ELECTRONIC WORKSHOPS IN COMPUTING

Series edited by Professor C.J. van Rijsbergen

Rainer Manthey and Viacheslav Wolfengagen (Eds)

Advances in Databases and Information Systems 1997

Proceedings of the First East-European Symposium on Advances in Databases and Information Systems, (ADBIS'97), St Petersburg, 2-5 September 1997

A Practitioner's View to the Integration of Virtual Enterprise Database Systems by Federation Techniques

Toralf Unger and Michael Höding

Published in collaboration with the
British Computer Society

A Practitioner's View to the Integration of Virtual Enterprise Database Systems by Federation Techniques

Toralf Unger

Agricultural Social Insurance of Hanover
Hanover, Germany

Michael Höding

Institute for Technical Information Systems, Otto-von-Guericke-University Magdeburg
Magdeburg, Germany

Abstract

In this paper we discuss the application of federation techniques for virtual enterprises. Virtual enterprises are a kind of cooperation between independent companies with a common goal. Requirements to an integrating information system are presented. We describe the integration of local information systems by a federated database system (FDBS) and discuss how an FDBS can meet the requirements. Additionally we compare different approaches by an estimation of benefits.

1 Introduction

Virtual enterprises seem to become a new paradigm in economics. A virtual enterprise [1] is a mutual connection of two or more real enterprises for realizing a common goal. An essential requirement for this kind of organization is the use of modern information technologies [2], e.g. group-ware supporting cooperative work [3]. Due to the fact that virtual enterprises are often established for a limited time, the development of a new shared information system is too costly. In addition, the partners in a virtual enterprise may be competitors in other business fields. Therefore, we have to deal with strong requirements for security mechanisms. In our opinion a suitable approach to the integration of local company information systems into one (virtual) system of the virtual enterprise could be the use of a federated database system.

To evaluate this idea we organized the remainder of this paper as follows: First we briefly introduce the basic ideas of virtual enterprises and discuss their requirements to an integrating software system influenced by a practitioners point of view. Then we sketch the application of an FDBS as the database system of the virtual enterprise. Moreover, we discuss how FDBS can fulfill the presented requirements. Then, we compare our approach with other possible integration approaches by the means of an estimation of benefits. Finally, we conclude by giving an outlook to future work.

2 Virtual Enterprises

An essential problem of medium and big enterprises is the lack of flexibility. To overcome this problem hierarchical structures in many enterprises are currently replaced by more process oriented organization structures [1]. Slogans like *business process re-engineering* and *group work* promise the solution of every problem inside the enterprises. Another catchword which is more related with our topic is *outsourcing*. This means that specific tasks of the business process, which do not belong to the main business tasks, will be transferred to other enterprises. In contrast to the transfer of subtasks a virtual enterprise is a common activity of two or more partners with equal rights.

Figure 1: Users, tasks and data in the context of virtual enterprises

According to [1, 4] a virtual enterprise is a type of corporation, which

- is established for a limited time of cooperation
- by partners (enterprises or persons) with equal rights
- to achieve a common goal in a specific business field for a mutual advantage,
- using modern information technology for cooperation in
- flexible networked organization structures
- without a central office or organization chart.

As the definition shows, the information system is an essential part of the virtual enterprise. Commonly, management and staff on the virtual level need up-to-date information from the information systems of the partner companies for high flexible business activities. To illustrate the overlapping domains of information systems, we sketch in figure 1 users, tasks, schema objects (classes), and data objects in the context of a virtual enterprise established by two companies. We simplify the local databases and show only the two classes *Products* and *Customers*. However, often these classes are the central entities in such an integration scenario. The data objects in the databases of the partner companies are partially the *same* objects like in the database of the virtual enterprise. For instance a customer of company A, which is represented by a database object in the database of company A, can also be a customer of the virtual enterprise. Therefore, the customer data should also be accessible from the virtual enterprise information system. Commonly, the extensions, i.e. the collection of data-objects, of local databases are partially overlapping in virtual enterprises. A subset of this overlapping part is the database of the virtual enterprise. We have to point out, that not the complete overlapping part should be accessible from the virtual enterprise level because it may contain critical or confidential data, that belongs to business fields where the partners are competitors.

Figure 2: Schema levels of a federated database system

Above the schema level we find the level of tasks, representing the organization structure of the enterprise. Tasks on the organization structure level are corresponding with application programs. They are solved by users or user groups on the top level of figure 1. As depicted these users or user groups can be the *same* in the real local enterprise and in the virtual enterprise.

Now, we will illustrate these facts by the means of an example. A popular example for a virtual enterprise is the cooperation between the German railway company Deutsche Bahn AG and the bank house Citibank. Both enterprises established a virtual enterprise to support the Bahn-Card, which allows one to buy half-price railway tickets for a fixed price in the period of a year in combination with the credit card of the Citibank. Both partners have major advantages by this cooperation. The product Bahn-Card is more attractive for the customer since it is usable as a free-of-charge credit card, too. The Citibank got access to the customers of the railway company and can additionally use the service of railway ticket counters for selling its credit cards. We like to point out, that the counter clerks of the railway company functions as staff of the virtual enterprise, too. The alliance was closed for a limited time because it was not sure that the combined Bahn-Card and VISA-Card would become a success.

However, in this example the structure of the virtual enterprise and the established information system (based on file exchange) is quite simple, since the virtual enterprise provides only one product and the partners are not competitors in a common market. Other more complex scenarios with a large amount of products or a bigger number of different products show a need for a common information system on the level of the virtual enterprise. One reason could be the common production of customer specific products. Often it is necessary to know the current production state or place of a specific product. Commonly, in a company such information is managed by a information system, e.g. a product data management system. Obviously, for the virtual information system it is necessary to establish a system providing similar functionality like usual systems.

Based on this reflection we derive the following essential requirements to an integrating information system, which are motivated by a practitioner's point of view:

1. *Homogeneous View to the Integrated Data:*

The central purpose of the information system is the integration of distributed data. Beside this, a homogeneous access to the integrated data would allow a comfortable and easy use of the shared information and therefore fast and effective application program development for the virtual enterprise.

2. *Global Read and Write Operations:*

To decrease the time of information exchange and decision making it is necessary to provide up-to-date

information on global level as well as on local level. To support correct information the information system of the virtual enterprise should provide global read and write operations which have an immediate access to the local data bases.

3. *Global Transactions:*

Normally the global system should provide similar transaction services like the local database systems. If this is not possible for all transactions there is a need to specify a set of critical transactions which are secure.

4. *Global Integrity Maintenance:*

For objects that are stored in different local databases a global integrity maintenance would be a useful benefit of an integrating information system. Especially for critical data such mechanism are necessary.

5. *Local Security:*

Partners in a virtual enterprise can be competitors in other business fields. Therefore, from a practitioners point of view it is one of the most important requirements to provide security mechanisms restricting the global view on the local data. Only information which are necessary for the virtual enterprise should be integrated.

6. *Local Systems must be Preserved:*

Due to the fact that the virtual enterprise and its information system could exist only for a limited time the local information systems must be preserved. In that way, the costs for establishing a global information system can be decreased.

7. *Flexibility:*

Commonly, the local information systems are heterogeneous on hardware, data model, schema and data level. An integrating information system has to cover this heterogeneity without changing the local information systems. Due to the fact that the virtual enterprise can exist only a limited time another important aspect of flexibility has to be considered: The possibility to decouple a local system from the integrating system without major efforts.

8. *Reuse of Local Applications on Global Level:*

The possibility to reuse application programs or user interfaces with that the user is familiar would be useful since the user on the virtual enterprise level is always the same person like a user in one of the partner companies. To support reuse the integrating system has to offer an interface which is compatible with the corresponding local database interface.

Beside this, there are a number of common requirements, e.g. performance, fault tolerance of hardware, etc. which depend more on the concrete configuration of the technical environment than on the concepts of the integration software.

3 Basics of FDBS

Federated database systems are suggested as an appropriate solution for the integration of heterogeneous information systems. In this section we briefly present the essential foundations of federated database systems (FDBS). A federated database system is a distributed system consisting of different heterogeneous database systems [5]. These cooperating but *autonomous* local systems are coupled by the federated database management system (FDBMS). The FDBS provides a uniform and transparent interface to the heterogeneous and distributed data sources of independently developed local systems. Therefore, new global applications have the possibility to read and write the federated data. The FDBS communicates with the local systems using their common user interfaces. In that way the local database systems and application programs can remain unchanged. For federated database systems the 5-level-schema-architecture of [6] is generally accepted (cf. figure 2):

Figure 3: A federated information system for a virtual enterprise

- *Local Schemata*: The local schema is the conceptual schema of the component database system (CDBS). It is expressed in the data-model of the component system. On this level schemata are based on different data-models.
- *Component Schemata*: To overcome data-model heterogeneity, the local schemata have to be transformed from the native data-model to a common data-model (the canonical data model of the FDBMS).
- *Export Schemata*: Export schemata filter a subset of schema information. If the local system offers view mechanisms it is possible for the administrator of the local system to use this views to restrict schema and data export. In other words, only exported schema elements and data are visible to the federation layer.
- *Federated Schema*: The federated schema is an integrated schema, which is the result of the integration of the export schemata. On this level schema heterogeneity is overcome. The federated schema is the schema of the global interface.
- *External Schemata*: External schemata serve specific views for different applications. This can include the change of the data model. Moreover, external schemata can be identical with local schema. In that way it is possible to use local application programs on the global level.

In conclusion, we can distinguish two main tasks for the design process. These are the *schema transformation* to overcome the data model heterogeneity and the *schema integration* to overcome the schema heterogeneity [7].

For the design of federated database systems different integration methodologies are available. Many methodologies use a semantical rich data-model, e.g. an object-oriented data-model [8, 9, 10]. The main idea of these approaches is based on the assumption that the rich object-oriented data models can accommodate the most concepts used by other data-models. Other integration methodologies are based on semantical poor data models, arguing that too many modeling concept leads to schema heterogeneity implying problems in the schema integration step. The *Generic Integration Model* GIM [11, 12], which is developed especially for the FDBS design, can be characterized as a semantical poor data model.

4 A Federated Information System for Virtual Enterprises

The definition of a federated database system as virtual view to the integrated data of different information sources implies the idea to use an FDBS as data server for the virtual enterprise.

In figure 3 we sketched the application of an FDBS for a virtual enterprise consisting of two companies. Both companies are independent and own their specific information systems. We depicted the local information systems consisting of database system and application programs as well as the users.¹ The business of company A is development, production and sale of fork lift trucks. The information about customers and products are stored in a relational database systems. On the other hand, the CAD/CAM systems for development and production of machines are based on an object-oriented database system. Company B offers a country wide service for sale, maintenance and repair of transport machines. For the management of information about repairs or other services the company uses a relational database system. The benefits of a virtual enterprise for both partners are obvious. Company B has direct access to the customer information and specific technical documentation of company A. On the other hand, company A can use the distribution system of company B. A closer look to the employees illustrates the fact that employees of company A or B who are users of the local data, can be also part of the global organization of the virtual enterprise. In this function they have access to the integrated data via the federation service. According to this discussion we think that federated database systems are a possible compromise solution for virtual enterprises.

sub-goal	SG1	SG2	SG3	SG4	SG5	SG6	SG7	SG8	Sum	Weight
SG1: Integration	x	2	1	1	1	0	2	1	8	0.6
SG2: global Read & Write	0	x	2	1	0	0	1	2	6	0.5
SG3: global Transactions	1	0	x	1	0	0	1	2	5	0.4
SG4: global Consistency	1	1	1	x	0	0	1	1	5	0.4
SG5: local Security	1	2	2	2	x	1	2	2	12	0.9
SG6: preserve local IS	2	2	2	2	1	x	2	2	13	1.0
SG7: Flexibility	0	1	1	1	0	0	x	2	5	0.4
SG8: Reuse of local Appl.	1	0	0	1	0	0	0	x	2	0.2

Figure 4: Calculation of weights for sub-goals

Now we will discuss how a federation service can fulfil the requirements (cf. section 2) of virtual enterprises to an integrating information system:

1. *Homogeneous View to the Integrated Data:*

One of the main tasks of a FDBS is to provide a homogeneous global view to integrated data [6, 5]. Therefore FDBS will fulfil this requirement as far as possible. The reason for this limitation is the possible occurrence of not resolvable conflicts.

2. *Global Read and Write Operations:*

According to the concept of FDBS as a database system with the usual database functionality, global read and write operations are supported. Such operations are part of each suggested canonical data model (cf. [8, 9, 11], etc.).

3. *Global Transactions:*

Global transactions are still a critical problem in FDBS design. As argued in [13] for global transaction management a restriction of local autonomy has to be accepted.

4. *Global Integrity Maintenance:*

Different approaches for global integrity maintenance are possible. In the FDBS prototype Efendi [9] a consistency maintenance, driven by global queries, is implemented. However, in this way temporary

¹In this case *user* is a unit which describes a position of responsibility which can be fulfilled by a person as well as by a group of persons.

inconsistencies between different local databases are possible. A solution for this problem was suggested in [14] using active mechanisms for immediate global consistency maintenance. Concepts for the integration and derivation of integrity constraints during the FDBS design process were presented in [12].

5. *Local Security:*

As argued before, strong security concepts for the information system are indispensable. The concept of local autonomy [6] is a good base for security concepts [15]. However, a federated database management system should offer as much as necessary but no other information to the global users. Therefore the adoption of local security mechanisms for the global interface should be possible. Moreover, the local security models should be integrated during the FDBS design process. Only a FDBMS which protects the local database against unauthorized access is acceptable for a virtual enterprise. In our opinion the problem of security in FDBS has still to be solved.

6. *Local Systems must be Preserved:*

The FDBS concept of local autonomy guarantees the preservation of the local information systems without major changes [6].

7. *Flexibility:*

The integration of databases which differ in hardware and software is an essential task of FDBS. Hardware heterogeneity can be overcome by networked computer infrastructures. The integration of databases with different data model [7, 6] as well as for file based system [16, 17] can be supported.

8. *Reuse of Local Applications on Global Level:*

The use of local application programs on the global level is supported by the concept of external schemata according to the five-level schema architecture of [6]. In that way, an FDBMS can offer an interface which presents the integrated data with the same structure as the corresponding local database interface. Especially the integration approach of [18] supports the derivation of such external schemata.

5 Comparison with Other Integration Approaches

To evaluate the presented FDBS based approach for the application domain of virtual enterprises, now we analyze its benefits in comparison with other possible integration architectures. The first idea is to establish a new centralized database system that obviously provides the usual database functionalities (scenario S1). The second approach remains the local information systems unchanged and supports a minimal integration based on data-exchange by periodical file-transfer (scenario S2). We compare this two extrema with “state of the art”-FDBS (scenario S3) and “future”-FDBS (scenario S4). The first describes our view of the FDBS prototypes and concepts that are available nowadays. The latter is a imaginary FDBS that should and could be designed in the future.

We use the *Estimated Benefits Analysis Method* of [19] to compare the scenarios. This method delivers a benefit value for each scenario and therefore it helps a decisionmaker to estimate how a scenario fits with his expectations. At first we briefly introduce the main idea of the comparison method. The first step to calculate benefit values is to put up a system of objectives. The main objective of the application domain of virtual enterprises is the integrated use of data from the different enterprises information systems. To fulfil the main goal one has to derive a number of goal criterions or sub-goals. According to section 2 we derive the sub-goals: homogeneous and integrated view (SG1), global read and write (SG2), global transactions (SG3), global consistency control (SG4), local security (SG5), preservation of local IS (SG6), flexibility (SG7), and reuse local applications on the global level (SG8).

The second step is the determination of weights for the sub-goals. For that [19] proposes a step by step comparison algorithm. In figure 4 we apply this algorithm to our application domain. The subgoals are noted at the columns and rows of the table. Then, we compare each sub-goal with all other. If the subgoal A is more important for the achievement of the main goal than a sub-goal B, the sub-goal A gets the value “2” and B the

requirement	weight	central DBS		independ. DBS		FDBS nowadays		FDBS future	
1. Integration	0.6	0.8	= 0.48	0	= 0.00	0.7	= 0.42	0.8	= 0.48
2. global Read & Write	0.5	1	= 0.50	0	= 0.00	1	= 0.50	1	= 0.50
3. global Transactions	0.4	1	= 0.40	0	= 0.00	0.4	= 0.16	0.8	= 0.32
4. global Consistency	0.4	1	= 0.40	0	= 0.00	0.4	= 0.16	0.8	= 0.32
5. local Security	0.9	0	= 0.00	1	= 0.90	0.5	= 0.45	0.9	= 0.81
6. preserve local IS	1.0	0	= 0.00	1	= 1.00	0.9	= 0.90	0.9	= 0.90
7. Flexibility	0.4	0	= 0.00	0.6	= 0.24	0.8	= 0.32	1	= 0.40
8. Reuse local Appl.	0.2	0	= 0.00	0	= 0.00	0.5	= 0.10	0.9	= 0.18
	= 4.40		= 1.78		= 2.14		= 3.01		= 3.95

Figure 5: Estimated benefits of different approaches

value “0”. If there is no major difference both sub-goals get a “1”. The values are summarized and normalized to 1.

The third step of the method is the estimation how every sub-goal is fulfilled by the different scenarios. This levels have to be estimated by the decisionmaker or the decision team. We have to point out that the values strongly depend from knowledge, experience, and viewpoint of the decisionmaker. As argued in [19], in the most cases this evaluation delivers subjective but useful results.

In figure 5 we present the estimated suitability for each scenario. In the first column of the table we list the requirements or subgoals, discussed in section 2. For each subgoal we list the calculated weights for the virtual enterprise domain. In the following columns we note the estimated level of realization for the four scenarios mentioned above from a practitioner's point of view. We multiply the levels of realization with the corresponding weights and summarize these values for each scenario. In the sequel, we briefly discuss the estimated levels of realization:

1. Homogeneous and Integrated Access to the Data:

- **S1:** Obviously, a new established central DBS fulfils this requirement. However, during the integration into one central system not all possible data conflicts can be solved.
- **S2:** There is no possibility for an integrated access to the distributed data source.
- **S3:** This is an eternal task for the FDBS approach. However not all conflicts can be solved today.
- **S4:** Hence strong research efforts future FDBS should fulfill this requirement better.

2. Global Read and Write Operations:

- **S1:** A central DBS offers read and write operations without restrictions.
- **S2:** There are no global read or write operations available.
- **S3:** FDBS allows global read as well as global write operations.
- **S4:** Already fulfilled by nowadays FDBS.

3. Global Transactions:

- **S1:** Nowadays database systems offers global transactions.
- **S2:** Commonly there is no global transaction management.
- **S3:** Some approaches support limited transactions for FDBS.
- **S4:** Future FDBS should provide an enhanced support for different kinds of global transactions.

4. Global Integrity Maintenance:

- **S1:** Once all data are migrated into one central DBS the DBS offers consistency control.
- **S2:** There are no mechanisms for global consistency maintenance.
- **S3:** Some (mostly passive) mechanisms support global consistency and redundancy control.
- **S4:** Future FDBS should offer more consistency control, e.g. based on active mechanisms.

5. *Local Security:*

- **S1:** Due to the merging of databases from different partner enterprises during the system integration there is no local security.
- **S2:** The restricted file exchange controlled by local database administrators, guarantees a maximum of local security.
- **S3:** The concept of local autonomy as well as the concept of export schemata for filtering offers useful functionality to support local security.
- **S4:** Enhanced mechanisms, e.g. global roles of local users and inheritance of local rights to the global level should support a higher degree of local security.

6. *Local Systems must be Preserved:*

- **S1:** The local systems are replaced by a new system.
- **S2:** The local systems can remain unchanged.
- **S3:** Local systems can mostly remain unchanged.
- **S4:** Same as S3.

7. *Flexibility:*

- **S1:** This solution overcomes heterogeneity by replacement of hardware and software. After the integration into one central DBS there is no real chance to go back.
- **S2:** To overcome heterogeneity it is necessary to specify and use common exchange formats. This problem is related with schema transformation and integration in FDBS design. On the other hand, each local information system can draw back from the virtual enterprise without major problems. Due to the merging of exported data it is difficult to separate the merged data.
- **S3:** FDBS covers heterogeneity on hardware and software level. Hence an FDBS supports only a virtual integration of data, there are no problems with data separation. However, nowadays FDBS have problems to deal with missing local systems.
- **S4:** The mentioned open problem should be solved in future FDBS.

8. *Reuse of Local Applications on Global Level:*

- **S1:** All applications have to be developed again for the interface of the new DBS.
- **S2:** There is no global level.
- **S3:** For specific cases, e.g. a local application use the common data model and the common data model offers views, the reuse on global level is possible.
- **S4:** New concepts should support equivalent interfaces on global level by specific adapters.

Beside this short informal discussion we have to point out that the calculated values are influenced by our subjective point of view. In our opinion the results presented in figure 5 could be a starting point for further discussions and give an overview about specific advantages and penalties of the different integration approaches for a virtual enterprises information system.

How expected the centralized approach (S1) fulfils only the first four requirements whereas the file exchange based approach (S2) only supports the requirements five to seven. Both approaches are not suitable for virtual

enterprises. However, we think that the file exchange architecture is the more reasonable solution since the costs, which are not considered in the estimation table, are much lower. Furthermore the table shows that already “state-of-the-art” FDBS are a good compromise for virtual enterprises. Additionally, the estimated values for future FDBS show the necessity of purposeful scientific work. These are especially the topics security, transactions and global consistency maintenance in FDBS.

6 Conclusion and Outlook

In this paper we discussed an approach to the application of federations techniques for virtual enterprises. We briefly introduced the foundations of virtual enterprises and presented requirements of an integrating information system. Additionally we described the application of FDBS in a virtual enterprise and discussed how FDBS can meet the requirements. Moreover, we compared the use of FDBS with other thinkable integration solutions by the means of a quantified estimation of benefits.

The results show a strong need for the investigation of security concepts in FDBS for the application in virtual enterprises. Also the problems of global consistency maintenance and global transactions have to be solved to improve the suitability of FDBS as a solution for virtual enterprises. Nevertheless, the results show that already “state of the art” FDBS are a promising solution for database integration in the discussed application domain.

Further work should examine other approaches based on component oriented architecture, e.g. CORBA [20]. Beside this, a hybrid system, consisting in an integrated part for (time) critical data based on federation techniques and a less critical part based on file exchange could be a suitable and less costly alternative.

Many thanks to Andrea Diekmann, Steffen Geschke, Gunter Saake, and Constance Schröter for many creative discussion and useful comments.

References

- [1] W. H. Davidow and M. S. Malone. *The Virtual Corporation: Structuring and Revitalizing the Corporation for the 21st Century*. New York, 1992.
- [2] E. K. Clemons and M. C. Row. Rosenbluth international alliance: Information technology and the global virtual corporation. *IEEE*, (1/92):678–685, 1992.
- [3] L. Nastansky and W. Hilpert. The GroupFlow System: A Scalable Approach to Workflow Management between Cooperation and Automation. In B. Wolfinger, editor, *Proc. of 24th GI annual conference during the 13th World Computer Congress IFIP '94*, pages 473–479, Springer-Verlag, Berlin, 1994.
- [4] P. McHugh, G. Merli, and B. A. Wheeler III. *Beyond Business Process Reengineering: Towards the Holonic Enterprise*. Chichester, 1995.
- [5] D. Heimbigner and D. McLeod. A Federated Architecture for Information Management. *ACM Transactions on Office Information Systems*, 3(3):253–278, July 1985.
- [6] A. P. Sheth and J. A. Larson. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3):183–236, September 1990.
- [7] C. Batini, M. Lenzerini, and S. B. Navathe. A Comparative Analysis of Methodologies for Database Schema Integration. *ACM Computing Surveys*, 18(4):323–364, December 1986.
- [8] M. Castellanos, F. Saltor, and M. Garcia-Solaco. A Canonical Model for the Interoperability Among Object-Oriented and Relational Databases. In M. T. Özsu, U. Dayal, and P. Valduriez, editors, *Distributed Object Management*, pages 309–314, San Mateo, CA, 1994. Morgan Kaufmann Publishers.

- [9] E. Radeke, R. Böttger, B. Burkert, Y. Engel, G. Kachel, S. Kolmschlag, and D. Nolte. Efendi: Federated Database System of Cadlab. In M. J. Carey and D. A. Schneider, editors, *Proc. of the 1995 ACM SIGMOD Int. Conf. on Management of Data, San Jose, CA*, volume 24 of *ACM SIGMOD Record*, page 481. ACM Press, June 1995.
- [10] M. Tresch and M. Scholl. Schema Transformation Processors for Federated Objectbases. In S. C. Moon and H. Ikeda, editors, *Proc. of the 3rd Int. Conf. on Database Systems for Advanced Applications (DASFAA'93), Taejeon, Korea, April 1993*, pages 37–46, Singapore, 1993. World Scientific Press.
- [11] I. Schmitt and G. Saake. Integration of Inheritance Trees as Part of View Generation for Database Federations. In B. Thalheim, editor, *Conceptual Modelling — ER'96, Proc. of the 15th Int. Conf., Cottbus, Germany, October 1996*, volume 1157 of *Lecture Notes in Computer Science*, pages 195–210, Berlin, 1996. Springer-Verlag.
- [12] S. Conrad, M. Höding, G. Saake, I. Schmitt, and C. Türker. Schema Integration with Integrity Constraints. In C. Small, P. Douglas, R. Johnson, P. King, and N. Martin, editors, *Advances in Databases, 15th British National Conf. on Databases, BNCOD 15, London, UK, July 1997*, volume 1271 of *Lecture Notes in Computer Science*, pages 200–214, Berlin, 1997. Springer-Verlag.
- [13] K. Schwarz and C. Türker. Investigating Advanced Transaction Models for Federated Database Systems. Technical Report 7, Fakultät für Informatik, Universität Magdeburg, 1997.
- [14] C. Türker and S. Conrad. Towards Maintaining Integrity of Federated Databases. In *Data Management Systems, Proc. of the 3rd Int. Workshop on Information Technology, BIWIT'97, July 2–4, 1997, Biarritz, France*, pages 93–100, Los Alamitos, CA, 1997. IEEE Computer Society Press.
- [15] W. Eßmayr, F. Kastner, S. Preishuber, G. Pernul, and M. A. Tjoa. Access Control for Federated Database Environments - A Taxonomy of Design Choices. In *Proc. Joint IFIP TC 6 and TC 11 Working Conference on Communications and Information Security (IT-Security'95)*, pages 117–132, Graz, Austria, September 1995.
- [16] S. Abiteboul, S. Cluet, and T. Milo. Querying and Updating the File. In *Proc. of the 19th VLDB Conference*, pages 73–84, Dublin, Ireland, August 1993.
- [17] M. Höding. An Approach to the Integration of File Based Systems into Database Federations. In *Proc. of the Int. Workshop on Heterogeneous Information Management*, Prague, Czech Republic, November 1996.
- [18] I. Schmitt and G. Saake. Schema Integration and View Generation by Resolving Intensional and Extensional Overlappings. In K. Yetongnon and S. Hariri, editors, *Proc. of the 9th ISCA Int. Conf. on Parallel and Distributed Computing Systems (PDCS'96), Dijon, France, September 1996*, pages 751–758, Six Forks Road, Raleigh, NC, 1996. International Society for Computers and Their Application.
- [19] Horst-Peter vom Dorp. *Economicalness Analysis for the Introduction of Integrated Information Processing Systems using the Estimation of Benefits Analysis* (In German). PhD thesis, Technische Universität Berlin, Faculty for Economical Sciences, 1974.
- [20] OMG (Object Management Group). *The Common Object Request Broker: Architecture and Specification, OMG Document Number 91.12.1*, December 1991.