Engineering Federated Information Systems
Report of EFIS '99 Workshop

http://infolab.kub.nl/act/conventions/efis99/

http://wwwdb.informatik.uni-rostock.de/efis99/

S. Conrad* W. Hasselbring† U. Hohenstein‡ R.-D. Kutsche§ M. Roantree¶
G. Saake* F. Salter‖

Abstract

After the successful first International Workshop on Engineering Federated Database Systems (EFDBS'97) in Barcelona in June 1997 [CEH+97], the goal of this second workshop was to bring together researchers and practitioners interested in various issues in the development of federated information systems, whereby the scope has been extended to cover database and non-database information sources (the change from EFDBS to EFIS reflects this). This report provides details of the workshop content and the conclusions reached in the final discussion.

1 Introduction

Federated Information Systems (FIS) are an emerging research area, addressing the goals of transparent interoperation of autonomous, heterogeneous information sources. As shown in many examples of information systems in application areas such as business, finance, manufacturing, environmental science, and medicine, there is not only an increasing need to make information available via some kind of network (enterprise network or even the WWW) for "global" applications, but also for local applications to use semantically related information. The EFIS'99 workshop tackles problems related to *engineering* FIS.

The workshop was organized by Andreas Heuer, Joachim Kröger, and Holger Meyer (University of Rostock, Germany) and took place in the Baltic town of Kühlungsborn from May 5 to May 7, 1999.

2 Federated Information Systems

We regard FIS as information systems that are integrated to some extent. The integration may be done by means of middleware tools, database management systems, etc. Obviously, we do not consider those tools themselves as being FIS. So, what is a FIS? In contrast to the classical notions of 'federated database systems' and 'multi-database systems', the term 'federated information system' intends to include not only structured information sources but also semi-structured and even unstructured information. These inclusions often have implications on the three dimensions of autonomy, heterogeneity, and distribution. A FIS, in general, may allow for a rather high degree of freedom in each of these dimensions and, therefore, require new techniques for coping with the resulting problems [BKLW99]. On the other hand, it is obvious that only severe restrictions in particular dimensions (e.g., mediator-based FIS with read-only query access) can help in the actual design of practicable federation solutions, which was one of the major issues for discussion throughout the workshop.

3 Workshop Contents

Workshop topics included the research issues identified in [CEH+97], particularly:

- Analysis, design and implementation of FIS
- CASE tool support for the development of FIS
- Software architectures for FIS
- Metadata management in FIS
- Managing heterogeneous information sources including semi-structured data
- Experience with middleware platforms for building FIS
- Technological support for the integration of legacy systems
- Influence of the application domain on the Design of FIS
- Evaluation of FIS approaches

*Department of Computer Science, University of Magdeburg, Germany
†Infolab, Tilburg University, Netherlands
‡Siemens AG, Germany
§Department of Computer Science, TU Berlin, Germany
¶School of Computer Applications, DCU, Dublin, Ireland
‖Universitat Politecnica de Catalunya, Catalunya

In addition to the seven long and seven short papers, which were selected for presentation at the workshop by the program committee, three invited talks were presented:

- Harald Gröger (IBM Germany):
  "IBM DB2 DataJoiner: A commercially available federated database system"
- Felix Saltor (UPC, Catalonia):
  "On Semantic Issues in Federated Information Systems"
- Jane Grimson (Trinity College, Dublin):
  "Engineering Federated Electronic Healthcare Records"

4 Workshop Observations

The following results are in the form of a number of observations on the emerging trends and consensus. These comments are not meant to be a definitive perspective on FIS but merely represent the opinions of those present at the workshop and authoring this report.

- Some commercially available *tools* for building FIS are available (examples are IBM DataJoiner, Gateways and other middleware tools). We do not consider those tools themselves as being FIS (see our definition of FIS above).
- In real live settings, FIS are built deploying those tools mentioned in the previous item, see e.g. [ROP+99].
- As observed in [CEH+97], the 5-level-schema-architecture of [SL90] is generally accepted as the basic structure of federated database systems (not FIS in general). However, in real systems, not all levels are actually used. This conforms with the reference schema architecture [SL90], whereby some levels contain no schemas. Using DataJoiner only the federated schema is defined by means of nicknames with mappings to local schemas. External schemas are views to those nicknames. Component and export schemas are not constructed.
- Schema Integration is a difficult process, which involves detecting and solving semantic heterogeneities among 1) structures, 2) integrity constraints, and 3) behavior of the component databases. Research has focused on 1) in the past, on 2) right now, and 3) remains to be solved. Schema Integration benefits from as much semantics as possible, and therefore it requires the discovery of semantics (semantic enrichment) in the component databases. Furthermore, the integration process has to be supported by tools which address several aspects of the process, e.g., integration of class hierarchies, constraints etc. [HSC+99].
- Object models, mainly extensions to ODMG, are often used as canonical data models in research prototypes for federated database systems [HE99, SR99].

  However, it appears that ODMG is not well-suited as a canonical model. Sometimes it is hard to reflect the ODL and OML semantics of specific concepts of commercial ODBMSs in ODMG. Even if it is possible to reflect concepts such as clustering or versions in ODMG for particular ODMBS, there is a need for a common simulation of *all* the advanced concepts of *all* the ODBMS, i.e., some kind of abstraction of those concepts. Otherwise is seems to be impossible to use those features for integration.

  Conversely, commercial integration tools are usually based on the relational model, which implies limited expressive capabilities for the federated schema.
- An obvious observation is that metadata and ontologies are highly relevant for interoperability in FIS.

  The introduction of a metamodel in ODMG 2.0 should assist FIS designers who use the ODMG model as a canonical model. However, this is offset by the fact that the ODMG schema access interface has yet to be implemented by any of the ODMG database vendors.

  Domain-specific standards and ontologies are useful for enabling interoperability [GT99, Has99]. However, it is hard to find those domain-specific standards that are detailed enough to allow for real interoperability.
- The need for explicit federation definition and query/manipulation languages was observed. Such specification languages may enable, for instance, the automatic generation of homogenizing adapters (wrappers) [HE99].
- XML is increasingly used as an exchange format for both data *and* metadata.

  XML may also be useful for the integration of semi-structured data that is stored in files. This is particularly needed if FIS technology is to be relevant to WWW applications.
- Active mechanisms are needed for integrity control [SR99, KK99].
- FIS transaction management requires more than the two-phase commit offered by traditional transaction monitors, and more than the CORBA object transaction service OTS addresses [SW99]. The borderline between the classical database view on

a whole and the understanding of FIS as part of transactional workflow management systems becomes somewhat narrow in the world of distributed objects.

- The application domain has an essential impact on the choice of architecture for FIS. Particularly, the requirements of the application domain determine the (database) functionality which has to be provided on a federation layer. For instance, certain global applications with read-only access to the locally stored data might not need global transaction management. Methodologies for building FIS which cover concerns of the application domains are not yet available. Domain-specific software architectures may provide a first step into that direction [Has99].

  Applications of information mining and fusion [SS99], healthcare, or digital libraries could benefit from FIS techniques. However, these challenging application domains often define new requirements which have to be addressed in current and future research, e.g., integration of semi-structured data, efficient query processing and the quality of data.

- Security and confidentiality were not addressed in the presentations at the workshop, but the participants felt that these are important issues to be solved.

5 Conclusions

The workshop proceedings contain all of the long and short papers accepted in the workshop together with one of the invited presentations [CHS99]. It was generally agreed by all participants that the workshop was a useful forum for the discussion on current research in engineering federated architectures and information systems. For this reason, it was decided to run the workshop on a yearly basis. Plans for next year's workshop have already begun and Dublin has been agreed as a venue for the EFIS '2000 event. It is intended that the third workshop should look to attract papers which target the issues raised at the EFIS '1999 workshop. Further information on the EFIS '2000 event shall be available via the EFIS '1999 home page: `http://infolab.kub.nl/act/conventions/efis99/`

References

[BKLW99] S. Busse, R.-D. Kutsche, U. Leser, and H. Weber. Federated information systems: Concepts, terminology and architectures. Forschungsbericht Nr. 99-9, Fachbereich Informatik, TU Berlin, April 1999.

[CEH+97] S. Conrad, B. Eaglestone, W. Hasselbring, M. Roantree, F. Saltor, M. Schönhoff, M. Strässler, and M. Vermeer. Research Issues in Federated Database Systems (Report of EFDBS '97 Workshop). *SIGMOD Record*, 26(4):54–56, December 1997.

[CHS99] S. Conrad, W. Hasselbring, and G. Saake, editors. *Engineering Federated Information Systems (Proc. EFIS'99)*. Infix-Verlag, Sankt Augustin (ISBN 3-89601-013-1), 1999.

[GT99] M. Gaedke and K. Turowski. Generic web-based federation of business application systems for e-commerce applications. In Conrad et al. [CHS99], pages 25–42.

[Has99] W. Hasselbring. Top-down vs. bottom-up engineering of federated information systems. In Conrad et al. [CHS99], pages 131–138.

[HE99] U. Hohenstein and A. Ebert. An integrated toolkit for building database federations. In Conrad et al. [CHS99], pages 43–60.

[HSC+99] M. Höding, K. Schwarz, S. Conrad, G. Saake, S. Balko, A. Diekmann, E. Hildebrandt, K.-U. Sattler, I. Schmitt, and C. Türker. $SIGMA_{FDB}$: Overview of the Magdeburg Approach to Database Federations. In Conrad et al. [CHS99], pages 139–146.

[KK99] A. Koschel and R. Kramer. Applying configurable event-driven services in heterogeneous distributed information systems. In Conrad et al. [CHS99], pages 147–157.

[ROP+99] F.d.F. Rezende, G.d.S. Oliveira, R.C.G. Pereira, U. Hermsen, and R. Keller. A unified database interface for multiple heterogeneous databases. In Conrad et al. [CHS99], pages 61–78.

[SL90] A. Sheth and J. Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys*, 22(3):183–236, 1990.

[SR99] A. Skehill and M. Roantree. Constructing multidatabase collections using an extended ODMG object model. In Conrad et al. [CHS99], pages 95–106.

[SS99] K.-U. Sattler and G. Saake. Supporting information fusion with federated database technologies. In Conrad et al. [CHS99], pages 179–184.

[SW99] R. Schenkel and G. Weikum. Experiences with building federated transaction manager based on CORBA OTS. In Conrad et al. [CHS99], pages 79–94.