Reuse of a repository of conceptual schemas in a large scale project

Carlo Batini[1], Riccardo Grosso[2]

[1] University of Milano Bicocca
Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy
{batini}@disco.unimib.it
[2] CSI-Piemonte,
Corso Unione Sovietica 216, Torino, Italy
{Riccardo.Grosso}@csi.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. This paper describes an experience of the use of an existing repository of conceptual schema, representing a wide amount of entities of interest for Central Public administration, in order to produce the corresponding repository of the administrations located in a region. Several heuristics are described and experiments are reported.

1 Structure of Italian Public Administration and previous experiences of conceptual schema Repositories

The goal of this paper is to describe an experience with a repository of conceptual schemas, related to Central Italian public administration, in order to build a first version of the corresponding repository of conceptual schemas of the local public administration located in one of the 21 regions of Italy. Due to limited available resources, several approximate techniques have been applied, that allow fast prototyping of the local repository, to be refined by 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, by gradually improving services provided by information systems and databases of its agencies. However, in the past the lack of co-operation among 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 administrations, 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 both central and local administration 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 have to be interchanged in an interoperable format, all the administration assign the same meaning to the same data, achieving database integration in the long term; this will enable the spread of information within government branches, a more easily accessible working environment, an increased quality of information management, and an improved state-wide decision making.

One of the first activities performed in the last decade, with the goal of designing a suitable data architecture, has been the project of building an inventory of existing information systems operating within the Central Public Administration in Italy. The activity was performed over about 500 data bases, whose logical schemas through reverse engineering activities were translated into Entity Relationship schemas.

In order to achieve cooperation among central and local administrations, it is now the moment 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 in a region 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 central agency exists, and is CSI Piemonte. But also in such a fortunate context, only logical relational schemas are available as input to the process of construction of the local repository. So, a methodology and tools are needed that allow approximate production of conceptual schemas to be arranged in a repository. This paper describes such a methodology and the experience so far in applying this to the context of the Piedmont Public Administrations.

The paper is organized as follows. In section 2 we discuss the structure of the Central Administration Repository and we recall the methodology for its construction. In section 3 we describe knowledge available for the design of the local PA repository. In section 4 we provide the methodology for building, starting from the central repository and local logical schemas, a first draft version of the local repository. Section 5 discusses experiences and future research work.

2. The Structure of the Central PA Repository

In this paper, a repository is defined as a set of conceptual schemas, each describing all the information managed by an organisation area within the information system considered. 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.

The primitives are: refinements, views; integration. Refinements allow the description of the same reality at different levels of abstraction. 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 descriptions of fragments of a schema. 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 it is possible to build a global description of data managed by an organisation area starting from local 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 schemes resulting from the integration of all the other schemes belonging to the same row (views of the integrated schema). In fig. 1 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.

In practice, when the repository is populated at the bottom level by hundreds of schemas, as in the case that we will examine, it is unfeasible to manage the three structuring primitives, and the view primitive is sacrificed. Furthermore, the integration/abstraction structuring mechanism is iterated, producing a sparsely populated repository such as the one symbolically represented in fig. 2, where, for instance, schema S123 results from the integration/abstraction of schemas S1, S2, and S3.

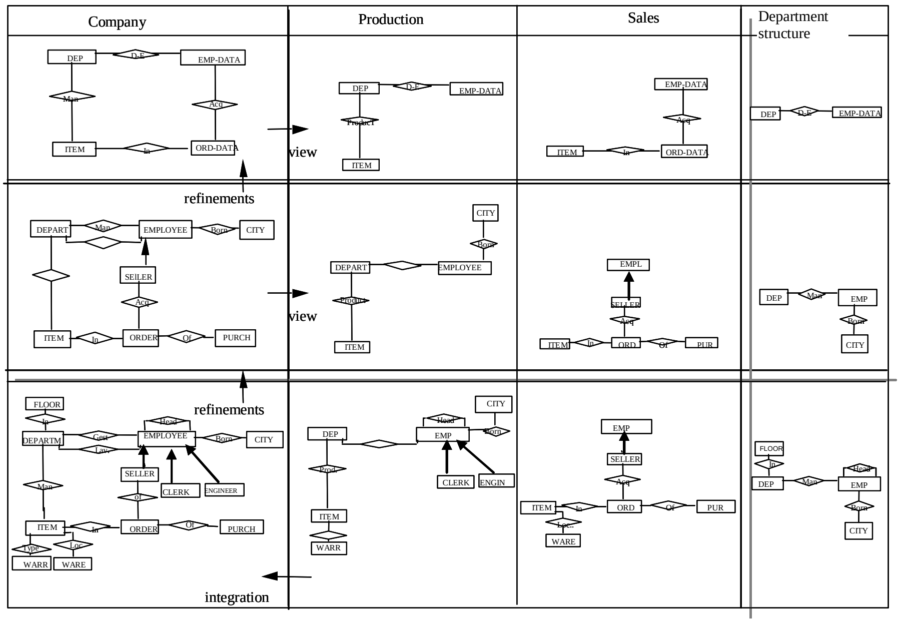

Fig. 1. An example of repository

SI12345678											
	SI123				SI456				SI78		
		S1	S2	S3		S4	S5	S6		S7	S8

Fig. 2. A fragment of repository

The repository structure described previously has been adopted for representing the conceptual content of a wide amount of conceptual schemas related to the most relevant databases of Italian central public administration in an integrated structure.

In order to build the whole repository, a methodology has been adopted, described in [1], [2]. About 200 person months were needed to produce the 500 basic conceptual schemas of the repository, while about 24 person months were needed to produce the 55 abstract schemas of the upper part (approximately 2 weeks per schema, both for basic and for abstract schemas). In figure 3 the schema at the top level of the repository is shown.

3 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 public administration (LPA) repository and the assumptions that have been made in the activity.

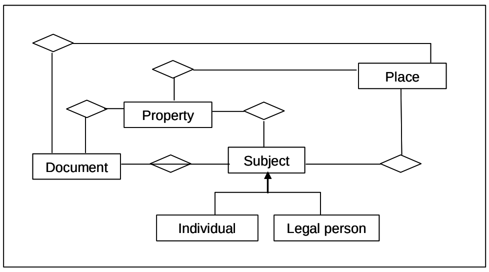

Fig. 3. The schema at the top level of the repository

A first relevant input available for the process is the Central Public Administration (CPA) Repository of schemas, made of basic and abstract schemas. A second input concerns Piedmont databases. Piedmont local public administration are centrally served by a unique consortium, CSI Piemonte, that created in the last years approximately 450 databases of 12 main local administrations, whose logical schemas are documented in terms of: relational database schemas, tables (approximately 17.000), textual descriptions of tables, referential integrity constraints defined among tables, attributes, definitions of attributes, identifiers. A very thin conceptual documentation has been created, that concern so called “supertypes of attributes” and “supertypes of relations”, corresponding to generalization abstractions of a few attributes and tables (about 10%) defined in the logical schemas. They have not been used so far in the process.

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 for local Piedmont administration the prevalent documentation concerns logical schemas.

A second relevant condition of our activity has concerned budget constraints; for the first year of the project we had only one person year available, so less than one tenth of the resources that were available for the construction of the central repository. So, in conceiving the methodology for the LPA repository production, we made a few significant assumptions, and 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 among 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 in some steps of the methodology a more manageable knowledge base than the 500 central 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 fig. 3, 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 fig. 4 a fragment of one of the hierarchies, the one referring to individuals.

- Individual
    - Employment
        - Unemployed
        - …
        - Retired
            - State pension retired
            - …..
            - Disability retired
    - Education
    - ……

Fig. 4. A fragment of the Individual generalization hierarchy

So, 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 fig. 5, have been:

1. The CPA Repository of 550 basic + abstract schemas
2. The five CPA Generalization hierarchies
3. The logical schemas of the 450 local PA databases.

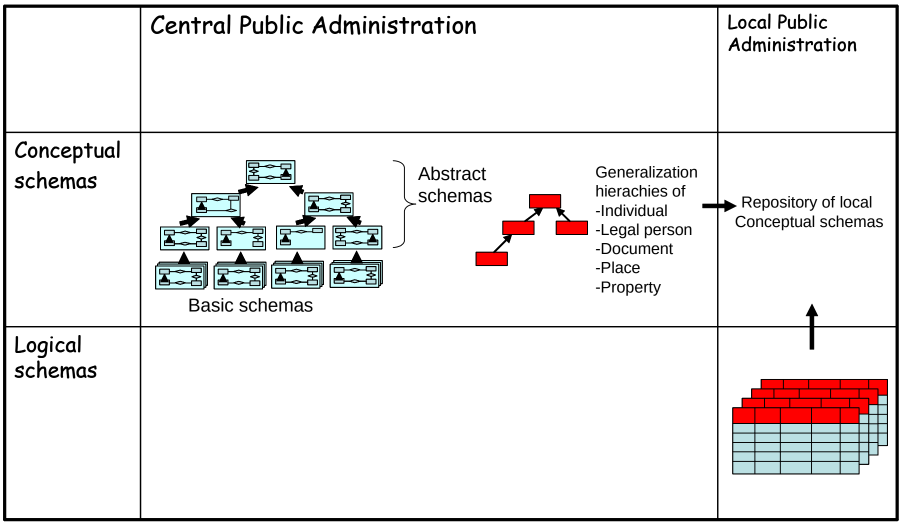
Central Public Administration
Local Public Administration
Conceptual schemas
Abstract schemas
Basic schemas
Generalization hierachies of
-Individual
-Legal person
-Document
-Place
-Property
Repository of local Conceptual schemas
Logical schemas

Fig. 5. Input knowledge for the production of the Repository of local conceptual schemas

4 The methodology for the construction of the local repository

In this paper, for reasons of space, we present only the methodology for building the basic schemas (its extension to abstract schemas is briefly discussed in Section 6). 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.

Step 1. Extract entities
Inputs: Central PA generalization hierarchies of concepts, one Local PA logical schema

Names of entities in hierarchies are compared with names and description of each table, and set of attributes of the logical schema. The comparison function makes use presently of a simple distance function among the different strings. The entities and corresponding frequency of matching are sorted, and a threshold is fixed: all the entities with frequency over the threshold are selected, resulting in a first draft schema made only of entities. The output is a draft schema made of disconnected entities.

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 pairwise compared with all the basic schemas in the CPA repository. For each pair of entities E21 and E2 several types of relationships are extracted by the basic schemas:

a. relationships defined exactly on E1 and E2;
b. relationships corresponding to chains of relationships defined among pairs E1-Ei; Ei-Ei+1; …; Ei+j-E2;
c. relationships defined among entities E1* and E2* corresponding to ancestors of E1 and E2 in the four generalization hierarchies.

Relationships collected in steps a and c are sorted according to the frequency of names. Here we have several 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 T1 and T2 in the logical schema, it is checked whether T1 and/or T2 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 in case 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 happen 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 schemas obtained after integrity constraints check and after domain expert check coincide. The output is the: final schema

We show in fig. 6 the schemas obtained as a result of the execution of steps 1 to 5 of the methodology in our case study. In this case, schemas obtained after integrity constraints check and after domain expert check coincide, and, consequently, are not distinguished in the figure.

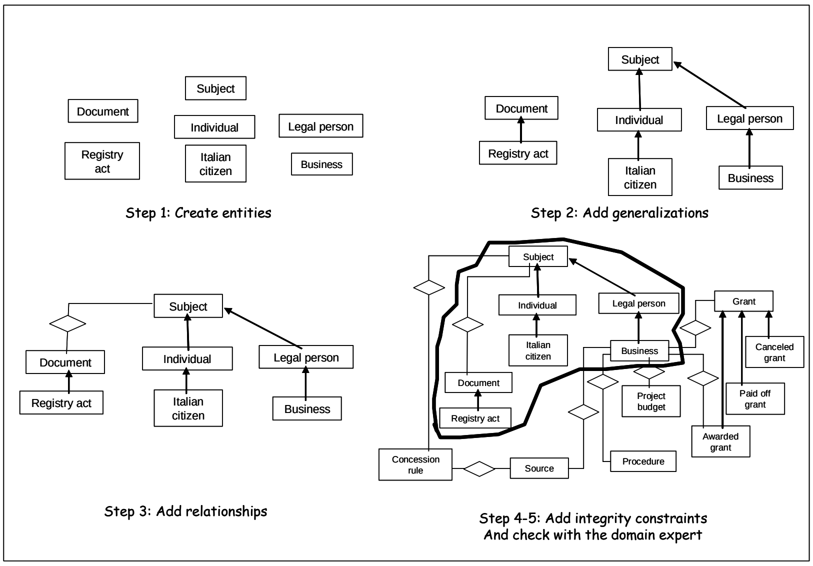

Fig. 6. Schemas obtained after steps 1-5

5. Experiments

In the present stage of the project we experimented with the above methodology in three different matters: businesses, health care, regional territory, and nine related areas. The total number of tables of the nine databases is approximately 350, corresponding to 2% of the total. We were interested in measuring two relevant qualities of the process:

1. the correctness of the conceptual schema with 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 catched 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 fig. 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 Central PA knowledge and Local PA knowledge reveals 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 tables are catched. 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 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 we are presently applying concerns their incremental update with abstract concepts, possibly generated in step 5. Such enriched hierarchies are progressively reconciled and brought near to hierarchies characteristic of local administrations, resulting in a corresponding more effective selection mechanism.

Table 1. Experiments results

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

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 time has been spent in steps 1-4, and 60% of time has been spent in the manual check. 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 1 person day the average final due effort, significantly lower than the typical 2-3 person/weeks needed for traditional design of one schema.

6. Concluding remarks

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

Repositories of conceptual schemas are proposed in several application areas (e.g. biosciences [9], reuse [3]). In [7] a solution and methodology are presented for reverse engineering of legacy databases using formal method-based techniques. Repositories of ontologies are proposed in several papers. The alignment and integration of ontologies is investigated in [4], [5], where information integration is enabled by having a precisely defined common terminology. A set of tools and services is proposed to support the process of achieving consensus on such a common shared ontologies by geographically distributed groups. Users can quickly assemble a new ontology from a library of modules.

Repositories of ontologies for public sector organizations are proposed in [6], [8]. The repository is used in [6] in a system supporting organizational activity by formalizing, sharing and preserving operational experience and knowledge for future use.

What seems new in our approach as regards the above mentioned papers is the abstraction/integration primitive adopted for structuring the repository and the attention devoted to feasibility aspects and resource constraints, and the consequent heuristic strategy. On the other side, we are conscious that our conceptual model is less powerful than ontology based models. A complete comparison with existing approaches is out of the scope of this paper.

We are now analyzing lessons learned and improving the methodology. First, we are extending the methodology to the production of abstract schemas in the repository. This step may effectively use the results of previous steps 1-5. In fact, 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 LPA logical schemas. We may conjecture that the initial schema is a candidate for abstract schema for the upper levels of the LPA repository, while the enriched schema, being a more detailed description representing a logical schema, populates the basic level of the repository. So, we may conceive two possible strategies for the repository update step.

In the first strategy, starting from the initial schema and the enriched schema we first complete the “local” repository of abstract schemas corresponding to the enriched schema; 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 with 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.

The first strategy is probably more effective when the actual LPA repository and the new schema represent very different knowledge, while the second strategy has the advantage of natively using the structuring paradigm of the repository, the abstraction/integration operation. We are currently experimenting with the two strategies, and other possible strategies, such as building small homogeneous repositories and then integrating them to obtain a larger repository.

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 similarity measures adopted.

References

1. C. Batini, G. Di Battista, G. Santucci - Structuring primitives for a dictionary of entity relationship data schemas -IEEE Transactions on Software Engineering vol.19 no.4 April 1993.
2. C. Batini, S. Castano, V. De Antonellis, M.G. Fugini, B. Pernici - Analysis of an Inventory of Information systems in the Public Administration - Requirements Engineering, 1996, 47-62
3. C. Batini, S: Castano, B. Pernici - Tutorial on Reuse Methodologies and Tools - Entity Relationships International Conference, Cottbus, Germany, 1996.
4. Jonathan DiLeo, Timothy Jacobs, and Scott DeLoach. Integrating Ontologies into Multi-agent Systems Engineering. Fourth International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS-2002). 15-16 July 2002, Bologna (Italy).
5. Adam Farquhar, Richard Fikes, Wanda Pratt, James Rice - Collaborative Ontology Construction for Information Integration -Knowledge Systems Laboratory Department of Computer Science, KSL-95-63, August 1995.
6. F. Fonseca, C. Davis, G. Camara – Bridging Ontologies and Conceptual schemas in Geographic Information systems – Geoinformatica 7:4, pp. 355 – 378, 2003.
7. J. Perez, I. Ramos, J: Cubel, F. Dominguez, A. Boronat, J. Carsì Data reverse engineering of Legacy Databases to object oriented conceptual schemas - Electronic Notes in Theoretical Computer Sceince 74 No. 4, 2002.
8. R. Slota, M. Majewska, M. Dziewierz, K Krawczyk, M. Laclavik, Z. Balogh, L. Hluchy, J. Kitowski, S. Lambert Ontology Assisted Access to Document Repositories for Public Sector Organizations, PPAM 2003, Czestochowa, Poland
9. Taxonomic Databases Working Group on Biodiversity Informatics 2004, Taxonomic Databases Working Group Annual Meeting, 11-17 October 2004 - University of Canterbury, Christchurch, New Zealand.