

**Proceedings of the 1997
Knowledge-based Systems Interoperability Workshop**

**3 & 4 November 1997
Gaithersburg, Maryland**

Robert H. Allen, Ram D. Sriram (editors)

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards and Technology
Manufacturing Engineering Laboratory
Manufacturing Systems Integration Division
Engineering Design Technologies Group
Gaithersburg, MD 20899

**Proceedings of the 1997
Knowledge-based Systems Interoperability Workshop**

**3 & 4 November 1997
Gaithersburg, Maryland**

Robert H. Allen, Ram D. Sriram (editors)

U.S. DEPARTMENT OF COMMERCE

William M. Daley, *Secretary*

Technology Administration

Gary Bachula, *Acting Under Secretary for Technology*

National Institute of Standards and Technology

Raymond Kammer, *Director*

January 1998

Disclaimer

These proceedings are a summary of the NIST-sponsored workshop on Knowledge-based Systems Interoperability: Standards and Implementation Issues, which was held on 3 and 4 November 1997. Because participants included software users and representatives from commercial vendors, certain products are identified in this report to present specific views and to facilitate understanding of concepts and implementations. The National Institute of Standards and Technology does not judge, recommend or endorse these products. The opinions expressed in this report are those of the workshop participants and not necessarily those of NIST or its employees.

Being published by the United States Government, this report is not subject to copyright.

Table of Contents

Table of Contents	<i>iii</i>
Introduction	1
Workshop Background and Goals	1
Workshop Results	2
References	3
Workshop Agenda	4
Appendix A - List of Participants	6
Appendix B - Presentation Abstracts	8
Appendix C - Presentation Slides	14
Issues in Deploying Knowledge Based Automation and Optimization Systems in a Production Environment S. Tong	
Appendix D- Presentation Slides	27
The ICAD System: A Generative KB Technology P. Katraganna	
Appendix E- Presentation Slides	53
Intelligent Systems Using Knowledge-based Engineering A. Chemaly	
Appendix F- Presentation Slides	70
Rule-Based Interoperability of Heterogeneous Systems S. Su	
Appendix G - Presentation Slides	80
Configurator Synchronization B. Ambler	
Appendix H - Presentation Slides	85
OKBC: A Programming Foundation for Knowledge-based Interoperability V. Chaudhri	
Appendix I - Presentation Slides	95
Knowledge Source Awareness Models for KB Systems R. Reddy	

Introduction

On 3 and 4 November 1997, the National Institute of Standards and Technology (NIST) sponsored¹ an information-gathering workshop that focused on "Knowledge-based Systems Interoperability." Held in Gaithersburg, MD on the NIST campus, and in response to the growing need for integrating knowledge in distributed computing environments, the workshop addressed the general issue of interoperability among knowledge-based systems² especially in engineering design and manufacture. The workshop, which had more than thirty participants, included seven presentations from developers, vendors and users, group discussions on knowledge-based system interoperability -- its present capabilities and some of its main drawbacks -- and a general session to target specific research and development, and end-user needs. This report documents the workshop background and goals, its participants and agenda, the speakers' abstracts and slides, and provides a summary of the workshop results.

Workshop Background & Goals

The purpose of this workshop was to bring together knowledge-based system (KBS) developers, vendors and users from different engineering disciplines to discuss matters of common interest concerning software interoperability. Functional interoperability is fundamental to the success of complex engineering processes such as collaborative design. Although much effort has been put forth in standardizing geometric product data exchange with the development of the international STandard for the Exchange of Product model data, STEP, ISO 10303 [ISO94], such standards do not yet address the exchange of parametric data such as design rationale, functional specification and design intent. To achieve functional interoperability, computer-aided engineering (CAE) applications in general, and KBS in particular, need to be implemented in such a way that the exchange of data and knowledge can occur without loss of information, tolerance or robustness. How to bring about this interoperation is precisely the reason for this workshop.

The workshop mission was to provide an open forum for KBS vendors, engineers and manufacturers, to discuss the state-of-the-art, identify gaps in current technology, and to begin proposing solutions to close those gaps.

Specific workshop goals include the following:

- to provide an overview of the state-of-the-art in KBS interoperability issues in industry, government and academia,
- to present industry case studies on current practices in KBS interoperability,
- to draw roadmaps that will aid in research and development in KBS interoperability, especially in collaborative engineering projects, and
- to identify interoperability standards and technology issues.

The workshop was organized as a series of presentations from speakers representing KBS developers, KBS researchers, and engineers who use KB and CAE systems in their design and manufacturing activities (two developers, three researchers and two engineers). NIST personnel provided additional input on the state of comparable standards and government activity. Following the morning of presentations, workshop organizers split the participants into two subgroups. Each subgroup brainstormed on one of these two themes:

I- State of the Art on KBS Interoperability

¹ Specifically, this workshop was sponsored by the Engineering Design Technologies (EDT) Group, a part of the Manufacturing Systems Integration Division (MSID), under the auspices of the Defense Advanced Research Projects Agency's (DARPA's) Rapid Design Exploration and Optimization (RaDEO) program.

² A KB system, also known as an expert system, is software that has some knowledge or expertise about a specific, narrow domain, and is implemented such that the KB and the control architecture are separated. KB systems have capabilities that often include inferential processing (as opposed to algorithmic processing), explaining rationale to users and generating non-unique results [Mah87].

II- Barriers and Requirements for KBS Interoperability

The subgroups reconvened to discuss the issues raised, and report on each subgroup's findings to the entire workshop. The second day was used for a general discussion, refinement of our findings, and for the group to agree on a list of action items to be taken.

Workshop Results

Of the more than a dozen issues identified by the groups (and listed below), two main themes emerged:

1. Interoperability among KB and CAE systems is a major bottleneck today.
2. Current standards do not address many of the interoperability issues associated with KBS.

Within these main themes, five concepts emerged as priority issues. These are:

Characterization	There is strong need to characterize - perhaps even standardize - the capabilities, behavior and underlying philosophy of KB systems.
Usability	Engineers and manufacturers who use KB and CAx systems must not be unduly burdened with interoperability issues.
Vocabulary	For design and manufacturing applications, a core set of primitives (such as artifact, design plan, goal, form, function and behavior) need to be understood and represented in a standardized way so that meaningful exchange of such knowledge can be achieved.
Collaboration	The commercial, academic and governmental communities must collaborate to address the interoperability issues in a most meaningful way.
Cost	The cost of KB systems and their interoperability must be manageable for midsize companies.

Participants also identified 14 issues as being important in KBS interoperability. These are listed below:

1. Knowledge representation (KR) is the critical element for interoperability because if different KR schemes need to interact, there must be some commonality among representations. One possible solution is to link different KR schemes by using the Knowledge Interchange Format, KIF [Gen92], with a formal explicit specification of a conceptualization, often referred to as a frame ontology [McG93].
2. Mediation is important for interoperability because it places context on a specific knowledge base, otherwise known as semantic heterogeneity.
3. Problem solving cooperation is necessary to limit the amount of knowledge sharing in specific interoperable transactions.
4. Knowledge base validation is important for interoperability because of the consistency issue associated with individual KBs, and the ramifications for downstream propagation of possible misinformation.
5. Negotiation is an important attribute in interoperable KB systems because of the nature of most engineering design and manufacture activities.
6. Knowledge base comprehension is important for global context. To efficiently interoperate, KB systems require agents that describe the knowledge a specific KB contains, thereby streamlining search.
7. Knowledge capture is clearly achievable for specific domains, yet this activity remains a bottleneck.

8. Knowledge history, or meta-knowledge, is important to trace the reason for a particular conclusion or action.
9. Knowledge types must be varied for interoperability to be effective. Many types of objects should be recognizable - business objects, design objects, management objects and manufacturing objects.
10. KIF was developed as an interchange format and may prove very useful as a building block in representing knowledge across different KR schemes.
11. Design rationale is one level of knowledge that must be made interoperable.
12. Common Object Broker Request Architecture, or CORBA [OMG96], compliance is important for communication across different platforms and applications implemented in different languages.
13. Java™ [Cam96] compliance may be important for distributing knowledge across networks.
14. Problem solving method libraries are important so that meta-knowledge can be used to locate appropriate knowledge sources.

Action Items

The workshop concluded with a set of five action items that participants agreed to address. These are:

1. Begin surveying KBS developers and characterizing existing tools.
2. Develop sample practical problem involving multiple KB and CAx systems.
3. Define a taxonomy of domain entities, or primitives, that lend themselves specifically to interoperability in design and manufacture.
4. Explore the similarities and differences between KIF and the STEP data modeling language, EXPRESS, and its extensions.
5. Draft position paper on KBS interoperability discussing goals, challenges, strategies and areas of application.

References

- [Cam96] Campione, M. and K. Walrath, "The Java Tutorial: Object-Oriented Programming for the Internet (Java Series)," Addison-Wesley, 1996.
- [Gen92] Genesereth, M. and R. Fikes, "Knowledge Interchange Format, Version 3.0 Reference Manual," Computer Science Department, Stanford University, Tech Report Logic 92-1.
- [ISO94] "STEP Part 1: Overview and Fundamental Principles," ISO TC184/SC4, November 1994.
- [Mah87] Maher, M.L. and R.H. Allen, "Expert System Components," In: *Expert Systems for Civil Engineers: Technology and Application* (M.L. Maher, editor)," ASCE, New York, 1987.
- [McG93] McGuire, JG, et al., "SHADE: Technology for Knowledge-Based Collaborative Engineering, In: *Concurrent Engineering: Research and Applications*, Volume 1, Number 3, September 1993.
- [OMG96] "The Common Object Request Broker: Architecture and Specification, Volume 1," Object Management Group, July 1996.

Workshop Agenda

Monday, 3 November 1997, Shops Building (304), Conference Room

8:00 - 8:30 Registration and continental breakfast

8:30 Welcome to NIST

Richard Jackson, Director
Manufacturing Engineering Laboratory (NIST)

8:45 KBS Interoperability in Design: A NIST Perspective

Ram Sriram, Leader
Engineering Design Technologies Group (NIST)

9:00 Overview and Workshop Goals

Robert Allen, IPA Researcher
Engineering Design Technologies Group (NIST)

9:10 Knowledge-based Design Automation and Optimization Systems in a Production Environment -

Siu Tong,
Engenious Software

9:35 The ICAD System: A Generative KB Technology

Prasanna Katragadda,
Concentra Corporation

10:00 Intelligent systems using KB Engineering

Adel Chemaly
TechnoSoft, Inc.

10:25 Rule-Based Interoperability of Heterogeneous Systems

Stanley Su
University of Florida

10:50 BREAK

11:15 Configurator Synchronization

Bruce Ambler
Lucent Technologies

11:40 OKBC: A Programming Foundation for KB Interoperability

Vinay Chaudhri
SRI International

12:05 Knowledge Source Awareness models for Interoperable KB Systems

Ramana Reddy
West Virginia University

12:30 LUNCH

- 1:30 Breakout Group Organization
- 1:45 Breakout Working Groups
 - State of the art in KBS interoperability
 - Barriers and requirements for KBS interoperability
- 3:15 BREAK
- 3:45 Joint panel discussions of BG session summaries
- 4:45 Software Demonstration
 - Engenious Software
- 7:00 Social Hour and Banquet, Gaithersburg Hilton

Tuesday, 4 November 1997, Bldg. 304 - Shops Conference Room

- 8:00 Continental breakfast
- 8:45 Summary of Day One Results
 - Robert Allen
- 9:00 Group Discussion
- 10:30 Break
- 10:45 Summary Discussion/ Action Items Identified/Adjourn

Appendix A

Participant List

Robert Allen
NIST
Building 220, Room 127
Gaithersburg, MD 20899
301 975-3818
robert.allen@nist.gov

Bruce Ambler
Lucent Technologies
101 Crawford Corner, 4J205
Homdel, NJ 07733
(732) 949-5315
bambler@lucent.com

Richard Bolton
NIIP
1055 Washington Blvd.
Stamford, CT 06901
(203) 978-2946
rbolton@vnet.ibm.com

Scott Chase
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-6535
scott.chase@nist.gov

Vinay Chaudhri
SRI International, MS: EI225
333 Ravenswood Avenue
Menlo Park, CA 94025
(650) 859-3368
vinay@ai.sri.com

Adel Chemaly
TechnoSoft, Inc.
4434 Carver Woods Drive
Cincinnati, OH 45242
(513) 985-9877
adel@technosoft.com

Ismail Deif
Concentra Corporation
61 Londonderry Road
Grafton, MA 01519
(781) 229-4698
ish@concentra.com

Rohinton Irani
General Electric Corporation
1 River Road, Box 8
Schenectady, NY 12301
(518) 387-5635
irani@ crd.ge.com

Kevin Jurrens
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-5486
kevin.jurrens@nist.gov

Prasanna Katragadda
Concentra Corporation
21 North Avenue
Burlington, MA 01803
(781) 229-4617
prasanna@concentra.com

Michael Kerstetter
The Boeing Company
P.O. Box 3707, MS 2T-20
Seattle WA 98124
(206) 544-5535
michael.s.kerstetter@boeing.com

Hans Killicote
ICES/CMU
Hamburg Hall 1201
Pittsburgh, PA 15213
(412) 268-4097
killicote@cmu.edu

Bill Lange
InfoTech Enterprises, Inc.
4900 Seminary Road
Alexandria, VA 22311
(703) 671-1900
blange@infotech.org

John Mitchiner
Sandia National Labs
PO Box 5800, MS: 0722
Albuquerque, NM 87185-0722
(505) 844-7825
jlmitch@sandia.gov

Edward Mytych
Boeing Rotorcraft
216 Elton Circle
West Chester, PA
(610) 591-7846
edward.j.mytych@boeing.com

Walter Nederbragt
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-4467
walter.nederbragt@nist.gov

Kishore Pochiraju
Mechanical Engineering Dept.
Stevens Inst. of Technology
Hoboken, NJ 07030
(201) 216-8053
kishore@dmi.stevens-tech.edu

Y. V. Ramana Reddy
Computer Science Dept.
West Virginia University
Morgantown, WV 26506
(304) 293-7226
rar@cerc.wvu.edu

Mark Schwabacher
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-4574
mark.schwabacher @nist.gov

Ram Sriram
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-3507
ram.sriram @nist.gov

Stanley Su
DBSystems R&D, 470 CSE
University of Florida
Gainesville, FL 32611-6125
(352) 392-2693
su@cise.ufl.edu

Rachuri Sudarasan
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-4264
rachuri.sударасан@nist.gov

Simon Szykman
NIST
Building 220, Room 127
Gaithersburg, MD 20899
(301) 975-4466
simon.szykman@nist.gov

Siu S Tong
Engenious Software
1800 Perimeter Park Drive
Morrisville, NC 27560
(919) 462-7903
tong@engenious.com

Appendix B

Abstracts

Issues in Deploying Knowledge Based Design Automation and Optimization Systems in a Production Environment

Siu S Tong
Engineous Software, Inc.

This presentation describes the successes and difficulties in deploying knowledge based design automation and optimization systems in a production environment. Twelve years of developing and implementing KBSs at GE, and three years experience vending software, form the basis of these observations and conclusions.

To meet the industrial challenge of drastically reducing the product development cycle and cost while maintaining product quality, a new software system was developed at GE Corporate R&D in the early 1980s. It combined traditional mathematical based design optimization algorithms and modern knowledge based system (KBS) approaches to automate, integrate, and optimize engineering design. The software, Engineous, was successfully deployed in eight of 13 GE businesses. In the past three years, a redesigned, commercial version of Engineous, called iSIGHT, was developed and tested in large corporations in five major industries — aerospace, defense, power & utilities, automobile, and industrial manufacturing.

The hybrid knowledge based system and mathematical approach has proven to be useful and efficient in solving complex problems such as the design of aircraft engine turbines, power generation equipment, satellites, transformers, utilities planning, and electrical devices. On average, this technology reduces the design cycle time and manufacturing costs by an order of magnitude, saving tens of millions of dollars. However, there are many challenges in large-scale deployment of this technology to commercial users. The most difficult one is enabling end-users, not knowledge based system developers, to create and maintain the knowledge base. The existing KB systems are too complex for most engineers to learn, and developing complex, practical KBS application often takes too much time and effort. Also, there are many CAD, CAE, and other productivity tools (e.g., spreadsheet) in use in most design environments and substantial development efforts are often needed to link these tools together.

This presentation will highlight some of these challenges, discuss the successes and failures in working around these problems, and suggest future development/improvement of KBS that could significantly increase its use in a practical design and manufacturing environments.

The ICAD System - A Knowledge Based Generative Technology

Prasanna Katragadda
Concentra, Inc.

The ICAD System is a Knowledge Based Engineering software solution used by world class manufacturers in aerospace, automotive and industrial equipment manufacturing, such as Boeing, British Aerospace, Pratt and Whitney, GM, Ford, Jaguar, Lotus, and others, to automate system-level design, product design, tooling and product configuration. The ICAD System uses generative technology to capture and apply generic product design knowledge - both geometric and nongeometric - which includes product structures, development processes and manufacturability rules. Companies that use ICAD greatly trim cycle time, reduce downstream costs, and provide a flexible environment in which to process engineering change orders. Ultimately, ICAD System users shrink a good portion of the design or configuration process, allowing it to be completed in significantly less time than nonusers.

Recently, the ICAD "vision" has grown beyond the individual engineering effort. Through its KBO (Knowledge Based Organization) initiative, The ICAD System is attempting to examine, understand and define such aspects of an organization's "knowledge" as how it is represented, stored, examined, used, exchanged, updated and refined.

Recognizing that in today's business and engineering environment, knowledge without means of interchange is not very useful, our presentation also includes anticipated interoperability issues, such as representation and access methods for knowledge, and the role of international standards in facilitating these tasks.

Rule-based Interoperability of Heterogeneous Systems in NIIP

Stanley Y. W. Su
Database Systems Research and Development Center
University of Florida

Heterogeneous information systems such as agent systems, knowledge-based systems, database application systems and CAx systems generally have different data and knowledge representations and run on different operating systems and dissimilar computing platforms. To make these heterogeneous systems interoperable as an integrated information system on a local or wide-area network, one popular approach is to encapsulate the functionalities and data of these systems as objects. By doing so, they can be uniformly represented and processed in the integrated information system. This approach is taken by the Object Management Group (OMG), which introduced CORBA and ORB to provide the architecture and communication infrastructure for the interoperation of distributed objects through method activations. In the NIIP project, distributed objects are modeled in terms of 1) their structural properties and constraints using the international standard modeling language EXPRESS, 2) their methods using OMG's IDL, and 3) their knowledge rules using an event-condition-action-alternative-action (ECAA) rule language developed at the University of Florida. The ECAA rules capture enterprise business rules, policies, security and integrity constraints, and other rules of interoperation associated with distributed objects. An object-oriented knowledge base management system (KBMS) is used to provide the following:

- 1) GUIs for modeling, editing, browsing, and graphically querying the conceptual model of an enterprise,
- 2) An object-oriented query language OQL for accessing and manipulating metadata and shared data, and
- 3) An event and rule server to provide both build-time and run-time event and rule services.

ECAA rules are pre-compiled into rule code which are incorporated into program bindings generated by an IDL compiler for distributed objects, thus achieving "rule-based interoperability" over an ORB. They can also be stored in the KBMS and triggered at run-time when the enterprise knowledge base is accessed and manipulated.

Configurator Synchronization

Bruce Ambler
Lucent Technologies

Lucent Technologies sells complex telecommunications equipment, where much of the equipment configuration is custom designed for each sale. Engineers configure this equipment with the aid of two knowledge-based systems: the first is a sales configurator the second is a factory configurator. The sales configurator is operated by sales people and configures the product to a level that it can be priced and contracted. The second cofigurator is executed when the order gets to the factory and configures the components to the level that the equipment can be built.

Because changes in product design require the configurators to be changed, there is a need for interoperability between a product information system and the configurators. The configurators must be kept in synch with the product and each other since the output of the sales configurator is the input to the factory configurator. The interoperability requirements include an event notification service and a data exchange mechanism. The nature of the data exchange depends on the nature of the knowledge based system. Rule based systems require different information than constraint resolution systems.

OKBC: A Programmatic Foundation for Knowledge Base Interoperability

Vinay K. Chaudhri, Adam Farquhar, Richard Fikes, Peter D. Karp, James P. Rice
SRI International and Stanford University

Open Knowledge Base Connectivity (OKBC) is an application programming interface for accessing knowledge bases stored in knowledge representation systems (KRSs). OKBC is being developed under the sponsorship of DARPA's High Performance Knowledge Base program (HPKB), where it is being used as an initial protocol for the integration of various technology components.

OKBC is a successor of Generic Frame Protocol (GFP) which was primarily aimed at systems that can be viewed as frame representation systems and was jointly developed by Artificial Intelligence Center of SRI International and Knowledge Systems Laboratory of Stanford University.

OKBC provides a uniform model of KRSs based on a common conceptualization of classes, individuals, slots, facets, and inheritance. OKBC is defined in a programming language independent fashion, and has existing implementations in Common Lisp, Java, and C. The protocol transparently supports networked as well as direct access to KRSs and knowledge bases.

OKBC consists of a set of operations that provide a generic interface to underlying KRSs. This interface isolates an application from many of the idiosyncrasies of a specific KRS and enables the development of tools, such as those currently being developed at SRI and Stanford.

Knowledge Source Awareness Models For Interoperable Knowledge Based Systems

R. Reddy
Concurrent Engineering Research center
West Virginia University

Knowledge Based Systems, by definition, depend on one or more sources of knowledge for their operation. In a stand-alone knowledge based system, these knowledge sources are usually “attached” to the inference engine – the heart of the knowledge-based system. With the emergence of the World Wide Web (W3) as a seamless global information infrastructure, it is now possible to construct problem solutions based on a collection of cooperating knowledge based systems. In such an endeavor, each component may depend partly on the knowledge sources associated with one or more knowledge based systems in the group. This can only be possible if these component systems can inter-operate, insofar as they can exploit each other’s knowledge sources. Let us take a simple example of a case where two members of a team, each using an “expert office assistant” program wish to manage scheduling and communications. Each system depends on its own knowledge source – say an address book. Unless each system knows about the existence of an address book and deal with converting each other’s formats to their own representation, they can never cooperate – because they can not inter-operate. To overcome this problem, the following characteristics are needed:

1. A classification system for various types of knowledge,
2. A means for transforming one representation into another (perhaps using an intermediate canonical representation), and
3. A meta-model, which may be used by each knowledge-based system, to discover the needed source from the domains of the co-operating systems.

This talk provides some plausible scenarios for dealing with the above imperatives.

Appendix C - Presentation Slides

**Issues in Deploying Knowledge Based Design Automation and Optimization
Systems in a Production Environment**

Siu S. Tong
Engenious Software, Inc.

Appendix D - Presentation Slides

The ICAD System: A Generative Knowledge-based Technology

Prasanna Katragadda
Concentra Corporation

Appendix E - Presentation Slides

Intelligent Systems Using Knowledge-based Engineering

Adel Chemaly
Technosoft, Inc.

Appendix F - Presentation Slides

Rule-Based Interoperability of Heterogeneous Systems

Stanley Su
University of Florida

Appendix G - Presentation Slides

Configurator Synchronization

Bruce Ambler
Lucent Technologies

Appendix H - Presentation Slides

OKBC: A Programming Foundation for Knowledge-based Interoperability

Vinay Chaudhri
SRI International

Appendix I - Presentation Slides

Knowledge Source Awareness Models for Interoperable Knowledge-based Systems

Ramana Reddy
West Virginia University