Research in Databases and Data-Intensive Applications Computer Science Dept. and FZI, University of Karlsruhe*

Birgitta König-Ries, Peter C. Lockemann
{koenig,lockeman}@ira.uka.de

The future world of computing will be governed by large networks of communicating and interacting persons and machines, geographic mobility, temporary attachment to networks, the disintegration of formerly monolithic organizations and systems into autonomously acting units, the substitution of cooperation regimes for centralized control, and an ever-increasing spectrum of ever more ambitious applications. In such a world the methods, techniques and tools of database technology will play new and more diversified roles, not so much in combination as parts of all-inclusive database systems but rather individually as indispensable ingredients of or desirable enhancements to novel communication, control and application systems. The two information systems groups whose work is presented in this report aim at meeting these new challenges. Our contributions are in the large field of what we call "distributed data-intensive applications".

The group at the Department of Informatics at Karlsruhe University explores new avenues in the technological directions of data modeling, database design techniques, object-oriented active and distributed databases, and in the applied directions of database support of creative engineering, data exploration, and constraint management. Its main application areas where it interacts with engineers are computer-aided manufacturing, building design, and information services in networks. The group at the Forschungszentrum Informatik (FZI), an independent research organization closely associated with the university, interacts with industrial partners, transferring the results of modern database research to them and receiving in return numerous suggestions for research topics that could fill the technical gaps observed in practical applications. This group has developed its main expertise in design frameworks, platforms for distributed systems, object-oriented database systems, and the integration of database, middleware and Internet technologies. The main applications are in reengineering of data-intensive software, environmental information systems, and electronic commerce. Both groups closely collaborate by regularly sharing their ideas, results and experiences.

A more detailed description of our work and a complete bibliography is available at the WWW-addresses given below.

1 Supporting Adaptability in Databases and their Services

U. Herzog, G. Hillebrand, U. Kölsch, B. König-Ries, P. Lockemann, J. Mülle, S. Pulkowski, C. Reck, R. Sturm, M. Wallrath, R. Witte, A. Zachmann

Today's production and service industry is characterized by the breakup of large, centrally controlled and diversified enterprises and institutions into a large number of narrowly focused, highly adaptive units which collaborate on a case-by-case basis whenever customers require problem solutions that encompass expertise from several of them. Informatics is challenged to meet these developments from an information technological perspective, by replacing its formerly monolithic and inflexible information systems by systems that are smaller and more specialized and that can easily be adapted to changing conditions in the real world they serve. Our own work takes up these challenges in the realm of database technology.

Database evolution. The rules governing the processes in the mini-world are first and foremost reflected in database schemas. If they evolve over time, database schemas must be adjusted to guarantee continued schema consistency. Second and much more expensive, given a database that satisfied the old schema, this database may have to be adjusted to regain schema/database consistency. Since usually there is a considerable degree of latitude for such adjustments, we provide the database designer with a declarative language in which he can describe the necessary changes to the object base. Because such programs may themselves be incor-

*Information Systems Group, Institute for Program Structures and Data Organization, University of Karlsruhe, P. O. Box 6980, D-76128 Karlsruhe, Germany. http://wwwipd.ira.uka.de.

Database Research Group, Forschungszentrum Informatik, Haid-und-Neu-Strasse 10-14, D-76131 Karlsruhe, Germany. http://www.fzi.de/divisions/dbs/dbs.html.

rect the designer would first have to run consistency tests against the old database before committing the results. Our approach replaces such an expensive brute-force execution by an advance estimation of the schema/database consistency via simulating the declared operations on an abstraction of the object base's state. In case of consistency violations, the database designer may decide whether and how to alter the schema, change the programs, or leave both unchanged but complement the programs by compensating operations.

Controlling the scope of design decisions. A design process can be characterized as a process of stepwise reduction of a design space by design rules and decisions. These may take the form of consistency constraints. Since design is a collaborative activity, such constraints encompass a design space shared by many participants. In connection with the architectural design of buildings, we employ the technique of stepwise addition or reformulation of constraints for reflecting legal and technical requirements and decisions. Each design decision and each constraint as well as the current activity of a designer is circumscribed by a work space. By introducing the concept of space collision we employ a uniform mechanism to deal with actions of the individual designer such as querying, updates, or selective backtracking of decisions as well as with the interaction between designers. By associating histories with all design decisions, including constraints, we allow queries, interactions and backtracking both in space and time.

Superimposed on the collision mechanism are a workflow management component, which allows to impose general strategies on the collaboration between the numerous design engineers, a work space model called A4 that organizes and integrates all design information in a multi-dimensional space, and a graphical interface developed by our partners from the school of architecture, which supports the individual engineer. Underlying the space model and the collision mechanism is an object-oriented database management system and a facility for handling ECA rules.

The work took place within the interdisciplinary project "Database supported coordination and integration of experts in architectural design".

[1] R. Sturm, J. Mülle, P. Lockemann. Collision of Constrained Work Spaces: A Uniform Concept for Design Interactions. In: *Proc. 2. Intl. Conf. on Cooperative Information Systems (CoopIS'97)*, Charleston, USA, 1997.

Consistency checking. As discussed above, consistency constraints play an important role when one wishes to deal with adaptive behavior in a controlled fashion. Consistency constraints could also be a major cornerstone for the reliability of information systems. Unfortunately, checking and enforcing consistency constraints carries a high computational penalty if the databases are large or complexly structured or the constraints are numerous. Consequently, we directed our efforts towards methods and techniques for improving the performance of consistency checking. Consistency constraints, declaratively specified in a first-order logic language, are translated into an extended relational algebra. The algebra had to be extended by new operators in order to make the complexity of constraints which arise from real applications more manageable, and to utilize the potential of the subsequent optimization techniques. Because many constraints must be checked simultaneously, considerable savings appear possible in a second step that identifies shared subexpressions in order to evaluate each just once. In a third step we generate parallel constraint execution plans for a parallel constraint execution engine which supports a dataflow-driven, set-oriented execution. The work was completed during 1996 with experiments with the engine, which demonstrated a clear performance gain up to one order of magnitude.

Uncertain information in design environments. A distinguishing mark of design applications is their need for data models with uncertainty capabilities. Typically, a design process starts with vague information that undergoes a series of refinements until it can be resolved to precision. Current data models are poorly equipped for capturing uncertain information, the only standard mechanism available being NULL-values.

We propose an enriched object-oriented data model based on the theory of fuzzy sets. This model has been successfully incorporated into an object-oriented database management system and employed in an architectural application. We now believe it is feasible to extend this model to other kinds of application, such as knowledge discovery, reengineering of software systems, and computer linguistics.

The big challenge, which seems to have been neglected in research on fuzzy databases so far, is the automatic processing and modification of uncertain information. We tackle this problem by defining new operations for fuzzy sets based on non-monotonous logic and belief revision.

Flexible service architectures. In large distributed systems with numerous, autonomous nodes the response to a user seeking a solution to a given problem must often be dynamically composed from the services available from several nodes. Database expertise can contribute to the technical infrastructure that provides the needed flexibility in two ways: language expertise, which on the basis of set-oriented

declarative languages allows to translate requests into execution plans, and modeling expertise, which can contribute to dealing with semantic heterogeneity.

Answering a user request usually involves more than one software component in the distributed system. The components themselves often access distributed and heterogeneous information sources in order to provide the services they offer. Generally, the user expects the system to provide transparent access to the functionality and information provided by the underlying software components and information sources. In the MAGIC project, we currently investigate how plans can be constructed that combine relevant service providers and regulate their cooperation in order to answer a complex user query. The approach is based on a modification of the concept of the universal relation to which a planner is applied.

The second focus in MAGIC is on overcoming semantic heterogeneity via mediators. In extension to other approaches which emphasize the (semi-) automatic generation of mediators from given specifications we focus on the (semi-)automatic generation of specifications. Only the combination of both will make the mediator approach truly scalable.

A third focus in MAGIC is on the automatic generation of wrappers providing both the technical integration of information sources (databases and html-sources) and a mapping of meta-information about the sources to a common model.

[1] C. Reck, B. König-Ries. An Architecture for Transparent Access to Heterogeneous Information Sources. In: *Proc. 1. Intl. Workshop on Cooperative Information Agents*, Kiel, Germany, 1997.

Re-engineering Information Systems The old monolithic and gigantic information systems provide essential information processing services, and they rely on huge information repositories that represent corporate memories of decades and thus are of immeasurable value. However, if ones wishes to adapt them to the new and more flexible business processes or incorporate them into the global network, they must be reengineered into collections of smaller and more focussed subsystems that can enter into configurations tailored to specific needs.

The notion of focussed subsystem matches neatly with the notion of object. Consequently, we base our methodology for reengineering legacy system (called R-objectification), that has been developped as part of the project DARE on object-oriented modelling techniques (presently OMT) and enhance it with a re-engineering process model which itself is integrated in a software lifecycle model. In accordance with OMT, which takes an equal and integral view of object structure, object functionality and dynamics within and between objects, we regard an information system as three cooperating components – the data, the program and the process system.

In the reverse engineering phase, the internal structures, relationships and reciprocal dependencies of the subsystems are analyzed in order to obtain an object-oriented specification of the legacy system. Based upon this specification, the system can be restructured and (partially) reimplemented. During this phase, an object-oriented data model that was derived from the specification of the actual business processes serves as a benchmark. By following the spiral model the entire re-engineering process becomes a cyclic process where the new system evolves from many smaller re-engineering steps that pursue intermediate goals. A central part of the method is the re-engineering environment which we place at the re-engineer's disposal. This environment supports the integration of a variety of analysis, rectructuring and visualization tools. It includes a repository which contains a meta model of the reengineering process and follows an object-oriented data model as well.

[1] U. Kölsch, J. Laschewski. Objectifying Legacy Application Systems Using a Specification Language Framework. In: *Proc. ICSE-19 Workshop on Migration Strategies for Legacy Systems*, Boston, USA, 1997.

[2] U. Kölsch, M. Wallrath. A Process Model for Controlling and Performing Reengineering Tasks. In: *Proc. 1. Euromicro Conf. on Software Maintenance and Reengineering*, Berlin, Germany, 1997.

2 Technologies for Distributed Data-Intensive Applications

P. Biegler, G. Hillebrand, A. Koschel, R. Kramer, P. Lockemann, G. Lukacz, R. Nikolai, D. Posselt, C. Rolker, D. Theobald, H.-D. Walter, G. von Bültzingsloewen

One of the metaphors for the information services in a modern, large and widely distributed network is the network as a single global database. Such a view challenges database technology to reflect on how well its traditional techniques suit the new environment and where it has to come up with new solutions. One sure answer is that database technology by itself will not suffice but must be tightly interconnected with data communications technology. Our own work has been directed towards this interconnection. We pursue it in close collaboration with applications, and we base much of our work on so-called middleware, under close observation of evolving worldwide standards.

An overview of our work in this area is given in [1].

[1] P. Lockemann, U. Kölsch,A. Koschel, R. Kramer,R. Nikolai,M. Wallrath,H.-D. Walter The Network as a Global Database: Challenges of Interoperability, Proactivity, Interactiveness, Legacy In: *Proc. of the 23rd VLDB Conf.*

Athens, Greece, 1997.

Databases in the World Wide Web. The tools and techniques of the WWW are an attractive means to make databases available on the Internet and on intranets in a user-friendly way. This is true even where the databases are heterogeneous and geographically dispersed, differ in contents, and vary in data models. Where WWW techniques can help is to achieve visualization uniformity, i.e., to give the user the visual appearance of a homogeneous database.

The classical approach to access databases via the Web is based on the Hypertext Transfer Protocol (HTTP) and the Common Gateway Interface (CGI). Once an application requires a higher degree of interactivity among the database systems, the limitations of this approach become apparent. These have to do with the stateless HTTP which prohibits the use of transactional techniques. Therefore, we augmented the classical approach by techniques that are based on the programming language Java and the standard of Java Database Connectivity (JDBC). A first version of a JDBC compliant driver for both read and write access to a relational database system has been developed. This driver has been successfully used in several national and international projects.

[1] A. Koschel, R. Kramer, R. Nikolai, G. Lukacs, T. Heinemeier. Data and Metadata Management in Distributed Environmental Information Systems. In: *2. Intl. Symp. on Environmental Software Systems (ISESS'97)*, Delta Whistler, Canada, 1997.

[2] R. Kramer. Databases on the Web: Technologies for Federation Architectures and Case Studies. In: *Proc. ACM SIGMOD Intl. Conf. on Management of Data*, Tucson, USA, 1997.

Federation architectures. Today's distributed information systems are a conglomerate of already existing resources that were originally developed with local concerns in mind. All the components must, therefore, interact across a network of heterogeneous platforms and services. Interoperability refers to the mechanisms that allow such an interaction. Interoperability concerns, e.g., data formats, terminology use, data models, communication protocols, and the semantics of data.

To cope with these interoperability issues we have developed a federation architecture which we included as part of our projects on Europe-wide public environmental information systems. This architecture is based on open Internet technologies (WWW, CORBA, Java) together with techniques which allow the access to heterogeneous data sources via a CORBA-based middleware layer. The layer has been enhanced by customized middleware clients which allow the integration of information from different data sources. On the semantic level, interoperability is achieved by terminological, multilingual thesauri of which several may be connected to the system and used in an integrated fashion, and by specific mediators in the form of library functions.

[1] A. Koschel, R. Kramer, D. Theobald, G. v. Bültzingsloewen. Evaluation and Application of CORBA Implementations. In: *ECOOP'96 Workshop on Putting Distributed Objects to Work, 10. Europ. Conf. on Object-Oriented Programming*, Linz, Austria, 1996.

[2] A. Koschel, R. Kramer, R. Nikolai, W. Hagg, J. Wiesel. A Federation Architecture for an Environmental Information System Incorporating GIS, the World-Wide Web, and CORBA. In: *3. Intl. Conf./Workshop on Integrating GIS and Environmental Modeling*, Santa Fe, USA, January 1996.

[3] R. Kramer, A. Koschel. Base Technologies for Distributed Environmental Information Systems. In: *2. Intl. Symp. on Environmental Software Systems (ISESS'1997)*, Delta Whistler, Canada, 1997.

[4] R. Kramer, H. Spandl. Making the Environmental Data Catalogue (UDK) and other Databases Available on the World-Wide Web. In: *Workshop on Internet for Environmental Communication*, Vienna, Austria, 1996.

Active mechanisms in distributed environments. In distributed information systems passivity is not always sufficient. There are many situations, such as pollution monitoring, where the information system should become aware on its own of what is happening around it, and react properly and spontaneously, e.g., by notifying a particular user. Active database management systems (ADBMS) are a response by the database community to these needs.

Since we solve interoperability in our federation architectures by means of middleware, it seems only natural to use such middleware also as a base mechanism for recognizing events and transporting them from the source to a destination. So far, CORBA provides only limited support in the form of a basic event transmission service across a distributed environment. Our research aims at an enhancement of CORBA by higher-level ECA rule services. In order to dissociate the services from any particular information source (e.g., database system), we started to develop a configurable, modular and distributed architecture for rule processing.

[1] G. v. Bültzingsloewen, A. Koschel, R. Kramer. Active Information Delivery in a CORBA-based Distributed Information System. In: *Proc. 1. IFCIS Int. Conf. on Cooperative Information Systems (CoopIS'96)*, Brussels, Belgium, June 1996.

[2] G. v. Bültzingsloewen, A. Koschel, R. Kramer. Accept Heterogeneity: An Event Monitoring Service for CORBA-based Heterogeneous Information Systems (Poster). In: *Proc. 2. Intl. Conf. on Cooperative Information Systems (CoopIS'97)*, Charleston, USA, 1997.

[3] A. Koschel. CORBA-based Active Functionality—ECA Rules, Application Classification, Configurability. FZI-Report 3/97, Forschungszentrum Informatik, Karlsruhe, Germany, 1997.

[4] A. Koschel, R. Kramer, G. v. Bültzingsloewen, T. Bleibel,

P. Krumlinde, S. Schmuck, C. Weinand. Configurable Active Functionality for CORBA. In: *ECOOP'97 Workshop on CORBA: Implementation, Use and Evaluation.* Jyväskulä, Finnland, 1997.

Metadata. A non-trivial problem for users of distributed informations systems is to identify and locate relevant data, and to obtain enough background information to be able to judge the suitability and trustworthiness of the sources. Metadata, i.e., data about data, are a means to deal with such problems. As part of the environmental information system described earlier, the environmental data catalogue (Umweltdatenkatalog, UDK) plays the role of metadata repository. The present one is used in German speaking countries to store and manage all environmental metadata. We embedded the catalogue in our WWW environment. Our experiences with the system allowed us an extension to a prototype system (WebCDS) for the transnational European level. Since in scientific evaluations of measurement data the results are produced in a stepwise, probing or iterative manner, the catalogs include information about the derivation process. These metadata are formulated and managed as extended predicate transition nets.

[1] R. Kramer, R. Nikolai, C. Rolker. World-Wide Web Access to CDS: Software Design and Demonstration of First Prototype. In: *Proc. 3. Workshop on Catalogue of Data Sources (CDS) and Thesaurus*, Copenhagen, Denmark, 1996.

[2] R. Kramer, R. Nikolai, C. Habeck. Thesaurus Federations: Loosely Integrated Thesauri for Document Retrieval in Networks based on Internet Technologies. *Journal on Digital Library*, 1997.

[3] R. Kramer, R. Nikolai, A. Koschel, C. Rolker, P. Lockemann, A. Keitel, R. Legat, K. Zirm. WWW-UDK: A Web-based Environmental Metainformation System. *ACM SIGMOD Record*, 26(1), 1997.

Cooperation in distributed object bases. Cooperative information systems such as groupware systems or workflow systems will play a key role in future electronic information processing. A common approach to designing cooperative information systems is the organization of such systems as sets of cooperating components (i.e., objects). One of the main problems with this approach is the guarantee for meeting application-specific cooperation rules because today's object models lack suitable abstractions to express cooperative behavior.

We have developed alliances as a special modeling construct that groups a set of objects with respect to their cooperation aspects. Alliances ensure inter-object constraints, and guard conditions concerning the temporal order of messages between objects. In case of errors, compensating actions will be initiated by alliances. Additionally, alliances control the distribution of objects. Alliances can be regarded as a capsule for an interaction protocol—therefore, the formalism we use to describe alliances is heavily influenced by rule-based protocol specification techniques that are widely used in the field of data communications.

[1] P. Lockemann, H.-D. Walter. Object-Oriented Protocol Hierarchies for Distributed Workflow Systems. *Theory and Practice of Object Systems* 1 (1995), pp. 281–300.

[2] O. Ciupke, D. Kottmann, H.-D. Walter. Object Migration in Non-Monolithic Distributed Applications. In: *16. Intl. Conf. on Distributed Computing Systems*, Hong Kong, 1996.

3 Value-Added Services for Databases

C. Breitner, P. Lockemann, J. Mülle, J. Schlösser, R. Schmidt, B. Schmitt, R. Sturm, S. Zwißler

Users employ information services when in the course of pursuing tasks they face a problem which is beyond their competence, and they delegate the problem in the hope that it will be solved without further involvement on their part. Value-added services are for us those services which deliver entire solutions that are composed from the technologies described in the previous sections.

Knowledge discovery in databases (KDD). The goal of KDD is to extract meaningful information from the huge and rapidly growing volumes of data. Knowledge discovery is not a single-step application of a magic method, but rather a long lasting, iterative and interactive process. During the process the user has to model a multitude of data derivation processes, execute them and interpret the results in order to compose new derivation processes. In the scope of the project "Citrus" we develop a powerful and user-friendly KDD-system in cooperation with industrial partners. Central to the project is the view of KDD as a process. From our point of view two aspects are of particular interest: Process support by information modeling and system performance.

Over the course of the KDD-process there is an evolution of a large number of data sets, derivation processes and knowledge results. Due to the iterative and interactive nature of the process the subsequent use of all this information is difficult to predict. In particular, much of it may be reused. Consequently, modeling data and processes in a uniform fashion that is understood by all tools employed within the process is an essential prerequisite. We have designed an information model which allows to represent the data derivation processes as gradually abstracting data-queries, and to reflect the queries in the models of the data and results. In addition, a complete process and result history can be established by using the information model.

The second issue, performance, is to a considerable extent a matter of computational complexity of the statistical and learning algorithms. However, most of them cannot deal with peripherally stored data. Consequently, performance in KDD processes also becomes more and more a database related issue. We develop an execution strategy which takes the runtime characteristics of the KDD-process—huge data sets and frequent iterations—into account. The basic idea is to ship the data preparation operations as far as possible into the relational database engine, in order to avoid expensive loading of the data into the data mining system. We also evaluate the process history in order to gain clues for a specific and automatic materialization of intermediate results and their reuse to avoid the recomputation of results when iterations occur. In a first prototypical implementation of the execution strategy we have been able to show significant gains in efficiency.

[1] R. Wirth, C. Shearer, U. Grimmer, T. Reinartz, J. Schlösser, C. Breitner, R. Engels, G. Lindner. Towards Process-Orientied Tool Support for KDD. In: *Proc. 1. Europ. Symp. on Principles of Data Mining and Knowledge Discovery (PKDD'97)*, Trondheim, Norway, June 1997.

[2] C. Breitner, J. Schlösser, R. Wirth. Process-Based Database Support for the Early Indicator Method. In: *Proc. 3. Intl. Conf. on Knowledge Discovery and Data Mining (KDD'97)*, Newport Beach, USA, August 1997.

Workflow support through component-based frameworks. Workflows can be viewed as application-oriented specifications of cooperation protocols. Therefore, our work on workflows neatly supplements our efforts on data-intensive distributed applications, notably data modeling and cooperation in distributed systems. In other words, we consider workflow management systems (WfMS) as an important extension to database management systems, where they provide flexible support of business processes. However, the use of WfMS is still constrained by their weak support for heterogeneous environments and low extensibility. Therefore, we use component-based frameworks to develop highly customizable and extensible WfMS architectures. This allows us to tailor WfMS to individual needs by integrating independently developed components. The design of workflow applications can be seen as a new paradigm for systems development. Under this viewpoint we develop transformations of business process models to workflow representations in component-based WfMS architectures, even those in heterogeneous environments.

[1] O. Ciupke, R. Schmidt. Components as Context-Independent Software Units. In: *ECOOP'96 Workshop on Component-oriented Programming (WCOP'96), 10. Europ. Conf. on Object-Oriented Programming*, Linz, Austria, 1996.

Electronic commerce. Electronic Commerce means the marketing of goods and services via electronic media. Most of today's emphasis still is on the development of user interfaces, with WWW and Java providing the major technological input. The focus of our group in this area is the use of the Internet for business-to-business commerce, a clean separation of information storage and information presentation, and the provision of complete solutions to a user by combining offers from several service providers. Hence, from our point of view electronic commerce requires the system integration of a number of technologies: database technology, cooperation protocols, interoperability. Due to the commercial aspects and the legal implications, high demands are placed on correctness, security, availability and performance. These demands are interdependent and require a global system view. In 1996 we began to build up a competence center in the field of Electronic Commerce with the aim of initiating electronic commerce activities within the region of Karlsruhe and coordinating all interested parties, technology and content providers alike.

Trading services for digital libraries. Traders are intermediaries that, given the needs of a shopper, identify and locate servers and provide information on them. They are often referred to as the "yellow pages" of a network.

In the light of digitized information, university libraries will have to become information brokers rather than archivers and lenders of printed documents. As such they will be embedded in a network of other libraries, documentation centers, traditional information brokers, publishing houses and individual information producers. Assuming that a university library finds its own place in such a competitive environment by focusing on a special clientele, namely the members of the university, we examine what range of services it should offer, whether these should be specialized towards particular member groups, and how these services interface with the wealth of information provision services off campus.

Our aim is to provide trading services adapted to the needs of university members. The trader manages the service offers of our campus libraries and in addition is able to cooperate with other external traders that handle services from different institutions. Since competition is a matter of pricing, we also plan to evaluate, in cooperation with the economics department, different cost models.