# Decoupling Geographic from Conceptual Information in Physical Hypermedia Models

**Silvia Gordillo[1,4], Gustavo Rossi[1,5†], Robert Laurini [2], Daniel Schwabe[3]**

*[1] Lifia. Facultad de Informática. Universidad Nacional de La Plata, Argentina.*
*[2] LIRIS, INSA-Lyon, France.[3]Depto de Informática, PUC-Rio, Brazil*
*[4]Also at CICPBA, [5]Also at Conicet*
*{gordillo,gustavo}@lifia.info.unlp.edu.ar, Robert.Laurini@insa-lyon.fr, dschwabe@inf.puc-rio.br*

## Abstract

*In this paper we address how to manage geographical information in physical hypermedia applications, i.e. those mobile applications in which real and digital objects are linked using the hypermedia paradigm. We briefly introduce the problem domain (physical hypermedia) and motivate our research with a simple example. We next present the core of our approach, and extension of the Object-Oriented Hypermedia Design Method (OOHDM). We analyze the nature of physical objects and show how to separate their geographic (or spatial) attributes, relationships and behaviors and propose a composition approach. Navigational aspects involving geographical features are finally presented. We compare our approach with other related work and conclude describing further research we are pursuing.*

## 1. Introduction and Background

The idea of physical hypermedia (PH) was introduced in [4] and later extended in [5] and [6] as a novel way to build ubiquitous software, in which real world objects has a physical presence. Physical objects are considered nodes in a hypermedia network and can be related with each other using the well-known hypermedia metaphor. In this way a mobile user can not only receive additional information about the real object he is facing, but also he can explore related objects by navigating as he used to in the WWW. When a link points to a pure digital object, navigation proceeds as usual, but when the target is another real object, he must "walk" the link to access this new object [6]. The application can show the user how to reach the object either by presenting him a map or by explaining the path to the object of interest. This simple explanation shows the need to manage geographical information, both about the actual position of the user, the location of the hypermedia nodes (those which correspond to real objects), the (spatial) relationships between objects, the maps involved in a navigation path, etc. As a motivating example suppose a mobile visitor in an archaeological site (for instance Pompeii or Herculaneum) with a handheld device. Around all objects of interest, a sort of influence sphere can be defined. Similarly, if we can sense the user's position, a sphere can also be defined around him, so that when he is near a place he can get information on that place as shown in Figure 1. Each object of interest (for instance a statue, a temple, a painting, etc.) can be considered a node in a physical hypermedia. When the *m*-visitor is within the sphere of influence of an object, the corresponding node is opened. This node not only will give the definition and the characteristics of the object itself, but also will allow navigating to other physical or digital objects. For instance, the node of a building will allow accessing to an image of the building at different historical periods (to see the evolution, or the reconstitution by computer-generated images), or to similar buildings in other excavated sites.

In this paper we emphasize the need to clearly separate geographic aspects from other application concerns to improve modularity, ease of maintenance and reuse.

We base our presentation in an extension of the Object-Oriented Hypermedia Design Method (OOHDM) [10] to support modeling of PH applications described in detail in [9]. Our contribution

† Partially funded by Secyt Project, PICT 13623

Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05)
1529-4188/05 $20.00 © 2005 IEEE

is twofold: first we indicate which geographic issues are relevant for these applications; besides we show a modular approach to separate and compose spatial information that can be easily applied to other kind of mobile or ubiquitous applications.

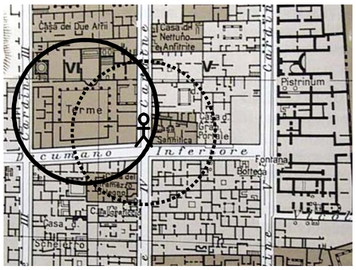

**Figure 1. *m*-visitor in an archaeological site**

The rest of the paper is organized as follows: In Section 2, we briefly introduce the OOHDM approach to PH; in Section 3 we stress the geographical aspects of PH software and present our proposal for dealing with these aspects separately. In Section 4, we discuss some related work and in Section 5 we present our concluding remarks and further work.

## 2. The Overall Design Approach

A PH application differs from a conventional hypermedia or Web application in basically three aspects: the fact that users interact with real world objects, the need to sense the user's location to determine whether he is within interaction range of a physical object, and the nature of "walking" navigation. For the sake of conciseness we ignore other aspects related with the user context that make PH an interesting kind of context-aware software.

### 2.1. The OOHDM Design Approach

OOHDM involves five activities: requirement acquisition, application modeling, navigation design, abstract interface design and implementation. After requirements are elicited, we describe the application classes and their relationships using UML [12]. For each user profile, we can define a different navigational structure, which will reflect objects and relationships in the conceptual schema according to the tasks this kind of user must perform. The navigational structure of a Web application is defined by a schema, containing navigational classes such as nodes, links, anchors and access structures (like indices) with the standard hypermedia semantics. Nodes are views on conceptual objects, while links reflect relationships in the intended domain. Static user profiling is useful to specify applications customized to particular users groups, i.e. showing certain specific features and relationships of the underlying domain (e.g. the view of an archeologist is different from the view of a tourist).

The abstract interface model defines which interface objects the user will perceive and which interface transformations will take place. A particular implementation of this design model takes into consideration particular aspects of the run-time environment. Obviously, mobile devices with small screens and wireless communications typical of PH software impose a set of constraints, not discussed in this paper.

We have extended OOHDM by adding the concept of physical objects, a simple user model and WLinks. Physical objects are those conceptual objects that possess a location. The user model contains sensed information about the user, the most relevant for this paper being the user's location. Finally WLinks (used when the target node is a physical object) are a kind of link whose traversal algorithm has been slightly modified; instead of allowing viewing the target object, they return either the location of the target, a map or other information specified by the designer. We next focus on the geographic aspects of these applications.

## 3. Dealing with Geographical objects

A physical object is an application object that can be explored "physically", i.e. it has a physical presence in the system and the user can be tracked if he is within interaction range of it. In our example, we may be interested in modeling statues, temples, monuments, and other buildings as physical objects.

Physical objects are characterized by an attribute *locator,* whose semantics depends on the location model being used, and operations to change location (if the object is mobile or displaced); *locator* can be just an identifier (e.g. if we use code bars or infrared sensing), or we might need a more complex representation. Physical Objects participate in spatial relationships (some of them generic and other application specific). These features clearly belong to a different concern with respect to other object's attributes (such as images of a temple, historical data, etc) and relationships (a monument belonging to the same time period, for example). We next describe how we manage these concerns separately.

### 3.1 Separating Geographic aspects

In our design approach we build two different models: a geographical model addressing spatial properties and relationships and a conceptual one that focuses on other application concerns as shown in Figure 2, and which are further integrated as described in Section 3.2.

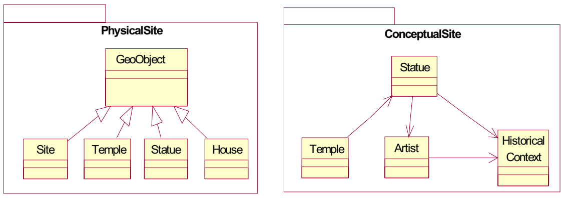


**Figure 2: Separating Geographic from conceptual issues**

While the conceptual model contains all application classes and relationships as shown in Figure 2 for the example of the archeological site, the geographical model only contains those application classes that must be treated as physical objects (i.e. the user can be sensed when being in their proximity), their properties and those spatial relationships among them (not shown in the Figure). Geographical classes are considered as sub-classes of the GeoObject abstract class as shown in Figure 2.

Sensing the user's position with respect to a geo-object and answering queries involving locations imply the definition of certain geographic features (such as the geometry and reference system in which objects are represented). In our approach we provide a basic set of locator models (symbolic and geo-referenced), reference systems and geometries, which are the most usual in this kind of applications. The (symbolic) locator of a physical object involves a value expressed either with a symbol (a URL, code bar, a string) or a geographic position expressed in a particular Geometry (point, line, polygon, etc.) and interpreted in one reference system (when the locator is geo-referenced). Each locator class possesses its own behaviors for reacting according to its nature. For the sake of simplicity we do not explain details of location interpretations, which have been well covered in the literature of geographic information systems.

Decoupling physical objects representation from their basic definition, as shown in Figure 3, allows using different locator models according to the object of interest, for example to combine outdoor, GPS-based location sensing, with indoor infrared or RFID-based sensing (associating strings or URLs to the object) or to easily change the location model of a particular object. Notice that for example we can use different models with instances of the same class according to technological restrictions or evolution. It is not necessary that all monuments are geo-marked with an absolute reference system; some of them may use a symbolic system.

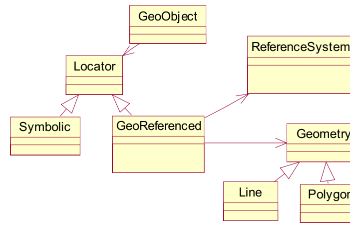


**Figure 3: Decoupling Locator from Geographic Objects**

Physical objects possess a default behavior that allows them to handle the event signaled by the user being within their interaction range, by opening (activating) the corresponding node. Additionally, they can inform how the user can reach them from any location; this behavior is triggered by "walkable" links to indicate how the user can "navigate" physically to the object. The object can either answer its absolute location, a plan or the route one must follow to reach it from the actual location. The designer must specify this last and eventually other behaviors as they are application and might be even instance dependent.

Geographical Objects are also related with each other with different kind of spatial relationships; some of them are generically defined in their abstract class as shown in Figure 4; others, application-specific, must be specified by the designer for each application. One of those relationships is the recursive composition that allows to represent geo-objects as composed of other geo-objects (a temple contains sections which them-selves contain rooms). Some of these parts might have a counterpart in the conceptual model (those which possess additional information of interest) and some will only be represented in the geographical model.

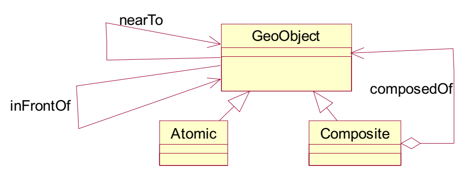


**Figure 4: Some Elemental Spatial Relationships**

Following the approach above we completely separate geographic from other application issues. The geographic model does not contain information that does not belong to this concern and reciprocally the conceptual model does not have to deal with spatial information. These two models can evolve independently and new geographic features can be added to the OOHDM meta-model without compromising the structure of the method.

Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05)
1529-4188/05 $20.00 © 2005 **IEEE**

### 3.2 Integrating both views

There are several strategies to compose design models (See Section 4.2). We chose to use the role abstraction to integrate the conceptual and geographic design models. Roles have been extensively used to model and integrate different points of views on the same reality both as conceptual modeling and design artifacts [11]. A role type (in this case a sub-type of the basic role GeoObject) indicates those properties and behaviors of an object when playing that role, i.e. when the object has a physical presence. Roles can easily be mapped to simple implementations either using the decorator pattern, or even Java interfaces. In Figure 5 we show how the integrated model looks like. Small boxes represent roles; in fact they are the geographical sub-classes of Figure 2, which relate with the conceptual ones, using the *role* abstraction. We omitted the role name for the sake of clarity.

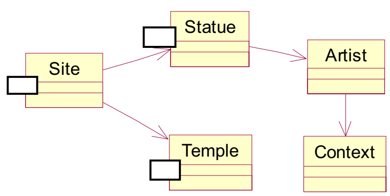


**Figure 5: Integrating models using roles**

### 3.3 Navigation Issues

There are basically two differences between conventional and physical navigation. First, nodes corresponding to physical objects are activated when the user is within the area of the object (while in conventional hypermedia nodes are opened by navigating links). The other difference involves WLinks. When we chose to navigate such links, by exercising an anchor in a (physical or digital) node, the link "just" informs us about the target destination, e.g. its location, how to reach it, etc. WLinks are implemented in OOHDM by changing the links default navigation behavior. Further explanation of this design structure is outside the scope of this paper.

Geo-based navigation as in physical hypermedia raises a set of challenging design (and implementation) issues. We next mention the most relevant ones that relate with geographic issues.

*(i)* - How to deal with the need to open several nodes at a time. This situation occurs (as shown in the example) when the user is in the interaction range of several objects. One possibility is to specify a slightly different "open" behavior consisting in the activation of an index pointing to the corresponding nodes (using "conventional" links)

*(ii)* - How to dynamically build WLinks according to some pre-defined criteria. Even though WLinks reflect conceptual relationships among objects in the application domain, it might be convenient to limit them in order to just connect "close" geographic objects. However the notion of "closeness" is fuzzy and it is necessary to express this condition precisely, according to the specific application, the geometries being used, and eventually the user profile.

*(iii)*-How to build geo guided-tours efficiently? OOHDM navigational contexts [10] allow specifying sets of nodes that fulfill some condition, e.g. being in the same street. By querying geographical information we can specify different types of tours according to the current user's position. Decoupling of geographic features allows us to keep this kind of specification modular.

## 4. Related Work

Separation of concerns has been the main driver of many projects in the software engineering arena; this topic has seldom been considered in the hypermedia and the geographic information systems domain. For the sake of conciseness we only focus on the GIS and Software Engineering fields.

### 4.1 Geographic Information Systems

The need to develop data models for dealing with geographic entities, relationships and behaviors has been early recognized by the GIS community [7]. Specialized data models such as [7], or extensions to well-known approaches like the entity-relationship model or the UML [8] have been devised. All these modeling approaches allow adding geographical features to application objects using well-known abstraction mechanisms such as generalization/ specialization or composition. Design models however are polluted with features that belong to different application concerns. For example a City class in [7] combines attributes and behaviors corresponding both to its intrinsic nature (being a city) with others corresponding to its use as a geographic entity. Even though some of these modeling approaches are powerful and expressive, tangling different requirements into the same class prevents extension, maintenance and reuse.

Our approach is quite simple from the point of view of the basic meta-model (so far, bare UML plus a set-of predefined geometries and reference systems), but involves a different philosophy with respect to the specification of geographic information. The engineering of the geographic features of the application can be done independently of other features (that may be themselves further decoupled). The integration process is simple, even when involving cross-cutting behaviors (See 4.2).

### 4.2 Object-Oriented Software Engineering

Many researchers have argued that the object abstraction is not enough to solve problems such as cross-cutting concerns, misalignment between requirements and designs and evolving behaviors. These problems have been addressed using Aspect-Oriented Programming [2], Subject-Oriented Programming and Design [3] and Role Modeling [11]. Aspect-orientation has focused mainly on non-functional requirements, such as persistence, caching, security, etc. Subject-Oriented Programming has been first used at the programming level and more recently for aligning requirements with designs. Our work is grounded on the ideas of Subject-Oriented Design, but given the nature of geographic features that are in general orthogonal to conceptual ones, we can rely on the role construct as an integration approach. To our knowledge, subject and role orientation have not been used to decouple spatial features so far.

## 5. Concluding Remarks and Further Work

In this paper we have shown how to deal with geographical issues in physical hypermedia design models. We have presented an approach for decoupling the specification of these aspects from other applications concerns. We have shown how to separate and how to integrate partial design models in a way that resembles well-known techniques of subject and aspect-oriented design. We are currently working in several research lines; first we are improving our notation to make it more "standard" by using UML extension mechanisms and defining stereotypes for physical objects and WLinks. We are also studying the integration of different modeling dimensions in the same conceptual model. For example, there may be relationships between physical objects that go beyond the conceptual ones, such as those involving proximity, vicinity, or other spatial relationships. It is interesting to analyze how these relationships might influence the navigation schema, i.e. in which way the user can choose to follow spatial relationships, even though they do not contain strong application semantics. We are finally evaluating how to incorporate a complete user model in the style of Adaptive Hypermedia [1].

## 6. References


[1] Adaptive Hypermedia Home Page: http://wwwis.win.tue.nl/ah/

[2] Special Issue on Aspect Oriented Programming. Comm ACM, October 2001.

[3] S. Clarke. "Composition of Object-Oriented Software DesignModels" Ph.D. Thesis, January 2001, Dublin City University.

[4] K. Gronbaek, J. Kristensen, M. Eriksen: "Physical Hypermedia: Organizing Collections of Mixed Physical and Digital Material". Proceedings of the 14th. ACM International Conference of Hypertext and Hypermedia (Hypertext 2003), pp 10-19, ACM Press.

[5] F. Hansen, N. Bouvin, B. Christensen, K. Gronbaek, T. Pedersen, J. Gagach: "Integrating the Web and the World: Contextual Trails on the Move". Proceedings of the 15th. ACM International Conference of Hypertext and Hypermedia (Hypertext 2004), ACM Press.

[6] S. Harper, C. Goble and S. Petit: "proximity: Walking the Link". In Journal of Digital Information, Volume 5, 1. ttp//jodi.ecs.soton.ac.uk/Articles/v05/i01/Harper/

[7] C. Parent, S. Spaccapietra, E. Zimanyi: "Spatio-Temporal Conceptual Models: Data Structures + Spaces + Time. In Proceedings of ACM GIS'99. pp 26-33, 1999.

[8] R.Price N. Tryfona, C. Jensen: Extending UML for Space- and Time-Dependent Applications. Advanced Topics in Database Research, Vol. 1 2002: 342-366

[9] G. Rossi, S. Gordillo, D. Schwabe: "Separation of structural concerns in physical hypermedia models". To be published in Proceedings of CaiSE 2005, LNCS forthcoming.

[10] D. Schwabe, G. Rossi: "An object-oriented approach to web-based application design". Theory and Practice of Object Systems (TAPOS), Special Issue on the Internet, v. 4#4, pp.207-225, October, 1998.

[11] Steimann, F.: On the Representation of Roles in Object-Oriented and Conceptual modeling. Data and Knowledge Engineering 35 (2000) 83-106

[12] UML home page: www.uml.org



Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05)
1529-4188/05 $20.00 © 2005 IEEE