ACM SIGMOD RECORD, Vol. 21, No. 1, pp. 123-132 (1992)

# An Annotated Bibliography on Object–Orientation and Deduction

Stefan Conrad* & Martin Gogolla

Technical University of Braunschweig, Informatik, Abt. Datenbanken
Postfach 3329, W–3300 Braunschweig, Germany
e–mail: {conrad|gogolla}@infbs.uucp

## Abstract

This note tries to briefly survey research activities and results on the integration of object–oriented concepts and deductive database languages.

## 1 Introduction

There are several efforts creating new deductive languages and systems extending conventional deductive database languages by involving concepts of higher order logics, especially the treatment of sets. In contrast to these value–based languages, object logics try to integrate the most important object–oriented concepts. Thereby, object identities and inheritance are available in these languages. Nevertheless, the ways to integrate object identities, classes and inheritance are quite different.

Besides these efforts in designing object–oriented deductive languages, the theoretical foundations of *object–oriented*, of *deductive* and of *object–oriented and deductive* systems are investigated. Whereas several well known and overall accepted semantics for logic programming exist, there is no clear consensus about what an object–oriented system is. Therefore, in this area a lot of work still has to be done.

This survey can only be a snapshot of a very lively research area. Up to now a lot of excellent results have been presented. However, there still are some important steps to go on the way towards a full integration of object–orientation into deductive database languages.

## 2 Integration of Object–Orientation and Deduction

In the sequel we will try to give an overview of important steps and results on the integration of object–oriented concepts into deductive languages and systems.

### 2.1 Extending First–Order Predicate Calculus

The first steps towards object–oriented deductive languages were the design and development of value–based languages. These languages extend the well known first–order predicate calculus or the corresponding deductive database languages like **DATALOG**

*Work reported here has been supported by BMFT under Grant No. ITS 9002 D

[CGT89, CGT90, etc.] with higher–order concepts.

The most popular deductive languages which support sets and set manipulation are **LDL** [TZ86, BNR∗87, NT89, BNST91], **LPS** [Kup87, Kup88, Kup90] and **COL** [AG88, AH88, AG91]. In these languages different ways of integrating sets are proposed. In LDL we have a grouping operator to create sets. LPS uses restricted universal quantifier to manipulate sets. COL offers data–functions to specify and manipulate sets.

Another way to extend first–order predicate calculus with a higher–order concept is realized in **HiLog** [CKW89]. Instead of sets and set manipulation, HiLog offers the possibility to manipulate predicates by using the predicate symbols as constants and by parametrizing predicate symbols.

## 2.2 Object–Oriented Logics

The so-called *object logics* are **Maier's O-Logic** [Mai86], **C-Logic** [CW89], **O-Logic (revisited)** [KW89] and **F-Logic** [KL89, KLW90]. These logics include object identities, type/class hierarchies and notions of inheritance. Whereas in Maier's O-Logic an object's identity is created through a variable in a rule head which does not appear in the body of the rule, object's identities are generated in the other logics by explicitly giving an object name or using an object constructor.

Maier's O-Logic, C-Logic and the revisited O-Logic have a class hierarchy while F-Logic has an object hierarchy and no classes. Within the hierarchy, the properties of the objects of a class are inherited to the objects of all subclasses. In F-Logic the properties of an object are inherited to all specialized objects.

Concerning properties, Maier's O-Logic offers only functional labels (attribute fields) and C-Logic offers only set–valued labels. In the revisited O-Logic and F-Logic both are available. In addition, labels in F-Logic can be parametrized. A detailed investigation of these object logics can be found in [Kes90].

In addition to these logic–oriented approaches, in the last years a lot of proposals for deductive *languages* involving object–orientation have come up.

In **IQL** [AK89] object identities are created through unbound variables in rule heads. The language is strongly typed, has explicit object constructors for complex objects and has an inflationary semantics. In the data model on which IQL is based, there is, besides classes, the concept of relations in which objects can participate. Each class has its own object identity.

**LOCO** [LVVS90, STV91, STVV91] is based on ordered logic [LSV90, LV90a, LV90b, LV91], a three–valued logic. Rules are assigned to objects and are inherited within the object hierarchy. Object names and object constructors determine object identities.

**LIVING IN A LATTICE** [HS90, HS91] is an object–oriented rule–based language. The basic data model EXTREM [Heu87, HS89] distinguishes between specialization and generalisation. By restricting the rules which determine the objects of a derived class to be all object–generating or all object–preserving, derived classes can automatically be inserted in the type/class hierarchy.

The language **LOGRES** [CCC∗90, CCT90, CCT91] offers multiple constructors for complex objects. Besides a tuple and a set constructor, there are also constructors for sequences (lists) and bags. The language ist strongly typed and resembles IQL. Moreover, for LOGRES, an automatic generation of referential integrity rules is defined. A clear update concept distinguishing between the intensional and extensional database part is provided by modules for database evolution.

An extension of LDL by object identities and inheritance is **OIL** [Zan89]. The semantics of an OIL program is defined by an equivalent

LDL program which can be received by transformation. The objects' identities are determined by tuple values. Therefore it is not possible to have to different objects with the same tuple value.

## 2.3 Further Approaches and Projects

There are a lot of further languages into which object–oriented concepts are tried to be integrated. Some of them should be mentioned here.

**LOGIN** [AN86] was one of the first declarative languages supporting inheritance. **TEDM** [MZO89, ZM89] is a data model and an object–oriented rule–based language. It is based on the ideas of Maier's O-Logic and can therefore be considered as an implementation of this logic.

The language **IDLOG** [She90] is a theoretical approach to express aggregation functions in a deductive language by adding tuple identifications to a pure first–order deductive language.

**DLP** (Distributed Logic Programming) [El91] is an extension of PROLOG with active objects. In DLP, parallelism and communication between concurrently active objects is possible. Within the objects, non–logical variables are used.

In a number of projects, object–oriented and deductive languages are implemented (e.g., LDL, F–Logic, LOGRES, TEDM). Within the **KIWIS**–project [ADJ∗91] an object–oriented, deductive and distributed knowledge base management system is developed and implemented. Its core language is LOCO (see 2.2).

## 2.4 Theoretical Foundations

The scope of theoretical basic research concerning the integration of object–orientation and deduction is enormous. First of all there are many activities on both areas seperatly, but also ways for integration are considered. Besides, other research areas have an influence which have not to be ignored; among others, these are the relational model (based on [Cod70]), semantic data models (e.g., TAXIS [MW80], SDM [HM81], IRIS [LK86], IFO [AH87], EER [GH91]) and data models for complex objects (e.g., [SS86], [PA86], [Heu87]).

### 2.4.1 Object–Oriented Aspects

Because no clear consensus exists about what an object–oriented system is, plenty of suggestions for *shopping lists* for object–oriented concepts (e.g., [Ban88, ABD∗89, Dit90, Kim90]) have arised.

In more detailed papers, single object–oriented concepts are investigated, e.g., object identity [KC86, MW90]. Also some aspects going beyond it, are considered for object–oriented systems, like meta–functions and contexts in [CS88].

In addition, there are more general frameworks [Bee91] and comparisons of object–oriented and relational database systems [Kim91]. Based on concrete systems (e.g., $O_2$ [LR89, O∗90]) some interesting topics like algebra based query languages for object–oriented database systems (for instance RELOOP [CDLR90]) and how to classify query results in a class hierarchy [HS91] are dealt with.

### 2.4.2 Deductive Aspects

In the area of logic programming well–known and overall accepted foundations exists [Llo84]. The combination of logic programming and (relational) databases is considered in several publications [GMN84, CGT90, GN90, Ram90]. DATALOG [CGT89] is the well–known prototype of a deductive database query language.

[Ull85] is a survey of different ways to implement logical query languages. The discussion of semantics and evaluation of rules over complex objects is manifold (see for example [HS89]). An approach to classify recursive formulas in deductive databases for query optimization is pointed out in [YHH88].

As a typical candidate for all implementation projects of deductive database systems, ALGRES [CCG∗88] is mentioned here. A short overview of further projects is given in [CGT90].

A historical survey of the development of deductive databases as a research field can be found in [Min88].

#### 2.4.3 Aspects for the Integration

Because the integration of object–oriented concepts into deductive languages is treated in many different ways within the publications on languages mentioned above, we will give here only a brief overview of additional approaches.

[Ban86] is one of the first approaches towards a semantically sound integration. Further investigations can be found in [Ab90, Bee90]. In [Ab90] the introduction of explicit control for realizing updates is discussed. The access to complex objects in a rule–based language is considered for instance in [San88].

A special semantics for object–oriented logic programs is developed in [LSV90, LV90a, LV90b, LV91]. The three–valued *ordered logic* in which several well–known semantics for logic programming are reflected is the basis for the language LOCO (see 2.2).

#### 2.4.4 Further Aspects

A lot of related works remains which we have not mentioned yet. There are extensions of relational database systems, like POSTGRES [SJGP90, SRH90] and EXODUS [CDV88], projects in which logic programming is used to implement a relational system (e.g., [BF89]) and different proposals for specifications of objects.

Among the specification approaches, we can distinguish more general algebraic ones [SE90, FSMS90, EGS91], logical object specifications with inheritance [BL91] and specifications with temporal logic [SJ90, JSH91] for modeling database evolution.

Great efforts will have to be taken on the field of updates for object–oriented databases. First results for deductive (relational) databases [Ab88, dMS88, Dec90] show that we are not at the end of that way.

A very popular point is the development of graphical support tools. Graphical query languages for object–oriented databases are presented for instance in [GPvG90a, GPvG90b, STT91].

Finally, we refer to a state–of–the–art–report and a brief survey of the development of database theory in [Ull87].

## 3 Final Remark

This survey does not claim to be complete. In the present state, integration of object–orientation and deduction is a reseach area of high activity. Therefore a final survey is not possible today. However, the multitude of research activities and results achieved up to now calls for a preliminary survey of this field. Here we have tried to give such a survey.

## References

[Ab88] S. Abiteboul: *Updates: a new frontier.* Int. Conf. on Database Theory (ICDT'88), M. Gysens, J. Paredaens,D. Van Gucht (eds.), Springer LNCS 326, 1988, pp. 1–18.

[Ab90] S. Abiteboul: *Towards a deductive object-oriented database language.* Data & Knowledge Engineering, Vol. 5, No. 2, 1990, pp. 263–287.

[ABD*89] M. Atkinson, F. Bancilhon, D. DeWitt, K. Dittrich, D. Maier, S. Zdonik: *The Object-Oriented Database System Manifesto.* Proc. Int. Conf. on Deductive and Object-Oriented Databases (DOOD'89), W. Kim, J.-M. Nicolas, S. Nishio (eds.), North–Holland, 1990, pp. 223–240.

[ADJ*91] M. Ahlsen, A. D'Atri, P. Johannesson, E. Laenens, P. Rullo, P. Rossi, F. Staes, L. Tarantino, L. Van Beirendonck, F. Van Cadsand, W. Van Sant, J. Van Slembroeck, B. Verdonk, D. Vermeir: *The KIWIS Knowledge Base Management System.* Proc. 3rd Int. Conf. on Advanced Information Systems Engineering (CAISE'91), 1991.

[AG88] S. Abiteboul, S. Grumbach: *COL: A Logic–based Language for Complex Objects.* Proc. Int. Conf. on Extending Database Technology (EDBT'88), J.W. Schmidt, S. Ceri, M. Missikoff (eds.), Springer LNCS 303, 1988, pp. 271–293.

[AG91] S. Abiteboul, S. Grumbach: *A Rule–Based Language with Functions and Sets.* ACM Transactions on Database Systems, Vol. 16, No. 1, 1991, pp. 1–30.

[AH87] S. Abiteboul, R. Hull: *IFO: A Formal Semantic Database Model.* ACM Transactions on Database Systems, Vol. 12, No. 4, 1987, pp. 525–565.

[AH88] S. Abiteboul, R. Hull: *Data Functions, Datalog and Negation.* Proc. ACM SIGMOD Conf. on Management of Data, 1988, pp. 143–153.

[AK89] S. Abiteboul, P.C. Kanellakis: *Object Identity as a Query Language Primitive.* Proc. ACM SIGMOD Conf. on Management of Data, 1989, pp. 159–173.

[AN86] H. Ait–Kaci, R. Nasr: *LOGIN: A Logic Programming Language with Built–in Inheritance.* Journal of Logic Programming, Vol. 3, No. 3, 1986, pp. 185–215.

[Ban86] F. Bancilhon: *A logic programming object oriented cocktail.* ACM SIGMOD Record, Vol. 15, No. 3, 1986, pp. 11–21.

[Ban88] F. Bancilhon: *Object–Oriented Database Systems.* Proc. ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems, 1988, pp. 152–162.

[Bee90] C. Beeri: *A formal approach to object-oriented databases.* Data & Knowledge Engineering, Vol. 5, No. 5, 1990, pp. 353–382.

[Bee91] C. Beeri: *Theoretical Foundations for OODB's — a Personal Perspective.* IEEE Database Engineering, Vol. 14, No. 2, 1991, pp. 8–12.

[BF89] J.B. Bocca, J.C. Freytag: *Rules for Implementing Very Large Knowledge Base Systems.* ACM SIGMOD Record, Vol. 18, No. 3, 1989, pp. 29–35.

[BL91] S. Brass, U.W. Lipeck: *Semantics of Inheritance in Logical Object Specifications.* Proc. Int. Conf. on Deductive and Object-Oriented Databases (DOOD'91), 1991.

[BNR*87] C. Beeri, S. Naqvi, R. Ramakrishnan, O. Shmuelli, S. Tsur: *Sets and Negation in a Logic Database Language (LDL1).* Proc. ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, 1987, pp. 21–37.

[BNST91] C. Beeri, S. Naqvi, O. Shmuelli, S. Tsur: *Set Constructors in a Logic Database Language.* Journal of Logic Programming, Vol. 10, No. 3, 1991, pp. 181–232.

[CCC*90] F. Cacace, S. Ceri, S. Crespi-Reghizzi, L. Tanca, R. Zicari: *Integrating Object-Oriented Data Modeling with a Rule-Based Programming Paradigm.* Proc. ACM SIGMOD Conf. on Management of Data, 1990, pp. 225–236.

[CCG∗88] S. Ceri, S. Crespi–Reghizzi, G. Gottlob, F. Lamberti, L. Lavazza, L. Tanca, R. Zicari: *The ALGRES Project.* Proc. Int. Conf. on Extending Database Technology (EDBT'88), J.W. Schmidt, S. Ceri, M. Missikoff (eds.), Springer LNCS 303, 1988, pp. 551–555.

[CCT90] S. Ceri, F. Cacace, L. Tanca: *Object orientation and logic programming for databases: a season's flirt or long–term marriage?* Next generation information system technology, J.W. Schmidt, A.A. Stogny (eds.), Springer LNCS 504, 1990, pp. 124–143.

[CCT91] F. Cacace, S. Ceri, L. Tanca: *Consistency and NON–determinism in a Database Programming Language.* Proc. Math. Fundamentals of Database Systems (MFDBS'91), B. Thalheim, J. Demetrovics, H.-D. Gerhardt, Springer LNCS 495, 1991, pp. 325–341.

[CDLR90] S. Cluet, C. Delobel, C. Lécluse, R. Richard: *RELOOP, an algebra based query language for an object-oriented database system.* Data & Knowledge Engineering, Vol. 5, No. 2, 1990, pp. 333–352.

[CDV88] M.J. Carey, D.J. DeWitt, S.L. Vandenberg: *A Data Model and Query Language for EXODUS.* Proc. ACM SIGMOD Conf. on Management of Data, 1988, pp. 413–423.

[CGT89] S. Ceri, G. Gottlob, L. Tanca: *What You always wanted to Know About Datalog (And Never Dared to Ask).* IEEE Transaction on Knowledge and Data Engineering, Vol. 1, No. 1, 1989, pp. 146–166.

[CGT90] S. Ceri, G. Gottlob, L. Tanca: *Logic Programming and Databases.* Springer New York, 1990.

[CKW89] W. Chen, M. Kifer, D.S. Warren: *HiLog as a Platform for Database Languages (or why predicate calculus is not enough).* Proc. Int. Workshop on DB Programming Languages, R. Hull, R. Morrison, D. Stemple (eds.), Kaufmann, 1989, pp. 315–329.

[Cod70] E.F. Codd: *A Relational Model of Data for Large Shared Data Banks.* Communications of the ACM, Vol. 13, No. 6, 1970, pp. 377–387.

[CS88] M. Caruso, E. Sciore: *Meta–Functions and Contexts in an Object–Oriented Database Language.* Proc. ACM SIGMOD Conf. on Management of Data, 1988, pp. 56–65.

[CW89] W. Chen, D.S. Warren: *C-Logic of Complex Objects.* Proc. ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, 1989, pp. 369–378.

[Dec90] H. Decker: *Drawing Updates from Derivations.* Int. Conf. on Database Theory (ICDT'90), S. Abiteboul, P.C. Kanellakis (eds.), Springer LNCS 470, 1990, pp. 437–451.

[Dit90] K. Dittrich: *Object–oriented Database Systems: The next Miles of the Marathon.* Information Systems, Vol. 15, No. 1, 1990, pp. 161–167.

[dMS88] C. de Maindreville, E. Simon: *Modelling queries and updates in a deductive database.* Proc. Int. Conf. on Very Large Data Bases (VLDB'88), F. Bancilhon, D.J. DeWitt (eds.), Kaufmann, 1988, pp. 395–406.

[EGS91] H.-D. Ehrich, M. Gogolla, A. Sernadas: *Objects and their Specification.* Proc. 8th Workshop on Abstract Data Type, Springer LNCS, 1991.

[El91] A. Eliëns: *Distributed Logic Programming for Artificial Intelligence.* AICOM, Vol. 4, No. 1, 1991, pp. 11–21.

[FSMS90] J. Fiadeiro, C. Sernadas, T. Maibaum, G. Saake: *Proof–Theoretic Semantics of Object–Oriented Specification*

*Constructs.* Proc. IFIP Conf. on Object–Oriented Databases: Analysis, Design & Construction (Preprint), 1990.

[GH91] M. Gogolla, U. Hohenstein: *Towards a Semantic View of an Extended Entity-Relationship Model.* ACM Transactions on Database Systems, Vol. 16, No. 3, 1991, pp. 369–416. Preliminary version (HG88): Technical University of Braunschweig, Informatik-Bericht Nr. 88-02, 1988.

[GMN84] H. Gallaire, J. Minker, J.-M. Nicolas: *Logic and Databases: A Deductive Approach.* ACM Computing Surveys, Vol. 16, No. 2, 1984, pp. 153–185.

[GN90] H. Gallaire, J.-M. Nicolas: *Logic and Databases: An Assessment.* Int. Conf. on Database Theory (ICDT'90), S. Abiteboul, P.C. Kanellakis (eds.), Springer LNCS 470, 1990, pp. 177–186.

[GPvG90a] M. Gyssens, J. Paradaens, D. van Gucht: *A graph–oriented object database model.* Proc. SIGACT SIGMOD SIGART Symp. on Principles of Database Systems, 1990.

[GPvG90b] M. Gyssens, J. Paradaens, D. van Gucht: *A graph–oriented object database model for database end user interfaces.* Proc. ACM SIGMOD Conf. on Management of Data, 1990.

[Heu87] A. Heuer: *A Data Model for Complex Objects Based on a Semantic Database Model and Nested Relations.* Proc. Nested Relations and Complex Objects in Databases, S. Abiteboul, P.C. Fischer, H.-J. Schek (eds.), Springer LNCS 361, 1989, pp. 297–312.

[HM81] M. Hammer, D. McLeod: *Database Description with SDM: A Semantic Database Model.* ACM Transactions on Database Systems, Vol. 6, No. 3, 1981, pp. 351–386.

[HS89] A. Heuer, P. Sander: *Semantics and Evaluation of Rules over Complex Objects.* Proc. Int. Conf. on Deductive and Object-Oriented Databases (DOOD'89), W. Kim, J.-M. Nicolas, S. Nishio (eds.), North–Holland, 1990, pp. 473–492.

[HS90] A. Heuer, P. Sander: *Preserving and Generating Objects in the LIVING IN A LATTICE Rule Language.* Proc. Int. Workshop on Foundations of Models and Languages for Data and Objects, J. Göers, A. Heuer (eds.), Technical University of Clausthal, Informatik–Bericht 90/3, 1990, pp. 1–36.

[HS91] A. Heuer, P. Sander: *Classifying Object–Oriented Query Results in a Class/ Type Lattice.* Proc. Math. Fundamentals of Database Systems (MFDBS'91), B. Thalheim, J. Demetrovics, H.-D. Gerhardt, Springer LNCS 495, 1991, pp. 14-28.

[HY90] R. Hull, M. Yoshikawa: *ILOG: Declarative Creation and Manipulation of Object Identifiers.* Proc. Int. Conf. on Very Large Data Bases (VLDB'90), 1990.

[JSH91] R. Jungclaus, G. Saake, T. Hartmann: *Language Features for Object-Oriented Conceptual Modeling.* Proc. 10th Int. Conf. on the ER-approach (T.J. Teorey, ed.), San Mateo, E/R Institute, 1991, 309–324.

[KC86] S.N. Khoshafian, G.P. Copeland: *Object Identity.* OOPSLA '86 Proceedings, ACM SIGPLAN Notices Vol. 21, No. 11, 1986, pp. 406–416.

[Kes90] F.N. Kesim: *A Survey of Object Logics for Object–Oriented Databases.* Imperial College London, Internal Report, 1990.

[Kil89] H. Kilov: *Rewievs of object–oriented papers.* ACM SIGMOD Record, Vol. 18, No. 4, 1989, pp. 50–55.

[Kim90] W. Kim: *Object–oriented Databases: Definition and Research Directions.* IEEE Transaction on Knowledge and Data Engineering, Vol. 2, No. 3, 1990, pp. 327–341.

[Kim91] W. Kim: *Object–oriented database systems: strengths and weaknesses.* Journal of Object–Oriented Programming, Vol. 4, No. 4, 1991, pp. 21–29.

[KL89] M. Kifer, G. Lausen: *F-Logic: A Higher-Order Language for Reasoning About Objects, Inheritance, and Scheme.* Proc. ACM SIGMOD Conf. on Management of Data, 1989, pp. 134–146.

[KLW90] M. Kifer, G. Lausen, J. Wu: *Logical Foundations of Object–Oriented and Frame–Based Languages.* Informatik–Manuskript 3/1990, University of Mannheim, 1990.

[Kup87] G. Kuper: *Logic Programming with Sets.* Proc. ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, 1987, pp. 11–20.

[Kup88] G. Kuper: *On the Expressive Power of Logic Programming Languages with Sets.* Proc. ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, 1988, pp. 10–14.

[Kup90] G. Kuper: *Logic Programming with Sets.* Journal of Computer and System Sciences, Vol. 41, No. 1, 1990, pp. 44–64.

[KW89] M. Kifer, J. Wu: *A Logic for Object–Oriented Logic Programming (Maier's O-Logic Revisited).* Proc. ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, 1989, pp. 379–393.

[LK86] P. Lyngbaek, W. Kent: *A Data Modeling Methodology for the Design and Implementation of Information Systems.* K.R. Dittrich, U. Dayal (ed.), Proc. Int. Workshop on Object-Oriented Database Systems, Pacific Grove (California), 1986, pp. 6–17.

[Llo84] J.W. Lloyd: *Foundations of Logic Programming.* Springer, New York, 1984.

[LR89] C. Lécluse, P. Richard: *The $O_2$ Database Programming Language.* Proc. Int. Conf. on Very Large Data Bases (VLDB'89), P.M.G. Apers, G. Wiederhold (eds.), Kaufmann, 1989, pp. 411–422.

[LSV90] E. Laenens, D. Sacca, D. Vermeir: *Extending Logic Programming.* Proc. ACM SIGMOD Conf. on Management of Data, 1990, pp. 184–193.

[LV90a] E. Laenens, D. Vermeir: *A Fixpoint Semantics for Ordered Logic.* Journal of Logic Computation, Vol. 1, No. 2, 1990, pp. 159–185.

[LV90b] E. Laenens, D. Vermeir: *A Logical Basis for Object Oriented Programming.* Logics in AI — European Workshop JELIA '90, J. van Eijck (Ed.), LNAI 478, Springer, 1991, pp. 317–332.

[LV91] E. Laenens, D. Vermeir: *On the Relationship between Well–Founded and Stable Partial Models.* Proc. Math. Fundamentals of Database Systems (MFDBS'91), B. Thalheim, J. Demetrovics, H.-D. Gerhardt, Springer LNCS 495, 1991, pp. 59–73.

[LVVS90] E. Laenens, B. Verdonk, D. Vermeir, D. Sacca: *The LOCO Language: Towards an Integration of Logic and Object Oriented Programming.* University of Antwerpen, Report 90-09, 1990.

[Mai86] D. Maier: *A Logic for Objects.* Proc. Workshop on Foundations of Deductive Databases and Logic Programming (Preprint), Washington D.C., 1986, pp. 6–26.

[Min88] J. Minker: *Perspectives in Deductive Databases.* Journal of Logic Programming, Vol. 5, No. 1, 1988, pp. 33–60.

[MW80] J. Mylopoulos, H.K.T. Wong: *Some Features of the TAXIS Data Model.* Proc.

Int. Conf. on Very Large Data Bases (VLDB'80), 1980, pp. 399–410.

[MW90] T. Matsushima, G. Wiederhold: *A Model of Object–Identities and Values.* Stanford University, Dep. of Computer Science, Report 90-1304, 1990.

[MZO89] D. Maier, J. Zhu, H. Ohkawa: *Features of the TEDM Object Model.* Proc. Int. Conf. on Deductive and Object-Oriented Databases (DOOD'89), W. Kim, J.-M. Nicolas, S. Nishio (eds.), North–Holland, 1990, pp. 511–530.

[NT89] S.A. Naqvi, S. Tsur: *A Logical Language for Data and Knowledge Bases.* Computer Science Press, 1989.

[O∗90] O. Deux et. al.: *The Story of $O_2$.* IEEE Transaction on Knowledge and Data Engineering, Vol. 2, No. 1, 1990, pp. 91–108.

[PA86] P. Pistor, F. Andersen: *Designing a Generalized NF2 Model with an SQL-Type Language Interface.* Proc. Int. Conf. on Very Large Data Bases (VLDB'86), W. Chen, G. Gardarin, S. Ohsuga (eds.), Kaufmann, 1986, pp. 278–285.

[Ram90] I. Ramos: *Logics and OO–Data Bases: a declarative approach.* Proc. Int. Conf. Database and Expert Systems Applications (DEXA'90), A.M. Tjoa, R. Wagner (eds.), Springer, 1990, pp. 229–233.

[San88] P. Sander: *Access to Complex Objects in a Rule–Based Language.* Proc. Int. Workshop on Relational Databases and their Extensions, V. Brosda (ed.), Technical University of Clausthal, Informatik–Bericht 88/4, 1988, pp. 91–119.

[SE90] A. Sernadas, H.-D. Ehrich: *What is an object, after all?* Proc. IFIP Conf. on Object Oriented Databases: Analysis, Design & Construction (Preprint), 1990.

[She90] Y.-H. Sheng: *IDLOG: Extending the Expressive Power of Deductive Database Languages.* Proc. ACM SIGMOD Conf. on Management of Data, 1990, pp. 54–63.

[SJ90] G. Saake, R. Jungclaus: *Information about Objects versus Derived Objects.* Proc. Int. Workshop on Foundations of Models and Languages for Data and Objects, J. Göers, A. Heuer (eds.), Technical University of Clausthal, Informatik–Bericht 90/3, 1990, pp. 59–70.

[SJGP90] M. Stonebraker, A. Jhingran, J. Goh, S. Potamianos: *On Rules, Procedures, Caching and Views in Data Base Systems.* Proc. ACM SIGMOD Conf. on Management of Data, 1990, pp. 281–290.

[SRH90] M. Stonebraker, L.A. Rowe, M. Hirohama: *The Implementation of POSTGRES.* IEEE Transactions on Knowledge and Data Engineering, Vol. 2, No. 1, 1990, pp. 125–142.

[SS86] H.-J. Schek, M.H. Scholl: *The Relational Model with Relation-Valued Attributes.* Information Systems, 1986, Vol. 11, No. 2, pp. 137–147.

[STT91] F. Staes, L. Tarantino, A. Tiems: *A Graphical Query Language for Object Oriented Databases.* 1991, Internal Report, University of Antwerpen.

[STV91] F. Staes, L. Tarantino, B. Verdonk: *A logic approach for supporting queries in object oriented databases.* IFIP Working Conf. on the Object–Oriented Approach in Information Systems, 1991.

[STVV91] F. Staes, L. Tarantino, B. Verdonk, D. Vermeir: *Supporting User Interaction with OODB's: A Declarative Approach.* Proc. Int. Conf. on Database and Expert System Applications (DEXA'91), D. Karagiannis (ed.), Springer, 1991, pp. 210-215.

[TZ86] S. Tsur, C. Zaniolo: *LDL: A Logic–Based Data–Language.* Proc. Int. Conf. on Very Large Data Bases (VLDB'86),

W. Chen, G. Gardarin, S. Ohsuga (eds.), Kaufmann, 1986, pp. 33–41.

[Ull85] J.D. Ullman: *Implementation of Logical Query Languages for Databases.* ACM Transactions on Database Systems, Vol. 10, No. 3, 1985, pp. 289–321.

[Ull87] J.D. Ullman: *Database Theory: Past and Future.* Proc. ACM SIGACT–SIGMOD–SIGART Symp. on Principles of Database Systems, 1987, pp. 1–10.

[YHH88] C. Youn, L.J. Henschen, J. Han: *Classification of Recursive Formulas in Deductive Databases.* Proc. ACM SIGMOD Conf. on Management of Data, 1988, pp. 320–328.

[Zan89] C. Zaniolo: *Object Identity and Inheritance in Deductive Databases — an Evolutionary Approach.* Proc. Int. Conf. on Deductive and Object-Oriented Databases (DOOD'89), W. Kim, J.-M. Nicolas, S. Nishio (eds.), North–Holland, 1990, pp. 7–24.

[ZM89] J. Zhu, D. Maier: *Computational Objects in Object–Oriented Data Models.* Proc. Int. Workshop on DB Programming Languages, R. Hull, R. Morrison, D. Stemple (eds.), Kaufmann, 1989, pp. 139–160.