Using Web-based Personalization on Spatial Data Warehouses

Octavio Glorio, Jose-Norberto Mazón, Irene Garrigós, Juan Trujillo
Lucentia Research Group
University of Alicante
PO BOX 99 E-03080
[oglorio,jnmazon,igarrigos,jtrujillo]@dlsi.ua.es

ABSTRACT
Spatial data warehouses (SDW) rely on extended multidimensional (MD) models in order to provide decision makers with appropriate structures to intuitively analyse spatial data. Several SDW development approaches provide a conceptual modeling and some guidelines in order to obtain logical schemas. However, there are two main drawbacks (i) spatial modeling is still complex for providing each decision maker with their own information needs, and (ii) SDW may be potentially large and spatial structures become increasingly complex to be analysed at a glance. Thus, representing and acquiring the required spatial information is more costly than expected and decision makers may get frustrated during the analysis. On the other hand, Web Engineering address similar problems (heterogeneous audience, different data sources and increasing amount and complexity of information) by using personalization rules. PRML (Personalization Rules Modeling Language) is a language that has been successfully applied to several Web systems in order to perform those personalization rules for every particular user and needs. Therefore, we have decided to use personalization rules and we have adapted the PRML to certain SDW aspects in order to introduce the right spatiality and deliver the correct information for every user needs. The great advantage of our approach is that each decision maker can easily include spatial data according to their own needs at conceptual level, while they can also conceptually get the right spatial schema instance avoiding exploring in a large and complex SDW.

Categories and Subject Descriptors
H.2.7 [Database management]: Database Administration – Data warehouse and repository; D.2.0 [Software Engineering]: General – Standards

General Terms
Design

Keywords
Spatial Data Warehouse, Geographic Information, Personalization

1. INTRODUCTION
Many years of data collection in very different systems and domains have generated significant volumes of heterogeneous information. Information that not only have to be stored, although it is required in analysis processes. Data warehouses (DW) are systems that store and explore huge quantities of information from many data sources. Therefore, these systems are suitable to cover current information requirements. For this purpose, the data is structured following the multidimensional (MD) model by organizing data into facts (containing the measures of the analysis) and dimensions (containing the contexts of analysis). These structures are more intuitive for designers and faster for exploring tools. The most typical tools for interactively explore the data warehouse are the BI (Business Intelligence) systems which include different analysis techniques such as OLAP (On-Line Analytical Processing) or data mining.

Data warehouses repositories also contain a lot of spatial data that are not used to their full potential and part of their richness is simply left out. In this paper we have considered spatial data which one whose represent geometric characteristics of a real world phenomena by an absolute and relative position, an associated geometry and some descriptive attributes. For example, traditional data fields such as customer or stores locations are represented by its address in an alphanumeric text. Due to this representation, there are some limitations while analysing data and interesting insights could be lost. An unresolved scenario could be if the decision maker wants to relate sales with the proximity between customers and stores or between stores and highway exits. Spatial Data Warehouses (SDW) address these limitations and integrate spatial data in their structures by extending the MD model in order to add spatiality at the conceptual level. In this paper, we have used our MD model [16] and its spatial extension [10], for further information we refer reader to these works.

Nowadays the wide diffusion of electronic devices containing geo-referenced information, volunteered geographic information activities (e.g. Wikimapia, OpenStreetMap), public initiatives (e.g. Spatial Data Infrastructures, Geoportals) and private projects (e.g. Google Earth, Microsoft Virtual Earth, etc.) generates a great production of spatial data. This overabundance of spatial data, in many cases,
does not help the efficiency of decision processes. Spatial modelling, analytical techniques and geographical analyses are therefore required in order to analyse data and to facilitate the decision process, with a clear identification of the geographical information needed.

Current SDW development approaches [5, 9, 12, 21] provide a conceptual modeling of spatial schemas, some guidelines in order to obtain logical schemas and querying languages over those schemas. Some of the most well-known approaches are cited in the related work in Section 2. However, none of them provide mechanisms to easily introduce just the required spatiality into the MD schemas according to each decision maker needs. But also, SDW may be potentially large and spatial structures become increasingly complex to be analysed at a glance and they do not consider mechanisms that allow to present the correct information for each user. As a consequence, representing and acquiring the required spatial information is more costly than expected and decision makers may get frustrated during the analysis. For example, a decision maker could only be interested on customers living near a store. In order to cover this spatial need, s/he has to address two complex tasks (i) deal with a great quantity of spatial data in order to introduce the necessary one, and (ii) deal with exploring a SDW instance potentially large in order to find the required data.

On the other hand, Web Engineering address similar problems, those are heterogeneous audience, complex data spaces and increasing amount and complexity of information. To overcome these problems, personalization processes have been intensively used in this area [15]. We understand personalization like the process of adapting the system to certain user-related information (e.g., user’s goals, needs, characteristics, behaviour, context) by using personalization rules. Therefore, due to the similarities, we argue that the multidimensional model should be personalized in order to add the required spatiality and to provide spatial customized views over the original data warehouse for every particular user and needs, thus better satisfying decision makers. In order to perform personalization rules for SDW, we have decided to use PRML (Personalization Rules Modeling Language). PRML is a language that has been successfully applied to several Web systems in order to perform personalization rules.

Therefore, this paper presents a modeling approach for SDW personalization at conceptual level (see Fig.1) by providing two new design artifacts together with the multidimensional models: (i) a spatial-aware user model which captures all the user-related information needed to personalization, and (ii) a spatial PRML extension in order to deliver the set of personalization rules which specify the required personalization actions. These new artifacts allow to personalize the multidimensional model with spatial data tailored to each decision maker. The great advantage of our approach is that each decision maker can easily include spatial data according to their own needs at conceptual level, while they can also conceptually get the right spatial schema instance avoiding exploring in a large and complex SDW. Conceptual design also avoid several system problems like difficult maintenance, no independence of the target platform, evolution of the information requirements, etc.

The remainder of the paper is structured as follows. Related work is reviewed next. Section 3 presents a modeling example to motivate our approach. Section 4 presents the proposed modeling approach for SDW personalization. A sample application of this approach is described in Section 5. Finally, conclusions are given in Section 6, together with a summary of our expected future works.

2. RELATED WORK

There are many works related to conceptual modeling for SDW systems. Some of the most well-known approaches are following cited. In [21], Bimonte et al. propose a multi-dimensional model (GeoCube) which integrates geographic information and ensure correct aggregation over this kind of measures. Then, they present GeWOlap [2], an extended model that support a Web implementation of spatial OLAP tools. Finally in [1], they propose the extension of the traditional spatial dimensions to support complex map navigation. Gomez et al. [12] also define a formal model for representing spatial data and a MD analysis tool named Piet [6]. Malinowski et al. [17] integrates spatial dimensions, measures, facts and levels in one concise model supported by a diagram editor named MADS. Also introduce into this model topological types that describe spatial hierarchies. Then in [18], they provide some guidelines to obtain implementations of spatial data warehouses using MADS for modeling.

Fidalgo et al. [5] propose a formal model named GeoDWFrame for guiding the designing of a Geographical Data Warehouse (GDW). Then, some extensions were made to this framework by the authors where a formal metamodel for GDW is defined [22] and a set of aggregation functions for spatial measures where developed [3]. Finally, they have presented a geographic and multidimensional data cube meta-model and a query language named GeoMDQL [4], which allows simultaneous usage of both multidimensional and spatial operators. In [9], we have proposed a framework for SDW developing aligned with the Model Driven Architecture (MDA) standard. In [10], we have added some geographic elements and an algebra to this framework in order to introduce geographic information external to the domain. Finally in [11], we have combined the spatial multidimensional analysis with data mining techniques at conceptual and logical level. However, none of them provide mechanisms to easily introduce just the required spatiality into the MD schemas according to each decision maker needs. But also, SDW may be potentially large and spatial structures become increasingly complex to be analysed at a glance and they do not consider mechanisms that allow to present the correct information for each user.

On the other hand, Web Engineering domain address similar problems such as heterogeneous audience, complex data spaces and increasing amount and complexity of information. We have tackled this scenario in [7] by using the novel PRML language in order to perform a personalization rules. The PRML is a rule-based high level language created to specify personalization upon Web applications. PRML is based on a MOF (Meta Object Facility) metamodel [20] and could be extended. Indeed, in [8], we have adapted it to peculiarities of OLAP systems such as complex operations required to analyse data.

To the best of our knowledge, spatiality has not been considered in DW personalization issues. Present work overcomes the aforementioned drawbacks since our personalization rules are able to (i) easily include the required spatiality in the MD structures for each user, and (ii) conceptually get
the correct spatial schema instance for each user avoiding exploring in a large and complex SDW.

3. MOTIVATING EXAMPLE

To show the advisability of our approach we define the following sample scenario: a sales department of a company is initially interested in analysing who bought (Customer), where (Store), what (Product) and when (Time). Assume the multidimensional model of Fig. 2 defined in a class diagram by means of the UML profile for multidimensional modeling presented in [16].

Sales are represented as a Fact class \(F\) and the contexts of analysis are represented as Dimension classes \(D\). Measures for Fact classes (i.e., UnitSales, StoreCost, and so on) are represented as FactAttributes \(F_A\). With respect to dimensions, each level of a dimension hierarchy is specified by a Base class. Every Base class \(B\) contains a number of descriptive attributes \(n\). Associations between pairs of Base classes \(B\) represent aggregation paths: role \(r\) represents the direction in which the hierarchy rolls-up (i.e., aggregating data in a coarser level of detail), whereas role \(d\) represents the direction in which the hierarchy drills-down (i.e., disaggregating data in a finer level of detail). In this example we only have expanded Store dimension to simplify Fig. 2.

Suppose now that the sales department is only interested in customers living near an airport in order to start a specific promotion. The multidimensional model for sales analysis (see Fig. 2) could not cover this spatial need. We overcome this scenario by using our approach and the final geographic multidimensional model (GeoMD) obtained after applying some spatial personalization rules (see Section 5) is shown in Fig. 6. It has been also defined in a class diagram by means of the UML profile for geographic multidimensional modeling presented in [10]. There, Stores are represented as a SpatialLevel class \(S\) and the airports are represented as Layer class \(L\). We have added also the Train Layer class in order to show later more complex spatial rules.

4. OUR APPROACH FOR MODELING SPATIAL PERSONALIZATION

Spatial Personalization can be influenced by several factors. We have based our considerations in a novel classification made on [8]. Therefore, our proposal considers (i) user-specific characteristics (independent of the domain), (ii) spatial user-behaviour in order to derive the preferences or interest on different elements of the system, and (iii) the changing spatial user-context in order to define personalization strategies. The structure of the data required for personalization is specified in a user model. This model has been also defined in a class diagram by means of a UML profile named SUS (Spatial-aware User model), as it will be explained throughout the next sections.

On the other hand, we can consider two different types of spatial personalization actions. Personalization can be applied over the schema of the SDW (e.g., becoming a level a spatial one if it is required), and it can also be applied over the instance of the SDW (e.g., selecting certain fact instances according to a geographic condition). PRML is a rule-based high level language created to specify personalization upon Web applications. Due to the Web area similarities (heterogeneous audience, complex data spaces and increasing amount and complexity of information), we have decided to use PRML. We have adapted this language to the peculiarities of SDW systems (such as complex spatial operations required to analyse data), as it will be explained throughout the next sections.

The SDW personalization process it is shown in Fig. 1, the designer starts building a MD model and defines some Spatial Schema Rules in order to add the required spatiality in the MD structures. Finally the Geographic Multidimensional Model (GeoMD) obtained is personalized using Spatial Instances Rules by applying different spatial PRML expressions.

4.1 Spatial-aware User Model

Personalization is a user-centered process, therefore, user modeling is the basis for personalization support [13]. In order to provide personalized spatial data warehouses, relevant knowledge about the decision maker should be captured. The structure of the data required for personalization is specified in the user model. This model should be defined based on the personalization requirements we want to support in a concrete system. The information specified in the user model builds the user profile and will be updated during the lifetime of the system. The information stored in the user model typically contains data related to the user (e.g., user characteristics like age or language, user geographic location, etc) and may also contain information related to the domain (e.g., preferences over data). Therefore, based in [8], we have considered the following criterias in order to take in consideration the spatial data on data warehouses:

Location Context:

This information characterize the surrounding environment on which an analysis session is performed. It refers to information relative to the geographical location of the
context of analysis. For example, a user may need only sales made in stores at less than 5 km of his location. Therefore, his geographic location is required to filter the stores that satisfy such condition (see Section 5).

Spatial Selection:
We can track the user spatial behaviour in the system and infer the interest or preferences he/she has on certain elements. For this purpose we store information about the spatial operations performed by the decision maker. For example, a user has enough interest in stores near airports then we also add the stores not so near but with train connection to the airport. To cope with this requirement, the first step is to gather the user interest in cities near airports (see Section 5).

The user model is represented by means of a UML profile in a class diagram. Several stereotypes have been defined in this UML profile as shown in Fig. 3.

A sample of a spatial-aware user model defined for the motivating example (see Section 3) is shown in Fig. 4. As aforementioned, in this model we store different information needed to fulfil the personalization requirements initially specified for the SDW. The specified requirements for this example are the following:

- The regional sales manager needs to analyse sales in stores near an airport. To fulfill this requirement we have to store the decision maker role in the user model as we can see in Fig. 4.

- If the decision maker has enough interest in cities near airports then we also add the cities not near enough but with a good train connection. To cope with this requirement we need to acquire the decision maker interest on this spatial selection. For this purpose we have the AirportCity class in the user model. This class represents a spatial selection event triggered by the user behaviour and stores the number of times it is performed.

All the required information is stored in the user model. When the information has to be gathered at runtime (e.g., user location, user interest) a PRML rule to update the user model is defined. Moreover personalization rules are needed to specify the actions needed to fulfill the personalization requirements. Some of these rules are defined in Section 4.2.
4.2 Spatial Personalization Model

After specifying the structure of the data needed for personalization, the designer should define the personalization actions to apply to the SDW system. As aforementioned, we define the personalization model by a set of Event-Condition-Action rules. The rules express the following: when an event is triggered if a condition is fulfilled an action is performed.

Fig. 5 shows (an excerpt of) the metamodel for the PRML language adapted for SDW systems. It defines the set of constructs of the language such as the different parts that form a PRML rule and the different events, actions and expressions supported. It is worth noting that the PRML metamodel can be extended if a new kind of element is detected. The main element of the metamodel is the Rule metaclass which represents the concept of rule containing the elements that define it. The elements defining a rule are the ones that represent its main structure and are explained along the following sections.

4.2.1 Tracking Events

The interest of the user in certain spatial elements is the main tracking event. For this purpose, we have defined a new event named SpatialSelection. This event is triggered when a spatial expression (SpatialExp) is satisfied and some specific instances are selected (by using SelectInstance action). Therefore, this method has two parameters, the GeoMD element selected and the spatial expression. In such way, we can notice that the user is enough interested, for example, in cities near an airport. Following, we have shown an example of this situation:

\[
\text{SpatialSelection}(\text{GeoMD.Store.City.geometry, Distance}(\text{GeoMD.Store.City.geometry, GeoMD.Airport.geometry}) < 20\text{km})
\]

Besides the previously described event, other ones related to the analysis session should be considered. On the one hand, the start session event is triggered when the session is initiated by the user, whereas the end session event indicates the end of the analysis session.

4.2.2 Rule Conditions.

When specifying conditions, PRML rules can refer to different elements of the conceptual models in order to define condition expressions. As already explained, personalization is mainly based on the user model information. A mechanism to access the user model structures is needed. To access a certain element of a model, PRML navigates over the model using path expressions (PathExp). These expressions are based on the path expressions defined in OCL [20]. The PathExp to access information defined in the spatial-aware user model, always contain the prefix “SUS” and the source concept is always the user class, to identify the user that is actually analysing the data. As an example for a PathExp over the user model defined in Fig. 4, consider that we want to access the name of the decision maker. The expression would be SUS.DecisionMaker.name. Analogous to OCL, we can navigate through the model concepts by the target roles of the relationships between model elements. For example, consider that we want to access the role of the decision maker, the PE expression would be SUS.DecisionMaker.dm2role.name.

Furthermore, a PRML rule can also refer to information from the multidimensional model for specifying some needed conditions to define personalization actions. In the same way, in the personalization actions we may need to refer to an element of the multidimensional model to be modified. In this case the PathExp contain the “MD” prefix and the source concept of the PathExp is always the Fact class we want to access. In this case to navigate through the model elements we go over the Base classes and Descriptor attributes of the multidimensional model, for instance, to refer to the name on the State we use MD.Sale.Store.State.name. If we want to define a rule using a spatial element of the initial Geographic Multidimensional Model (GeoMD), we use the prefix “GeoMD”. For example, if we want to access to the geometric description of the Store we use GeoMD.Sale.Store.geometry. For more details about this MD model extension, we refer reader to [10, 11].

4.2.3 Spatial Expressions.

Once all the elements can be accessed, the condition expressions have to be defined. Originally, PRML only considered boolean expression because there were not spatial data. In this paper, we have added some new operators to PRML metamodel in order to define spatial expressions. This new operators are higher level spatial operations compliant with the ISO [14] and OGC [19] standards. In this way we ensure a final logical processing over an implementation.

The operators that we have considered are the traditional topological relations that returns a boolean value whether it is satisfied or not. Those are Intersect, Disjoint, Cross, Inside and Equals. Then, we have considered the Distance operator that returns a numeric value according to the distance between involved elements. Finally, we have added the Intersection operator that returns another geometric object depending on the involved elements and the order. For example, if we intersect LINE type with POINT the operator returns a COLLECTION type of sublines. However, if it is POINT intersecting LINE type the operator returns a
COLLECTION type of points. In Fig. 5 it is shown an excerpt of the adapted PRML metamodel including the new spatial operators.

4.2.4 Spatial Personalization Actions.

Personalization rules can contain three kinds of actions. First, as aforementioned, satisfying a personalization requirement may imply acquiring knowledge about the user at the runtime. For this purpose, an acquisition action (SetContent) has been defined to update the spatial-aware user model. Second, we have the actions that modify the structure of the DW. Either adding some geometric description to a previously multidimensional element (BecomeSpatial action) or adding new geographic elements (AddLayer action). These actions allow to add spatiality at conceptual level according to each decision maker needs. In order to describe the geometry of the new elements, the geometric types defined in the spatial-aware user model are used (GeometricTypes). Finally, we have the actions taken over the instances of the SDW (SelectInstance action). The great advantage of this action is when it is used in combination with spatial conditions. In such way, each decision maker could take advantage in order to satisfy their own required information even if the BI tool used for the analysis does not support spatial data. For example, a decision maker wants to analyse sales fact with an OLAP engine without spatial support. But, s/he is interested only on sales instances made in cities near an airport (spatial condition). Therefore, we can personalize the SDW to cover this need and when the OLAP session begins the spatial analysis have been done even if the analysis tool does not support spatial data processing. All these actions are described as follows:

- **SetContent(Property p, ValueSpecification v):** this action allows to update the value of a property p of the user model or the different multidimensional models. The new value v can be a literal or a formula.
- **SelectInstance(Variable i):** this action allows to select a specific instance i. It is mainly used for filtering the SDW by combining spatial and boolean PRML expressions. For example, the decision maker could select only the sales made less than 5 km from a hospital.
- **BecomeSpatial(Element e, GeometricType g):** this action allows to add a spatial description g to a MD element e. For example, in order to correlate the distance between Stores and Clients, we have to spatially describe both elements.
- **AddLayer(String s, GeometricType g):** this action allows to add new geographic data to the MD structure. These data is grouped in a thematic layer named s and geographically described by g. For example, in order to correlate sales with the distance between stores and highway exits, we have to add a thematic layer describing highways.

The next section shows the applicability of our approach by means of a sample scenario based on the multidimensional model described in Section 3.

5. SAMPLE APPLICATION

In order to show the applicability of our approach, this section defines a couple of sample situations in which spatial personalization rules are applied. First, we have defined a schema rule taken over the multidimensional model described in Section 3 in order to add required spatial data and obtain a GeoMD model. Then, we have defined an instance rule by using spatial operations over the GeoMD model obtained after the execution of the first rule. Finally, we have defined a more complex rule that track the user interest in certain spatial elements and add other geographically related data that could also be interesting for the user.

**Example 5.1 (Spatial Schema Rule).**
Different users may need to add new external spatial data in order to be introduced in the analysis. This should be the first rules to be applied in order to obtain a spatial MD model.
(GeoMD). For example, a regional sales manager needs to analyse sales in stores near an airport. Therefore, a personalization rule is required to add airport geographic layer to describe the position of every building but also the stores have to be spatially described. It is worth noting that the user role has been previously gathered from user requirements and stored in the spatial-aware user model.

Rule: addSpatiality When SessionStart do
If (SUS.DecisionMaker.dm2role.name = 'RegionalSalesManager') then
  AddLayer('Airport', POINT)
BecomeSpatial(MD.Sales.Store.geometry, POINT)
endIf
endWhen

This rule is triggered when users log in. If the user role is “RegionalSalesManager” then a Layer class Airport is created in the MD model and the Store Base class is stereotyped as SpatialLevel class. The POINT data type is correct to describe instances of both, airports and stores. In Fig. 6, it is shown the GeoMD model obtained after applying the rule to the MD model of Section 3. Airport is represented by Layer class and the Store level is represented as SpatialLevel class.

Example 5.2 (Spatial Instance Rule). Different users may need to select certain fact instances considering some geographical condition. For example, a user may need only the sales made in stores at less than 5 km of his location. Therefore, a personalization rule is required to filter the stores that satisfy such condition. It is worth noting that in this rule it is used data of the domain (stores location) and external to the domain (user location).

Rule: 5kmStores When SessionStart do
Foreach s in (GeoMD.Store)
  If(Distance(s.geometry, SUS.DecisionMaker.dm2session.s2location.geometry) <5km)
    then
      SelectInstance(s)
  endif
endForeach
endWhen

This rule is also triggered when users log in. Then, for every store, the distance to the user is calculated. If this value is less than 5 km, the store is selected. Therefore, all the succeeding analysis in any BI tool will have the sales fact instances only made in selected stores.

Example 5.3 (Spatial User Interest Rule). Interest of the users in different elements can be inferred by their data analysis behaviour. For example, we can add some instances to the analysis if the user has manifested enough interest in other related elements. Let define the following requirement: if the decision maker has enough interest in cities near airports then we also add the cities not near enough but with a good train connection. To cope with this requirement, the first step is to gather the user interest in the cities near an airport. For this purpose we define the interest of the user as the times s/he select cities near airports, and we store this information in the spatial-aware user model by means of the following rule:

Rule: IntAirportCity When
SpatialSelection(GeoMD.Store.City, Distance(GeoMD.Store.City.geometry, GeoMD.Airport.geometry)<20km) do
SetContent(SUS.DecisionMaker.dm2airportcity.degree, SUS.DecisionMaker.dm2airportcity.degree+1)
endWhen

This rule updates the user spatial selection degree of cities at less than 20 km of an airport when the user select instances that satisfy such spatial condition. Now, the personalization rule has to be defined to also select cities not near enough an airport but with a good train connection if the spatial selection degree is higher than a threshold defined by the designer.

Rule: TrainAirportCity When SessionStart do
If (SUS.DecisionMaker.dm2airportcity.degree >threshold) then
  AddLayer('Train', LINE)
Foreach t, c, a in (GeoMD.Train, GeoMD.Store.City, GeoMD.Airport)
  If(Distance(Intersection(Intersection(t.geometry, c.geometry), a.geometry))<50km) then
    SelectInstance(c)
  endif
endForeach
endWhen

When the user spatial interest exceeds the threshold, the Train Layer class of LINE type is added. Then, for every train line and if the line contains a city and airport points, the distance of the corresponding segment is calculated. If such value is less than 50 km the city is selected because it has a near enough train connection to an airport.

6. CONCLUSIONS AND FUTURE WORKS

Current SDW development approaches provide a conceptual modeling of spatial schemas, some guidelines in order to obtain logical schemas and querying languages over those schemas. To the best of our knowledge, they lack important issues: (i) spatial modeling is still complex for providing each decision maker with their own information needs, and (ii) SDW may be potentially large and spatial structures become increasingly complex to be analysed at a glance. Therefore, to overcome the SDW development limitations, this paper presents a modeling approach for spatial data warehouses personalization at conceptual level (see Fig.1) by providing two new design artifacts together with the multidimensional models: (i) a spatial-aware user model which captures all the user-related information needed to personalization, and (ii) a set of spatial personalization rules which specify the required personalization actions. The great advantage of our approach is that each decision maker can easily include spatial data according to their own needs at conceptual level, while they can also conceptually get the right spatial schema instance avoiding exploring in a large and complex SDW.

As a short-term future work we plan to integrate the approach in our model driven developing framework [9]. Moreover, we plan to extend this approach considering visualization aspects of the SDW mainly focus on spatial BI tools.
7. ACKNOWLEDGMENTS

This work has been partially supported by the ESPIA project (TIN2007-67078) from the Spanish Ministry of Education and Science and by the QUASIMODO project (PAC08-0157-0668) from the Castilla-La Mancha Ministry of Education and Science (Spain). Octavio Glorio is funded by the University of Alicante under the 11th Latin American grant program.

8. REFERENCES


