On Intelligent Access to Heterogeneous Information

Fèlix Saltor
Universitat Politècnica de Catalunya
E-08034 Barcelona (Catalonia)
saltor@lsi.upc.es

Elena Rodríguez
Universitat Politècnica de Catalunya
E-08034 Barcelona (Catalonia)
malena@lsi.upc.es

Abstract

The process to provide integrated access to several, independent information sources is not easy, due to semantic heterogeneities which lead to semantic conflicts or contradictions. In this paper we present a discussion about the existence of contradictions, their importance and how they can be handled when the need of cooperation arises.

1 Introduction

The growing need to access information from several information sources that have been designed independently and operate in a autonomous way has become an active research area nowadays. A possible solution to satisfy this need of cooperation is provide *integrated access*. This means that a user can formulate a single query and receives a single, consolidated answer. The collection of information sources (from now on called Component DataBases, CDBs), jointly with the software layer that manages the integrated access to data stored in CDBs, is known as an *interoperable* or *Federated DataBase System* (FDBS) [SL90].

The main characteristics of such systems are the autonomy and heterogeneity of CDBs. *Autonomy* means that CDBs are under separated and independent control. This implies that preexisting applications and users can continue working, although CDBs participate in a FBDS. Different types of autonomy are distinguished: *design* (each CDB chooses its design with respect to different matters like data being managed, data representation, semantic interpretation of data, operations supported, etc.), *communication* (a CDB decides when and how responds to federated requests), *execution* (each CDB decides the order of federated requests under its control with respect to its local operations) and *association* (a CDB is able to decide whether participate or not in one or more federations, as well the possibility of its disassociation of a federation).It is important to note that some degree of autonomy must be traded off for the sake of integrated access.

On the other hand, *heterogeneity* appears as consequence of the design autonomy of CDBs and we can distinguish between systems heterogeneities and semantic heterogeneities. *Systems heterogeneities* include differences in hardware platforms, in DataBase Management Systems (DBMSs) like different data models and data languages, in security and transaction management and so on. *Semantic heterogeneities* include differences in the way the real world is conceived and modeled, above all when CDBs represent overlapping or related parts of the Universe of Discourse (UoD). Therefore semantic heterogeneities lead to *semantic conflicts* or *contradictions* that must be dealt during the process of schema integration of CDBs.

The aim of this paper is to show that contradictions are meaningful and how they can be handled in FDBS. With this objective in mind, we have organized this paper as follows: in section 2 and 3 we discuss the different semiotic levels and the concept of semantic relativism in order to better understand semantic heterogeneities. In section 4 and 5 we partially present our classification of semantic heterogeneities and how our methodology handles them. Finally in section 6 we draw our conclusions.

2 Semiotic Levels

In this paper we focus on the level of *semantics*, and on contradictions at this level. What do we mean by

Proceedings of the 4th KRDB Workshop
Athens, Greece, 30-August-1997

(F. Baader, M.A. Jeusfeld, W. Nutt, eds.)

http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-8/

"semantics"?

Traditionally, the definitions of "semantics" as a science, as part of *semiotics* or the theory of signs – together with *syntax* and *pragmatics*–, given by C.W. Morris or C. Cherry are (in our context, "data elements" should be substituted for "signs"):

- "The study of the relations of signs to the objects to which the signs are applicable" [Mor38],
- "deals with the significance of signs in all modes of signifying" [Mor46], and
- "signs and their relations to "outside world" (designation) – a rather questionable notion" [Che66].

More recently, Ron Stamper developed not just three, but six semiotic levels of interest in information systems, which were taken up by the FRISCO report [FVS+97]: *physical world*, *empirics*, *syntactics*, *semantics*, *pragmatics* and *social world*. They may be represented as "*the semiotic ladder*", as in figure 1.

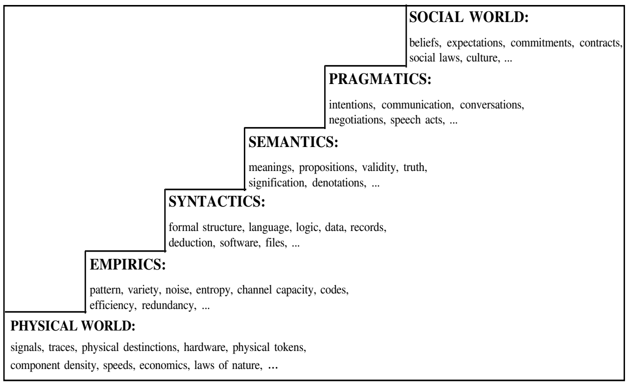

Figure 1: The Semiotic Ladder

Contradictions, or more generally, *heterogeneities*, may appear at all levels. Those at semiotic levels below semantics are out of the scope of this paper. The two upper levels are taken into account as long as they affect the level of semantics. To better understand semantic heterogeneities, let us first discuss the main reason for their existence.

Seen from a semantic perspective, the process of designing any information source, such as a database, proceeds from the external world to the data.

The outside world at large is perceived by the designer; his perceptions are abstracted in *concepts* and thoughts, and new concepts are created in his mind, so that he develops his own *conceptualization* of the outside world (this is a very simplified description of the process; in fact, a number of people will try to explain their own conceptualizations to the designer). Following the "Cartesian dualism" (see [Che66]) , there are two separate worlds (or realms): the external or *outside world*, also known as the "real world", and the internal, "mental", or *conceptual world*. In the outside world there are "objects" (in the etymological sense of the word: "ob-jacta" = "thrown outside", or "out there", as opposed to "sub-ject"), that we will call *real world objects* (we reserve *"objects"* for database objects). In the conceptual world there are concepts (this is one of several schools of thought).

The relationship from a concept to the outside world is *not* by "meaning", but by *denotation* or designation: a concept denotes some real world objects (including none in cases such as the concept of unicorn). The *meaning* of a concept lies in the set of its relationships to the rest of the concepts in the designer's conceptualization. We may say that all these concepts with their relationships form his conceptualization, also called his "personal semantics" or his "real world semantics"; but note that semantics is not part of the outside world alone, it needs someone conceiving it.

This person is placed in a given context, where *context* is the part of the outside world being modeled, and from which point of view it is seen.

The designer turns his conceptualization into a database design. The database, and particularly its schema, *represents* his conceptualization. The other way around, starting at the database, we arrive at the conceptual world by *interpretation*. We have met a third world, or realm: the world of representations, the computational or data world, that we will call the *database world* (also known as "model world"). Anything in this third world will be considered a database object, *object* for short.

Therefore, the relationship between the outside world and the database world *is not direct*; it is composed of two direct relationships: between the outside world and the conceptual world (conception/denotation) and between this one and the database world (representation/interpretation). When we speak of *semantics of an object*, we mean just the second relationship applied to it, that is, the concept represented by the object. From the concept, we can cross the first relationship to reach its denotation in the outside world.

From the concept represented by an object we can also look at the set of its relationships to other concepts, that is, its meaning. Since each such relationship will be represented in the database world by some *construct* (for example by a method), we will say that the meaning of the concept is represented by a set of constructs, the relationships of the object to other objects. Starting from the object, we may as well look first at this set in the database world, and then cross to the conceptual world to find out the meaning.

3 Semantic Relativism and Semantic Heterogeneities

A person p1 has his own conceptualization of the outside world; another person p2 will have another conceptualization of the outside world, even if the context is the same. There is no such thing as an absolute semantics, there is not just one "real world semantics": semantics is relative (for a discussion of semantic relativism, see [SGS93]).

If p1 designs an information source, for example database CDB1, he will reflect this conceptualization of his in the database design. Analogously, a database design made by p2 will result in a different database CDB2. The two databases will have semantic heterogeneities. For example, married couples may be represented in CDB1 by objects of the class COUPLES, with attributes HUSBAND and WIFE (among others), while in CDB2 there is a class PERSONS with a SPOUSE attribute.

This was an example of semantic heterogeneity. In [DKM+93] the following candidate definition of semantic heterogeneity is given: "variations among component database systems in the structure, organization, and conceptual description of information facts (units), units of behavior (procedures), and semantic integrity constraints". From an O-O approach, semantic heterogeneities include differences between classes, between the structures formed by the classes (generalization and aggregation), and between object instances. Section 4 deals with the classification of semantic heterogeneities.

The main reason for semantic heterogeneities is the difference in conceptualizations, as already explained, but other causes are possible. Systems heterogeneities, and particularly DBMS heterogeneities, may force the adoption of different models (relational versus O-O, for example); because of that, a given conceptual relationship may have to be represented by different structures (referential integrity in relational versus aggregations in O-O), or simply not represented if the model does not support it (specialization structures, methods, in O-O, unsupported in relational).

Another reason for semantic heterogeneities may be the difference in the frequency of updates. If CDB1 is updated in real time, while CDB2 is only updated each night, data values in CDB1 and CDB2 that should be equal will progressively differ, until the databases are re-synchronized the following night.

Detecting that a data element in source 1 and a data element in source 2 represent the same conceptualization of the outside world (in a *3-world philosophy*) is a difficult problem. It involves finding an *object identification function* or *oif* to correlate extensions (see [GSSC95b], [CKSGS94]). This is a prerequisite to detecting contradictions.

Semantic relativism, again, may give rise to different oifs. Let us use a book comparison example: when two books are the same book? For person p1, the answer is: if they have the same ISBN; his oif is "BOOK1.ISBN = BOOK2.ISBN". Person p2 takes printing into account, and her oif is "(BOOK1.ISBN = BOOK2.ISBN) AND (BOOK1.PRINTING = BOOK2.PRINTING)". Librarian p3 differentiates between two copies of the same printing, which have different lending histories, and uses library internal identifiers; her oif is "BOOK1.LID = BOOK2.LID".

4 Classification of Contradictions

The goal of a FDBS is provide integrated access to a collection of users (called federated users). For this purpose, it is necessary to identify the objects in the CDBs that represent concepts that are semantically related. Once these relationships have been detected global views could be offered to federated users. This interface with federated users takes the form of *federated schemas*. Depending on whether different federated users share or not the same conceptualization and from identical point of view, one or more federated schemas will be derived.

The process of obtaining federated schemas consists of overcoming semantic heterogeneities which appear to the FDBS as semantic conflicts or contradictions. It is a difficult process due to the lack of sources of semantic knowledge. It is also important to note that a contradiction can exist only between objects that represent similar concepts. In order that contradictions can be dealt they must be first detected. We have organized semantic heterogeneities in a classification ([GSCS96]) which subsumes previous classifications by Kim ([KCGS93], [KS91]) and by Sheth ([SK93]).

Our classification is based on a rich object-oriented data model that has been chosen as the *Canonical Data Model* (CDM) of the FDBS.This CDM in our case is BLOOM ([CSGS94]). For a discussion about the suitability of object-oriented models as CDMs see [SCGcS91]. We classify semantic heterogeneities into three groups:

- Heterogeneities between object classes
- Heterogeneities between class structures
- Heterogeneities between object instances

4.1 Heterogeneities Between Objects Classes

This group consists of differences between classes of objects in different CDBs that are *corresponding classes* (i.e. they represent the same concept in their respective context). One type of heterogeneity into this group are *differences in extensions* which occur when corresponding classes in different CBDs can denote different sets of real world objects. Therefore classes may or may not have some extensional relationship: extensions can be disjointness or nondisjointness. In the last case there can be a relationship of inclusion or overlapping.

Our characterization also includes other heterogeneities in this group, for example, *differences in names* both in attributes, methods and classes (synonyms and homonyms), *differences in attributes and methods* (e.g. presence or not presence of them), *differences in domains* (both semantic and syntactic domains) and *differences in constraints* (e.g. for corresponding attributes in corresponding classes, differences in constraints on attributes like multivaluation limits, uniqueness and nonnull allowance).

4.2 Heterogeneities Between Class Structures

This group classifies heterogeneities between the structures formed by corresponding classes in different CDBs. Given that BLOOM distinguishes dimensions of generalization/specialization, aggregation/decomposition and classification/instantiation, inconsistencies are classified along these three dimensions.

Examples of inconsistencies along the generalization/specialization dimension could be *differences in the specialization criteria*. We assume that the generalization semilattice (multiple inheritance is allowed) is done according to some criteria. For example, the designer of CDB1 can have specialized class PERSONS into two disjoint subclasses MALE and FEMALE according to SEX criteria and into four disjoint subclasses CHILDREN, TEENAGERS, ADULTS and ELDERS according to AGE criteria, whereas in CDB2 its corresponding class PEOPLE has just one specialization by SEX criteria with disjoint subclasses MEN and WOMEN. Furthermore, if we have two corresponding classes and corresponding specialization criteria, the number of subclasses in each CDB can be different (*inconsistencies in specialization degrees and characterization*). Following the example above, imagine CDB3 where class PERSONS also exists and is also specialized according AGE criteria, but in this case into three subclasses MINORS, ADULTS and SENIORS.

Inconsistencies along the aggregation/decomposition dimension include differences related to the fact that an object of a class in a CDB can be represented as a simple aggregation of its attributes (they can refer to objects of predefined or non predefined classes) or can be created by aggregation of objects of other classes through their respective attributes, so an object cannot exist without them. There could be *inconsistencies in the kind of the aggregation*, for example, designers of different CDBs can have modeled the assignment of employees to projects in different ways:

- The designer of CDB1 has modeled the assignment of employees to projects in a class ASSIGNMENTS which is created by aggregation of one object belonging to class EMPLOYEES and one object belonging to class PROJECTS. Class ASSIGNMENTS has also properties, by simple aggregation, in particular the attributes BEGINNING_DATE, ENDING_DATE and OFFICE.
- The designer of CDB2 has represented the assignment of employees to projects through a class ASSIGNMENTS which is the aggregation of objects belonging to classes EMPLOYEES, PROJECTS and OFFICES respectively. In addition, the properties of ASSIGNMENTS are represented by the attributes ENDING_DATE and RESPONSIBILITY.

Finally, inconsistencies along classification/instantiation dimension, also called *schematic discrepancies*) ([KLK91]),appear when some values or data in one CDB are seen as part of the schema (metadata) in another CDB. In order to illustrate them, imagine we have three CDBs, storing the parents of a group of persons:

CDB1: **class** PARENTHOODS
    simple aggregation of
      CHILD: PERSONS,
      PARENT: PERSONS,
      PARENT-TYPE:(FATHER, MOTHER).

CDB2: **class** PARENTHOODS
    simple aggregation of
      CHILD: PERSONS,
      FATHER: PERSONS,
      MOTHER: PERSONS.

CDB3: **class** FATHERS
    simple aggregation of
      CHILD: PERSONS,
      FATHER: PERSONS,
  class MOTHERS
    simple aggregation of
      CHILD: PERSONS,
      MOTHER: PERSONS,

Note that FATHER is a value in CDB1, the name of an attribute in CDB2, and (with a final S) the name of a class in CDB3, and that class names and attribute names are part of the schema (metadata).

4.3 Heterogeneities Between Object Instances

This group of semantic heterogeneities basically includes differences in values for corresponding classes and corresponding attributes of the classes. Just an example: if the Michelin guide assigns 1 star (maximum is 3) to restaurant GAIG of Barcelona and the Campsa guide assigns 2 suns to it (maximum is 3), we cannot say that one is true and the other is wrong; every guide is true according to its way of classifying restaurants. The example above is what is called *value discrepancy* in our classification.

Other contradictions in this group are the presence or not presence of null values of attributes (*null/nonnull discrepancies*) and *discrepancies in the number of values* for multivalued attributes.

5 Handling Contradictions

In any situation where access to heterogeneous informations sources is required, we know that contradictions, or more generally what we called semantic heterogeneities, will appear, and we need an intelligent approach to handle them. Everybody: final user, administrator, technical expert, knowledge engineer, must be prepared to live with this fact, not just as an exception, but as a way of life.

The main reason for these heterogeneities is semantic relativism, as explained in section 3 above. This means that we are not to consider one of the sources as correct and true, and the others as wrong, but to admit that each one is true according to the conceptualization, or "universe of discourse" or "ontology" of the designer (and users) of that source. If "solving" contradictions means finding out which one is true and nothing more, then this "solution" ignores the real issue.

The fact of heterogeneity is in itself *meaningful*, and should be taken into account by any approach. If we were able to understand *why* the conceptualizations are different, then we would have a deeper knowledge about the contradicting data. In the case of the restaurant example, if we could know the criteria the different guides are using, then we would be better placed to provide each user with the solution more suitable to him.

This is the reason why semantic enrichment is important. We call *semantic enrichment* the process of making explicit any semantic information implicit in the information source (database, file, application, etc.) which was not explicit in the schema or metadata of that source, in case such an schema did exist ([CSGcS94], [Cas93]). This process may involve extracting dependencies from the data, as in [CS93]. Semantic enrichment could be considered similar or a particular case of data mining/knowledge discovery [PSF91].

5.1 Transparency of the sources

One important consideration is the support for each user of the degree of transparency he wants.

1. Some users may need full source transparency, so that the query and the answer do not reflect the fact that different sources are accessed.

2. Some users want to pose the query independently from sources, but need that each data element in the answer is tagged with its source ("source tagging").

3. Some users want to query some sources and not others, and have answers tagged with their sources.

4. Transparency/no transparency may be influenced by the context of each user, in the sense given to "context" in section 2.

5. Business practices may determine which degree of transparency is needed, as explained in [WM90].

A good methodology should support any desired degree of transparency, as we do by means of discriminants in the generalization.

5.2 Discriminated generalization

Many methodologies use the operation of conventional "generalization" to create a superclass which generalizes classes from different informations sources. For example, class EMPLOYEES from CDB1,and class EMPLEATS from CDB2 are generalized into a more general class EMPREGADOS in the federated schema. But then each employee, when seen through this federated superclass, has lost his source, and source tagging cannot be supported.

The operation of *discriminated generalization* used in our methodology (see [GSS91], [GSSC95a], [GSCS95]), together with our CDM BLOOM ([CSGS94]) –designed according to the framework developed in [SCGcS91] – and an extended architecture ([SCRR96], [ROS97]), make it possible to preserve all information, including contradictions and identification of the source (for non transparency purposes).

This way, each user is able to specify how he likes contradictions to be handled according to his point of view, if and when they appear. In the case of the example in subsection 4.3, for a given restaurant, if there is some contradiction:

- User A could prefer to receive just the Michelin rating.

- User B could prefer to receive both (without transparency).
- User C could prefer to receive the highest (with transparency of which of the guides assigns this rating);
- User D could prefer a weighted average; etc.

5.3 Other techniques

In addition to discriminated generalization, other techniques to help solving semantic heterogeneities exist, and they have been implemented in tools and systems. In particular, we have developed tools for detecting and solving classes of semantic heterogeneities such as those presented in section 4, including schematic discrepancies. See [GSCS96] for a complete description.

6 Conclusions

People from database field have a lot to say about Intelligent Access to Heterogeneous Information. When these database people meet with knowledge representation experts, meaningful contradictions appear, incompleteness are detected, both sides are semantically enriched and learn models and languages of the other side. The result should be better solutions and better quality as exemplified by the Intelligent Integration of Information [We96].

Acknowledgments

This work has been partially supported by the Spanish PRONTIC program, under project TIC96-0903. We are grateful to the anonymous referees for their comments.

References

[Cas93] M. Castellanos. A Methodology for Semantically Enriching Interoperable Databases. In Worboys and Grundy, editors, *Advances in Databases (Proceedings of the 11th Bristish National Conference on Databases (BNCOD'11)*, pages 58–75. Springer Verlag, LNCS 696, 1993.

[Che66] C. Cherry. *On Human Communication. (2nd ed.)*. The M.I.T. Press, Cambridge and London, 1966.

[CKSGS94] M. Castellanos, Th. Kudrass, F. Saltor, and M. García-Solaco. Interdatabase Existence Dependencies: A Metaclass Approach. In *Proceedings of the Third Int. Conference on Parallel and Distributed Information Systems (PIDS'94)*, pages 213–216. IEEE-CS Press, 1994.

[CS93] M. Castellanos and F. Saltor. Extraction of Data Dependencies. In Jaakkola, Kangassalo, Kitahashi, and Markus, editors, *Information Modelling and Knowledge Bases V (3rd European-Japanese Seminar on Information Modelling and Knowledge Bases)*, pages 400–420. IOS Press, Amsterdam, 1994, 1993.

[CSGcS94] M. Castellanos, F. Saltor, and M. Garcia Solaco. Semantically Enriching Relational Databases into an Object Oriented Semantic Model. In D. Karagiannis, editor, *Proceedings of the 5th International Conference on Database and Expert Systems Applications (DEXA'94)*, pages 125–134. Springer Verlag, LNCS 856, 1994.

[CSGS94] M. Castellanos, F. Saltor, and M. García-Solaco. A Canonical Model for the Interoperability among Object-Oriented and Relational Databases. In Ozsu, Dayal, and Valduriez, editors, *Distributed Object Management*, pages 309–314. Morgan Kaufmann, 1994.

[DKM+93] P. Drew, R. King, D. McLeod, M. Rusinkie wicz, and A. Silberschatz. Report of the Workshop on Semantic Heterogeneity and Interoperation in Multidatabase Systems. *SIGMOD RECORD*, 22(3), Sep 1993.

[FVS+97] E. Falkenberg, A. Verrijn-Stuart, et al. A framework of information systems concepts. Technical report, IFIP WG 8.1, Leiden University, 1997.

[GSCS95] M. García-Solaco, M. Castellanos, and F. Saltor. A Semantic-Discriminated Approach to Integration of Federated Databases. In S. Laufmann and S. Spaccapietra, editors, *Proceedings 3rd International Conference on Cooperative Information Systems (CoopIS'95)*, pages 19–31. Univ. Toronto, 1995.

[GSCS96] M.A. García-Solaco, M. Castellanos, and F. Saltor. Semantic Heterogeneity in Multidatabase Systems. In O. Bukhres and A. Elmagarmid, editors, *Object Oriented Multidatabases*. Prentice-Hall, 1996. Invited chapter.

[GSS91] M. García-Solaco and F. Saltor. Discriminated Operations for Interoperable

Databases. In *Proceedings 1st Int. Workshop on Interoperability in Multidatabase Systems (IMS-91)*, pages 222–225. IEEE-CS Press, 1991.

[GSSC95a] M. García-Solaco, F. Saltor, and M. Castellanos. A Structure Based Schema Integration Methodology. In *Proc. 11th Int. Conference on Data Engineering (ICDE'95), Taipei*, 1995.

[GSSC95b] M. García-Solaco, F. Saltor, and M. Castellanos. Extensional Issues in Schema Integration. In *Proc. 6th Int. HK DB Workshop: Database Reengineering and Interoperability, Hong Kong*, 1995.

[KCGS93] W. Kim, Y. Choi, S. Gala, and M. Scheevel. On Resolving Schematic Heterogeneity in Multidatabase Systems. *Distributed and Parallel Databases*, 1(3):251–279, 1993.

[KLK91] R. Krishnamurthy, W. Litwin, and W. Kent. Interoperability of Heterogeneous Databases with Schematic Discrepancies. In *Proceedings 1st International Workshop on Interoperability in Multidatabase Systems (RIDE-IMS 91)*, pages 144–151. IEEE-CS Press, 1991.

[KS91] W. Kim and J. Seo. Classifying Schematic and Data Heterogeneity in Multidatabase Systems. *IEEE Computer*, 24(12), Dec 1991.

[Mor38] C.W. Morris. *Foundations of the Theory of Signs.* Int. Encyclopedia of Unified Science Series, Vol.I, 32, Univ. of Chicago Press, Chicago, 1938.

[Mor46] C.W. Morris. *Signs, Language and Behavior.* Prentice-Hall, New York, 1946.

[PSF91] G. Piattetsky-Shapiro and W.J. Frawley. *Knowledge Discovery in Databases.* AAAI Press/The MIT Press, 1991.

[ROS97] E. Rodríguez, M. Oliva, and B. Saltor, F.and Campderrich. On Schema and Functional Architectures for Multilevel Secure and Multiuser Model Federated Database Systems. In S. Conrad et al., editor, *Proceedings of the International CAiSE'97 Workshop on Engineering Federated Database Systems (EFDBS'97)*, pages 93–104. Otto-von-Guericke-Universitat Magdeburg, Fakultat fur Informatik, 1997.

[SCGcS91] F. Saltor, M. Castellanos, and M. García Solaco. Suitability of Data Models as Canonical Models for Federated DBs. *ACM SIGMOD Record*, 20(4):44–48, 1991.

[SCRR96] F. Saltor, B. Campderrich, E. Rodríguez, and L.C. Rodríguez. On Schema Levels for Federated DB Systems. In Yetongnon and Hariri, editors, *Proceedings of the International Conference on Parallel and Distributed Computing Systems (PDCS'96)*, pages 766–771. ISCA, 1996.

[SGS93] F. Saltor and M. García-Solaco. Diversity with cooperation in database schemata: Semantic relativism. In *Proc. ICIS conference (Orlando)*, 1993.

[SK93] A. Sheth and V. Kashyap. So Far (Semantically) yet so Near (Semantically). In D. Hsiao et al., editor, *Interoperable DB Systems (DS-5). Proceedings IFIP WG 2.6 Conference on Semantics of Interoperable Database Systems (Data Semantic 5).* North Holland, 1993.

[SL90] A.P. Sheth and J. Larson. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3):183–236, Sep 1990.

[We96] G. Wiederhold (ed). *Intelligent Integration of Information.* Kluwer Academic Publishers, 1996. A Special Double Issue of the Journal of Intelligent Information Systems 6(2-3).

[WM90] Y. Wang and S. Madnick. A Polygen Model for Heterogeneous DBs: The Source Tagging Perpective. In *Proceedings of the 16th Int. Conference on Very Large Data Bases (VLDB'90)*, pages 519–538, 1990.