

## DESIGN OF REPOSITORIES OF CONCEPTUAL SCHEMAS IN THE SMALL AND IN THE LARGE

**Carlo, Batini** University of Milano Bicocca, Via Bicocca degli Arcimboldi 8 20126 Milano, Italy  
batini@disco.unimib.it

**Riccardo, Grosso** CSI Piemonte, Italy  
Riccardo.Grosso@csi.it

**Guglielmo, Longobardi** Cnipa, Italy  
longobardi@cnipa.it

### **Abstract.**

*Large amounts of data are managed by organizations, available to be viewed and analysed from multiple perspectives, which becomes a fundamental resource to the effectiveness of the organizations. An organization can achieve full benefit from the available information by managing its data resource, through the planning of its exploitation and its maintenance. The concept of data repository fulfils these requirements, due to the fact that it contains the description of all types of data produced, managed, maintained and exchanged in an organization. Data descriptions should be organized in a repository to enable all the users of the information system to understand the meaning of data and the relationships among them. This paper describes two experiences of the use of repositories in Italy for central and local public administrations, showing that the “exact” methodologies used in the first case with large amount of resources have to be changed into a heuristic step when limited resources are available.*

*Keywords: Repository, Conceptual schema, Public Administration, eGovernment, Information management*

### **1 STRUCTURE OF ITALIAN PUBLIC ADMINISTRATION AND PREVIOUS EXPERIENCES OF CONCEPTUAL SCHEMA REPOSITORIES**

The goal of this paper is to describe two experiences of the use of a repository of conceptual schemas, related respectively to Central and Local public administration in Italy. In the first case, we were able, due to the large amount of resources available, to build the repository using an “exact” methodology. In the second case, referring to one of the 21 regions of Italy, the Piedimont region, due to the limited amount of available resources, several approximate techniques have been applied, that allow for fast prototyping of the local repository, to be refined by the domain expert, resulting in a resource consumption one order of magnitude lower than that with a traditional process.

The Italian government’s policy, in the past few years, similarly to many other governments in the world, has been to improve the quality of services to the citizen and businesses, by gradually improving services provided by information systems of its agencies. However, in the past the lack of co-operation between the departments led to the establishment of heterogeneous and isolated systems. As a result, two main problems have arisen: duplicated and inconsistent information, and difficult data access. Moreover, the government efficiency depends on the sharing of information between the departments, due to the fact that many of them are usually involved in the same procedures, while using different, overlapped, heterogeneous databases.

Therefore, in the long term, a crucial aspect for the overall project is to design a cooperation architecture that allows the administrations to share information in such a way as to be able to provide services to citizens and businesses on the basis of the “one stop shop” paradigm. A crucial aspect of such cooperation architecture is the data architecture: data has to be interchanged in an interoperable format, all the administrations have to assign the same meaning to the same data, achieving integration in the long term. This will spread information within the government branches and will result in a more easily accessible working environment, an increased quality of information management, and an improved state-wide decision making.

Data integration has to be achieved in the complex scenario of the Public Administration. The structure of the Public Administration (PA) in Italy consists of central and local agencies that together offer a suite of services to help citizens and businesses to fulfil their obligations towards the PA. Central PAs are of two types, Ministries such as the Ministry of the Interiors and Ministry of Revenues, and other central Agencies such as Social Security Agency, Accident Insurance Agency and the Chamber of Commerce. Main types of local PAs correspond to Regions (21), Provinces (about 100) and Municipalities (about 8.000).

The approach to cooperation between administrations followed in Italy to address these problems is based on the concept of Cooperative Information Systems (CIS), i.e., systems capable of interacting by exchanging services with each other. The general cooperative architecture for the back office nationwide CIS of the Italian PA is shown in figure 1.

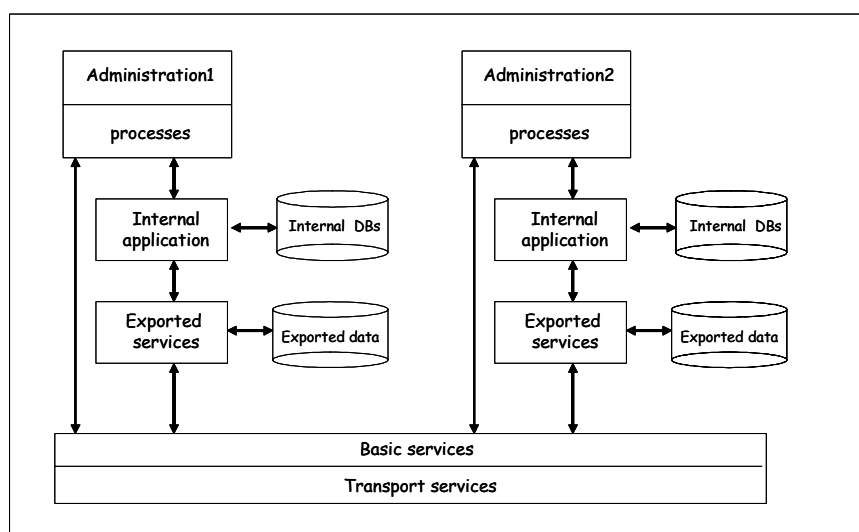


Figure 1.: The structure of the cooperative architecture

Besides transport and basic services, a cooperative services layer is shown, including application protocols, repositories, gateways, etc. The main idea is to define a *domain* as the collection of all the computing resources, networks, applications and data that belong to a specific administration. Each domain defines the set of cooperative interfaces that include data and application services made available to other domains in an interoperable format.

One of the first activities performed in the last decade, with the final goal of designing a suitable data architecture, has been the project of building an inventory of existing information systems operating within the Central PA in Italy. The activity was performed over about 500 relational data bases, whose logical schemas, through reverse engineering activities, were translated into corresponding conceptual schemas. A conceptual schema (schema in the following) is the representation of the reality of interest described in the database by means of representation structures in a conceptual model. We have chosen the Entity Relationship model as the conceptual model, where concepts are entities, relationships, attributes of entities and relationships, minimum and maximum cardinalities of

relationships, subset and generalization hierarchies. For further details on the Entity Relationship model see Elmasri and Navathe (2004). In order to provide a structure for such a large amount of schemas, a methodology for building repositories of conceptual schemas, described in Batini, Di Battista, and Santucci (1993) was used. We describe briefly this methodology in the next section.

A large repository of schemas such as the one built in the 90's in Italy can be used in the design of the new data architecture for several purposes:

1. analysis and resolution of redundancies;
2. creation or cleaning of national registries for the most relevant type of data, such as individuals, businesses, territory, etc.;
3. standardization of the representation of data in databases and data flows exchanged among administrations, with consequent improvement of the overall data quality;
4. reuse of concepts in the design of new databases and consequent improved semantic interoperability;
5. definition of the owner administration for the most relevant types of data, and creation of public and subscribe systems that discipline the update of data, in which only the owner is allowed to perform the update and the interested administrations subscribe the update.

Now, in order to achieve cooperation between the central and the local administrations, it is important to design a data architecture that covers both types of administrations, and, consequently, it is necessary to develop a similar repository for local administrations. For this reason, several regional administrations are now designing their own data architecture. The most advanced organizational context among local administrations is when they are coordinated by a regional agency that provides services to all or at least to the majority of them. This is the situation of administrations of the Piedmont region, where such a central agency exists, and is the CSI Piemonte consortium. But also in such a positive context, only logical relational schemas are available as the input to the process for the construction of the local repository; furthermore, limited human resources are usually available for the process. So, an approximate methodology and tools need to be arranged to allow the production of the repository. In this paper we describe this methodology and the experience we have achieved so far in applying it "in the large" to the context of the Piedmont PA, and compare such methodology with the "exact" one applied to the context of central PA.

The paper is organized as follows. In section 2 we discuss the structure of a repository of conceptual schemas as first described in Batini, Di Battista, and Santucci (1993). In section 3 we briefly recall the methodology for repository construction, using as case study the Central PA repository, and we discuss related work. In section 4 we apply in detail the methodology "in the small" to a few schemas of the Central PA Repository. In section 5 we describe the knowledge available for the design of Piedimont PA Repository. In section 6 we provide the approximate methodology "in the large", conceived for building, starting from the Central PA repository and the logical schemas of local administrations of Piedimont, a first version of the corresponding local repository. Section 7 discusses several experiments conducted to evaluate the efficiency and effectiveness of the approximate methodology. Section 8 discusses future research work

## **2 THE STRUCTURE OF THE CENTRAL PA REPOSITORY**

A repository, in the context of the paper, can be defined as a set of conceptual schemas, each one describing all the information managed by an organisation area within the information system considered. In particular, the data repository referenced in this paper uses the Entity Relationship model to represent conceptual schemas. However, a simple set of schemas does not display the relationships among schemas of different areas; the repository has to be organised in a more complex structure, through the use of the structuring primitives.

Structuring Primitives

The primitives used in our approach are: abstraction, view, and integration. *Abstractions* allow the description of the same reality at different descriptive levels, from detailed to abstract ones. We will call *refinement* the inverse primitive, that allows to proceed from abstract representations to more detailed ones. This mechanism is fundamental for a data repository, since it helps the user to perceive a complex reality step by step, going from a more abstract level to a local one. *Views* are fragments of schemas. They allow users to focus their attention just on the part of a complex reality of interest to them. *Integration* is the mechanism by which a set of local schemas is merged into a unique global schema, after solving all heterogeneities present in the input schemas. By jointly using these structuring primitives we obtain a repository of schemas. Each column of the repository represents an organisation unit while each row stands for a different abstraction level. The left column contains the schemas resulting from the integration of all the other schemas belonging to the same row (that correspond to views of the integrated schema). In figure 2 we show an example of repository, where the Production, Sales, Department Schemas are represented at different refinement levels respectively in the second, third and fourth column, while the Company schema in the first column is the result of their integration. Entities are represented by boxes, relationships are represented by diamonds, and generalization hierarchies by arrows. We name in the following *basic schemas* the conceptual schemas defined at the bottom level of the repository, *abstract schemas* the schemas at the upper levels.

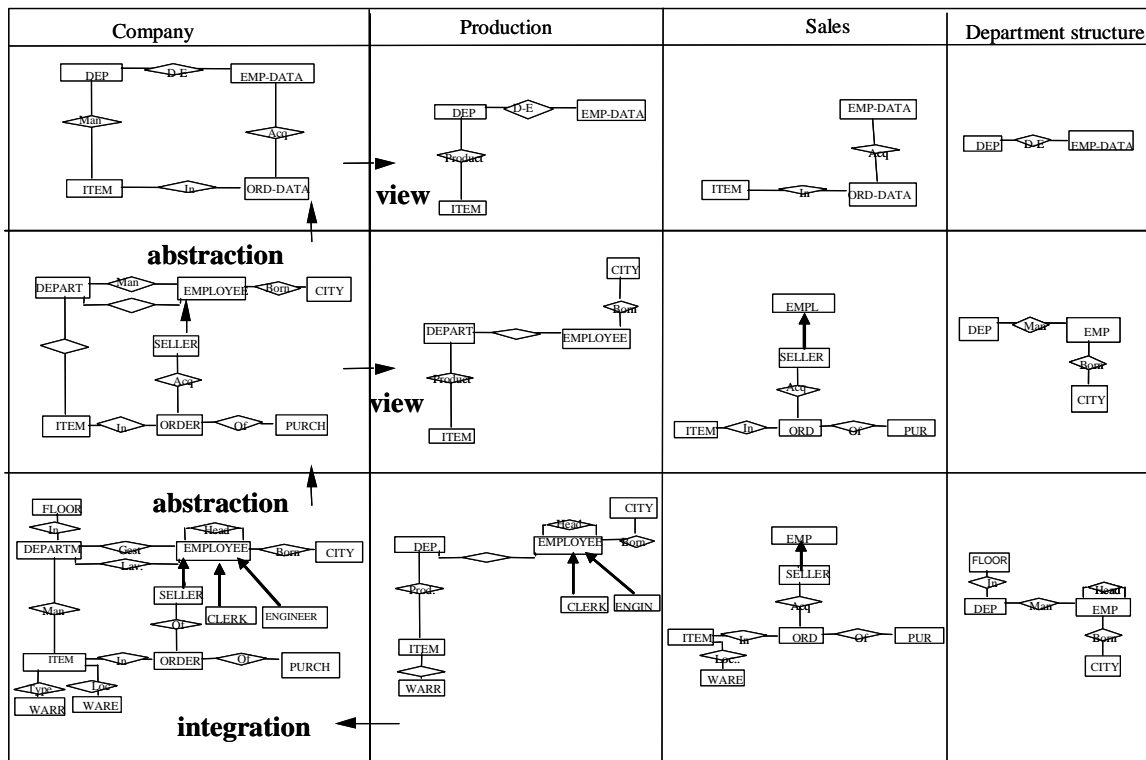


Figure 2. : An example of repository

In practice, when the repository is populated at the bottom level by hundreds of schemas, as in the case that we will examine in the following, it is unfeasible to manage the three structuring primitives, and the view primitive is sacrificed. Furthermore, integration and abstraction are applied together, resulting in the application of a new composed primitive, the *integration/abstraction* primitive. The integration/abstraction is iterated, producing a sparsely populated repository such as the one

symbolically represented in figure 3, where e.g. schema IS123 (Integrated Schema 123) results from the integration/abstraction of schemas S1, S2, and S3.

IS12345678											
	IS123				IS456				IS178		
		S1	S2	S3		S4	S5	S6		S7	S8

Figure 3.: A fragment of repository

### 3 A METHODOLOGY FOR THE DESIGN OF THE DATA REPOSITORY AND RELATED WORK

#### 3.1 Short description of the methodology

The repository structure described in the previous section has been adopted to represent the conceptual content of a wide amount of conceptual schemas related to the most relevant databases of the Italian central PA in an integrated structure. At the bottom level of the repository, approximately 500 conceptual schemas are defined, corresponding to the logical schemas of data bases of the 21 most relevant central administrations in Italy.

In order to build the whole repository, organized according to the repository structure defined in the previous section, the following procedure has been adopted, defined in more detail in Batini Di Battista and Santucci (1993). The methodology is made up of two phases, the first one for basic schema production, the second one for abstract schema production.

Phase 1. Basic Schema production – Starting from logical relational schemas or requirement collection activities, traditional methodologies for schema design have been used (see e.g. El Masri and Navathe (2004)), that lead to the production of about 500 basic schemas, with approximately 5.000 entities and a similar number of relationships.

#### Phase 2. Abstract Schema production

Phase 2 is made of two steps.

*Step 1. Schema clustering* - First, conceptual schemas representing the different organization areas are grouped in terms of homogeneous classes, corresponding to meaningful administrative areas of interest in central PA; 27 different areas have been defined: examples of areas are social security, finance, cultural heritage, and education.

*Step 2. Iterative integration/abstraction* - Each group of basic schemas is integrated and abstracted, resulting in a unique schema for each area, that populates the second level of the repository, resulting in 27 second level abstract schemas. In fig. 4 the different levels of the repository are represented, starting from the second level; for instance, the Internal security second level schema results from the integration/abstraction process, performed over 6 schemas corresponding to 130 concepts. The integration/abstraction process is iterated, producing higher level schemas, corresponding to more abstract matters, such as financial resources, human resources, social services, economic services, finally producing a unique integrated schema, that is further abstracted resulting at the topmost level of the repository in the schema shown in fig. 5, that represents the most significant concepts represented in the information systems of any PA, i.e. *Subject, Individual, Legal person, Property, Place, and Document*. The resulting pyramid of schemas provides a natural representation of concepts at different

abstraction levels, and, with suitable approximation, finds the common parts among databases pertaining to different agencies.

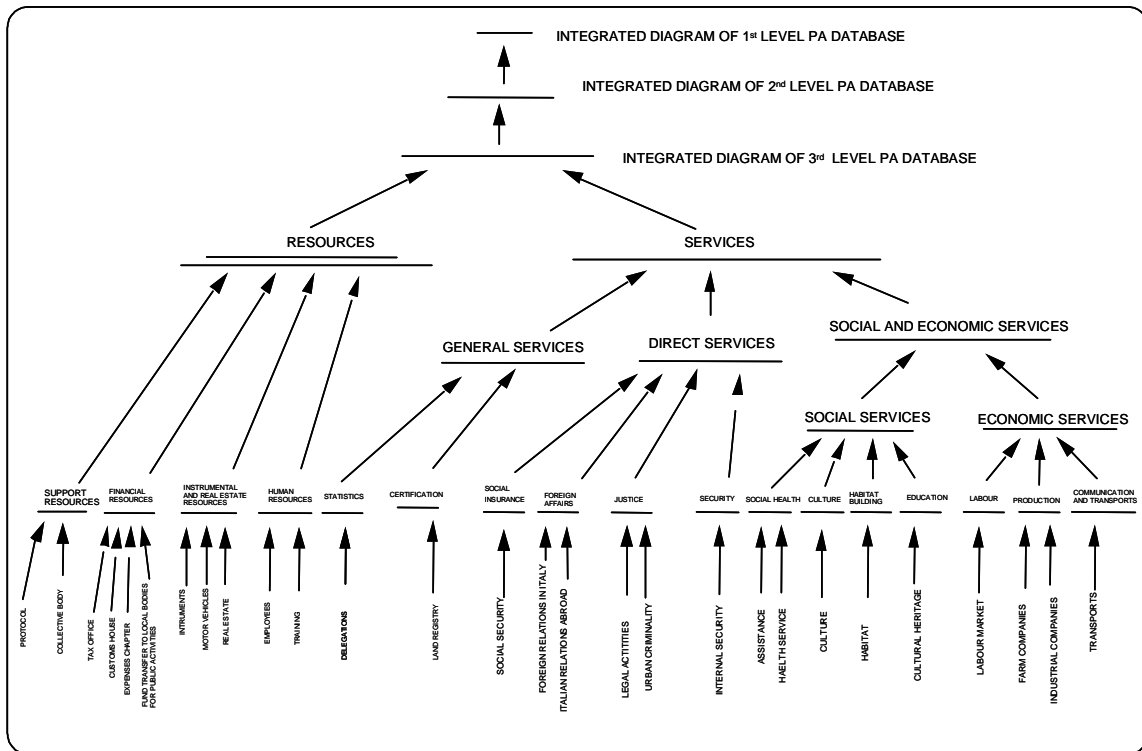


Figure 4.: The whole repository of schemas.

In order to produce the repository, about 200 person months were needed to produce in Phase 1 the 500 basic conceptual schemas, while about 24 person months were needed to produce in Phase 2 the 55 abstract schemas of the upper part of the repository (approximately 2 weeks per schema, both for the basic and for the abstract schemas).

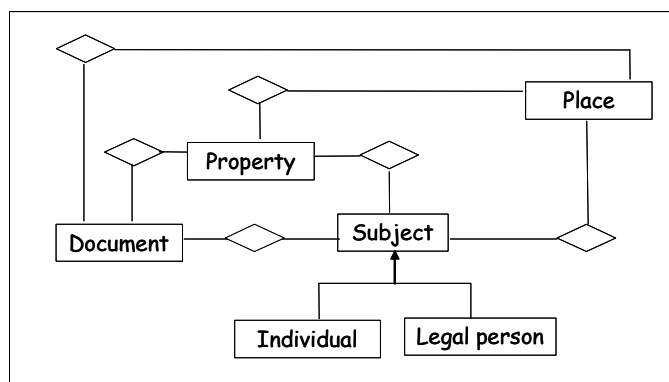


Figure 5.: The schema at the top level of the repository

### 3.2 Related work

The problem addressed in this paper, and the related conceptual tools, are not new in literature.

Concerning primitives and methodologies, using a descriptive model based on words and concepts, Mirbel (1997) proposes primitives for integration of object oriented schemas that generate abstract concepts as a result of the integration process. Shoval, Danoch and Balabam (2004) introduce the concept of conceptual schema package as an abstraction mechanism in the Entity Relationship model. Several effective techniques are proposed to group entities and relationships in packages such as dominance grouping, accumulation and abstraction absorbing.

Repositories of conceptual schemas are proposed in several application areas (e.g. in biosciences see Taxonomic Databases Working Group on Biodiversity Informatics (2004), for reuse see Ruggia and Ambrosio (1997)). Castano and De Antonellis (1997) propose criteria and techniques to support the establishment of a semantic dictionary for database interoperability. Similarity-based criteria are used to evaluate concept closeness and, consequently, to generate concept hierarchies. The techniques allow the analysis of conceptual schemas of databases in a federation and the definition and maintenance of concept hierarchies. Experimentation of the techniques in the PA domain is discussed. A data repository is used in Palopoli, Terracina and Ursino (2003) as the core structure of a mediator-like module supporting the user-friendly integrated access to available data resources. The core of the system is the extraction and exploitation of the inter-schema knowledge (in the form of inter-schema properties) relative to the involved database schemas.

Repositories of ontologies are proposed in several papers. The alignment and integration of ontologies is investigated in Wang and Gasser (2002), Di Leo, Jacobs and DeLoach (2002), Farquhar, Fikes, Pratt, and Rice (1995), where information integration is enabled by having a precisely defined common terminology. A set of tools and services supports the process of achieving consensus on such common shared ontologies by geographically distributed groups. Users can document the decisions of the domain expert made in the fifth step. quickly assemble a new ontology from a library of modules. Repositories of ontologies for the public sector organizations are proposed in Slota et al. (2003). The repository was used in a system supporting organizational activity by formalizing, sharing and preserving operational experience and knowledge for future use.

Our approach is new with respect all of our approaches, in aspects related to feasibility and resource constraints, and in the heuristics proposed in section 6.

## 4 THE INTEGRATION/ABSTRACTION STEP “IN THE SMALL”

In this section we expand upon the methodology for the Central PA Repository discussed in the previous section, describing the integration/abstraction step in detail, and applying it to a small set of schemas. The methodology for integration/abstraction is made up of the following steps.

### 1.Integration

This step is inspired to the methodology first described in Batini and Lenzerini (1984). It consists of:

1.1.Pairwise comparisons of entities of local schemas - The aim is to discover and solve every type of conflict between data representations in different schemas. Two main activities may be distinguished:

**a. name conflict analysis** to establish naming correspondences for concepts. There are basically three sources of name conflict: synonyms, homonyms, and multiple names. The first ones occur when schema objects (in different schemas) with different names represent the same concept, while the second occur when schema objects with the same name represent different concepts. Multiple names are homonyms for a concept in one schema, while correspond to different concepts in another schema. Therefore, whenever name conflicts are detected a renaming is required to solve the conflict;

**b. structural conflict analysis** to discover conflicts between different representations of the same concept. The use of an entity and an attribute to represent the same concept in two different schema is a typical example of structural conflict. Each structural conflict can be solved by applying an equivalence transformation to the schemas involved (a transformation which does not change the schema contents).

1.2.Production of amended schemas - At the end of this stage we obtain a set of “amended schemas” that can be syntactically integrated, since all name and structural conflicts have been solved. The resulting schemas are suitably transformed in order to conform to each other.

2.Abstraction

The production of the abstract schema can be seen as a two-step procedure in which we merge the amended schemas and at the same time we select the abstract concepts corresponding to objects in the original schemas, so that the integrated/abstracted schema is built.

All the above activities are now briefly described in order to improve the understanding and the subsequent application of the methodology to a case study referred to the Land Agency of the Italian Ministry of Revenues. This Agency is in charge of the valuation of real estate property, to determine direct and indirect tax assessment and to issue real estate certifications. Moreover, this Agency administers and records all state properties in regard to their financial affairs; it is in charge of: the acquisition of new state properties, the management of properties when authorised, the care and supervision of state properties, and the maintenance of an inclusive inventory.

Seven databases have been located within the Land Agency information system, three in the General Land Office and four in the State Property Office. For reasons of space, in the construction of a Repository in the small (see figure 9) we will focus our attention on the following schemas:

1. General Land Office: Urban Database (see fig. 6);
2. General Land Office: Land Database (see fig. 7);
3. General Land Office: Mortgage Registry Database (see fig. 8);
4. State Property Office: Real Estate Database (see fig. 9).

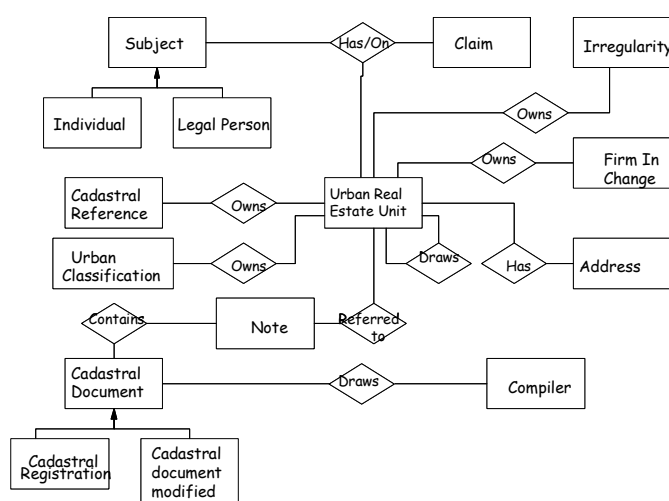


Figure 6.: General Land Office: Urban Database

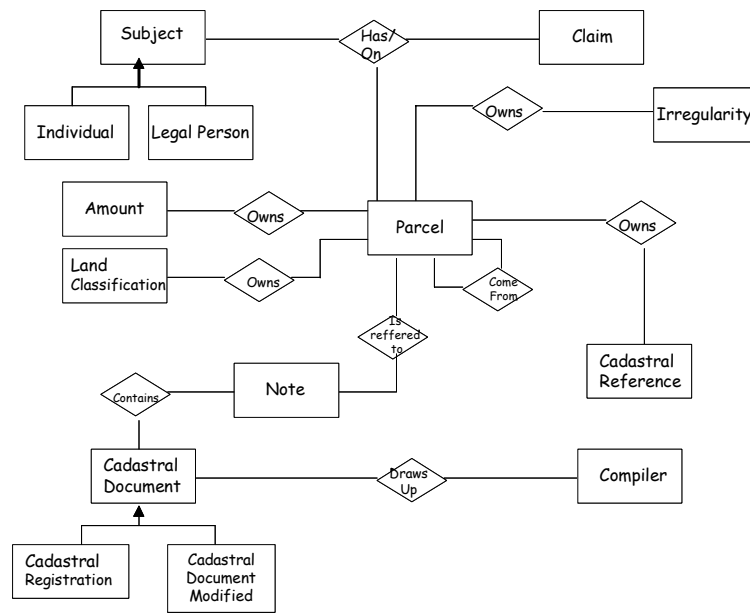


Figure 7.: General Land Office: Land Database

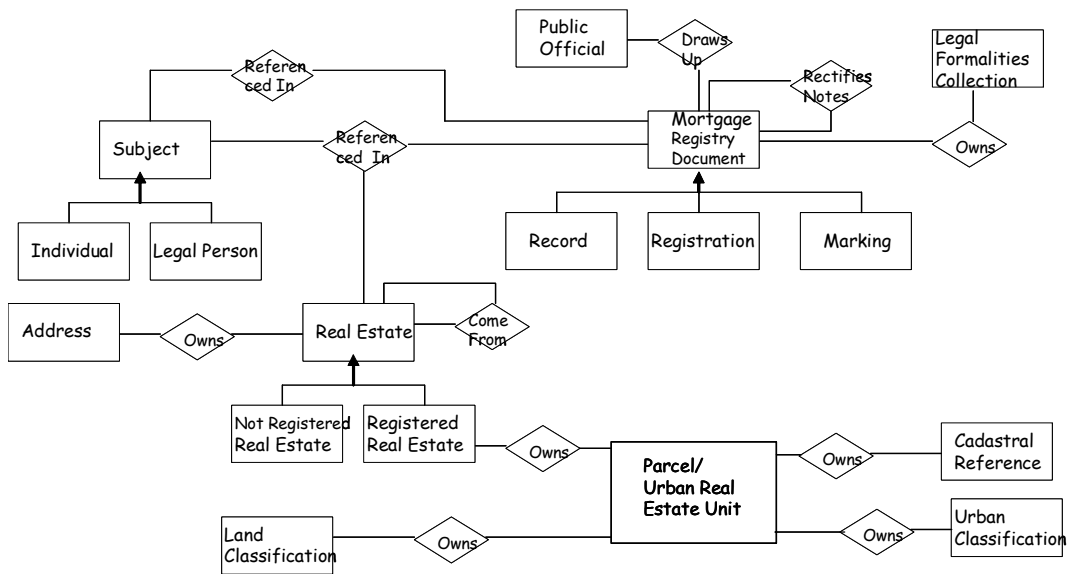


Figure 8.: General Land Office: Mortgage Registry Database

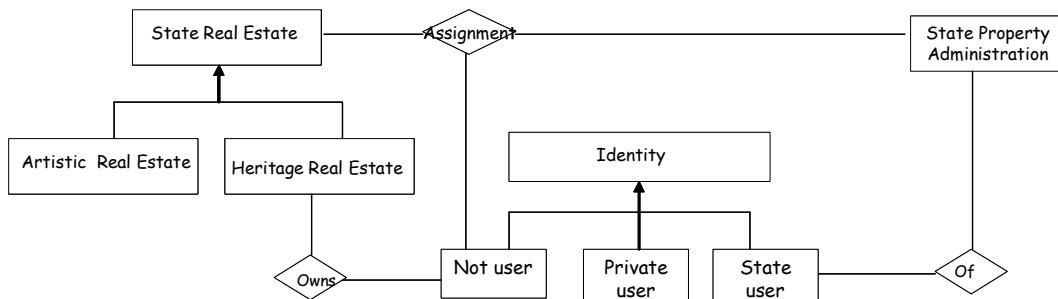


Figure 9.: State property Office: real estate Database

Schemas are shown with entities, relationships, subsets and generalization hierarchies; identifiers, attributes and cardinalities of relationships are not shown. The four schemas are the starting point for the activity of data repository design. We now examine the different steps. To provide a detailed description of every single step required by the method, however, it would become an overly task. Therefore, only some of the activities required to obtain the amended schemas will be described below.

## **1.Integration**

### **1.1 Pairwise comparisons of entities**

#### a. Name conflict analysis

Considering the starting local schemas, we discover a “multiple name” conflict represented by the entity *Parcel/Urban Real Estate Unit* in the Mortgage Registry schema and the entities *Parcel* and *Urban Real estate Unit* listed in the Urban and Land schemas of the General Land Office. An examination of the concepts leads to the conclusion that *Parcel* and *Urban Real Estate Unit* are child-entities of a more general concept that can be referred to as *Real Estate Unit*. We decide to keep both the entities in the General Land Office schemas and replace the entity of the Mortgage Registry with a generalization hierarchy having *Real Estate Unit* as father entity, and *Parcel* and *Urban Real Estate Unit* as child entities.

Secondly, we discover a synonymy among entities *Subject* in the first three databases and the entity *Identity* in the State Property Office schema. We solve the synonymy by changing the name *Identity* into *Subject* in this last schema.

#### b. Structural conflict analysis

Looking at identifiers of entities, we detect a case of concept incompatibility between the entity *Subject* in the schemas representing the General Land Office databases and the entity with the same name in the State Property Office schema. The first group of entities is identified by an account number while it has the owner tax number as one of its attributes. On the contrary, the entity in the State Property Office schema is identified by a tax number. This conflict can be solved by choosing tax number as identifier and keeping account number as an attribute with (0,1) as minimum and maximum cardinalities. In fact the relationship between identity and tax number has in both cases cardinalities (1,1) and therefore this attribute can be used as the identifier of the entity. Moreover, the entity *Subject* in the State Property Office schema has more attributes (name/firm name, personal data, office location) than the one in the first three schemas. We can find these attributes in those schemas as well, because they are linked to the child-entities of *Subject: Individual* and *Legal entity*. We solve the conflict including the hierarchy in the State Property Office schema, to be added to the other hierarchy, including entities *Not user*, *Private user*, *State user*.

### **1.2. Production of amendend schemas**

At this stage of the methodology all name and structural conflicts have been solved, obtaining the amended schemas. The next step is to merge these schemas in order to build the global one.

## **2. Abstraction**

At this point we have finally obtained an integrated schema. We now have to abstract groups of semantically related concepts into higher level concepts, obtaining at the end of the process a global description of the Land Department information content, shown in figure 10. We see that, in order to keep in the abstract schema the most relevant concepts appearing in the four original schemas, we have to make several choices. First, concerning the cluster of “Subject” concepts, among the three

concepts *Not user*, *State Administration User* and *Private User*, we decide to keep only the first two concepts, since they are linked with relationships to other concepts of the schema. The two concepts are linked with *Subject* by means of subsets.

Finally, all the concepts directly related to the cluster of “Subject” concepts, such as *Irregularity*, *Cadastral Reference*, etc. are collapsed and included in the *Real Estate Unit* concept in the abstract schema. Concerning the cluster of “Real estate” concepts, we decide to keep in the schema *Real Estate* and *Real Estate Unit*, due to the importance of the two concepts in the whole domain, *State Real Estate* due to its role in the State property sub domain, and finally *Parcel* and *Urban Real Estate Unit* do to their presence in all the three schemas of the Land Office.

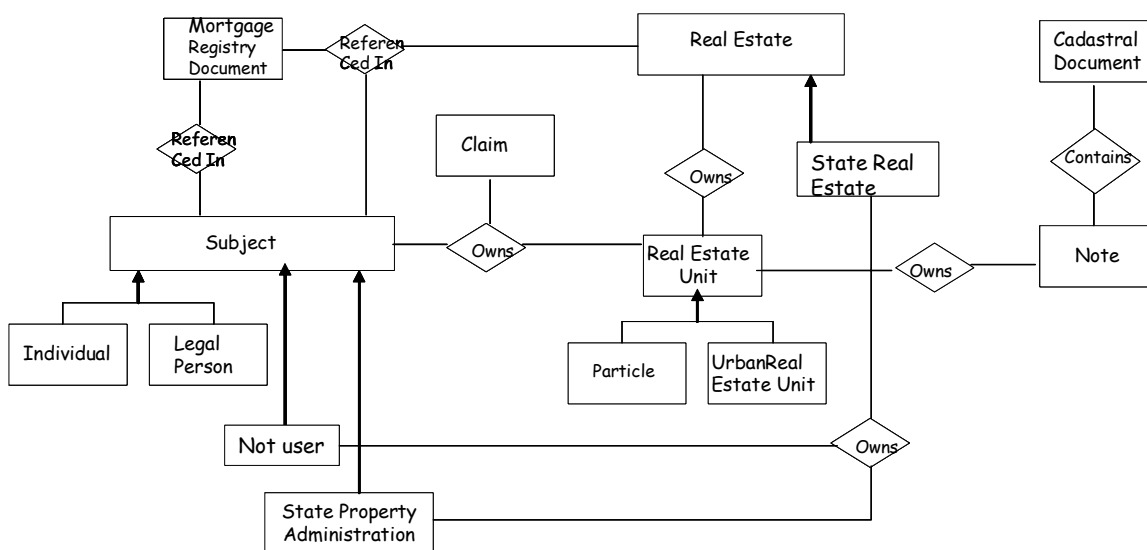


Figure 10.: The integrated/abstract schema

## 5 THE REPOSITORY OF PIEDMONT LOCAL ADMINISTRATIONS: BASIC KNOWLEDGE AVAILABLE

In this section we describe in more detail the knowledge available for the design of the Piedmont Local Administration Repository (LPA Repository in the following) and the assumptions that have been made in the activity.

A first relevant input available is the Central PA Repository of schemas (CPA Repository in the following), made of basic schemas and abstract ones. Such Repository was built in the years 1994-96, so it does not consider the entities that have been created in recent years, and, most importantly, represents several entities that concern functions subsequently transferred to local administrations. A look up table of CPA most frequent attributes, extracted from the CPA Repository of basic schemas is another source for the procedure.

A second input concerns the Piedmont databases. As we said, Piedmont local PA is centrally served by a unique consortium, that created approximately 450 databases in the last years, whose logical schemas are documented in terms of: relational database schemas, tables, description of tables, referential integrity constraints defined among tables, attributes, definitions of attributes, identifiers.

The basic sources of knowledge available for the production of the LPA repository, as results from the above discussion, are very rich, but characterized by two significant heterogeneities: the conceptual

documentation concerns central administration, while the prevalent documentation for local Piedmont administration concerns logical schemas.

A second relevant condition of our activity has been budget constraints. For the first year of the project we had only one tenth of the resources that were available for the construction of the central repository. Therefore, in conceiving the methodology for the LPA production, we made a few significant assumptions, we have used heuristics and approximate reasoning, in order to reduce human intervention as much as possible.

A first assumption we made has been that, while basic schemas of the CPA Repository and the LPA Repository may probably differ, due to the different functions between the central and local administrations, the similarity should be much higher among the abstract schemas of the CPA Repository and basic + abstract schemas of the LPA Repository.

In consequence of the above assumption and resource constraints, we decided to use a more manageable knowledge base in some steps of the methodology than the CPA Repository 500 basic schemas + the 50 abstract schemas. Such schemas can be represented in terms of a much more dense conceptual structure, that corresponds to the generalization hierarchies that have at their top level the five concepts defined in the schema of figure 5, and having at lower levels the concepts in more refined abstract schemas and basic schemas, obtained applying top down the refinements along the integration/abstraction hierarchy. We show in figure 11 a fragment of one of the hierarchies, the one referring to individuals.

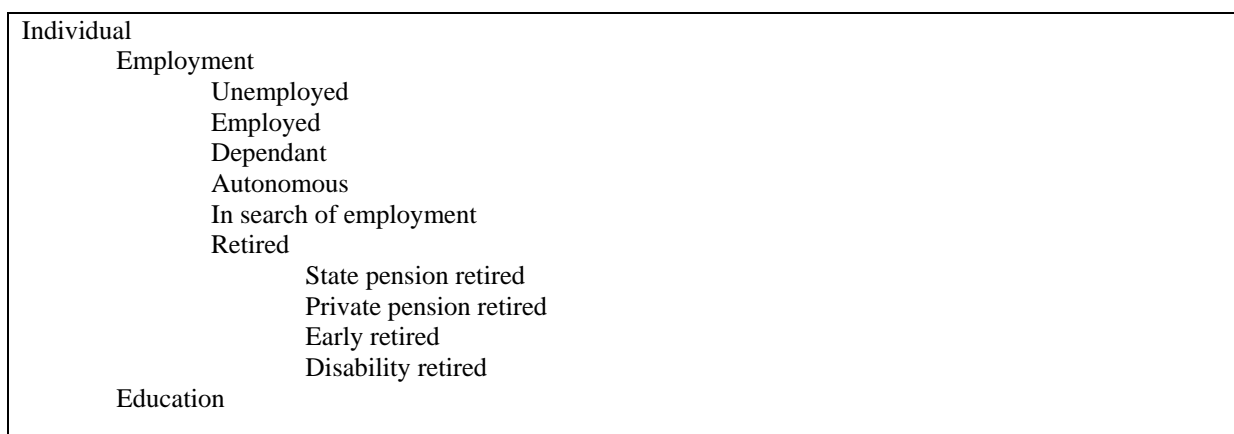


Figure 11.: A fragment of the Individual generalization hierarchy

Therefore, a second idea we implemented has been to use, besides the basic schemas and the abstract schemas, the five generalization hierarchies of Individual, Legal Person, Property, Document, Place. As a consequence of the above assumptions, constraints and choices, the inputs to the methodological process, shown in figure 12, have been:

1. the Repository of 550 Central PA basic + abstract schemas
3. the five central PA generalization hierarchies
4. the logical schemas of the 450 local PA databases.

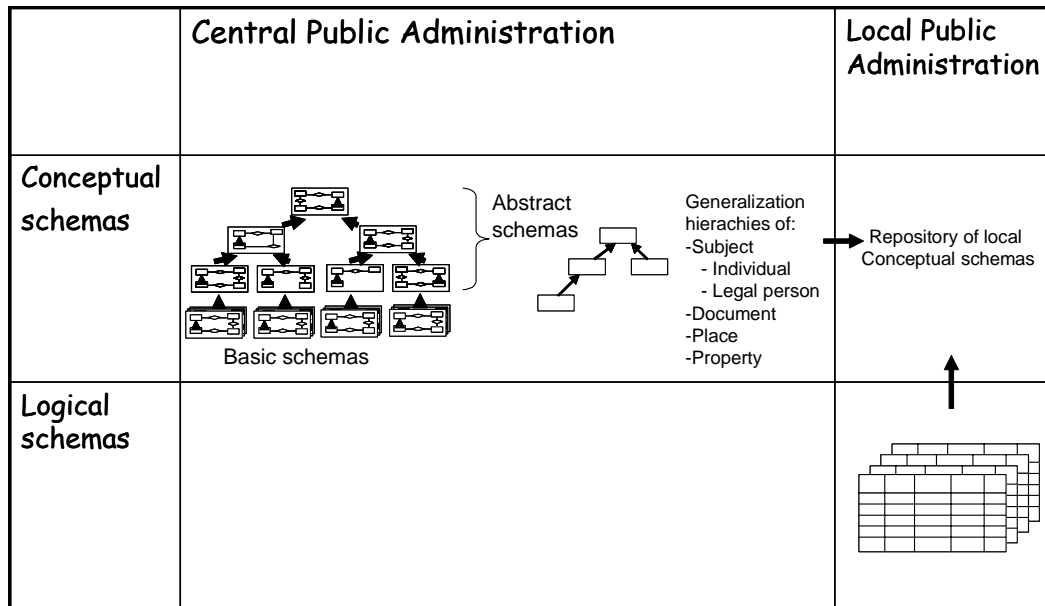


Figure 12.: Input knowledge for the production of the Repository of local conceptual schemas

## 6 THE METHODOLOGY FOR THE CONSTRUCTION OF THE LOCAL REPOSITORY

First, we provide the rationale of the methodology, and then we detail specific steps. The methodology (whose first version appears in Batini and Grosso (2005)) follows a mixed approach in building the basic schemas and the upper part of the repository, and consequently, can be seen divided in two phases. For each local logical schema, available conceptual/central and logical/local knowledge is used in Phase 1 to reverse engineer a basic local conceptual schema. Then, in Phase 2, the abstract schemas of the local repository are built. We now examine the two phases in detail, while in section 7 we discuss the result of a first set of experiments performed for Phase 1, in order to measure the effectiveness of the methodology and tune it. Each step is described with a common documentation frame, describing the inputs to the step, the procedure, and in some cases, when relevant, the outputs of the step. An example is provided, related to a logical schema concerning grant monitoring of industrial business activities.

### Phase 1: Construction of basic schemas

#### Step 1. Extract entities and attributes

Inputs: CPA generalization hierarchies of concepts, one LPA logical schema

Names of entities in hierarchies and names of attributes in the look up table of the CPA Repository most frequent attributes are compared with names and description of each table, and names and descriptions of attributes of the logical schema. The comparison function makes use presently of a simple distance function among the different text strings. A point in a four dimensional space is associated to each concept extracted, defined as

$$P(\text{concept}) = \langle \# \text{table\_names}, \# \text{table\_descriptions}, \# \text{attribute\_names}, \# \text{attribute\_descriptions} \rangle$$

where each item, e.g. #table names corresponds to the number of table names having a distance function lower than a fixed threshold<sup>1</sup>. A concept is selected as potential entity or attribute if the sum of the four items is greater than a second threshold<sup>2</sup>. In order to decide if a concept to be an entity or an attribute, we compute the distance in the four dimensional plane between P(concept) and the two points:

- $P_{entity} = \langle \#table\_names, \#table\_descriptions, 0, 0 \rangle$
- $P_{attribute} = \langle 0, 0, \#attribute\_names, \#attribute\_descriptions \rangle$

and assign entity or attribute according to the closer point. We have to decide finally for each attribute  $A_i$  which is the corresponding entity  $E_j$ . This step is performed assigning  $A_i$  to the closer entity, assuming as distance the corresponding sets of table names, table descriptions, attribute names, attribute descriptions extracted. The entities and corresponding frequency of matching are sorted, and a threshold<sup>3</sup> is fixed: all the entities with the frequency over threshold<sup>3</sup> are selected, resulting in a first draft schema made only of entities. The output is a draft schema made of disconnected entities and related attributes.

#### Step 2. Add generalizations

Inputs: the draft schema obtained in the previous step and the four CPA generalization hierarchies.

Visit the generalization hierarchies and add to the draft schema subset relationships present in hierarchies, defined among the entities in the draft schema.

#### Step 3. Extract relationships

Inputs: the draft schema + all the basic schemas in the CPA repository

Entities of the draft schema are pair wise compared with all the basic schemas in the CPA repository. For each pair of entities  $E_1$  and  $E_2$  several types of relationships are extracted from the basic schemas:

a. relationships defined exactly on  $E_1$  and  $E_2$ ;

b. relationships corresponding to chains of relationships defined among pairs  $E_1-E_i$ ;  $E_i-E_{i+1}$ ; ...;  $E_{i+j}-E_2$ ;

c. relationships defined among entities  $E_1^*$  and  $E_2^*$  corresponding to ancestors of  $E_1$  and  $E_2$  in the four generalization hierarchies.

Relationships collected in steps a-c are sorted according to the frequency of names. Here we have two possibilities:

- a. The most frequent name is chosen as the name of the relationship;
- b. The name is assigned by the domain expert.

#### Step 4. Check the schema with referential integrity constraints defined among logical tables

Input: the draft schema + constraints defined in tables

For each referential integrity constraint defined among two tables  $T_1$  and  $T_2$  in the logical schema, it is checked whether  $T_1$  and/or  $T_2$  have been already selected as entities in the draft schema, and in case added as new entities. Furthermore, it is checked whether a relationship is defined among the entities, and if necessary it is added.

#### Step 5. Domain expert check of the draft schema and construction of the final schema

Input: the draft schema

In this step the schema produced by the semi automated process is examined by the knowledge domain expert that may add new concepts, cancel existing concepts, or else modify some concepts. Since step 5 is performed after addition of relationships and entities resulting from integrity

constraints, it may occur that too many concepts have been added, and the manual check of the domain expert leads to delete concepts. Sometimes new concepts are added, resulting in an enriched schema whose kernel is the initial schema. More frequently the schemas which were obtained after the integrity constraints check and after the domain expert check coincide.

Output: the: final schema

We show in figure 13 the schemas obtained as a result of the execution of steps 1 to 5 of the methodology in our example (we do not show the attributes). In this case, schemas obtained after integrity constraints check and after domain expert check coincide. We have shown that this is a common result of the procedure also in other experiments.

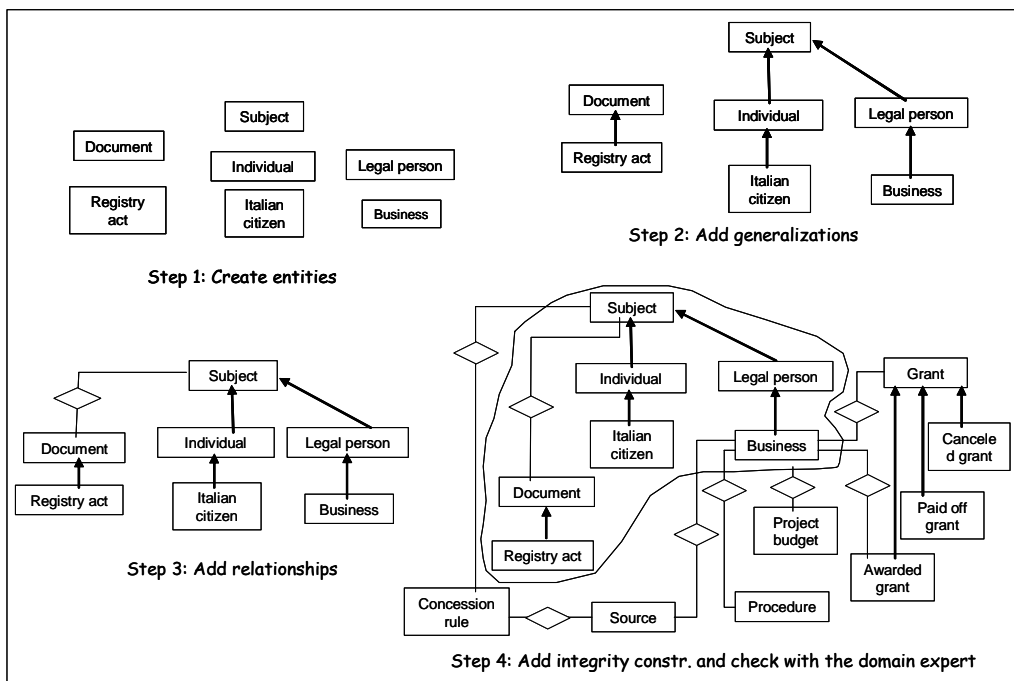


Figure 13.: Schemas obtained after steps 1-5

## Phase 2: Construction of abstract schemas

This activity can be performed with different strategies, all of them sharing the following conjecture. The initial schema obtained after steps 1-3 inherits high level abstract knowledge from the CPA Repository and basic knowledge from the LPA logical schemas, while the enriched schema obtained in steps 4-5 encapsulates basic knowledge from the local PA logical schemas. So, we may conjecture that the initial schema  $S_{as_i}$  is a candidate for abstract schema for the upper levels of the repository, while the enriched schema  $S_{basic}$ , being a more detailed description of a logical schema, populates the basic level of the repository (see figure 14).

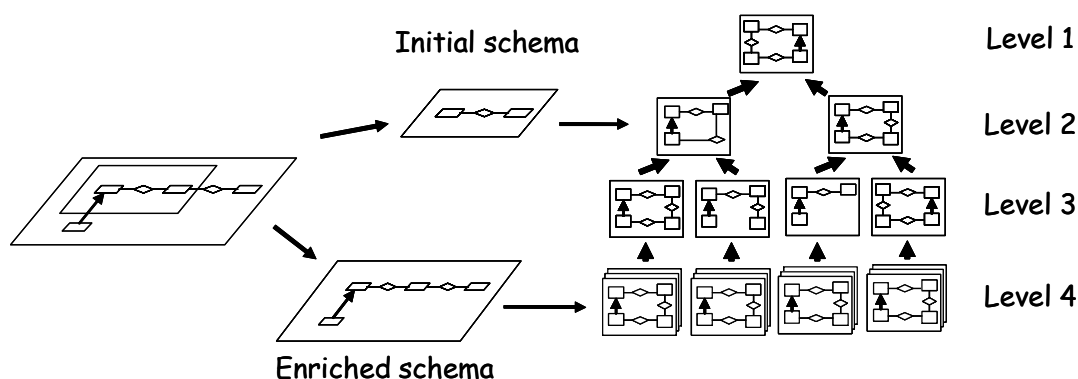


Fig 14.: Presumable locations in the repository of initial and enriched schemas

Now, remember that, for each local schema  $LS_i$ , all the entities in  $Sas_i$  belong to the CPA generalization hierarchies. So, we may associate an abstraction level to each  $Sas_i$  that, intuitively, captures the relative position of its entities with regard to the four hierarchies. More formally, in order to compute the abstraction level we may use the following algorithm:

1. Split the set of entities of  $Sas_i$  into four groups corresponding to the subsets of entities belonging to each of the four hierarchies.
2. For each group  $G_j$  compute its abstraction level as the sum of the distances from the topmost level in the hierarchy ratio the number of concepts.
3. The abstraction level of  $Sas_i$  is the weighted average of the four abstraction levels.

An abstraction level can be attached to each schema in the CPA repository of figure 4, defined with a similar algorithm. Correspondingly, we may associate an average abstraction level  $AL$  ( $layer_k$ ) to each  $layer_k$  in the CPA Repository. At this point we may define the procedure for Phase 2.

Input: the actual version of the LPA repository and a new initial and enriched schema.

For each initial schema  $Sas_i$  and enriched schema resulting from the application of steps 1-5

1. Make the enriched schema an  $S_{basic}$  schema, assigning it to the cluster of schemas associated to the closer matter, where closeness can be measured on the basis of common concepts with schemas in the different clusters in the CPA repository;
2. Calculate the abstraction level  $AL_i$  of the initial schema  $Sas_i$ ;
3. Associate  $Sas_i$  to the layer  $L_k$  of the CPA Repository with the abstraction level closest to  $AL_i$ .
4. For each schema  $SCK_j$  in the layer  $L_k$  of the CPA Repository, extract concepts of  $SCK_j$  that appear also in  $Sas_i$ , and assign them to  $SL_{kj}$ , the schema that at the end will substitute  $SCK_j$  in the LPA repository, building a new enriched version of  $SL_{kj}$ .

Output: final LPA repository

The above strategy has the advantage of being fully automatic, while producing potentially incomplete schemas, due to the limited number of concepts (only concepts present in generalization hierarchies) used for building abstract schemas. We may conceive two other strategies for this step, that also make use of enriched schemas, but need expert effort. We describe the two strategies, referring also to figure 15.

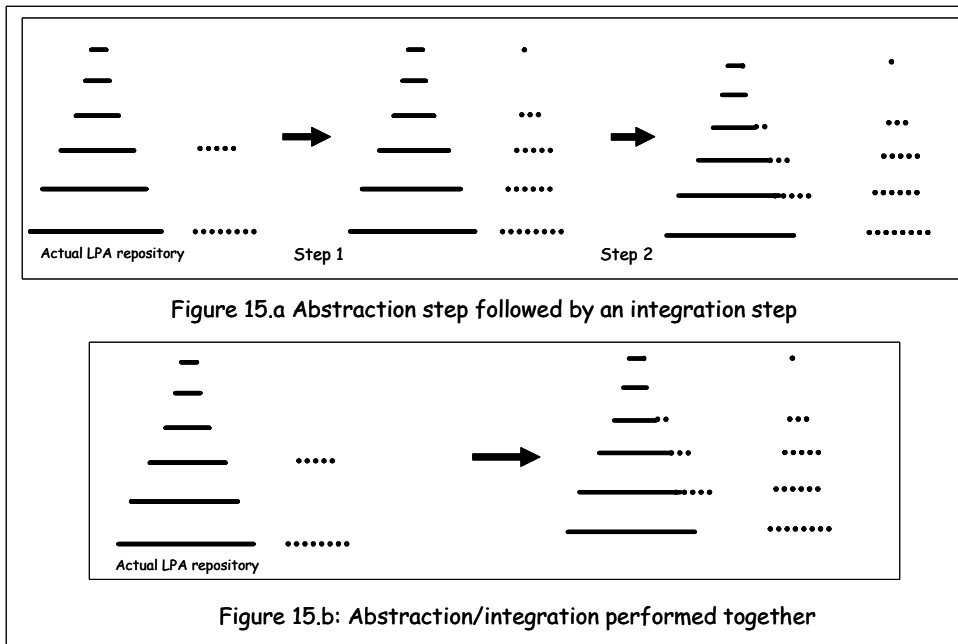


Fig 15.: Two possible strategies to update the LPA repository

In the first strategy, starting from the initial schema and the enriched schema we first (fig 15.a) complete the “local” repository of abstract schemas corresponding to the enriched schema (see in figure 16 the result of the step for our example), we then integrate the local repository with the actual one: it may happen that we have to update, due to similarities between concepts, the abstract schemas of the actual repository, or else add new schemas, autonomous in respect to the previous ones.

In the second strategy the new repository is obtained through abstraction/integration activities on the actual LPA repository and the initial and refined schemas (see fig 15.b).

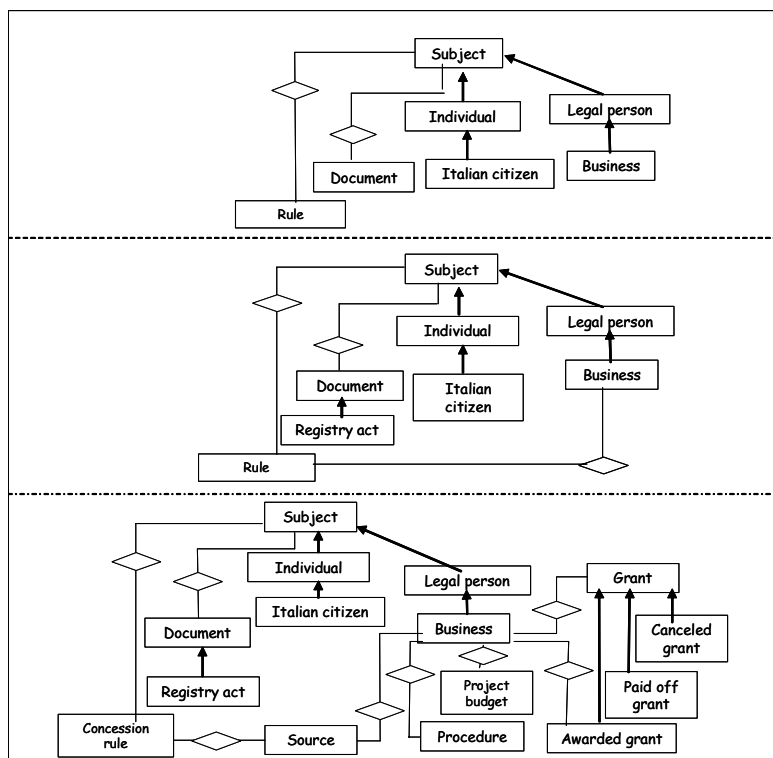


Fig 16.: abstract schemas obtained from the basic schema

## 7 EXPERIMENTS

In the present stage of the project we have experimented the Phase 1 of the methodology: Production of basic schemas, in three different areas and nine related sectors. They are:

### Businesses

- Grants to industrial businesses
- Economic/productive activities
- Farms register

### HealthCare

- Abortions/miscarriages statistics

### Regional territory

- Provincial Road register
- Dam register
- Laws on water management
- Hydrogeological network
- City toponymic

There are a total of approximately 350 tables of the nine databases which correspond to 2% of the total. We were interested in measuring two relevant qualities of the process:

1. the *correctness* of the conceptual schema in respect to the “true” one, i.e. the schema that could be obtained directly by the domain expert through a traditional analysis or else a reverse engineering activity. Correctness is measured with an approximate indirect metrics, corresponding to the

percentage of new/deleted concepts in the schema produced by the expert at the end of step 5 with respect to concepts produced in the semi automatic steps 1-4.

2. the *completeness* of the conceptual schema with respect to the corresponding reengineered logical schema. Completeness is measured by the percentage of tables that are captured in steps 1-5, in comparison with the total number of tables, after excluding tables not carrying relevant information, such as redundant tables, tables of codes, etc.

Table 1 summarizes main results of experiments. Concerning correctness, in general the schemas obtained after step 4: check with integrity constraints, and after step 5: domain expert check are very similar, i.e. domain experts tend to confirm and consider complete entities and relationships added in the previous step. The overall figure for the nine experiments results in more than 80% of concepts common to the two types of schemas. We see also that the add constraints step introduces approximately 30% of new concepts in comparison with the extract entities step. Consequently the joint application of the CPA knowledge and LPA knowledge reveals that it is effective. These are, in our opinion, encouraging results, considering the highly heuristic nature of the methodology.

Concerning completeness, results are less reassuring. On the average, only 50% of the tables are captured. This value changes significantly in the different areas. Furthermore, as was to be expected, completeness decreases significantly when the referential integrity constraints are not documented or partially documented, resulting in lower quality (completeness) conceptual schema when the input schema is characterized by poor documentation. Apart from the quality of the documentation, another cause of reduced completeness is the static nature of generalization hierarchies used in step 1, and the unequal semantic richness in representing related top level concepts. For instance, in the initial Subject hierarchy, 20 concepts represent individuals, while only 3 represent legal persons. An improvement which we are presently applying concerns their incremental update with abstract concepts generated in Phase 2. Such enriched hierarchies are progressively reconciled and brought near to hierarchies characteristic of local administrations, resulting in a corresponding more effective selection mechanism.

Step	# of tables extracted	% of tables extracted
Create entities	172	30
Add constraints	219	41
Domain expert check	275	51

*Table 1: Experiments results*

A final comment on resources. The amount of resources spent in the experiments has been on the whole 30 person/days, corresponding to 3 person/day per schema. About 30% of the time has been spent in steps 1-4, and 60% of the time has been spent in manual checking. So, the domain expert has been engaged for 2 days per schema. We have to add to this variable cost a fixed cost of a 3 days course. We may expect a greater efficiency as long as the activity proceeds, and fix in a person day the average final due effort, significantly lower than the typical 2-3 person/weeks needed for the traditional design of one schema.

## 8 CONCLUSIONS

In this paper we have proposed a structure for repositories of conceptual schemas, that makes use of integration/abstraction primitives in order to provide a structure to a large number of conceptual schemas. Using the integration/abstraction primitive, basic schemas are iteratively transformed into abstract schemas. Furthermore, we have compared two methodologies for the construction of repositories. The first methodology is "exact", but requires relevant resources, while the second one is approximate, but needs one order of magnitude lower resources. The two methodologies are compared

in the activity of production of basic schemas, according to correctness and completeness criteria. Results are satisfactory for correctness, while for completeness we are presently improving the heuristics.

We are now analyzing lessons learned and improving the methodology along the lines discussed in previous sections. We are also investigating new techniques that use more complex similarity measures in matching between generalization hierarchies and logical schemas. Furthermore, since some of the local PA schemas (and corresponding hierarchies) have been independently developed, especially in the regional territory area, we are using such schemas as training examples to tune semiautomatic steps of the methodology and adopt similarity measures.

We are also developing a tool that, in its first release, will fully automate the first four steps of the activity of production of basic schemas, and can document the decisions of the domain expert made in the last step of the activity.

## References

- Batini C. and Lenzerini M. 1984. 'A methodology for data schema integration in the Entity Relationship model'. *IEEE Transactions on Software Engineering*.
- Batini C., Di Battista G., Santucci G. 1993. 'Structuring primitives for a dictionary of entity relationship data schemas'. *IEEE Transactions on Software Engineering* 19(4).
- Batini C., Castano S., De Antonellis V., Fugini M.G., Pernici B. 1996. 'Analysis of an Inventory of Information systems in the Public Administration'. *Requirements Engineering* 47-62
- Batini C., Castano S., Pernici B. 1996. 'Tutorial on Reuse Methodologies and Tools'. Entity Relationships International Conference, Cottbus, Germany.
- Batini C., Longobardi G., Fornasiero S. 1987. 'An Experience of Integration of Conceptual Schemas in the Italian Public Administration'. International Conference on the Entity Relationship Approach 313-322.
- Batini C., Grosso R. 2005. 'Reuse of a repository of conceptual schemas in a large scale project'. Proceedings of EMMSAD Conference, Porto.
- Castano S., De Antonellis V. 1997. 'Semantic dictionary design for database interoperability'. 13th International Conference on Data Engineering, University of Birmingham, Birmingham, U.K.
- DiLeo J., Jacobs DeLoach T. 2002. 'Integrating Ontologies into Multiagent Systems Engineering'. Fourth International Bi-Conference Workshop on Agent-Oriented Information Systems, Bologna (Italy).
- Elmasri R., Navathe S.B. 2004. 'Fundamentals of Database Systems', 4<sup>th</sup> Edition, Pearson Education Inc., Addison Wesley.
- Farquhar A., Fikes R., Pratt W., Rice J. 1995. 'Collaborative Ontology Construction for Information Integration'. Knowledge Systems Laboratory Department of Computer Science, KSL-95-63.
- Fonseca F., Davis C., Camara G. 2003. Bridging Ontologies and Conceptual schemas in Geographic Information systems. *Geoinformatica* 7(4) 355 – 378.
- Mirbel I. (1997) Semantic integration of conceptual schemas, *Data and Knowledge Engineering*, 21(2), 183-195.
- Palopoli L., Terracina G., Ursino D. 2003. 'DIKE: a system supporting the semi automatic construction of cooperative information systems from heterogeneous databases, *Software Practice & Experience*, 33(9) 847– 884.
- Ruggia R., Ambrosio A.P. 1997. 'A toolkit for Reuse in Conceptual Modelling – Proceedings Caise Conference.
- Shoval P., Danoch R. & Balabam M. (2004) Hierarchical entity-relationship diagrams: the model, method of creation and experimental evaluation, *Requirements Engineering* 9, 217-228.
- Slota R., Majewska M., Dziewierz M., Krawczyk K., Laclavik M., Balogh Z., Hluchy L., Kitowski J. 2003. Lambert S. *Ontology Assisted Access to Document Repositories for Public Sector Organizations* PPAM 2003, Czestochowa, Poland.

Taxonomic Databases Working Group on Biodiversity Informatics 2004, Proceedings of the Taxonomic Databases Working Group Annual Meeting, 11-17 October 2004 - University of Canterbury, Christchurch, New Zealand

Wang J., Gasser L. 2002. 'Mutual Online Ontology Alignment' AAMAS-2002 Workshop on Ontologies for Agent Systems.