

Metadata Integration and Geodata Integrity

Farshad Hakimpour
Department of Geography
University of Zurich
farshad@geo.unizh.ch

Xiangru Yuan, Andreas Geppert
Department of Information Technology
University of Zurich
{yuan,geppert}@ifi.unizh.ch

1 Overview

Metadata Integration and Geodata Integrity (MIGI)¹ is a research project aiming to contribute in solving existing problems of data and metadata integration focusing in the domain of spatial databases. This report discusses the barriers causing the two main problems addressed in the proposal. It also presents a detailed framework to approach the relevant and suitable solutions.

1.1 Introduction

Nowadays, providing a user with huge amount of data from different sources across networks or inter-networks in a short amount of time is done by interoperable systems. The interoperability at the hardware and software level (operating system, network protocol, etc.) is not a major issue any longer. However, the provided data to a user should be in a way that s/he can simply use such amount of available data. Users of spatial data expect information systems to help them to search and process the data as well as supporting them with necessary information and knowledge about the data – i.e., metadata. They also expect to have homogenous interfaces for managing, modeling and processing the data from different sources.

Though interoperability has to overcome complexity of conversion and integration process, there is a long way from data transfer and data format exchange (or conversion) to system interoperability. Interoperability issues not only refer to different structures and models of data sets, but also to the different methods and operations applying to the data. In the domain of GIS (Geographic Information System) interoperability, differences in data sources, disciplines, tools and repositories can cause heterogeneity [1]. Interoperability helps to reuse geodata and avoid waste of our assets. It has the potential to offer a prompt reaction to obtain suitable data set when dealing with geographic information analysis (such as natural disaster management).

In order to achieve the sharing and exchange of existing data between different departments, semantic heterogeneity between data from different component systems must be taken into account. Without knowing the semantics of data (i.e. their meaning), it is impossible to integrate or update data appropriately. Unfortunately, there are only few integration approaches that consider the semantics of data. However, without an applicable solution, the development of integrated data management systems can not be achieved with reasonable costs.

1. The project is funded by Swiss National Science Foundation (SNSF). Project Number: 2100-053995

The integration tasks of this project are to solve the conflicts and to keep the quality of spatial data. The conflicts of spatial data involve three types. A first type of conflict reflects metrical differences. This type of conflict may, for instance, arise if data have been acquired with different precision. A second category conflicts concerns topological problems between data from multiple sources. Topological conflicts may occur due to different dimensionalities of objects that have to be integrated. The third and most severe type of conflict is a semantic nature.

In the process of spatial data integration, it is very important to keep the quality of data from the semantic point of view. The quality of spatial data includes lineage accuracy, positional accuracy, attributed accuracy, completeness consistency, logical consistency [56], temporal and semantic accuracy [51].

1.2 Main Purpose

From the architectural point of view, there are several solutions for data integration. They are manual integration, common user interface, integration by applications, integration by middlewares, uniform data access, and common data storage [20]. Federated database management systems (tightly coupling) meet the requirements of MIGI project.

The semantics of data can be used, e.g., to determine not only structural, but also content-related differences between data from different sources. This semantics must be stored explicitly to make it usable for integration tools. Only a few approaches formalize the semantics of spatial data, but neither its implementation (based on concrete measures) in data dictionaries nor its (tool-based) utilization for integrating data from heterogeneous systems has been accomplished yet.

There is an urgent need for research to represent the semantics of data at the conceptual level, utilize this semantics when integrating heterogeneously represented spatial data from different sources, and resolve semantic conflicts between data from different local systems. Many tools are required to support the integration process. Eventually, an integrated system should provide a global unified view on heterogeneous distributed data and ensure global integrity constraints over multiple data management systems.

These issues are the focus of the project MIGI. The integration of spatial data will be considered from the *semantic* point of view, whereas existing integration approaches have focused on the structural aspect.

The project will result in a framework for specifying the semantics of data in terms of well-defined categories and a predefined measure for each category. This semantics will be stored explicitly in a data dictionary and utilized when integrating data from multiple sources.

We should consider the integration of integrity constraints that are applied in local systems as well as the implementation of global integrity constraints using database technology. They are based on an already existing data management platform for standard data such that the MIGI project can focus on research issues and are less concerned with solving basic technological problems. The resulting integrity module shall be used to implement spatial integrity constraints within the database management system.

The reminders are as follows. The next section discusses the semantic conflict problems and resolution for semantics heterogeneity. Section 3 discusses integrity control in the federated database systems.

2 Semantic conflict resolution

The total number of geodata providers in different contexts is increasing - while some are disappearing. This implies the need for a flexible approach that can deal with the existing and the future geodata providers in interoperable systems. A standard or uniform model for spatial data is the first step to approach the solution for schema heterogeneity. Nevertheless, low flexibility (considering the further development of the local systems and keeping their autonomy) and high complexity for administration may cause more problems rather than solving the existing ones (such as the canonical model suggested in [81]). Besides, it is unlikely to force all software developers to use one specific modeling approach, no matter how good that approach may be [28]. The OGC (Open GIS Consortium) specification aims to solve the problem of heterogeneity at the spatial data modeling level.

2.1 Problem Definition

Nowadays, information communities with their own data sets already exists. Each of them is using its own vocabulary and semantics specification. Explicit representation of such specification (in metadata) can help in detection (and possibly resolution) of semantic heterogeneity. The key issue is a formalism that can convey the detailed specification in a way that minimum interpretation will be left to the users' knowledge and implicit interpretation of data. It requires the understanding of the terms and definitions in the metadata of both the data set (i.e., source) and the user (i.e., target) information community.

This problem concerns users working with different systems (user interfaces) or one system with data from different resources. Meanings of the words or understanding of a concept may vary from one community to another. A solution can help users who may not be able (if they are willing) to interact with different interfaces based on different concepts. On the other hand, considering, the differences in user communities make it too difficult and idealistic to provide users with such homogeneity [28].

Any type of data exchange needs an exchange of knowledge on a higher level [73]. This knowledge has been referred to as semantics. If we consider the metadata not only as a description of the schema definition in a data set, but also a description of the conceptualization of the reality, then representation of semantics is a part of metadata. Semantics refers to user's interpretation of the computer representation of the world –i.e., the way users relate computer representation to the real world [49]. Considering that, users always abstract the real world by their needs (i.e., they neither observe nor represent all the details of the real world); therefore, any user or application has its own semantics (or way of interpreting computer representation of the real world). This causes what is called semantic heterogeneity.

Semantic heterogeneity is more than differences in modeling approaches used to design or represent schemas (e.g., relational and object oriented, or topological and spaghetti spatial modeling approaches) or differences in schemas used to represent the same entity type by the same formalism. Challenging obstacle of semantic heterogeneity, which is the main concern of this part of the project, is the lack of sufficient specifications that causes misinterpretation of concepts by common sense [8]. As an example of semantic heterogeneity consider a concept *population density*. It may be represented in different communities as following:

name: P-D value:17 → 17 person per acre for inhabitants of a town represented as a 2 dimensional spatial object.

name: POP-DEN value: 53 → 53 person per square mile for inhabitants of a city represented by a 0 dimensional spatial object.

name: Population-Density value: 3 → 3 person per flat for residents of a building represented by a 0 dimensional spatial object.

name: Density value: 1.2 → 1.2 person per 100 Quebec meter for residents of a building represented by a 2 dimensional spatial object.

name: P-Den value: 1.5 → 1.5 patient per room for patients in a hospital represented by a 2 dimensional spatial object.

The minimum common implicit assumptions are:

Population density is a measure for number of people,

Density is a ratio related to space,

It can be linked to 0-dim and 2-dim features,

The known unit for people is persons.

In spite of the common assumptions, you can see much of differences between the interpretation of the values.

2.2 Research Questions and Goals

The important questions to be approached by this plan of work are as follows:

1. *How can semantics be expressed and used?*

We need a type of formalism to carry on the semantics –i.e., that formalism should have the capability to represent semantics. Such formalism should convey those aspects of knowledge needed for common understanding the shared data between Geographic Information Systems.

2. *What is semantic heterogeneity? What type of conflicts is caused by semantics?*

Semantic Heterogeneity or other related terms such as semantic mismatch [58] or semantic similarities [8] are terms that need clear definitions and understanding. Semantic heterogeneity has been analyzed based on the differences in the schema and is approached by mapping between schemas [5][8]. However, what makes the first initiative to compare two schema components is similarity between them - i.e., if there is not absolutely any similarity between two entities, then comparing their schema definitions (if possible) cannot be meaningful. It is also important to know what type of conflicts is caused by semantic heterogeneity. Classification of semantic heterogeneity based on the resolution methods will be a contribution in the domain of GIS.

3. *How can description of semantics in GIS be utilized to resolve semantic heterogeneity?*

There is a need to specify the kind of provided tools or supported services by systems taking part in semantically interoperable systems. We need to explore system components and features that play important role in semantically interoperable systems in the domain of GIS. This work aims to propose architecture for semantically interoperable systems.

2.3 Prior Research

As mentioned in the NCGIA report (in [28]), semantic interoperability is a challenging and difficult impediment in inter-operating GISs. Problems at the other levels (such as: hardware communication, data structure and software protocols) have long been solved.

At the schema level data definition languages such as INTERLIS [22] and EXPRESS [74] (STEP [62]) can help the data format and schema transformation. The OGC also provides specifications for information and service interchange between systems [58], which helps at

the same level. Kim *et al.* performed a comprehensive study for classification of schema heterogeneity [44][42]. They present solutions for several types of schema heterogeneity in RDBs and OODBs.

At the next level semantic heterogeneity should be detected and resolved. It is important to notice what is referred to by semantics in this text is the meaning of the concepts (such as: classes, attributes, values, relations), not the semantics of the spatial models (or semantics of space). For instance, Kuijpers *et al.* (in [43]) discuss semantics of two spatial models, or Casati *et al.* (in [16]) depict the basis of geographic representation. Their concern is semantics of space and the way space and objects in space are represented, not the semantics of objects represented in a data set.

Sheth (in [70]) presents an overview of the interoperability issues. He divides the course of its development to three eras until the need for resolving semantic issues. He also considers ontologies, contexts, and semantic correlation as important issues in future of interoperable systems. Kashyap and Sheth (in [45]) state that database schemas do not convey sufficient information for resolution of semantic heterogeneity. They discuss the need for domain specific ontologies, and represent them by description logic.

Bishr (in [8]) presented a model (SFDS semantic formal data structure) based on FDS [50]. He illustrates an architecture for semantically interoperable systems, and mostly concerns with the schematic aspects of heterogeneity. The ontology has also been discussed in his thesis but not much of detail.

Rosenthal and Sciore are introducing an architecture for semantic interoperability in [67]. They explore different kinds of interoperability problems in a distributed object management environment. The presented architecture is based on four main constituent: argument-describers (functions to determine the assumption about the meaning and representation of arguments), conversion functions (a library), a planner (determines a strategy for converting a property-value to another, by considering argument descriptor and available conversion functions) and a request broker. They present a sound architecture, though, they resolve the schematic heterogeneity problems rather than semantics heterogeneity.

OGC paid attention to semantics issues of GIS. There is an interest group under OGC and a draft standard on semantics. Since the Semantics and Community Metadata are the bases of an information community [57], to transfer a data set from an information community to another OGC suggests the use of something called Semantic Translator [58].

An important possible solution to semantics problems, which has been attracting attention, is formal ontology [35]. In the domain of philosophy ontology explains the nature and essential properties and relations of all beings (Webster's Unabridged Dictionary) and is based on the truth and the nature of the beings independent of our knowledge. In the domain of artificial intelligence it is a specification of conceptualization. In this domain ontologies have been used for sharing and reusing knowledge [34]. This goal is very close to the goal of MIGI project. Ontologies are considered more than schema definitions. A shared common ontology guarantees the consistency of the understanding of communities from the world. This implies a consistency in the conceptual level. The main advantage of ontologies is that it helps us to be independent from the background knowledge of the community or at least have the minimum dependency on the background knowledge of the community.

On2broker [21] (new release of Ontobroker [17] and SHOE [37]) are two projects using ontologies as a concept for searching World Wide Web. On2broker is using ontologies represented in a language based on Frame-logic. It is based on the closed-world assumption and deals with a domain specific ontology for every query. SHOE is another example of a search

engine using ontologies that is based on Description-logic. SHOE is based on open-world assumption. Both systems are based on some extra tags which must be added to the HTML pages by the authors of the pages.

Ontolingua [23] and (KA)² [7] are examples of the projects for designing ontologies. Ontolingua provides users with tools to define their ontologies and export the ontologies in different formats (e.g., KIF) to be used by several existing knowledge bases. Participants in (KA)² project try to develop an ontology for knowledge acquisition community.

2.4 Proposed Research Method

2.4.1 Study on Real Data

A study on integrating data from different resources (from different disciplines) can help in understanding the nature of the conflicts caused by semantic differences. This can be a good preliminary step to investigate the problems in practice. An appropriate application (within the GIS domain) area should be defined. This study can be performed only in case of availability of the real data.

Objectives: this case study approaches the questions 1 and 2 mentioned in section 2.

2.4.2 Study on Ontologies

The study proposed here is to specify a set of common concepts, which is used in one information community and an attempt to represent the ontology in a formalism for the set of concepts. Next step is finding the minimal definition of concepts that two (or more) information communities can agree upon - i.e., minimum restrictions for the information communities committed to a common ontology [34].

Objectives: By this study, we can approach two questions 2 and 3 addressed in section 2. Since ontology is an important means to represent semantics [45], ontologies can help to formalize part of semantics. As we try to formalize a common ontology it can be a base for evaluating the similarity or heterogeneity of semantics.

2.4.3 Study on Semantically Interoperable Systems

Another issue to be considered is how systems can cooperate. Where and how the defined ontologies and semantics in previous case studies can help in a semantically interoperable system. Therefore, a study on existing architectures adopting one (or a combination of them) to develop a prototype is proposed.

Objectives: This study approaches questions 1 and 3 in section 2. Classification of semantic heterogeneity can be done based on different criteria but after this case study we approach the classification based on the resolution method. We can also realize the problems in the existing architecture and try to find features to improve the architecture of semantically interoperable systems.

3 Integrity Control

This section gives problem definition, research questions and goals, related work, and proposal research method about integrity control in the federated database systems.

3.1 Problem Definition

This part provides problem definition about integrity control in the federated database systems. It contains concept, representation and implementation of integrity control.

3.1.1 Integrity Control

A database (DB) is a collection of data, used to represent information of interest to an information system. A database management system (DBMS) is a software system able to manage collects of data. A database together with a DBMS is a database system (DBS) [68]. In a word, a database is a collection of data managed by a DBMS [2].

In the federated database systems [72], we need to consider further restrictions on the data derived from the mini-world. These restrictions are called integrity constraints. It is very important to keep data integrity in the process of integration at the global level and update at the local and global levels. Integrity data meets all these restrictions. At the same time, only integrity data is stored in the database. In the field of information systems and database, the term *integrity* normally refers to the correctness or validity of the stored data, as defined explicitly by means of integrity constraints or integrity rules [26]. There are several terms – integrity, consistency, validity and correctness- used differently by different authors. We prefer to the term *integrity*.

Motro [52] makes the distinction between two components of integrity: *validity* and *completeness*. We can say that a database has integrity if all its data are corrected (valid) and if it contains all relevant data (it is complete). To *ensure integrity*, one should check systematically the facts with respect to the Universal Discourse. The goal of ensuring integrity is to find out a mechanism for integrity constraint representation and implementation.

Integrity control involves the avoidance of semantic errors and semantically inconsistent database states through the observance and monitoring of database *integrity constraints*. Integrity control considers semantic integrity, using integrity constraints. Access control aims at ensuring the confidentiality (security) of data [40]. Operational integrity is considered by transaction, which includes concurrence control where locking and timestamp techniques are used. Reliability control deals with the prevention of errors due to the malfunctioning of system of system hardware or software, using recovery and replication techniques.

Integrity constraint (in short, IC) means a constraint that must remain true for a database to preserve integrity. When an integrity constraint is represented and implemented as a rule, such as a deductive rule and an active rule, we call it as an integrity rule (in short, IR).

3.1.2 Tasks of Integrity Control

We assume that architecture of this project is shown in Figure 3.1. First, we have several component databases (in short, CDB), such as relational databases, deductive databases, active databases, object databases, etc. They are called as local databases. Based on the standard object-oriented database model [CB+97], we translate every component database into component object database (in short, CODB). We also assume that the task of translation is finished and out of this project. Then, we integrate CODBs to form a federated database (in short, FDB) with a uniform interface. It is called as a global database. We assume that FDB is still

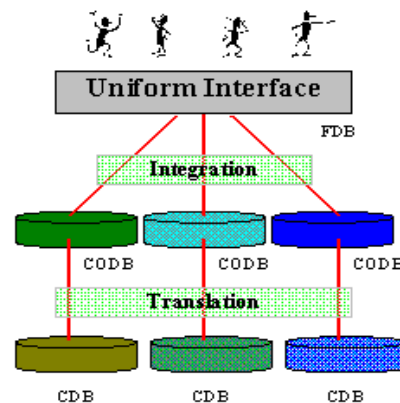


Figure 3.1 Integration Problem

based on object data model. This project focuses on this level.

At the same time, we assume that a CDB or a CODB is an integrity database, and has three components:

The extensional database, E_{DB} ;

The intensional database, I_{DB} ;

The integrity constraints, IC.

The E_{DB} is the database's collection of facts, which is intended to represent specific data. The I_{DB} is the database's collection of rules, which is intended to generate data. The IC is intended to verify that the data (both E_{DB} and I_{DB}) in the database is consistent with the general model of the world.

For a federated database, we hope that it is still an integrity database and has also three components: The extensional database, GE_{DB} , the intensional database, GI_{DB} , and the integrity constraints, GIC. And

$$GE_{DB} = \Sigma E_{DBi}$$

$$GI_{DB} = \Sigma I_{DBi}$$

$$GIC = \Sigma IC_i$$

This symbol, Σ , means integration. GE_{DB} of FDB is the integration of E_{DB} , GI_{DB} is the integration of I_{DB} , as well as GIC.

So, the tasks of integrity control are as follows:

To keep the integrity of federated database when integrate E_{DB} , I_{DB} , and IC.

To keep the integrity of federated database when update E_{DB} , I_{DB} , and IC.

To keep the integrity of local database when update GE_{DB} , GI_{DB} , and GIC

We also assume that the integrity of local database is guaranteed by local database management systems (DBMS), and that of federated database is guaranteed by federated database management systems (FDBMS).

There is an urgent need to represent the semantics of data at the conceptual level, utilize the semantics when integrating heterogeneously presented spatial data from different sources, and resolve semantic conflicts between data from different local systems. Many tools are required to support the integration process. Eventually, an integrated system should provide for a uniform interface, an integrated view – a global unified view – on heterogeneous distributed data and global access methods – ensuring global integrity constraints – over multiple data management systems.

3.1.3 Representation of Integrity Constraints

Besides the definition of integrity constraint, the representation of integrity constraint is also important. We assume that the representation of IC is specified in the database. There are different methods to represent the integrity constraint. Integrity constraint(s) is or are:

Metadata. Integrity constraints are metadata that specify the conditions that a database state (sequence of states) must satisfy in order to be consistent [38].

Formula. Integrity constraint is a closed first-order formula that the database is required to satisfy [59].

Query. Integrity constraint is a closed query that must always be true after a database update [30].

Conditions, a declarative specification of the conditions that a database state (or sequence of states) must satisfy in order to be legal [25]; State conditions to be satisfied by each state of the database [55].

Statement. A statement of a condition that must be met in order to maintain data consistency [39].

Boolean expression, A boolean expression over a global database schema [36]. Integrity constraints are well-typed boolean expressions built using the names and classes of the schema and general operators [6].

Rule, which guarantees the integrity of a database [4] [9][12] [80].

When integrity constraint is represented in a data model, it is called as integrity model. Integrity models are interrelated with data models and transaction models, by offering concepts for representing constraints to mini-world states and constraints to mini-world processes [68]. A semantic data model is used to model data secrecy and integrity in [66]

An extension to the UML meta-model is used to represent integrity constraint [61]. Transition graph, a kind of graph, is used to monitor integrity constraint [71] [32]. Rule hypergraphs are used to agree the possible sequences of triggered rules, and define critical paths in associated rule hypergraphs to correspond the propagation of conditions in [69].

The above representations are based on specific purposes. Most of them are for local databases, which are not very suitable to federated database. For this project, we need to find out a mechanism, which is based on some assumptions, to represent integrity constraint for database integration and update.

3.1.4 Implementation of Integrity Constraints

After defining and representing integrity constraint, implementation of integrity constraint is in hand. We assume that the implementation of IC is carried out by a DBMS. Integrity constraint implementation includes enforcement (checking and maintenance) and management (verification, handling & repairing). The former is necessary part that should be provided in the global level. But, the latter is a hard problem really in federated systems. And further problem is specific to spatial data.

Integrity constraint enforcement means that databases must incorporate some mechanisms to ensure that integrity constraints are always satisfied after the application of a transaction. There are several approaches to be considered depends on the semantic of the integrity constraints and of the database. The best known approaches are *integrity constraint checking* and *integrity constraint maintenance*.

Integrity constraint checking is the most conservative approach to deal with integrity constraint. It rejects the transaction that, if applied, would violate some integrity constraints. An important drawback of this approach is that user may be completely lost regarding possible changes to be made to the transaction to make it obey the integrity constraints [65].

An alternative approach, aimed at overcoming this limitation, is integrity constraint maintenance, which is concerned with trying to identify additional updates (i.e. repairs) to be added to the original transaction to guarantee that resulting transaction does not violate any integrity constraints [54].

Integrity constraint management includes *verification*, *repairing* and *handling*. When new integrity constraints are defined on a database or formed on a federated database, it has to be checked if the constraints themselves are validate syntactically and semantically. This process is called *integrity constraint verification* [27].

3.2 Research Questions and Goals

The Figure 3.2 shows databases integration and update. For instance, we have a local compo-

nent object database $CODB_1$, which contains an object O_1 , a local integrity constraint LIC_1 , a local rule R_1 and domain D_1 . Another local component object database $CODB_2$ contains local object O_2 , local integrity constraint LIC_2 , a local rule R_2 and domain D_2 . We integrate these two objects and integrity constraints to form a federated database with a global uniform interface and a unified view. In this federated database, we have an object O , a global integrity constraint GIC , a global rule R and domain D . Therefore, we have following representations,

$$\begin{aligned} O &= O_1 \oplus O_2 \\ GIC &= LIC_1 \oplus LIC_2 \\ R &= R_1 \oplus R_2 \\ D &= D_1 \otimes D_2 \end{aligned}$$

Where \otimes means some operators; \oplus means integration.

Integrity constraints are an important part of database. A user on the global level needs transparency. That is, the user is not required to have knowledge about the federation or the local database. The global user knows only the global objects, relate to global integrity constraints and global rules. Only operations, e.g. inserting a new object, which are allowed by one of the local systems, can be performed. This must be reflected by global integrity constraints. Otherwise an operation could be rejected without a visible reason for the global user.

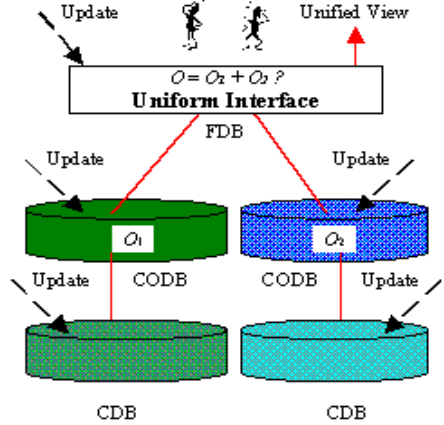


Figure 3.2 Integration and Update

The questions involve two aspects: *integration* and *update*. The integration of databases, which is applied in local systems, includes local object integration, local integrity constraint integration and local rule integration. The update of database is applied in local systems and global system, including local object update, local integrity constraint update, local rule update, global object update, global integrity constraint update and global rule update. In this project, we just consider local object update and global object update, and keep the integrity of databases at these two processes.

Many research have been done on integrity constraint in centralized / homogenized database systems. Integrity constraints checking and maintenance after potentially violating update have been performed. The most problems of integrity constraint checking and maintenance have been solved. Unfortunately, for the federated database system, the database management systems does not have (full) control over local data stores. First, it can not determine when updates have occurred. Second, traditional approaches to integrity constraints are not feasible. So, in this project, we focus on the general problem of integrity control in the federated database system, that is to check or maintenance integrity constraints when facts of federated database are updated. We leave the research, updates of integrity constraints and rules, to the future.

According to the above example, the important questions about integration and update of databases that should be approached by this plan are as follows:

1. What's the meaning of IC in the federated database? How to define it?

We need to give the exact meaning of IC in the federated database and define the concept of IC. Different concepts need different approaches to represent, and are carried out by different methods of implementation.

The goal of this plan is to ensure the integrity of federated database and local component object database, which are based on object database model. One solution is to develop an integ-

rity control model for federated database systems. This model should be more general than the local integrity control models.

So, it is necessary to study the relevant literature on integrity types in federated systems and for spatial data, to classify the integrity constraint types. Then, it needs to investigate local integrity control models, for instance ECA rules, if applied, and compare existing approaches, in order to develop a classification framework for integrity control. Furthermore, it is important to find out a method to represent integrity control and a mechanism to define integrity control model.

2. *How to integrate ICs? How to form ICs when integrate facts and rules?*

An example shown in figure 3.2.1 is how to integrate integrity constraints to form global integrity constraint GIC and what the exact meaning for the representations: $O = O_1 \oplus O_2$, $GIC = LIC_1 \oplus LIC_2$, $R = R_1 \oplus R_2$, and $D = D_1 \otimes D_2$.

First, it needs to understand the local initial consistent states clearly, since a first step towards the integration of integrity constraints is a homogenization of the local integrity constraints in CODB. It means that we need to transform the local data model to form a global (canonical) data model, for instance ODMG object database model [CB+97].

Then, a contribution is to find out an approach to form new consistent states in the global level, in order to specify the new integrity constraints and rebuild the initial integrity constraint states. For instance, local integrity constraints LIC_1 and local integrity constraint LIC_2 are formed in different ways. Before integration, they should be rebuilt and have the same form.

Third, it needs to let global integrity constraints reflect the changes of local facts and integrity constraints (since the facts are changed first). When the object and integrity constraints in the local system, for instance object O_1 and integrity constraint LIC_1 , have been changed, a policy is needed to modify the corresponding object and integrity constraint, such as object O and integrity constraint GIC , in the global level. It is necessary to have a mechanism of communication protocol between local integrity constraints and global integrity constraints.

In one word, a method for conflict resolution policies and integrating local integrity constraints is waited for.

3. *How to keep the integrity of database when update facts?*

When we need to update the global object O , which ones should be updated for the local object? Are they object O_1 , or object O_2 , or both, or neither, or just parts of them? How to solve the inconsistencies in the process of integrity constraints handling and repairing? When we need to update the local object, such as O_1 , how to keep the integrity of federated database? Therefore, some policies, requirements and assumptions for fact updates are necessary to be identified and provided in the federated database system.

4. *How to describe integrity control from the architectural point of view?*

The architecture of integrity control model must be defined, where the different support for integrity control in CDBMS (if any) is taken into account. The protocols for interacting with other FDBMS components such as the global query manager and the global transaction manager must be defined.

The policy for the integrity enforcement (checking and maintenance) and management (verification, handling and repairing) should be investigated. And some requirements and

assumptions for integrity control model must be identified.

3.3 Prior Research

This section gives prior researches about integrity control from four levels: the concept of IC, the representation of IC, the implementation of IC and IC in the federated database system.

Concepts: [52] gives the meaning of the integrity of the database, which refers to the correctness or validity of the data in the database. [26] provides a survey for integrity control in the relational database systems. [63] gives a survey of methods for definition and enforcement of dynamic IC. At the same time, some semantic integrity concepts are specified in [19][18] [33] [64] [78][79].

Representations: A description and assessment of two data integrity models: the Biba model and the Clark-Wilson model are provided in [83]. [30] examines various possible semantics of ICs and describes a number of nontraditional but promising application of constraints. A kind of graph for the monitoring of temporal IC, called transition graph, is used in [71][32]. [75] gives semantic IC representation models. [24] embeds declarative IC into ODMG-93, and a set-theoretic model is used to express dynamic IC in [10]. [60] provides a method to reduce all dynamic ICs to a form that can be evaluated in each state, taking into account only the facts of current and previous states.

Implementations: A survey of the early methods in the area of IC maintenance is founded in [25] and that of the current methods are in [55]. [77] provides a comparison of some techniques for view updating and IC maintenance. IC maintenance based on the generation and execution of active rules are provides in [13][29][48]. [15] [47] aim at incorporating the information provided by IC into the update request then unfolding the resulting expression. [77] takes into account the IC every time that a new update is considered. [46][32][53] develop dynamic IC enforcement. [41] describes a proposed model for incorporating declarative IC maintenance in an object-oriented database management system. [36] provides protocols for IC checking in federated database. [27] gives a specification and implementation of IC in object-oriented database systems.

FDBS: regarding to the specific step of the federated database design process in which they occur, integrity constraints could be especially classified into several types (see Figure 3.3) [14]. There are *Local integrity constraints (LIC)*, *Transformed integrity constraints (TIC)*, *Exported integrity constraints (EIC)*, *Integrated integrity constraints (IIC)*, *Local integrity constraints (GIC)*, *External integrity constraints (XIC)*.

The integration of relations with conflicts based on relational data model is shown in Figure 3.4[76], which categorizes the types of conflicts as follows:

- 1) *Value-to-value conflicts: data representation conflicts, data scaling conflicts, inconsistent data.*
- 2) *Value-to-attribute conflicts,*
- 3) *Value-to-table conflict,*
- 4) *Attribute-to-attribute conflicts: synonyms, homonyms, naming conflicts.*
- 5) *Attribute-to-table conflicts,*
- 6) *Table-to-table conflicts: missing data.*



Figure 3.3 Constraints Types in FDBS

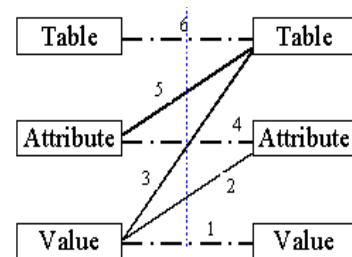


Figure 3.4 Integration of Relations

In the object-oriented federated database environment, [3] gives some methods for integrity rule merging, and six integrity rules merging methods are provided.

3.4 Proposal Research Method

In order to enforce and manage the integrity constraints in the tightly coupling federated database system, an appropriate model is necessary to be developed. This model is named as integrity control model. With this model, Ensuring integrity of databases in the processes of integration and update of databases are approached, and the representation and implementation of integrity constraints should be carried out. Therefore, for this model, it should be take the following studying items into account.

3.4.1 Study on the Concept of IC

The first task of this project is to define the concept of IC in the federated database systems by the integrity control model. In order to build up integrity control model in the federated database systems, object – oriented paradigm should be taken into account. Object – oriented techniques could provide powerful glue for integration. It is semantically rich canonical model. Object-oriented techniques can provide facilities for the complex tasks of the federated database systems. So, the integration and update of databases should be implemented, based on the object-oriented model.

Objectives: This study approaches the questions 1 and 4 in section 3.2.

3.4.2 Study on the Representation of IC

Since IC can be regarded as rules, we need to study on the ECA-rules paradigm. It is possible that we can consider the ICs as the conditions part of ECA-rules, such as regarding rules as objects, or representing rules with objects.

UML is used to represent IC; transition graph, a kind of graph, is also used to monitor ICs. Rule hypergraph is used to agree the possible sequences of triggered rules. So, this project needs to study on UML, transition graph, and rule hypergraph to represent IC, to specify the semantics, management and enforcement of ICs.

Objectives: This study approaches the questions 2 and 3 in section 3.2.

3.4.3 Study on the Implementation of IC

In order to implement operational integration based on object data model, the integrity control model should consider component-based techniques. This project needs to study on component-based techniques from the architectural point of view.

Objectives: This study approaches the questions 2, 3 and 4 in section 3.2.

4 References

1. G. Alonso, A. Abbadi, Cooperative Modeling in Applied Geographic Research, in International Journal of Intelligent and Cooperative Systems Vol. 3, No. 1, pp. 83-102, 1994
2. P. Atzeni, S. Ceri, S. Paraboschi, and R. Torlone, *Database System: Concepts, Language and Architecture*, McGraw-Hill Companies, 1999.
3. R. M. Alzahrani, M. A. Qutaishat, N. J. Fiddian, and W. A. Gray, Integrity Merging in an Object-Oriented Federated Database Environment, *BNCOD'95*, 1995, pp. 226-248

4. D. Beneventano, S. Bergamaschi, S. Lodi, and C. Sartori, Consistency Checking in Complex Object Database Schemata with Integrity Constraints. *IEEE Transactions on Knowledge and Data Engineering* 10(4): 576-598, 1998
5. S. Bergamaschi, S. Castano, S. De Capitani di Vimercati, S. Montanari, M. Vincini, An Intelligent Approach to Information Integration, in: *Formal Ontology in Information Systems*, edited by N. Guarino, pp. 256-268, IOS Press, 1998
6. V. Benzaken, and A. Doucet, Thémis: A Database Programming Language Handling Integrity Constraints. *VLDB Journal* 4(3): 493-517, 1995
7. V. R. Benjamins, D. Fensel The Ontological Engineering Initiative (KA)², in: *Formal Ontology in Information Systems*, edited by N. Guarino, pp. 287-301, IOS Press, 1998
8. Y. Bishr, Semantic Aspects of Interoperable GIS, ITC publication number 56, The Netherlands, 1997
9. C. Bauzer-Medeiros and P. Pfeffer, Object Integrity Using Rules, *ECOOOP'91*, pp. 219-230, 1991.
10. E. O. de Brock, A General Treatment of Dynamic Integrity Constraints, *Data & Knowledge Engineering*, 32, 233 – 246, 2000
11. R. G. G. Cattell, D. Barry, etc, *The Object Database Standard: ODMG 2.0*, San Francisco, California: Morgan Kaufmann Publishers, Inc., 1997.
12. S. Ceri and P. Fraternali, *Designing Database Application with Objects and Rules: The IDEA Methodology*, Addison Wesley Longman, 1997.
13. S. Ceri, P. Fraternali, S. Paraboschi, and L. Tanca, Automatic Generation of Production Rules for Integrity Maintenance. *ACM Transactions on Database Systems* 19(3): 367-422 1994
14. S. Conrad, M. Hoeding, S. Janssen, etc. Integrity Constraints in Federated Database Design, *Preprint Nr. 2/96*, University of Magdeburg, Germany, 1996.
15. L. Console, M. L. Sapino, and D. Theseider, The Role of Abduction in Database View Updating. *Journal of Intelligent Information Systems* 4(3): 261-280 1995
16. R. Casati, B. Smith, A. C. Varzi, Ontological Tools for Geographic Representation, in: *Formal Ontology in Information Systems*, edited by N. Guarino, pp. 77-85, IOS Press, 1998
17. S. Decker, E. Erdmann, D. Fensel, and R. Studer, Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In *Semantic Issues in Multimedia Systems. Proceedings of DS-8*, Edited by Meersman R. et al., Kluwer Academic Publisher, Boston, 1999, pp. 351-369. URL: <ftp://ftp.aifb.uni-karlsruhe.de/pub/mike/dfe/paper/rdf3.ps>
18. A. D. Deo, *Spatio - Temporal Constraints Database built on Object-Oriented Frameworks*, Ph. D Thesis, GMD-First Berlin, Germany, 2000
19. S. Dessloch, *Semantic Integrity in Advanced Database management Systems*, Ph. D. Thesis, University of Kaiserslautern, Germany, 1993.
20. K. R. Dittrich and D. Jonscher, All Together Now – Towards Integrating the World's Information Systems, *Advances in Database and Multimedia for the new Century – A Swiss – Japanese Perspective*, Tyoko, Japan, 1999.
21. D. Fensel, J. Angele, S. Decker, M. Erdmann, H. Schnurr, S. Staab, R. Studer, and A. Witt, On2broker: Semantic-Based Access to Information Sources at the WWW. In: *Proceedings of the World Conference on the WWW and Internet (WebNet 99)*, Honolulu, Hawaii, USA, October 25-30, 1999. URL: <ftp://ftp.aifb.uni-karlsruhe.de/pub/mike/dfe/paper/webnet.pdf>
22. Federal Directorate for Cadastral Surveying *INTERLIS A Data Exchange Mechanism for Land-Information-Systems*, November 1997. URL: <http://www.gis.ethz.ch/interlis/refdocs/iliv1r1e.pdf>, 1997
23. A. Farquhar, R. Fikes, J. Rice, The Ontolingua Server: a Tool for Collaborative Ontology Con-

- struction, URL: ftp://ftp.ksl.stanford.edu/pub/KSL_Report/KSL_96_26.ps, 1996.
24. C. Fahrner, T. Marx, and S. Philippi, DICE: Declarative Integrity Constraint Embedding Into the Object Database Standard ODMG-93. *Data & Knowledge Engineering* 23(2): 119-14, 1997.
 25. P. Fraternali and S. Paraboschi, A Review of Repairing Techniques for Integrity Maintenance, *RIDS'93*, pp. 333-346, 1993
 26. P. Grefen and P. Apers, Integrity Control in Relational Database Systems - An Overview, *Journal of Data and Knowledge Engineering*, 10 (2), 187-223, 1993.
 27. A. Geppert and K. R. Dittrich, Specification and Implementation of Consistency Constraints in Object-Oriented Database Systems: Applying Programming-by-Contract, G. Lausen (ed.): *Proceedings. GI- Conference BTW*, Dresden, Germany, March 1995
 28. M.F. Goodchild, M. J. Egenhofer, R. Fegeas, Interoperating GISs, Report of a Specialist Meeting Held under the Auspice of the Varenus Project (NCGIA), URL: ftp://ftp.ncgia.ucsb.edu/pub/Publications/Varenus_Report/Inter-op.pdf, 1997.
 29. M. Gertz, Specifying Reactive Integrity Control for Active Databases. *RIDE-ADS* pp. 62-70, 1994
 30. P. Godfrey, J. Grant, J. Gryz, and J. Minker, Integrity Constraints: Semantics and Applications. *Logics for Databases and Information Systems* 1998: 265-306
 31. N. Gehani and H. V. Jagadish, Ode as and Active Database: Constraints and Triggers, *Proceedings of the 17th International Conference on Very Large Database*, 1991
 32. M. Gertz and U. W. Lipeck, Deriving Optimized Integrity Monitoring Triggers from Dynamic Integrity Constraints. *Data & Knowledge Engineering*, 20(2): 163-193 .1996
 33. P. Grefen, *Integrity Control in Parallel Database Systems*, Ph. D. Thesis, University of Twente, 1992
 34. T. R. Gruber, Towards Principles for the Design of Ontology Used for Knowledge Sharing, in: *Formal Ontology in Conceptual Analysis and Knowledge Representation* edited by Guarino N. and Poli R., the International Workshop on Formal Ontology March, 1993, Kluwer Academic Publishers. URL: <http://ksl-web.stanford.edu/knowledge-sharing/papers/onto-design.ps>.
 35. N. Guarino, Formal Ontology and Information Systems, in: *Formal Ontology in Information Systems*, edited by N. Guarino, pp. 3-17, IOS Press, 1998.
 36. P. Grefen and J. Widom, Protocols for Integrity Constraint Checking in Federated Database, *International Journal of Distributed and Parallel Database*, 5(4): 327-355, 1997.
 37. J. Heflin, J. Hendler and S. Luke, SHOE : A Knowledge Representation Language for Internet Applications. Technical Report CS-TR-4078 (UMIACS TR-99-71), Departments of Computer Science, University of Maryland at College Park, 1999
 38. H. Ibrahim, W. A. Gray, and N. J. Fidian, The Development of s Semantic Integrity Constraint Subsystem for a Distributed Database, *Advances in Databases, BNCON 14*, pp74 – 91, 1996.
 39. M. Jarke, S. Mazumdar, E. Simon, and D. Stemple. Assuring Database Integrity. *Journal of Database Administration* 1 (1): 391 – 400, 1990.
 40. D. Jonscher, *Access Control in Object-Oriented Federated Database Systems*, Ph.D. Thesis, University of Zurich, 1998.
 41. H. V. Jagdish and X. Qian, Integrity Maintenance in an Object-Oriented Database, *Proceedings of 18th International Conference on Very Large Database*, 1992
 42. W. Kim, I. Choi, S. Gala, M. Scheevel, On resolving Schematic Heterogeneity in Multi-database Systems, in: *Distributed and Parallel Databases: An International Journal*, Vol. 1, No. 3, July 93, pp. 251-279, 1993.
 43. B. Kuijpers, J. Paredaens, L. Vanderzen, Semantics in Spatial Databases, in: *Semantics in Data-*

- bases edited by Thalheim B., Lecture Notes in Computer Science 1369, Springer-Verlag pp. 114-135, 1995.
44. W. Kim, J. Soe, Classifying Schematic and Data Heterogeneity in multi-databases Systems, in: *IEEE Computer*, December 91, pp.12-18, 1991.
 45. V. Kashyap, A. Sheth, Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies, in: *Cooperative Information Systems: Current Trends and Directions*, edited by Papazoglu M., Schlageter G., Academic Press, pp 139-178, 1998
 46. U. W. Lipeck, M. Gertz and G. Saake: Transitional Monitoring of Dynamic Integrity Constraints. *Data Engineering Bulletin* 17(2): 38-42, 1994
 47. J. Lobo and G. Trajcevski, Minimal and Consistent Evolution of Knowledge Bases, *Journal of Applied Non-classical Logics*, 7 (1-2), 117-146, 1997.
 48. E. Maabout, Maintaining and Restoring Database Consistency with update Rules, *Workshop DYNAMICS'98*, Manchester, 1998.
 49. R. Meersman, An essay on the role and evolution of data (base) semantics, in: *database Application Semantics*, Proceedings of IFIP WG 2.6 Working Conference on Database Application Semantics edited by R. Meersman and L. Mark 1995.
 50. M. Molenaar, O. Kufoniyi, T. Bouloucos, Modeling Topological Relationships in Vector Maps, in: *Advances in GIS Research, Proceedings of the Sixth International Symposium on Spatial Data Handling*, Edited by T. C. Waugh and R. G. Healey, September 1994, Vol. 1, pp. 112-126.
 51. J.L. Morrison. Spatial Data Quality. In: S.C. Guptill and J. L. Morrison (eds.): *Elements of Spatial Data Quality*. Elsevier Science Ltd., Oxford.
 52. A. Motro, Integrity = validity + completeness, *ACM Transactions on Database Systems*, 14(4), 480-502.
 53. C. Martín and J. Sistac: Applying Transition Rules to Bitemporal Deductive Databases for Integrity Constraint Checking. *Logic in Databases 1996*: 117-134, 1996
 54. E. Mayol and E. Teniente, Structuring the Process of Integrity Maintenance, *8th International Conference on Database and Expert Systems Applications (DEXA'97)*, LNCS-1308, Touse, France, September 1997, pp. 262-275.
 55. E. Mayol and E. Teniente, A Survey of Current Methods for Integrity Constraint Maintenance and View Updating, In *Advanced in Conceptual Modeling*, P. P. Chen and D. W. Embley, eds., Springer, 1999
 56. National Institute of Standards and Technology. *Federal Information Processing Standard Publication 173* (Spatial Data Transfer Standard Part 1, Version 1.1), U.S. Department of Commerce, 1994
 57. The OpenGIS Abstract Specification - Topic 14: semantics and information communities (Version 4).
 58. The OpenGIS Guide, Third Edition
 59. A. Olive, Integrity Constraints Checking in Deductive Databases, *Proceeding of the 17th International Conference on Very Large Data Bases*, pp 513 – 523, 1991
 60. A. Olive, Integrity Constraints Specification, *Technical Report LSI*, University of Catalunya, Barcelona, Spain.
 61. Y. Ou, On Using UML Class Diagrams for Object-Oriented Database Design. Specification of Integrity Constraints. *UML 1998*: 173-188
 62. J. Owen, *STEP An Introduction*, Publisher Information Geometers Ltd, 1997.
 63. M. A. Pac, Dynamic Integrity Constraints Definition and Enforcement in Database: a Classification Framework, In *Integrity and Internal Control in Information Systems*, Edited by. Jajodia, W.

- List, G. McGregen and L. Strous, Lodon: Chapman & Hall, 1997.
64. A. Plexousakis, *Semantic Integrity Enforcement in Knowledge Base*, Ph. D. Qualifying Examination Paper, Department of Computer Science, University of Toronto, Canada, 1991
 65. A. Plexousakis, Compilation and Simplification of Temporal Integrity Constraints, In *Proceedings of the 2nd International Workshop on Rules in Database systems*, pp. 260-274, Athens, Greece, September 1995
 66. G. Pernul, A. M. Tjoa and W. Winiwarter, Modelling Data Secrecy And Integrity, *Data & Knowledge Engineering*, Vol. 26, pp. 291-308, 1998.
 67. A. Rosenthal, E. Sciore, Description, Conversion, and Planning for Semantic Inter-operability, in *Database Application Semantic*, Proceedings of the IFIP WG 2.6 Working conference on Database application Semantics, edited by Meersman R. and Mark L., 1995.
 68. S. K. Scherrer, Specification and Prototypical Execution of Integrity Concepts for Domain – Specific Database Management Systems, Ph.D. Thesis, University of Zurich, 1994
 69. K. Schewe, Consistency Enforcement in Entity-Relationship and Object-Oriented Models. *Data & Knowledge Engineering* 28(1): 121-140, 1998
 70. A. P. Sheth, Changing Focus on Interoperability in Information Systems: from System, Syntax, Structure to Semantic, in: *Interoperating Geographic Information Systems* edited by Goodchild *et al.*, Kluwer Academic Press, 1998
 71. S. Schwiderski, T. Hartmann, G. Saake: Monitoring Temporal Preconditions in a Behaviour Oriented Object Model. *Data & Knowledge Engineering* 14(2): 143-186,1994
 72. A.P. Sheth and J.A. Larson, Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 20(3): 183–236, September 1990.
 73. M. Staudt, A. Vaduva, T. Vetterli, Metadata Management and Data Warehousing, Internal report, URL: <ftp://ftp.ifi.unizh.ch/pub/techreports/TR-99/ifi-99.04.pdf.gz>.
 74. D. Schenck, P. Wilson, *Information Modeling: the EXPRESS Way*, Oxford University Press, 1994.
 75. V. C. Storey, H. Yang and R. C. Goldstein, Semantic Integrity Constraints in Knowledge-Based Database Design Systems, *Data & Knowledge Engineering*, Vol. 20, pp. 1-38, 1996.
 76. [TCY98] F. C. Tsang, J. Chiang, and W. Yang, Integration of Relations with Conflicting Schema Structures in Heterogeneous Database, Systems, *Data & Knowledge Engineering*, Vol. 27, pp. 231-248, 1998
 77. E. Teniente and A. Olivé, Updating the Knowledge Base while Maintaining their Consistency, *The VLDB Journal*, Vol.4, No. 2, 1995, pp. 193-241.
 78. C. Tuerker: *Semantic Integrity Constraints in Federated Database Schemata*. Ph. D thesis, Infix Verlag, St. Augustin, Germany, 1999.
 79. M. Vermeer, *Semantic Interoperability for Legacy Database*, Ph. D. Thesis, Netherlands: Department of Computer Science, University of Twente, 1997.
 80. J. Widom and S. Ceri, *Active Database Systems: Triggers and Rules for Advanced Database Processing*, Morgan Kaufmann Publishers, Inc. 1996.
 81. M. F. Worboy, S. M. Deen, Semantic Heterogeneity in *Distributed Geographic Databases*, in SIGMOD RECORD, Vol. 20, No. 4, Dec. 1991.
 82. S. B. Yoo and S. K. Cha, Integrity Maintenance In A Heterogeneous Engineering Database Environment, *Data & Knowledge Engineering*, Vol. 21, pp. 347-363, 1997.
 83. M. Zviran and C. Glezer, Towards Generating A Data Integrity Standard, *Data & Knowledge Engineering*, Vol. 32, pp. 291-313, 2000