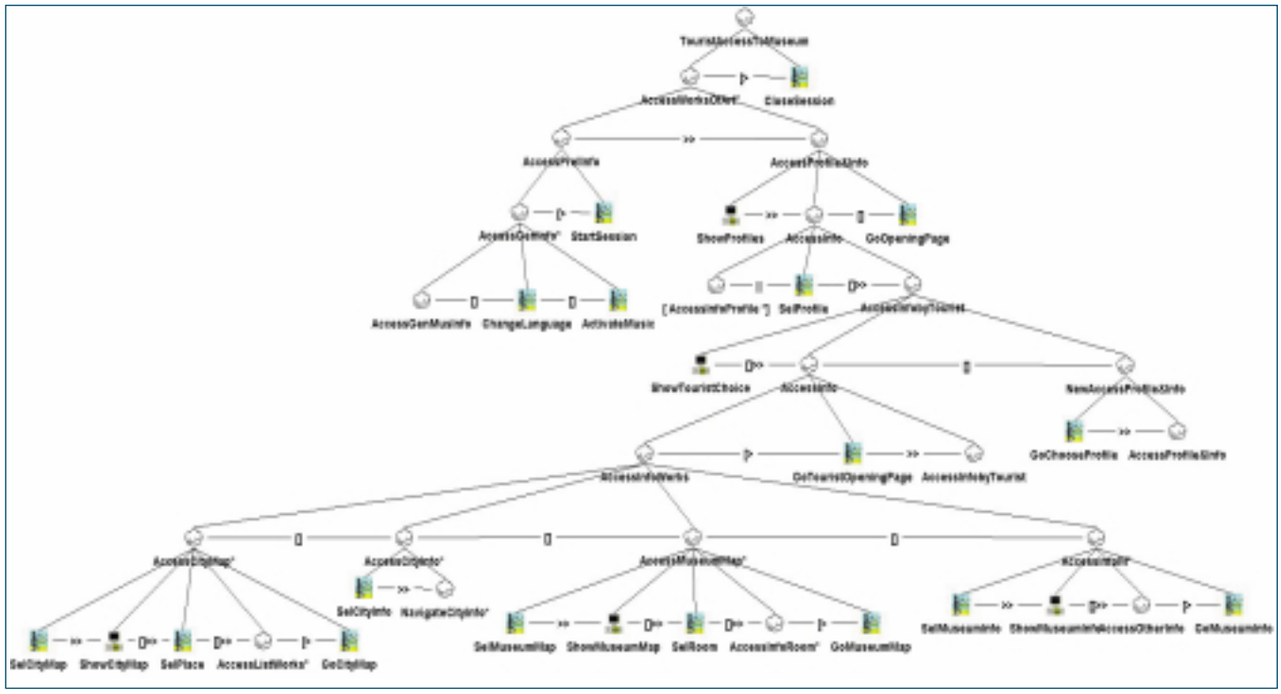

Figure 2: Example of task model specification (tourist user).

the user interface that will be used to communicate with the user. Specifically, the task model is useful for designing more user-oriented interactions because they will be structured according to the user's conceptual model of the possible activities. Thus, it is important to avoid inconsistencies between what has been specified in the task model and what can really be done in the implementation. For example, if at the task level it is specified that after presenting one work it is possible to pass to the next work (according to a given order) at the hypermedia level, we must have a link between the presentations of the two works to support their sequential access.

We have developed a method that allows designers to use the task model to identify the elements of a hypermedium and their relationships. To this end we have developed different types of rules:

- *Rules for designing the presentation.* Depending on different aspects, such as the type of task, we have different requirements for choosing the media and related interaction techniques in the user interfaces that should support performance of the task.
- *Rules for designing the navigation.* These rules are mainly based on analysis of the temporal relationships among tasks that should be supported and indicate how to use the links in the hypermedium.
- *Rules to structure the information.* These rules indicate how to structure the data contained in the hypermedium.

Structuring the Presentation

In designing the presentations supporting the various tasks we must take into account many aspects:

- *Task type.* We can decide the type of presentation according to the type of task, taking into account what information the task has to communicate. For example, if the task has to present a spatial relationship, it is important to provide a graphical presentation that highlights the elements defining it.
- *Cognitive effort.* Different ways exist for reducing the cognitive effort required from the user. It is important to balance the use of different media, especially when they are used to support concurrent tasks. An example is when brief information, complementary to that presented on the screen, is presented by audio. Generally, it is possible to read a text and hear a sound or to speak and watch images simultaneously, whereas it is problematic to hear a long description and read text at the same time. It is important to exploit the capabilities of our cognitive system to blend information that is perceived by different perception channels; however, this blending needs to be helped by carefully synchronizing information that is presented by different media simultaenously.
- *Task frequency.* Optimizing the resources used within a certain medium to support frequent tasks or frequent sequences of tasks is important. For example, if we know that the user often has to use a scroll bar and then select a button, the button control should be placed close to the scrollbar.
- *Contextual environment.* We must take into account the context in which the application is used; for example, if the context is a communal area it may be noisy and audio will not be effective.
- *User knowledge.* Depending on the user's knowledge of the application domain we should present the information in a different way. If the user is a beginner, a limited amount of information with clear fonts and colors is preferred.
- *Layout optimization.* Performance of the same task sometimes requires different amounts of information depending on the specific instances of objects involved (for example, some works of art require longer descriptions). This means that the structure of the presentation remains the same (the interac-

tion techniques and the links are in the same part of the presentation), but either to avoid leaving large parts of the screen unused, some part of the layout is automatically resized or when there is a lot of information, it has to be split into multiple presentations sharing the same structure but with different information.

Structuring the Navigation

The task model can give useful information on how to design the dynamic behavior of the user interface of the hypermedia document. For example, by analyzing the structure of the task model, we can understand the type of selections and possibilities available to the user at the beginning of the session (tasks on the left side of the task tree). Tasks at the higher level of the task tree are usually those indicating how to structure a user session, whereas tasks at the lower levels indicate how access to the detailed units of information can be performed.

Depending on the temporal relationships among tasks we can decide when some interaction techniques should be available to the end user and when links supporting the hypermedia navigation should be included.

If a task disables another task, an anchor supporting the disabling task should be inserted. For example, in the task specification in Figure 2, the AccessGenInfo task is disabled by the StartSession task. When information related to the first task is presented, a link allowing performance of the second task and disabling of the first one should be available for the end user.

If two tasks have a sequential enabling relationship, the presentation of information associated with the second task is automatically activated only after termination of the first task. For example, if the first task allows the user to start an application session (StartSession task), and the second task allows the system to show a set of defined profiles (ShowProfiles task), and there is a sequential enabling relationship between them, then possible user profiles will be presented immediately after the accomplishment of the first task. This sequential constraint can be supported in other ways within the same presentation unit (for example, graying out the interaction techniques associated with the disabled task or just making them inoperable to user interactions).

Using a choice operator means that the availability of the interaction techniques to support the performance of the choice depends on whether the choice is made by the user or the application. If the choice is made by the application, only information related to the task chosen will be presented. If the user makes the choice, at least the first actions of the possible tasks are enabled, for example, the selection of a link associated with each possible choice. Then, depending on the user's choice, the information and interaction techniques associated with the chosen task (and, in some cases, tasks that should be performed immediately after) will be made available, and the information associated with other choices will disappear. For example, the user can decide to get information about the town or about the museum. Each choice is supported by a specific link. Depending on the choice, only town-related or museum-related information will appear.

If two tasks are concurrent (the first action of one task can be performed before the last action of the other task), the related presentation techniques can be available to the end user, taking into account the criteria indicated in the earlier section "Structuring the Presentation."

Structuring the Information

Using a top-down analysis of the task tree it is possible to determine the structure that must be used for multimedia data. The tasks at the high and intermediate levels are useful for beginning to structure the data of a hypermedium. We use high-level tasks to identify the entity types of the hypermedium; specifically, we consider the abstract tasks and the objects that they manipulate. The basic idea is to associate the entities of the hypermedium with the objects relevant to the high-level

tasks. How to structure the entities into components depends on the subtasks at the lower levels and the objects that they need to manipulate.

In structuring the hypermedium data, requirements raised by the task models of all the user profiles must be satisfied.

In our current implementation we have as entities Work, Sculptor, Town, Museum, Historical Period, Type of Work. Each entity is further broken down into components. For example, the components of the Work entity include a general description, the techniques used, location, the state of conservation, possible restorations, bibliography, and picture.

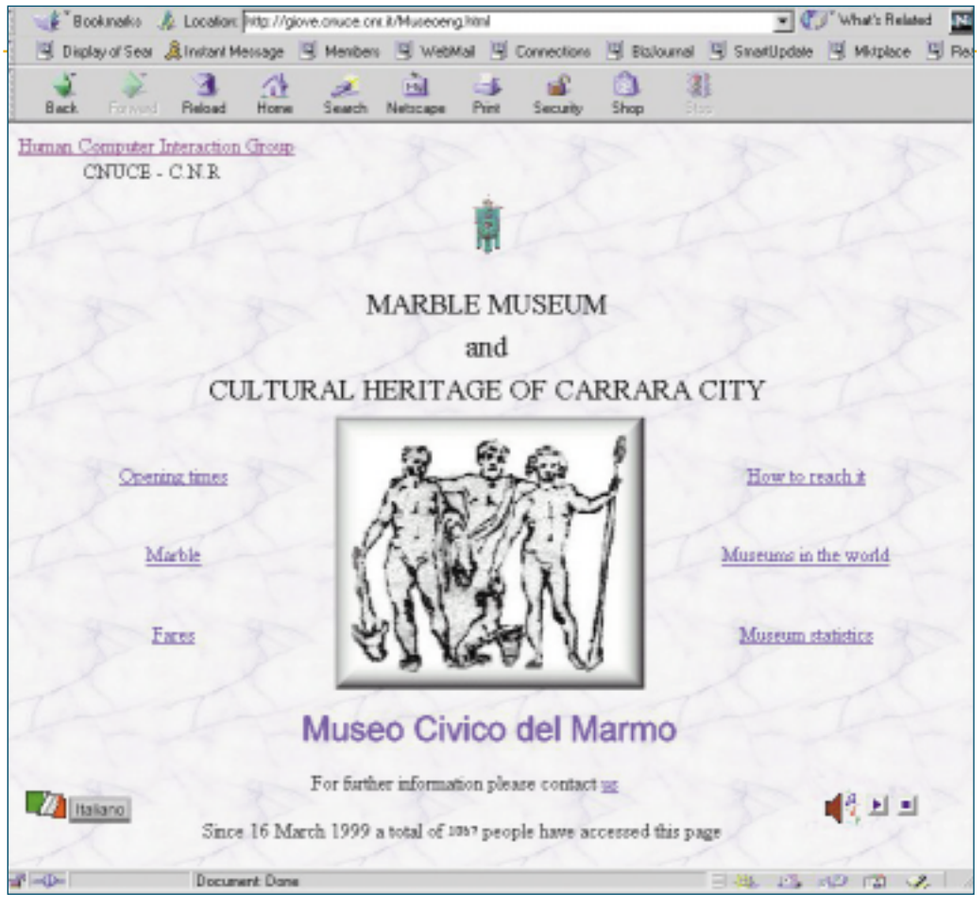

Figure 3: The initial page of the resulting hypermedium.

Adaptability Supported

Figure 3 shows the initial page of the resulting hypermedium (available at giove.cnuce.cnr.it/Museo.html). It supports a few tasks that are independent of the user profile—the choice of language and some general information on the museum. Neither of these involves access to the information about works of art (opening time, how to get to the museum, fares).

Once the museum is selected the user is asked to choose one of the possible profiles. Users can request information about the profiles available. They will then be able to navigate in the hypermedium using the navigation styles associated with the user profile chosen.

We developed three task models—one for each main type of user identified (tourist, student of art, and expert). They mainly differ in three aspects:

- Initial access to museum information. The expert can directly make specific requests, whereas the student can access information only by lists indicating the information available. Tourists access information mainly by spatial representations of the museum and the town.
- Presentation of the information related to the works of art. This model takes into account the basic knowledge of the type of users and the different tasks that they are likely to wish to perform.
- Navigation in the hypermedium. This model is more structured and preordered for tourists, whereas more navigational freedom has been given to expert users.

As we mentioned earlier, different views of the same information are possible depending on the type of users. For example in the lower part of Figure 4, we can see the tourist's view of information about a sculpture. As you can see, it is possible to be guided in the naviga-

tion. A user can go to the next work; in this case, that means the next work in the Modern Sculpture Section of the museum. The user can also gain access to the list of works made using this material, the works performed using different materials (wood, bronze, and others), and the museum map that drives the visit of the tourist in the museum hypermedium.

An expert user (upper part of Figure 4) can gain access to the information more immediately (for example, by just giving the name of the author), to more detailed information (for example, precise dimensions and precise date of creation), and to further information on the material, author, biography, or other information. The expert can go immediately back to her initial page to request something completely different.

A user can change the current user profile (tourist, student, expert) interactively during the session and on the screen, they will have feedback on the current profile at the bottom right side of the presentations. Thus a user can have different views of the information available and different navigation styles without having to start a new application session.

In order to check and optimize the design of the hypermedium achieved using a task-based approach, we organized an empirical testing of the hypermedium.

The issues detected by empirical testing were specific and relatively minor. This means that our method, aiming at guaranteeing a design consistent with the task model, succeeded in providing a usable hypermedium. Indeed, developing the task model with strong user involvement guaranteed a final design that was easy to interpret and understand, even by end users without experience in interacting with software applications.

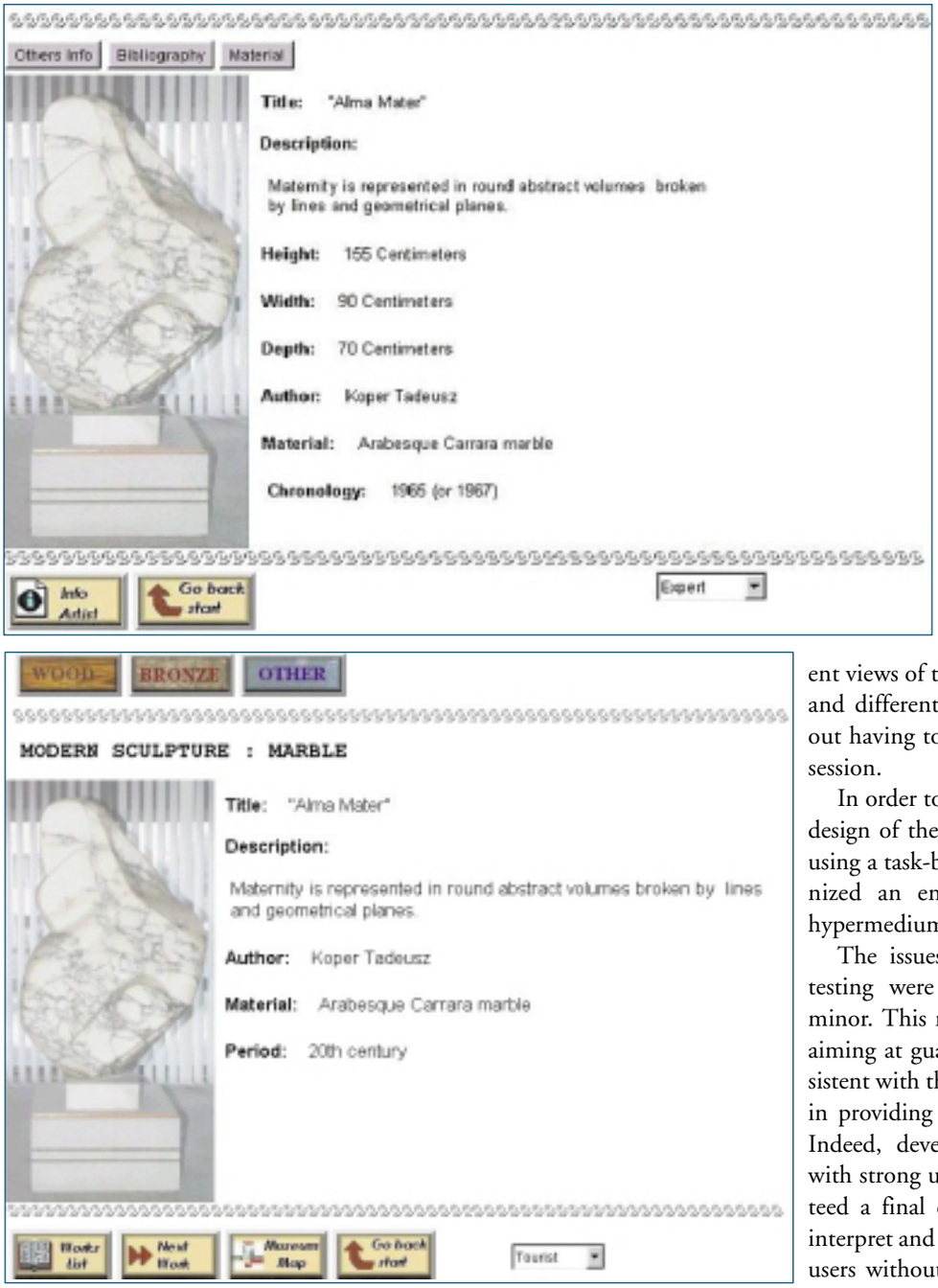

Figure 4: Example of different views of the same information.

Conclusions

In this paper, we have discussed a method that aims to support the designer in developing an easy-to-use hypermedium and its application in a hypermedium containing museum information. The support is based on task models that should reflect the users' view of the activities to perform. The task models are the result of an interdisciplinary discussion involving the relevant skills required in designing interactive applications. In this way the craft and the intuition of the designer are augmented to guarantee a design consistent with the task model.

We have considered that different types of users can perform different tasks while gaining access to the same information. We have therefore structured the resulting hypermedium in order to adapt its presentations and possible navigation to better support these different needs. We have shown a sample application of the proposed method to a hypermedium containing museum information.

Further work has been dedicated to improve the dynamic adaptivity of the resulting hypermedium to different users (Paternò and Mancini 1999).

Acknowledgments

This work was supported by the GUITARE R&D ESRIT Project (giove.cnuce.cnr.it/guitare.html).

References

1. Bodart, F., Hennerbert, A., Leheureux, J., and Vanderdonckt, J. 1994. A model-based approach to presentation: a continuum from task analysis to prototype. In *Proceedings of Design, Specification and Verification of Interactive Systems '94*, Springer Verlag, Berlin, pp. 77–94.
2. Fraisse, S., Nanard, J., and Nanard, M. 1996. Generating hypermedia from specifications by sketching multimedia templates. In *Proceedings of ACM Multimedia '96*, pp. 35–363.
3. Garzotto, F., Mainetti, L., and Paolini, P. 1995. Hypermedia design, analysis, and evaluation issues. *Communications of the ACM 38*, 8 (August), pp. 74–85.
4. Narayanan, H. and Hegarty, M. Model-based design of hypermedia presentations. Late Breaking Results, ACM CHI '97, pp. 355–356.
5. Paternò. 1999. Model-Based Design and Evaluation of Interactive Applications. Springer Verlag, London.
6. Paternò, F. and Mancini, C. 1999. Engineering the design of usable hypermedia. *Empirical Software Engineering* 4, pp. 11–42.
7. Paternò, F. and Mancini, C. 2000. Effective levels of adaptation to different types of users in interactive museum systems. *Journal of the American Society for Information Science*, 51,1. John Wiley & Sons, pp. 5-13.
8. Puerta, A., Cheng, E., Ou, T., and Min, J. 1999. MOBILE: User-centered interface building, In *Proceedings of CHI '99*, pp. 426–433.
9. Schneiderman, B. 1997. Designing information-abundant web sites: issues and recommendations. *International Journal of Human-Computer Interaction 47*, pp. 5–29.
10. Szekely, P., Sukaviriya, P., Castells, P., Muthukumarasamy, J., and Salcher, E. 1995. Declarative Interface models for user interfaces construction tools: the Mastermind approach. In *Proceedings of Engineering for Human-Computer Interaction '95*, Chapman & Hall, pp. 120–150.