ISIS: A Semantic Mediation Model and an Agent Based Architecture for GIS Interoperability

Eric Leclercq Djamal Benslimane Kokou Yétongnon
LE2I - Université de Bourgogne
B.P. 47870, 21078 DIJON Cedex - FRANCE
{Eric.Leclercq,Djamal.Benslimane,Kokou.Yetongnon}@u-bourgogne.fr

Abstract

The diversity of spatial information systems promotes the need to integrate heterogeneous spatial or geographic information systems (GIS) in a cooperative environment. This paper describes the research project ISIS (Interoperable Spatial Information System) which is a semantic mediation approach to support GIS interoperability. Its key characteristic is a dynamic resolution of semantic conflicts which is adequate for achieving autonomy, flexibility and extensibility. We propose a spatial OO data model and a mediation architecture based on multi-agent paradigm tu support GIS interoperability.

Keywords: Geographic information system, data heterogeneities, multi-agent systems, semantic interoperability, context mediation.

1 Introduction

Geographic information systems (GIS) are used to manage spatial or geo-referenced data. Typically, GIS were designed as ad hoc monolithic information systems for specific applications. With the rapid development of networking technologies and WEB-type environments, a growing number of heterogeneous and autonomous spatial data sources are becoming avalaible to users. GIS interoperability has emerged as an important challenge in the design of complex applications in which collections of autonomous information systems cooperate to carry out tasks.

Of late years, a need to cooperate different GIS has gained importance and a lot of work has been done in the area of GIS interoperability [2, 4, 6, 7, 9]. Three main approaches have been proposed. Tyey correspond to the top four levels described by Bishr et al [2]: Platform level, Syntactic level and Application level interoperability. The platform level is concerned with hardware, operating systems and network protocols. The syntactic level provides functionalities and tools to define persistent and uniform views over multiple heterogeneous spatial data sources. Access to remote data sets is done via either a high level language or an application interface. The application level is concerned with providing seamless system interoperation in which users can access multiple GIS as if they were centralized or integrated spatial system without having to worry about data models, data location, and data semantics. At this level, three major approaches can be distinguished: federated, schema mediation and context (or semantic) mediation approach:

- The federated approach focuses on providing integrated global views over information systems, constructing integrated schemas to combine the information contents of component systems. Several authors have discussed extensions of traditional integration to handle spatial heterogeneities[4] and to represent spatial data transformation techniques [3, 9].
- The schema mediation approach which is based on the wrapper/mediators architecture and aims at extending many functionalities including common data models to incorporate spatial data types. For example, Amann in [1] describes a schema mediation approach that uses ODGM 93 as a common object model extended with spatial data types. Another example concerns the OASIS project where each GIS is seen as a persistent store for spatial objects. These objects are described in a common data model [8] which is based on the OpenGIS specifications.
- The context mediation approach is based on an explicit representation of the semantics of spatial data through the notion of context. A very little works has been done. Bishr in the SEMWEB project [2] explores this approach. He defines a context by a set of rules and constraints attached to object definitions. The concept of proxy context is used to mediate between two local contexts. Context comparison is achieved by semantic translators which enable users to query remote objects without knowing their semantics, localization or representation.

In this paper, we present the ongoing research project

ISIS (Interoperable Spatial Information System). It is a semantic mediation approach which aims to support GIS interoperability. In ISIS, the emphasis has not been placed on static integration methodologies in which export schema are integrated to resolve semantic conflicts, but rather on a dynamic resolution of semantic conflicts that is more adequate for achieving autonomy, flexibility and extensibility. The remainder of the paper is organized as follows. Section 2 is devoted to a brief description of ISIS's architecture. Section 3 presents the AMUN data model. Finally section 4 concludes the paper.

2 The Multi-agent system architecture

The architecture of ISIS is based on the concept of agents (see Figure 1). An agent is a software component which has a role to play in the functioning of the system. The degree of granularity is not equal for all agents: some play more important roles than others. Different types of agents are defined: wrapper, cooperation, ontology, global query processor, semantic router and user interface. We describe below each agent by its general functionality in the system, the services it provides and the data and knowledge it owns:

- Wrapper Agent (WA): it is used to encapsulate a local GIS in a generic spatial object server which has the capability of processing data retrieval request concerning the local GIS. The main role of a wrapper is to process OQL queries originating from its corresponding cooperation service. This consists of several steps: 1) translation of the OQL query/subquery to target local query/subquery language, 2) execution of the target query on a local GIS and 3) forwarding local results to cooperation service. To hide schematic heterogeneities, local schema associated with wrapper are represented by AMUN objects using the core concepts.
- Cooperation Agent (CA): it coordinates high-level query processing. It has knowledge about only one local GIS. This knowledge constitutes a cooperation schema (cooperation objects) which describes the semantics of objects contained in the wrapper schema. A CA agent processes global queries initiated by itself or sub-queries submitted by other CA agents. The queries are translated from the cooperation context to the local context by using definitions of cooperation objects to rewrite the query in term of local objects. The translated query is sent to the local wrapper to retrieve data from the local GIS.
- Ontology Agent (OA): it allows communication among different type of agents. In order to exchange queries without operating on global schema, all the CA agents use a common ontology which provides mutual understanding of the concepts used in submitted queries. The link between the ontological concepts and the cooperation objects are created by ontological agreements.
- Semantic Router Agent (SRA): it stores ontological agreements. A SRA's only goal is to provide to a CA agent the identity and location of other CA agents which can participate in the execution of a query.
- Global Query Processor Agent (GQPA): it takes as an input a query originating from a CA agent expressed in terms of ontological objects schema and uses ontological agreements obtained from SRA agent to: 1) identify relevant information sources and to 2) create an execution plan which is sent back to the CA agent.
- User Interface Agent (UIA): it is an interface between a user and a CA agent. Its role is to receive a query from a user, send the query to CA agent and to deliver query result to a user. Each user interface is connected to only one CA agent. The user interacts directly with only one interface even if query processing step required more than one CA agent.

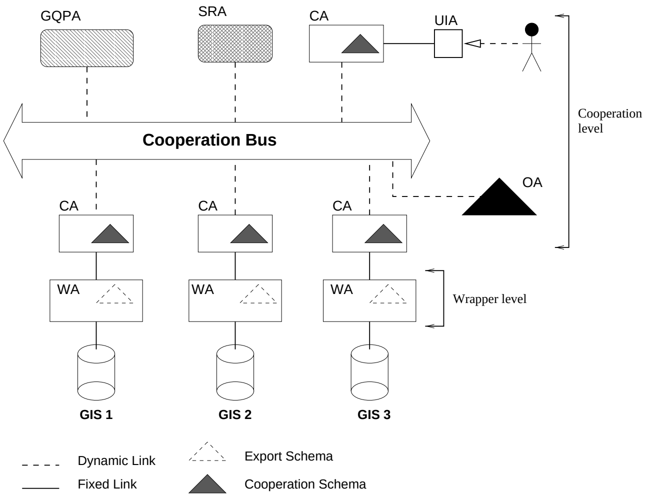

Figure 1. ISIS General Architecture

3 The AMUN data model

The data model AMUN that is used to represent information at both wrapper and cooperation levels. The primary intent of AMUN is to provide a set of concepts 1) to represent traditional textual information (thematic properties) and spatial information, 2) to define semantic contexts, 3) to provide a foundation for the resolution of semantic differences among different contexts and 4) to convert and transfer data objects between systems. We present in this section a brief description of AMUN.

3.1 Wrapper level

The wrapper level is used to describe local data sources in export schema. It comprises a set of core concepts which are used to represent real world entities, including spatial data types and object-oriented core concepts.

3.1.1 Spatial data types

The predefined spatial data types provided by AMUN are based on a subset of the spatial types of the OpenGIS specifications [3]. OpenGIS defines two basic geodata types: features and coverage. A feature type is used to represent real world entities, and a coverage type represents associations between points or polygons with a value (for example depth of a lake, wind speed over an area). In the current version of the project ISIS, we only consider geometric data types (feature types). Figure 2 shows the hierarchy of spatial data type used in the AMUN data model. `Geometry` which is the highest spatial type in the hierarchy represents general geometric information. `CoordinateGeometry` is a subtype of `Geometry`. It is used to model spatial objects that contain coordinate information. The lowest level of the hierarchy contains the basic spatial data types: `Point`, `LineString`, `Polygon`, etc.

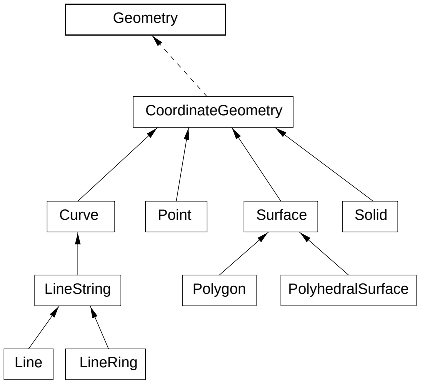

Figure 2. Spatial type hierarchy of Amun

3.1.2 Object-oriented core concepts

An object comprises a state which is defined by the values of its attributes and a behavior which is defined by a set of methods. The attributes of an object can be thematic or spatial types. Complex thematic or spatial types can be created by the usual set or tuple constructors. The values of attributes can be simple values, complex values or references to other objects. We denote the set of all the types by $\mathcal{T}$.

The specific spatial attribute Geo is included in the description of an object to model the spatial characteristics of the object. It can be an aggregation of features (set or tuple). For example a lake can have different geometric shapes, one form for each season.

Object classes organize objects into sets of similar entities that share the same structure and behavior. Let $\mathcal{C}$ denote the set of all the classes. A class $c \in \mathcal{C}$ is a tuple $c =< Name, AttList, MethList >$ where $Name$ is the name of c, $AttList$ and $MethList$ are respectively the list of the attributes and methods belonging to c. The IS-A (subclass) relationship is an acyclic relationship between classes. It states that if a class c is a subclass of another class c' then all the instances of c must also belong to c' and $AttList(c)$ is contained in $AttList(c')$.
Example: In the example GIS S_1 above, the entities farm worker and parcel can be represented at the wrapper level by the following classes: class `Parcel` contains the special spatial type Geo.

```
<Parcel,Parcel#: integer, Crop-
Type: string, Farmer: FarmWorker, Geo: POLYGON,
{Surface(): real, WOwnerName (Name: string),
ROwnerName(): string ... }>

<FarmWorker, Name: string,
Address: [City: string, Street: string],
BirthDate: date,Status: string,
Sex: char,{Age():integer ...}>
```

In AMUN, virtual classes represent (non materialized) views over one or more existing classes. They can be defined and used to: 1) restructure the values of objects, thus allowing multiple representations derived or calculated from the values of an object, 2) allow aggregation of information spread in different classes. As will be discussed in detail below, this is done by incorporating virtual classes into the definition of cooperation classes.

3.2 Cooperation level

The cooperation level is devoted to the resolution of semantic discrepancies among heterogeneous GIS. To achieve coordination between different schemas with heterogeneous semantics, we introduce the concept of context which can be used to express semantic information contained in schemas, and to record the assumptions under which schemas are designed. Two types of contexts are defined in the ISIS architecture: 1) reference context used to model common semantic of an application domain, 2) cooperation contexts used to interpret the common reference context in terms of local objects of sites.

3.2.1 Reference context

The reference context serves as a common vocabulary [5] (ontology), identifying and recording informations relevant to a particular application domain. It contains mediation classes which are defined by three types of descriptions: static properties, behavior (list of methods) and semantics. The semantics associated with a mediation class is value oriented, i.e. it is used to specify constraints or precise knowledge about possible values taken by an attribute. It can be:

- a domain value (an enumerated type) which spells out the set of values allowed for an attribute. For example, the type of attribute `CropType` of a `Parcel` can be precisely specified by $\{$wheat, corn$\}$.
- a semantic value which is used to express the meaning of an attribute. Typically, a semantic value describes units, coordinate systems or other quality or properties of an attribute. For example, a semantic value `acre` may be associated with the attribute `Surface` of `Parcel` to state that the surface is measured in acres, or to express that the nitrate concentration is expressed in milligrams per litre.
- a logic expression that represents knowledge assertion or a constraint. For example, to state that parcels cultivated with wheat are grain parcels, a semantic rule is defined: `GrainParcel(X)=>Parcel(X) and (X.CropType="wheat")`.

A formal definition of mediation class is given in [7].

Example: The following mediation classes describe a part of an agricultural ontology: 1) a class `Person` to describe a person by name, birth name, sex, birth date. It includes a method age. A person is a male or a female so the domain associated with attribute sex has a type string restricted to an enumerated set that contains two possible values and 2) a class `Parcel` to describe cadastral parcels with parcel identifier, owner and address.

Two sub-classes of `Parcel` are defined. `GrainParcel` is a specialization of `Parcel`, it inherits all the attributes of `Parcel` and add a crop type. `IndustrialParcel` is a `Parcel` and add an attribute to record type of plant.

```
<Person, BirthName: string, Sex: string dom-
val {"male","female"},
BDate: date, {Age():integer}>

<Parcel, PID: integer, Owner: Person,
Address: string, Surface: integer, Geo: POLYGON>

<CerealParcel: Parcel, CropType: string>
<IndustrialParcel: Parcel, PlantType: string>
```

3.2.2 Cooperation context

On a site, a cooperation context acts as a mediator between a reference context and the local data. It consists of cooperation classes which are used to express local interpretations (mediation roles) of mediation classes.

A. Mediation roles

To cooperate and reconcile semantic differences, participant GIS need a set of commonly understood objects which are used to interpret data from other sites. Interaction between sites will be done through different perspectives of the commonly agreed on objects. In our approach, the common objects are represented by mediation classes and the different interpretations are different roles played by the mediation classes on different sites. A mediation role is defined as follows: 1) describe the subset of attributes (of a mediation class) on which the local site agrees, 2) use a qualification to restrict the properties or semantics of the objects that plays the role. Like a class, a mediation role has a set of attributes and methods which define its properties and behavior. But unlike class it doesn't create or delete any objects. Formally, a mediation role is defined as follows.

Definition 1 *(Mediation Role)*
Let $\mathcal{MR}$ be the set of all the mediation role. A mediation role $mr \in \mathcal{MR}$ is a tuple $mr =< mc, AttList, MethList, Q >$ where:

- $mc \in \mathcal{MT}$ is a mediation class whose interpretation is mr
- $AttList(mr) \subseteq AttList(mc)$ is a subset of attributes of mc built by using algebraic operations Select and Project
- $MethList(mr) \subseteq MethList(mc)$ is a subset of mc methods
- Q is a logic formula (qualification formula) associated with mr. It can be used to specify a constraint on the objects that play the role mr.

□

The mediation class mc is the role player of the mediation role mr. For example, to express that `Farmer` in site S_1 have the semantics of mediation class `Person`, we create the following mediation role $mr1$:

```
mr1=<Person, BirthName: string, Sex: string,
BDate: date, {Age():integer}, BDate>"01/01/1928">
```

We note that all the information in mediation class is supplied and the qualification associated with `DateN` is used to state the fact that age of the farmers in the GIS S_1 are less than 70 years.

B. Cooperation classes

Cooperation classes are the means by which local objects are shared between GIS. Cooperation classes from two GIS are semantically equivalent if they are defined using the same term of the ontology. Cooperation classes can be simple objects or complex objects if they are aggregated by virtual classes. A cooperation class definition incorporates a mediation role, a view defined by a virtual class which links the mediation role to local objects, and context transformations which are used to transform object description from one cooperative context to another. Definition 2 shows a formal definition of cooperation class.

Definition 2 *(Cooperation Class)*
A cooperation class $cc \in \mathcal{CC}$ is a tuple
$cc =< Name, AttList, MethList, cv, context_C >$ where:

- $Name(cc)$ is the name of the cooperation class
- $AttList(cc) = \{A_i : T_{A_i}\},\ i = 1..n,\ A_i \in Att_{name},\ T_{A_i} \in \mathcal{T}$ is the set of attributes of cc
- $MethList(cc) = \{M_j : \{p_k : T_{p_k}\} | M_j : \{p_k : T_{p_k}\} : T_{res}\},\ j = 1..m,\ k = 1..q,\ M_j \in \mathcal{MN},\ T_{p_k} \in \mathcal{T},\ T_{res} \in \mathcal{T},\ p_k \in \mathcal{PN}$ is the set of the methods attached to cc

- $context_C = \{< mr, \mathcal{CTF} >\}$ defines the context of cc. It is a set of tuples where mc is a mediation class such as $\exists mr \in \mathcal{MR} mr.c = mc$, $\mathcal{CTF}$ is a set of context transformations.
- cv is a virtual class encapsulated by cc such that $AttList(cc) \subseteq AttList(cv)$ and $AttList(mr.c) \subseteq AttList(cv)$ and $MethList(cc) \subseteq MethList(cv)$ and $MethList(mr.c) \subseteq MethList(cv)$. $\square$

C. Context transformation

A context transformation is a function which establishes a mapping from one value domain (local) to another (cooperation). A context transformation is associated with each mediation role to allow objects to migrate from a local context to a cooperation context.

Formally, a contexte transformation is defined as floows:

Definition 3 *(Context Transformation)*
$\mathcal{CTF}$ is the set of the context transformation: $\mathcal{CTF} = \{\uparrow_o^{mr}\} \cup \{\downarrow_o^{mr}\}$.
They are defined by:
- the function type $\uparrow_o^{mr}$, is used to translate a local value of an object into its value for the mediation role mr.

$$\uparrow_o^{mr}: dom(A_1) \times \ldots \times dom(A_n) \rightarrow dom(A'_i),$$
$$A_1, \ldots, A_n \in AttList(c),\ A'_i \in AttList(mr)$$

- the function type $\downarrow_o^{mr}$ is used to translate a mediation role value of an object into its local value.

$$\downarrow_o^{mr}: dom(A'_1) \times \ldots \times dom(A'_k) \rightarrow dom(A_l)$$
$$A'_1, \ldots, A_k \in AttList(rm),\ A_l \in AttList(c)\ \square$$

Example: In GIS S_1 the attribute `sex` of `FarmWorker` is coded by a single character (M or F) while in the mediation class `Person` this attribute is coded as male or female.

```
↑_o^mr Person.Sex(FarmWorker.Sex)=
 {if FarmWorker.Sex="M" return("Male")
          else return("Female")}
↓_o^mr FarmWorker.Sex(Person.Sex)=
 {if Person.Sex="Male" return("M")
           else return("F")}
```

Finally, the cooperation class `CCFarmWorker` encapsulates both the virtual class `CVFarmworker` and its context (mediation role and context transformations). The cooperation class `CCFarmWorker` is defined below :

```
CCFarmWorker =: <
  CVFarmworker: <Name: string, Sex: char,
Address: [city: string, street: string],
BirthDate: date, Status: string,
{Age(): integer...},

  mr1: <Personne,Name: string, Sex: string,
BDate: date, {Age():integer, ...},
BDate>"01/01/1928">,

  CTF: {
↑_o^mr Person.Sex(FarmWorker.Sex)=
 {if FarmWorker.Sex="M" return("Male")
      else return("Female")}
↓_o^mr FarmWorker.Sex(Person.Sex)=
 {if Person.Sex="Male" return("M")
      else return("F")} > ... >} >.
```

4 Conclusion

The aim of this paper has been to examine the design of interoperable GIS. We argue that resolution of semantic differences among various systems must be based on the representation of context informations which can be used to capture the semantics of various systems. The paper present a promising direction using agent paradigm and semantic mediation to realize semantic GIS interoperability. It has concentrated upon the distributed spatial object model AMUN that has the following main characteristics: 1) definition of spatial data types to deal with spatial objects, and 2) definition of a set of concepts to deal with distribution and heterogneties: virtual classes, mediation class, mediation role, context tranformation and cooperation class. A prototype for the ISIS architecture is underway. It uses Java language and Corba middleware to implement different types of wrappers agents (O2, Access, Postgres) which provides basic access to different spatial databases. The initial stages of our project are devoted to the definition of the architecture and the data model. Our future work will focus on spatial query processing to handle the distribution and sharing of not only spatial objects but also specialized spatial operators.

References

[1] B. Amann. Integration gis components with mediators and CORBA. Technical report, 1997. ftp://sikkim.cnam.fr/pub/Reports/GISMed.ps.gz.

[2] Y. Bishr. Overcoming the semantic and other barriers to gis interoperability. *Int. Journal of Geographical Information Science*, 12:299–314, 1998.

[3] O. G. Consortium and T. O. T. Committee. The Open GIS abstract specification. Technical report, 1996.

[4] T. devogele, C. Parent, and S. Spaccapietra. On spatial database integration. *Int. Journal of Geographical Information Science*, 12:335–352, 1998.

[5] T. Gruber. Towards principles for the design of ontologies used for knowledge sharing. *Int. Journal of Human-Computer Studies*, 43:907–928, jun 1995.

[6] R. Laurini. Spatial multi-database continuity and indexing: a step toward seamless gis data interoperability. *Int. Journal of Geographical Information Science*, 12:373–402, 1998.

[7] E. Leclercq, D. Benslimane, and K. Yetongnon. Semantic mediation between cooperative spatial information systems: The amun data model. In *Proceedings of IEEE ADL'99, Advances in Digital Libraries Conference*, USA, May 1999.

[8] S. Nittel, R. Muntz, and E. Mesrobian. geoPOM: a heterogeneous geoscientific persistent object system. In *Statistical and Scientific Database Management (SSDBM)*, 1997.

[9] N. Tryfona and J. Sharma. On information modeling to support interoperable spatial databases. In *Advances Information System Engineering, 8th International Conference, CAiSE'96, Heraklion, Crete, Greece, May 20-24*, pages 210–221, 1996.