NIST PUBLICATIONS

NISTIR 6159

NIST Design Repository Workshop
19-20 November, 1996
Workshop Proceedings

Edited by:

Simon Szykman
Ram D. Sriram

U.S. DEPARTMENT OF COMMERCE
Technology Administration
Manufacturing Systems Integration Division
National Institute of Standards and Technology

Stephen J. J. Smith

Department of Mathematics and Computer Science
Hood College

NIST

NISTIR 6159

NIST Design Repository Workshop
19-20 November, 1996
Workshop Proceedings

Edited by:

Simon Szykman
Ram D. Sriram

U.S. DEPARTMENT OF COMMERCE
Technology Administration
Manufacturing Systems Integration Division
National Institute of Standards and Technology
Building 220, Room A356
Gaithersburg, MD 20899-0001
szykman@cme.nist.gov
sriram@cme.nist.gov

Stephen J. J. Smith

Department of Mathematics and Computer Science
Hood College
401 Rosemont Avenue
Frederick, MD 21701-8575
sjsmith@nimue.hood.edu

June 1998

U.S. DEPARTMENT OF COMMERCE
William M. Daley, Secretary

TECHNOLOGY ADMINISTRATION
Gary R. Bachula, Acting Under Secretary for Technology

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Raymond G. Kammer, Director

Disclaimer

In this document, certain commercial equipment, software, instruments or materials are identified in order to facilitate understanding. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the items identified are necessarily the best available for the intended purpose.

Opinions expressed in this report are those of the workshop participants and not necessarily those of the National Institute of Standards and Technology or its employees.

Table of Contents

Presentation Slides 59

Papers 373

Introduction

Workshop Overview

The NIST Design Repository Workshop was aimed at understanding the needs and requirements for the development of large scale design repositories and databases. The two-day workshop addressed issues associated with representation and indexing schemes to facilitate the storage and retrieval of data, information, and knowledge required at various stages of the design process, such as design rationale, solid model, and assembly data. Other issues discussed included data sharing, collaborative engineering, interfaces, the role of the Internet in this research area, and critical technology gaps. The workshop included industry case studies, as well as overviews of related research being conducted in industry, government, and universities.

Note from the Editors

The text of the summaries that appear in this volume is based on the presentations made during the workshop. These summaries do not consist of a verbatim transcript of the workshop. Rather, they contain a distillation of the significant points of each presentation and the discussions that followed. The reader should not attribute any direct quotations to any of the participants in the workshop on the basis of this text.

Agenda

Tuesday, 19 November, 1996

8:00 am—8:30 am: **Registration, Coffee and Refreshments**

8:30 am—9:00 am: **Welcoming Remarks: Dr. Richard Jackson, Director, MEL**

9:00 am—9:30 am: **Overview and Goals of Workshop, Ram D. Sriram**

9:30 am—10:45 am: **Presentation Session #1: Industry Needs**

Lalit Chordia, Thar Designs
Information Needs for Pump Design

James Michael, Diebold Inc.
Information Retrieval During Design of a Medication Dispenser

10:45 am—11:00 am: **Break**

11:00 am—12:30 pm: **Presentation Session #2: Industry Needs (continued)**

Rich Zarda, Lockheed Martin
Development of an Adaptive Modeling Language (AML) for Knowledge-Based Engineering with Application to Interactive Gimbal Design

Tim Malueg, CRC
MADE-IGD Electro-Mechanical Gimbal Subcomponent Database

Carl Izurieta, Lockheed Martin
SAVE: Simulation Assessment Validation Environment

D. Navinchandra, IndustryNet
Product Announcement Clearinghouse: From the Trenches

Ed Harter, Boeing
Integrated Product Data Environment (IPDE)

12:30 pm—1:30 pm: **Lunch**

1:30 pm—3:20 pm: **Presentation Session #3: Industry Needs (continued)**

Gary Coen, Boeing
Design Data Treatment Issues at Boeing Helicopters

Wayne Collier, D. H. Brown & Associates
Seven Proposed Applications of Product Data Management

Gene Allen, The MacNeal-Schwendler Corporation
Use of Standards for Data Storage and Exchange

Kim Cobb, Lockheed Martin
Integrating a Distributed, Agile, Virtual Enterprise in the TEAM Program

3:20 pm—3:30 pm: **Break**

3:30 pm—4:45 pm: **Presentation Session #4: Government Programs**

Simon Szykman, NIST
Engineering Repository Projects at NIST
Michael Case, US-CERL
Open Collaborative Engineering
David Thompson, NASA Ames Research Center
Integrated Design Systems: Aeronautics Design Process Improvement

4:45 pm—5:00 pm: **Break**

5:00 pm—6:00 pm: **Presentation Session #5: University Research**

Yumi Iwasaki, Stanford University
Model-Based Support for Collaborative Engineering—Focus on Ontologies: How Things Work Project
Susan Urban, Arizona State University
The Shared Design Manager: A Repository for Integrated Product Data
Peter Will, USC-ISI
Active Catalogs: A Designers Aid
Sham Navathe, Georgia Institute of Technology
Engineering Design-Related Database Research at Georgia Tech

7:00 pm: **Dinner at Hotel**

Wednesday, 20 November, 1996

8:00 am—8:30 am: **Coffee and Refreshments**

8:30 am—10:00 am: **Breakout Sessions: Current Work, Technology Gaps and R&D Agenda**

Breakout Session 1: Industry Perspective (Chair: Mike Barbieri)
Breakout Session 2: Database/Knowledge-Base Frameworks (Chair: Pradeep Khosla)
Breakout Session 3: Standards for Information Modeling/Knowledge Representation (Chairs: Wayne Collier & Simon Szykman)

10:00 am—10:15 am: **Coffee Break**

10:15 am—12:30 pm: **Presentation Session #5: Government Programs (continued)**

Kevin Lyons, DARPA
Rapid Design Exploration and Optimization (RaDEO)
David Gunning, DARPA
High-Performance Knowledge Bases (HPKB)
Sharon Kemmerer, NIST
How Does a Standard Become a Standard?

12:30 pm—1:30 pm: **Lunch**

1:30 pm—3:00 pm: **Summaries of Breakout Sessions**

3:00 pm: **Workshop Close**

Glossary of Acronyms

- AI: Artificial Intelligence
- AML: Adaptive Modeling Language
- AMRF: Automated Manufacturing Research Facility
- AP: Application Protocol
  - AP 203: Configuration-Controlled Design
  - AP 209: Composite and Metallic Structural Analysis and Related Design
- API: Application Programming Interface
- ASIS: American Society for Information Science
- ASME: American Society of Mechanical Engineers
- ASTM: American Society for Testing and Materials
- AWACS: Airborne Warning And Control System
- BAA: Broad Agency Announcement
- CAD: Computer-Aided Design
- CAE: Computer-Aided Engineering
- CAM: Computer-Aided Manufacturing
- CATIA: Computer Aided Three-dimensional Interactive Applications
- CATIS: Computer Aided Tactical Information System
- CDME: Collaborative Design Management Environment
- CD: Committee Draft
- CD-ROM: Compact Disc—Read Only Memory
- CERL: Concurrent Engineering Research Laboratory
- CFD: Computational Fluid Dynamics
- CORBA: Common Object Request Broker Architecture
- COTS: Commercial Off-The-Shelf
- DAI: Data Access Interface
- DARPA: Defense Advanced Research Projects Agency
- DAU: Data Access Unit
- DFM: Design For Manufacture
- DFx: Design For "x"
- DIS: Draft International Standard
- DoD: Department of Defense
- FDA: Food and Drug Administration
- FDIS: Final Draft International Standard
- FEA: Finite Element Analysis
- HIPED: Heterogeneous Intelligent Processing for Engineering Design
- HPKB: High-Performance Knowledge Bases

- HTML: Hyper-Text Mark-up Language
- HTTP: Hyper-Text Transfer Protocol
- I3: Intelligent Integration of Information
- IBM: International Business Machines
- IDEF(0): Integrated Definition Language
- IEC: International Electrotechnical Commission
- IGD: Interactive Gimbal Design
- IGES: Initial Graphics Exchange System
- IPDE: Integrated Product Data Environment
- IS: International Standard
- ISO: International Organization for Standardization
- JCAHO: Joint Commission on the Accreditation of Hospital Organizations
- JSF: Joint Strike Fighter
- MADE: Manufacturing Automation and Design Engineering (precursor to the RaDEO program)
- MEL: Manufacturing Engineering Laboratory
- MIME: Multipurpose Internet Mail Extensions
- MSC: the MacNeal-Schwendler Corporation
- NAMT: National Advanced Manufacturing Testbed
- NASA: National Aeronautics and Space Administration
- NIIIP: National Industrial Information Infrastructure Protocols
- NIST: National Institute of Standards and Technology
- NWI: New Work Item
- OEM: Original Equipment Manufacturer
- PC: Personal Computer
- PDES: Product Data Exchange using STEP
- PDM: Product Data Management
- ProE: ProEngineer
- RaDEO: Rapid Design Exploration and Optimization
- RFQ: Request For Quote
- RRM: Rapid Response Manufacturing
- SAVE: Simulation Assessment Validation Environment
- SC4: ISO Technical Committee 184, Subcommittee 4
- SDM: Shared Data Manager; Shared Design Manager
- STEP: Standard for the Exchange of Product model data
- TEAM: Technologies Enabling Agile Manufacturing
- UL: Underwriters Laboratory
- USC: University of Southern California
- VCU: Version Control Unit
- VLSI: Very Large-Scale Integration
- VRML: Virtual Reality Modeling Language

Presentation Summaries

Welcome to NIST

Dr. Richard H. F. Jackson, NIST (jackson@cme.nist.gov)
(20 slides start after page 61)

Dr. Jackson said that this talk was one of his more pleasant tasks: welcoming the attendees, meeting them, and learning what they're doing. He told the attendees that they were about to do an important thing, that is, participate in a workshop. This will help NIST learn what the attendees need, that is, industry's needs for measurement, metrology, and standards. Thus, when technology is ready, the required measurement, metrology, and standards should be in place.

Dr. Jackson said that NIST was the only national research laboratory whose specific and primary mission is to serve U.S. industry, and that this mission had been substantiated by legislation. NIST has served U.S. industry since 1901 as the National Bureau of Standards, and since 1988 as the National Institute of Standards and Technology. It was in 1988 that the government added "assist industry in the development of technology and procedures" to NIST's mission. Guest researchers at NIST—who stay from a couple of weeks to a couple of years—are one means of technology transfer to industry.

Dr. Jackson said that the NIST mission was: "To promote U.S. economic growth by working with industry." He pointed out that "working with industry" was right up front in the mission. Many of NIST's resources support this mission.

NIST sponsors four main programs: the Advanced Technology Program, an industry research cost-sharing program; the Manufacturing Extension Partnership which helps small and medium-sized businesses adopt new technology; the National Quality Program which administers the Malcolm Baldrige National Quality Award; and the NIST Laboratories which focus on measurements and standards, including standard reference databases.

Manufacturing research occurs in some form in all of NIST's laboratories. The Manufacturing Engineering Laboratory (MEL) has four technical divisions. It also operates the Fabrication Technology Division, also known as "shops"—the people who make things for NIST. MEL serves primarily the mechanical, discrete parts manufacturing industry, but also addresses issues that cut across all manufacturing industries.

MEL is concerned with both physical and informational standards. Physical standards for length and mass are derived from first principles; all other physical standards are derived from these. Information standards for interoperability include IGES (Initial Graphics Exchange System), STEP (Standard for the Exchange of Product model data), and PDES (Product Data Exchange using STEP). MEL considers how to develop and disseminate these standards, and how to provide calibration services and reports.

MEL is organized around four basic programmatic thrusts. The Manufacturing Systems Integration thrust is working toward research and development of interoperability standards and integration technologies for the implementation of virtual manufacturing enterprises. The Intelligent Machines thrust addresses interface standards and performance measures for intelligent control systems. The Manufacturing Metrology thrust investigates metrology both in the laboratory and in the less-than-ideal manufacturing world. The Manufacturing Processes and Equipment thrust is concerned with high-speed high-precision machines.

Manufacturing has changed radically throughout history. When industry was moving from a mass production paradigm to an automation paradigm in the late seventies and early eighties, NIST realized that it had to change its programmatic efforts and research programs, so that it could be ready when industry moved to an automation paradigm. So NIST built the Automated Manufacturing Research Facility (AMRF). The AMRF project, which began in 1980, involved the technology transfer necessary to achieve automation. It was a completely automated manufacturing facility, involving both computer-integrated manufacturing and human-integrated manufacturing.

AMRF's accomplishments include contributions to IGES and PDES. In 1996, NIST decided to close the program, because automated manufacturing was no longer a technology of the future, but rather a technology of the present. It is not NIST's intention to compete with industry (though if a technology is commercializable, to promote successful commercialization NIST will seek to license the technology to industry; thus NIST has patented technologies). There is still more to be done in automation: machine tool controllers can be automated, and systems integration is also important.

If automated manufacturing is the present, what are the manufacturing technologies of the future? They may include flexible manufacturing, agile manufacturing, rapid manufacturing, rapid prototyping, and niche markets. Dr. Jackson described his "What's Next in Manufacturing?" slide as "lots of concepts and buzzwords." He summarized it by saying that in the future, there would be lean organizations, global in outlook, distributed in operation, and agile and flexible to adapt to customer needs. Information technology is changing what we do and how we operate. Information-based manufacturing is a new program at NIST. It used to be that the primary inputs to a manufacturing process were capital, material, and labor. Now, the addition of information technology and knowledge have become essential.

It is incumbent on MEL to address issues of measurement and standards for information-based manufacturing. Thus, MEL built the National Advanced Manufacturing Testbed (NAMT). NAMT is a showcase for the future of manufacturing, where people, computers, and software are networked. NAMT addresses standards and problems in automated manufacturing. NAMT will have connections from the control room to the factory floor. These connections may come from elsewhere in NIST, elsewhere in the country, or in the world. People with different languages and cultures are using the same piece of software to address problems, and will sell (or already have sold) the results.

There are four NAMT start-up projects exploiting information technology. The Manufacturing Framework program involves integration of manufacturing systems, and the development of a framework for standards and metrology. The Machine Tool Performance Model aims at a remote characterization of machine tools. Characterization, Remote Access, and Simulation of Hexapod Machines involves the development of high-quality simulation models for manufacturing processes and machine tools; the actual performance measurements that result will require a large-scale data repository.

The fourth program is Nanomanufacturing of Atom-Based Artifacts. Atom-based artifacts are the next generation of artifact standards, or "meter sticks." Atom-based artifacts push the limits of physics; in the future, standards will be based on atomic standards. Standards may involve counting atoms across line widths. Standard artifacts on this geometric scale are affected by air; thus these artifacts must be manufactured and must exist inside a vacuum. They are delivered inside a vacuum suitcase that connects to a scanning tunneling microscope. They can be used to calibrate machine tools remotely, across the street, across the country, perhaps even teleoperated from NIST.

One of NIST's services to industry is to provide high-quality data repositories. The goal of this workshop is to investigate technological needs for the design of such repositories. Such repositories must consist of individual assemblies and subassemblies, and might capture design intent. Questions to answer today include: Should NIST be doing something to support large-scale design repository development? Should NIST have some design data repository, or some collection of problems or designs?

Dr. Jackson finished by saying that he saw NIST as serving its customers. He reiterated the objectives of the workshop: to help NIST learn industry's needs for measurement, metrology, and standards. He expressed his hope that it would be a good workshop.

Opening Remarks and Introductions

Dr. Ram Sriram, NIST (sriram@cme.nist.gov)
(talk given without slides)

Dr. Sriram thanked the workshop participants for coming. He described them as falling into four classes of people: designers, researchers supporting infrastructure for designers, university researchers (who advance the state of the art), and government employees (who help industry through the development of new technology and standards). Part of the goal of the workshop is to gather requirements. Requirements-gathering is not taught in academia.

This workshop mixes the small-tool industry with large manufacturers. Lots of information goes into the design process, such as design knowledge. What are academia and industry doing? They are working at the conceptual design stage, with knowledge and constraints. This is information-rich.

The workshop objectives include industry case studies. The industry participants vary widely. For instance, Dr. Chordia works for a small business of twelve people, whose experience is with AutoCAD; small businesses are a large part of the United States manufacturing base. Diebold, Inc. is a company with five thousand people. They have design experience with paper handling and cash machines, and a division of about thirty people working on design of medication dispensers.

The workshop objectives also include roadmaps for research, from industry issues, to fundamental research issues, to standards. At first in the workshop we will discuss the state of the art, and then discuss the technology gaps, which will lead to a research and development agenda.

Information Needs for Pump Design

Dr. Lalit Chordia, Thar Designs (chordia@thardesigns.com)
(14 slides start after page 82)

Dr. Chordia started his talk by emphasizing that he was discussing the needs of a small business. The needs of small businesses are different from those of a large company.

Let's consider day-to-day design. Dr. Chordia's case study is high-pressure fluid pump design. Such small specialty manufacturing makes up the bulk of US industry. These pumps must withstand up to ten thousand pounds of pressure. A pump might be designed to combine two substances. For example, the combination of coffee with carbon dioxide results in decaffeinated coffee and caffeine by supercritical fluid extraction. Hops can similarly be extracted from beer. Restrictions on available space place strict constraints on viscosity and polymerization. Cleaning is also important; machine parts are cleaned with supercritical carbon dioxide. It takes 830 to 10,000 pounds of pressure to compress liquid carbon dioxide. Although pumps are a very old idea, there is very little information in the literature on carbon dioxide pumping.

Why did a small company among many pump manufacturers in the market decide to design one for liquid carbon dioxide? Existing pumps did not work well for high-pressure applications. Other pumps were designed for liquids, which are incompressible. Carbon dioxide is compressible; the dead volume is important. Ignoring the dead volume leads to inefficiency.

During design, we need to compensate for heat. This leads to cooling requirements. Cost-efficiency is also important; in order to be successful, a superior design would preferably also be lower in cost than what's available in the market.

A chart from Stanford University relates the volume inside pump heads to density, pressure, and temperature. This allows us to simulate what is inside the pumps. How tight should we keep tolerance to minimize dead volume? What manufacturing processes are required? How do we assemble pumps? Bearings allow us to maintain alignment and minimize dead volume. Stronger materials expand less, and expansion adds dead volume. Reducing heat generation is important because hot fluid is less dense, and therefore less efficient. Stainless steel pumps have a high frictional coefficient. Sapphire has a low frictional coefficient and high thermal conductivity; sapphire is better than ceramics, but more expensive as well. Making decisions on material requires not just mechanical engineers, but all engineering disciplines.

We believe, in general, that smaller is cheaper. A smaller pump head is a less expensive pump head. More critically, a smaller design has less stroke volume, which means that it requires more speed, which increases the friction; but the friction needs to be kept low. This is not an O-ring design. It is a C-channel design with a self-sealing ring inside it. The force on the piston is optimized by the force generated by the liquid itself. Strong material allows us to reduce length and diameters.

Because we are a small company, we need to bootstrap everything. Our concept formulation asks: what's available? what's not working? what do we need to do? The design of the chassis that holds the pump, and not the pump itself, was the most difficult problem. The final product was half the price of any competitors. It was scalable: it could pump forty ml/min, or forty thousand ml/min.

How could we have done better? We could have had rapid access to information: what's out there? what's not? what's good and bad about what's out there? what other pumps exist? how do people reduce friction? Perhaps we could have adapted someone else's design. We could have had finite element analysis to help us with the design.

The most important small business issue is that there are so many different kinds of technology and information. We need skilled labor and we need more of it every year. We get mechanical engineers who don't know AutoCAD; we encourage them to learn it via the Internet. Our future information requirements are many. We need access to information and tools. We need a design concepts database: we'd like to be able to use a motor from some other company in our design by means of a "drag-and-drop" operation. Should such a database be on the Internet or on CD-ROM? Few small companies have PCs (Personal Computers) with a CD-ROM on everybody's desk. We can design faster, better, and more competitively if such a database is on the World Wide Web.

A question-and-answer period followed Dr. Chordia's talk. Questioners are identified by number.

Q1: You went from a traditional design to a high speed pump with a small footprint. What happened to the lifetime of the pump.
A: It increased.
Q1: Do you have a baseline for that?
A: No, but we have a customer who's been using one for two years.

Q2: Would you want your access to design concepts to be via text or pictures?
A: Graphical or video.
Q2: It seems you would not want an exact solid model, but rather a vague description.
A: Exactly.

Q3: What would you want in a design concept versus what it takes to get a patent?
A: That's a tough question. We don't want a product patent; we want a concept patent.
Q3: Would such a patent application be too detailed?
A: Yes.

Q4: It took you six months to come up with a prototype. How long would it have taken if you had better information?
A: It took us six months to come up with a product and to work out the bugs. This would drop to three to four months with better information access and management.

Q5: How many prototypes did you need? Did you use ceramics in the prototype head or in the product head?
A: We needed two prototypes. Our head is a mix of ceramic and sapphire. A ceramic-coated steel piston was not as effective as we had hoped.

Q6: What information resources did you use?
A: Magazines called *Design Yields*, *Machine Design*, and *Design Facts*. We checked a *Thermos Catalog* to see what pumps were available.

Q7: What information technology tools did you use.
A: Absolutely none. We would have liked to use some.

Q8: What variables are most important for a small business?
A: Skill levels.
Q8: How about cost of tools?
A: That's important too. We use AutoCAD Lite. There's not enough work to justify spending $3,000 for AutoCAD

Q9: What do you budget each year for hardware and software training?
A: We do it on the fly.
Q9: How much did you spend last year?
A: $12,000.
Q9: What capabilities did the training add?
A: In a small business, the squeaky wheel gets the grease.

Information Retrieval During Design of a Medication Dispenser

James Michael, Diebold, Inc.
(11 slides start after page 97)

Mr. Michael introduced himself as a mechanical development engineer with Diebold, Inc. in its MedSelect Systems division. He has been with Diebold for a couple of years. Previously he worked with high-speed paper systems. He built his first design with Lego blocks when he was five years old.

People know Diebold as a manufacturer of automatic teller machines, or for physical and electronic security systems. We also work in information systems, material management, and billing. This talk focused on the design of a medication dispenser called a Unit Dose Module. During this design, there was information that was difficult to access during product development.

A hospital network has a database that includes data such as patient information. Each dispensing station has a computer. A nurse or doctor picks a patient and a medication, and the medication is then dispensed. The medication dispenser is similar to an electronically controlled chest of drawers. The Unit Dose Module is two feet high. It keeps medications under lock and key. It uses helix dispensing for tablets and capsules, and gated dispensing for vials and ampules. The medication drops off shelves into the chute at the bottom. You ask for exactly one dose and you get exactly one dose.

The Product Development Cycle slide shows a simplified version of the product development cycle. Information requirements included product requirements, competitive analysis, industry standards, and legislation. We obtained these information requirements through customers, marketing, patents, libraries, the World Wide Web, and competitive literature. We would have also liked to obtain these requirements through competitive products, information consolidation, and resource links.

The initial concept of development assumes that engineering is more heavily involved. We have obtained a marketing requirements specification and are now producing a functional requirements specification. We have a concept of what a solution is, but we need to do a technology search and test our solution.

We look into standards such as JCAHO (Joint Commission on the Accreditation of Hospital Organizations), a hospital standard, and ASIS (American Society for Information Science), a security standard. We found state regulations hard to obtain. Our questions included: what are the state regulations? how are they applicable? We read magazines such as *Machine Design*, as well as magazines from the ASME. We checked with medication suppliers to understand all the variables (including sizes, weight and weight distribution, material, and frequency of use), and their variability. Frequency of use is important: why design for medication that is rarely or never used?

The security of the locked cabinet is electronically accessed. To restock, someone must log in. But we also need a manual key override, for power failures and the like. Should the manual key override have one lock? Two locks? What are the relevant regulations?

A workshop participant asked if anybody at his company was developing advisory systems to aid in the design process. Mr. Michael answered that nobody was developing such systems; information requirements were simply being fulfilled, one design at a time.

“DFx” refers to “Design For something”: design for assembly, manufacturability, environment, disassembly, cost, or even design for environment though fewer people are familiar with the latter. In our design, do we need to design a new part, or can we use outside manufacturing through an original equipment manufacturer (OEM)? Is there an available supplier? We also satisfy some information requirements by ourselves, with information scribbled in notebooks and binders. We used TelTech, a technical knowledge service. Larger corporations have standards, such as UL (Underwriters Laboratory) standards for safety packaging. One of our OEM supplier resources, Thomas Register, recently went on-line.

We used CAD/CAE (Computer-Aided Design/Computer-Aided Engineering) databases. Diebold is moving to a CAD package with better information storing capabilities. Our PDM (Product Data Management) involves processes, building materials, and drawings. In the future, we see CAE-PDM integration.

Information access was accomplished via common access methods, such as the World Wide Web. Imagine a software package or a web site. On the "Information Consolidation/Resource Links" slide, below each phase of the development cycle is listed the relevant requirements. There are a large number of resources. We want appropriate links from requirements into the pool of information. We want links to private information, such as corporate standards. This information would not be available to the rest of the world. We also want links to personal information: "my" experience, and my associates' experience. This information also would not be available to the rest of the world.

Perhaps the software could have reminders that asked the user: "what about your experience?" or suggested: "check the corporate standards". Better yet, the software could allow the user to personalize the interface. Such personalization might include "bookmarks" or "notes." What would this offer the designer? It offers structure, a checklist for the cycle, and appropriate access that allows customization.

A question-and-answer period followed Mr. Michael's talk. Questioners are identified by number.

Q1: What is better about the new CAD package you'll be using?
A: It has a better solid modeler, and it enhances concurrent development with better communication and better conveyance of information. One can store preferred components in the CAD package.

Q2: How do you retrain "People Who Draw" to turn them into "solid modelers"?
A: We all currently use the CAD system. Some use the solid modeler. We would introduce some sort of training program. There are always some stragglers.

Q3: Our engineers get 70% of their information from their colleagues. Is your experience similar?
A: Yes. Your associates are right there. We are not trying to get away from that; we just want other resources to be accessible.
Q3: What percent of your product is made by outside vendors? How much is off-the-shelf and how much is made-to-order? Do you as an engineer interact with vendors, or do you get purchasers to do it based on specifications that you provide.
A: We have no OEM design team. Rather, we use OEM components. They are a significant percentage. We use a standard part when possible because it's cheaper.

Q4: What percent of the cost is used to buy things that you don't manufacture?
A: It varies from product to product. With the Unit Does Module, it is approximately 15%.

Q5: Do you have a library of suppliers' parts? Do they provide you with this information?
A: We created our own list of preferred tools.

Q6: How often do you create a new dispenser? What are the new requirements?
A: Typically we create new types of dispensers. The requirements include industry requirements and hospital requirements for the Unit Dose Module. We generally use a modification of an existing solution. Development is an ongoing activity.

Development of an Adaptive Modeling Language for Knowledge-Based Engineering with Application to Interactive Gimbal Design

Dr. Rich Zarda, Lockheed Martin (zarda@slr.orl.mmc.com)
(16 slides start after page 104)

Dr. Sriram said that the first set of talks had been given by people who were designers. The second set of talks concerned how people in industry support, or plan to support, designers such as Dr. Chordia and Mr. Michael. Dr. Sriram requested that there be no questions in the series of three Lockheed-Martin-related talks until the end, and introduced Dr. Zarda.

Dr. Zarda discussed the automation of the gimbal design process. A gimbal is similar to your head in that it has rotational degrees of freedom that correspond to a "neck" and "eyes". The most complicated Lockheed-Martin domain is the Apache Helicopter Gimbal. It takes twelve people nine months to design, and includes optical capabilities, mechanical capabilities, servo guides, thermodynamic analysis, and missiles. The investment into the development of an Adaptive Modeling Language (AML) for knowledge-based engineering with application to Interactive Gimbal Design (IGD) has significant impact on gimbal design. Indeed, it has significant impact on all programs.

On the "Metrics for Impact and Progress" slide, Design 1 was created "from scratch". It took about 8,634 hours to create. Design 2 was created from Design 1, but it still took about 7,000 hours to create, even though there were no hours spent for the servo, rather than 1,383. (Note that these are not actual figures—the time requirements have all been multiplied by a constant factor in order to protect proprietary information. Since the constant was the same for all figures, the ratios of times are correct.)

AML drives Pro/E, M-Vision, ManufacturingSoft, and many other products. The objective of the program is to develop and enhance AML. Then, we apply AML to our IGD system, and create a gimbal database with the intent of reducing design and redesign times. There are many requirements in an IGD system. There is also a material database, and there are many processes. We want to be able to save a gimbal design. No two gimbals are the same; this creates a nightmare for packaging.

Our program has accomplished several milestones. We had our kick-off meeting. We captured knowledge from the conceptual design process for optical design, and created a precision design product. Our optical simulation is now complete. We are developing a direct interface between Pro/E and AML. We would like to go through STEP (Standard for the Exchange of Product model data), but STEP is not available. We want an intelligent STEP database, and we want STEP to be an official standard as soon as possible.

The human head is a five-axis gimbal. More typically, we work with a two-axis gimbal. Most communication in the design process is word-of-mouth, not automated. The problem is that we pick a Pro/E product to make things, but it won't make gimbals. DARPA (Defense Advanced Research Projects Agency) and DoD (the Department of Defense) say that we have to cut costs by 50%, which means that we have to change everything.

What could help us? A mature STEP definition. A three-dimensional automated packing algorithm; it's easy to put a two-dimensional puzzle together, but products aren't two-dimensional—we need three dimensions! Electrical components are two-and-a-half-dimensional, and mechanical components are three-dimensional, which is why electrical design is ahead of mechanical design.

MADE-IGD Electro-Mechanical Gimbal Subcomponent Database

Dr. Tim Malueg, CRC (tmalueg@hsv.crc.com)
(11 slides start after page 121)

Dr. Malueg wants to capture data—not intelligent data, but performance data. Then, we can automate selection of particular components based on performance criteria.

Conceptual geometry involves a grammar for a fundamental three-dimensional model. Such a grammar includes primitives, such as a block and a cone. It should do interference checking. There are several benefits of such technology. Many components are not used as-is; we need to standardize the selection of components. We need to use a database to expand the availability of searchable subcomponent manufacturers. We need to assist in the identification and resolution of standards by talking to different suppliers of data, and establishing standards for characteristics. We anticipate vendor support through limited access to vendor databases. A flexible, dynamic architecture should support ad-hoc queries, and create new attributes.

In our system network design, proprietary designs are input and stay at the facility where they are proprietary. They stay behind a firewall. Our system architecture includes three interfaces to the Oracle database. Our replication process replicates the magnitude of data to our facility.

Consider bearing data. Compliance is the reciprocal of stiffness. We plan embedded applets and Java routines. We plan for tables in our architecture. We intend to dynamically generate tables on-the-fly with indices that expedite the performance of queries.

Important issues include speed versus flexibility. A relational database is fast, but it's static. Our focus is totally dedicated to gimbal subcomponents and engineers' processes. Should units be vendor-specific or standard?

SAVE: Simulation Assessment Validation Environment

Carl Izurieta, Lockheed Martin (izzy@mar.lmco.com)
(9 slides start after page 133)

Mr. Izurieta started by summarizing the SAVE (Simulation Assessment Validation Environment) program. SAVE integrates modeling and simulation tools into the environment. All of these tools are commercial, off-the-shelf tools; none are homegrown. They include the CATIA (Computer Aided Three-dimensional Interactive Applications) CAD system, as well as emerging technologies from DARPA programs.

We are validating the environment on the F-16 and F-22. This eight-million-dollar project started April 1995. We want the program to be in place in approximately the year 2000.

We want to simulate, cutting metal, rather than actually doing it. We are trying to make a plug-and-play simulation of the factory from Deneb Robotics. We want implementation and best practices: there is much technology being integrated in the current design process, so we can't do the design process in the same way we did it in the past. The benefits of SAVE for JSF (Joint Strike Fighter) were about 2%, 1.97% to be precise. But for about three thousand aircraft, about 2% is several billion dollars in savings.

The SAVE team brings expertise and experience to the problem. The SAVE program is managed in Marietta, Georgia. There are checks and balances in place for performance of the contract. The operational task force makes suggestions about what are right ideas and wrong ideas. The advisory board evaluates the suggestions.

This year we demonstrated the product at the defense manufacturers conference. The F-16 was redesigned for a horizontal tail in the mid-eighties. We used this redesign as a baseline; we validated our tools on that redesign—the system produced a similar redesign decision. The next phase is the development of a relational database interface. The interim demonstration, which will be given in February 1998, will include engine inlets. The final demonstration will be given at the end of 1999.

Product Announcement Clearinghouse: From the Trenches

Dr. D. Navinchandra, IndustryNet (dchandra@cs.cmu.edu)
(slides unavailable)

Dr. Navinchandra started his talk by saying that engineering design and the business process are transforming the buying and selling process. This led to the idea of using design repositories. Dr. Navinchandra first became interested in this idea in 1983 while working for a design engineer, doing engineering based on precedents. Say we are trying to build on something that already exists, i.e., not "clean-sheet" design. How do we represent designs? Money has poured into this problem for ten years.

Dr. Navinchandra became involved with IndustryNet to supply information needs for engineers. He was studying the control of metal-working materials, and manufacturing rules. He joined the company to make things happen in the real world. IndustryNet is one of the few companies that have tried to bring large design repositories on-line in a real-world area. There are hard issues, but an engineer's short-term needs don't require new research. Too much new research is rehashing of old work.

What do engineers require in the short term? What is important to survive in a real environment? The relationship between the specifying engineer and the prototypers is important. What is an OEM? How does a design engineer exist in the process with vendors and the costing department? A study from the automotive manufacturing industry revealed that 2% to 10% of the cost of a product was involved in buying. 10% to 40% was involved in selling. That left about 60% for the intrinsic value of the product.

Information liquidity is important. How do we provide intelligent catalogs? How do we provide information on day-to-day deliverables? Fancy finite element analysis systems are fine, but they are only a small part of the process. We need to integrate the decision-making process throughout the organization, not just with the engineer. A day in the life of a design engineer is our area of interest. We are trying to understand where the maximum percentage of the total value of the product is determined during the design phase. In business-to-business transactions—"sourcing"—most of the cost is during the design phase.

In automotive manufacturing, workers on the production line can perform multi-modal searches on catalogs. They search on part number, attribute, function, name, or application. They are trying to make an internal catalog standard into a national standard. Supplier database connection is not glamorous work, nor is ordering through central purchasing. This work must be recognized as valuable by the entire company. Each piece must be recognized as useful and optimized; integration is difficult.

One can locate vendors, searching by functionality, materials, or cost, and evaluate them. In the selection process, one needs to look at the catalog itself. Then comes the sourcing step: "where do I get it?"—the biggest value is determined during this step. Next, one creates, sends, and fulfills the order—less value is determined here. Then support assessment, which determines more value again. The whole transaction "wraps around".

In a long-running business-to-business transaction, the sourcing step involves locating and evaluating suppliers. No longer do design engineers give specifications for someone else to place the orders; they place the orders themselves. We are dealing with automated negotiation via intelligent agents (details of which are outside the scope of this workshop). In general, commerce occurs on private channels. It occurs on intranets and private networks, not on the web. This is not good, but it is a reality.

Interactive content architectures, such as product announcement clearinghouses, are a basic type of business-to-business technology platform. We've been beta-testing such a product announcement clearinghouse for a couple of months. Suppose you read trade magazines, see product announcements, clip and file them in an unindexed manner. Our goal is to automate that process with automatic notification, by fax, of product announcements. Fax is what engineers very often want, though we'll support email as well. There is a lot of mundane work involved. Dozens of people in a room develop an ontology for automatic classification of these products. Those interested in demonstration systems can contact Dr. Navinchandra.

We are testing agent architectures to provide information to design engineers. We want a standard common architecture for catalog information. We might develop a variety of different search methods: name, keyword, natural language, parametric, application-based, and function-based. Methods appearing earlier in that list are easier to achieve but less useful. The problem with parametric searching is twofold: getting data in the system, and normalizing it. For example, a drill bit consists of section for clamping the bit, and the cutting portion of the bit; some say the length of a bit is just the length of the cutting part, while others include the clamping portion in the length.

We need to carry cost information, but cost information is different for different companies. We would love to have an algorithm that can take a table in a catalog and interpret it. Several Ph.D. dissertations can be written in the area of "table understanding". Alas, this area of research is not "sexy".

Unlocking information is the knowledge engineering bottleneck. One needs to get data into the system. We found out in focus groups that we lose the customer if we don't produce the answer to a query quickly. There are massive amounts of information flowing into the system. We are preparing a paper-to-electronic conversion by retyping the paper and representing it with some strange version of the interleaf format. We want unskilled laborers to take the information and put it in the system cheaply. We are distributing the data entry tasks around the world. We want to automatically publish databases in whatever form is required.

To summarize the transformation of buying and selling relationships: The design engineer is important to the cost of the product.

A question-and-answer period followed Dr. Navinchandra's talk. Questioners are identified by number.

Q1: You're not authoring information, but rather reprocessing it. How do you deal with the copyright issues?

A: In two ways. First, we are reprocessing information on behalf of the ASME (American Society of Mechanical Engineers). They allow us to provide the information.

Q1: What if there's a mistake? Somebody uses your system and loses a million dollars. Who's responsible?

A: I don't know. The second way is private channels: intranets. We take responsibility for mistakes there, with a cap on the costs.

Q1: Do you have technical information?

A: Talk to me off-line.

Q2: Who are the customers? What is your business model?

A: We have 2500 free registered users. The system is an advertiser-paid model where the money comes from companies, sellers, who put their information on line.

Integrated Product Data Environment (IPDE)

Dr. Ed Harter, Boeing (edward.d.harter@boeing.com)
(39 slides start after page 143)

Dr. Harter asked the attendees to refer back to Dr. Jackson's remarks. One purpose of the workshop is for industry to communicate its needs. What are Boeing's needs and interests?

Dr. Harter's group, Boeing Defense and Space Group engineering provides computer support. The immediate issue is pressure to bring development costs down while increasing performance. Consider design analysis and engineering applications (e.g., aerodynamic analysis, with static stress, dynamic stress, infrared signatures, cross-sections, and so on). This analysis is informal and slow. There is data duplication in different product geometries. It takes nine months for one iteration, which is too long.

One solution is to integrate applications, and to extend the integration to manufacturability and assemblability issues. If you have point-to-point interfaces, if you add one application, you add n-1 interfaces. Thus n nodes have n(n-1)/2 interfaces. This approach is expensive and brittle, and our company (and other companies) are trying to avoid it.

In contrast to application integration, our approach is data sharing in order to avoid point-to-point interfaces. Another reason to use shared data is to decouple our work from specific vendor solutions. The old way locks us into specific vendors, which can be anything from very expensive to catastrophic. When our CAD vendor went from version 3 to version 4, it took us 500,000 person-hours to modify the support software, and it still didn't do everything we wanted. The key to data sharing is standards. Dr. Harter voiced support for STEP.

Boeing is involved in the DARPA MADE program, but their focus is slightly different from Dr. Zarda's. Boeing works on integration of design and analysis programs through sharing and standardization of data. The IPDE (Integrated Product Data Environment) is a subset of MADE. MADE involves the improvement of any kind of evaluation of design alternatives. We think our IPDE vision is indeed a vision, and not a hallucination. We may not achieve all of the vision, but we hope to achieve at least some of it.

We need to separate applications from data. The high-level view: we need an integrated product database based on standards. We need reading and writing via an SDM (Shared Data Manager). We need a DAI (Data Access Interface) for each domain, not for each product. For example, we need a DAI for geometry, and another for static stress analysis. Our CAD/CAE application produces a Part 21 file which is parsed by the DAI. The DAI takes only the relevant parts and breaks them up into Units of Functionality (UoFs), such as wireframe and topology. Units of Functionality is a concept borrowed from STEP. The SDM parses the UoFs into database entities, used by the beta version of Oracle 8. The entire process must be reversible.

We have very little money, so we are working with a small set of applications. For CAD we are taking CATIA. For aerodynamic flow analysis, we are using the Boeing A500 code. Our stress analysis application comes from MSC (the MacNeal-Schwendler Corporation). Our feature recognition system comes from Arizona Statue University. We must ultimately resolve the relationships between UoF models. Some of this can be done by the DAI if two UoFs are within the application domain. Some must be done by the SDM if one UoF is outside the application domain.

Prototype 1A of IPDE will be completed on schedule at the end of this month. It is a minimal version of a multidisciplinary optimization. Prototype 1B is the initial version of the SDM, including an interface for static stress analysis. The December 1996 deadline is sliding, because the Oracle version 8 software was eight weeks late. The release of IPDE version 1 is set for June 1997. CAD, Computational Fluid Dynamics (CFD), and Finite Element Analysis (FEA) should be in place, with SDM and a shared database. The demonstration article is a section of the wing of the V22. The problem with some applications was the sensitivity of data and information. We wanted to use an AWACS (Airborne Warning And Control System) example, which the military approved, but the commercial part of Boeing did not.

Design Data Treatment Issues at Boeing Helicopters

Dr. Gary Coen, Boeing (gary.a.coen@boeing.com)
(5 slides start after page 183)

Dr. Coen introduced himself as working for Boeing Helicopters in Philadelphia, which builds the V-22, Comanche, and Chinook.

Most of the design work at Boeing Philadelphia involves use of a database tool called EPIC, and CATIA datasets. This system indexes and retrieves data by aircraft model number plus a drawing tree representation of aircraft subsystems. For each dataset, the index is formulated as a composite key. For instance, the index 901-X-0001 identifies the V-22 aircraft model with the 901- prefix, X identifies a configured subsystem (e.g., avionics, structures, propulsion, etc.), and the 0001 picks out a section, assembly, or detail part of that subsystem. The idea is to view the index scheme as incorporating a limited notion of metadata. This indexing scheme has been sufficient to produce helicopters. It has no information about requirements, constraints, costs, or a mapping between requirements and constraints. If any of this information were there, it would certainly be used.

An indexing scheme has to be useful. Users tell us that it would be useful if the repository of designs had a good indexing scheme useable by each suborganization.

In the past, various attempts at process re-engineering, such as Group Technology, were initiated, but each had its shortcomings. Group Technology was about the "piggybacking" of auxiliary information on the indexing schemes, for example the tables on the "Feature-Based Design Storage and Retrieval" slide. This idea was very well-received and there was a lot of requirements-gathering. But the system was not ideal for users who had to respond based on the back-end features of an expert system, which therefore had to be error-free; this was impossible. Thus it was ultimately rejected and is now a historical article. It did not capture design intent.

If a new system were to be developed and integrated with a PDM, it would be useful for capacity planning by process planners and industrial engineers. We do not have a dedicated factory floor, but rather a competitive dynamic schedule.

Feature-based information would be very helpful for designers. A supplement to the CATIA/EPIC system is the feature-based ProEngineer design tool combined with IGES data translation. We are focusing on these systems to provide output for MADE. In our use of ProEngineer, we still use the assembly drawing number as the primary index. A design feature index would have a high return for the company. A standard catalog of parts would be useful to engineers. A feature library, though constructed over time, would pay for itself through re-use.

Engineers want real-time construction of lessons learned—design instances that satisfy constraints and requirements. One candidate for an indexing scheme would be a feature-centric indexing scheme, referring to design elements and constraints that were believed to be satisfied.

The ultimate rationale for design repository schemata is to encode, for each design, useful metadata to support retrieval and reuse. But even this will not be sufficient to guarantee capture of intangibles like intent and rationale. Kevin Lyons relates a story about how Lee Iacocca imposed a three-inch length reduction on a Mustang in order to affect a nine-inch reduction in space requirements for three Mustangs, in order to fit three (not two) Mustangs on a single rail car for transport to market. The underlying rationale for this cost-driven reduction in length cannot, in general, be associated with the drawings which implement it. However, a PDM could be useful in coordinating the distribution and use of such information.

To summarize my perception of what Boeing requires, design engineers and others in the enterprise need access to information about the entire aircraft, and this can only be implemented by capturing the relevant feature geometry, thus supporting clustering of designs which, in turn, will support reuse. We must capture design rationale, which currently remains unidentified, and we must capture design intent, perhaps deriving it from construction history, by analyzing design change trends made evident in its revision history. Most importantly, we need to integrate the design data, lessons-learned documentation, and product data management technology.

A question-and-answer period followed Dr. Coen's talk. Questioners are identified by number.

Q1: Do you have plans to incorporate process yield, reject history, or cost data? It doesn't show the effect of design decisions.
A: Group Technology did that, and it was useless. The defense industry players implemented a shallow standard rejection process. That information is valuable for the design repository, but it really is—or should be—part of the requirements.

Q2: Have you tried using conventional object-oriented techniques?
A: Our techniques have been very ad-hoc to this point. We're trying to make them less so.

Seven Proposed Applications of Product Data Management

Wayne Collier, D. H. Brown & Associates (wayne@dhbrown.com)
(slides unavailable)

Dr. Sriram introduced Mr. Collier as an expert in product data management (PDM). Mr. Collier said that he would do comparisons between product data management systems.

Mr. Collier described Arthur C. Clarke's short story, "Superiority". In the story, a military officer testifies in front of a tribunal that his side had lost because its enemy had had far inferior technology. They won the first battle, having more and better technology than their foes. But they didn't win as strongly as they had hoped, and they realized the war would take too long. The technology research group said "We haven't updated our technology for one hundred years. We can't make it just a little bit better. We need to make it all better at once." In an attempt to do this, the results was advanced technology that was almost right, but consistently failed because it was only "almost" right. In industry, Mr. Collier works in areas that are, similarly, extremely good, almost right, and very frustrating.

There was a structure proposed in *Administrative Magazine* in 1990. The structure was not quite "worse is better", but something like "technology fails in established firms when one small mundane thing is wrong". The important question is: is new technology in the context of the previous framework? The idea can be expressed in a two-by-two grid:

	Limited Technology Change	**Substantial Technology Change**
Core Framework Maintained	Incremental Innovation	Modular Innovation
Core Framework Overturned	Architectural Innovation	Radical Innovation

A company learns less well with modular innovation than with incremental innovation. A company learns less well with architectural innovation than with incremental innovation. But a company learns very well with radical innovation.

A new core business process is a move from maintaining the core framework to overturning the core framework. New tools are a move from a limited technology change to a substantial technology change. It's easy to move from an incremental innovation to a modular innovation. It's worst to move from an incremental innovation to an architectural innovation or to a radical innovation—yet that's what consultants often say to change.

The most successful progression is from incremental innovation, to modular innovation, to architectural innovation, and only then to radical innovation. Progressing from modular innovation to architectural innovation means restricting the use of tools and moving to a new business process. We did this with word processors and office automation products. You have time to learn the meaning of your business processes.

If you ask someone what a PDM system does, you get different answers. An engineer says that it's a design repository. A process control manager says that it's a change and configuration management system, or a parts classification management system, or a product management system. A manufacturer says that it's a build-on-demand system. A business process manager says that it's a process workflow system. This doesn't make sense, but what is really happening is that people are describing what they need. To be successful, listen to the customers: choose one, become good at it, and ignore the rest.

For the first application, change management, Agile Software makes a PC-only tool for domain-management. For the second application, part classification, use Aspect with commercial part libraries on line, or CATIS (Computer Aided Tactical Information System) with internal part database normalization. There are problems with domain-specific solutions. For the third application, design repositories, Workgroup Technology makes an up-front STEP-free integration system that saves parameters, as parametric systems are more generally known and useable. Some problems would be easy to solve if the right people would work on boring things. For the fourth application, we should build and configure engineering data on demand. This would involve generic and generative bills and materials.

The product family concept is important. For modular structure, define core components that are similar across many families. Modules are defined in terms of interfaces to core components. Another layer might be provided by AT&T Bell Labs/Lucent Technologies. As we configure, consider cost constraints and variable constraints. This is how AT&T develops telephone switching hubs (with researchers from the University of Eindhoven). An IBM product manager is ahead of Metaphase. There is nothing like serial change management.

For the fifth application, we need a collaborative workgroup enabler. This is what the original assembly tools were built for. What is the day-to-day environment? We need non-invasive tools in the concurrent engineering context. For the sixth application, we need imaging, replacing paper with scanners and printers. For the seventh application, we need global product life-cycle modeling, beginning with the commercialization of a product and ending with its obsolescence.

Use of Standards for Data Storage and Exchange

Gene Allen, The MacNeal-Schwendler Corporation (gene.allen@macsch.com)
(8 slides start after page 189)

Mr. Allen described himself as having worked with industry and government for ten years in collaborative research and development programs.

Ours must be a rapid-response program, reducing the response time in design. Collaborative development is the vision for most people. We have common requirements to achieve to arrive at that vision: we must have an accurate model of reality, with interoperability, that is easy to use.

CAD, CAM (Computer-Aided Manufacturing), and CAE evolved from different parts of the industry. They were not originally intended to be integrated.

The Rapid-Response Manufacturing (RRM) program has as its objective a heterogeneous architecture. It was adopted by NIIIP (National Industrial Information Infrastructure Protocols). EXPRESS captures information. The "Architecture for Interoperability" slide is a high-level overview of the model: much work needs to be done, and different pieces are at MSC (the MacNeal-Schwendler Corporation). Ed Stanton's generic pre- and post-processor can analyze anybody's CAD/CAM output. The company adopted EXPRESS internally.

MSC is working on interfaces to PDM systems through CORBA (Common Object Request Broker Architecture) wrappers. These interfaces are ease-of-use issues. Industry drivers are a product-specific focus.

Materials are as variable as geometry. MVISION provides materials information in a computer-readable form. End users perform analysis as part of the design process. The MVISION database architecture supports interoperability, reusability, and object-orientation. It is modeled in EXPRESS and compliant with STEP Part 45, and can be tied in with any PDM system.

Integrating a Distributed, Agile, Virtual Enterprise in the TEAM Program

Kim Cobb, Lockheed Martin (cli@ornl.gov)
(24 slides start after page 198)

Mr. Cobb began by describing the mission of the TEAM (Technologies Enabling Agile Manufacturing) program. Agility is the ability to respond quickly to unanticipated changes. The TEAM program includes GM, Ford, and Pratt & Whitney, and NIST in three of five project areas. TEAM focuses on implementation, not development. The idea is to find tools, put them in a toolkit, and demonstrate the toolkit through product vehicles. The TEAM goals are: better, faster, cheaper. TEAM hopes to provide a plug-and-play environment.

The "Team Product Realization Model" slide shows three phases of customer requirements. Concept optimization involves design/manufacturing tradeoffs during the performance evaluation phase. Virtual manufacturing involves the simulation and analysis of products and resources.

In March 1996, TEAM demonstrated a part of interest to the aerospace, automotive, and defense industries. There is a web repository for files. The part was created at GM and inspected at Ford. Enterprise Integration Thrust Area Activities include the Intersite File Manager. It was enhanced for Netscape. An early issue encountered was the need to deal with firewalls—the Intersite File Manager gave us an opportunity to deal with this real-world problem. The manager uses existing hardware and software, and it has user authentication to provide access control. It puts users into groups; if a file in a given group is updated, all group members are notified. The Intersite File Manager screen capture slide shows what has been done up to this point. One can get to a file simply by clicking on it.

Next summer, TEAM moves to seamless deployment, providing infrastructure for various tools. The concept of a cockpit is important: one web-based front-end to several tools. The Web Integration Manager knows what needs to be performed and invokes the appropriate tools. It notifies the next person in line, and provides files to that person.

Clicking on "optimization" in the Web Integration Manager results in a detailed model. The Web Integration Manager knows what inputs are expected, what outputs will be created, and what resources need to be invoked. Consider automation of concept optimization. We can change the values of ProEngineer parameters and run the optimization engine until the conceptual design is optimized. One conceptual designer can do CAD analysis, thermal analysis, stress analysis, and cost analysis.

Using the Concept Optimization Cockpit, we can update the ProEngineer model without being on the ProEngineer workstation. For example, we can change the diameter of a cylinder bore and have the ProEngineer model automatically be updated. We can click on buttons to perform analysis or optimization. We can do this for a detailed design as well as a conceptual design; the "Concept Optimization Cockpit" slide just shows a proof-of-concept—fine meshes lead to a slow demonstration, coarse meshes lead to a fast and therefore a better demonstration.

For additional information, you can take a look at our web pages at
<http://cewww.eng.ornl.gov/team/home.html>.

A question-and-answer period followed Mr. Cobb's talk. Questioners are identified by number.

Q1: How good are the object-oriented design facilities?
A: Not as good as we had hoped.
Q1: How do you handle Phase 0, Phase 1, and Phase 2 management processes?
A: I glossed over that in the talk. The Intersite File Manager provides version control; scripts have files that "scripting" doesn't change, even if we update the version.
Q1: Who's commercializing TEAM?
A: Ask me off-line.

Engineering Repository Projects at NIST

Dr. Simon Szykman, NIST (szykman@cme.nist.gov)
(18 slides start after page 223)

Dr. Szykman discussed engineering repository projects at NIST. He focused on small-scale projects. Dr. Szykman gave an overview of his talk: he would discuss motivations for the work, then two existing case studies, then future work.

The motivation is to bring information together into a single repository, or into a distributed information repository that may contain many different kinds of information. Currently, ideas are broad and the long-term objectives require better definition.

The first project, the Design Repository Project, is a subset of what's on the "Design Repository Project" slide. The last three ovals are commercial, off-the-shelf databases. At the top are interfaces, either existing or to-be-developed. Our focus is what's in the middle. The idea of the Design Repository Project is not to let the representation constrain the way that people create the design, and to provide a human-interpretable representation for engineering artifact information.

The Design Repository Project uses STEP AP 203 (Configuration Controlled Design) as one aspect of the representation. STEP is limited primarily to geometry; thus additional representations for other types of information used by designers are required.

For example, consider a power drill. The attribute buttons shown in the slides bring up additional windows. Click on a drill subsystem, drill_system_1, then click on motor_function_1. Keep clicking for additional information. The aggregate idea represented is that the function of the motor is to transform electrical current coming in from the black wire, into rotational motion at the clutch base. A STEP-based viewer allows visualization of the artifact geometry.

The second case study, the NIST Design, Planning, Assembly Repository, contains parts and assemblies in different formats. These formats include ProE, IGES, and PostScript. The repository also contains some process plans. The goals of the repository are to identify benchmark cases and problems, and to make them publicly available over the World Wide Web. These are real models, not toy models, many of which were provided by industry contributors.

Areas for future research include coming up with indexing schemes for information. We need a better definition of data first. Limitations inherent in STEP, such as lack of representations for features and constraints, must also be addressed.

A question-and-answer period followed Dr. Szykman's talk. Questioners are identified by number.

Q1: What do I do with a process plan?
A: You'll have to talk to Dr. Bill Regli (wregli@mcs.drexel.edu), who maintains the parts repository. Some plans are numerically controlled (NC) machine process plans; some are more sophisticated.

Q2: We could at least see what processes were being used.
A: One idea: have a part, find a similar part, or retrieve a process plan. This allows case-based planning, or variant process planning.

Q1: Do you sort the repository by similarities?
A: There is no sorting mechanism in the repository. You can search on attributes of the part models, such as keywords in file names, CAD file types, and so on.
Q1: Do you have any generic thoughts on design repositories—what should be in them, how to access them?
A: Those are among the reasons we organized this workshop; I could give my thoughts but I'd rather not impose my biases on the audience. At the very least, web-based access is a good idea.

Open Collaborative Engineering

Dr. Michael Case, Concurrent Engineering Research Laboratory
(m-case@cecer.army.mil)
(14 slides start after page 242)

Dr. Case's problem domain is facilities and buildings. There's no machining and they don't fly, but otherwise the problem domain has similar variables to those discussed by the previous speakers, and a variety of issues associated with design repositories.

The Army Corps of Engineers is the largest construction "company" in the world. But square footage of new construction is shrinking daily and funds are shrinking even more quickly. People now care about entire life cycles, not just the design stage. People who design buildings never meet people who live there. Buildings remain for ten, twenty, or fifty years; information for old buildings may be stored in old formats on eight-inch disks. There are thus archival problems.

The Architect's Associate is software designed to help architects work. It is an object-oriented design repository with links into a CAD system. It is shown on the slide in AutoCAD, though it could be MicroStation. Conflict resolution is done with a distributed design, not client-server.

Consider the "Discourse Model" slide. On that slide, ontology is equivalent to library; the messaging systems are event-based; there is management of permissions and ownership of artifacts. The implementation is in a homogeneous environment, as is the CAD system, which is undesirable.

The Agent Collaboration Language project involves building a design on the Internet. Stanford worked on the facilitator model, Massachusetts Institute of Technology worked on the structural engineering, Carnegie Mellon University worked on another system, and so forth. An interesting note: the project abandoned the idea of a common representation and embraced point-to-point translation which comes back to the n^2 translators problem—but n is small in this case. The translation engine could easily become "bogged down"; it was on a single machine, which was slow when it was busy, thus it has now been distributed.

The CITYSCAPE architecture is used for installation and management. The idea is to break things up into several services, such as CAD, Geographic Information Systems, and so on. The goal is to manage workflow and work orders. The architecture takes front-end engineering model designs, and has them flow through to those who need them. They are spending millions and millions of dollars to put maps and other hard-copy information back into the system.

The Modular Design System arose because the Army Reserve, like the post office, wanted standardization of building components. The technology they were using was dated, and they wanted to update it. The Modular Design System allows the user to drag and drop pre-configured reusable components. Design time was cut from eighteen months to six months. The system produces 80% of construction documents. Its limitation is that it focuses on Army Reserve Centers; it is difficult to add new types of buildings.

In the next four years, the intention is to introduce new technologies (from the top of the Open CE CRaDA slide) to produce new products (at the bottom of the slide). Right now, resource files are hard to maintain. First, we need something maintainable, at least. Next, we need to move to an object-oriented database. Also, we need to allow multiple people to work together. We have a laundry list of research and development issues. The sole reason for plug-and-play is that we don't want to be locked into specific vendor solutions. We want to reason about relationships between objects, so we need a separate relationship repository. We hope to be able to browse on the web, find a heat exchanger, and drop it into a design; we need an enabling language to do this.

Suppose in 1996, we create a drawing, and keep it around for fifty years, and the specifications change? There is a schema evolution problem. Versioning for designs has been derived from software engineering versioning. But we need a better versioning system. Chronology involves how things change over time. Ownership is important metadata: who created a particular design?

We need software agents to manipulate models. We need visualization; no application comes without its own custom user interface. We need visualization engines; VRML (Virtual Reality Modeling Language) is evolving into one. We are interested in replication, as different parts of models may show up in different places.

Dr. Collier interrupted Dr. Case to ask if they used pre-existing commercial tools. Dr. Case replied that they were using only relational databases, not translation. Dr. Collier commented that toolkits for passing around design tools exist, but are criticized because they are not used in real applications. Because Dr. Case was creating a real application, Dr. Collier urged him to use these toolkits. Dr. Collier continued by saying that we will always need translation. University of Southern California, Stanford University, and IBM came up with win-win strategies for conflict management.

Integrated Design Systems: Aeronautics Design Process Improvement

Dr. David E. Thompson, NASA Ames Research Center,
Computational Sciences Division (dethompson@mail.arc.nasa.gov)
(20 slides start after page 257)

Dr. Thompson began by mentioning that NASA has shifted its focus from basic research and development to providing more support for the economic competitiveness of the United States. One possible focus for this support is Integrated Design Systems. As part of that focus, Dr. Thompson works on Aeronautic Design Process Improvement. The first goal in this work is to assess what the aeronautics industry needs NASA Ames to work on. The Airbus plane is a heavy, modular aircraft, and is generally cheaper to build than typical U.S. commercial aircraft. And because of its relatively lower price, the U.S. aerospace industry needs to reduce cost by about 25% to compete with Airbus pricing. The cost driver provides a variety of opportunities for improvement. NASA is not involved in commercial business practices, cost models, or profit strategies; those are industry's job. But NASA can work at improving the overall design process with the infusion of advanced information technology tools and systems.

Much of a commercial aircraft designer's work is derivative, involving redesign. NASA focuses not just on the conceptual stage, but across the whole development lifecycle. Dr. Thompson's program strategy consists of providing a testbed to co-develop technology with industry so that the overall lifecycle can take advantage of emerging technology.

Dr. Thompson defined a "thread" to be a design activity involving one or more engineering disciplines. One design process improvement target is reducing time duration for a single thread. For example, consider wind tunnel usage. Currently, validation and analysis of wind tunnel models is an iterative process, lasting about 24 months, consisting of 3 eight-month cycles. Frequently, by the time the wind tunnel testing of a model is complete, the designers have changed the design. To make the design process quicker, one can improve the use of wind tunnels by providing on-line remote access to the experimental data as well as to similar computational fluid dynamics results. This eliminates some of the analysis and redesign needed, decreasing the time for each iteration and increasing overall effectiveness of time spent at the tunnels.

Another design process target is the speeding up and unification of multiple threads. In the 1990's, after the requirements are given, designers independently work on an aerodynamics thread, a structures thread, a stability and controls thread, and so forth. After some time, configuration management occurs, and the designs are reconciled, making a moderate design improvement at the cost of a significant amount of time. In the decade of the 2000's, advanced software systems that support configuration management and control of the product model can control more frequent updates, resulting in a high frequency of small design improvements. Because the design is under a deadline, these higher frequency changes will allow faster response to changing requirements, exploration of a greater number of alternatives, and will produce a better product.

Consider a design activity example using the type of interactive system envisioned. Suppose you're a designer, and your job on a given day is to select a radar for a plane under design. First, you acquire and display the data and constraints for radars available from vendors. Understanding how stiff or restrictive the design constraint-space is helps determine what vendors could provide an acceptable radar that fits in the plane nose-cone. A non-interactive design system might lead you to think that there is only one acceptable radar.

But what about understanding the nature of the constraints? Say there is a nose-cone volume constraint which, if relaxed, would allow another acceptable radar at a significant savings in cost. You might try increasing the size of the dome in the nose of the aircraft to provide the additional volume. Then, the design system would allow you to start up a collaborative activity, consulting with experts in manufacturing and operations, structures, and aerodynamics, to evaluate the new design configuration. The experts in structures and manufacturing might approve the change based on acceptable materials and size, and the aero expert might launch a CFD code that indicates that the larger dome is still small enough that no new turbulence would be induced. The change would then be approved, the product model updated, and the less costly radar system ordered.

A workshop participant interrupted Dr. Thompson to ask if he had such a system. Dr. Thompson replied that NASA is building a first prototype of such a system this year [1996-97].

Dr. Thompson said that NASA has invested in intelligent database and knowledge extraction techniques, intelligent tools and modeling representations, smart systems integration agents and data fusion. NASA is now investing in collaborative communication tools and infrastructure, immersive environments and navigation capabilities, and a distributed, high performance simulation design environment. The collection of these capabilities will provide the backbone of the future improvements envisioned.

Dr. Thompson's program plan involves developing only what can't be acquired commercially. The technology transfer plan is unique because the technology is being co-developed so that transfer evolves as development evolves. Roughly 80% of the effort is being done collaboratively.

A question-and-answer period followed Dr. Thompson's talk. Questioners are identified by number.

Q1: Who are your partners from the commercial information technology environment?
A: We are still sorting them out. Ames was awarded more than anticipated for this prototype system, and we are in the process of putting together a work plan and spending plan.

Q2: Some companies say they focus on collaborative design display--would you spend money on them?
A: That's too narrow at this point. Suppose the problem is: what is the amount of lift during landing? The design problem is: what is the best proportion of overlap versus gap for the wing flap to maximize the lift? Using a new visualization tool, a designer might get contours, showing a variety of configurations that give maximum lift. But given deployment constraints, then the best configuration must satisfy those constraints, which can be plotted against the results from simple computational fluid dynamics runs. Such a design activity requires completely new types of visualization capabilities that are not yet commercially available.

Model-Based Support for Collaborative Engineering—Focus on Ontologies: How Things Work Project

Dr. Yumi Iwasaki, Stanford University (iwasaki@ksl.stanford.edu)
(6 slides start after page 278
A paper on this topic begins after page 375)

Dr. Iwasaki said that her focus was on declarative ontologies for collaborative and multidisciplinary designs, and on ontology libraries. Her work is part of DARPA's MADE program, and also supported by other DARPA programs. Dr. Iwasaki has been working for the last several years on model-based reasoning technologies, and automatic modeling formulation for quantity and quality. The project is in year one of three, a new phase in which she will deploy and extend technology in support of collaborative engineering. In this talk, she would be concentrating on the ontology aspects of her work.

An ontology is defined as a body of terminology used to describe a particular domain of discourse, including classes and relations. Ontologies are similar to frame-base systems and define your vocabulary for each domain. The problem in supporting many disciplines is that each definition has its own ontology. For example, a database, a company, and a program might have slightly different ontologies.

Suppose there is a multidisciplinary team working on some device. For example, an electrical team, a mechanical team, an optical team, and a thermal team might all be working on the same design. All teams have different ontologies: across teams, the same word might have slightly different meanings, and the same meaning might be described with different words.

We have developed a declarative representation of ontology with definitions. We have created a web-based ontology editor that is available to the public. Having the ontology of one domain is very useful; it shows a conceptualization of the world. An ontology shows a classification of objects that you have. Ontologies are even more important in collaborative work.

The space of ontology looks something like the "Domain Specific Ontologies" slide. There may be some co-ontology common to all disciplines; for example, mathematical concepts. But most ontologies are not are broadly applicable. The declared ontology of disciplines helps, but domain-specific models are also necessary.

To allow communication, the system engineer's view of the model is necessary. For example, mechanical engineering words may not exist in an electrical engineer's vocabulary. The CDME (Collaborative Design Management Environment) model can facilitate accurate communication among disciplines. CDME is not a universal model; it is just sufficient to allow communication. Our hypothesis is that it is not very productive to force everyone to use the same vocabulary.

Our particular application domain is a microsatellite. This is a satellite that you can just about put your arms around. Suppose you want to launch and operate it. We are modeling the electrical and mechanical aspects.

Another direction of our research is function-based information retrieval. We were told many times by electrical engineers that information retrieval is a big problem. Unless you know what you're looking for, it is very difficult to find a product. Existing catalogs are organized along conventional classifications. We are developing two ontologies of functions: one large, one small. We allow you to translate the familiar into the canonical.

The Shared Design Manager: A Repository for Integrated Product Data

Dr. Susan Urban, Arizona State University (s.urban@asu.edu)
(7 slides start after page 282)

Dr. Urban introduced her talk as a continuation of Dr. Harter's talk. The Shared Design Manager (SDM) is a repository, based on STEP, for managing data sharing. SDM goes beyond file check-in and check-out, addressing issues like how entities within files are related? There are configuration management issues as well: what are the relations? what are the iterative processes?

Units of Functionality (UoFs) are subsets of STEP AP 209 (Composite and Metallic Structural Analysis and Related Design). The idea is to avoid point-to-point communication by using UoFs as the main units of data transfer, consisting of geometry, finite element analysis (FEA), FEA controls, and CFD results. A tool can use a Unit of Functionality to derive additional information, and then check it.

The "Shared-Design Manager Architecture" slide is a high-level picture. Applications have interfaces to communicate with DAIs. Either a DAI breaks STEP into Units of Functionality, or it passes the STEP to the SDM to break into Units of Functionality. The Integrated Product Database (IPDB) is based on the beta version of Oracle 8. Querying the repository is much simpler with a relational database, in comparison to an object-oriented database. EXPRESS maps better to a relational database than to an object-oriented database.

Data can pass inside the SDM through the Version Control Unit (VCU) to the Data Access Unit (DAU). Other boxes represent other functionality to be added to SDM.

A workshop participant interrupted Dr. Urban to ask whether event notification and user directories were standard components. Dr. Urban answered yes, they were.

Dr. Urban continued by describing the SDM metadata. Applications map down to Units of Functionality, which sometimes map further down. Sometimes EXPRESS is mapped to Oracle 8. The ST-Developer, an application from STEPtools, Inc., is an EXPRESS compiler, parser, and generator. The "SDM Data Access Unit" slide illustration shows that a Part 21 file comes into the system. The file either has one Unit of Functionality, and goes directly to the integrated product database, or multiple UoFs, which the SDM splits appropriately before storing in the integrated product data base. The DAI/SDM interface is defined using CORBA, Orbix in particular, and extracts relationships of UoFs from Part 21 files.

Active Catalogs: A Designers Aid

Dr. Peter Will, University of Southern California (will@isi.edu)
(17 slides start after page 290)

Dr. Will said that his talk initially consisted of a project pitch, not as advice. However, he was recasting it into advice. When he first began this project, he had been working in engineering management for five years. He wanted to get products out at half the cost and with one-tenth the time to market. Getting cost down is all in the first ten percent of the design, or maybe even in the first ten minutes of the design! This project is working toward bringing costs down very early.

Hewlett-Packard makes thousands of products a year, but not thousands of new designs a year. New products are slow because an error, such as a bug in VLSI (Very Large-Scale Integration) chip design in a PC board, requires reworking the design.

His advice is to collect data, then mine it, then come up with a reasonable probability model to describe various design attributes. This should cut down on time. Time is more important than money, because time can establish a market. Hewlett-Packard designs instruments based on models. If they expose the models to users, they can design based on these exposures; this is a good sales tool.

The "Concurrent Design Activities" slide shows the sort of thing that should be in an active catalog. Reuse should allow one to mix and match components. We want to reformulate a partial functional description into a set of reasonably queries. We need to go through the loop on the "Active Catalogs Scenario" slide lightly and not deeply.

In general, the consumer of libraries is the human eyeball. But in CAD, geometric models are important. We need to know the three-dimensionality and the motion of a part. We need a searchable catalog like the Thomas Register. Objects in active catalogs have richness and modality. Active catalog components are interoperable; usually we need vector optimization, global optimization across domains.

What has been done? A pump ontology has been created that users can browse through. A semantic network is what allows the system to answer user queries. The problem is that it took four months to generate this ontology; researchers must get together to build and share these ontologies! What do you do once we have these ontologies? The system shown in the "Active Catalogs Architecture" slide has been built. It distributes execution to the available resources – load balancing. In the future, it will run on the World Wide Web.

Engineering Design-Related Database Research at Georgia Tech

Dr. Sham Navathe, Georgia Institute of Technology (sham@cc.gatech.edu)
(21 slides start after page 308
A paper on this topic begins after page 384)

As the last talk of the day, Dr. Navathe stated that he wanted to make sure he started with a conclusion. There are many challenging problems; we need to look at the basic problems in detail. We hope that industry is receptive to funding investigation of these basic problems, not just to funding the delivery of a running system.

HIPED (Heterogeneous Intelligent Processing for Engineering Design) was recently finished for DARPA. The idea was to marry AI (Artificial Intelligence) tools to database back ends. Most AI tools are content with main memory, and are not interested in data repositories. HIPED was part of DARPA's Intelligent Integration of Information (I3) program. I3 did not concentrate on engineering design; rather, it saw design as an application domain.

We map a request based on knowledge from the metadata. The problem in matching tables is that what is a column name in one database is a relation in another and a value in a third. There is a heterogeneous back-end correspondence, which describes how relations are represented. Query mapping rules take the query and process it for the back-end.

CORAL is a Prolog-like declarative language. It deals with rule bases that are bigger than main memory. Facts and rules encode information. Database correspondence rules describe meta-knowledge processing.

The querying interface processes certain types of requests for demonstration purposes. Such requests are illustrated in the slides by the (Prototype battery) (Property current) query. Consider two databases. How can one map battery requests to both databases?

A workshop participant interrupted Dr. Navathe to ask if an accurate rephrasing might be to make consistent queries across a federation of data models. Dr. Navathe said yes, plus the flexibility to add data models and relationships. A consistent front-end marries data in the back.

Dr. Navathe continued by saying that he was looking for a partnership with someone in industry to push his work one step forward. There is a need to improve user performance for large document, full-text repositories. This goes along with visualizing the document space in a much better manner. Schema evolution becomes the main target of an object-oriented database system, a problem that we saw today.

Dr. Navathe concluded by mentioning other big mechanical engineering projects at Georgia Tech. The multimedia "smart catalog", meant to be used by naive users through experts, interfaces databases with text, videos, and inventory. The engineering databases in the School of Architecture are concerned with research into view updating.

Breakout Session #1—Industry Perspective

Discussions summarized by Dr. Stephen Smith (sjsmith@nimue.hood.edu)
(discussions done without slides)

Participants:
- Michael Barbieri (chair)
- David Flater
- Peter Hart
- Yumi Iwasaki
- Carl Izurieta
- Kevin Lyons
- Irena Nagisetty
- Linda Schmidt
- Stephen Smith
- Rich Zarda

In the discussion that follows, only Mr. Michael Barbieri, the chair, is identified by name; other participants are not identified.

Q: Let's brainstorm on topics to consider, and then decide which to present to the group, or a couple of related topics.

Q: Ideas for what? Needs for design repository or broader?

Mr. Barbieri: We could broaden if you want.

Q: That's broad enough.

Q: We want a nationally-supported material database with as much meat as we can put in it. It should support any type of simulation, stress dynamics, ordnance analysis. It should have a good pedigree: where it came from. It should especially support simulation on composites.

Q: MVISION is a mechanism to have that data, but we have to populate it.

Q: We need standard testing procedures and results.

Q: We need chemical properties, physical properties, process for manufacturability (how materials behave with processes), thermal coefficients, and electrical properties.

Q: Should we worry about bottoming out with specificity? Industry has in-house procedures for using data.

Mr. Barbieri: NIST puts data in the database and certifies you if you have fulfilled standards, set by industry professionals at NIST.

Q: Are there industry standards in place?

Mr. Barbieri: I don't think so.

Q: ASTM (American Society for Testing and Materials) has standards, but all tests have certain restrictions. For example, how did the environment affect the testing, and associated information about how the testing was conducted.

Q: Even with ASTM standards, companies perform their own tests.

Q: There's so much research funded by the government, but there's a lot of overlap. There should be a database with all government projects in one place that you can search by keywords, for example, "high-speed machining". Go to a web page, type in a word search, find out the point of contact, the purpose , the schedule, the funding agency, the principle investigator, and a link to the web page of project. Call it the "National Research Database".

Q: There's some technical center in Wheeling, West Virginia called the Byrd Center, that you can search free of charge, that includes 70% of the labs and contacts for the other 30%. Any government-funded activity, when funded, requires you to put in information in the form of a one-page summary.

Q: If there were a repository, what information would be most useful for your design process, to make it cost-effective and to minimize time to market?

Q: What if parts and subassemblies were in a case-study repository?

Q: The Lockheed-Martin needs: for 500 parts in a gimbal, 150 of the 500 are off-the-shelf, so we go to catalogs. We're trying to collect these catalogs as part of MADE/RaDEO (Rapid Design Exploration and Optimization). We need a national effort, beyond STEP. For bearings, we need stiffness and dimensions from graphs, balls, pad, raceway, all on-line. A list of catalogs would help, but a standardization of the format would really be useful.

Q: A lot of vendors let customers, but not other vendors, have information. Proprietary information requires security.

Q: Does PartNet let you search for bearings? What's missing?

Q: The details!

Q: What's PartNet?

Q: A research project that became a commercial outfit, but it's very low-scale.

Q: Gimbal requirements are very specific. Bearings are precision-mechanical. They have very tight tolerances, beyond what Ford or GM might need.

Q: Get industry users together to set requirements.

Q: But there are thousands of user groups!

Q: For the RaDEO program, say "this is what we want."

Q: If McDonnell-Douglas builds gimbals, do they need that information?

Q: About 70% of it.

Q: The basic issue is that we need a standardized taxonomy of parts and assemblies. Physical properties should be indexed to manufacturing needs. This doesn't exist.

Q: We need standards of description for electronic components, or any kind of components.

Q: NIST should set some requirements for minimal characterization attributes and put the information in a database.

Q: Is the information you need in catalogs, or do you need to call a vendor?

Q: It depends. Preliminary decisions and half of the final decisions can be made with catalogs. The other half require calls to a vendor.

Q: There's another problem. Suppose that there is a bearing whose characteristics are in the database, and it is close to, but not quite, what is wanted. 90% of the time you can issue an RFQ (Request For Quote) and put in an order for something slightly different than what's off-the-shelf.

Q: A bearing has an inner radius, an outer radius, and a hub radius. An IGES/STEP file might have more detail, but you don't need it. You can't, for motors and other parts with more than three parameters, search on the STEP file. Also, you need a footprint to perform interference checking.

Q: Perhaps a database with different levels of detail. A pointer to a STEP file, and certain geometric constraints.

Q: We want something in-between a pointer to a STEP file and certain geometric constraints for interference checking and packaging problems.

Q: A user group on one side could type in requests, send them to a set of suppliers, and then they say, "yes, here it is, here are the characteristics, here's a picture", and so forth.

Q: We need constraint-based queries to the database. The information should be structured in a format to be queried, a way to get to the information without getting too much detail.

Q: Pick whatever five parts you think meet your requirements, send questions to suppliers asking, "do they meet it?'

Q: These constraint-based queries should be summary/abstracting/derived views based on the STEP model: "data needs".

Q: We need more than a geometric model of a motor. For example, we need key surfaces, torquer's footprint, performance attributes, an idea of the connections, behavior, and functionality.

Q: There is no functionality in STEP. You need to pull it out based on user needs, perhaps kinematics and dynamics, perhaps footprints.

Q: Engineers at Boeing specify information by writing on drawings, making "buy/build" decisions. There's a lag time between setting the requirements and getting the order. The problem is not getting what's required, re-specifying the particulars, maybe doing some testing. Then give names of vendors who meet requirements to the purchasing department.

Q: Supply chain is a major issue. Suppose an engineer uses an automated method to send an RFQ, and the company has internal processes to decide whom to send an RFQ... it would help if we had testing information attached to the supplier.

Q: There is great variation in what is received in bearings, for example for helicopter engines. We need industry standards for what minimum testing is required.

Q: Users sign proprietary agreement and the vendor gives the data.

Q: Should we pick a test part for the database?

Q: Bearings, gears, hubs, parts of drive chain.

Q: 70% of the cost of a missile is the gimbal, and all the various components that go in the gimbal, including gyros, and focal point arrays.

Q: There's a lot of overlap in the ways Boeing and Lockheed-Martin do this.

Q: A design repository should be a collection of all this information; electrical catalogs; perhaps search engines for historical databases, other people's databases, or CAD files. The big issue is accessing the information; we already have the information.

Mr. Barbieri: So far, there are three main ideas. A commercial off-the-shelf (COTS) database, a national research database, and a national material database.

Q: Should we have a common repository for designs, not just design parts?

Q: That would most likely be internal.

Q: There are some obstacle issues: taxonomy of parts of assemblies, design artifacts—the physical things that result from manufacturing processes.

Q: There should be a common classification. For example, an angle is different from a rotor—how? There need to be umbrella concepts.

Q: There are two issues. One, we need an overall tree structure for assemblies, subassemblies, and parts. Two, what characteristics and details should be at each node of the tree.

Q: Typically, airframe drawings are owned by the U.S. Government, except in the case of McDonnell-Douglas. It is up to the government to decide who has access.

Q: If this information was used to build the F-18, can they give it to the F-22 team?

Q: A classification system implies a way of searching, but all people have their own ontologies.

Q: We need a translation between one ontology and another.

Q: Should there be one large database with a standardized format? Or should there be many databases with one information broker that interacts with the user, and is smart enough to interact with different databases.

Q: There are projects at Stanford University, University of Southern California Information Sciences Institute, and Microtheories working on an information broker.

Q: You could tell a central information broker what your representation is.

Q: Does a broker have a taxonomy in its head?

Q: There are two different problems: One, a taxonomy for artifacts and design so that we can share design data with our partners. This involves no ambiguity. It solves a design problem. Two, an information broker so that we can find out what suppliers can supply apart. This involves some ambiguity. It answers a resource discovery problem. These are two fundamentally different kinds of searching.

Q: Design at different levels of detail.

Q: If you've done searches of bearing manufacturer databases, you can put the design in your overall design.

Q: In the COTS database, put the minimum information. When you get a query from a customer, you get more information.

Q: Do we need an all-encompassing taxonomy, or is that not realistic?

Q: The primary goal of a central database is to match up buyers with suppliers.

Q: Isn't this what we already have? People doing inventory management have on-line catalogs of products. Many organizations do things this way, even though there are no standardized semantics.

Q: Standardized semantics would help small businesses, who don't have inventory management, more. Chinook was more than 60% supplied by outside vendors. The 777 was more than 50% supplied by outside vendors.

Q: Lockheed-Martin Orlando could say: "You want to do business with us? Put data in this database, here's the template."

Q: I hope all of Lockheed-Martin would do this, and I dream that all of DoD would.

Q: But you need to be able to get people who aren't already doing business with you.

Q: I speak in favor of one massive database. The information broker has its own ontology, and knows about other ontologies. It has a model of confidence of each database, and knows which supplier is possibly capable of supplying a part.

Q: The information broker should be capable of formulating catalog-specific queries, combining query results into a summary, and perhaps asking the user for more information.

Q: What if you have to completely design in-house, no OEM, no off-the-shelf? Could you use a lessons-learned/design-intent database? It would be internal to the company, and capture the experience of designers who retire or quit. The experience would include the process, the result, and the knowledge used to make the decision. What parts can be made in-house, and what parts can be outsourced.

Q: Find some way to extract value from completed designs, or at least as-you-go-along designs.

Q: NIST could come up with a template for a design diary, using knowledge-based engineering paradigms.

Q: Parametric description of a solid model expresses simple design intent.

Q: What's more complicated is: here is this bearing, here is why it was chosen, here are the characteristics that met particular requirements.

Q: This NIST document: for every design, you fill it out and save it. It captures design rationale.

Q: This is a difficult problem—Fund this research!

Q: Every organization and designer has her or his own way of doing design. What mechanism will fit in well?

Q: You have to support, say, stress analysis, material analysis, and a standard format to document the design.

Q: Capturing style of design for successful designers seems important. Parse the design itself, not just the processes.

Q: Style can be parsed out of the design. Decision cannot: what trade studies did you do? Why did you decide "no" on this? It could just be because a part was temporarily unavailable from the supplier.

Q: Suppose we had a Disciplined Design Diary. Would it work? Would a designer use it? Software engineers don't.

Q: If you asked, "Why did you pick that bearing?", the answer would be, "We had requirements, this is how we met them." Other things may not be as quantifiable, but this is a start.

Q: There exists a tool called the Multidisciplinary Engineering Collaborative Environment at Lockheed-Martin Palo Alto. It captures some design intent; it's an engineering notebook.

Q: I don't like STEP. It's not mature enough.

Q: We want to automatically put together commercial packaging problems.

Q: They're developing a commercial package at Carnegie Mellon University.

Q: There's an incredible DARPA-funded project at Massachusetts Institute of Technology. They have a two-dimensional mechanical design switch with two or three kinematic motors. The project is intended to capture design intent. It could divorce most of the geometry from the design intent and it could look at different configurations in an automated way. It automatically generated eight different designs that did the same thing. I want research like that to be funded. One might call it "design generation tools". It parsed a two-dimensional drawing to come up with design rationale, which included very little spatial information. It did not include the traditional notions of design parameters, but rather conceptual design intent.

Q: I use grammars (symbolic reasoning) to start from a functional representation. I automatically generate designs to make carts from Erector Sets. If there were a commonly agreed-upon set of functions that mechanical systems perform, this research would not be stalled. I need a general set, for a larger domain.

Q: Most of the information is not in design geometry. We understand that a motor is a motor, not a heater, even though it generates heat. Geometry is a summary, but there are other views of data.

Q: Production rules describe what is of interest to you.

Q: Start from intent. I want, for example, transformational motion.

Q: What the Massachusetts Institute of Technology researcher did was to use a sketch pad for creating the design, based on design intent, perhaps via a semantic net.

Q: The requirements for such a project include a fixed information packaging scheme that describes how information and functionality are modeled.

Q: There needs to be a standardization of design requirements. An external regulatory standardization, and internal standardization of processes, centers of excellence, and business practice.

Q: There was a requirement that cabin door don't implode under pressure. This implied that they open outward. Boeing thought that opening outward was a requirement, but it wasn't.

Q: It is important to divorce design intent and functionality from geometry.

Q: The Jami Shah paradigm is to have multiple variants of a particular design, and to map between them based on design primitives; how they are instantiated, and what the relations between them are.

Q: We need to get concrete stuff done with STEP. Turn up the heat on them, maybe give them a deadline.

Q: We have to transfer from CAD system A, say ProE, to CAD system B, say I-DEAS Master Series. I wish you could transfer intelligent geometry: features and the relationship between them.

Q: I have eight applications. If I am to automate them, I need a standard to drive them all.

Q: Vendors have nothing to work with. We need a standard for portable capture and exchange of feature information.

Q: That is, we need a completed, coherent, complete standard that covers all bases. Some APs are still in draft stages. They need to be completed.

Q: Industry must push vendors to use STEP.

Q: DARPA hired a Cambridge firm to answer the question: why have the circuit board designers performed so well? Their design process is ten times faster! The answer: they do two-dimensional or two-and-a-half-dimensional design. Mechanical design is three-dimensional. We need more research in three-dimensional packaging to produce commercial codes based on design intent.

Mr. Barbieri: Now we need to pull these together into a purpose, an approach, and some problems. Our outline:

1. There should be a universal National Material Database that includes standard information on any type of material that meets minimum testing procedures set by NIST. This information should include a "pedigree": where the data came from and who did the testing. This information should also describe how the material is consistent with ASTM standards.
2. There should be a universal COTS (Commercial Off-The-Shelf) Database that allows multilevel querying, either broad or specific. There should be a NIST database with minimal information, and NIST standards, including normalized data. This database should enable user interaction with suppliers. There should be an information broker, an alternative to a grand unified database that goes to distinct databases.
3. There should be a standard for design intent databases, or a set of ISO guidelines. This would be an internal database that captured design experience: knowledge, design decision trail, design standards—this is the format you use to capture the information. NIST should set up the requirements but not the database itself. The requirements have the same intent: review the processes, trade studies. Call it, perhaps, the "Design Documentation System".

4. There should be a National Research Database, with a set of guidelines for documenting government funded research. This should all be accessible from a NIST web site, with HTML able to be edited by the people doing the research. This database should also contains requests for research, funded by government, industry, and universities. All of this should include point-of-contact.
5. STEP has been too little, too late—they've been working on it for ten years. We want more of it sooner. STEP currently helps with about 10% of design.
6. Research to produce commercial products for 3-D packaging.

Breakout Session #2—Database/Knowledge-Base Frameworks

**Discussions summarized by Dr. Ronald Giachetti (giachett@cme.nist.gov)
and Dr. Mark Schwabacher (schwabac@cme.nist.gov)**
(discussions done without slides)

Participants:

- Michael Case
- Alexei Elinson
- Ronald Giachetti
- Sandie Kappes
- Pradeep Khosla (chair)
- Tim Malueg
- Sham Navathe
- Mark Schwabacher
- Sharad Singh
- Eric Stephens
- Eswaran Subrahmanian
- Susan Urban

The group facilitator, Pradeep Khosla, led us in a brainstorming exercise to first identify what the elements of a design repository are. These categories evolved during the session. For each category we attempted to determine what the current state-of-the-practice is, what laboratory research is being performed, and what is desired or needed for further design repositories. The three categories are:

1. Functionality
2. Content, structure, and capture of design knowledge
3. Use and deployment

These notes are intended to capture what was said with as little interpretation as possible.

FUNCTIONALITY:

- Today
    - Today we capture files - Eric
    - Capture design artifacts - Michael
    - Library of pre-drawn geometry (geometric libraries) such as Bentley or AutoCAD
    - Printed documents capturing requirements - Tim
    - Isolated information - Eswaran
    - File management systems - Susan
    - Design data management, configuration control
    - Physical automated design notebooks - Susan
    - No means of traversing repository or of checking -Alexei
    - Design prescriptions available - Tim

- Links from CAD to relational databases for cost estimating - Michael
- Current Research In Laboratory
  - Object-oriented databases (missile design) and assembly modeling - Tim
  - Design artifacts repository API (interface to existing tools) - Sandie and Michael
  - Life-cycle knowledge through constraints - Ronald
  - Design process capture from specifications to design "as done" -Eswaran
  - Measure of design similarity (prismatic approaches) - Alexei
  - Web-based hypertext tools (intranet too) linking design documents - Susan
  - Limited automation of relationships or constraints - Tim
  - Functional requirement specification and rational capture systems - Michael
- Future Needs and Desires
  - Standard interface - Michael
  - Standard mechanism to implement repository (STEP) - Sandie
  - Draw functionality from existing database management systems, plus more. -Sham
  - Designers to dynamically describe what they are doing in an unobtrusive manner - Susan
  - Uniform naming scheme to support queries - Alexei
  - Possible integration of existing design ontologies - Sham
  - Private and public repositories in same framework - Eswaran
  - Behavior representations at multiple levels of fidelity. -Eric
  - Specification and automatic enforcement of constraints -Sham
  - Event notification. - Michael
  - Uncertainty management (tolerances) - Eric
  - Short and long transactions (long defined as do on portable with no network) - Eric
  - Security and privacy issues addressed - Sharad
  - UNIX type security read/write/etc. is sufficient ? - Eric
  - Locking on attribute values, not just files - Sham

CONTENT, ITS STRUCTURE, AND ITS CAPTURE:

- Today
  - Geometric files - Tim
  - Text files
  - Hypertext - Michael
  - Point-to-point translators for the above 3 file types
  - Process in the form of work instructions - Eric
  - Content of MIME files: specs, requirements, bill of materials, test data, images, process descriptions, work flow
  - Content of databases: materials, part numbers, CAD, manufacturing resource planning (MRP), inventory, geometry, spatial data
  - Hand populated - Pradeep
  - Inflexible process modeling tools - Susan

- For example IDEF0, ProcessWeaver, BPWin
- Static 2D Structures
- Shared Relational Databases inferencing/explaining

- Future
  - Analyzable process description/models and workflows functional representation - Susan
  - Sharable design objects - Sandie
  - Automated means of data capture, e.g., optical character recognition - Sharad
  - Requirements accountably linked to product and process attributes - Eric
  - Make explicit the knowledge that is implicitly stored in computer programs - Mark
  - Multimodal version control management - Eswaran
  - Linking process description and design artifacts - Susan
  - Economic and cost data - Sham
  - Multiple alternatives for designs - Michael
  - Rationale for design decisions - Susan
  - Inter- and intra-project repositories/dynamic topologies - Eswaran
  - Ontologies for linking content of different databases - Susan
  - Self-describing data objects - Sham
  - Use of object request brokers (ORB) for communication between distributed data sources and tools
  - Dynamically extensible schemes
  - Active database concepts (event/condition/action rules)
  - Interoperability (between heterogeneous, multimodal database management systems)

USE AND DEPLOYMENT:

- Today
  - Expensive discovery of already existing information - Michael
  - Manual processes - Tim
  - Licensing of CAD tools is author only - Eric
  - Isolated and ad-hoc implementation of repositories with incompatible interfaces - Sharad
  - Poor human-systems interface - Pradeep
- Future
  - Ease of use and client-independent human-system interface (HSI) - Pradeep
  - Tools viewed as service rather than resource - Eric
  - Scaleable in users and repositories without performance degradation - Pradeep
  - Standards for federation of repositories - Michael
  - Methods for reuse of design histories - Susan
  - Methodology for development and deployment of repositories - Eswaran
  - Navigation across repositories

Breakout Session #3—Standards for Information Modeling/Knowledge Representation

Discussions summarized by Dr. PVM Rao (pvmrao@cme.nist.gov)
(discussions done without slides)

Participants:

- Mike Benoist
- Scott Chase
- Kim Cobb
- Edward Harter
- John Josephson
- Linda Lawrie
- Doug Peter
- PVM Rao
- Ram Sriram (chair)
- Simon Szykman
- David Thompson
- Philip Tsung
- Peter Will

Most of the participants pointed out that current industrial practice for product design is centered around commercial CAD systems and not product data standards. In the electro-mechanical product domain the standard of primary interest is ISO 10303 or STEP. It was pointed out that use of STEP, particularly AP (application protocol) 203 is presently restricted to exchange of geometric data from one CAD system to another. In practice, it is still not being used for the purpose of common product data sharing. There were many reasons pointed out, which are listed below.

- Representing constraints, parametric information, features, tolerance information etc. in the present version of STEP is still not possible.
- Reliable STEP translators for various CAD systems are not yet available. Existing ones often result in the loss of information. One of the issues mentioned in this regard is the problem of geometric precision.
- Tools for representing a product as a system of components, representing product assemblies, and representing key components in the product are not available in either STEP or STEP translators.
- Strict deadlines, together with availability of some reliable translators from some CAD systems to others, have allowed people to avoid the use of standards.
- Addressing integration issues with downstream applications (DFx) is difficult at present. Specific mention was made of representing process data (manufacturing) and computational fluid dynamics data (analysis) in design.
- There is a lack of representations for measurement data, such as that obtained from coordinate measuring machine (CMM), and a mechanism for relating this data to design process and design data.

In summary, most of the people felt the real need for speeding the process of STEP development to accommodate all the life cycle issues that relate to design. The members of the group also discussed the present practices of using part libraries and material libraries for product design. The importance of standard libraries for design retrieval and redesign purpose was felt by most of the members. The upcoming standard ISO 13584 or P-Lib was mentioned by one of the members. Unavailability of an established standard for representing part libraries forced people to re-design the same component rather than extracting the same from previous designs. Many members also indicated a need for fast retrieval systems and retrieval systems based on function.

Members felt the need for reliable tools and standards for representing product function, product behavior and conceptual geometries for design. Another interesting aspect of the discussion was the need for tools linking conceptual design with detailed design.

Representing design information of electronic products is another subject that was discussed. It was pointed out that the problems of representing design information in this domain are less difficult due to simpler product geometries as well as because of strong coupling between product form and function.

Information/knowledge representations for architectural design, and the need for standards in this area were mentioned. Some of the standards initiatives undertaken in this direction were discussed in some detail.

Rapid Design Exploration and Optimization (RaDEO)

Mr. Kevin Lyons, DARPA (klyons@darpa.mil)
(14 slides start after page 330)

Mr. Lyons started by noting that the name of the MADE program had been changed to RaDEO (Rapid Design Exploration and Optimization). The concept of a design repository is a key issue for RaDEO. The marketplace is changing; teaming between companies is encouraged or required to remain competitive. We are getting more computer-literate suppliers, but they have incompatible systems. Merging corporations can't communicate.

On the "Program Goal" slide, all of the "ability to..." points were discussed in various breakout sessions: get a better way to look at current databases, identify the problem, and pull out solutions. The "Realization of a Seeker System" slide describes how a conceptual optics design finds its way into different applications. The "System-Centric Design" slide expresses the idea: "I am a gimbal designer and all else focuses around there," or "I am an airfoil designer," or something else.

RaDEO is structured around 27 contractors in four problem areas. One focus area is Design Exploration and Advanced Design Representation. We want to store, retrieve, and index all different types of design information. We need to do a better job of structuring the information to allow intelligent queries. Another focus area is Multi-Disciplinary Optimization and Simulation. This involves ways to balance trade-off analysis: weighting, schemes based on marketplace concepts, and other ways. A third focus area is Integration Frameworks. This is "bare-bones" work: very early conceptual frameworks. The final focus area is the designers interface.

Information is the power of a design team. The more information you have, the more power you have. There is a lot of historical information that you cannot ignore in the present enterprise. RaDEO has several efforts involved with capturing design intent in various ways and aspects. STEP is of some help, but not enough.

Intelligent querying involves the retrieval of information. Nobody lacks information, they only lack a way of accessing it. We need to narrow the scope of the information to present it to the engineer in a meaningful way. We need ontologies and taxonomies. We want to decompose systems into common structures. We hope this fundamental work will have huge benefits.

We are focusing on real demos with DoD impact. Our customer sets include missiles, aircraft, and helicopters. We form contractors into demo groups, which we pull from all categories, and they take the systems to the next level.

A workshop participant interrupted Mr. Lyons to ask how, with 27 separate contractors, did he as program manager pull it all together. Mr. Lyons answered that he works directly out of the program office with the aid of a number of agents in other government agencies that help manage individual projects. Also, demo groups take ownership of the systems.

Mr. Lyons concluded by directing participants to the DARPA web page, for solicitations under the Information Systems Office for High-Performance Knowledge Bases (HPKB), BAA (Broad Agency Announcement) 96-43.

High-Performance Knowledge Bases (HPKB)

Dr. David Gunning, DARPA (dgunning@darpa.mil)
(16 slides start after page 345)

Dr. Gunning began his talk by saying that there were historical reasons for the High-Performance Knowledge Bases (HPKB) name. The HPKB goal is very specific and not necessarily achievable. Today you build a knowledge base with two thousand to ten thousand axioms, rules, and frames, and it takes great manual effort. HPKB wants to move to ten thousand to a hundred thousand axioms, rules, and frames, giving in-depth coverage of the entire domain.

What is a knowledge base? It is something more declarative: axioms, or statements in some declarative language. Domain theories could be formulated, for example, in predicate calculus. But this is just an example, it is not a necessity. We want to be able to compose and check knowledge bases. There are libraries of reusable information, especially libraries of domain knowledge.

The envisioned knowledge base development is a three-step process. First, in Foundation Building, one needs tools to build and reuse libraries of foundation knowledge. One also needs tools to compose and edit these libraries, to quickly compose a knowledge-base framework. Second, in Knowledge Acquisition, it is best to use machine learning or natural language processing to fill up the knowledge base. Then, in Efficient Problem Solving, do more efficient reasoning, use inference methods, or transform knowledge into compiled modules. The BAA is looking for technology for each of the three steps, and for integration. The four-year goal is complete systems.

Foundation Building involves very large lexicons, such as WordNet. These lexicons should be broad, not deep. There should be a library of in-depth theories of concepts such as time and space. These may be present in different ways. There should be a library of problem solving strategies. The user should be able to quickly compose ontologies and theories into a unified tool.

In Knowledge Acquisition, we want to automatically generate a user interface so that a non-knowledge engineer can enter the knowledge. We want to import knowledge from lexicons and dictionaries, and extract information from the text with natural language processing or machine learning. These tools enable domain experts to collaborate to extract and define domain knowledge.

Efficient Problem-Solving technology is technology we hope to build and integrate.

Our approach is shown on the "Development Approach" slide. The program should go through these steps in some sequence. In step 5, products should become DARPA application products. We hope that HPKBs will have general use for many domains. But for now, they are largely intended for military use: battlefield awareness, command and control, and logistics. People in DARPA are interested in design domains, but are bent to military applications.

DARPA is looking for people to define, manage, and maintain challenge problems. First, they ask developers to build knowledge bases for challenge problems, and see how quickly they can do that. Second, they need some more broad-but-shallow challenge problems that are more generally applicable. The Challenge Problem Process starts with the receipt of terminology, ontology, and expertise about the domain. Then, you have one month to build the knowledge base. DARPA is soliciting approaches to define challenge problems, as well as technology itself. Challenge problems should not allow a shortcut answer.

BAA 96-43 is calling for proposals in two areas. 5 December 1996 was the proposal deadline, and Dr. Gunning was looking forward to an interesting set of proposals. He felt that it was time to look at knowledge base technology again.

A brief question-and-answer period followed Dr. Gunning's talk. Questioners are identified by number.

Q1: Is the end user aware of using a knowledge base?

A: Not necessarily.

Q1: Is it more ubiquitous?

A: Yes.

Q1: Obviously you are using an expert system, should you make the user less aware of using it?

A: Yes. Part of efficient problem solving is packaging data to fit it into another application.

Q1: My background is in control theory. I spent significant effort in a knowledge base for real-time control with commercial neural network and fuzzy logic systems. The real problem is integrating lower-level components into larger systems. The problem with operations in military is that high-level logic has seamless, useful integration. The declarative approach doesn't mention the evolution component, or composability into larger models. It describes the real world as a set of rules.

A: Agreed.

Q1: Are you looking for evolution as well as representation?

A: Yes. Integrate knowledge at different levels of abstraction. Declarative languages didn't work well before, but after new technology, it's time to go back and investigate.

Q2: Most of the areas you described are engineering and manufacturing. How would you assess a proposal that used a generic knowledge base, with specific application to engineering and manufacturing?

A: That's a tough question. It's a good idea, but we will have to try the tools against military challenge problems. Before DARPA reorganized, the original version of the program included both design and battlefield, and the design went to another office. At least 80% of the effort must be on the battlefield.

How Does a Standard Become a Standard?

Ms. Sharon Kemmerer, NIST (kemmerer@cme.nist.gov)
(3 slides start after page 362)

Dr. Sriram introduced Ms. Kemmerer as a STEP expert who would give a presentation on how a standard becomes a standard. Ms. Kemmerer started by describing ISO, the International Organization for Standardization. ISO is in charge of all non-electrotechnical standards. It had 85 member countries at the end of 1995. The IEC (International Electrotechnical Commission) is in charge of all electrotechnical standards. It had 50 member countries at the end of 1995. The host organizations for both ISO and IEC reside in Geneva. You can check out their web sites for more information.

Say that a country comes forward with a new work item. The active liaison, or a technical organization, needs to come forward with a standard. They need at least two pages to describe the impact, or a full standard if it needs to be fast-tracked through the international community.

A new work item (NWI) proceeds through the following stages:

- Five of nineteen "P members"—actively involved countries—within ISO TC184/SC4 must actively participate in the development. Ten of nineteen must approve the statement of impact. There is a three-month ballot cycle to subcommittee members, which usually generates some comments.
- Upon approval of NWI working draft is created. Most ISO standards run 50-150 pages. Most Subcommittee 4 (SC4) standards are longer, running 1500-2000 pages.
- Next comes the Committee Draft (CD) cycle. There is a four-month cycle to subcommittee members, which usually generates many more comments.
- Then, there is a Draft International Standard (DIS), where members make sure that recommended technical changes have been implemented. There is a five-month ballot cycle to all ISO members. A two-thirds majority of all voting members must approve.
- Next, there is a Final Draft International Standard (FDIS). No technical comments are made at this point—only editorial comments. Finally, after a two-month ballot cycle to all ISO members, it becomes an International Standard.

A question-and-answer period followed Ms. Kemmerer 's talk. Questioners are identified by number.

Q1: I have a very specific question: did AP 209 just become DIS?

A: It is currently in the CD stage. They are incorporating comments. It may have already been approved to become DIS.

Q1: Do you have to develop test cases?

A: A resolution specific to SC4 says that we must have proof of validation. Also, an abstract test suite must support DIS in SC4. STEP is one of four ISO standards that SC4 is working on. The others are a standard for parametrics, a suite of standards for parts libraries, and a suite of standards for manufacturing management data.

Q2: DoD requires us to cut our life-cycle costs in half. Can we pass that on to you?

A: Standards will help.

Q2: I agree, but standards aren't doing enough.

A: We're working on application protocols.

Q2: It seems as if you have no goals for a particular date.

A: Our goals are to move from an NWI to a First Working Draft within six months, then to a CD within eighteen months, and then to an FDIS within three years. But our goals are slipping in fifty of our programs. Standards can be subject to cancellation or reconfirmation.

Q3: This morning, the FDA (Food and Drug Administration) said that you cannot put the word "lowfat" on 2% milk, you must instead use "reduced fat." What is your current policy on supporting standards? How can you help vendors play by the rules?

A: This is a voluntary standards process. In most of the world, standards are driven and funded by governments. The U.S. is mostly industry-driven and industry-funded.

Q3: Is there ongoing activity for independent testers to verify performance?

A: Yes. NIST offers a beta testing suite for AP 203. ProSTEP of Germany has an interoperability testing program.

Q4: So you're not analogous to the FDA?

A: Unlike, for example, safety standards, the role of the government for product data standards is not that of a regulatory agency but more of a neutral facilitator for standards development.

Presentation: Breakout Session #1—Industry Perspective

Dr. Michael Barbieri, Lockheed Martin (MPBarbieri@lmtas.lmco.com)
(3 slides start after page 366)

Industry needs:

- NIST Material Database with data summary of materials.
- COTS Database with the ability to send email to suppliers; NIST should supply the format.
- Information/Design Broker to handle different terminology for the same things, and break down barriers.
- Design intent/diary format, an "ISO" format for saving information.
- A national research database. There is much duplicated research. There needs to be a place for researchers to input data, and find out about Points of Contact.
- Speed up STEP.
- Commercialization of three-dimensional packaging.

Presentation: Breakout Session #2—Database/Knowledge-Base Frameworks

Dr. Pradeep Khosla, Carnegie Mellon University (pkk@cs.cmu.edu)
(presentation given without slides)

Dr. Khosla said that one positive note was that most of the topics from the breakout session were discussed by Dr. Gunning in a different context. The fundamental question: What is a design repository? It should have functionality, content, structure, use, and deployment.

Today, we use engineering practice, and research laboratory practice. Tomorrow (in five to ten years), where would we like to be?—mostly from a research point of view, although industry is involved in use and development.

For example, today we have file capture. Tomorrow, we would like to have integration with design repositories. One issue is integrating multiple repositories of the same type. Another is a human-system interface to databases. We need to populate and operate databases efficiently.

There was a National Academy of Sciences report on Information Technology in Manufacturing—for information, contact Dr. Will (will@isi.edu). This is a resource worth looking at.

Presentation: Breakout Session #3—Standards for Information Modeling/Knowledge Representation

Dr. Simon Szykman (szykman@cme.nist.gov)

(3 slides start after page 369)

Current practices involve some use of standards in industry. Sometimes there are translator problems. One issue: there is no data sharing yet. There are many database issues, and there are barriers to use of standards, including slow STEP development.

There are a number of needs and recommendations. Immediate needs involve improving standards for common design and design-related activities in the short term. Priorities include DFM (Design For Manufacture) standards to represent information about manufacturing in the context of both design and manufacturing processes.

There are geometric precision problems, in that there are mathematical precision differences across CAD systems. Information issues include standards to help with conceptual design, analysis, DFM and DFx. Design retrieval and redesign is another concern; today we often redesign existing parts because it's easier than looking up an old design.

There are long-term issues as well. We need standards and integration for conceptual designs. We need to represent not just geometry, and not just individual parts. There are multidisciplinary issues that become important with multiple disciplines, but that are not commonly addressed because they don't arise in a single discipline. Ontologies are very important as well. In electronics, name, form and functionality are more strongly linked than in mechanical design and manufacturing. Finally, there is a new set of issues associated with the use of the Internet as a medium for communication.

Presentation Slides

Welcome to NIST

Dr. Richard H. F. Jackson, NIST (jackson@cme.nist.gov)

The summary for this presentation can be found on page 11
20 slides follow

Manufactuirng Engineering at the National Institute of Standards and Technology

Richard H.F. Jackson, Director
Manufacturing Engineering Laboratory
National Institute of Standards and Technology
http://www.nist.gov/mel/

MEL NIST

NIST Functions

- **Assist industry in the development of technology and procedures needed to**
  - improve quality
  - modernize manufacturing processes,
  - ensure product reliability, manufacturability, functionality, and cost-effectiveness, and
  - facilitate the more rapid commercialization...of products based on new scientific discoveries.
- **Develop, maintain national standards of measurement.**
- **Assure international compatibility of national measurement standards.**
- **Advise industry and government on scientific and technical problems.**
- **...**

NIST Resources

Fiscal Year 1996

- 3320 Employees
- About 1250 Guest Researchers
- $259M Appropriated Laboratory Funding
- $739M Operating Budget
- $60M for Construction of Research Facilities

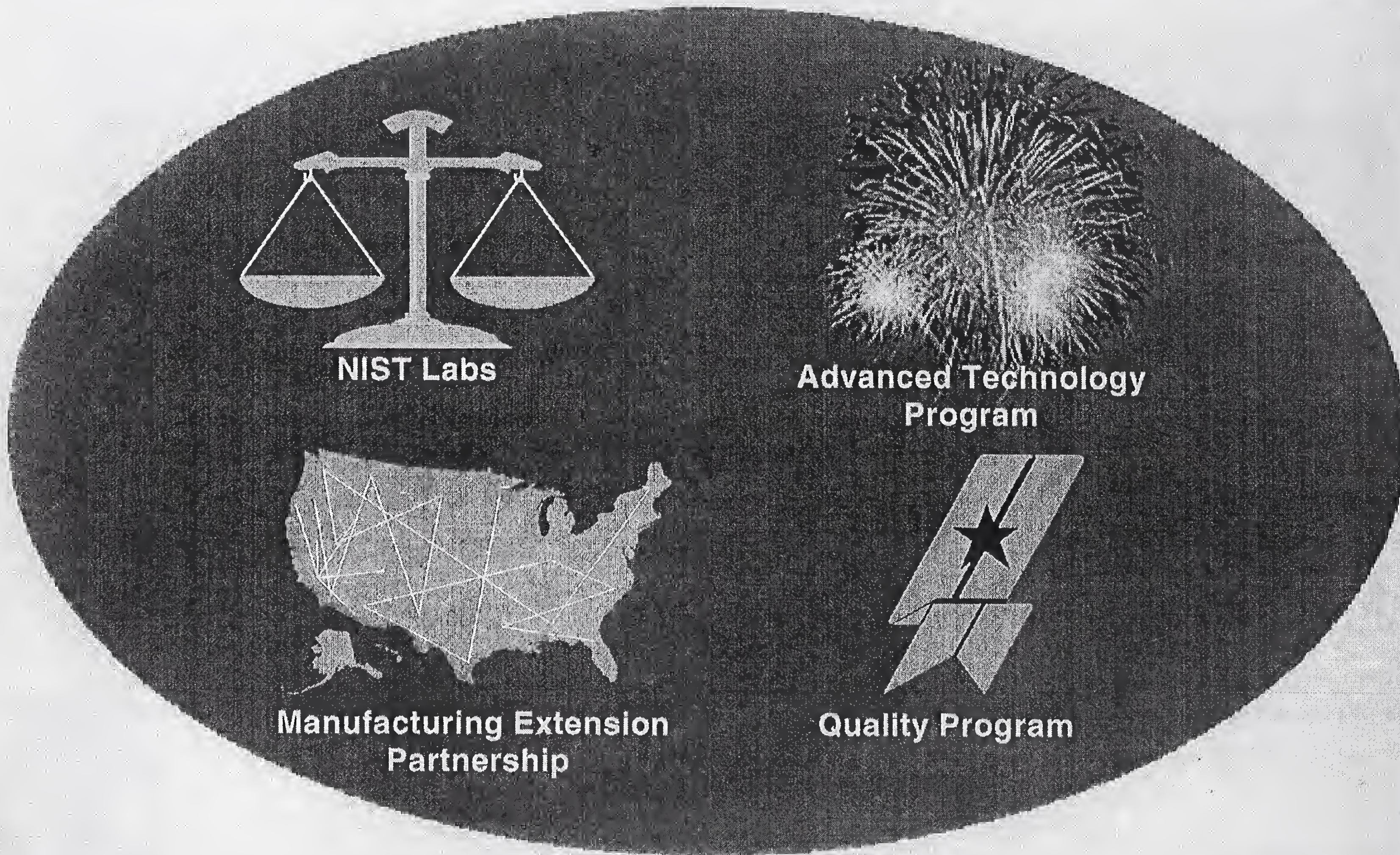
MEL NIST

NIST Mission

To promote U.S. economic growth by working with industry to develop and apply technology, measurements, and standards.

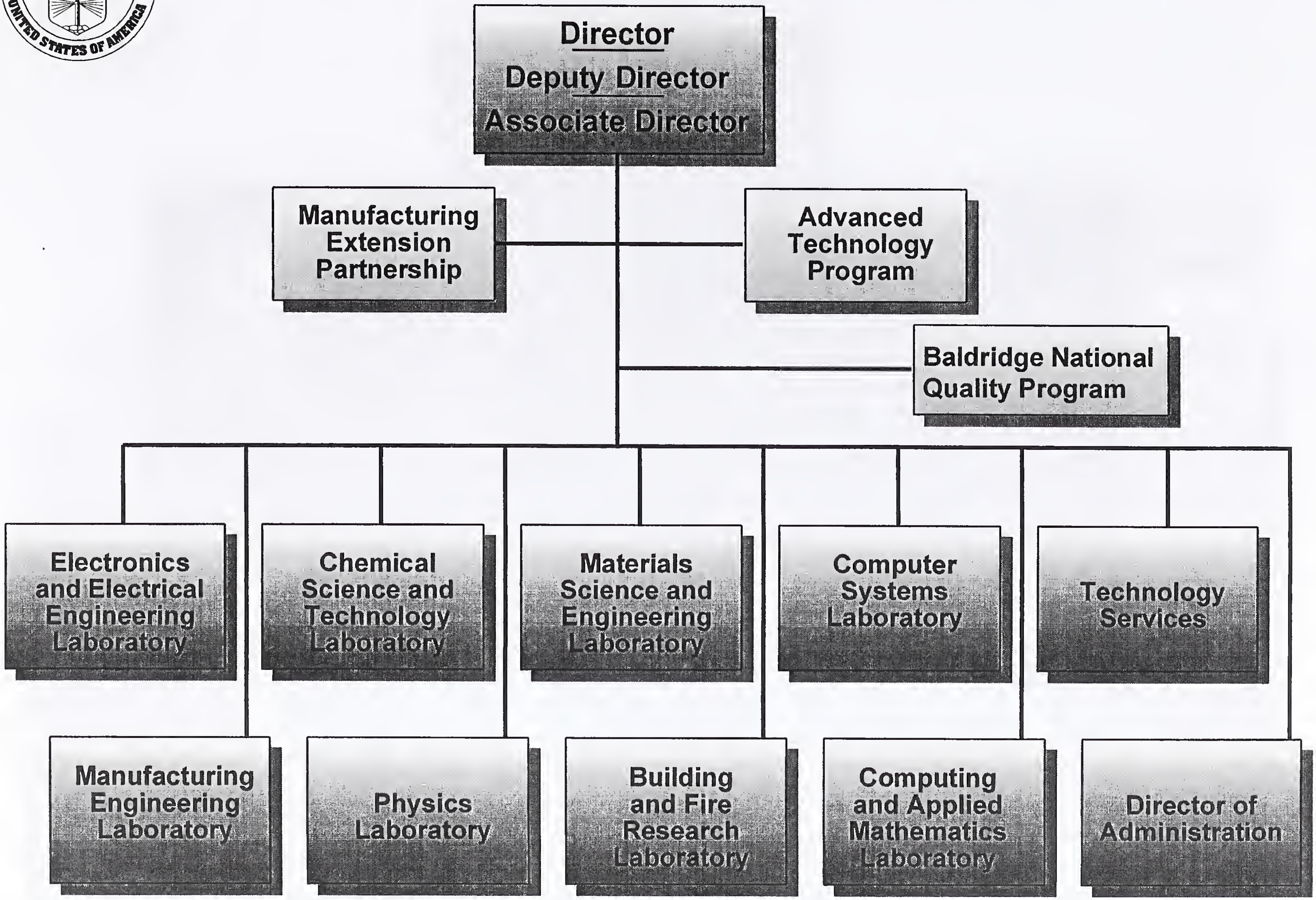
MEL NIST

National Institute of Standards and Technology

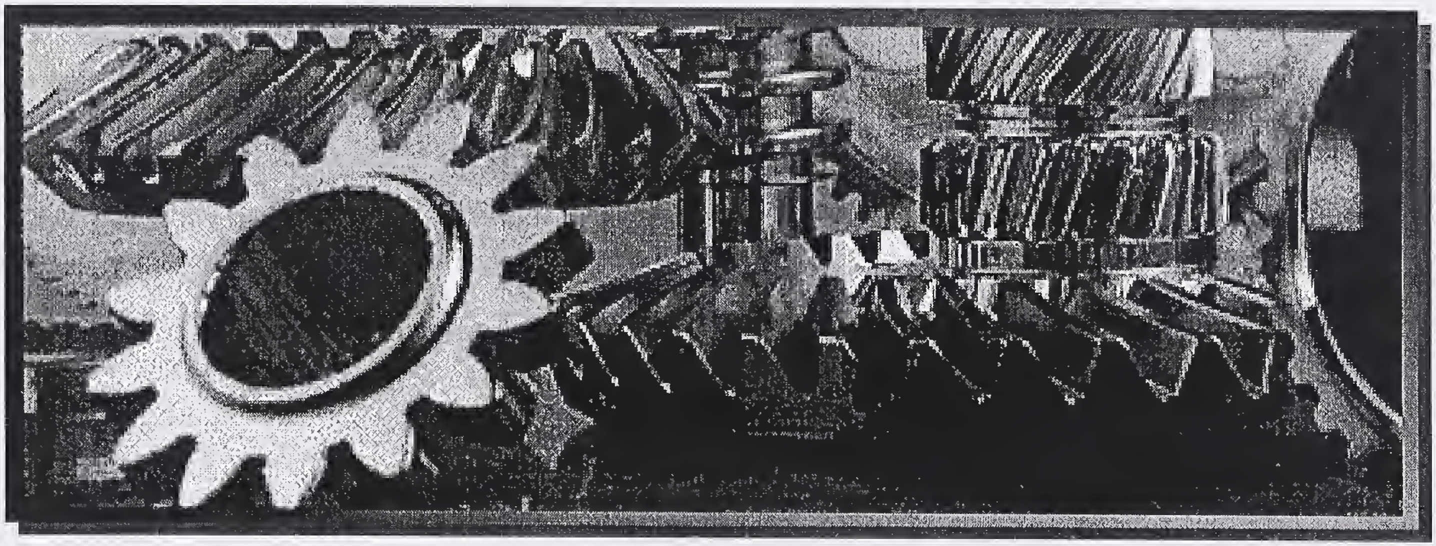


MEL NIST

National Institute of Standards and Technology



Manufacturing Engineering



Laboratory

Manufacturing Engineering Laboratory

Director
Richard H.F. Jackson

Deputy Director
Mark E. Luce

OFFICE OF MANUFACTURING PROGRAMS
Merrill M. Hessel, Chief

PRECISION ENGINEERING DIVISION

Dennis A. Swyt, Chief

Large Scale Coordinate Metrology
Steve Phillips

Mid-Scale and Complex Form Metrology
Howard Harary

Surface and Microform Metrology
Clayton Teague

AUTOMATED PRODUCTION DIVISION

Donald Blomquist, Chief

Acoustics, Mass & Vibration
Don Eitzen

Ultrasonic Standards
Gerry Blessing

Sensor Systems
Alkan Donmez

Sensor Integration
Kang Lee

Force
Simone Yaniv

INTELLIGENT SYSTEMS DIVISION

James S. Albus, Chief

Performance Measures
Ken Goodwin

Intelligent Controls
Fred Proctor

Systems Integration
Maris Juberts

Sensory Intelligence
Marty Herman

Unmanned Systems
Elena Messina

MANUFACTURING SYSTEMS INTEGRATION DIVISION

Howard Bloom, Chief

Nat'l PDES Testbed & Apparel Tech. Office
Howard Bloom (Acting)

SIMA Office
Jim Fowler

Applied Systems
Neil Christopher

Mfg. Standards Methodology
Mary Mitchell

Mfg. Collaboration Tech.
Steve Ray

Eng. Design Tech.
Ram Sriram

Mfg. Systems Eng.
Chuck McLean

FABRICATION TECHNOLOGY DIVISION

Richard L. Rhorer, Chief

Support Activities
Richard L. Rhorer (Acting)

Central Shops
Dana Strawbridge

Special Shops
John Evans

MEL Role

Serve the ***whole*** manufacturing enterprise, ***each*** of the manufacturing sectors, and the **range** of manufacturing technologies, with special focus on the mechanical manufacturing industry.

Provide ***physical*** and ***informational*** standards.

Disseminate the dimensional and mechanical standards of length mass, vibration, force, acoustics, and ultrasonics.

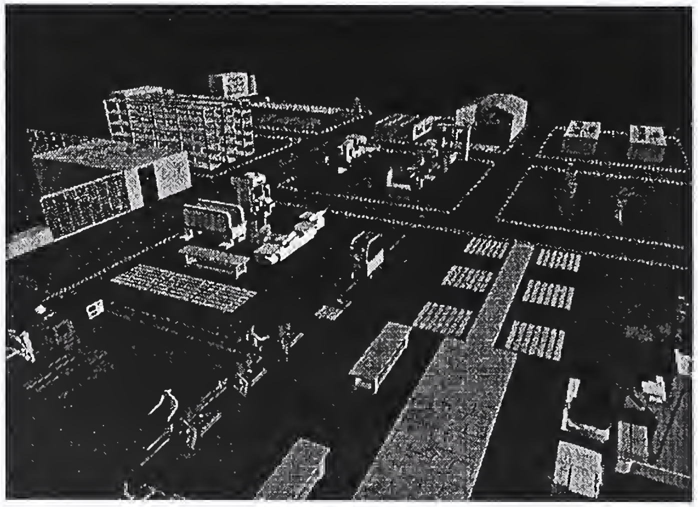
Realize the SI units of length and mass.

Provide ***fabrication*** services to NIST.

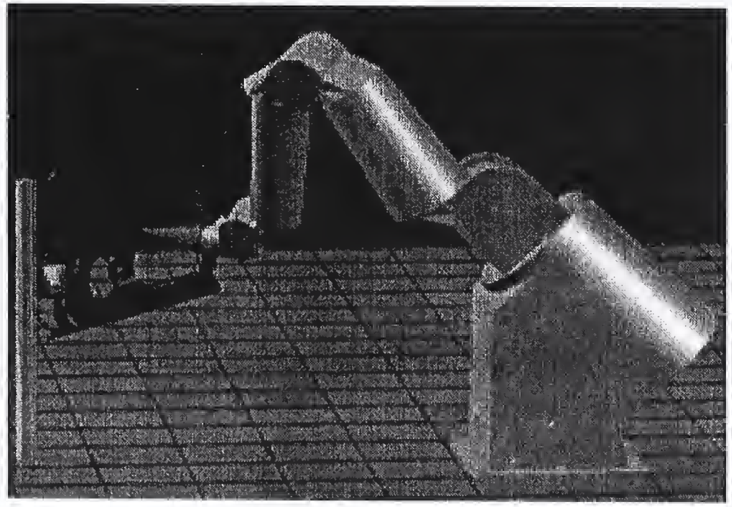
MEL NIST

Programmatic Thrusts

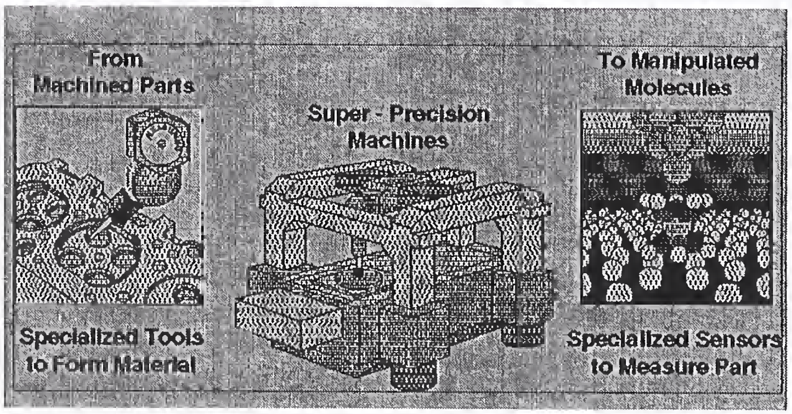
Manufacturing Systems Integration



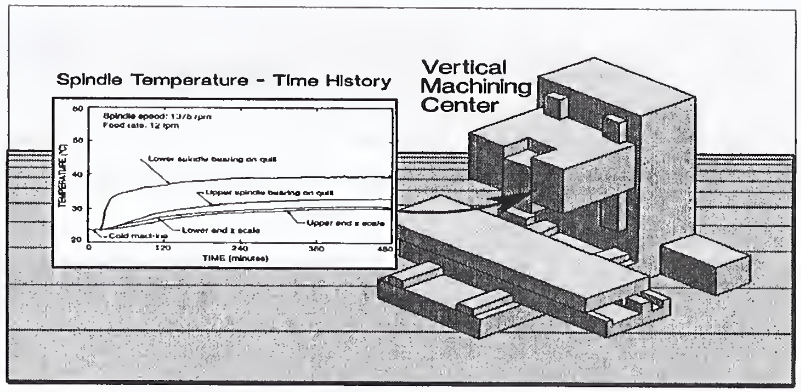
Intelligent Machines



Manufacturing Metrology



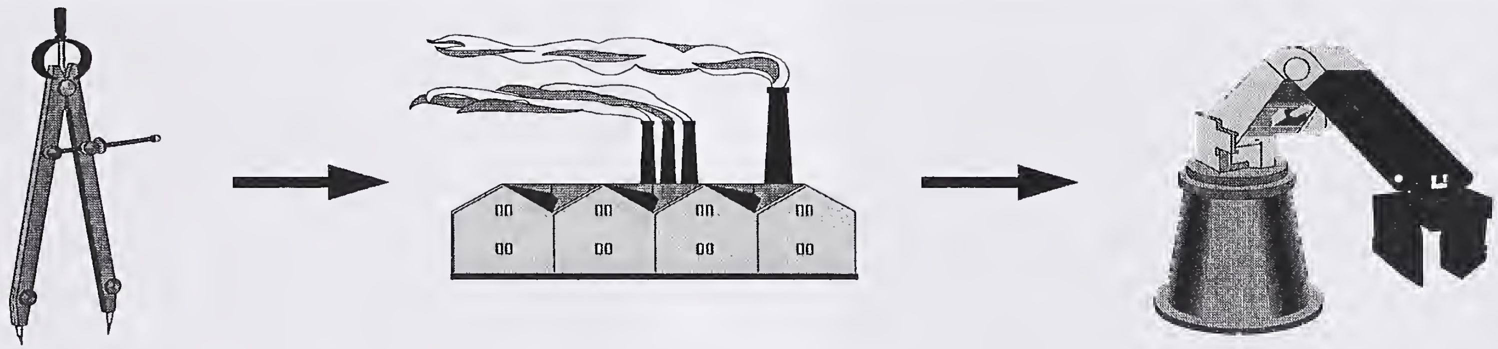
Manufacturing Processes & Equipment



MEL NIST

Trends in Manufacturing

Craft → *Mass Production* → *Automation*



Process improvements driven by a desire to increase marketshare by improving *quality*, reducing *cost* and decreasing *time* to market.

MEL NIST

The Automated Manufacturing Research Facility

- A research test bed for the factory of the future
- A U.S. Navy Center of Excellence

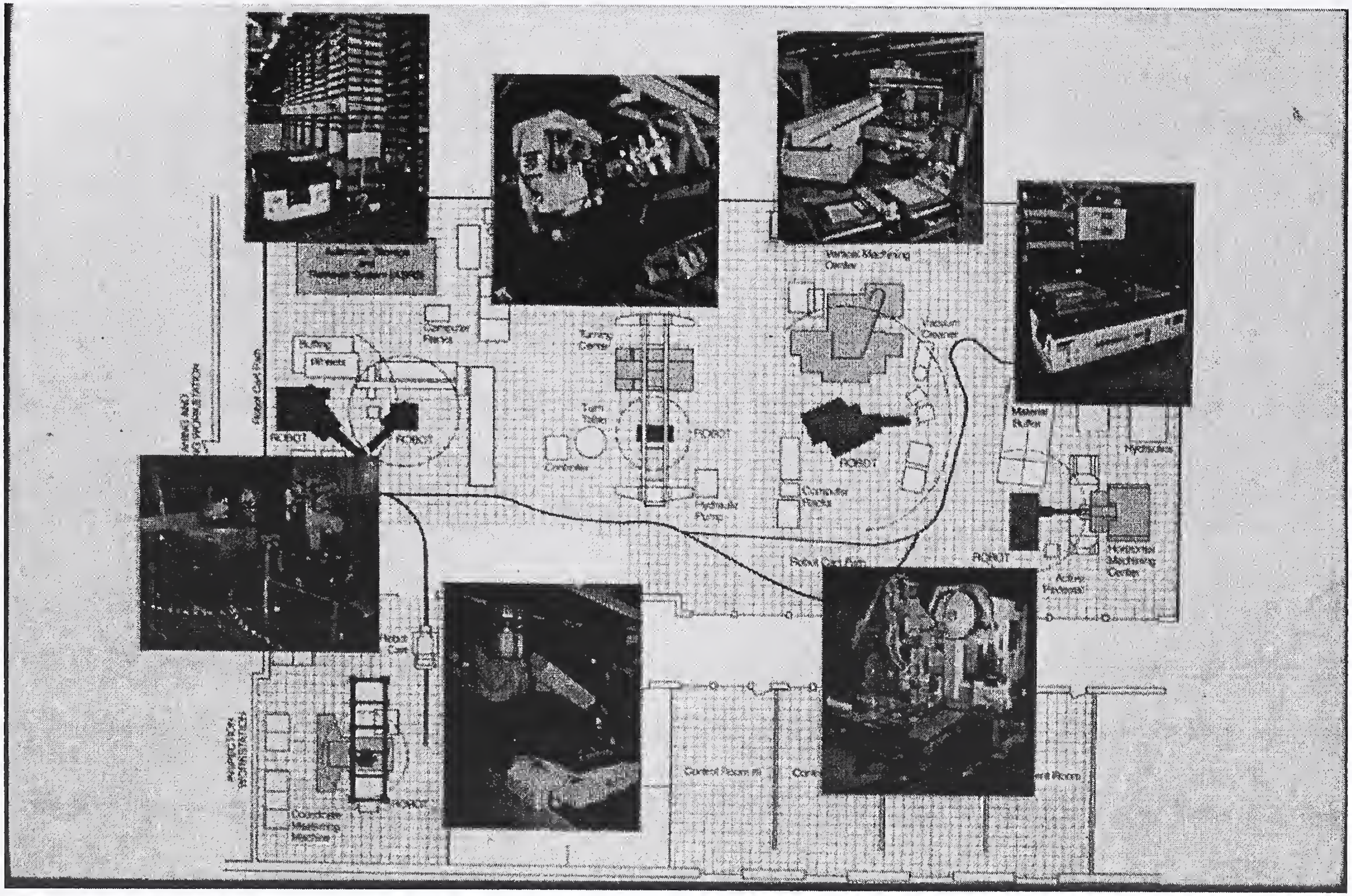
MEL NIST

AMRF Project

- Build a test bed flexible manufacturing system
- Support manufacturing systems research by NIST, Academia, Industry, and other agencies
- Conduct continuing studies of interface standards
- Conduct continuing studies of advanced metrology
- Transfer technology to American Industry

MEL NIST

AMRF Floorplan



MEL NIST

AMRF Accomplishments

- 21 Standards
- 14 Products
- 17 Patents
- 11 Subsystems
- 80 Research associates from 50 firms
- 15 Theses from 40 academic connections
- 9 OA projects totaling more than $10M
- Donations totaling more than $20M

MEL NIST

What's Next in Manufacturing ?

Flexible

Supplier Integration

Distributed

Adaptive

Reconfigurable

Virtual

Global

Customer Responsiveness

Niche Markets

Rapid Response

Intelligent

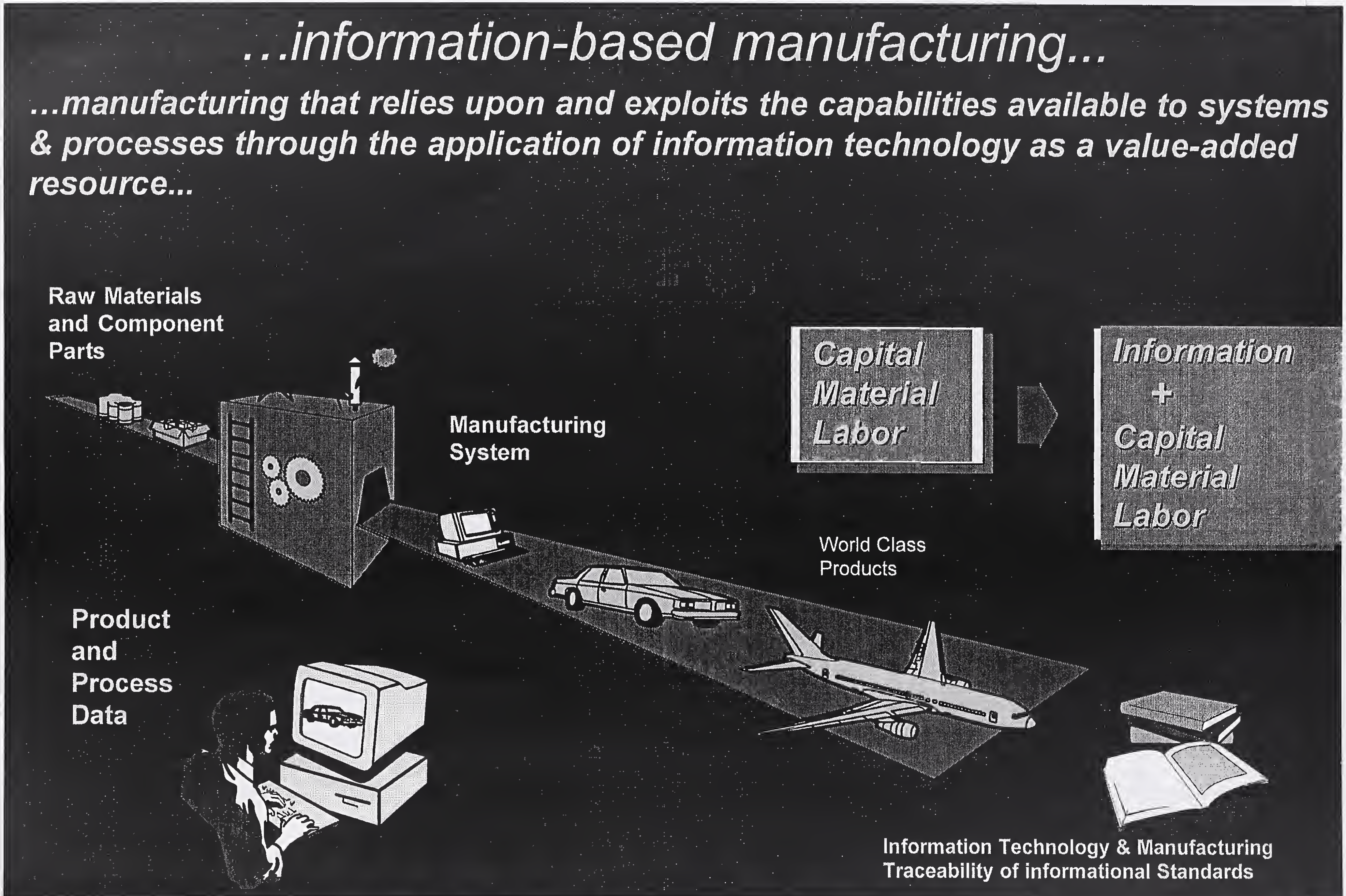
Lean / Agile

Customized Products

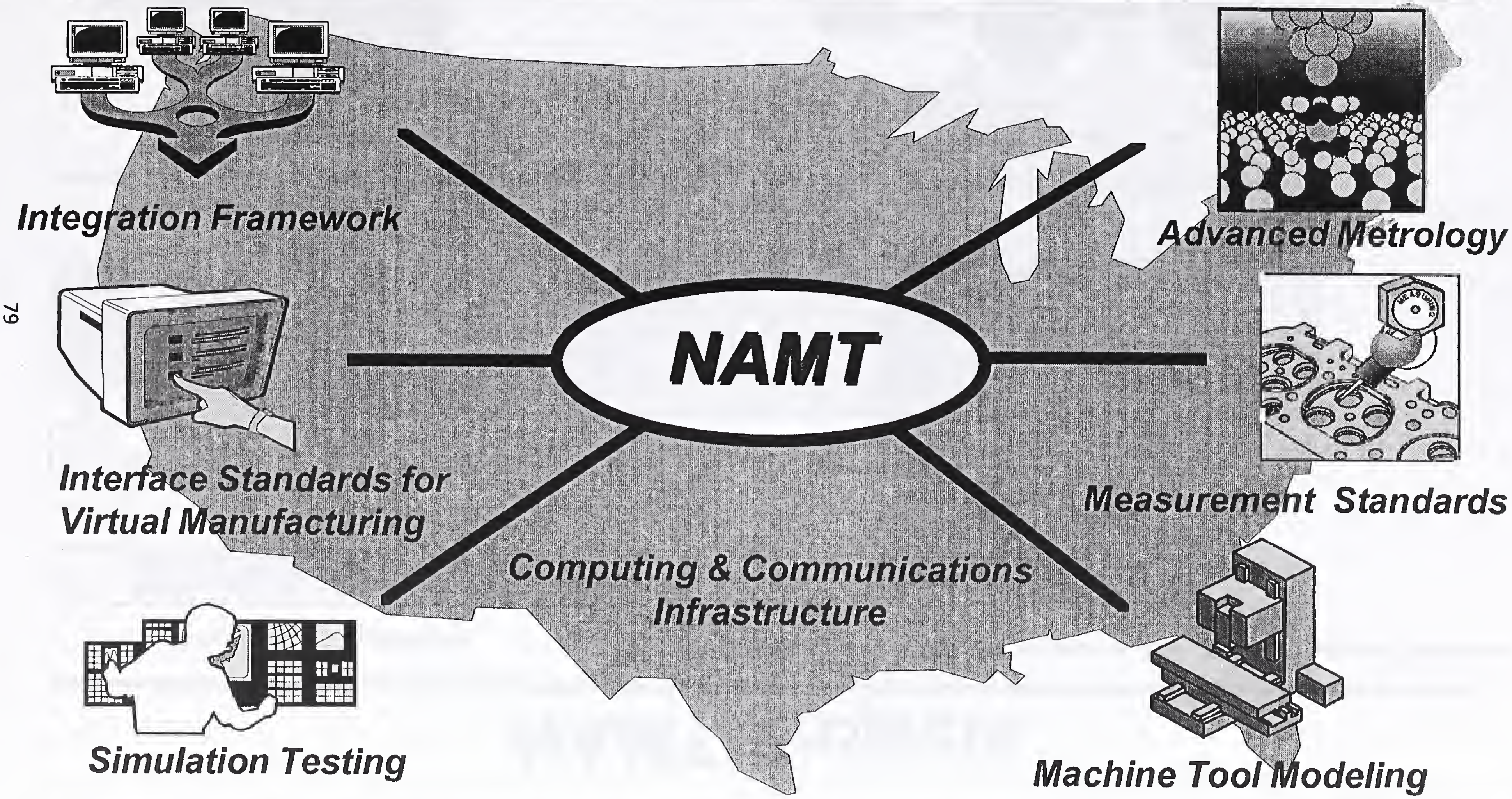
World Class

"Information Technology is the Key Enabler"

MEL NIST


...information-based manufacturing...
...manufacturing that relies upon and exploits the capabilities available to systems & processes through the application of information technology as a value-added resource...
Raw Materials and Component Parts
Manufacturing System
Capital Material Labor
Information + Capital Material Labor
World Class Products
Product and Process Data
Information Technology & Manufacturing
Traceability of informational Standards

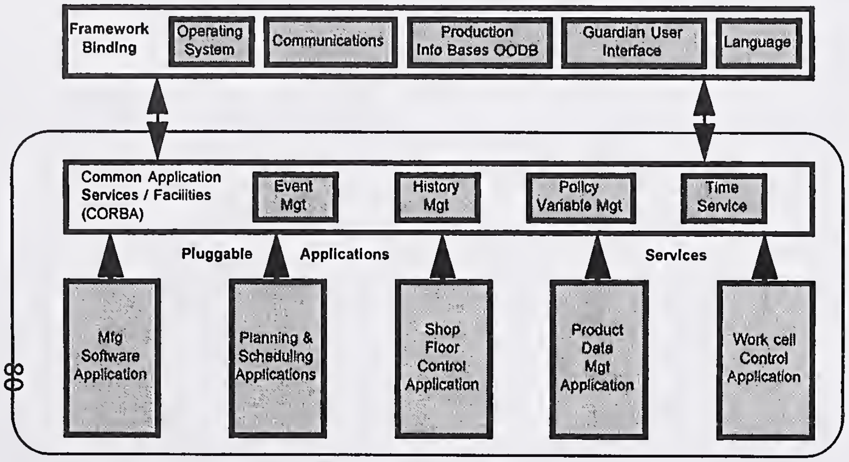
Measurements & Standards for Information-Based Manufacturing



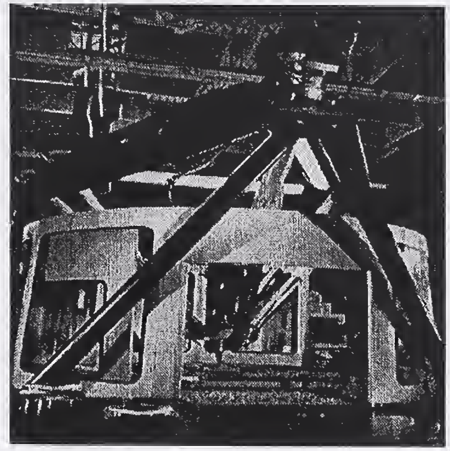
MEL NIST

NAMT Projects

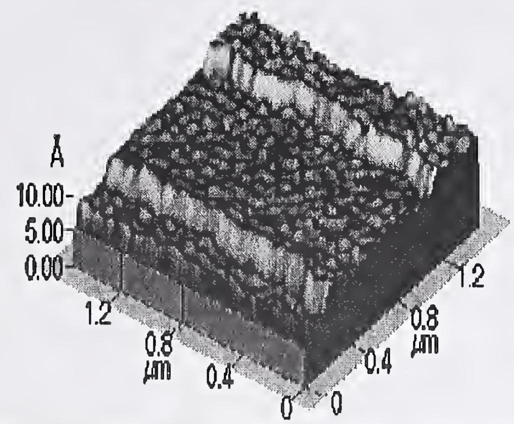
Manufacturing Framework



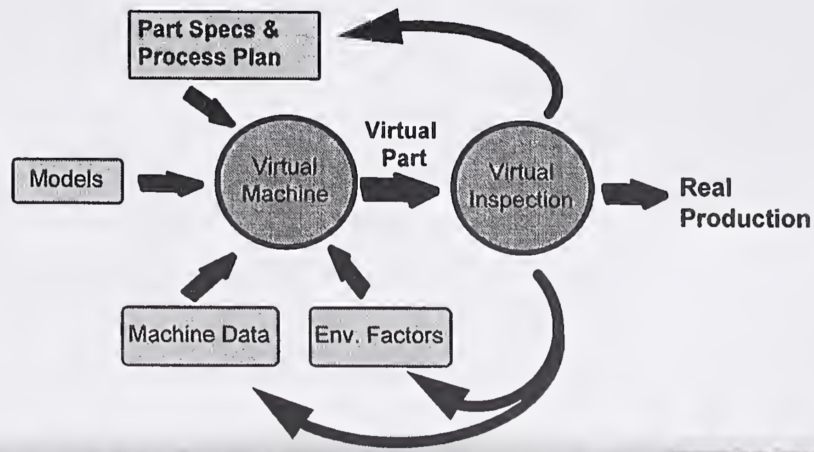
Characterization, Remote Access, & Simulation of Hexapod Machines



NanoManufacturing of Atom-Based Standards



Machine Tool Performance Model



MEL NIST

Conclusion

- Four programs of support for industry
- Cooperative planning and research
- Solid support for overcoming barriers

MEL NIST

Information Needs for Pump Design

Dr. Lalit Chordia, Thar Designs (chordia@thardesigns.com)

The summary for this presentation can be found on page 14
14 slides follow

THAR DESIGNS

INFORMATION REQUIREMENT IN DESIGN

Needs of a Small Business

Lalit M. Chordia, Ph.D.
Thar Designs, Inc.
Pittsburgh, PA 15238

November, 1996

THAR DESIGNS

Today's Talk

- Background
- Focus
- Future Information Requirements

THAR DESIGNS

Background

- Small Specialty Manufacturer
- Design, Develop and Manufacture
- High Pressure Components
  - Pumps
  - Vessels
  - Back Pressure Regulator
  - Cyclones

THAR DESIGNS

Focus: Pumps

- Application: Compress Liq. Carbon Dioxide
- Market: To Buy
- Type: Compressor, Reciprocating, Diaphragm
    - Literature: Very few papers available
    - Vendors: Expensive, Poor Performance
    - Analyze: Adapted from Liquid Pumps
        - Incompressible Liquids Vs Compressible CO_2
    - No Choice: But to Design

THAR DESIGNS

Issues in Design:

- Minimize Dead Volume
- Reduce Heat Generation
- Efficient Cooling Requirements
- Small Footprint
- Cost Effective

THAR DESIGNS

Minimize Dead Volume:

- Develop an Algorithm Relating
  - Density, Pressure and Temperature
- Tolerance Issues:
  - Tight Tolerance Vs Mfg. Process Vs Cost
  - Assembly Issues Became Critical Early
- Bearing Requirements were a key
- Importance of High Strength Materials

THAR DESIGNS

Reduce Heat Generation

- Why is this Important
    - Heats Fluid, Decreases Density
- Types of Heat: Compression & Frictional
- Frictional Heat
    - Sapphire Piston
    - Ceramic Coated Piston

THAR DESIGNS

Efficient Cooling

- Incorporated Heat Generation in Algorithm
  - Very Clear: Cooling is Critical
- Pre-Cool Fluid: Increases Density
- Pump Head Cooling:
  - Thermal Conductivity
  - Designed in Cooling Ports

THAR DESIGNS

Small Footprint

- Smaller Size in Most Cases Reduces Cost
  - Critical Manufacturing Parameters
  - Decreasing Stroke Volume
    - Increased Speed, Increases Friction
  - Reasonable Seal Life: New Seal Designs
  - High Strength Material

THAR DESIGNS

Design Time

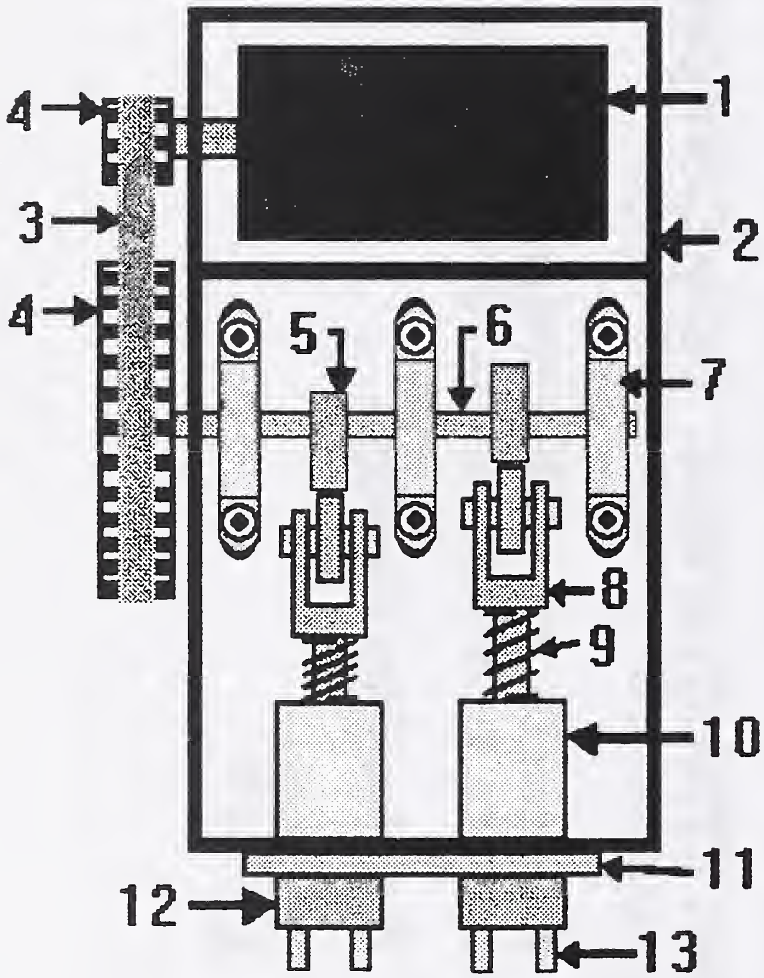
- Concept Formulation: 2 months 1993
- Prototyping: 4 months 1994
- Chassis Design: Took the longest time

THAR DESIGNS

Final Product:

- Performance: Works Extremely Well
- Cost Effectiveness: Very Competitive
    - All Parts were Machined
    - Scaleable Design

Pump Assembly



1	SERVO MOTOR
2	PUMP FRAME
3	TIMING BELT
4	TIMING PULLEY
5	CAM
6	CAM SHAFT
7	ROT. BEARING BLOCK
8	PISTON HOUSING
9	SPRING
10	LIN. BEARING BLOCK
11	SEAL BACKUP PLATE
12	PUMP HEAD
13	COOLING TUBES

THAR DESIGNS

How Could We have done Better:

- Rapid Access to Information:
  - Technology & Product: Same & Analogous
- Tools: Simulation, Manufacturing Process
- Rapid Prototyping Access
- Skilled Labor
- Small Business Limitations

THAR DESIGNS

Future Information Requirements:

- Rapid Access to Information:
  - Technology & Product: Same & Analogous
- Tools: Simulation, Manufacturing Process
- Database: Design Concepts, 3D of Products
- Where: Best Place is Internet
-

Information Retrieval During Design of a Medication Dispenser

James Michael, Diebold, Inc.

The summary for this presentation can be found on page 16
11 slides follow on six pages

Information Retrieval During the Design of a Medication Dispenser

MedSelect Systems

A Division of Diebold, Incorporated

James Michael, Mechanical Development Engineer

November 19, 1996 1

Discussion Overview

- Diebold & MedSelect Systems
- Medication dispenser
- Information access during product development
  - Information requirements (❒)
  - Access of information (✓)
  - Desired access of information (♪)
- Recommendations
- Open discussion

James A. Michael, Diebold, Incorporated 96/11/19 2

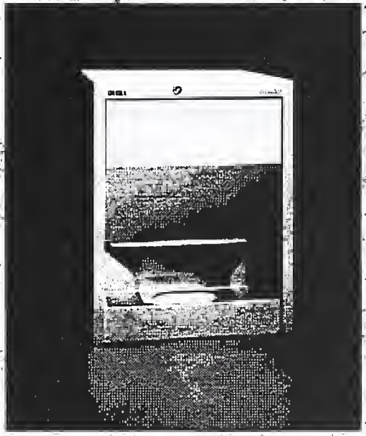
Diebold, Incorporated MedSelect Systems

- Physical & Electronic Security
- ATM's
- Medication / Medical Supply Tracking & Dispensing
  - Dispensers distributed throughout medical facility - ER, OR, ICU, . . .
  - Linked to medical facility information systems

James A. Michael, Diebold, Incorporated 96/11/19 3

Medication Dispenser - Unit Dose Module

- Dispensing of expensive and controlled substances



James A. Michael, Diebold, Incorporated 96/11/19 4

Product Development Cycle

product definition — initial concept — detailed design

James A. Michael, Diebold, Incorporated 96/11/19 5

Development: Product Definition

- ❐ Product Requirements
- ❐ Competitive Analysis
- ❐ Industry Standards, Legislation
- ✓ Customers / Marketing
- ✓ Patents, Libraries, WWW, Competitive literature
- ♪ Competitive products
- ♪ Information Consolidation / Resource Links

James A. Michael, Diebold, Incorporated 96/11/19 6

Development: Initial Concept

❐ Detailed Definition

❐ Technology Search & Conception Testing

✓ Industry standards - JCAHO, ASIS

✓ Patents, WWW, Magazines, Libraries, CD's, Conferences, Seminars, Med suppliers

♪ State regulations

♪ Information Consolidation / Resource Links

Development: Detailed Design

❐ Design Techniques - DFx

❐ Part Availability

✓ Texts, Associates, Self, Confer, Seminars, WWW, Technical knowledge services, Corp. standards

✓ Product standards - safety, packaging

✓ Suppliers, Thompson's, WWW

♪ CAE databases, PDM, Guidelines

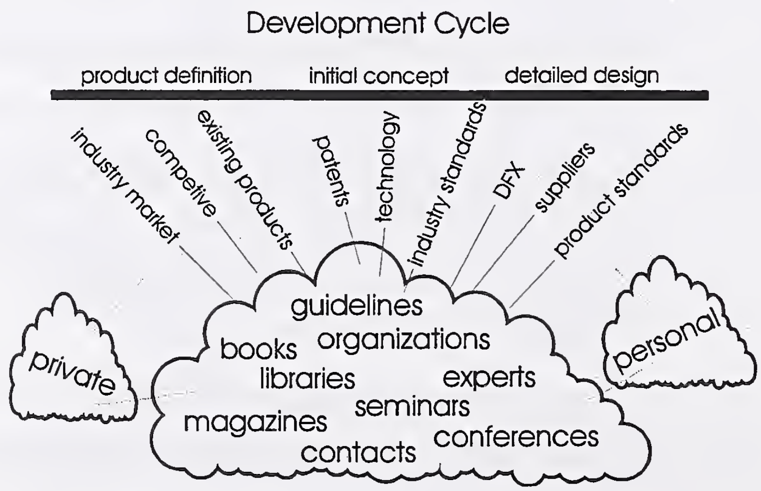
♪ Information Consolidation / Resource Links

Information Access

- Information requirements differ throughout development cycle
- Common access methods used
- Access during development was *good*
- Access *ease* can improve greatly

♪ Information Consolidation / Resource Links

Information Consolidation / Resource Links



James A. Michael
Diebold, Incorporated

Summary

- Product development successful with current information access methods
- With Information Consolidation / Resource Links *tied* to Development Cycle
  - *greater ease of access*
  - *more organized information*
  - *shorter development cycles*
  - *higher quality products sooner*

James A. Michael, Diebold, Incorporated 96/11/19 11

Development of an Adaptive Modeling Language for Knowledge-Based Engineering with Application to Interactive Gimbal Design

Dr. Rich Zarda, Lockheed Martin (zarda@slr.orl.mmc.com)

The summary for this presentation can be found on page 18
16 slides follow

LOCKHEED MARTIN

Administrative

Development of an Adaptive Modeling Language (AML) for Knowledge-Based Engineering with Application to Interactive Gimbal Design (IGD)

Prime Contractor:

Lockheed Martin Electronics & Missiles – *P. Richard Zarda, Chris Johnson*

Sub-Contractors:

TechNoSoft – *Adel Chemaly*

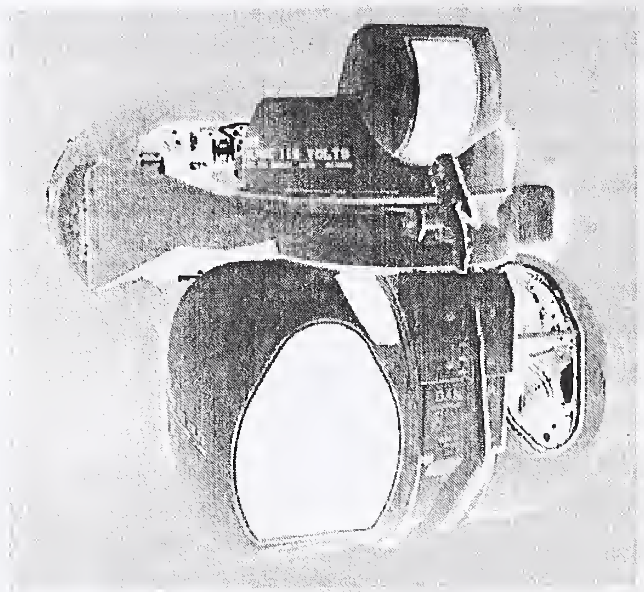
Coleman Research – *Tim Thornton*

Florida A&M University – *Ben Wang*

Lambda Research – *Groot Gregory*

Contract Start: March, 1996

Contract End: May, 1998



Apache TADS/PNVS Gimbal

Presented at the MADE PI Meeting, Detroit, MI, Aug 1996

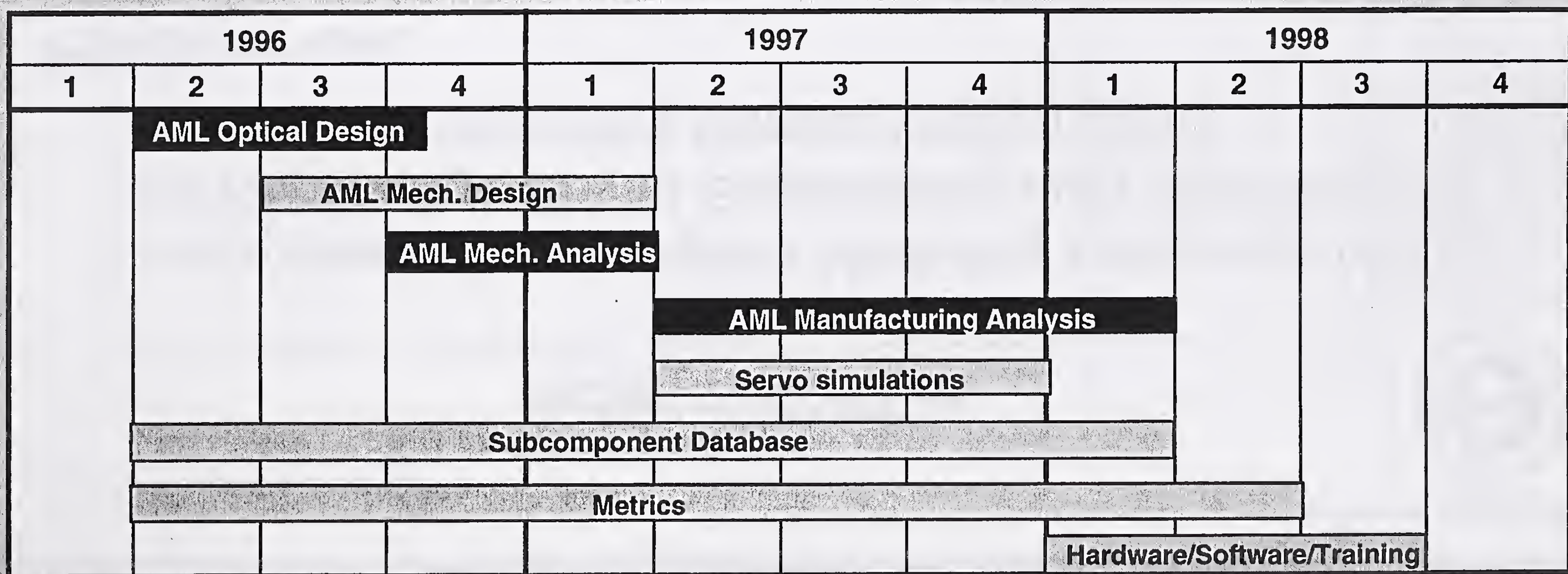
Program Roadmap

Objectives

1. AML: Development/Enhancement of an Adaptive Modeling Language for Knowledge Based Engineering that allows industries to capture their processes and the associated knowledge.

2. IGD: Interactive Gimbal Design (IGD) System will be developed based on AML objects that will capture the gimbal design process followed at Lockheed Martin. The IGD structure will allow for a creative design environment and capture the knowledge of that creativity.

3. Gimbal Database: An electromechanical, standardized, distributed gimbal sub-components database will be developed and integrated with the IGD System.



Program Roadmap

Functional Capability of the IGD System

Requirements
- Optical – aperture, field of view, wide, narrow, magnification, zoom
- Servo – minimum natural frequency, transfer functions, timing
- System – range, stabilization, field of regard
- Mechanical – g loads, frequency loads, random loading, shock, number of axes
- Thermal – optics temperature, heat dissipation, gyro and torquer, electronics

Previous Gimbal Designs

Standardized Databases
- Torquers
- Gimbals
- Bearings
- Gyros
- Resolvers
- Lenses
- Mirrors
- Belts

Optical Design
- Prescription for lens design
- Mirror design
- Field stops
- Fold lines
- Geometry of optical elements
- Restricted volumes
- Geometry of optical train
- ACCOS-V

Interactive Gimbal Design
Run-time executables of off-the-shelf software products integrated to form a multidisciplinary, interactive software tool. The integration will allow for the concurrent assessment of design decisions and offer a multidisciplinary optimization process

Gimbal Mechanical Design
- 3-D geometry of gimbals
- Assembly of gimbals and optical train
- Stabilized mirror, stabilized platform
- Two-axis gimbals, drive ratios
- Bearing selection with size, mass props, port for lens cell, friction-to-inertia ratios, stiffness
- Torquer selection with size, mass props, torque rate, voltage requirement, analysis
- Assembly and manufacturing considerations
- Gyro selection with size, noise, mass props, accuracy, sampling rate, digital, analog, location, quantification of size, noise accuracy, sampling rate, inertia-to- friction ratios, digital, analog

Dynamic Analysis
- Line of sight jitter calculations
- Natural frequencies of components and assemblies
- Incorporation of optical equations
- Lens and mirror distortion
- Flexibility of bearings
- Transfer functions for servo
- NASTRAN

Servo
- MATRIX-X

Thermal Analysis
- Maximum temperature of optical components
- Thermal gradients in optical components
- Thermal distortion of optical components
- Cooling of torquers, gyros, and optical geometry
- Cooling of bearings
- FEM/SINDA

Materials Database
- M-Vision

Manufacturing Processes
- Automatic generation of machining (NC) and inspection (CMM) programs
- Capability to monitor and improve product designs (early)
- Reduce process cost through simulation (manufacturing plans)
- Early assessment of materials and processes
- Automatic part setup for fixturing/tooling

Milestones

Accomplishments to date:

- LMC MADE-IGD kickoff meeting completed.
- ORACLE was reviewed and picked.
- Review of FEM. Gimbals to be modeled as solid elements.
- Large aspect ratio right angle parabolic tetras recommended.
- Gimbal design time quantified (metric).
- A high-end HP workstation purchased. Pro/E, Matrix-X, Master Series, PATRAN, MSC/NASTRAN, AML installed.
- All contractors AML trained.
- Wide-area-networks (WAN) reviewed. Frame-Relay chosen and installed.
- 60 Gimbal Sub-components vendors contacted.
- MADE-IGD home page deployed.
- Software/hardware contributions from Oracle, PTC, SDRC, AVS, MSC, and HP.
- AML class-objects developed for optical design.
- Reviewed transfer of intelligent parametric solids.

Milestones

Accomplishments over next 6 months:

- The optical simulation using AML class-objects will be completed. Tool will be used and measured on existing programs (metric).

- AML class-objects will be developed to model the functionality of mechanical gimbal components and subcomponents (inner and outer gimbals, torquers, lenses and mirrors, gyros, resolvers, belts, and bearings).

- AML class-objects will be developed for finite element modeling of solid parts. Integration with PATRAN, Master Series and MSC/NASTRAN.

- AML class-objects will be developed for dynamic component modal synthesis of gimbal assemblies.

- AML class-objects will be enhanced to assess manufacturing processes of gimbal components.

Milestones

Accomplishments during the 6-12 month time frame:

- A direct interface will be developed using a "history script file" between Pro/E and AML.

- An electromechanical, standardized distributed gimbal subcomponent library will be developed based on an Oracle database manager using the STEP standard. The IGD system will reference this database.

- AML class-objects will be developed for control system simulations.

- IGD System demonstrations of mechanical design/analysis and integrated optical design.

LOCKHEED MARTIN

Interactive Gimbal Design (IGD)

MANUFACTURING AUTOMATION AND DESIGN ENGINEERING PROGRAM

Milestones

Companies Contributing to the Gimbal Subcomponent Database:

Gyros

Allied Signal Aerospace	616-582-6528	Boyne City MI
Condor Pacific Ind	818-865-3000	Westlake Village, CA
Delco Systems	805-961-6370	Goleta, CA
Honeywell	612-851-1000	Minneapolis, MN
Humphrey Inc	619-565-6631	San Diego, CA
Litton Special Devices	610-328-4008	Springfield, PA
Rotech Electronics	717-752-2288	Berwick PA
BEI Systron Donner/Inertial Div	510-671-6412	Concord, CA
Northrop Corporation	213-600-3000	Hawthorne, CA

Resolvers

Astro Instrument Corp	305-698-6000	Deerfield Beach, FL
Farrand Controls	814-751-2600	Valhalla, NY
Honeywell	phone#	Address
Kollmorgen Inland Motor	540-639-9045	Radford, VA
Lambda Novatronics	305-942-5200	Pompano Beach, FL
Vernitron Group	519-428-5581	San Diego, CA

Bearings

Allied Signal	518-686-7301	Hoosick Falls NY
Arow Components and Fasteners	800-227-2901	Hayward, CA
General Bearing Corp	814-358-6000	Blauvelt, NY
Kaydon	516-755-3741	Muskegon, MI
Keystone Carbon	814-781-7648	Saint Marys PA
Micro Plastics	501-453-2281	Flippin, AR
Miniature Precision Bearings	07 3252-2488	Address
International	517-3252-2499	Address
US Thermoplastics	800-548-4844	Temecula, CA

Accelerometers

PCB Piezotronics	716-684-0001	Depew, NY
Setra Systems	800-257-3872	Acton, MA
Sunstrand	815-226-6000	Rockford IL

Actuators

Allied Signal	219-231-3000	South Bend IN
BEI Sensors	619-744-5671	San Marcos, CA
Crane, John	904-253-0528	Daytona Beach, FL
E-Systems	214-661-1000	Dallas TX
Greystone Electronics	800-704-7048	Waterbury, CT
Kollmorgen	540-639-9045	Radford, VA
Kearfott	704-686-3811	Black Mountain, NC
Keanne Controls	714-526-3341	Anaheim, CA
Industrial Devices	800-747-0064	Novato, CA
Litton	610-328-4000	Springfield, PA
Northern Magnetics	805-257-0215	Santa Clara, CA
Philips Components	407-745-3300	Jupiter FL
Werner	815-568-8001	Marengo, IL

Attenuators

Anaren Microwave Inc	315-432-8909	East Syracuse, NY
Atlantic Microwave Corp	508-779-6963	Bolton, MA
Hewlett-Packard	516-226-8900	Amityville, NY
Hughes Aircraft	310-517-6000	Torrance, CA
Kay Elemetrics Corp	201-628-6200	
Merrimac Inds	201-575-1300	West Caldwell, NJ
Microtech	203-272-3234	Cheshire, CT
Millitech	413-665-8551	South Deerfield, MA
Mini-Circuits Inc	718-834-4500	Brooklyn, NY
Newark Electronics	800-281-4320	Chicago, IL
Penstock Inc	800736-7862	Sunnyvale, CA
Shallco Inc	919-934-3135	Smithfield, NC
Vernitron	813-347-2181	Saint Petersburg, FL
Victor Microwave	817-245-4472	Wakefield, MA

Potentiometers

Allied Electronics	800-433-5700	Fort Worth, TX
BEI Electonics	415-956-4477	Tustin, CA
Brel Intern Components	800-237-4564	Sarasota, FL
C&K Components	800-635-5838	Watertown, MA
Humphrey	619-565-6631	San Diego, CA
Greystone Electonics	800-704-7049	Waterbury, CT
Marsh Electonics	800558-1238	Milwaukee, WI
Memcor-Truman Inc	800-445-3442	Huntington, IN
National Electonics	714-892-7748	Garden Grove, CA
Philips Components	800-447-3762	Riviera Beach, FL
Precision Control	206-281-8380	Seattle, WA
Servo Instrument Corp	608-356-6623	Baraboo, WI
Time Electonics	800-772-8638	Peabody, MA
Tocos America Inc	708-884-6664	Schaumurg, IL
Vernitron	813-347-2181	Saint Petersburg, IL
Vishay Resistive Systems	610-640-9081	Malvern, PA

Inductors

Allied Electronics	800-433-5700	Fort Worth, TX
American Precision	716-652-3600	East Aurora, NY
Caddell-Burns	516-746-2310	Mineola, NY
Communication Coil, Inc	708-671-1333	Schiller Park, IL
Digi-Key Corp	800-344-4539	Thief River Falls, MN
Electro Assemblies	708-498-6520	Northbrook, IL
Frontier Electronics	805-522-9998	Simi Valley, CA
Inductor Supply	800-854-1881	Anaheim, CA
Inductor Tecnologies	405-422-3222	El Reno,OK
Magnetic Component Eng	800-989-5656	Inglewood CA
Newark Electonics	800-281-4320	Chicago, IL
Precision Electonics	617-834-6677	Marshfield, MA
Torotel	816-761-6314	Grandview MO
Wilco Corp	317-293-8300	Indianapolis, IN

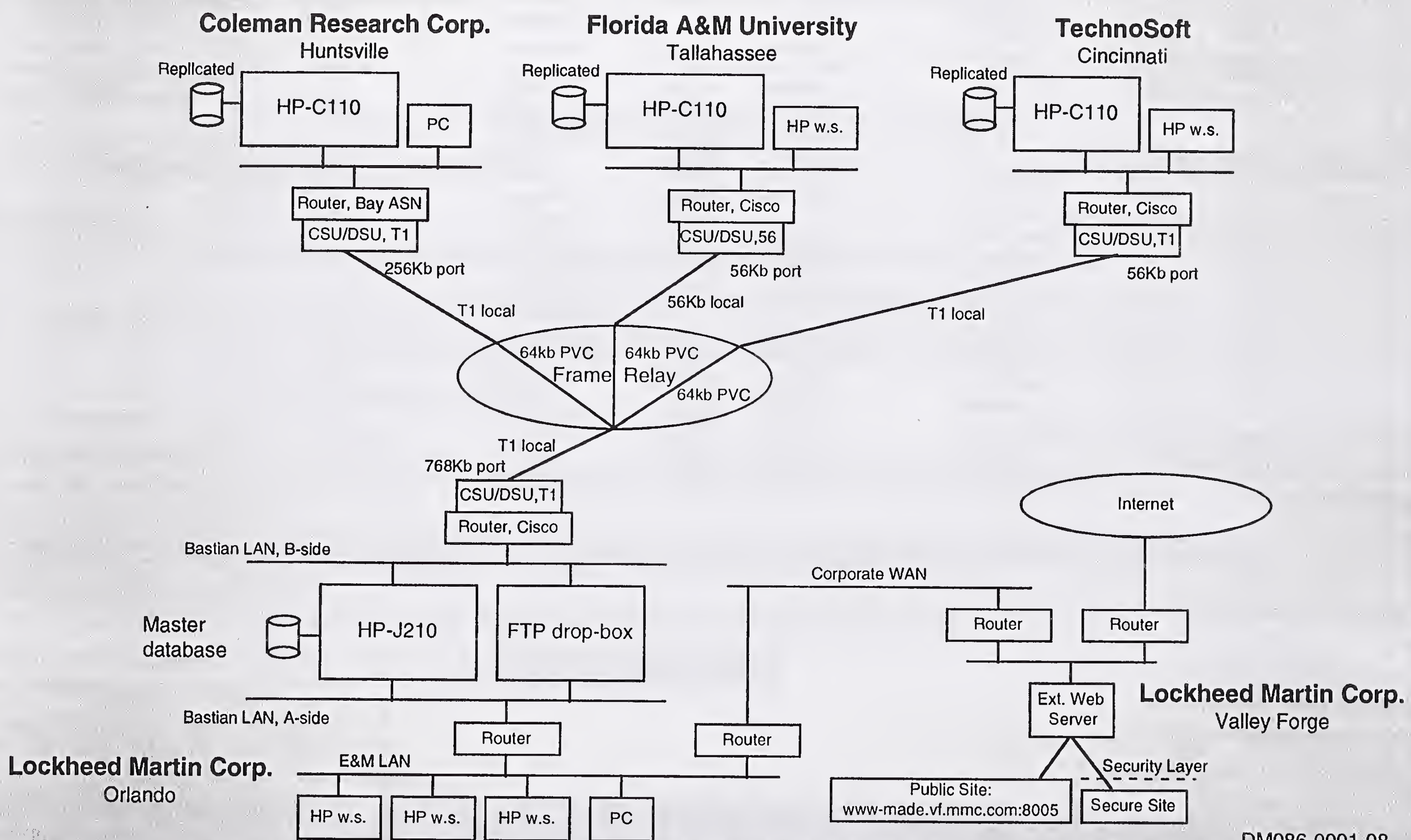
Motors

Kearfott	704-686-3811	Black Mountain, NC
Kollmorgen Inland Motor	540-639-9045	Radford, VA
Berry Controls	305 834 1244	Address
Lord Aerospace	407 365 2233	Address
Shellcast Foundries	514 322 3760	Address
Allied Signal	218-231-3000	South Bend IL
BEI	618-744-5671	San Marcos, CA
Sunstrand	815-226-6000	Rockford IL
Vernitron Group	818-428-5581	San Diego, CA
Kollmorgen Inland Motor	540-639-9045	Radford, VA
Sager	800-724-3780	Hingham, MA
McLean Engering	609-799-0100	Princeton Junction, NJ
RS Electronics	800-366-7750	Livonia, MI
Deneltron Corp	310-530-3530	Torrance, CA
International Specialists	813-949-7409	Tampa, FL
Sanyo Denki America	310-212-7724	Torrance, CA

DM086-0001-07

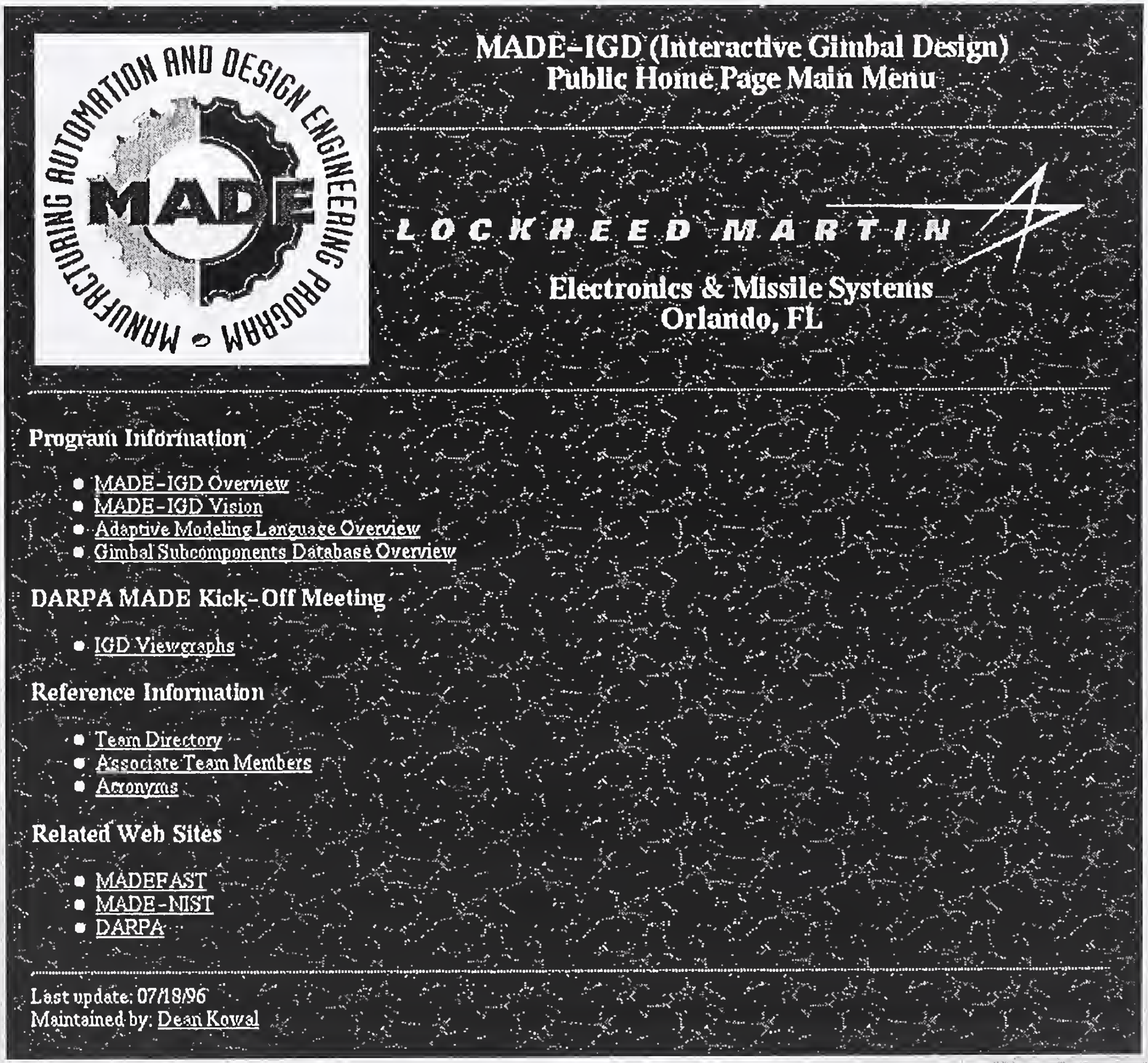
Milestones

MADE-IGD Wide Area Network Diagram



Milestones

MADE-IGD Home Page - http://www-made.vf.com:8005/



MADE-IGD (Interactive Gimbal Design)
Public Home Page Main Menu

LOCKHEED MARTIN
Electronics & Missile Systems
Orlando, FL

Program Information

- MADE-IGD Overview
- MADE-IGD Vision
- Adaptive Modeling Language Overview
- Gimbal Subcomponents Database Overview

DARPA MADE Kick-Off Meeting

- IGD Viewgraphs

Reference Information

- Team Directory
- Associate Team Members
- Acronyms

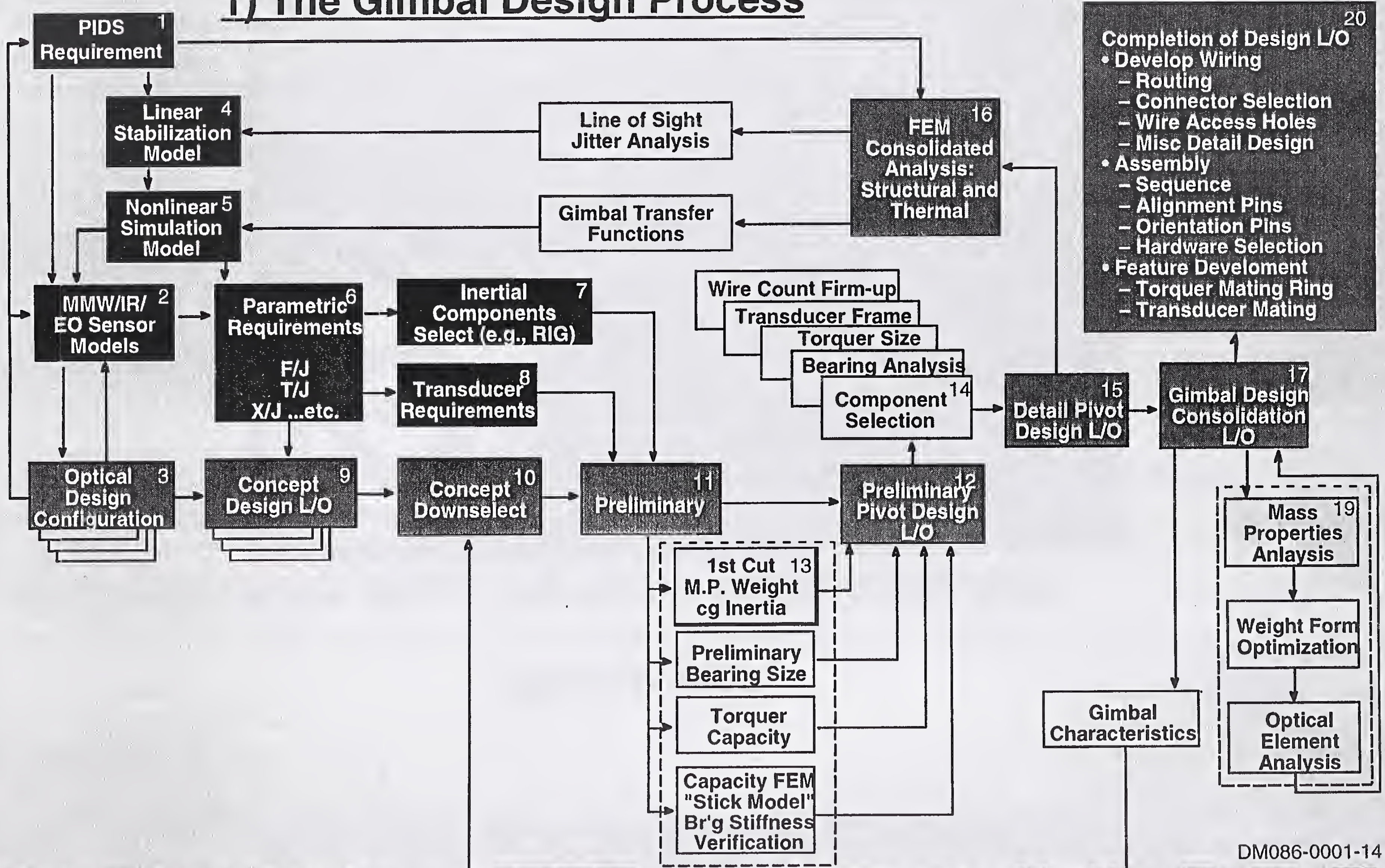
Related Web Sites

- MADEFAST
- MADE-NIST
- DARPA

Last update: 07/18/96
Maintained by: Dean Koval

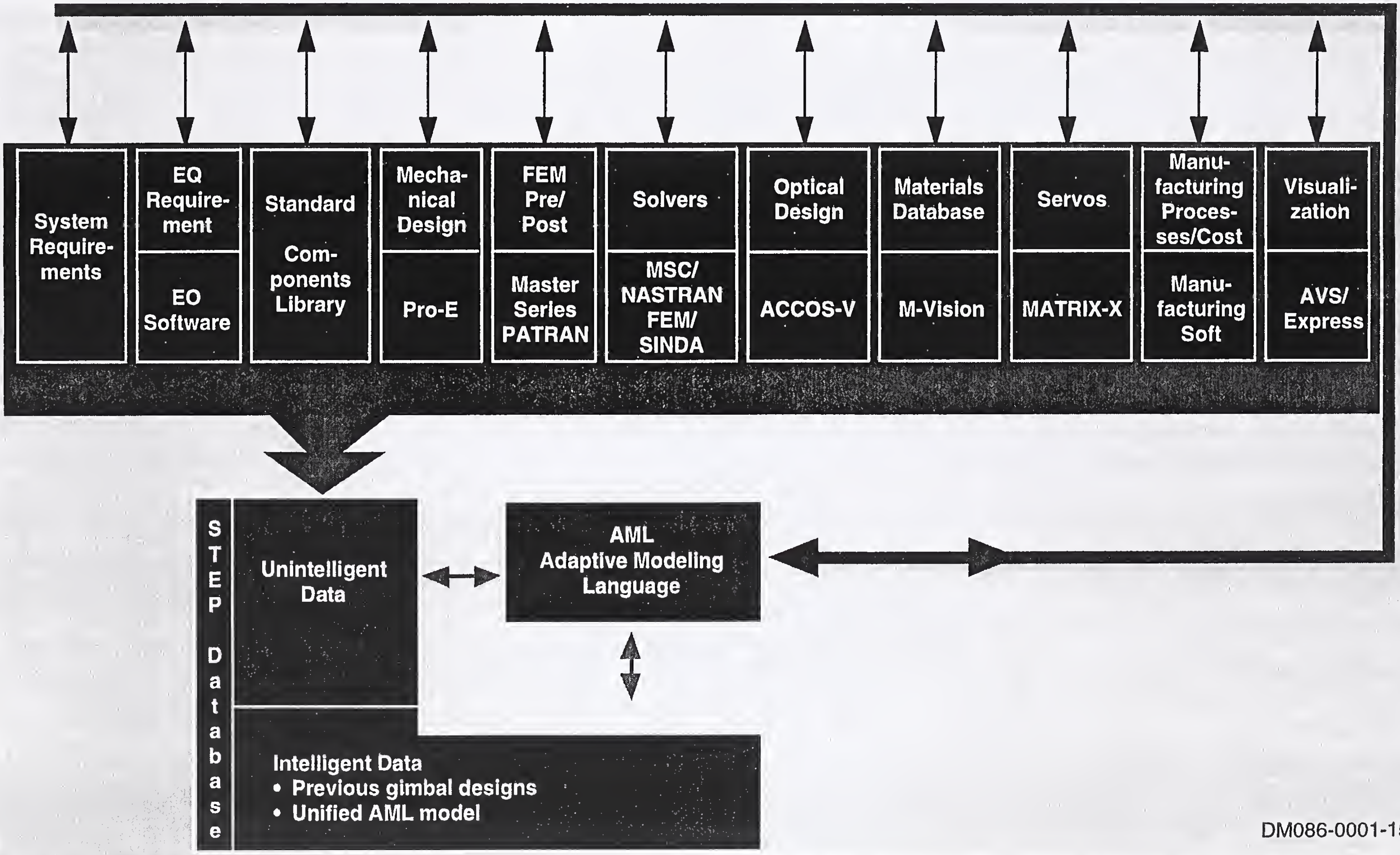
Technical Approach

1) The Gimbal Design Process



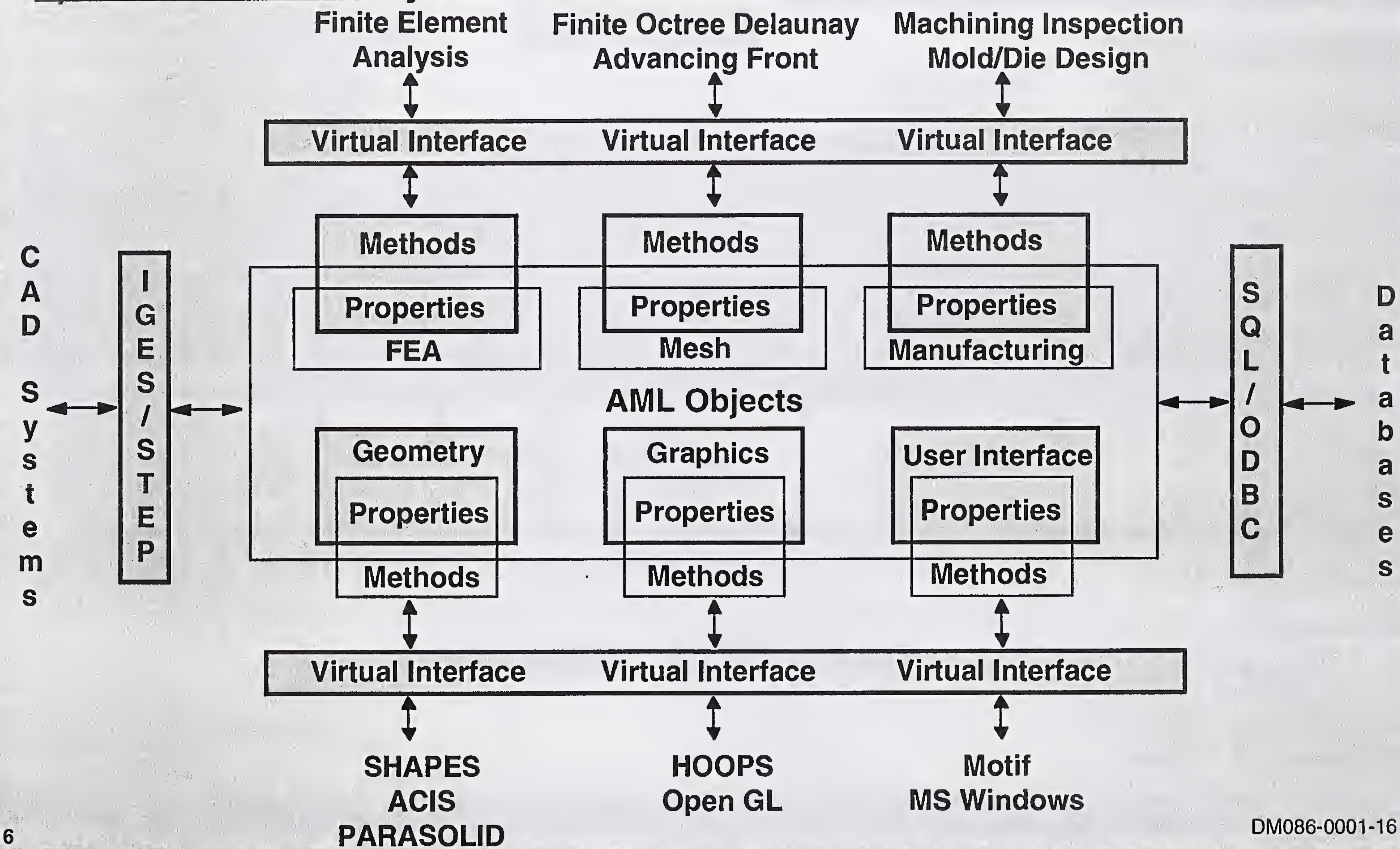
Technical Approach

2) Basic Structure of the IGD System

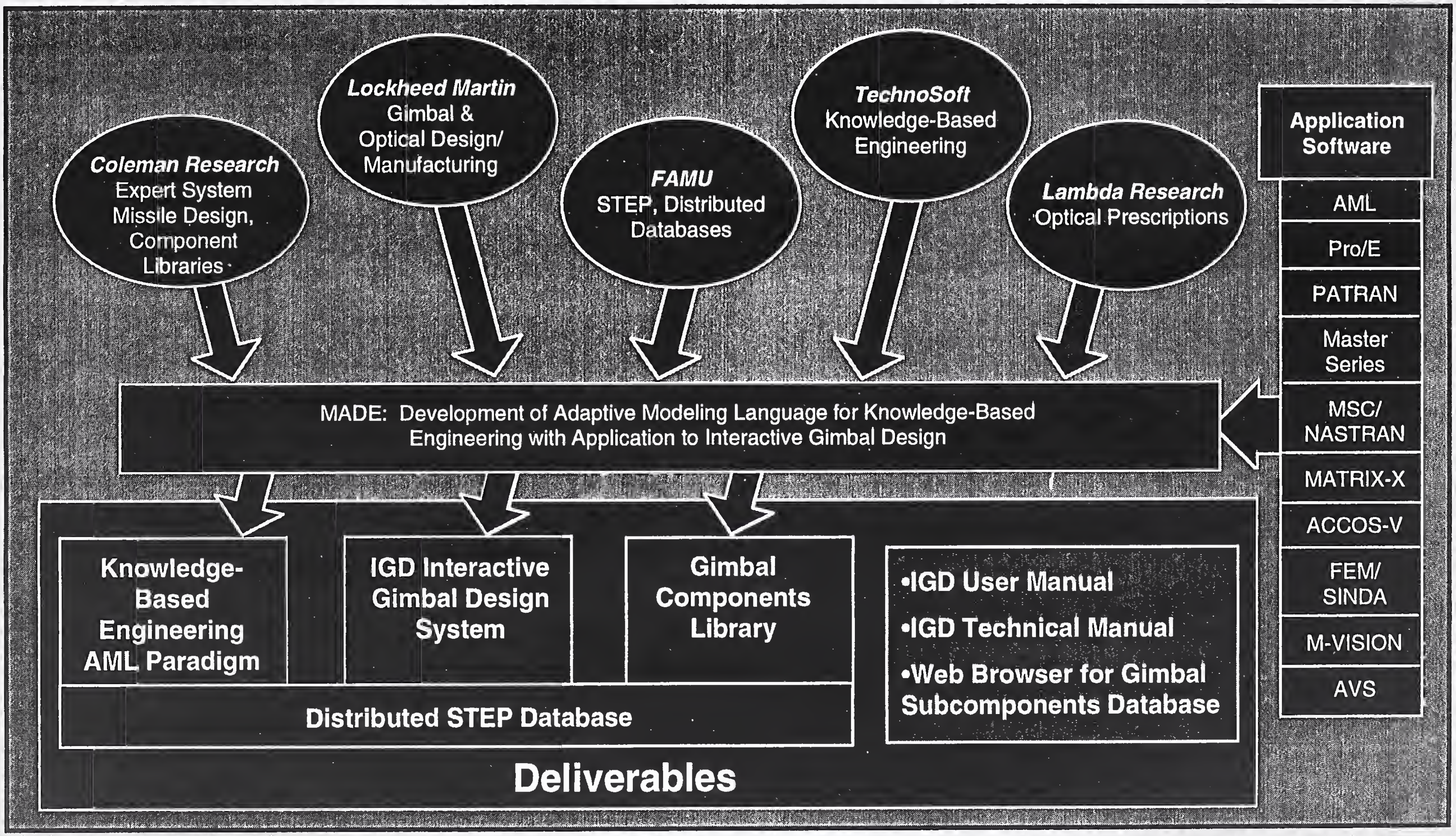


Technical Approach

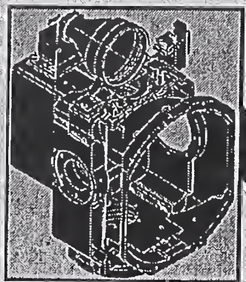
3) AML's General System Architecture

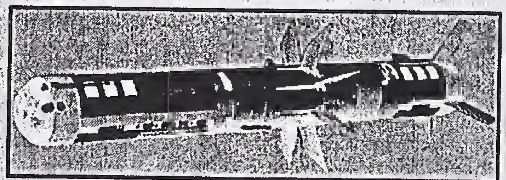


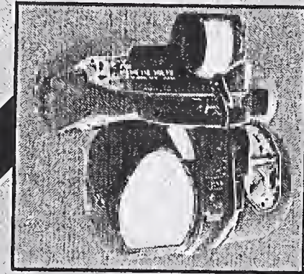
Deliverables



Transition



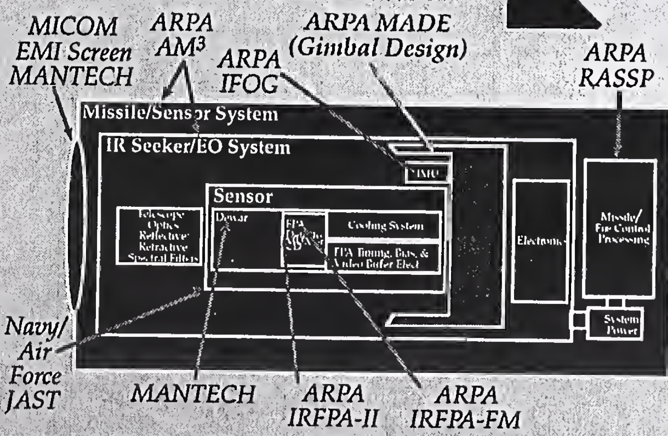




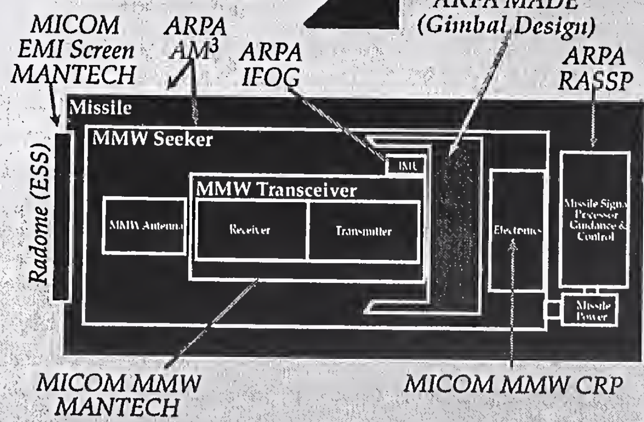
Development of Adaptive Modeling Language (AML) for Knowledge-Based Engineering with Application to Interactive Gimbal Design (IGD)

- Proves KB framework
- Applies to all DoD and commercial electromechanical systems
- Provides a revolutionary design cost edge
- Complements existing Government investments in reducing seeker/sensor life cycle cost

EO/IR Seeker and Sensor



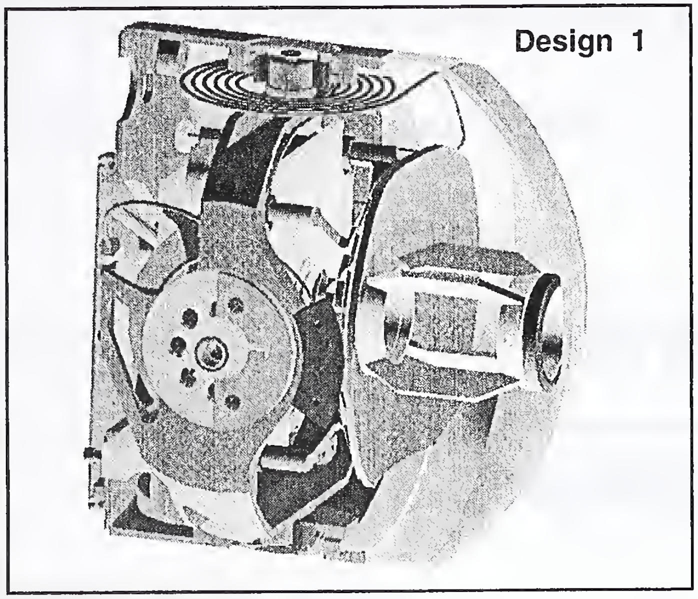
MMW Missile Seeker

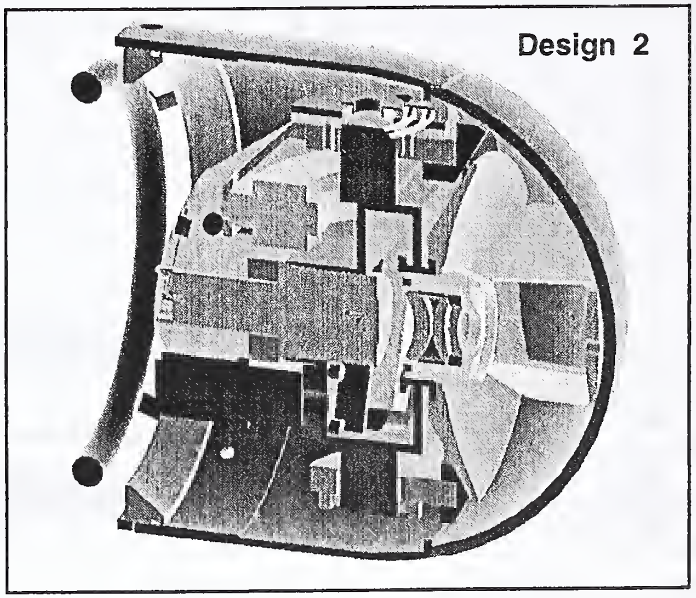


Metrics for Impact and Progress

Statistics for the Gimbal Design of an In-House Missile System

	Optics	Mechanical	FE Analysis	Servo	Total
Design 1	1,082[1] hrs	4,955 hrs	1,214 hrs	1,383 hrs	8,634 hrs
Design 2	1,286 hrs	2,356 hrs	3,384 hrs	0 hrs	7,026 hrs





[1]All data in this table has been multiplied by the same constant factor in order to protect proprietary information.

Technology Gaps

Additional Technology Gaps to be Challenged:

- Mature STEP Definition.
- 3-D Automated Packaging Algorithms.
- Automated FEM Modeling (beams/shells/solids) of Assemblies with Intelligent Coupling of Assembled Parts.
- National Support of the Gimbal Sub-components Database.
- National Support of a Material's Database.
- Reliable Manufacturing Costing Algorithms.

MADE-IGD Electro-Mechanical Gimbal Subcomponent Database

Dr. Tim Malueg, CRC (tmalueg@hsv.crc.com)

The summary for this presentation can be found on page 19
11 slides follow

MADE-IGD ELECTRO-MECHANICAL GIMBAL SUBCOMPONENT DATABASE

19 November 1996

MADE-IGD ELECTRO-MECHANICAL GIMBAL SUBCOMPONENT DATABASE

PURPOSE:

- TO PROVIDE A REPOSITORY WHICH CAPTURES COMPLEX ELECTRO-MECHANICAL GIMBAL SUBCOMPONENT DATA, PROGRAM-SPECIFIC AND COTS, FOR GIMBALED SYSTEMS

NEEDS:

- SIMULTANEOUSLY SUPPORT IGD SYSTEM (WITH COTS AND PROPRIETARY GIMBAL SUBCOMPONENTS) AND INTERNET QUERIES (WITH COTS GIMBAL SUBCOMPONENTS)
- PROVIDE SECURITY OF PROPRIETARY DESIGNS
- AUTOMATE QUERIES OF PERFORMANCE CURVES GIVEN DIFFERING INPUT CRITERIA
- PROVIDE FOR UNITS OF PREFERENCE FOR DATA INPUT AND DATA QUERIES
- PROVIDE TWO-TIER LEVEL OF QUERIES, ONE FOR DISCRETE, STATIC ATTRIBUTION AND ANOTHER FOR COMPUTED PERFORMANCE CHARACTERISTICS
- PROVIDE CONCEPTUAL AND DETAILED GEOMETRY

MADE-IGD ELECTRO-MECHANICAL GIMBAL SUBCOMPONENT DATABASE

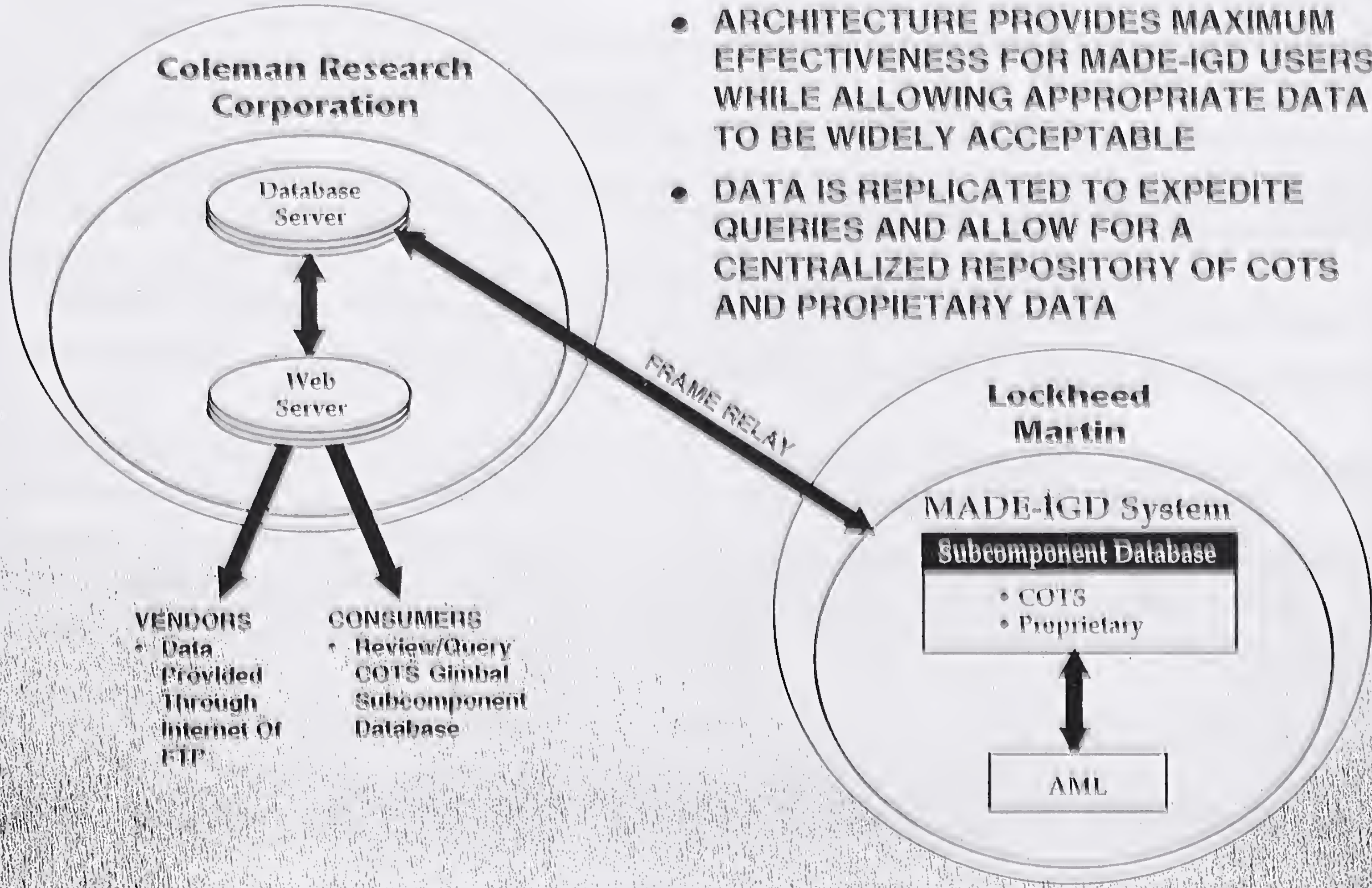
BENEFITS:

- EXPEDITES SELECTION PROCESS THROUGH ON-LINE DATA ACCESS
- EXPANDS THE SEARCHABLE SUBCOMPONENT MANUFACTURERS AVAILABLE AND PROVIDES CONSISTENCY OF DATA
- CAPTURES PROGRAM-SPECIFIC DESIGNS FOR RE-USE
- ASSISTS IN THE IDENTIFICATION AND RESOLUTION OF STANDARDS FOR ELECTRO-MECHANICAL GIMBAL CHARACTERIZATIONS
- SUPPORTS THE NATIONAL INFORMATION INFRASTRUCTURE THROUGH WWW ACCESS TO THIS DATA

MADE-IGD ELECTRO-MECHANICAL GIMBAL SUBCOMPONENT DATABASE

- **USED TO REVIEW, SELECT, AND AUTOMATICALLY TRANSFER COTS/ PROPRIETARY COMPONENT DATA INTO IGD DESIGN ENVIRONMENT**
  - RE-USE AVOIDS NON-RECURRING ENGINEERING COSTS AND SHORTENS DESIGN CYCLE TIME
- **COMPRISED OF ELECTRO-MECHANICAL COMPONENTS ASSOCIATED WITH GIMBAL SUBSYSTEMS (COTS AND COMPANY PROPRIETARY); BEARINGS, RESOLVERS, LENSES, GYROS, MIRRORS, MOTORS, SYNCHROS, INDUCTORS, TACHOMETERS, ETC.**
- **VENDOR PARTICIPATION DESIGNED TO SUPPORT VARIOUS RANGES: FROM HARDCOPY INFORMATION TO BULK DATA TRANSFER TO DIRECT VENDOR DATABASE ACCESS**
- **DISTRIBUTED, WITH SOME SYMMETRIC, ASYNCHRONOUSLY REPLICATED TABLES**
- **FLEXIBLE, DYNAMIC, AND EXTENDABLE ARCHITECUTRE FACILITATES EVOLVING CHARACTERIZATIONS OF SUBCOMPONENTS WITH MINIMIZED IMPACT**
- **"STEP-LIKE" STRUCTURE (EVOLVING STEP STANDARDS) FOR DETAILED GEOMETRY AND CUSTOMIZED FORMAT FOR SIMPLISTIC, "CONCEPTUAL" REPRESENTATIONS**
- **COTS AND PROPRIETARY DATA DIRECTLY ACCESSIBLE THROUGH INTERACTIVE GIMBAL DESIGN SOFTWARE INTERFACE**
- **COTS DATA ACCESSIBLE THROUGH INTERNET/WWW TO SUPPORT NATIONAL INFORMATION INFRASTRUCTURE**

SYSTEM NETWORK DESIGN



- ARCHITECTURE PROVIDES MAXIMUM EFFECTIVENESS FOR MADE-IGD USERS WHILE ALLOWING APPROPRIATE DATA TO BE WIDELY ACCEPTABLE
- DATA IS REPLICATED TO EXPEDITE QUERIES AND ALLOW FOR A CENTRALIZED REPOSITORY OF COTS AND PROPIETARY DATA

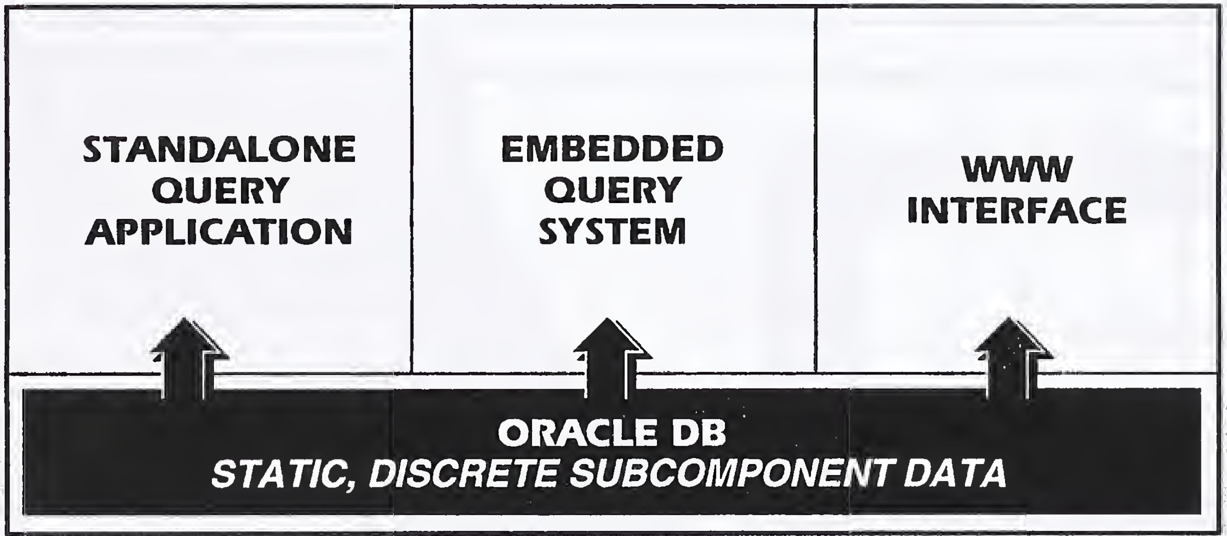
STANDARD COMPONENTS LIBRARY

- **EXAMPLE COMPONENT CLASSES**
  - MOTORS
  - INERTIAL SENSORS
  - GYRO
  - GEARS
  - BEARINGS
  - FOCAL PLANE ARRAY
  - OPTICS
  - LENS
  - MIRROR
  - ANTENNA
  - SIGNAL PROCESSOR
  - SERVO CONTROL
- **EXAMPLE MOTOR SUBCLASS**
  - MOTOR
    - BRUSHLESS
    - GEAR-DRIVEN
    - BELT-DRIVEN

STANDARD COMPONENTS LIBRARY
– Example Component and Attributes –

- **GYRO**
  - VENDOR NAME, ADDRESS, PHONE, EMAIL, WEB URL, ETC.
  - VENDOR CONTACT
  - MODEL NUMBER
  - SIZE
  - MASS PROPERTIES
  - SCALE FACTOR ERROR
  - NON-ORTHOGONALITY
  - MISALIGNMENT
  - ANGULAR ACCELERATION SENSITIVITY
  - G-SENSITIVITY
  - G2-SENSITIVITY
  - BIAS
  - NOISE
  - NATURAL FREQUENCY
  - DAMPING RATIO
  - QUANTIZATION
  - CONCEPTUAL GEOMETRY
  - DETAILED GEOMETRY (STEP/ IGES)

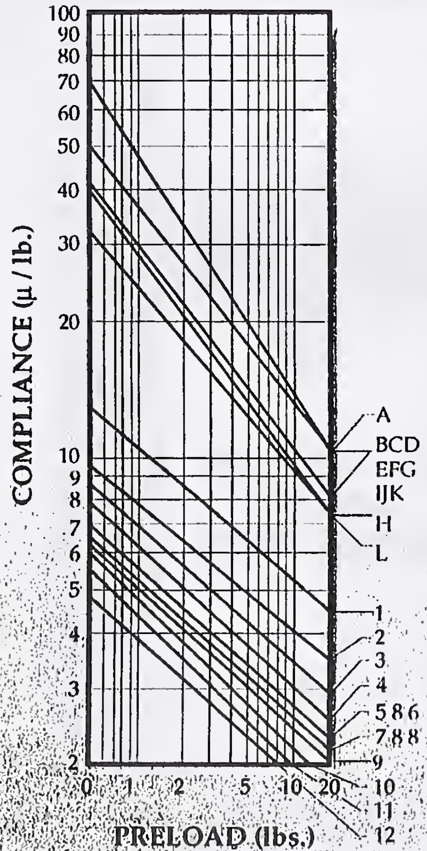
SYSTEM ARCHITECTURE



STATIC DATA IS RETRIEVED AND PERFORMANCE CALCULATIONS PERFORMED AT THE APPLICATION LEVEL

SAMPLE BEARING DATA

COMPUTED PERFORMANCE DATA



STATIC DATA

INSTRUMENTATION SERIES BEARINGS - CROSS REFERENCE

BORE (in.)	O.D. (in.)	N	d	Curve	MPB Basic Part No.	NHBB Basic Part No.	Bardin Basic Part No.
0.06	0.19	6	1mm	A, 1	3	R1	R1
0.09	0.31	6	1/16	C, 3	5	R1-5	R1-5
0.13	0.25	7	1mm	B, 2	418	RI-418	–
0.13	0.38	7	1/16	D, 4	R2	R2	R2
0.16	0.63	6	1/8	I, 9	34	R-1640X2	34
0.19	0.5	7	3/32	E, 5	R3	R3	R3
0.24	0.75	6	9/64	J, 10	36	–	36
0.25	0.5	11	1/16	F, 6	814	–	R188
0.25	0.63	8	3/32	G, 7	R4	R4	R4
0.31	0.5	11	1/16	H, 8	8,516	RI-8516	R1810
0.32	0.87	7	5/32	K11	38	R2280	38
0.38	0.88	7	5/32	L, 12	R6	R6	R6

TABLE ARCHITECTURE

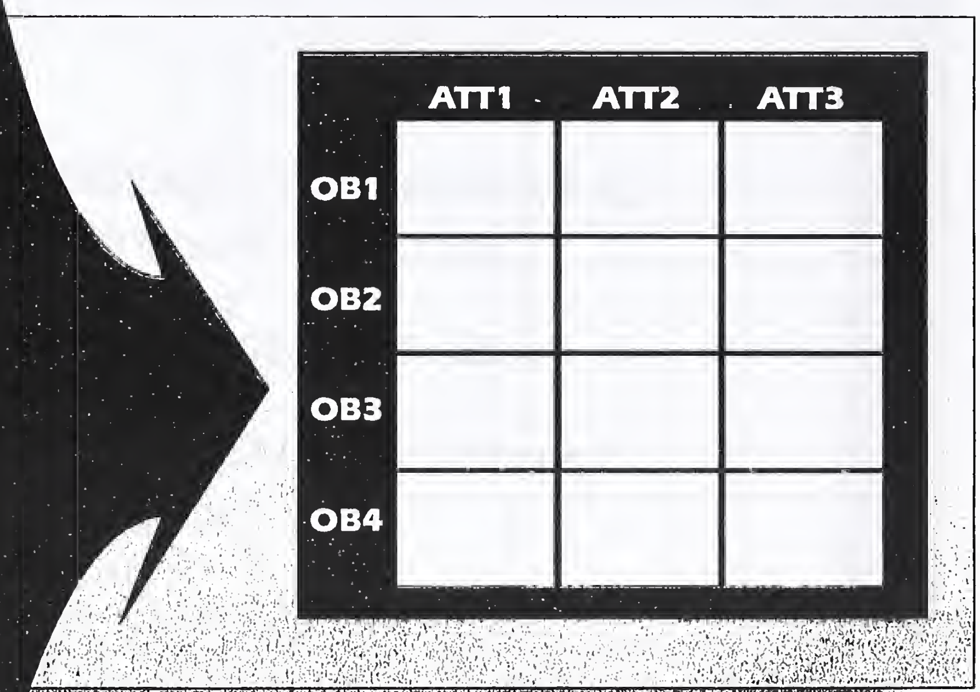
- TABULAR REPRESENTATION CAUSES INFLEXIBLE HARDCODED, INFLEXIBLE DESIGNS
- ANTICIPATE MANY CHANGES, THEREFORE, APPLYING OBJECT ORIENTED APPROACH USING RELATIONAL DATABASE
- GENERATE "STANDARD" TABLES TO ENHANCE QUERY PERFORMANCE

ATTRIBUTE TABLE
- COMPONENT TYPE ID
- ATTRIBUTE ID
- ATTRIBUTE NAME
- RANGE OF VALUES
- UNITS
- NOTES

COMPONENT TYPE TABLE
- COMPONENT TYPE ID
- COMPONENT TYPE NAME
- COMPONENT SUPERCLASS TYPE ID

COMPONENT TABLE
- COMPONENT TYPE ID
- MADE COMPONENT ID
- SUPPLIER ID
- SUPPLIER COMPONENT ID
- ETC.

COMPONENT ATTRIBUTE TABLE
- MADE COMPONENT ID
- ATTRIBUTE ID
- ATTRIBUTE VALUE



ISSUES

- SPEED VERSUS FLEXIBILITY
- STANDARDIZATION OF ELECTRO-MECHANICAL REPRESENTATIONS (STEP)
- MULTI-USE OF DATABASE DEPENDS UPON SCOPE OF REQUIREMENTS
- CONNECTIVITY WITH OTHER DATABASES
- SECURITY AND PERSISTANCE ON THE INTERNET
- UNITS

SAVE: Simulation Assessment Validation Environment

Carl Izurieta, Lockheed Martin (izzy@mar.lmco.com)

The summary for this presentation can be found on page 20
9 slides follow

Simulation Assessment Validation Environment

CDRL-A006
LMTAS-JSTPR-0040-05
DI-ADMN-81373/T
F33615-95-C-5538
FY 1133-95-05125

SAVE

Semi-Annual Review

30 May 1996

Distribution Statement C: Distribution authorized to U.S. Government Agencies and their contractors. Other requests for this document shall be referred to the PEO ...T.

Program Summary

- PROGRAM APPROACH:
  - INTEGRATE AND IMPLEMENT MODELING AND SIMULATION TOOLS INTO A VIRTUAL MANUFACTURING ENVIRONMENT TO REDUCE JSF LIFE CYCLE COST
- LOCKHEED MARTIN APPROACH:
  - APPLY COMMERCIAL OFF THE SHELF TOOLS
  - TIGHTLY COUPLE DESIGN TOOLS WITH COST ASSESSMENT TOOLS
  - USE EMERGING INTEGRATION TECHNOLOGIES FROM ON-GOING DARPA PROGRAMS
  - VALIDATE ON F-16 AND F-22 PROGRAMS
- AWARD DATE 4/95
- AWARD VALUE $8.8M
- CONTRACT DURATION 51 MO. + 3 MO. REPORT
- CUSTOMER WL/MTI - JAMES POINDEXTER
  JSF JPO - JON SCHREIBER/
  JIM BROWNING

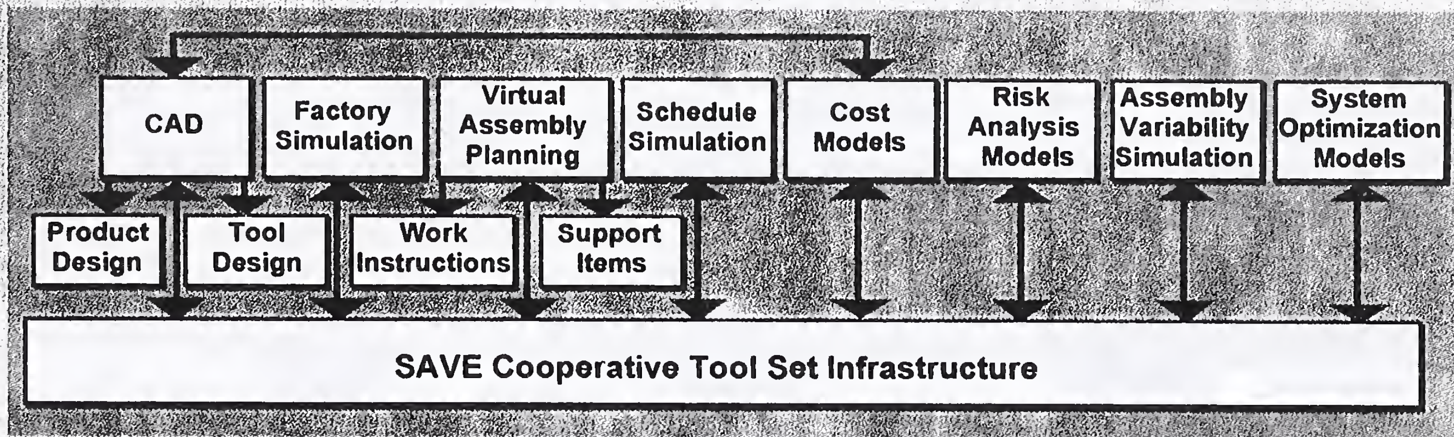
CDRL-A006
LMTAS-JSTPR-0040-06
DI-ADMN-81373/T
F33615-95-C-5538
FY 1133-95-05125

Program Objective

- **Provide an IP/PT environment that enables:**
  - Rapid assessment of manufacturing impacts of product/process decisions
  - Low risk, affordable transition of weapon system technologies from design to E&MD

CDRL-A006
LMTAS-JSTPR-0040-06
DI-ADMN-81373/T
F33616-95-C-5538
FY 1133-95-05125

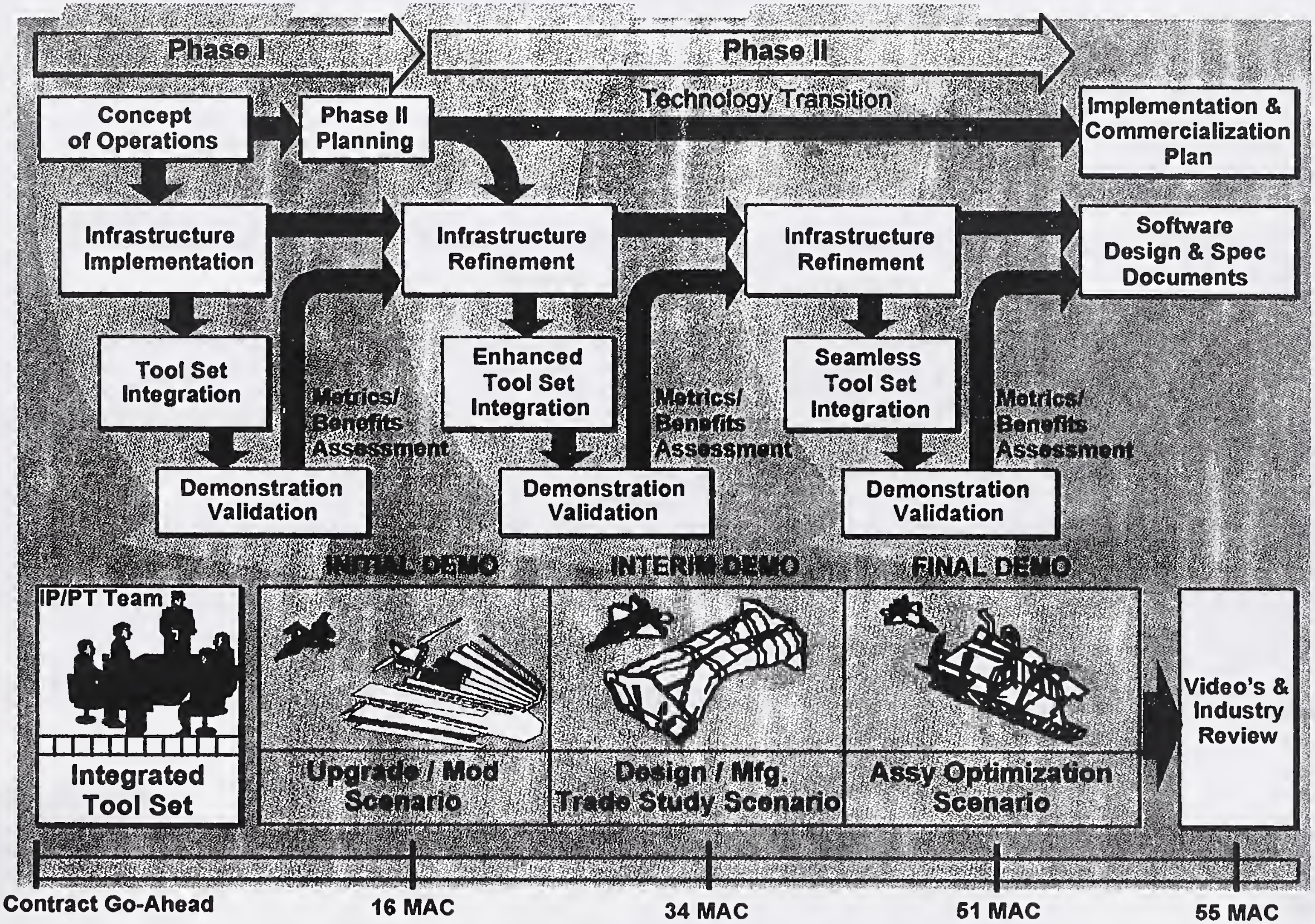
Program Approach



- Commercial Tools
- Infrastructure Technologies From DARPA Initiatives
- Tightly Coupled CAD-Cost Models
- Implementation and Best Practices

CDRL-A006
LMTAS-JSTPR-0040-05
DI-ADMN-81373/T
F33615-95-C-5538
FY 1133-95-05125

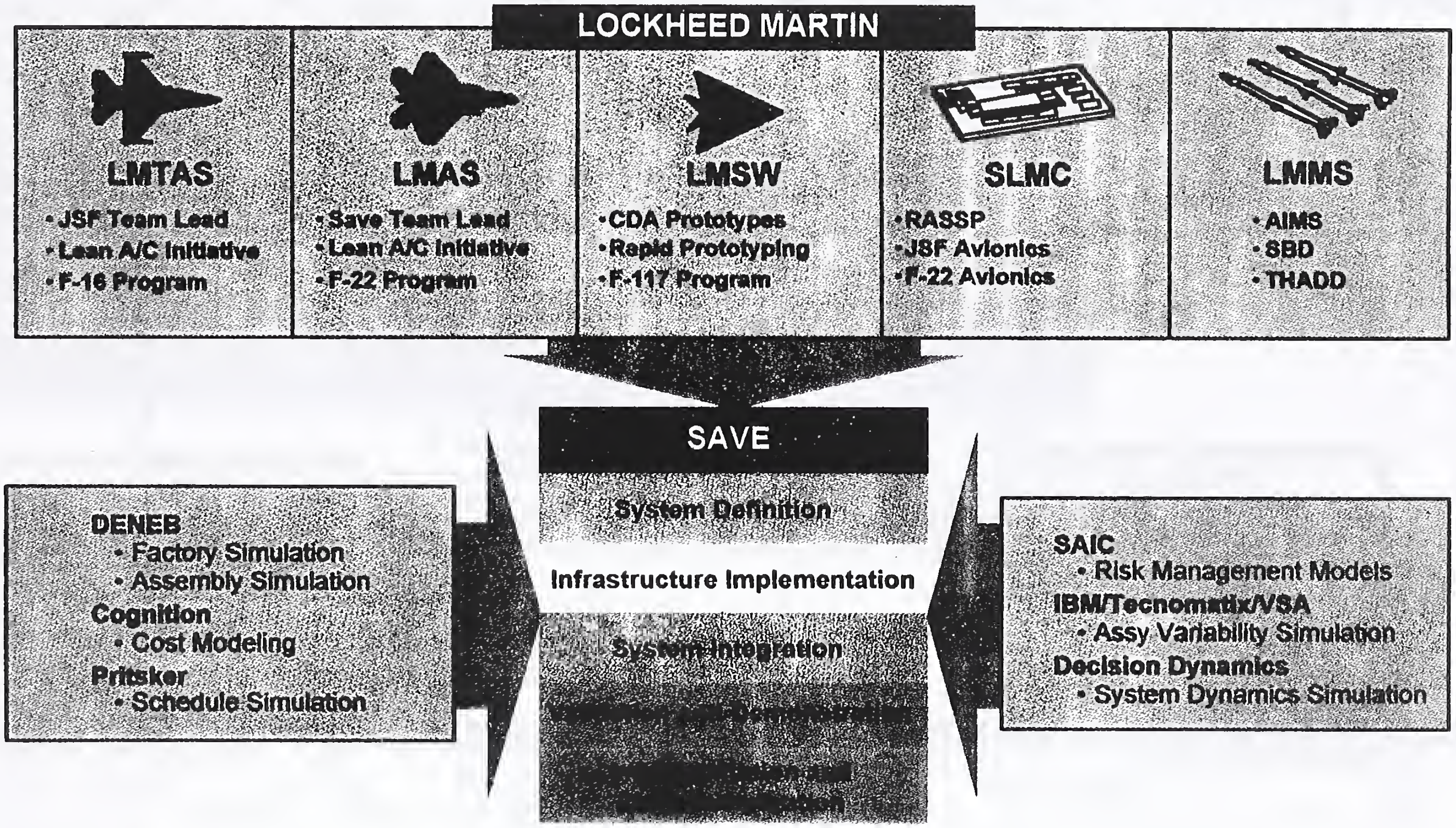
Program Plan



Simulation Assessment Validation Environment

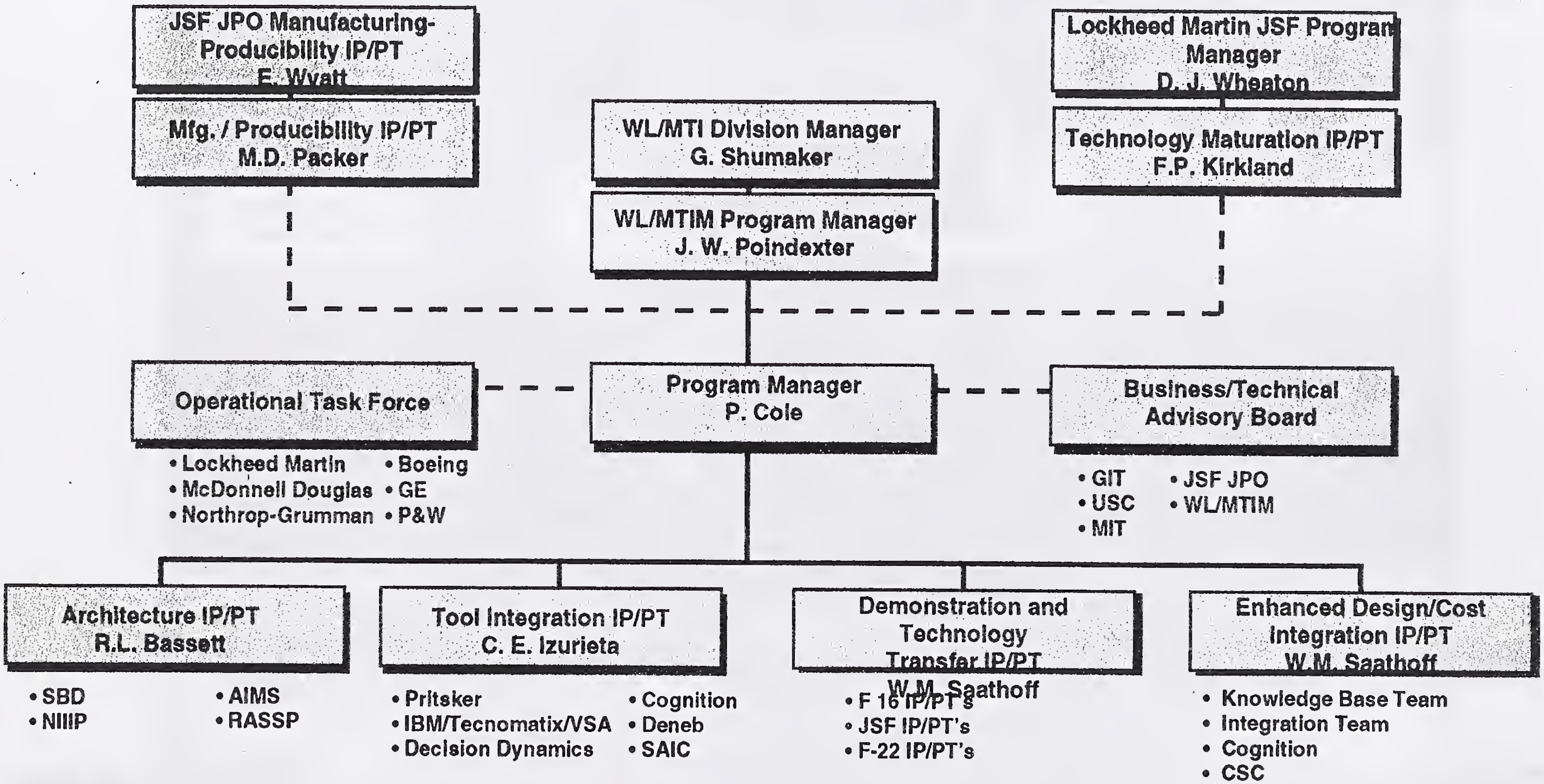
CDRL-A006
LMTAS-JSTPR-0040-05
DI-ADMN-81373/T
F33615-95-C-5538
FY 1133-95-05125

SAVE Team



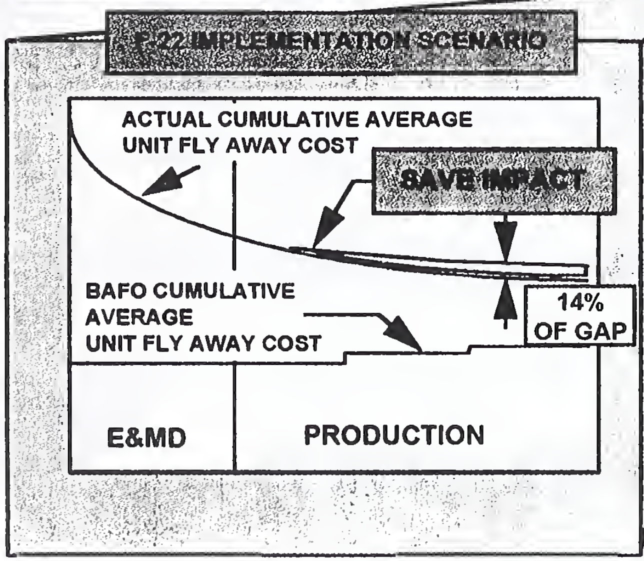
Simulation Assessment Validation Environment

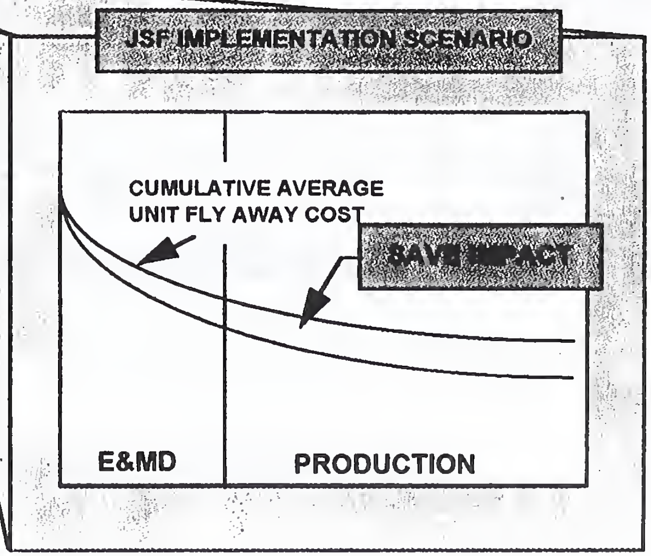
SAVE Program Organization



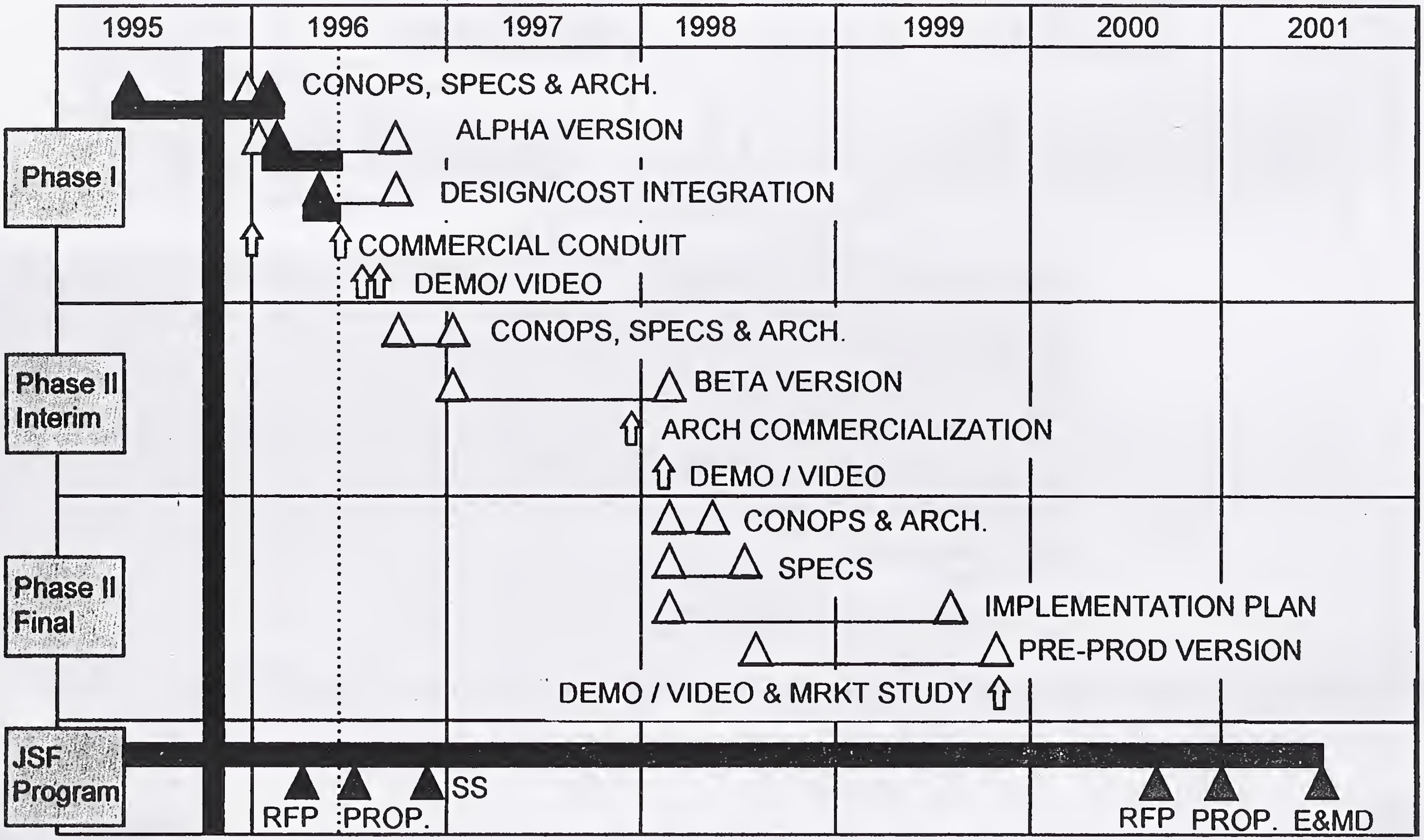
JSF Benefits

PRODUCT/PROCESS METRIC	SAVE IMPACT TO METRIC (%) F-22	SAVE IMPACT TO METRIC (%) JSF	UNIT COST SAVINGS ($K) F-22	UNIT COST SAVINGS ($K) JSF	JSF LCC SAVINGS ($B)
Design to Cost Data Accuracy	25	12	109	237	1.23
Lead Time Reduction	N/A	10	N/A	156	.13
Design Change Reduction	15	28	326	325	.83
Scrap Rework & Repair Reduction	15	11	80	200	.51
Process Capability	10	5	97	107	.19
Inventory Turn Increase	5	2	25	23	.02
Fab & Assy Inspection Reduction	13	6	79	85	.08
TOTAL			716	1,100	3B (1.97%)





Product Schedule



Integrated Product Data Environment (IPDE)

Dr. Ed Harter, Boeing (edward.d.harter@boeing.com)

The summary for this presentation can be found on page 23
39 slides follow

NIST Design Workshop
November 19, 1996
Gaithersburg, Maryland

Integrated Product Data Environment
IPDE

Edward D. Harter
Boeing Defense & Space Group
(206) 773-1982
edward.d.harter@boeing.com

IPDE_01

The MacNeal-Schwendler Corporation

General Information

- Part of the DARPA Manufacturing Automation & Design Engineering (MADE) Program
- IPDE Partners
  - Boeing Defense & Space Group, Engineering ..... Edward Harter
  - Arizona State University, Engineering .... Jami Shah
  - MacNeal-Schwendler Corporation .... Thomas Mack
- Period of Performance
  - May, 1996 - May, 1998
- Goal of MADE
  - To improve our ability to evaluate design alternatives
- Goal of IPDE
  - To improve the engineering design & analysis process through the integration of information

DARPA BOEING ASU The MacNeal-Schwendler Corporation

Part 1
Objectives & Strategies

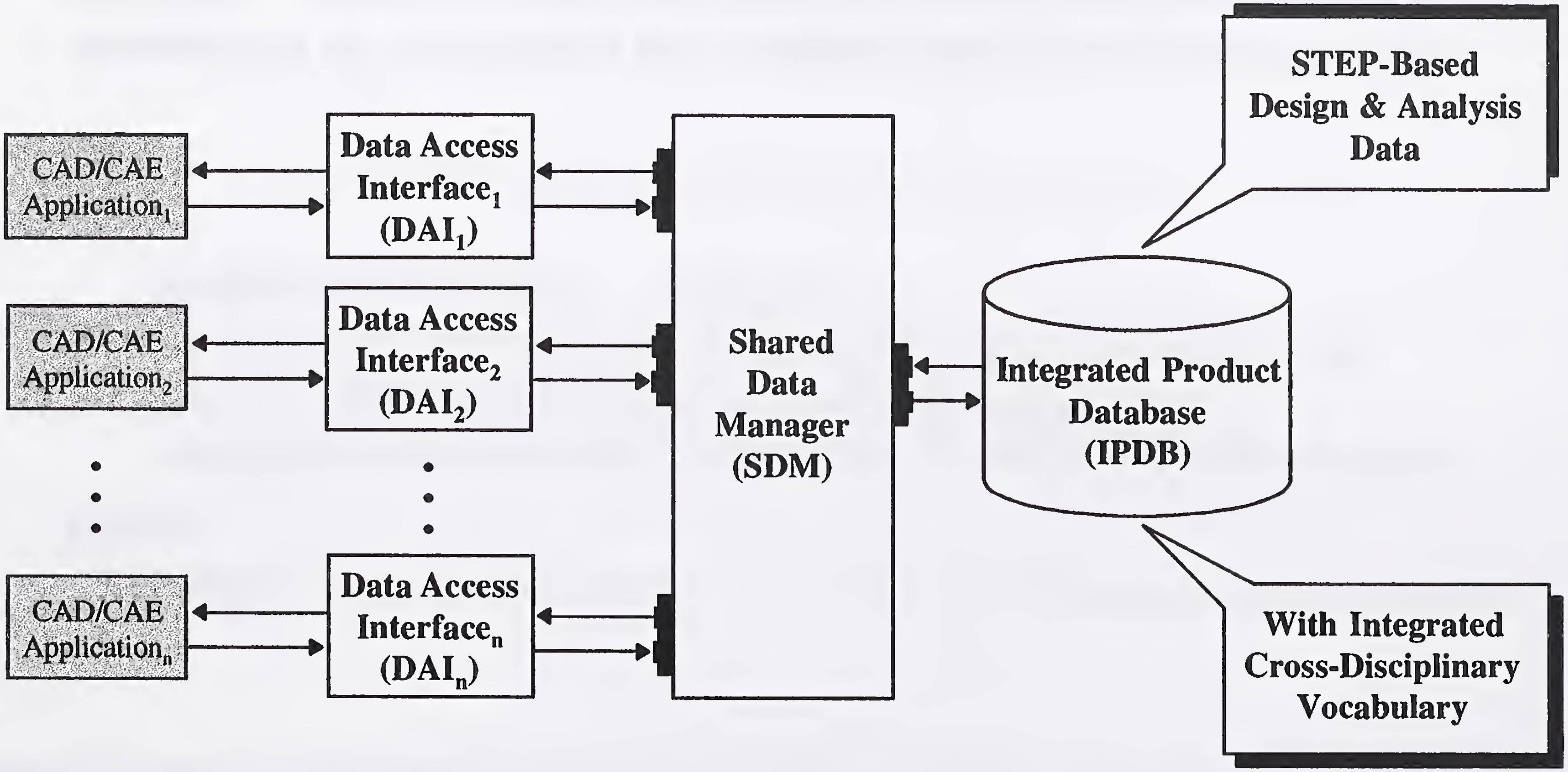
Objectives

- **Integration of Technical Data:** **Develop a Common Information Model to Enable**
  - **STEP-Based Database Implementation of Design and Analysis Data**
  - **Data Sharing Among Design and Analysis Disciplines**
  - **Configuration Management of "In-Work" Data Throughout the Design/Analysis Cycle**

- **Integration of Vocabulary for Complex Electromechanical (CEM) Systems:** **Addresses design representation for retrieval and reuse of partial design fragments**

The MacNeal-Schwendler Corporation

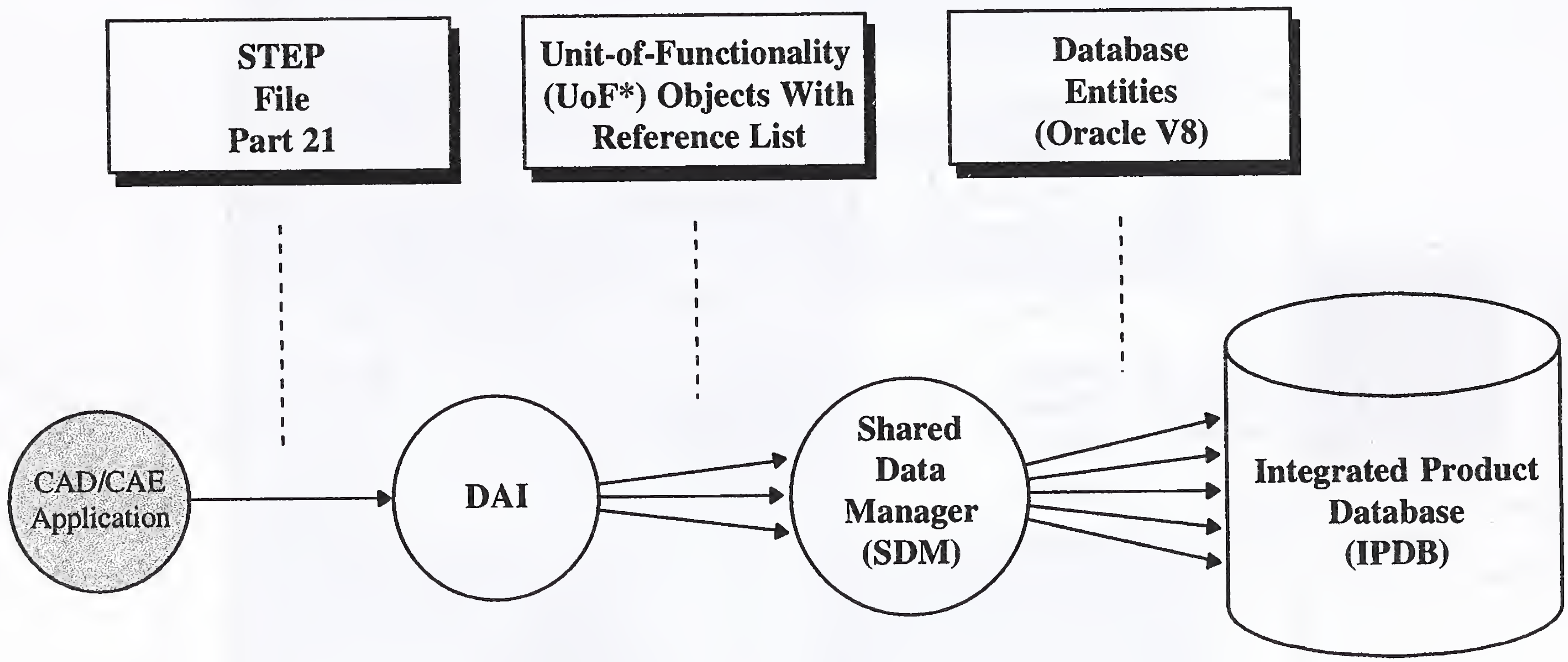
IPDE Vision



IPDE_05

DARPA BOEING ASU The MacNeal-Schwendler Corporation

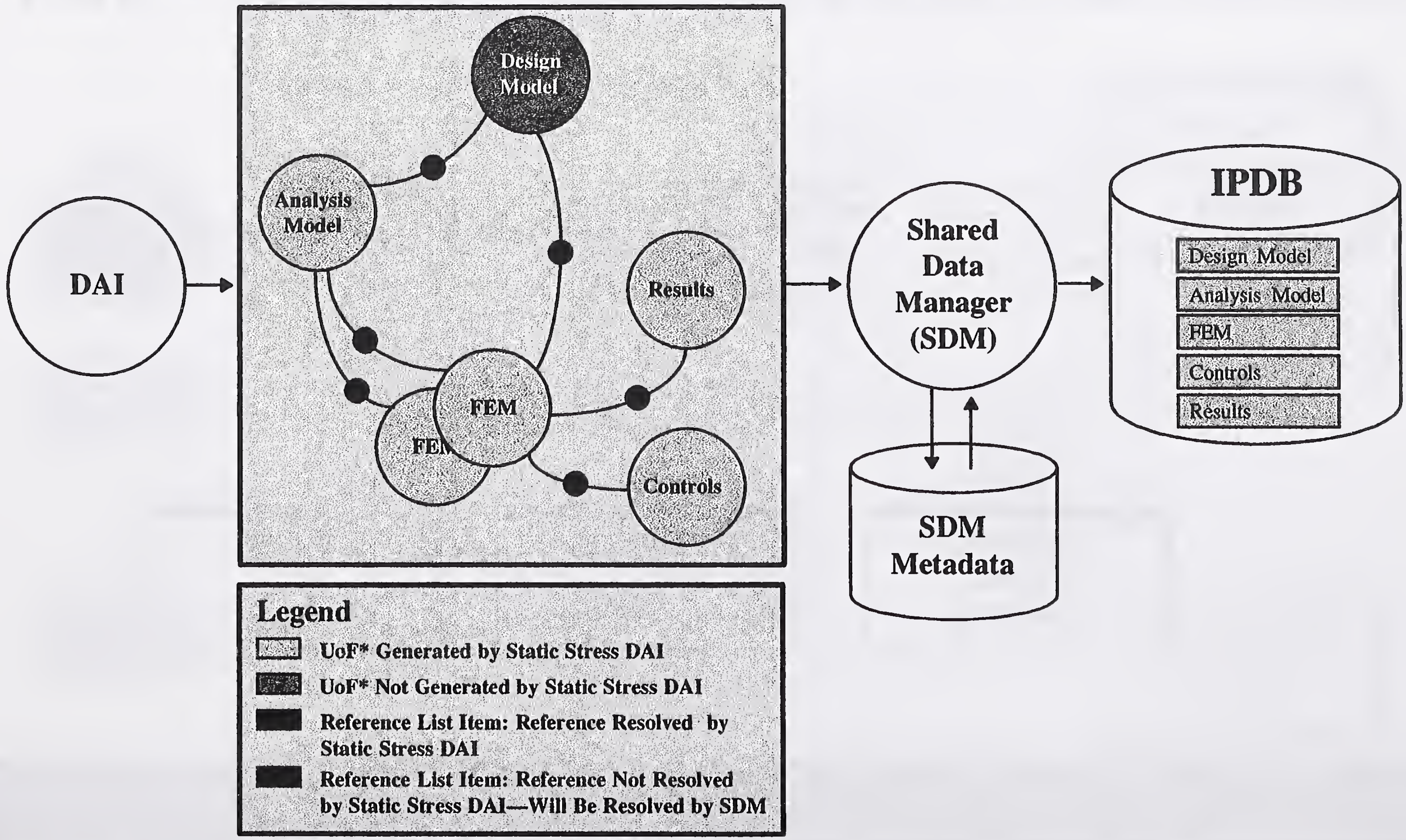
Data Manipulation



UoF* Concept Borrowed From STEP . . . Might Be Modified In This Implementation

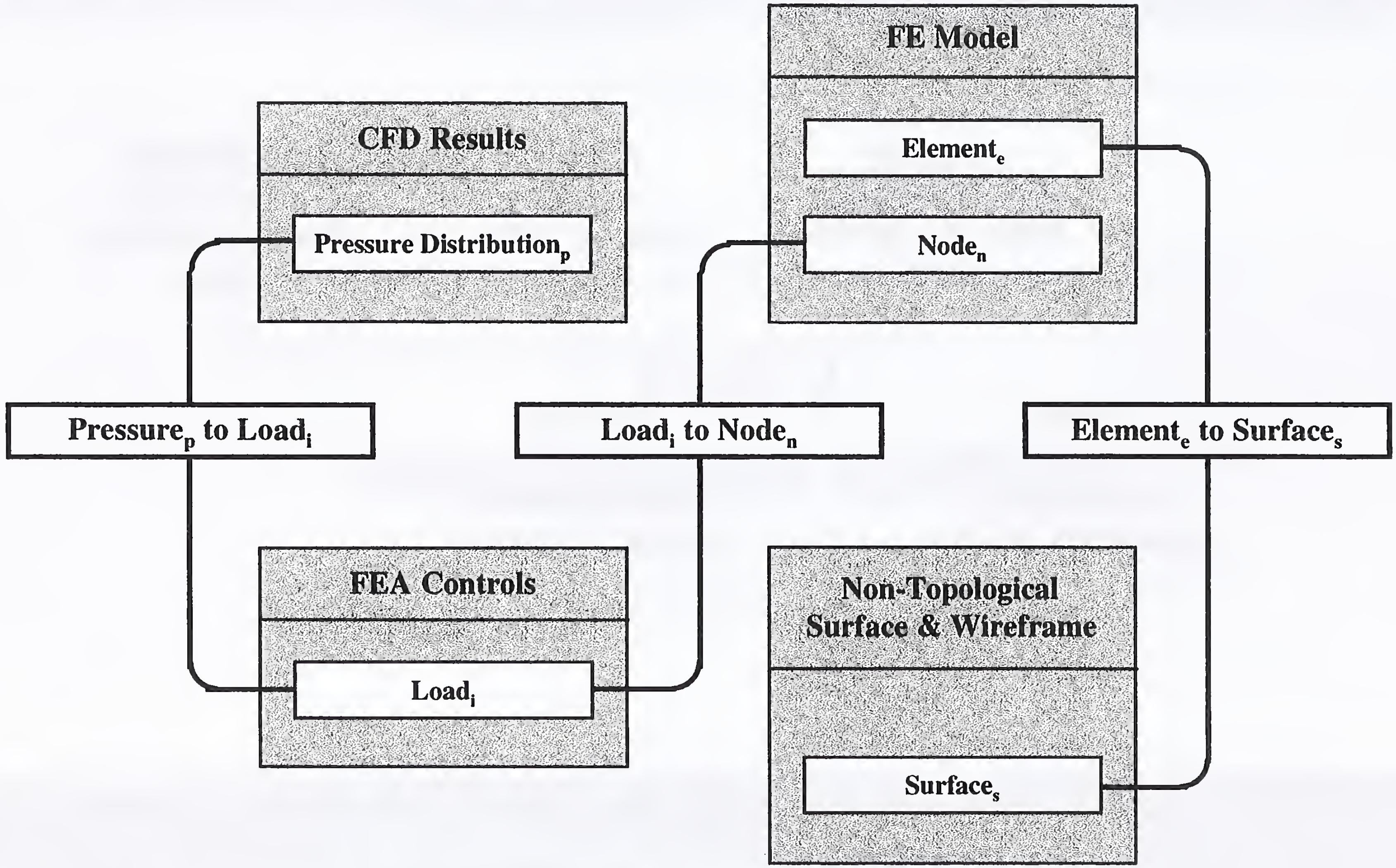
DARPA BOEING ASU The MacNeal-Schwendler Corporation

DAI Input & Output: Static Stress Example



The MacNeal-Schwendler Corporation

UoF*-to-UoF* References (UtUs) in IPDB



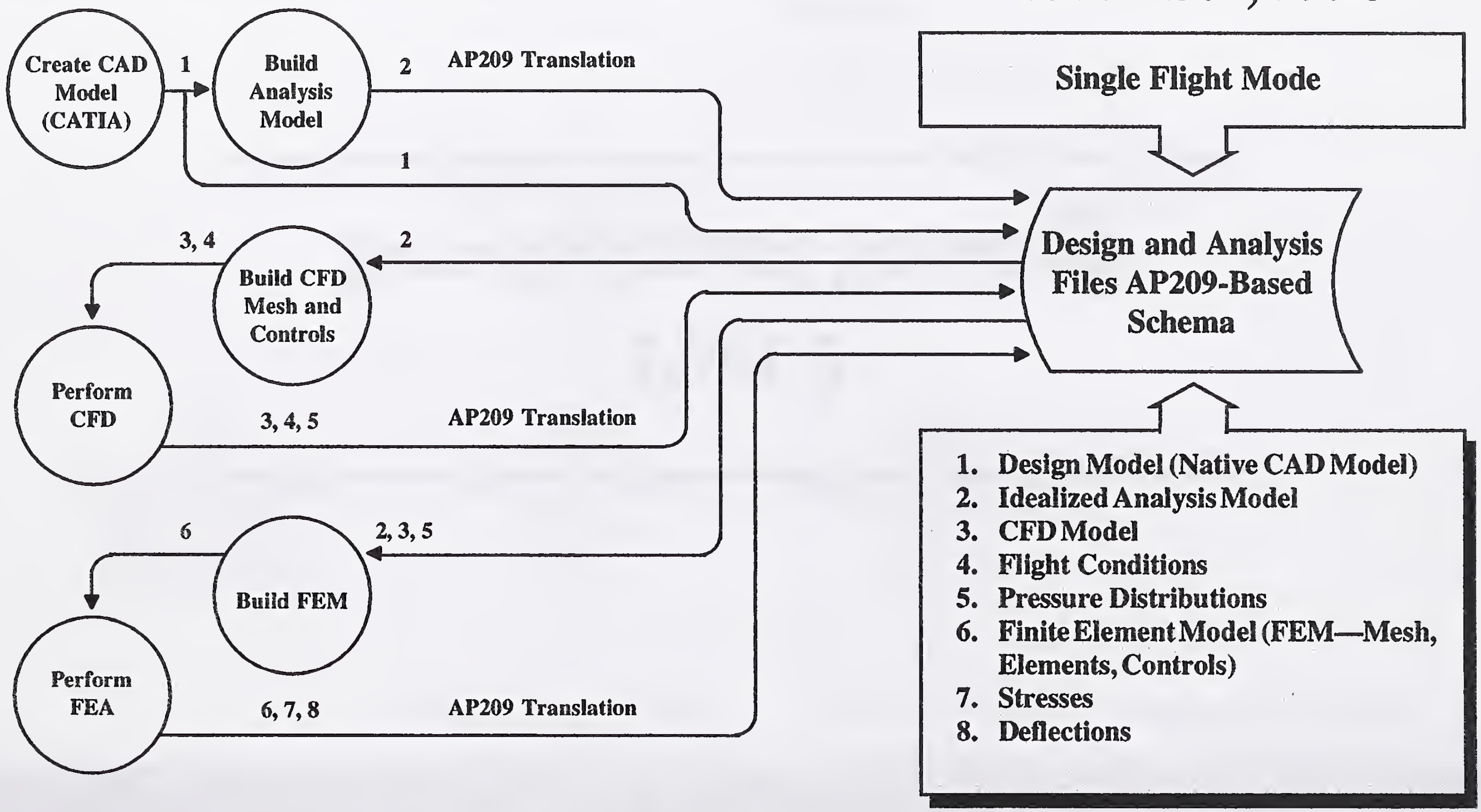
IPDE Work Breakdown

- **IPDE** «Boeing»
  - **Design & Analysis Data Integration** — Low-Level, Narrow Scope «Boeing»
    - **IPDB** — Schema & Database Development «Boeing/ASU»
    - **SDM** — Access & Control Services «ASU»
    - **DAIs** — Application Interfaces to SDM «Boeing»
      - **Aerodynamic Flow Analysis** «Boeing»
      - **Finite Element Analysis** «MSC»
      - **Manufacturability Evaluation** «ASU»
  - **CEM Vocabulary Integration** — High-Level, Broad Scope «ASU»

Part 2
Development Strategy

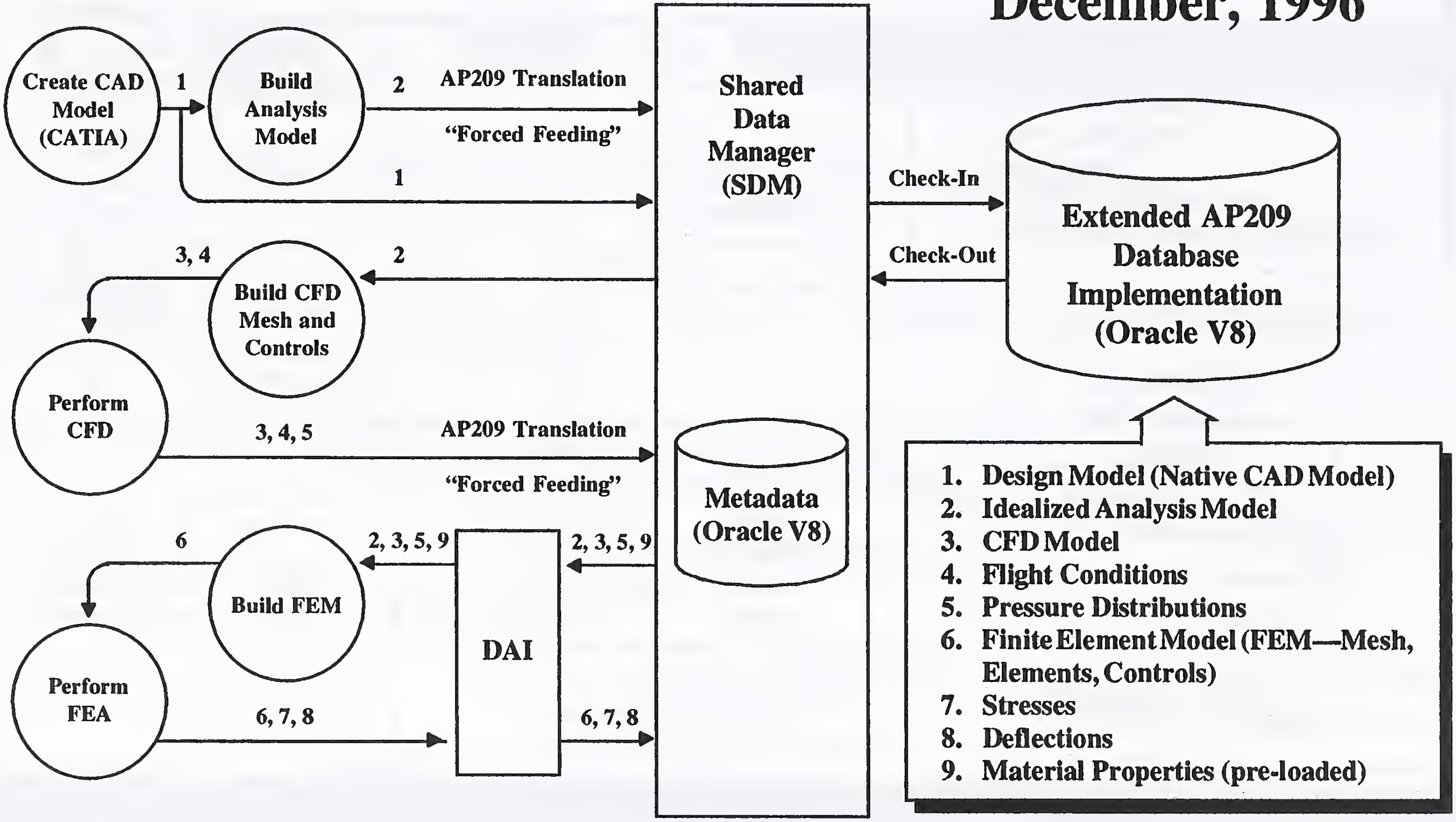
Prototype 1A

November, 1996



Prototype 1B

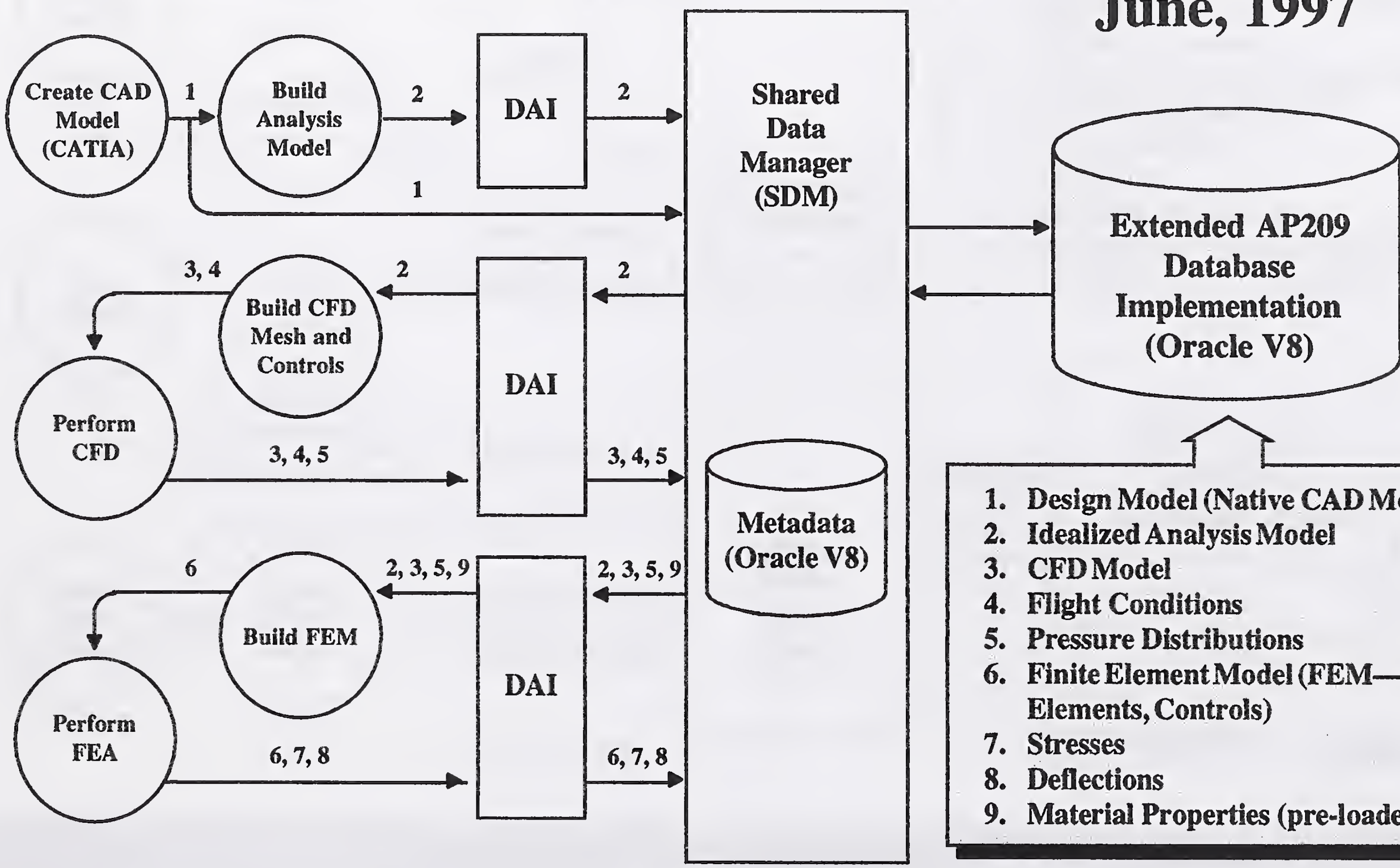
December, 1996



IPDE_12

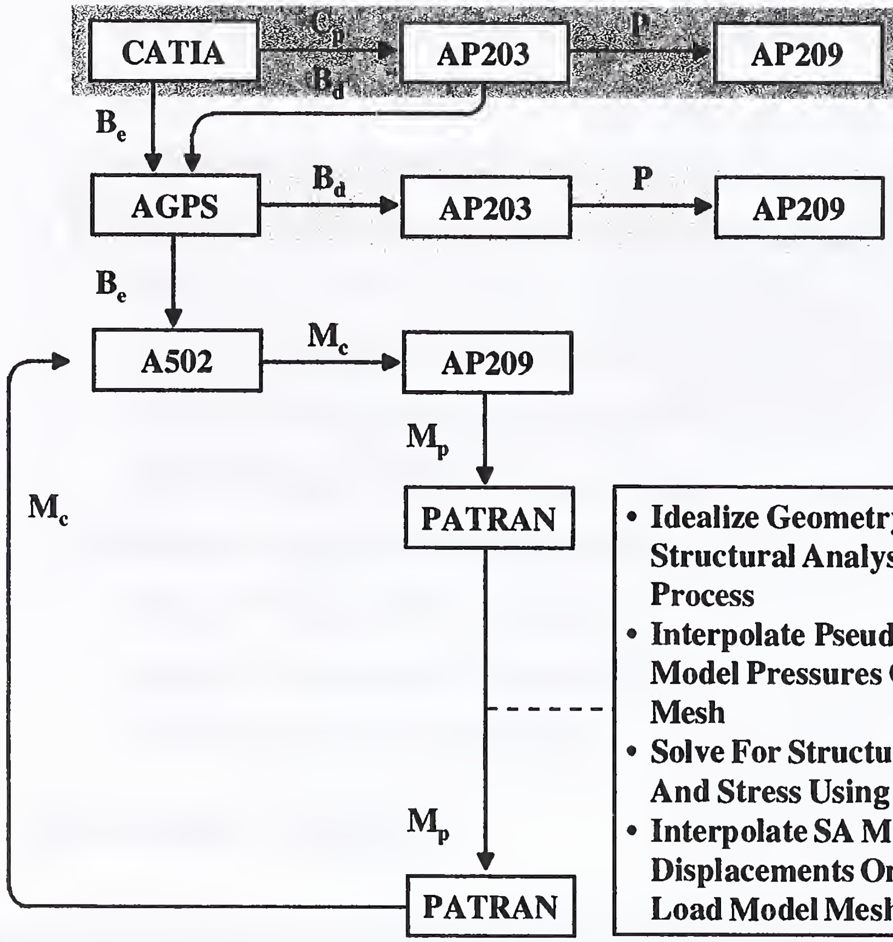
IPDE Version 1

June, 1997



1. Design Model (Native CAD Model)
2. Idealized Analysis Model
3. CFD Model
4. Flight Conditions
5. Pressure Distributions
6. Finite Element Model (FEM—Mesh, Elements, Controls)
7. Stresses
8. Deflections
9. Material Properties (pre-loaded)

Prototype 1A Detail



Design Geometry

Idealized Geometry

Pseudo-A502 FEM Load Model

- Idealize Geometry To Support Structural Analysis (SA) FEM/FEA Process
- Interpolate Pseudo-A502 FEM Load Model Pressures Onto SA FEM Mesh
- Solve For Structural Displacements And Stress Using SA FEM Model
- Interpolate SA Model Displacements Onto A502 FEM Load Model Mesh

Legend

B_d	Boeing New Development
B_e	Boeing Enhancement
C_p	CATIA Product
M_c	MSC Customized Translator
M_p	MSC Commercial Product
P	PDES, Inc., Context Mapper

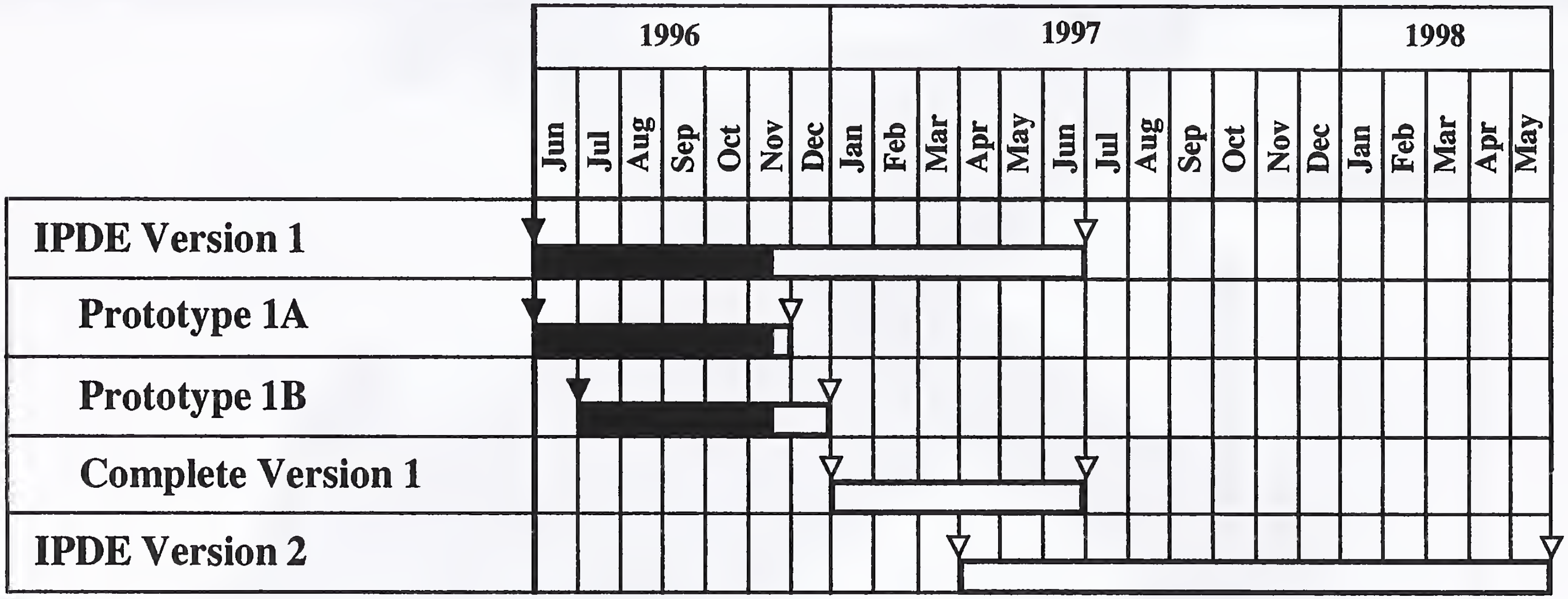
The MacNeal-Schwendler Corporation

Prototype 1A Status

Executable Module	Who	Status
• Bridge from CATIA to AGPS	Boeing	In Test
• AP203 to AGPS Data Translator	Boeing	In Test
• AGPS to AP203 Data Translator	Boeing	In Test
• AGPS to A502 (Enhanced Surface Paver)	Boeing	In Test
• A502 to AP209 FEM Data Translator	MSC	In Test
• AP209 to A502 FEM Data Translator	MSC	In Test
• AP209 to PATRAN.db FEM Data Translator (Load Cases Only)	MSC	In Progress
• AP209 to PATRAN.db FEM Data Translator	MSC	In Progress
• PATRAN.db to AP209 Data Translator	MSC	Complete
• AP209 to PATRAN.exp for Geometry	MSC	In Progress
• PATRAN.exp to AP209 for Geometry	MSC	In Progress
• PATRAN.db to PATRAN.exp for Geometry	MSC	In Progress
• File Append Script for Appending Independent Repository Models	MSC	In Progress
• File Append Script for Appending Pressures/Displacements in FEM Load	MSC	In Progress

The MacNeal-Schwendler Corporation

Program Roadmap



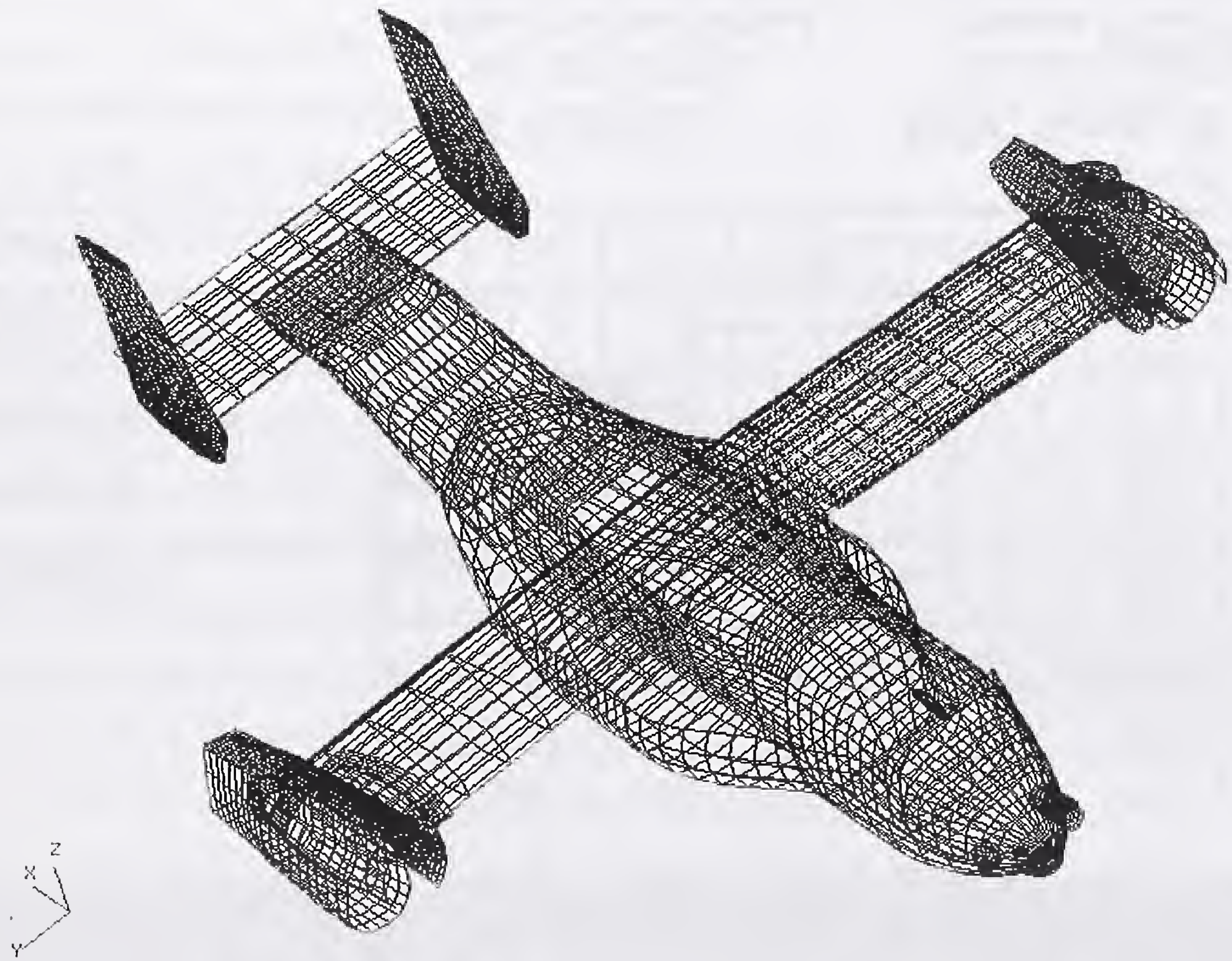
IPDE Version 1
- CAD, CFD, & FEA
- Data Structure Defined
- Database Implemented
- DAIs Developed
- Access Control
- Integrity Constraints

IPDE Version 2
- Manufacturability Evaluation
- Multiple Design Iterations
- Version Management
- Change Notification
- Workflow Management
- CEM Vocabulary Integration

IPDE_16

The MacNeal-Schwendler Corporation

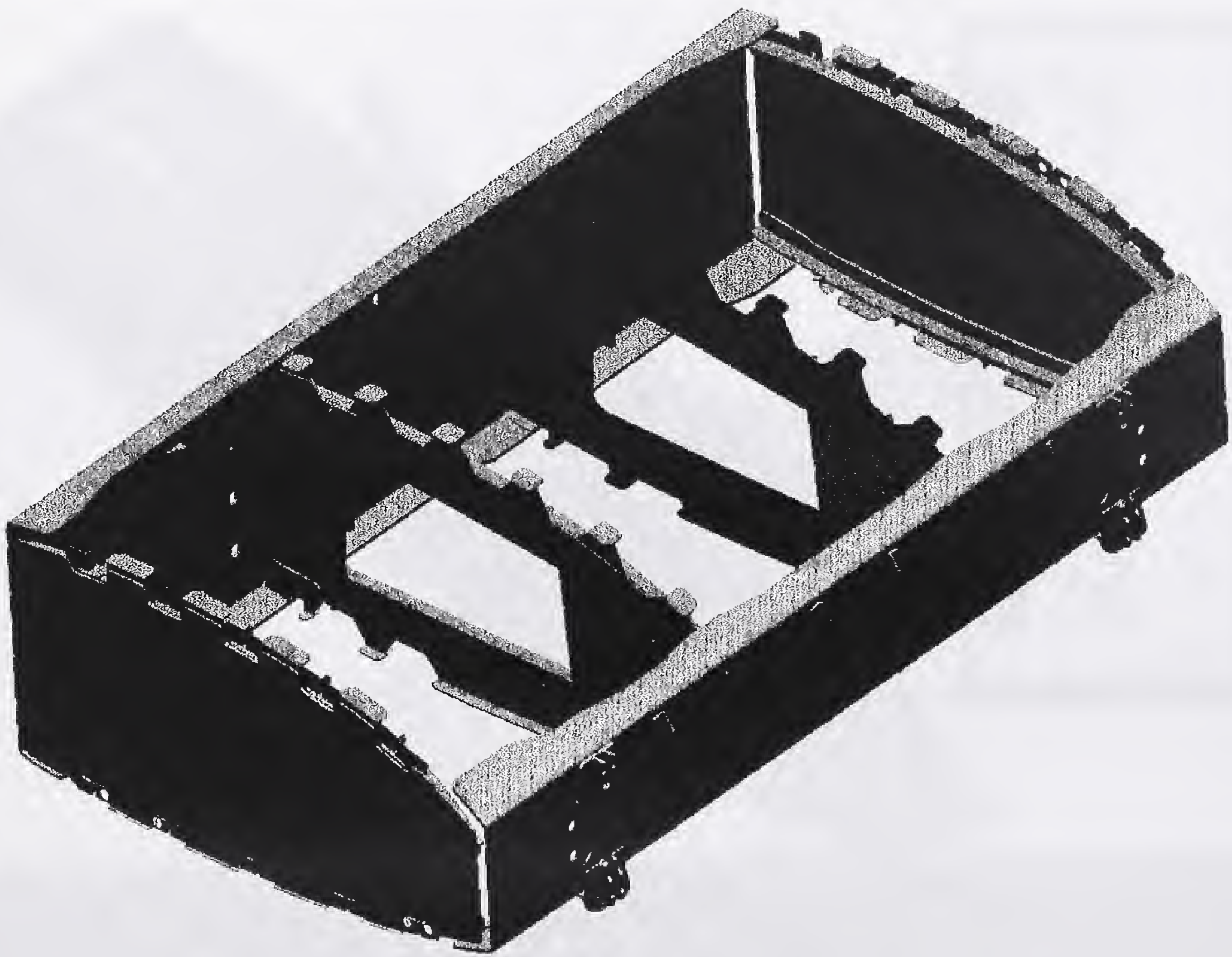
Demonstration Article: V-22 Wing



IPDE_17

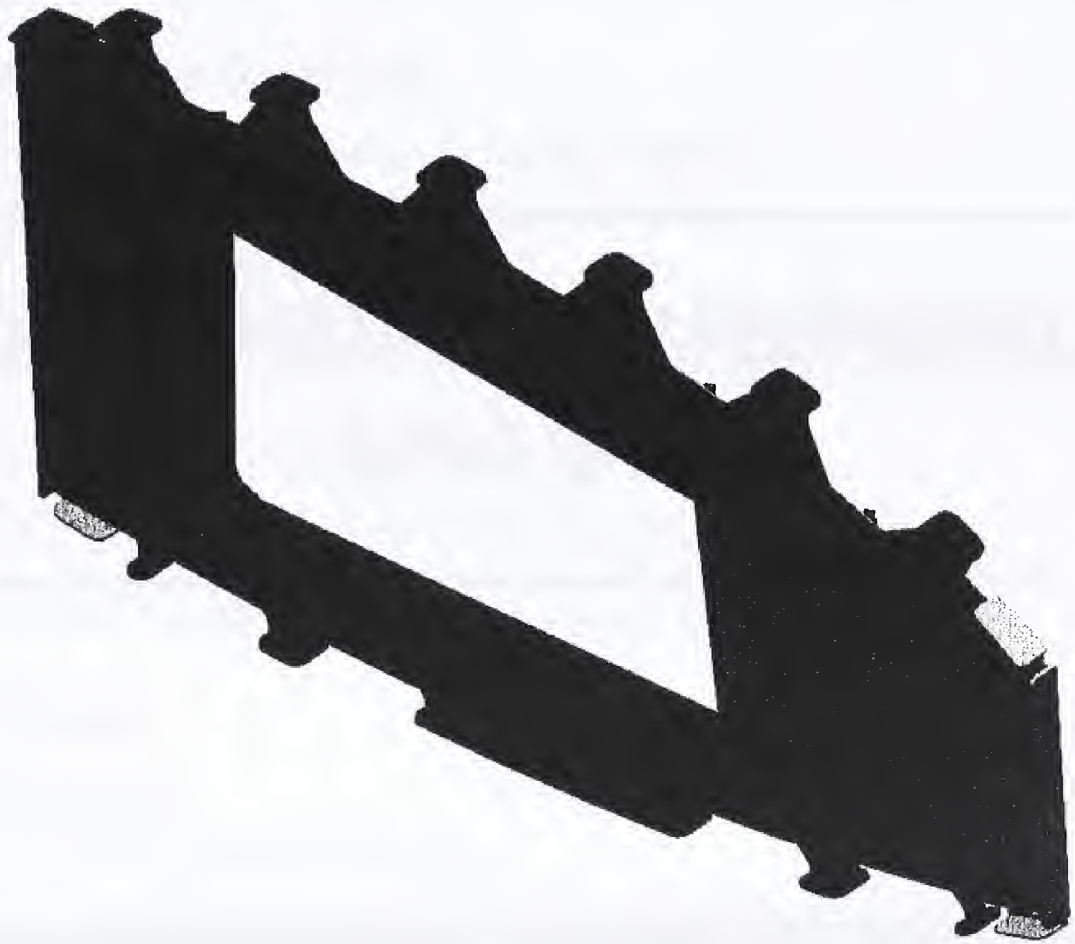
V-22 Wing CFD Pressure Distributions

V-22 Wingbox Internal Structure



The MacNeal-Schwendler Corporation

Rib For Manufacturability Evaluation



IPDE Program

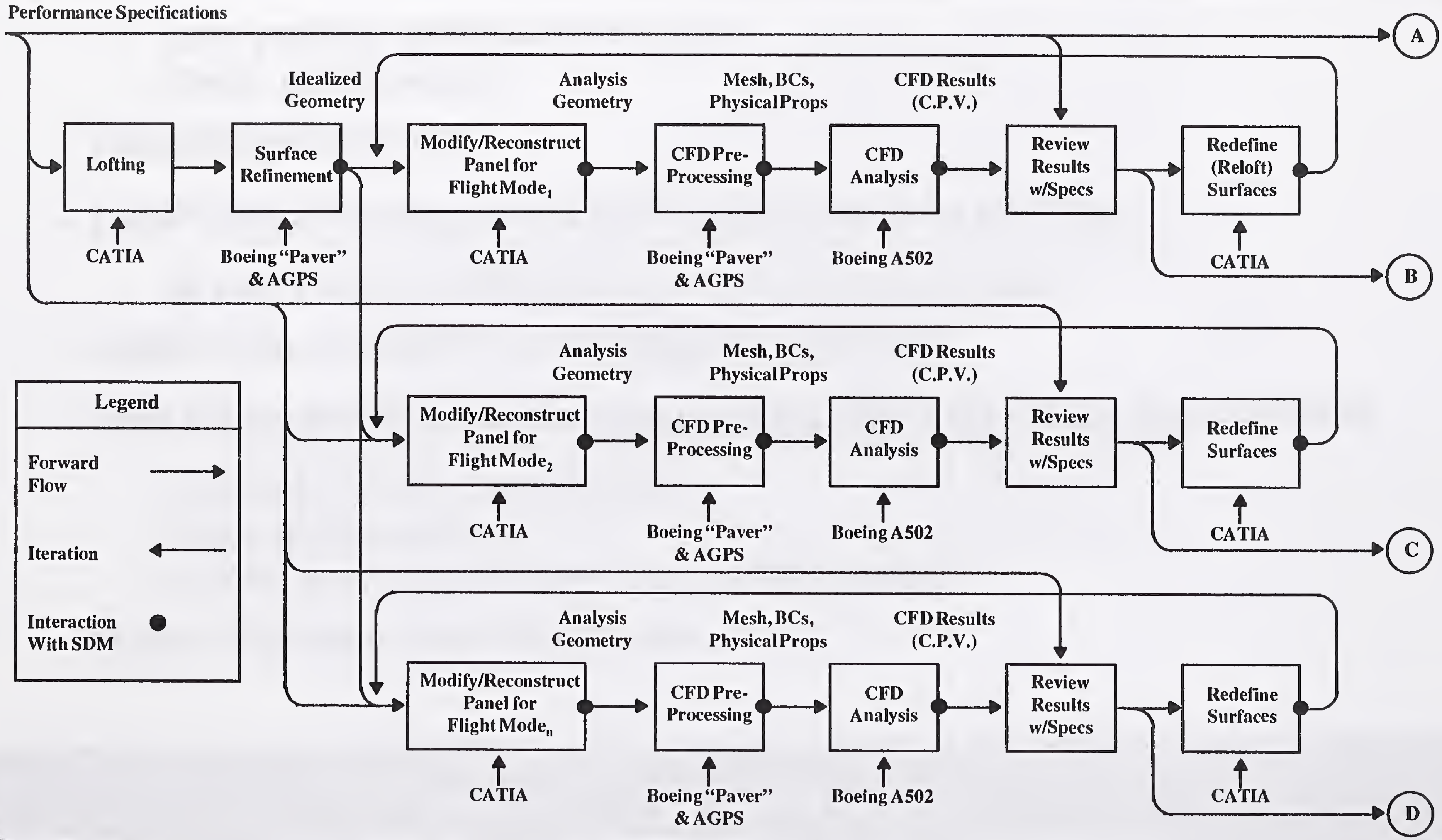
Part 3
IPDB Database Development

The MacNeal-Schwendler Corporation

Design Methodology

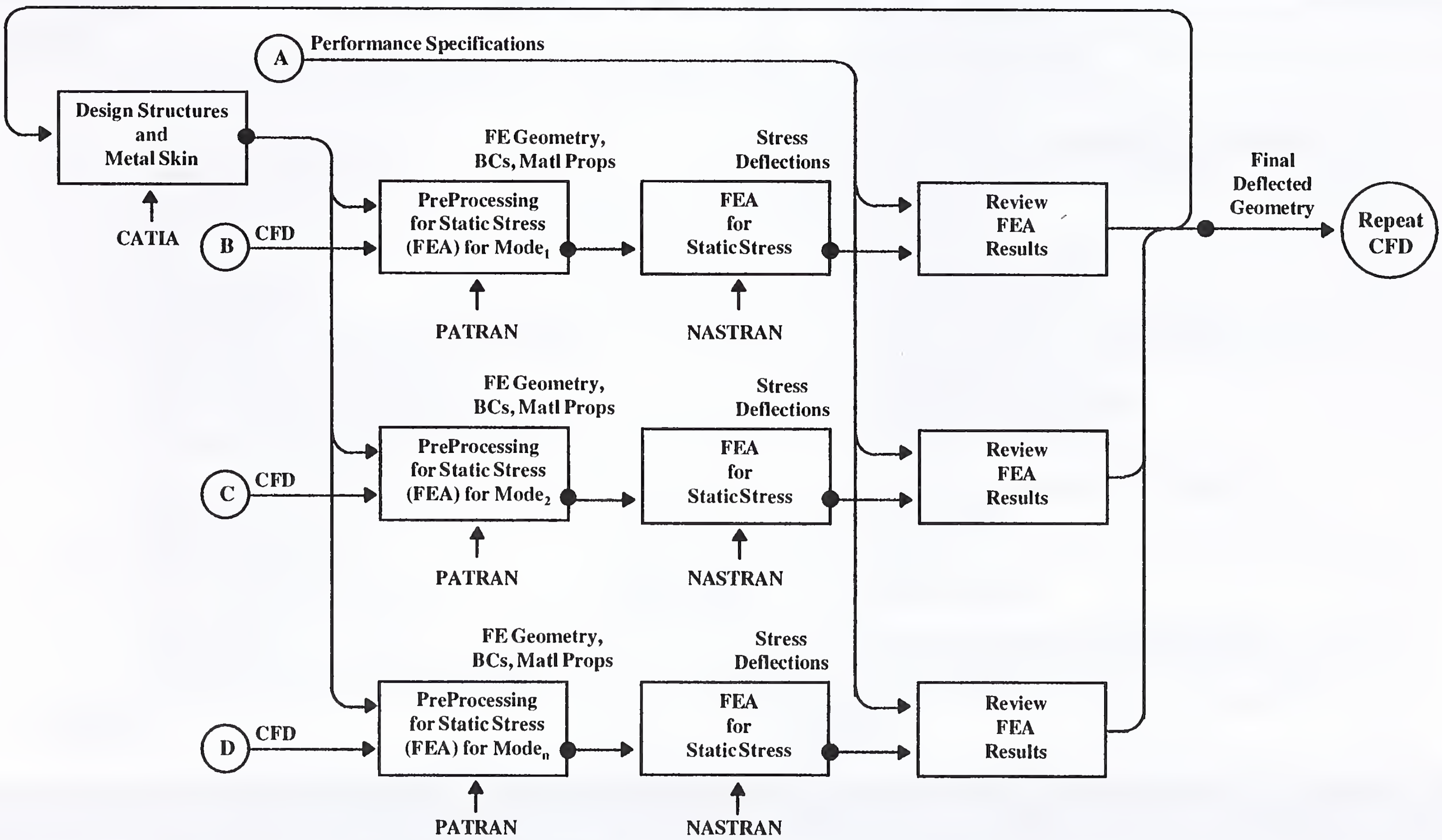
- Define Information Requirements
  - Process Models, Data Flows, and Usage Scenarios
  - Query Requirements
  - Persistent Storage Requirements
- Map Information Requirements to STEP APs, UoFs, and Other Sources
- Define UoF*s as Binary Large Objects (BLOBs)
  - BLOBs Used by Applications But Not Queried by IPDE
- Define UoF*-to-UoF* References (UtUs) for Data Integrity
- Develop Metaschema
  - Query Requirements
  - Data Integrity Requirements (UtUs)
- Develop Database Schema

Process & Data Flow, Slide A



IPDE_23

Process & Data Flow, Slide B



IPDE_24

The MacNeal-Schwendler Corporation

STEP UoFs

- **AP203**
  - √ **advanced_boundary_representation**
  - √ **authorization**
  - _ bill_of_material
  - √ **design_activity_control**
  - √ **design_information**
  - _ effectivity
  - √ **end_item_identification**
  - _ facetted_boundary_representation
  - _ manifold_surface_with_topology
  - √ **non_topological_surface_and_wireframe**
  - √ **part_identification**
  - √ **shape**
  - _ source_control
  - √ **wireframe_with_topology**

- **AP209**
  - √ **activity_control**
  - √ **advanced_boundary_representation**
  - √ **analysis_report**
  - _ assembly
  - √ **authorization**
  - _ composite_constituents_representation
  - _ effectivity
  - √ **end_item_identification**
  - √ **environment**
  - _ facetted_boundary_representation
  - √ **FE_analysis_control**
  - √ **FE_analysis_results**
  - √ **FEA_model**
  - _ manifold_surface_with_topology
  - √ **non_topological_surface_and_wireframe**
  - _ part_composite_constituents
  - √ **part_identification**
  - _ part_laminate_table
  - √ **part_shape**
  - √ **wireframe_with_topology**
  - _ zone_composite_constituents_and_their_reps

Legend

- **Checkmark (√) denotes likely candidate for IPDE**
- **Red font denotes overlap among existing APs**

STEP UoFs

- **AP214**
  - _ advanced_3D_presentation
  - _ advanced_draughting_presentation
  - √ **advanced_mass_properties**
  - √ **advanced_tolerances**
  - _ assembly_structure
  - _ associtive_draughting
  - √ **B_rep_model**
  - _ basic_kinematics
  - √ **basic_mass_properties**
  - √ **basic_material_properties**
  - √ **basic_product_management_data**
  - _ basic_surface_conditions
  - _ basic_surface_presentation
  - √ **basic_tolerances**
  - _ basic_wireframe_presentation
  - _ complete_product_structure
  - _ compound_B_rep_model
  - √ **CSG_model**
  - _ explicit_draughting
  - _ explicit_draughting_presentation
  - _ external_reference_mechanism
  - _ facetted_B_rep_model
  - √ **FEA_control**
  - √ **FEA_material_properties**
  - √ **FEA_mesh**
  - √ **FEA_result**
  - _ multiple_viewing
  - _ single_viewing
  - √ **surface_model**
  - √ **wireframe_model_2D**
  - √ **wireframe_model_3D**

Legend

- **Checkmark (√) denotes likely candidate for IPDE**
- **Red font denotes overlap among existing APs**

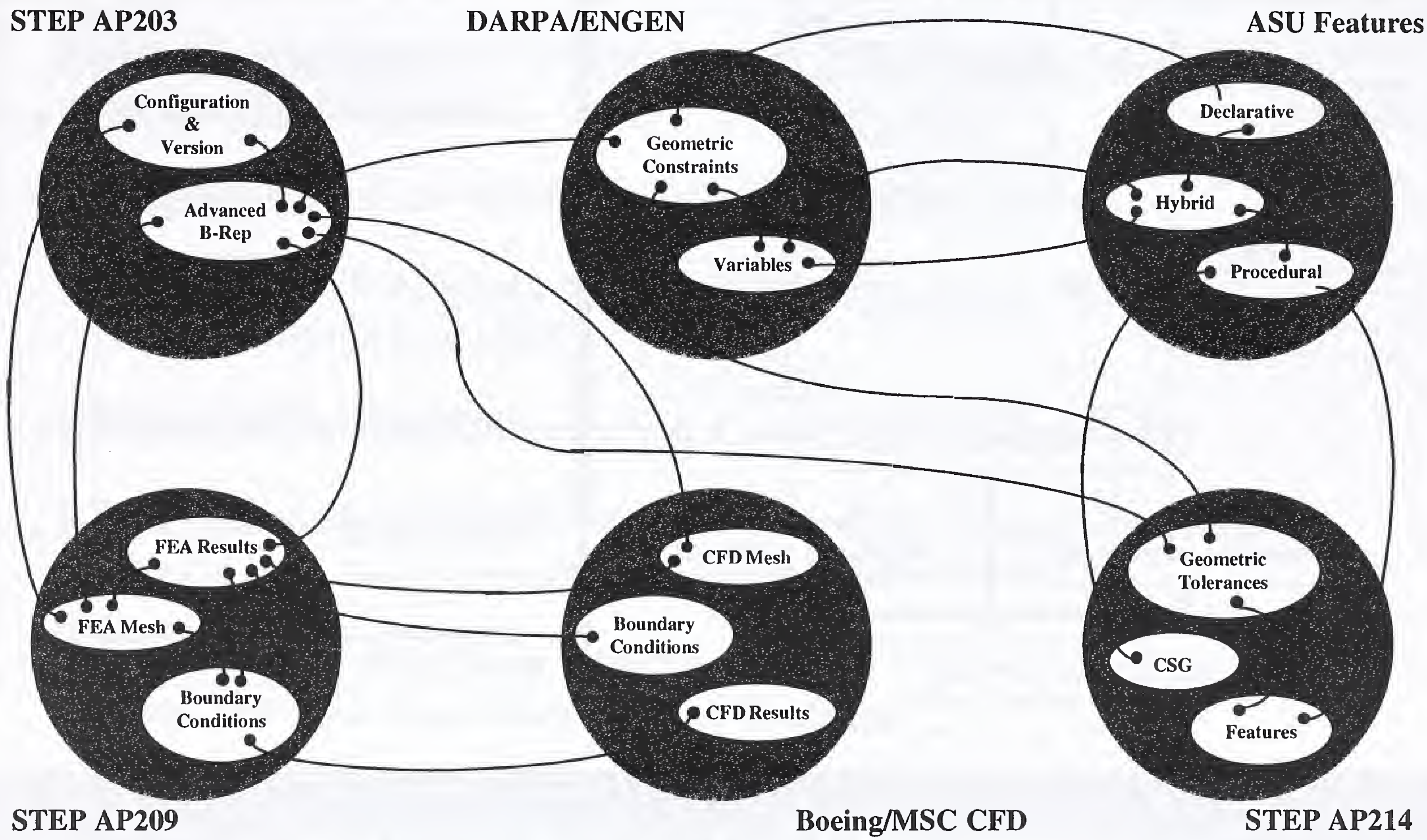
DARPA BOEING ASU The MacNeal-Schwendler Corporation

IPDB Schema

- Draws From Six Topical Domains
  - Design Model
  - FEA Static Stress
  - CFD
  - Features
  - Geometric Tolerances
  - Geometric Relations
- Employs the Unit-of-Functionality (UoF*) Concept
- Creates New Associations Between UoF*s in Each Model Needed for the Engineering Analysis Cycle
- Based on STEP Standards & STEP-Like Extensions
  - for CFD (Boeing, MSC)
  - for Constraints (DARPA/ENGEN)
  - for ASU Feature Model (STEP & ENGEN-based)
- Explores the Extensibility & Interoperability of STEP APs

The MacNeal-Schwendler Corporation

Source Schemata

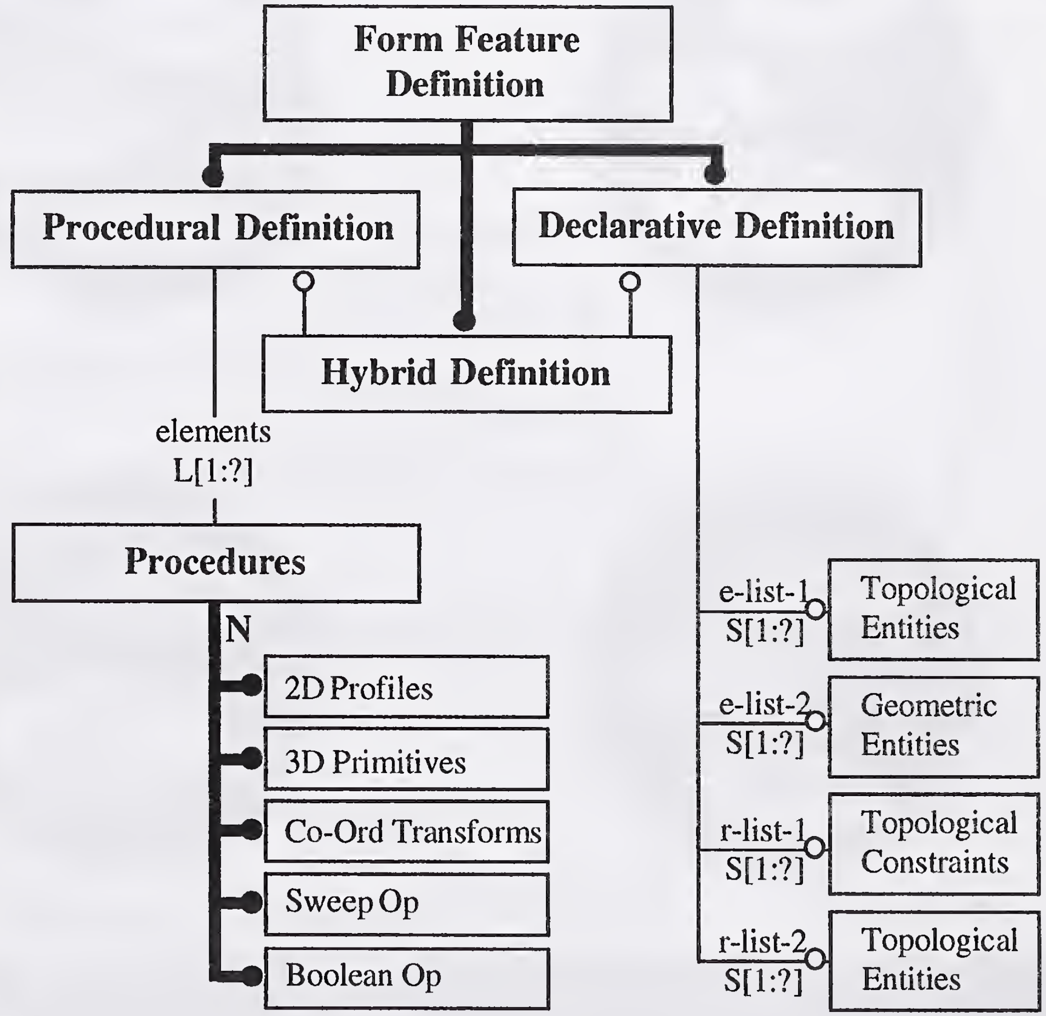


IPDE_28

DARPA BOEING ASU The MacNeal-Schwendler Corporation

Neutral Form Feature Model

- Support Design-by-Features
- Support Feature Recognition
- Support Feature Definition
  - Procedural
  - Declarative
  - Hybrid
- Based On STEP-AP203 and DARPA/ENGEN Data Models



DARPA BOEING ASU The MacNeal-Schwendler Corporation

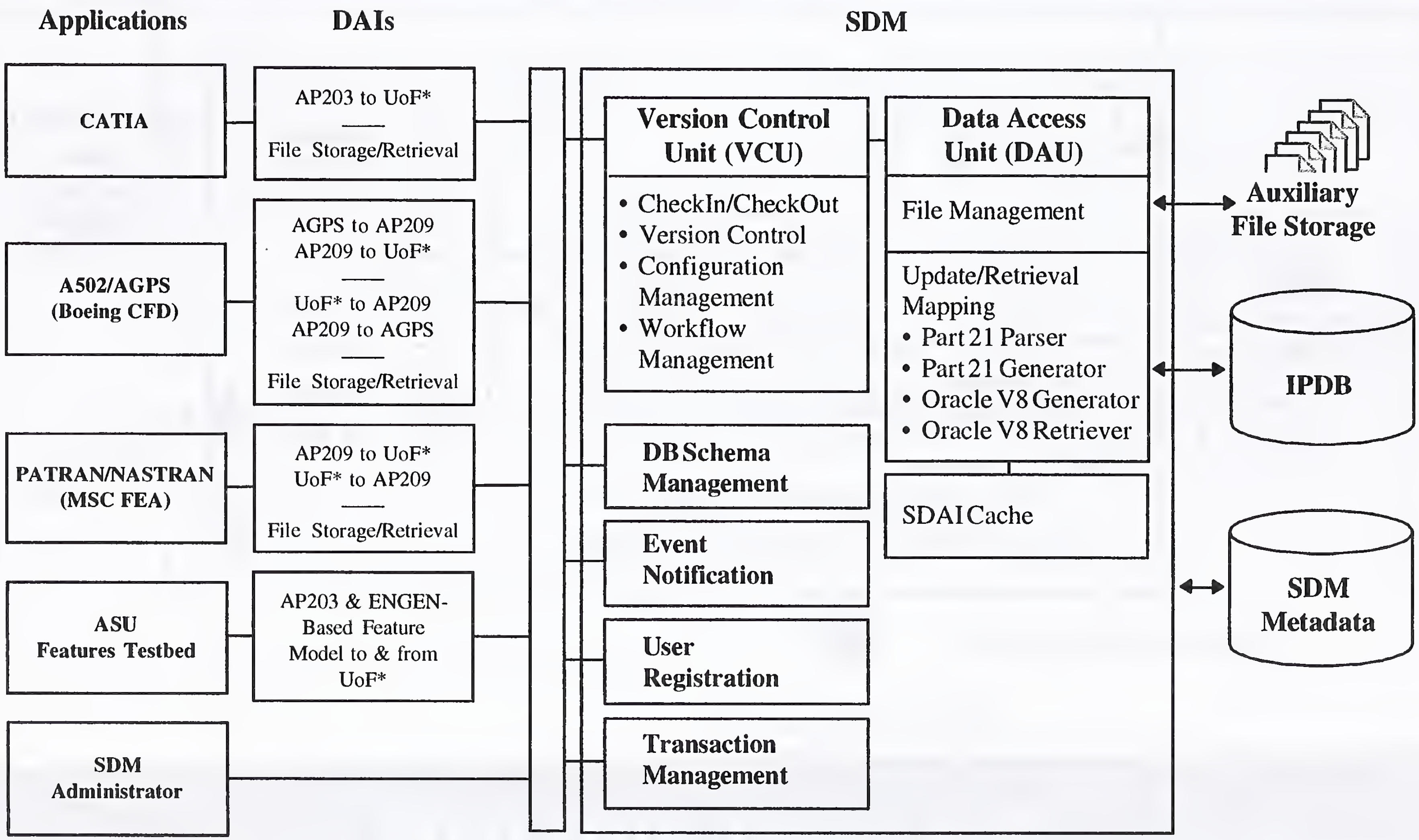
IPDB Technology Issues

- Identify a Sufficient & Complete Set of UoF*s for the Selected Application Domains
- Determine the Correct Level of Granularity for UoF*s
- Partition STEP APs Into "Internal" (Database) and "External" (File-Hosted) Sets of UoF*s
- Resolve Interobject References & Constraints Among UoF*s
- Resolve Interobject References & Constraints Among Lower-Level Objects (e.g., AIM Objects) Embedded Within UoF*s
- Use an Object-Relational Database (Oracle V8) to Address an Industrial Scale Data Management Problem
- Manage the Required Volumes of Data (VLDB Issues)

Part 4
SDM Development

The MacNeal-Schwendler Corporation

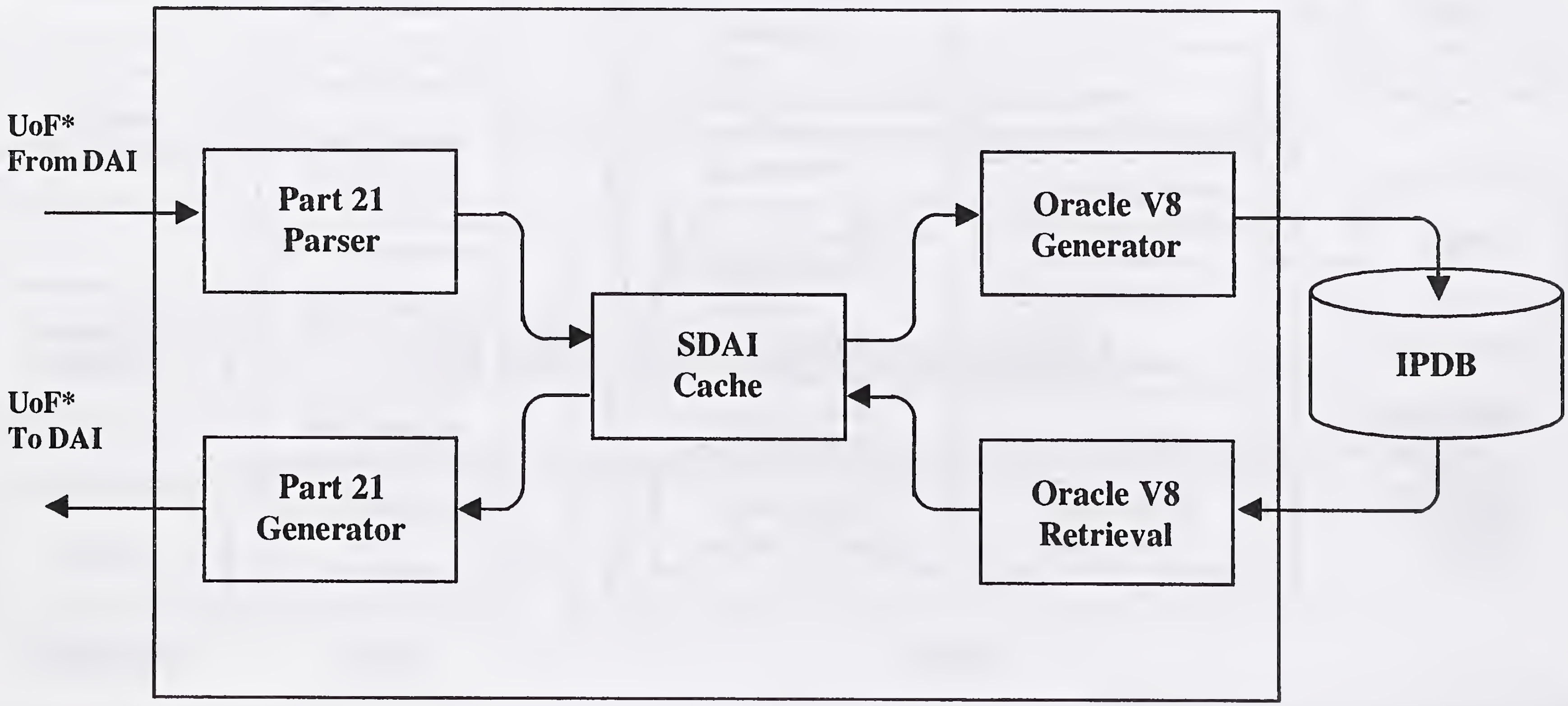
SDM Architecture



IPDE_32

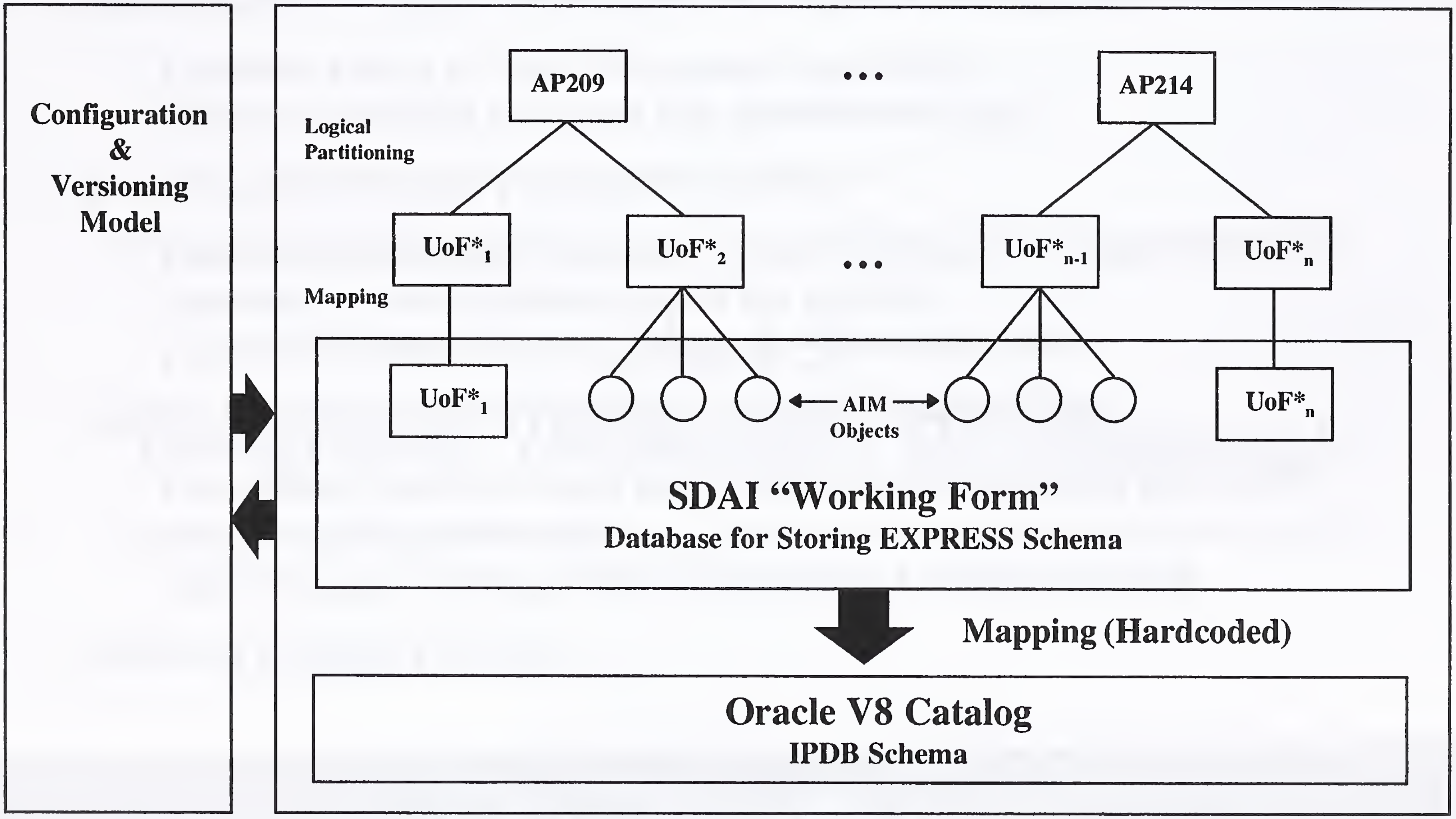
DARPA BOEING ASU The MacNeal-Schwendler Corporation

SDM Data Access Unit



The MacNeal-Schwendler Corporation

SDM Metadata



DARPA BOEING ASU The MacNeal-Schwendler Corporation

SDM Technology Issues

- Define DAI/SDM Interface
  - Manage UoF*s as the Primary Unit of Data Transfer for IPDB
  - Resolve Cross-Domain UtUs
  - Coordinate Version/Control Data With Checkin/Checkout Operations
- Define Version/Control Model for Evolving Design Data
  - Understand Industry (e.g., Boeing) Practices and Needs
  - Maintain Control Without Stifling the Process
  - Evaluate Applicability of Version/Control Models in AP203 and AP209
- Perform Internal SDAI Caching of UoF*s
  - Enforce Constraint Checking and Management UtUs
  - Translate UoF*s To and From Oracle V8 Objects
- Resolve Storage Management Issues for Both UoF* BLOBS and Native Files

The MacNeal-Schwendler Corporation

IPDE Program

Part 5
Integrated Vocabulary Development

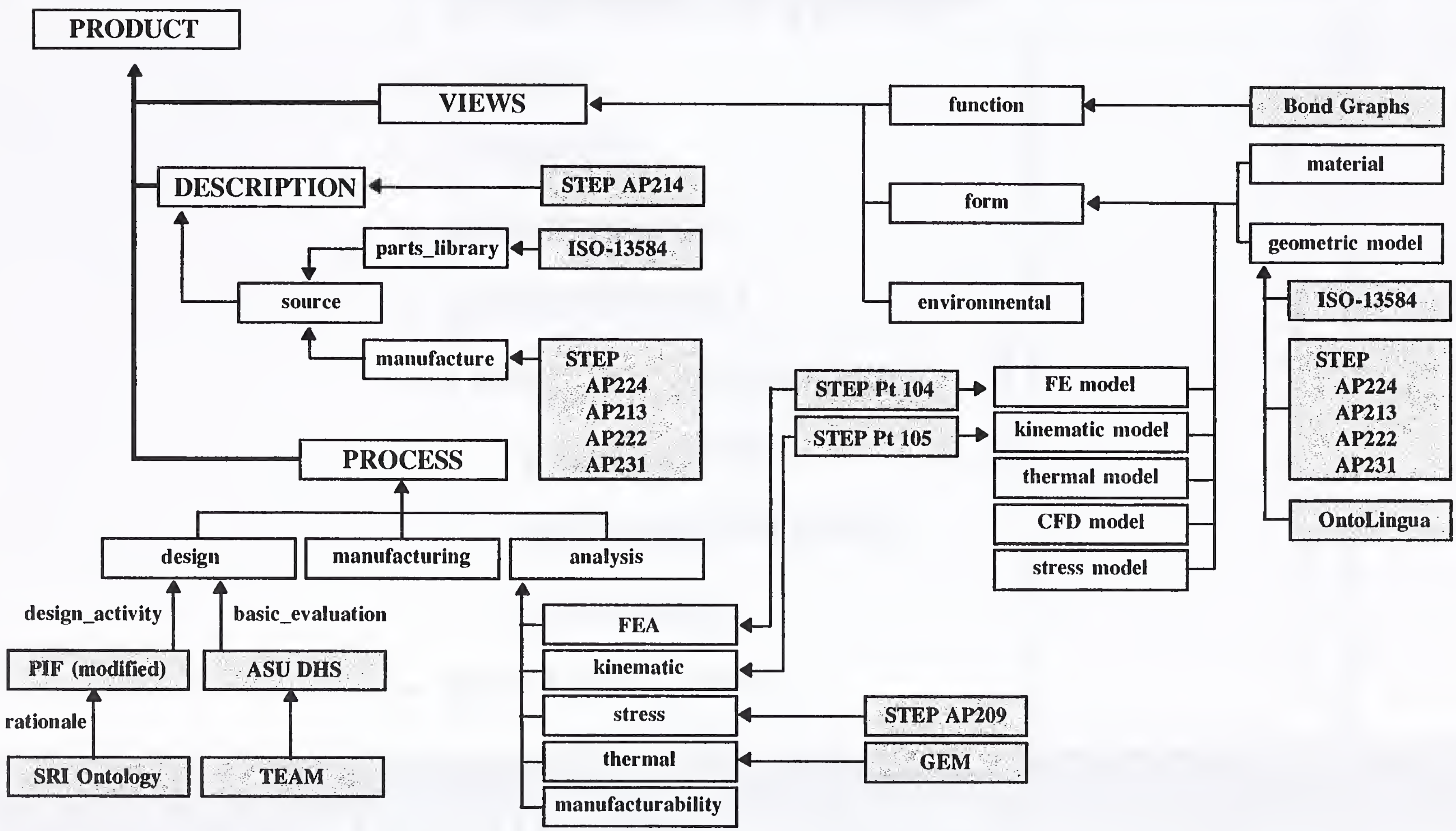
The MacNeal-Schwendler Corporation

CEM Vocabulary Integration

- Motivation
  - All Design is Redesign (Leifer, Rule 2)
  - Create Design Information Model for Archival and Query
  - Retrieve Design Fragments for Reuse
- Guiding Principles
  - Design Retrieval for CEM Systems Requires More Than Just Part Number or CAD Models—Broader Set Includes:
    - » Performance Specs
    - » Function & Behavior
    - » Decision Rationale
    - » Organizational Data
    - » Cost Data
    - » Design Procedure
    - » Artifact Characteristics
    - » Etc.
  - Make Full Use of Existing Standards, & Pseudo-Standards (e.g., STEP, P-LIB, KIF, PIF, IBIS, OntoLingua Frame Model)
  - Create an Infrastructure & Methodology for a Design Information Scheme That Spans Multiple Domains
    - » A Given Domain Might be Addressed by More Than One Data Standard (e.g., STEP, KIF)

IPDE_37 DARPA BOEING ASU The MacNeal-Schwendler Corporation

CEM Vocabulary Integration



IPDE_38

Vocabulary Integration Issues

- Diversity Issues
  - Semantic
  - Representational
  - Contextual
- Levels of Abstraction
- Heterogeneity
- Redundancy
- Conflicts
- “Holes”
- Ambiguities in Mapping
- Scope

The MacNeal-Schwendler Corporation

Design Data Treatment Issues at Boeing Helicopters

Dr. Gary Coen, Boeing (gary.a.coen@boeing.com)

The summary for this presentation can be found on page 24
5 slides follow

Gary Coen

Enabling Technologies
Information Management Systems
Helicopters Division
Boeing Defense and Space Group

October 28, 1996

Conventional Design Storage and Retrieval

Conventional Use of EPIC (CATIA)

Principle Datum Attribute: Assembly Drawing Number
Metadata Inherent in Indexing Scheme
Part Number Index (Model+Subsystem)

- 901H0001 Hydraulics
  - 901H0010
  - 901H0020
    - 901H0024 LG Main LH Tire
    - 901H0024 LG Main RH Tire
- 901D0001 Drive Systems
- 901P0001 Power Plant
- 901R0001 Rotors
- 901S0001 Structures
- 901V0001 Avionics
- 901X0001 Configuration

No Explicit Provision for Design Intent, Rationale
Designs Tabulated with Aircraft Numbers
Product-Centric Data Model

Feature-Based Design Storage and Retrieval

Explorations with Group Technology

Principle Datum Attribute: Assembly Drawing Number
Metadata Unneccesary to Indexing Scheme
Design Feature Index
Attribute Value Collection
Design Restriction

- Accuracy of featural information dependent on user input
- Parts must be constructed as unique solids within the model

No Explicit Provision for Design Intent, Rationale
Feature-Centric Data Model

Table 1: Ply Table

Attribute	Description
PARTNUM	Part Number
EGLASS	E-Glass Covering
COCURE	Is Part Cocured
PLYPAC	Ply Sequence Number
NUMPLYS	Number of Plys
PLYMATL	Material Number of Ply
PLYANG	Ply Orientation Angle

Table 2: Part Table

Attribute	Description
PARTNUM	Part Number
STRSHAPE	Structural Cross-Section Shape
REVDEC	Revision Level
MFROM	Make from Part Number
THKNESS	Thickness of All Plys
LEG90G	Leg Angle < 90
LEG1LEN	Leg One Length
LEG1RAG	Leg One Radius
LEG1THK	Leg One Thickness
LEG2LEN	Leg Two Length
LEG2RAD	Leg Two Radius
LENGTH	Length of Part
WEBHGT	Web Height
WEBTHK	Web Thickness
HOLES	Does Part Have Holes

Feature-Based Design Storage and Retrieval

Conventional Use of ProEngineer, IGES, etc.

Principle Datum Attribute: Assembly Drawing Number
Metadata Unneccesary to Indexing Scheme
Design Feature Index
Design Restriction
- Feature library must be constructed over time

Design Enhancements
- No attribute value collection
- Features may be added or deleted from designs
- Re-use of feature library returns cost of library construction

No explicit provision for design intent, design rationale
Feature-Centric Data Model

Challenges for Design Repository Schemata

Feature-Based Design Repository

Metadata Unneccesary to Indexing Scheme
Potential for Design Clustering
Explicit Treatment of Design Rationale

- Applications collect design rationale
- Data models relate rationale to constraint satisfaction
- Rationale may contain crucial metadata

Explicit Treatment of Design Intent

- Applications compute design intent
- With multiple versions of a design, intent can be derived from construction history
- Data models represent intent as a design attribute

Lessons Learned Database

- Rationale-based indexation and retrieval (design + feature content + constraints satisfied)
- Support memoization for design optimization processes

Use of Standards for Data Storage and Exchange

Gene Allen, The MacNeal-Schwendler Corporation (gene.allen@macsch.com)

The summary for this presentation can be found on page 28
8 slides follow

CUSTOMER VISION

PROVIDE **ANY USER** WITH THE CAPABILITY TO DO **ANY ANALYSIS** FROM THE **SAME PRODUCT MODEL** IN **REAL TIME** WITH THE ABILITY TO **MANUFACTURE THE PRODUCT DIRECTLY** FROM THE MODEL

MSC COLLABORATIVE DEVELOPMENT

DEVELOPMENT AREAS TO REALIZE CUSTOMER VISION

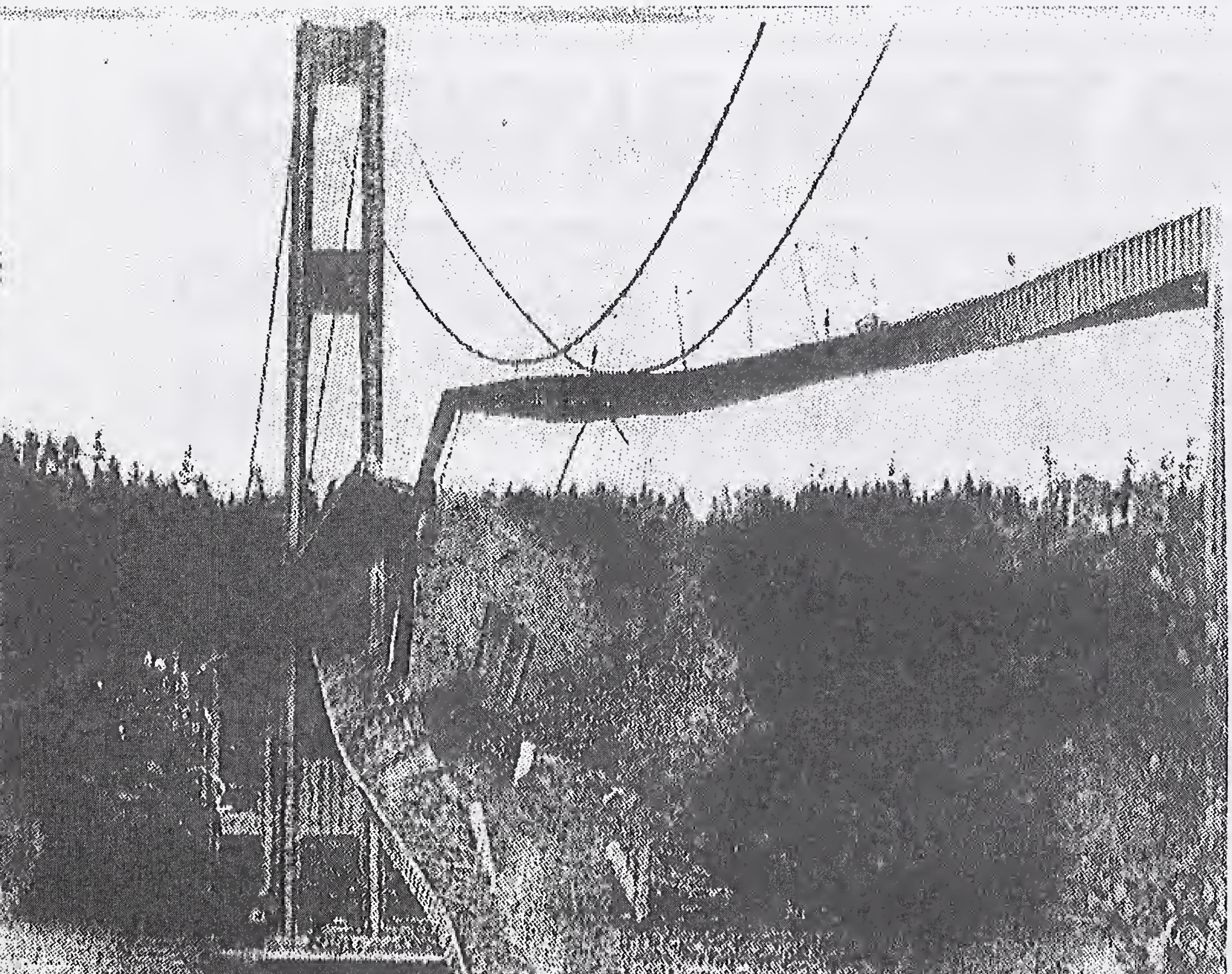
ACCURATE MODEL OF REALITY

INTEROPERABILITY

EASE OF USE

MSC

MODEL ACCURACY



TACOMA NARROWS BRIDGE COLLAPSE

SOFTWARE FAILURES ARE SELDOM SEEN, BUT OCCUR WITH MUCH GREATER FREQUENCY, AND WITH EQUALLY DISASTROUS RESULTS

MSC

INTEROPERABILITY

PROBLEM:

DIFFERENCES BETWEEN

CAD

CAM

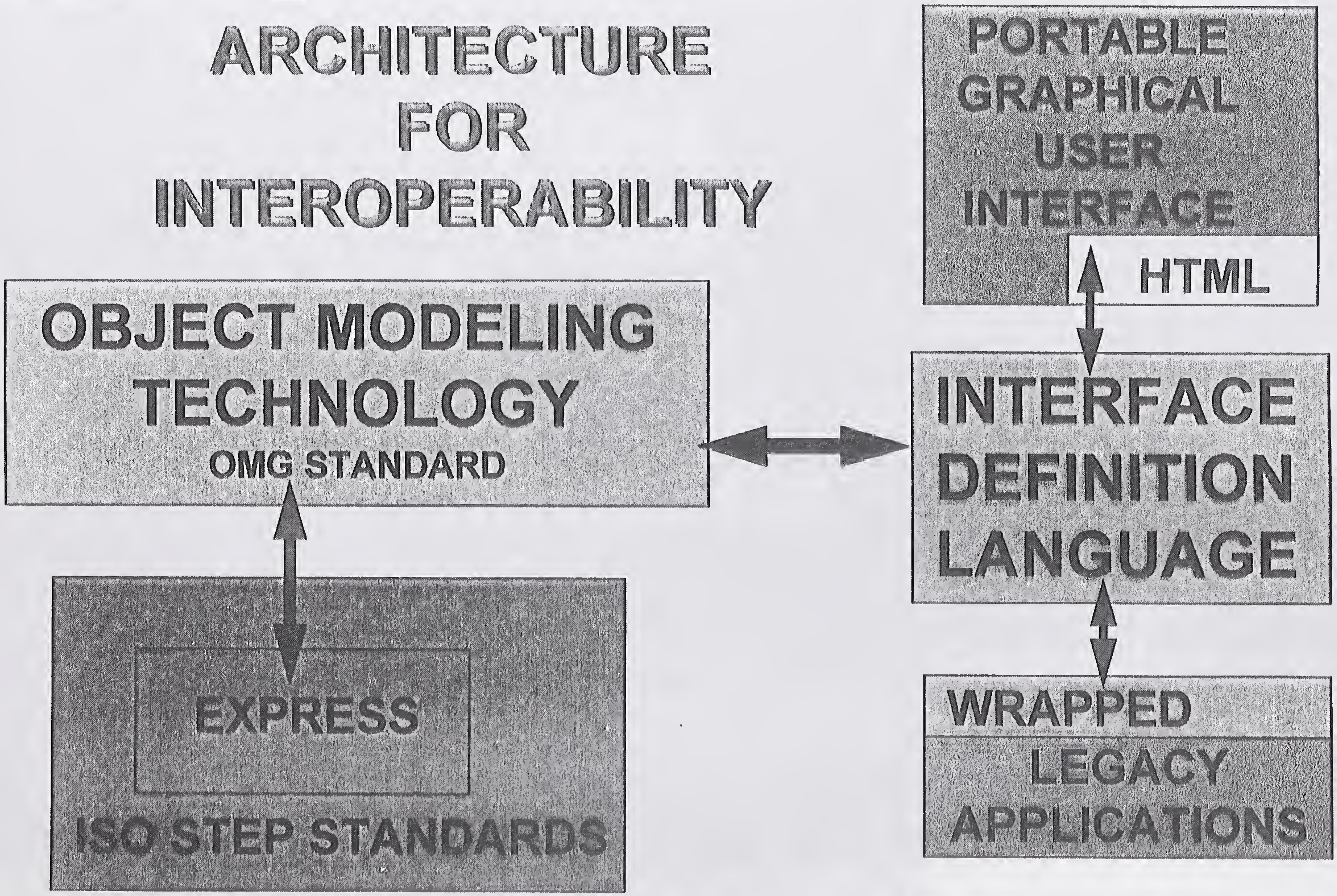
CAE

DIFFERENT SHAPE REPRESENTATION

DIFFERENT MODEL CONSTRUCTION

COLLABORATIVE DEVELOPMENT

ARCHITECTURE FOR INTEROPERABILITY



Industry Drivers

- Predictive Engineering
  - simulation increases the demand for accurate, electronic materials information
- New Materials
  - availability of new materials makes materials a design variable, not a constraint
- Standards for Data Integration
  - demand for enterprise discipline drives companies to standards and systems for data exchange

Key Challenges to Managing Materials Data

- Goal: Best of Class Manufacturing
  - Increased performance, Faster time-to-market, Lower costs
- Approach:
  - Analysis-driven
    - Need data to support analysis (e.g. plastics from -40 to 200 °F)
  - Consistency
    - How do we share consistent data across multiple engineering disciplines for a common program?
  - Corporate standards
    - How do we foster corporately approved materials data usage?
  - Accurate, validated data
    - Where do we find statistically accurate, traceably validated, analysis-ready materials property data?

MSC VISION

MSC/MVISION Integrates Materials Information for Predictive Engineering

Materials Information Producers

Corporate Knowledge
(Experience)

Standards
(Handbooks)

Materials Producers
(Producer Data)

Materials Test
(In-House Characterization)

M

Control
Consistency
Quality

Materials Information Users

Design

Analysis

Manufacturing

Integrating a Distributed, Agile, Virtual Enterprise in the TEAM Program

Kim Cobb, Lockheed Martin (cli@ornl.gov)

The summary for this presentation can be found on page 29
24 slides follow

Integrating A Distributed, Agile, Virtual Enterprise In The TEAM Program

Kim Cobb, cli@ornl.gov

Lockheed Martin Energy Systems

Y-12 Plant, Oak Ridge, TN

TEAM Enterprise Integration Thrust Area Leader

Technologies Enabling Agile Manufacturing

TEAM

TEAM Mission

- The mission of the Technologies Enabling Agile Manufacturing program is to provide DOE and American industry with the critical, enabling technologies needed to implement Agile Manufacturing concepts that will enhance the global economic competitiveness of U.S. industries

What Is TEAM?

- TEAM is a partnership of government, industry, and academia
- TEAM focuses on implementation
  - An enabling technologies toolkit
  - Proof through industry led Product Vehicles

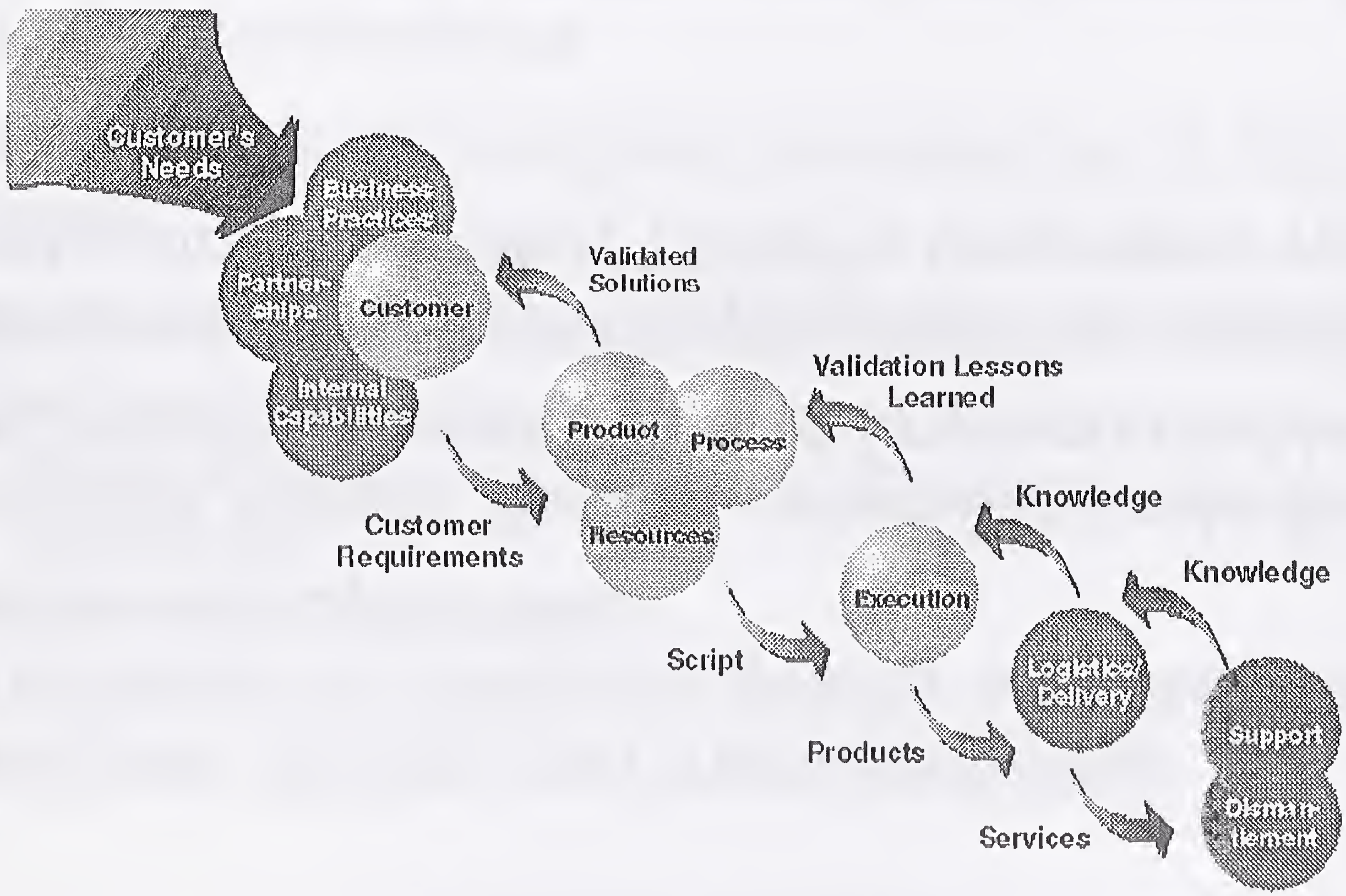
Scope and Alignment of TEAM

- Approximately 40 partners who provide in-kind contributions
- Enabling technologies provided by 5 Thrust Areas
- Applications of those technologies through Product Vehicles
  - Material Removal, Forming, Electromechanical Assembly
  - Defense Programs

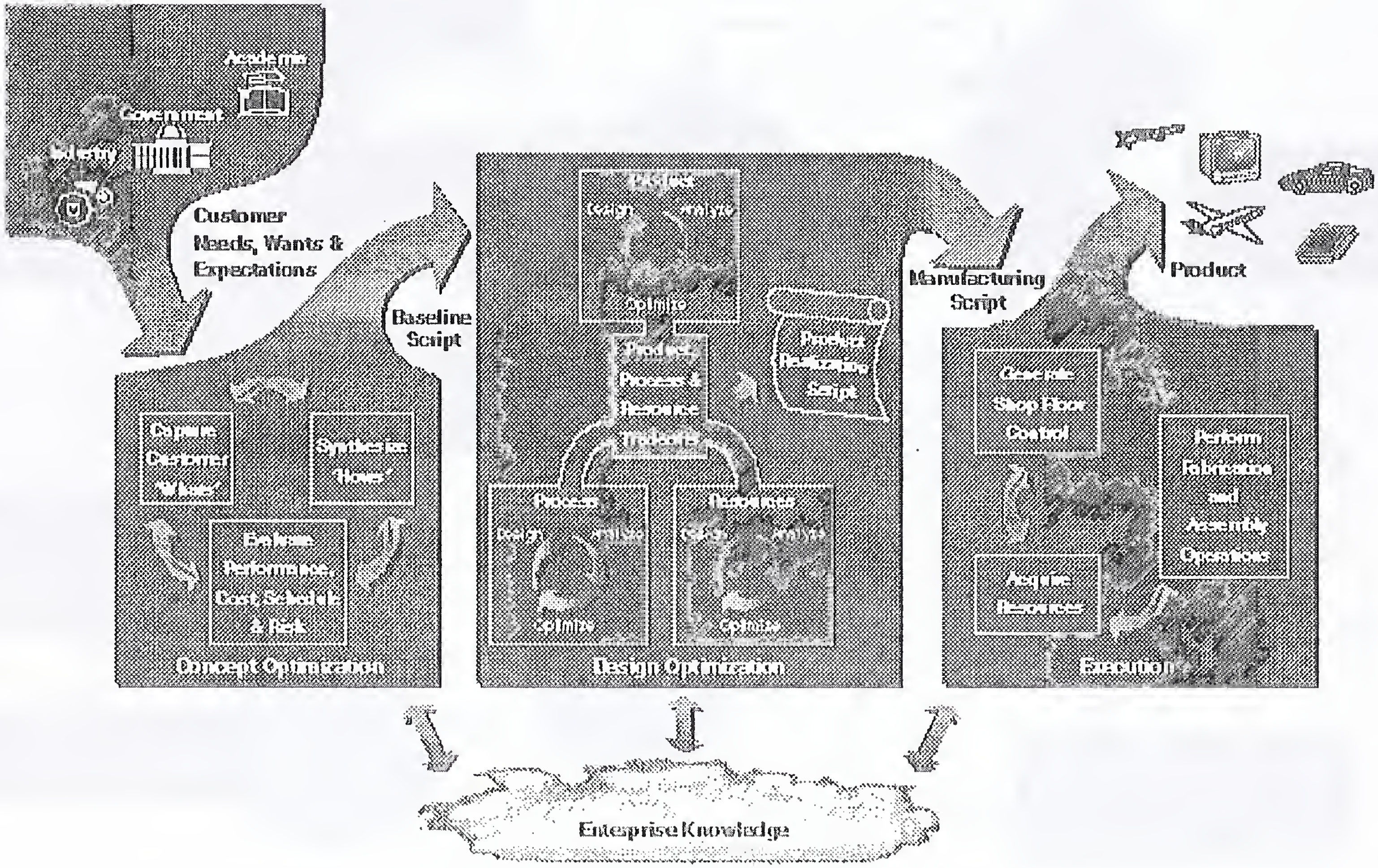
TEAM Goals

- Integrate design and manufacturing processes to enhance quality and reduce costs and cycle time
- Provide robust, flexible, modular tools that are readily accessible and implementable
- Maximize near-term deployment of enabling technologies within TEAM's long-term vision

The TEAM Enterprise Model



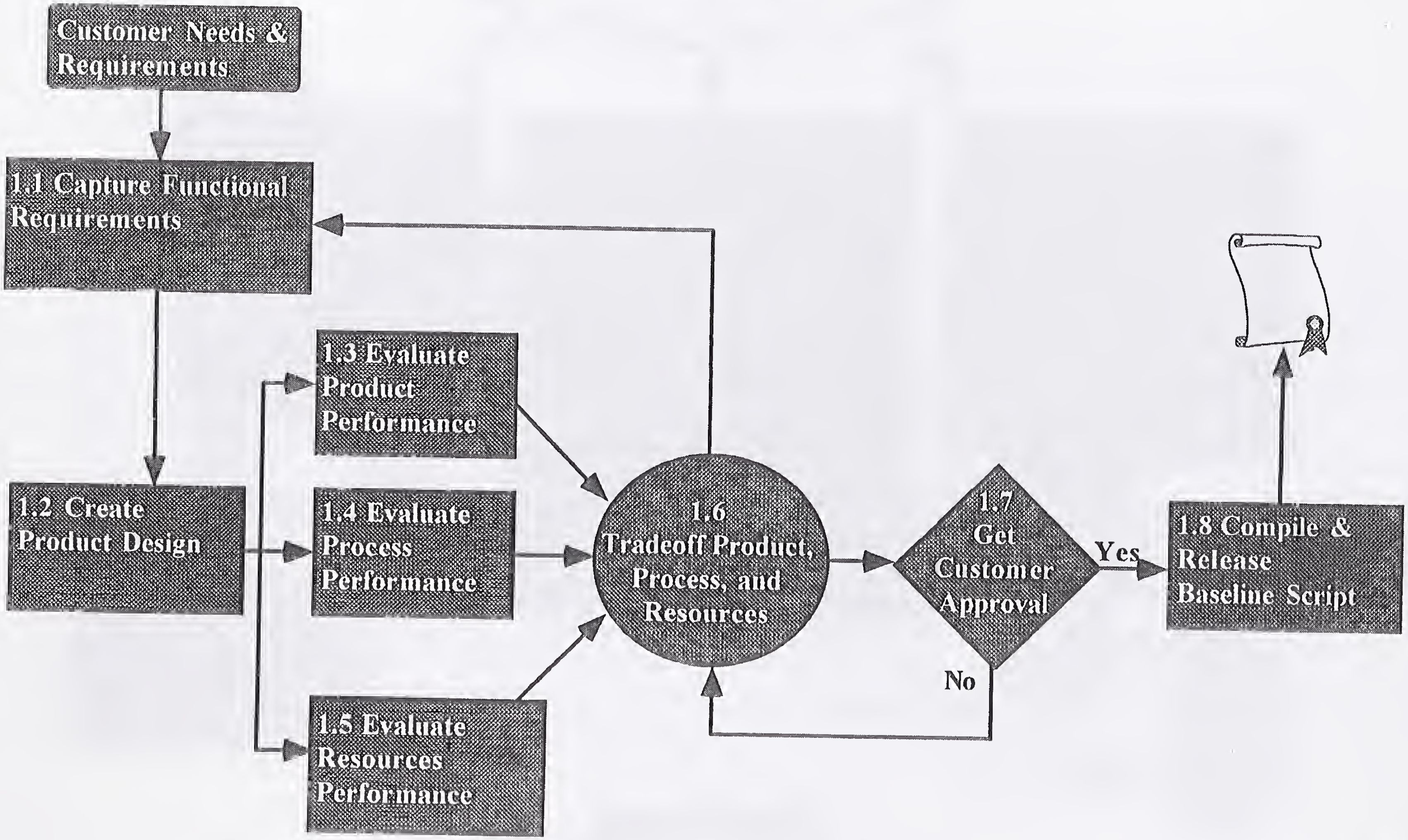
The TEAM Product Realization Model



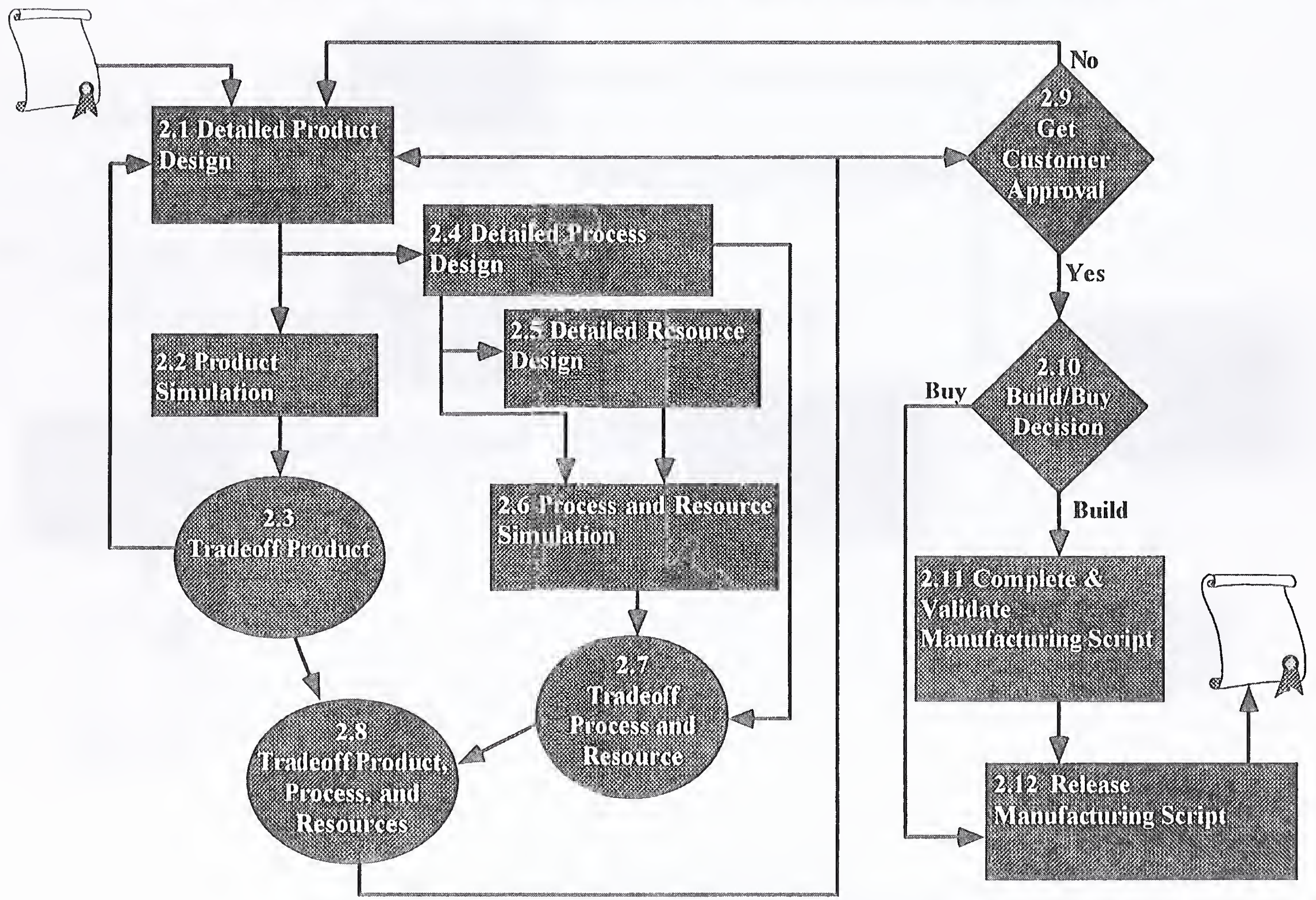
Technologies Enabling Agile Manufacturing

TEAM

Concept Optimization Model



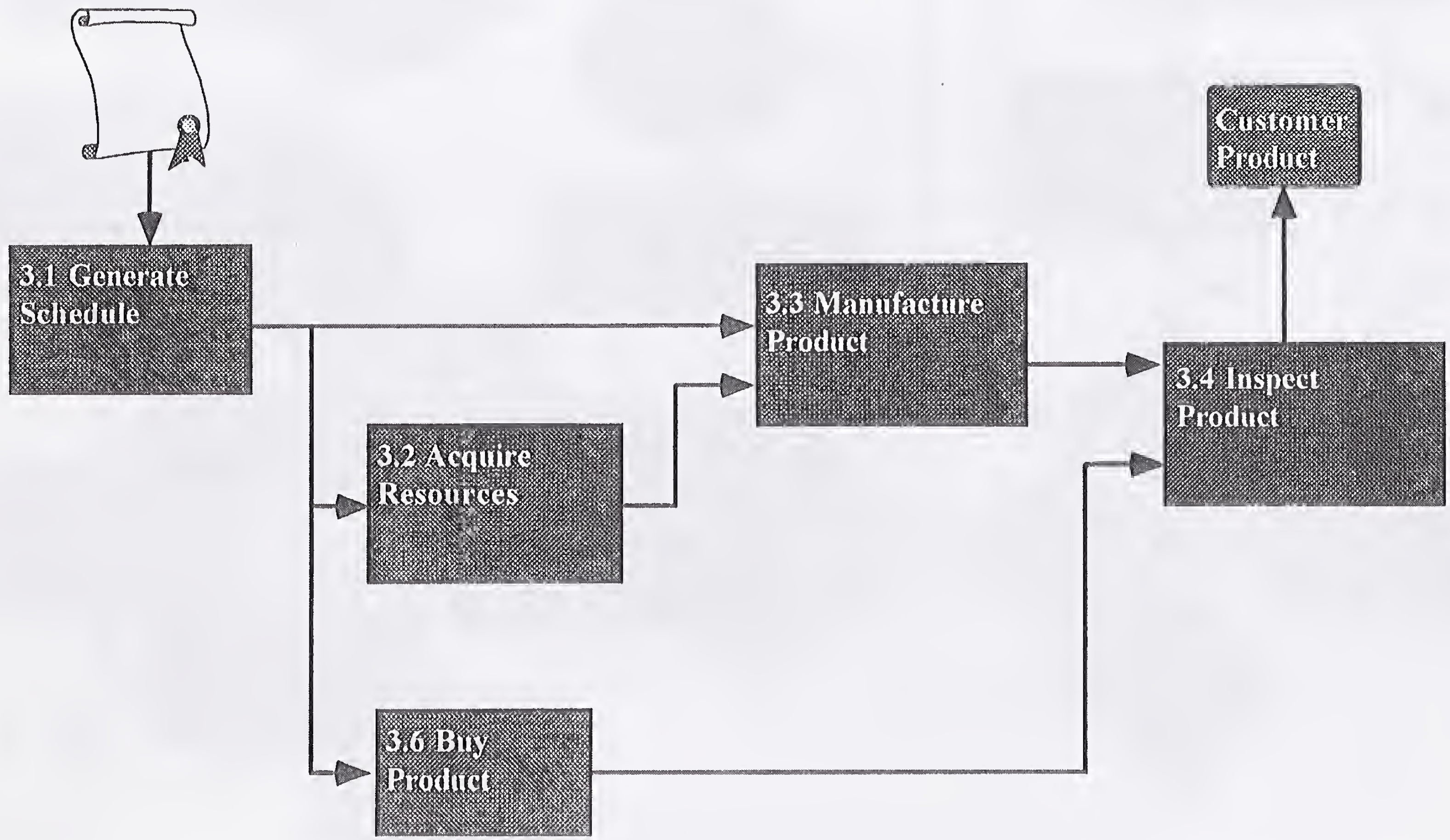
Design Optimization Model



Technologies Enabling Agile Manufacturing

TEAM

Execution Model



TEAM Demonstrations

- March 1996
  - "Interconnected" Material Removal
- June 1997
  - Integrated Material Removal
  - "Interconnected" Forming
  - "Interconnected" EM Assembly

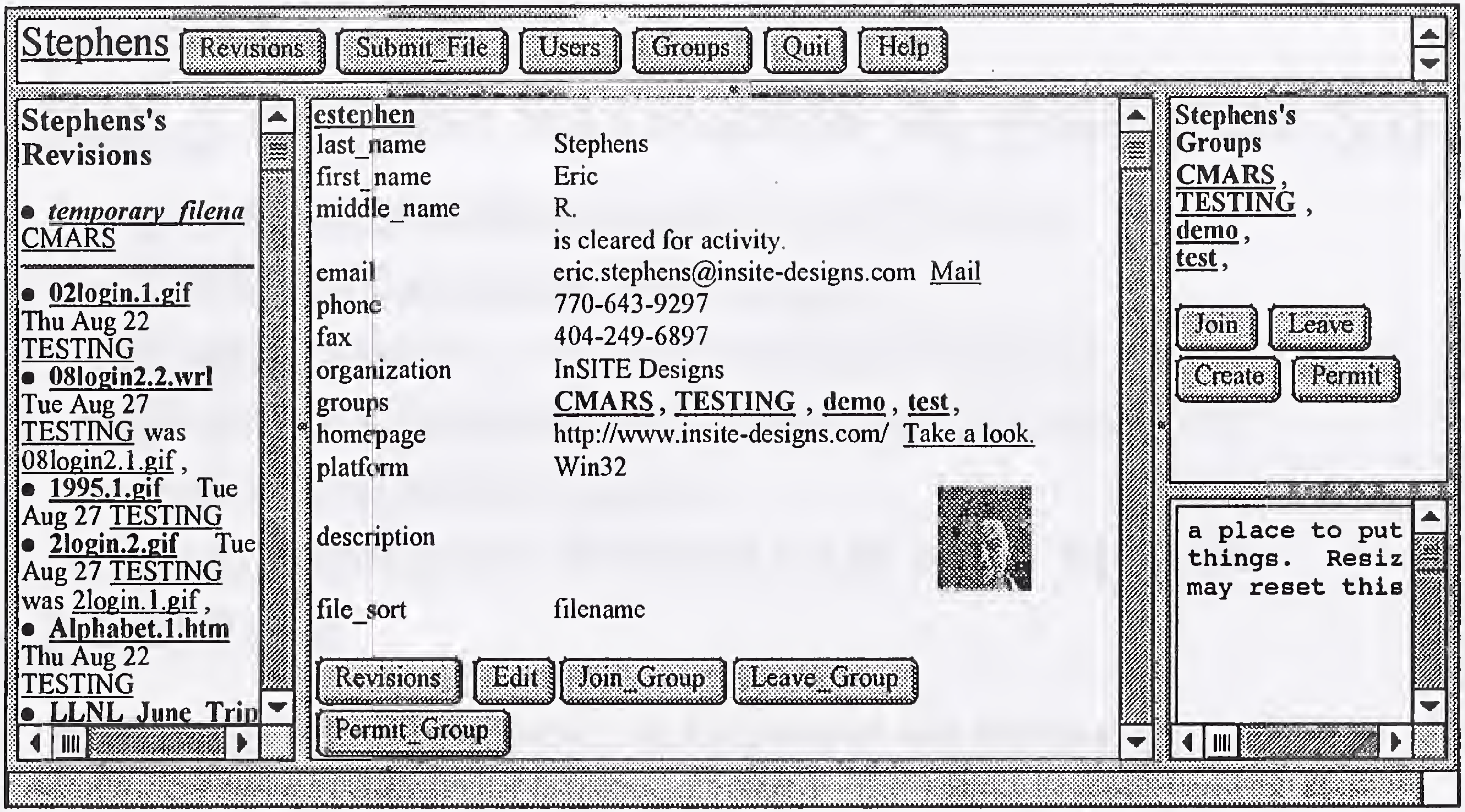
Enterprise Integration Thrust Area Activities

- March 1996
  - Intersite File Manager
- June 1997
  - Integration framework activities
  - Web Integration Manager
  - Automated concept optimization
  - Object-oriented knowledge repository

Intersite File Manager

- Web-accessible; firewall enabled
- Features
  - Multi-platform support via Web clients
  - User authentication
  - Access control
  - Notification
  - Version control
  - Script management
- IFM 2.0 takes advantage of Netscape 2.0 features

Intersite File Manager



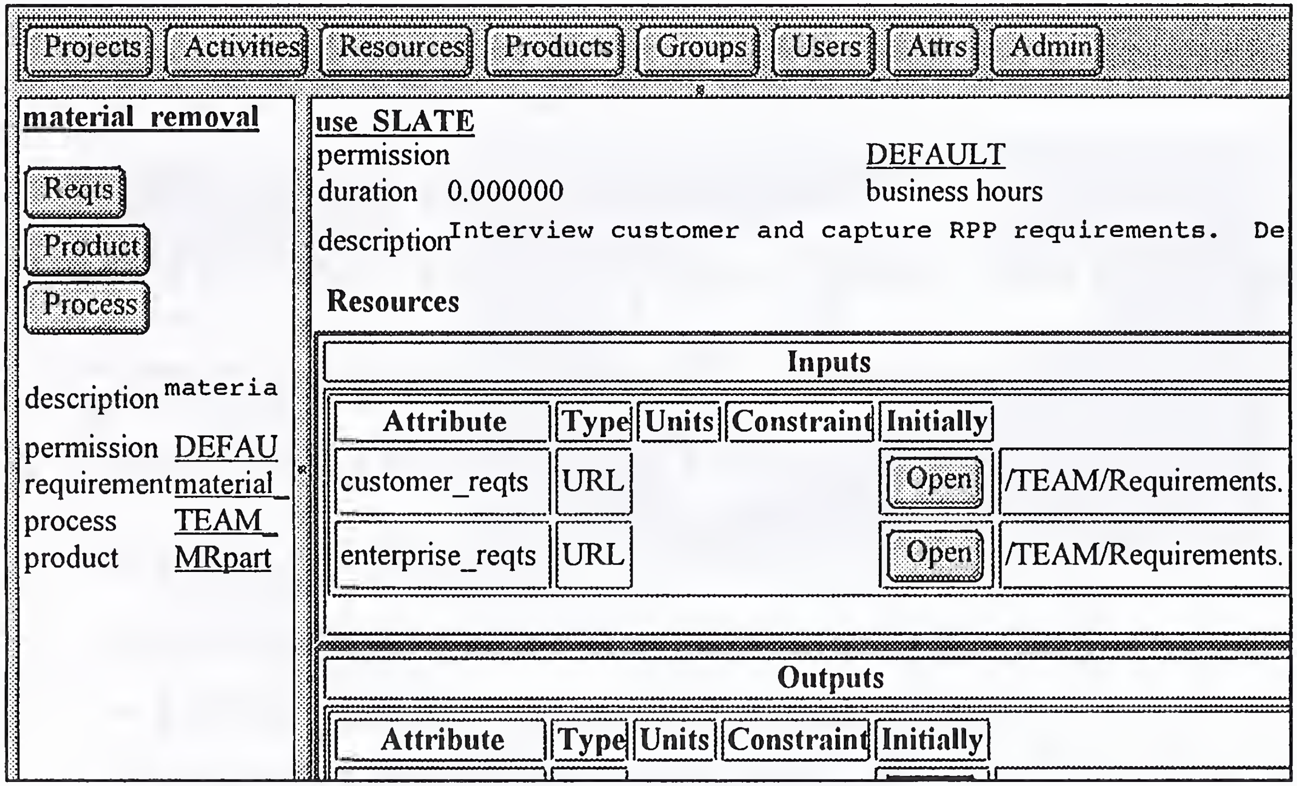
Integration Framework Activities

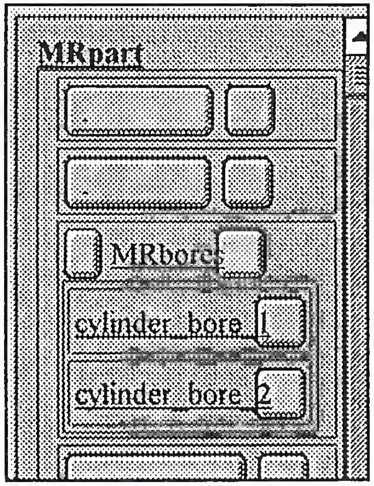
- Leveraging activities of TEAM participants
  - Sandia/CA Product Realization Environment (PRE) project
  - NIST NAMT framework project

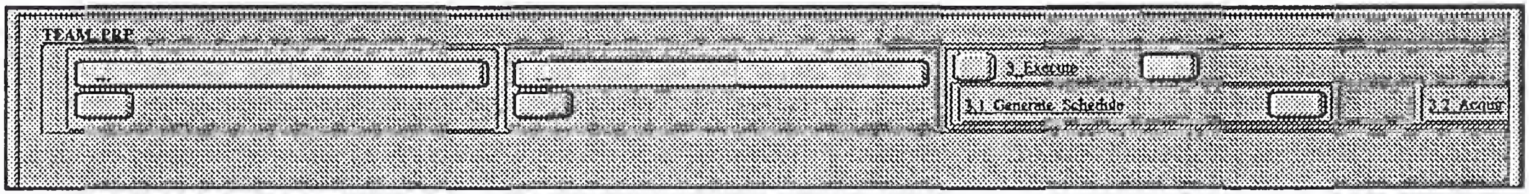
Web Integration Manager

- Web-accessible; firewall enabled
- Horizontal integration of the TEAM product realization process for the integrated Material Removal demo
  - Tool invocation
  - User notification
  - File delivery
- Will be compliant with the PRE framework

Web Integration Manager



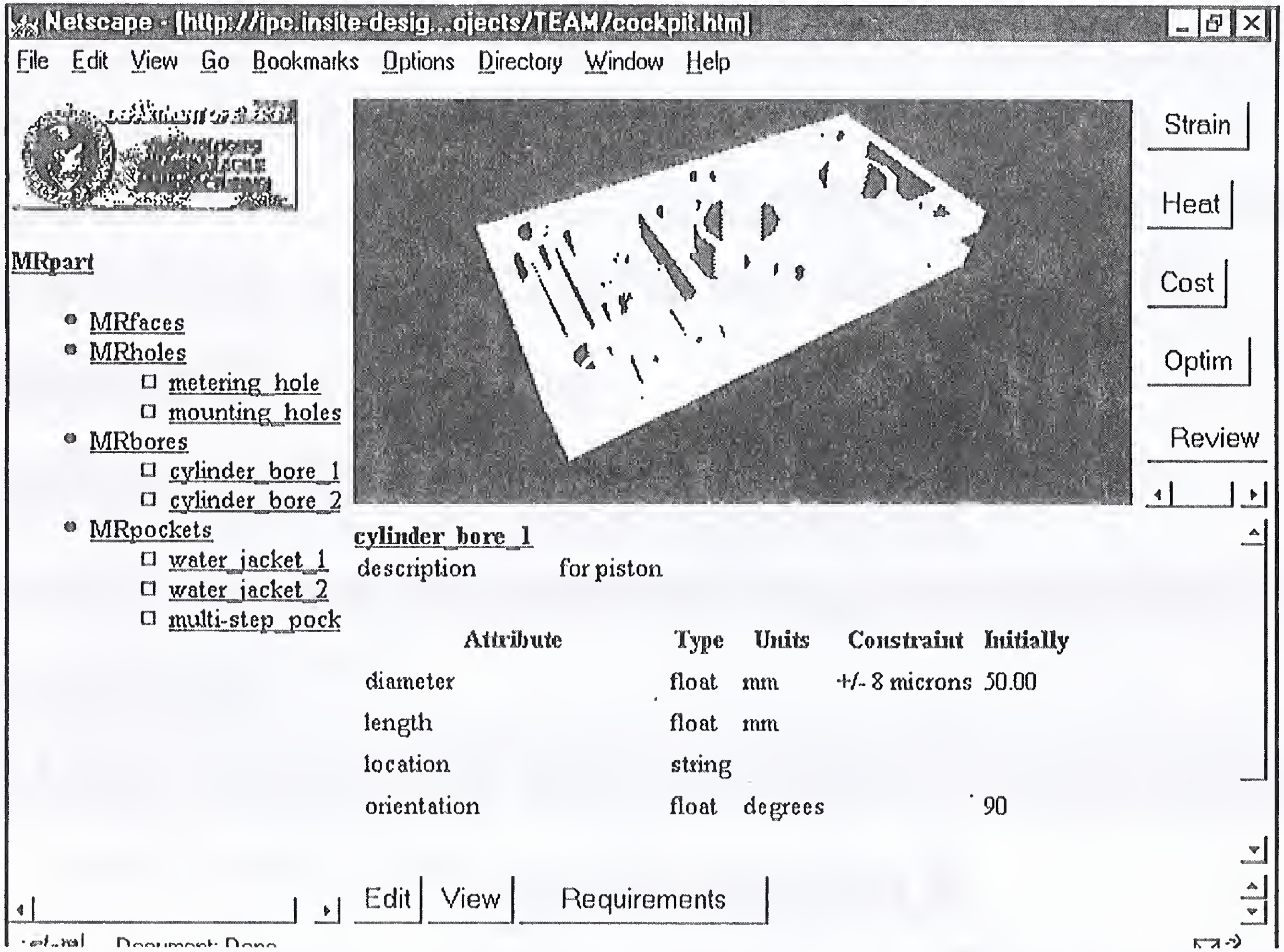




Automation of Concept Optimization

- Tight, seamless, vertical integration of tools
  - Requirements flowdown
  - Parametric design variation
  - Analysis (stress, thermal, cost)
  - Optimization
- Conceptual designer performs all Conceptual Design tasks
- Requires only Netscape; applications executed but not seen

Concept Optimization Cockpit



Technologies Enabling Agile Manufacturing

TEAM

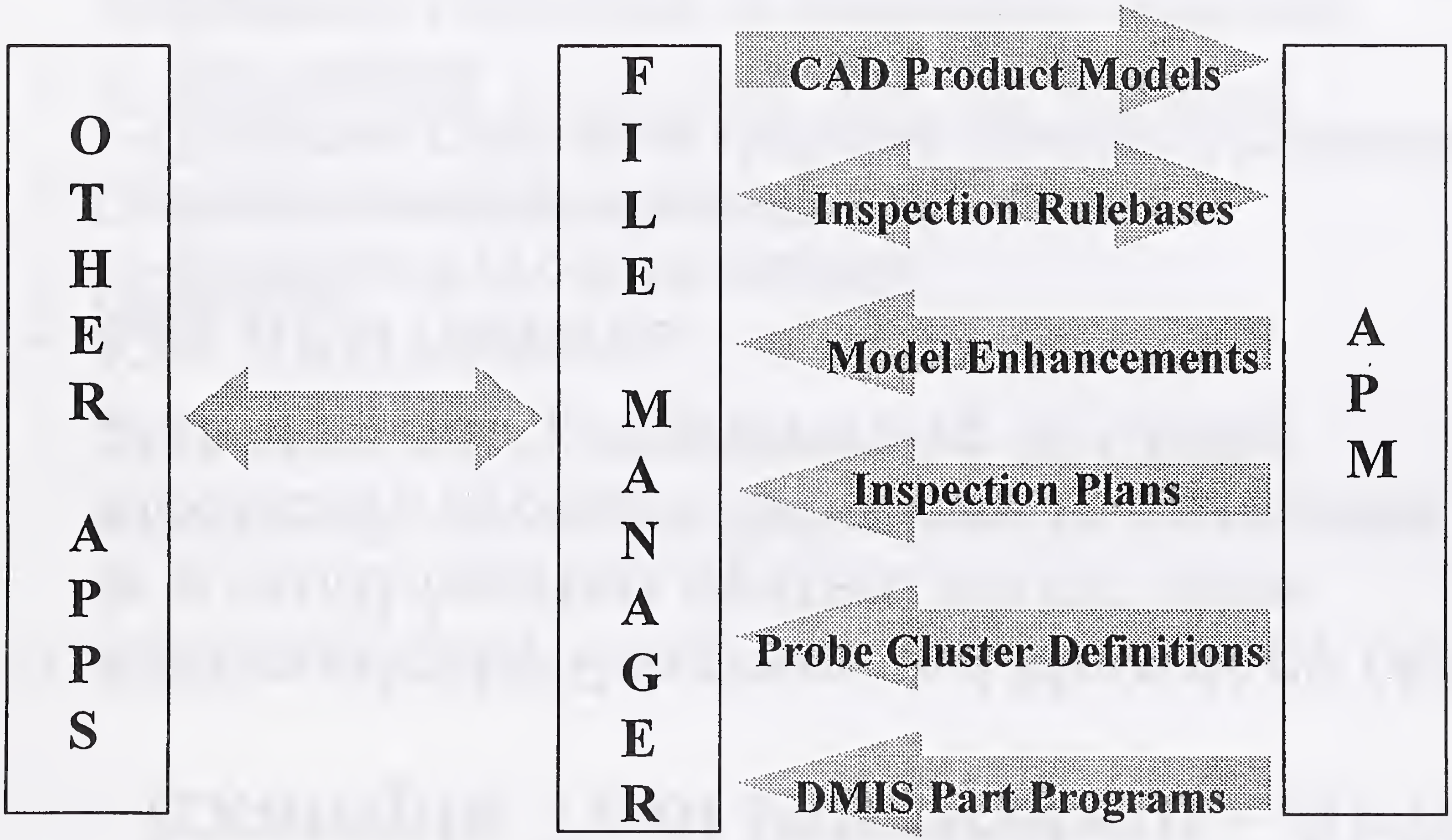
Object-Oriented Engineering Repository

- STEP schemas for geometry, tolerances, and features
- Customized schemas will be defined
- Distributed translation services
- Queries

Example Tool Integration - APM

- Automatically Programmed Metrology (APM) is a multi-faceted system which uses electronic product definition to automate the planning and programming of CMMs
- Key APM features
  - Enhanced product models
  - Feature recognition algorithms
  - External rulebases defining standard inspection techniques
  - External rulebases of inspection methods
  - NC program generation with 3-D collision-free path planning

Example Tool Integration - APM



Technologies Enabling Agile Manufacturing

TEAM

Summary

- Benefits of Enterprise Integration Thrust Area activities...
  - Will provide integrated process, application, and database capabilities
  - Will enable collaboration based on distributed, shared information
  - Will enable techniques for rapidly responding to change
- TEAM partners are beginning to realize program goals with the delivery of demonstration results

TEAM On The Internet

- TEAM home page
  - http://cewww.eng.ornl.gov/team/home.html

Engineering Repository Projects at NIST

Dr. Simon Szykman, NIST (szykman@cme.nist.gov)

The summary for this presentation can be found on page 30
18 slides follow

Engineering Repository Projects at NIST

Ram Sriram, Simon Szykman, William Regli
Manufacturing Systems Integration Division
National Institute of Standards and Technology
Gaithersburg, MD

William Murdock
Department of Computer Science
Georgia Institute of Technology
Atlanta, GA

Outline

- Introduction
- Design Case Study Framework
- Design, Process Planning and Assembly Repository
- Areas for Future Research

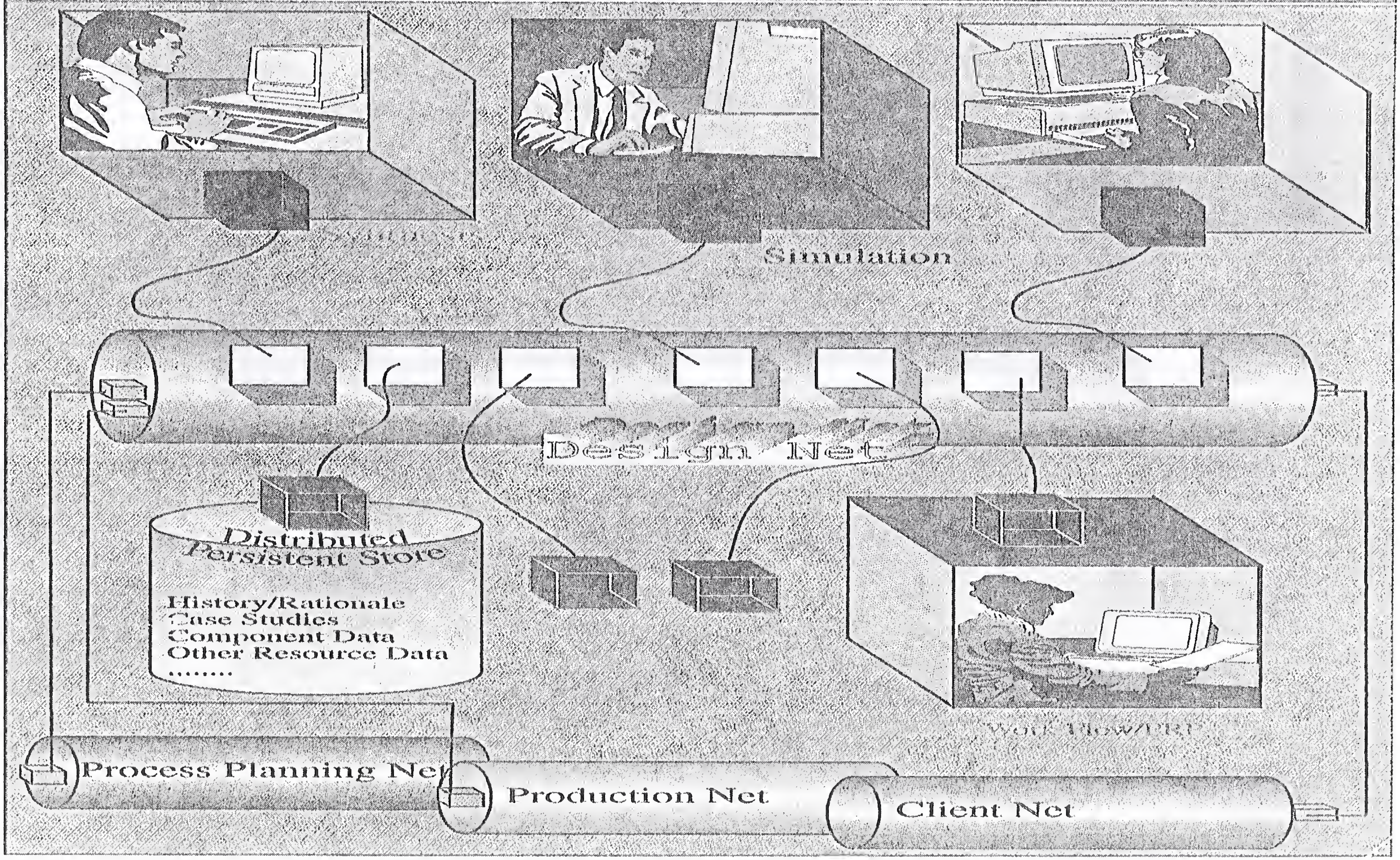
NIST

Introduction

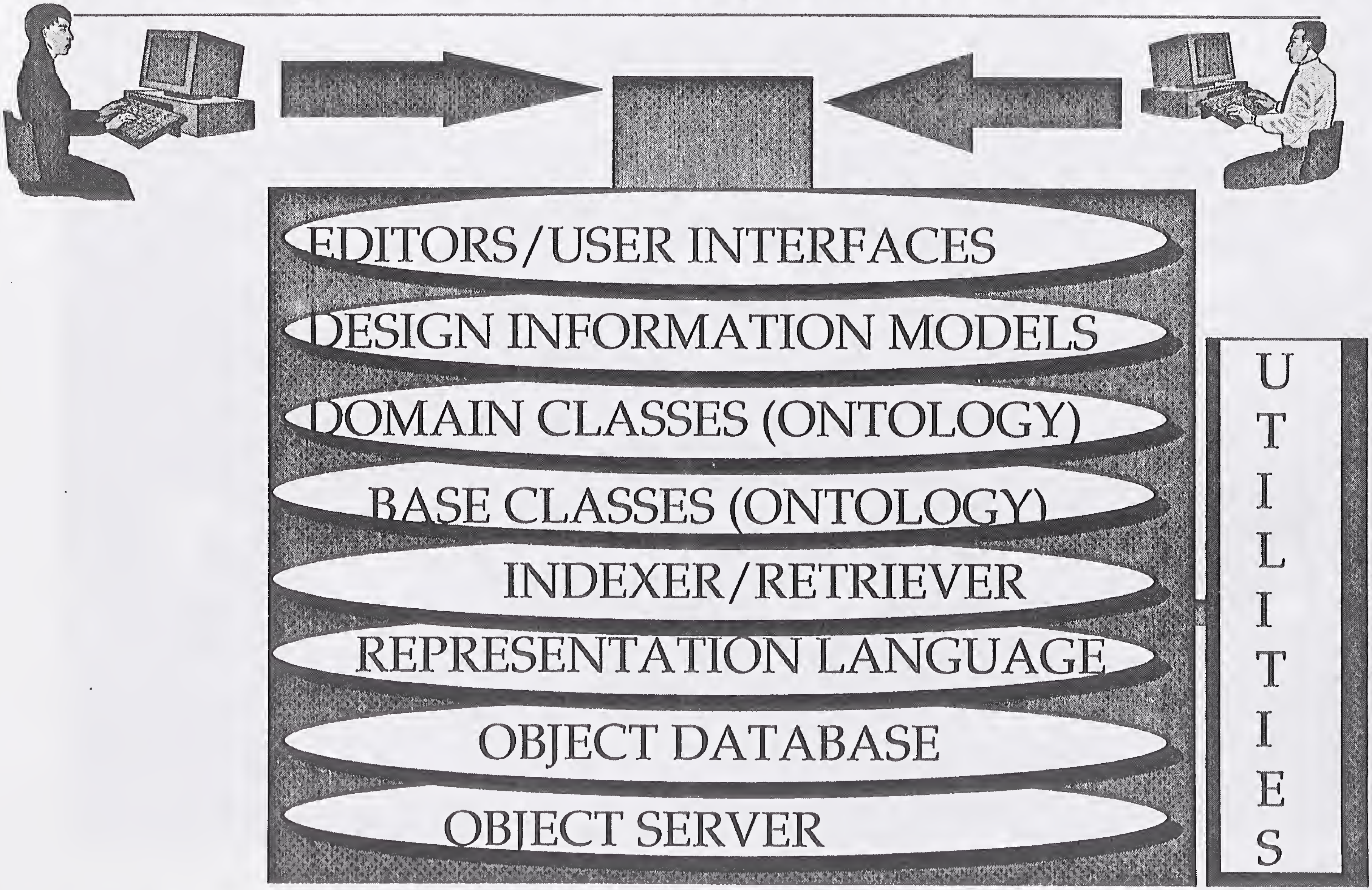
- Designers need different kinds of information at various stages of the design process
- Timely access to this information can help make US industry more competitive by:
  - improving quality
  - reducing costs
  - reducing product development time
- Objective: to facilitate the development of comprehensive design knowledge/data bases that enable rapid access to information at various levels of detail

NIST

COLLABORATIVE DESIGN



Design Case Study Framework



Design Case Study Framework

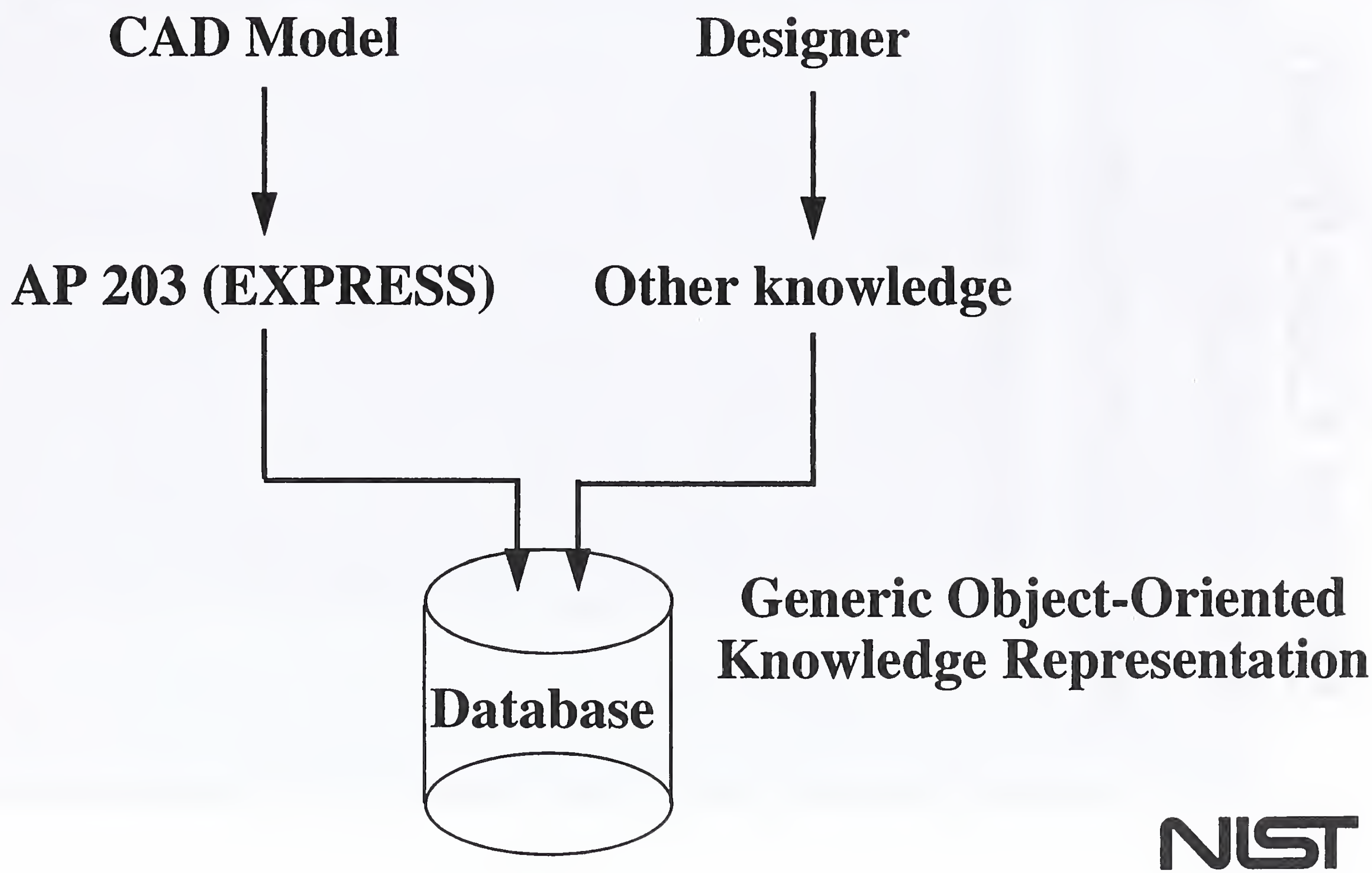
- Generic knowledge representation
- Design knowledge organized in layers from basic principles to domain specific information
- Supports multiple problem-solving approaches:
  - top-down functional decomposition
  - bottom-up design
  - stepwise refinement
  - constraint propagation

Design Case Study Framework

- Object model represents:
  - Artifacts
  - Form, function, behavior
  - Methods
  - Relationships
  - Constraints

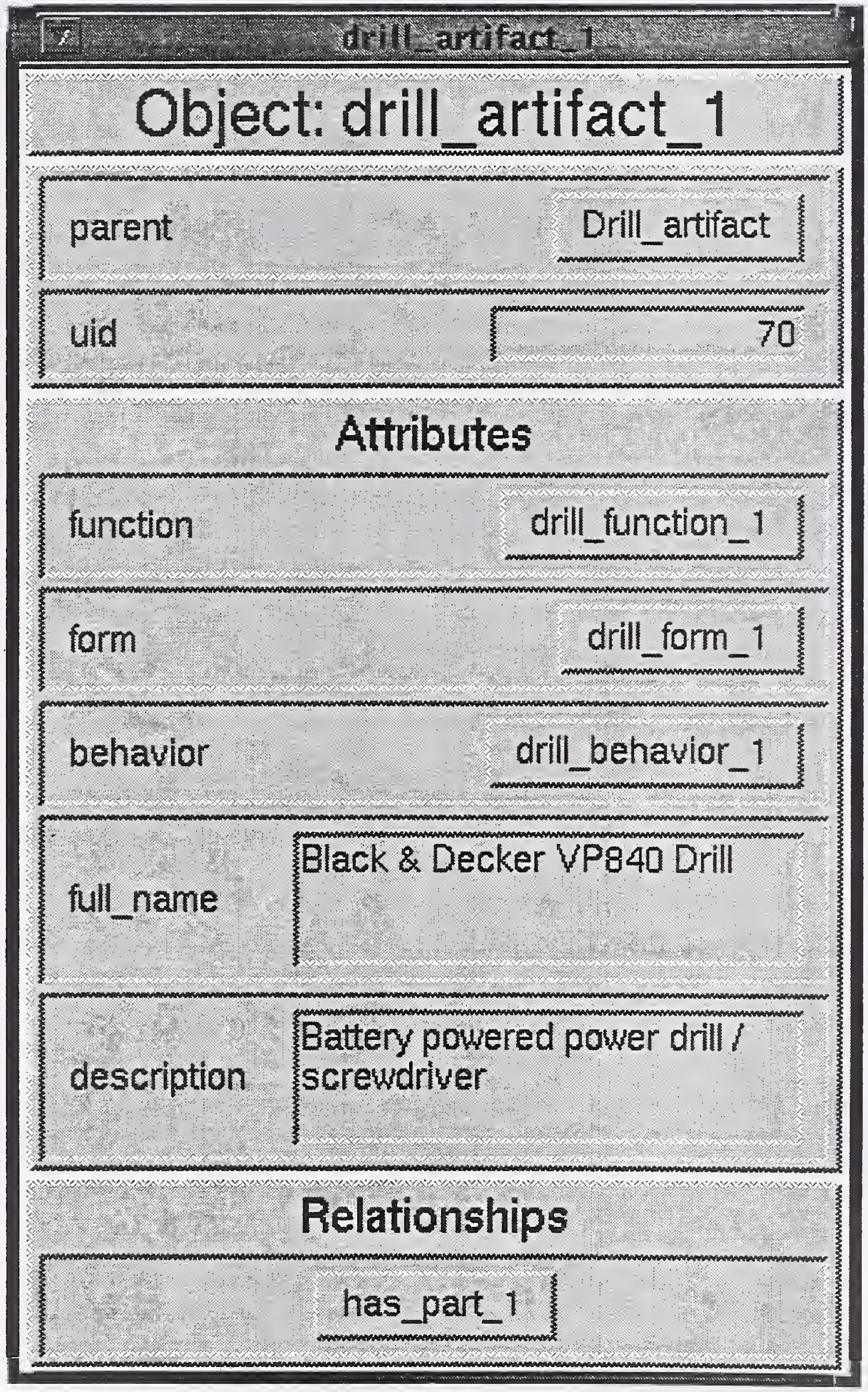
NIST

Design Case Study Framework

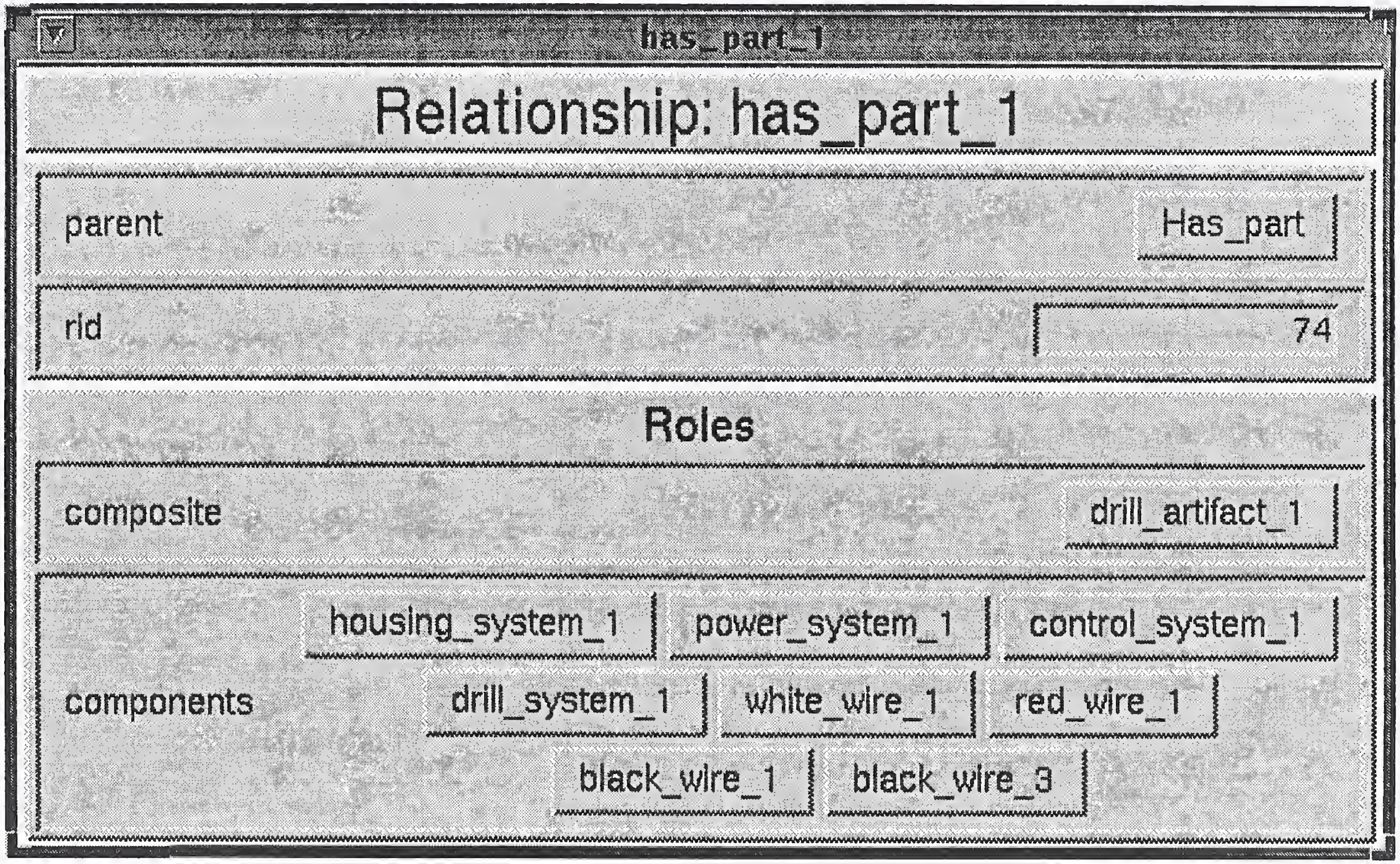


NIST

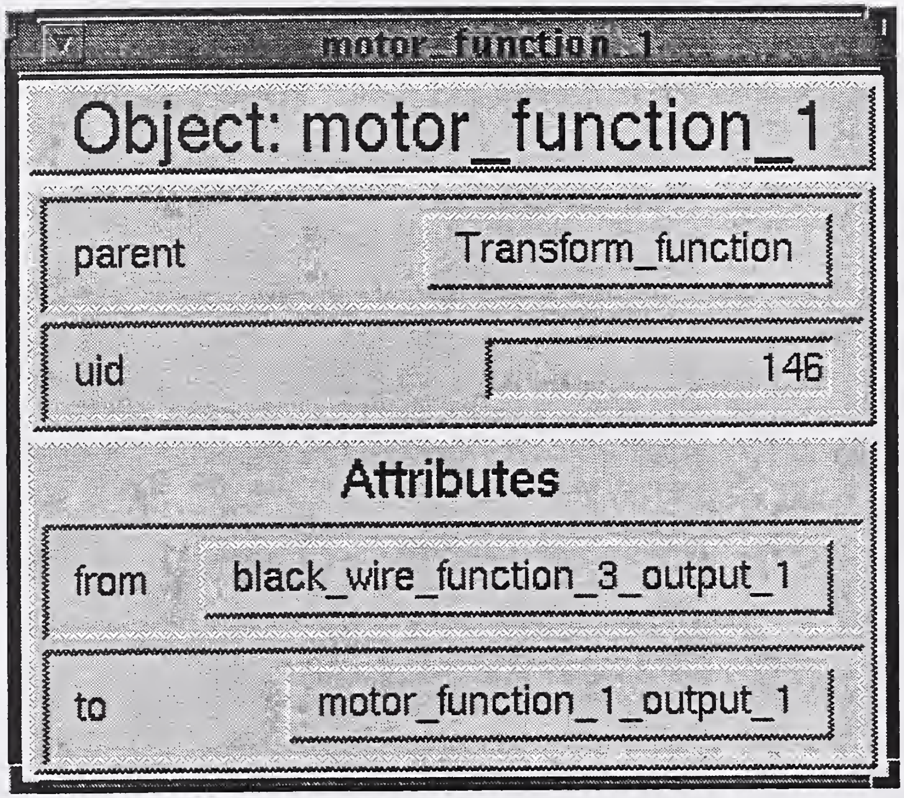
Design Case Study Framework

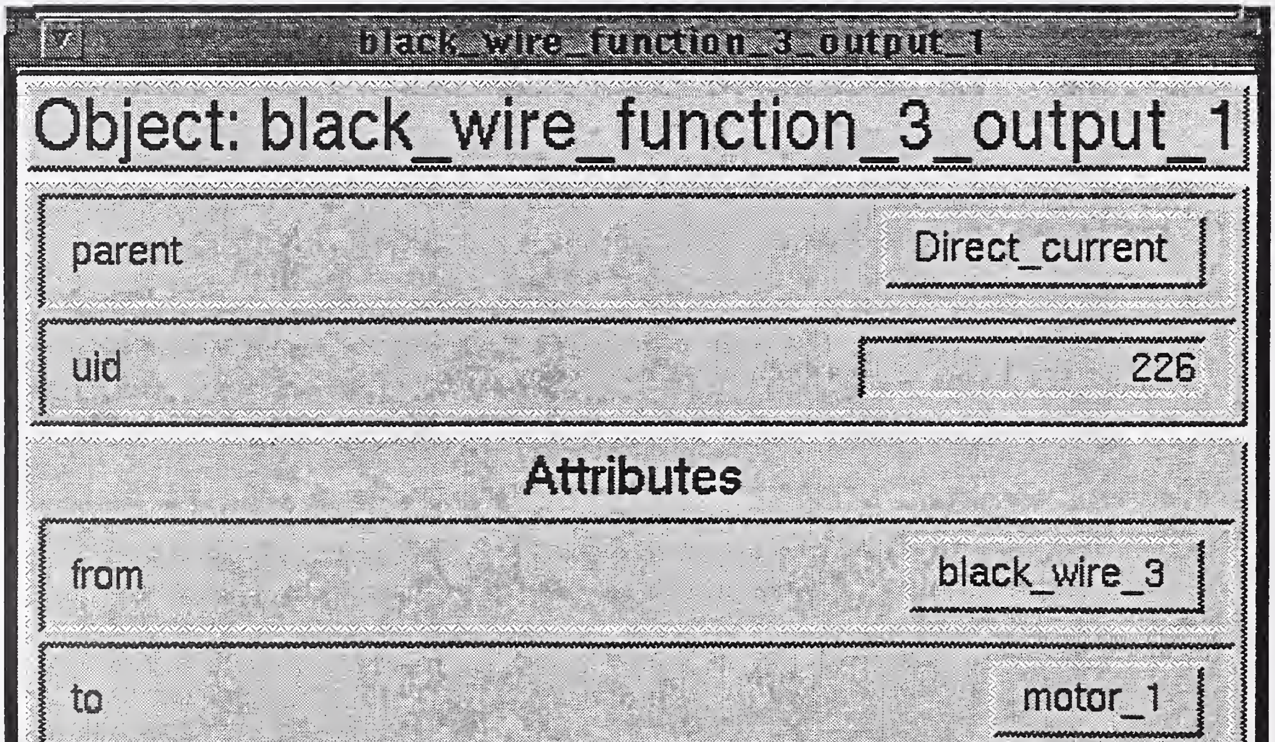


Design Case Study Framework



Design Case Study Framework





Design Case Study Framework

motor_function_1_output_1

Object: motor_function_1_output_1

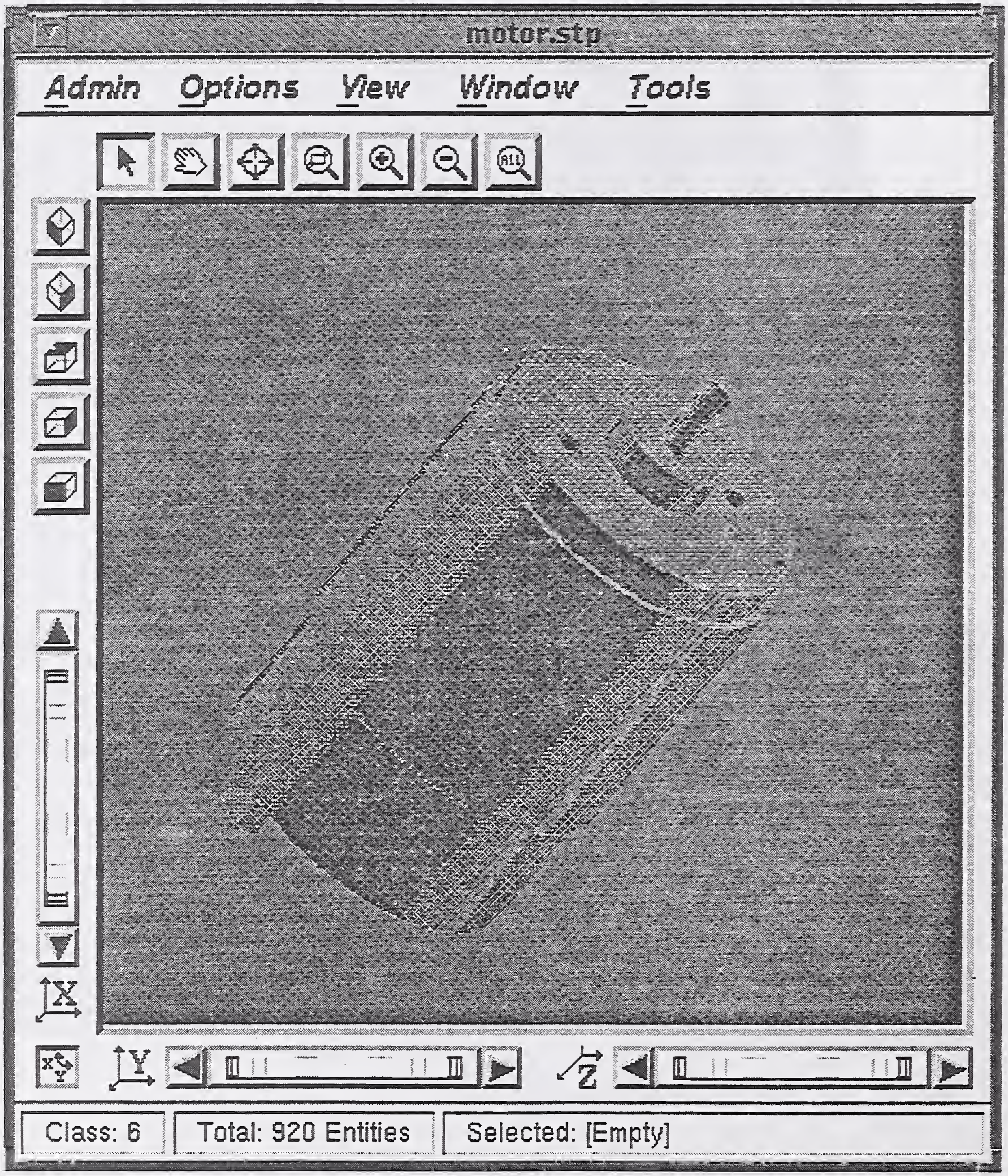
parent	Rotational_motion	
uid	230	

Attributes

from	motor_1	
to	clutch_base_1	
parameters	bit_function_1_output_1_speed	bit_function_1_output_1_direction

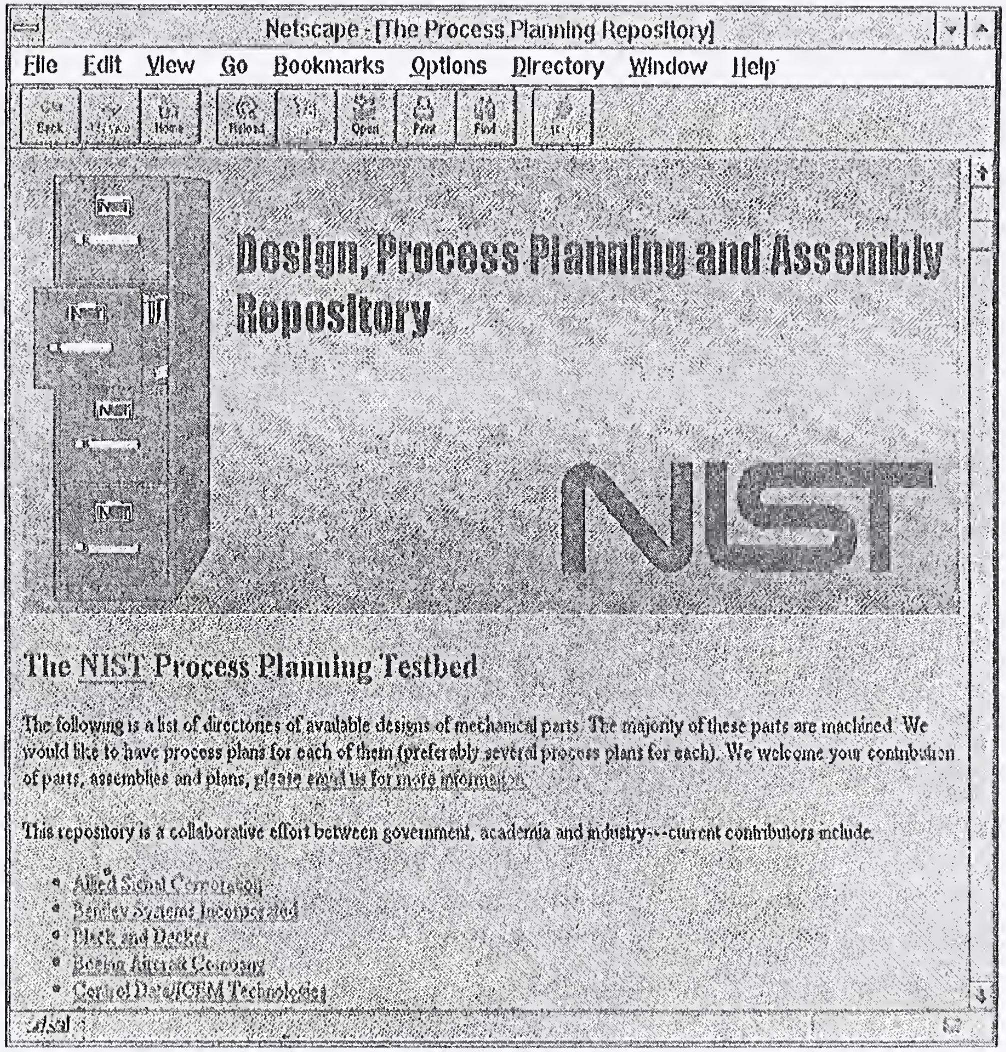
NIST

Design Case Study Framework



NIST Design, Planning, Assembly Repository

- Over 750M of designs, solid models and assemblies
- Enables national and international participation
- Links in related resources
- Facilitates collaboration
- Simplifies FTP access
- Can be coupled with intelligent search tools

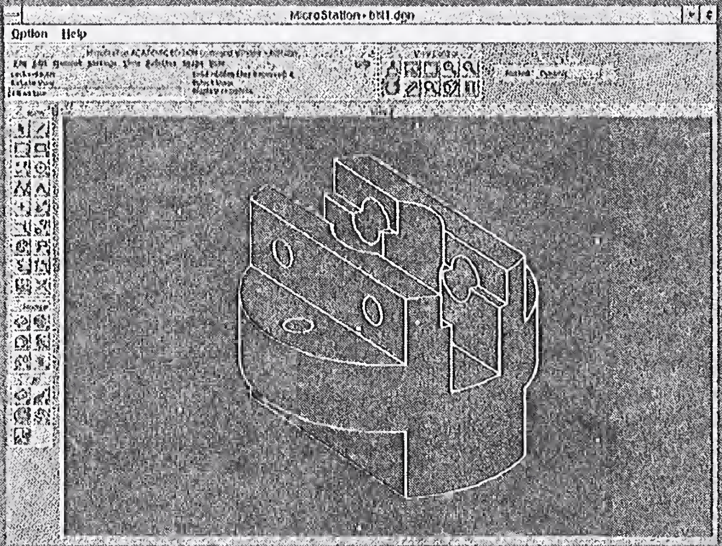


http://www.parts.nist.gov/parts

The Goals of the Repository...

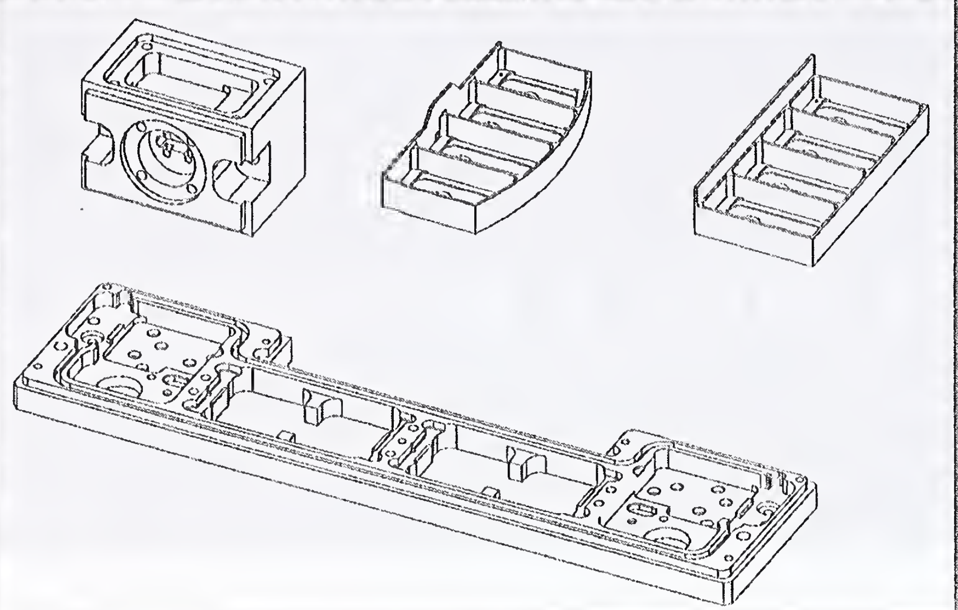
- Build a dynamic repository of CAD/CAM data
  - Individual CAD models
  - Assembly models
  - Process plans
  - Process models
  - Resource information
- Provide access through the Internet
- Identify benchmark cases and problems
- Facilitate collaboration
- Establish consensus on R&D issues

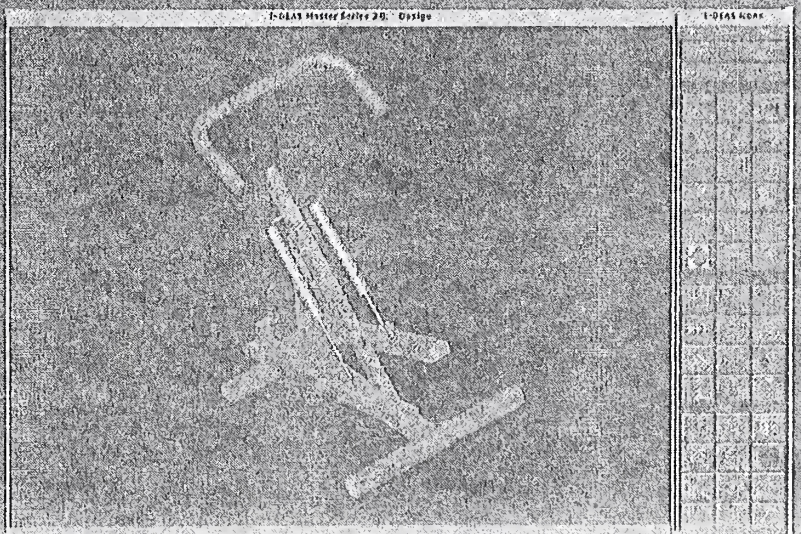
Example Repository Content:




```
(put_data 'pplans 'anc101_plan7 '(anc101_plan7
 header (header
        plan_id            anc101_plan7
        design_id          anc101_design
        setup_id           anc101_setup7
        workpiece_id       anc101_after6
        material           aluminum
        fixture_id         vise
        vise_width         5.0
        mrsevs             anc101_mrsevs7)

 data   (data
           1   (1   work_element    initialize_plan
                    program_name    anc7nc)
           2   (2   work_element    set0_corner
                    tool_type_id    probe_0.25
```





Current Contibutors...

Academia	Industry
▪ Carnegie Mellon University	▪ Bentley Systems Inc.
▪ The University of Maryland, College Park	▪ SDRC
▪ University of Southern California	▪ Spatial Technologies
▪ Purdue University	▪ Control Data / ICEM Technologies
▪ University of Kansas	▪ Allied Signal
	▪ Black and Decker
	▪ Boeing

and the list is

Areas for Future Research

- Develop indexing schemes
- Populate repository with designs
- Web-based interfaces to repositories
- Standards for information not included in STEP
- Representation for design process, not just design artifact

NIST

Open Collaborative Engineering

Dr. Michael Case, Concurrent Engineering Research Laboratory
(m-case@cecer.army.mil)

The summary for this presentation can be found on page 31
14 slides follow

Open Collaborative Engineering

Michael Case
Engineering Processes Division
Planning and Management Laboratory

Design Repository Workshop, NIST, 19-20 November, 1996

US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

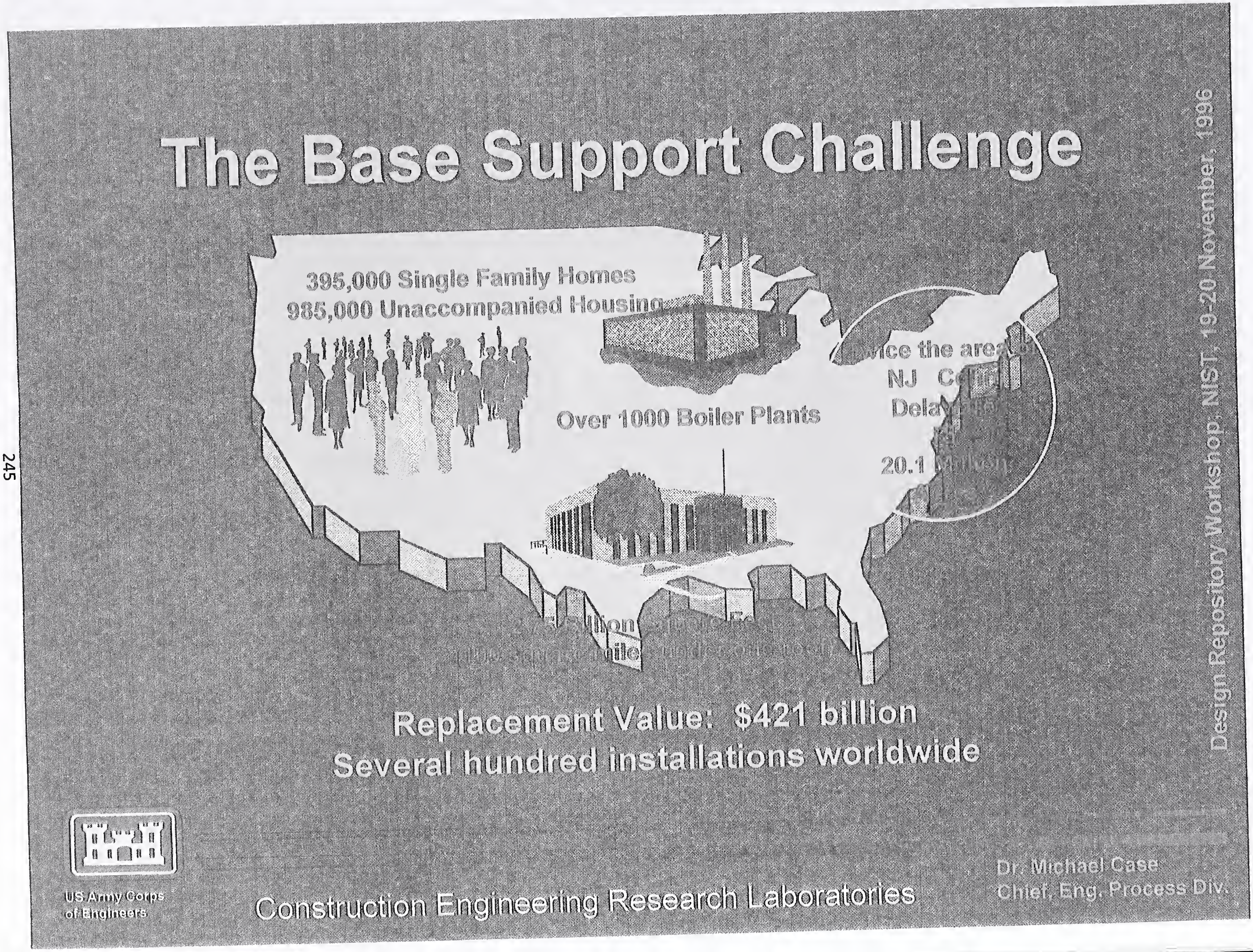
Overview

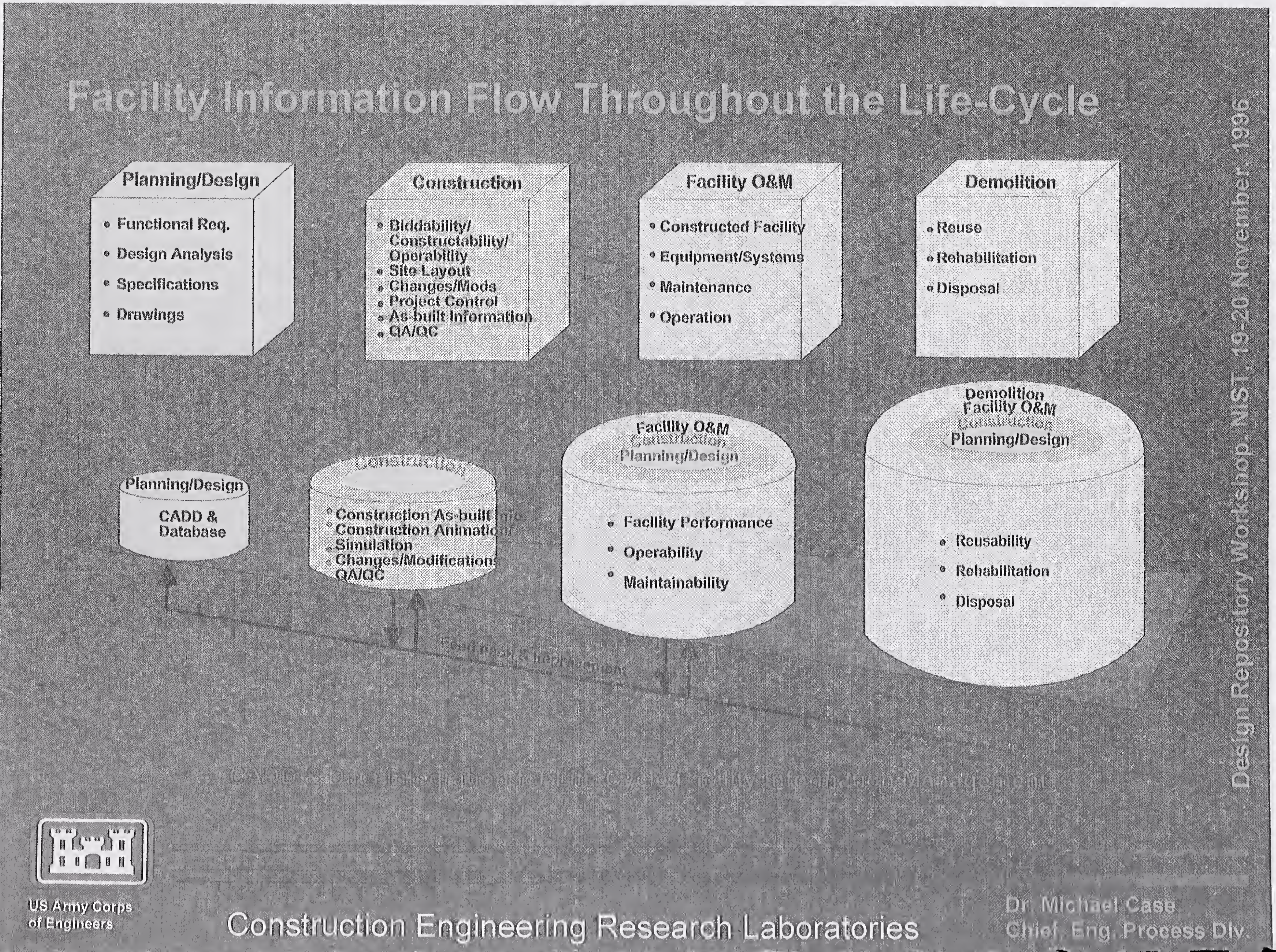
Problem Domain

Research Programs

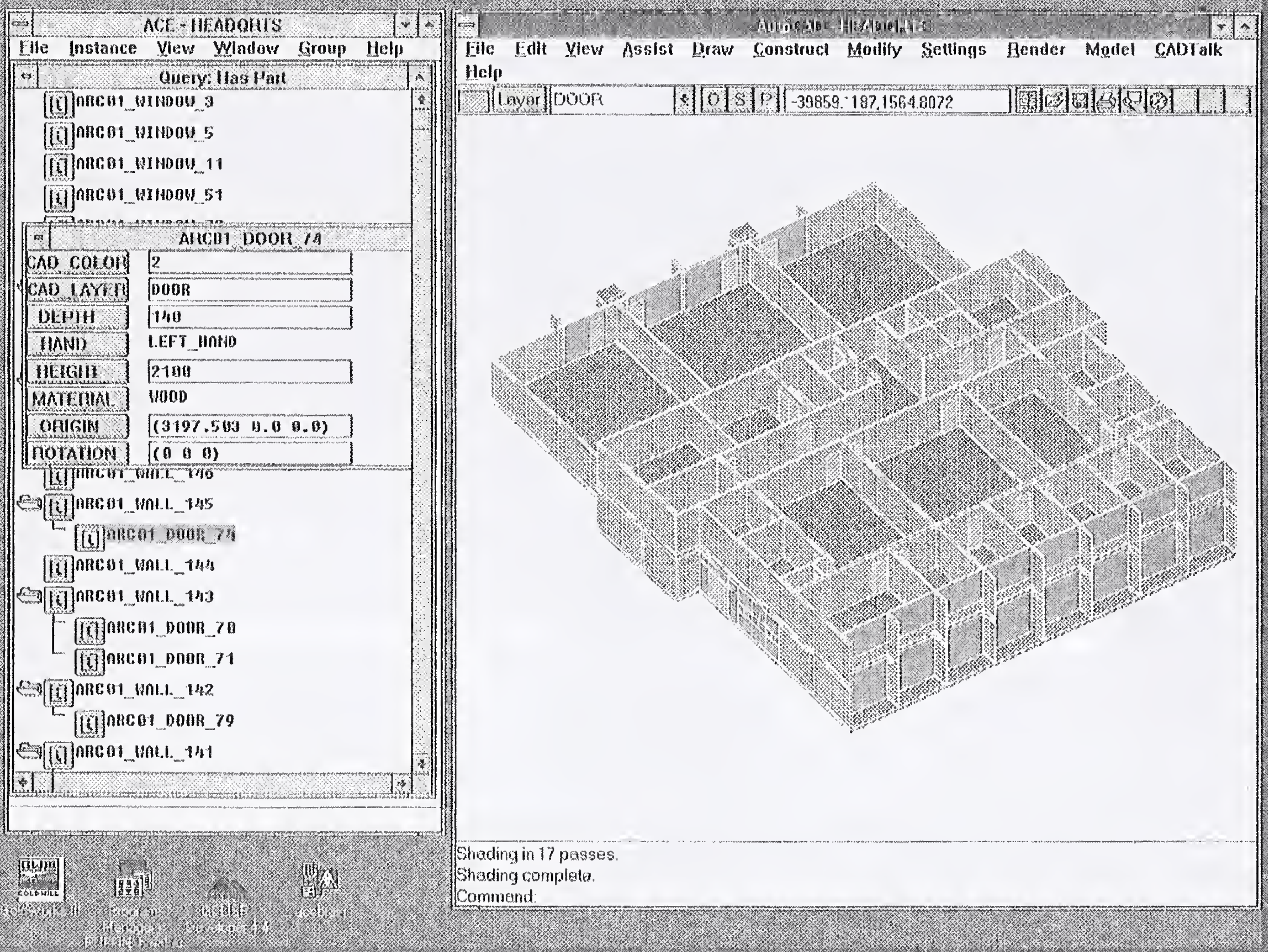
Design Repository Service Requirements

Dr. Michael Case
Chief, Eng. Process Div.


The Base Support Challenge
395,000 Single Family Homes
985,000 Unaccompanied Housing
Over 1000 Boiler Plants
NJ
20.1
Replacement Value: $421 billion
Several hundred installations worldwide
US Army Corps of Engineers
Construction Engineering Research Laboratories
Dr. Michael Case
Chief, Eng. Process Div.
Design Repository Workshop, NIST, 19-20 November, 1996


Facility Information Flow Throughout the Life-Cycle
Planning/Design
Functional Req.
Design Analysis
Specifications
Drawings
Construction
Biddability/ Constructability/ Operability
Site Layout
Changes/Mods
Project Control
As-built Information
QA/QC
Facility O&M
Constructed Facility
Equipment/Systems
Maintenance
Operation
Demolition
Reuse
Rehabilitation
Disposal
Planning/Design
CADD & Database
Construction
Construction As-built
Construction Animation
Simulation
Changes/Modifications
QA/QC
Facility O&M
Construction
Planning/Design
Facility Performance
Operability
Maintainability
Demolition
Facility O&M
Construction
Planning/Design
Reusability
Rehabilitation
Disposal
Design Repository Workshop, NIST, 19-20 November, 1996
US Army Corps of Engineers
Construction Engineering Research Laboratories
Dr. Michael Case
Chief, Eng. Process Div.

Architect's Associate



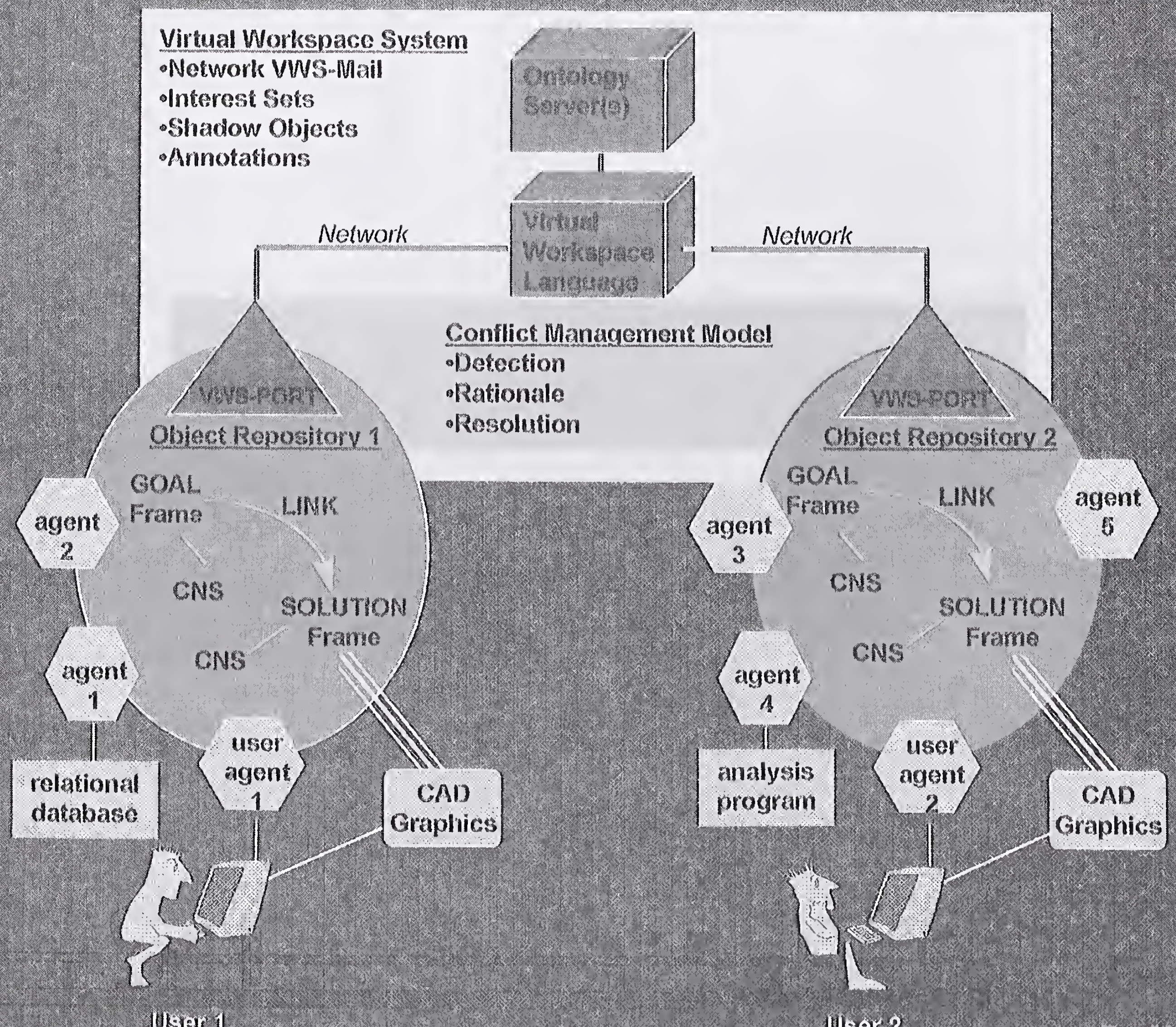
US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

Design Repository Workshop, NIST, 19-20 November, 1996

Discourse Model



US Army Corps of Engineers

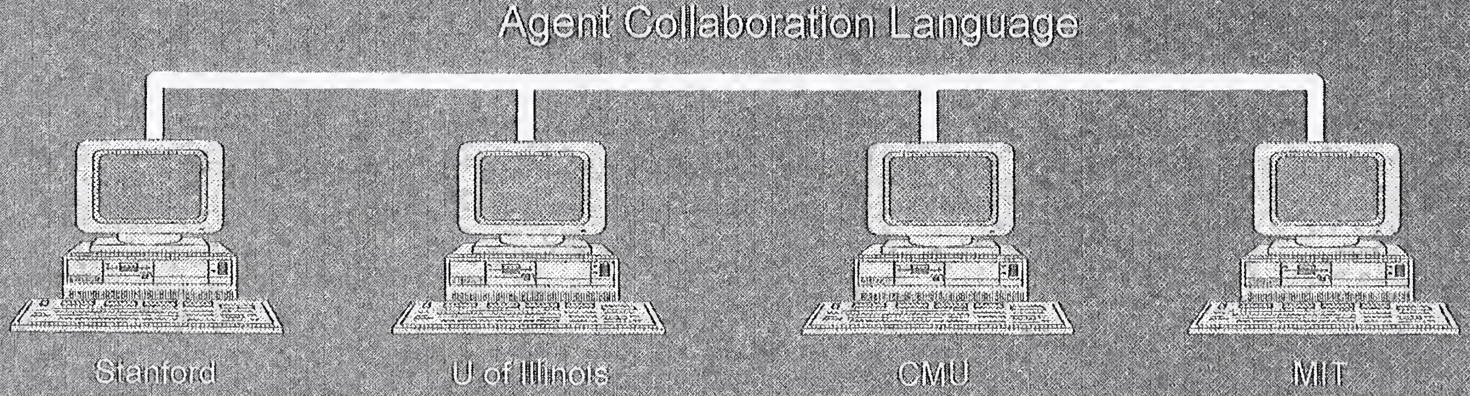
Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

Design Repository Workshop, NIST, 19-20 November, 1996

Agent Collaboration Language Project

- CERL, Stanford, CMU, MIT, UIUC
- Building Design on the Internet
- Facilitator Model - Translation
- Object Modeling Language
- Facilitator API



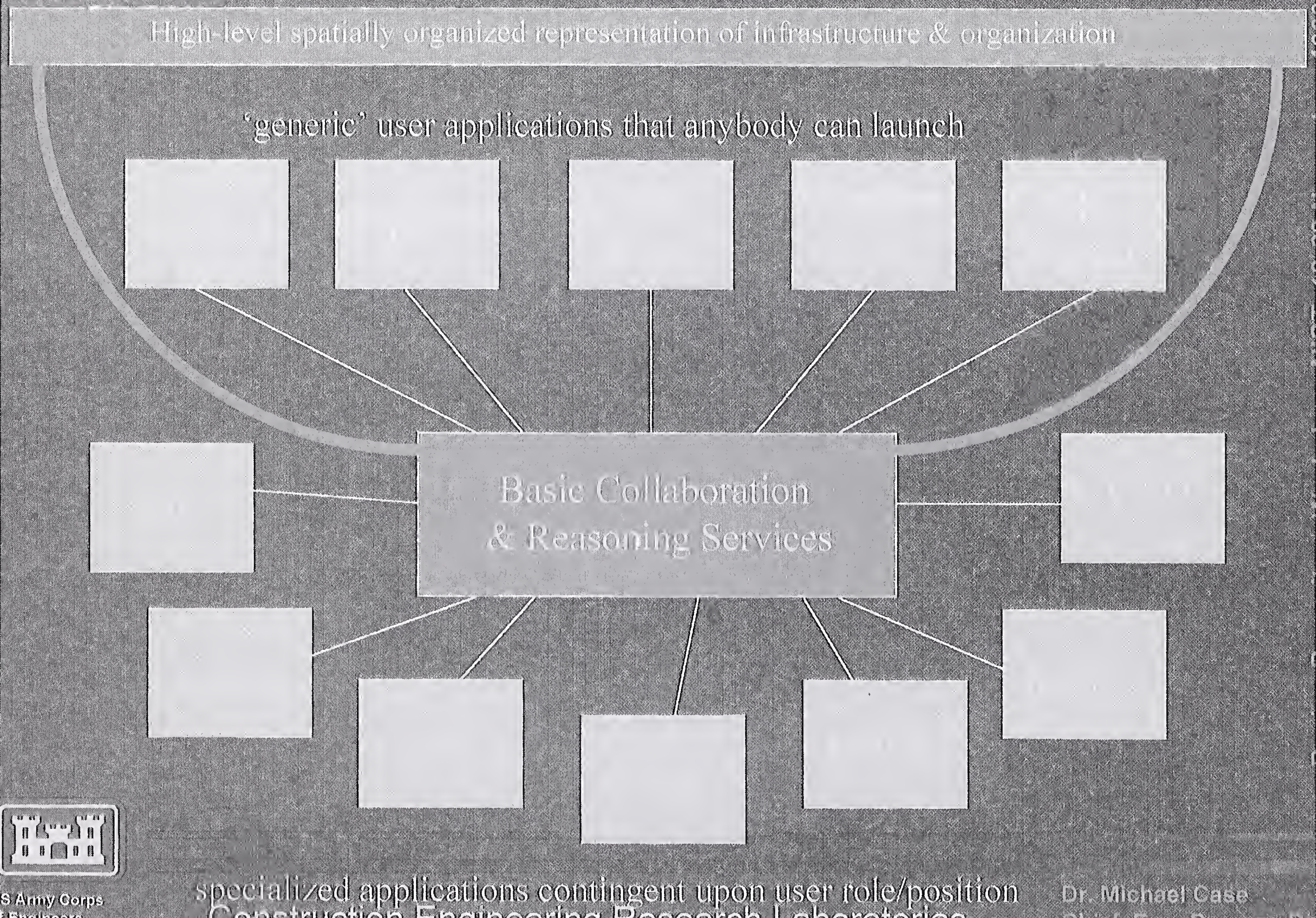
US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

Design Repository Workshop, NIST, 19-20 November, 1996

CITYSCAPE Architecture



US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

Design Repository Workshop, NIST, 19-20 November, 1996

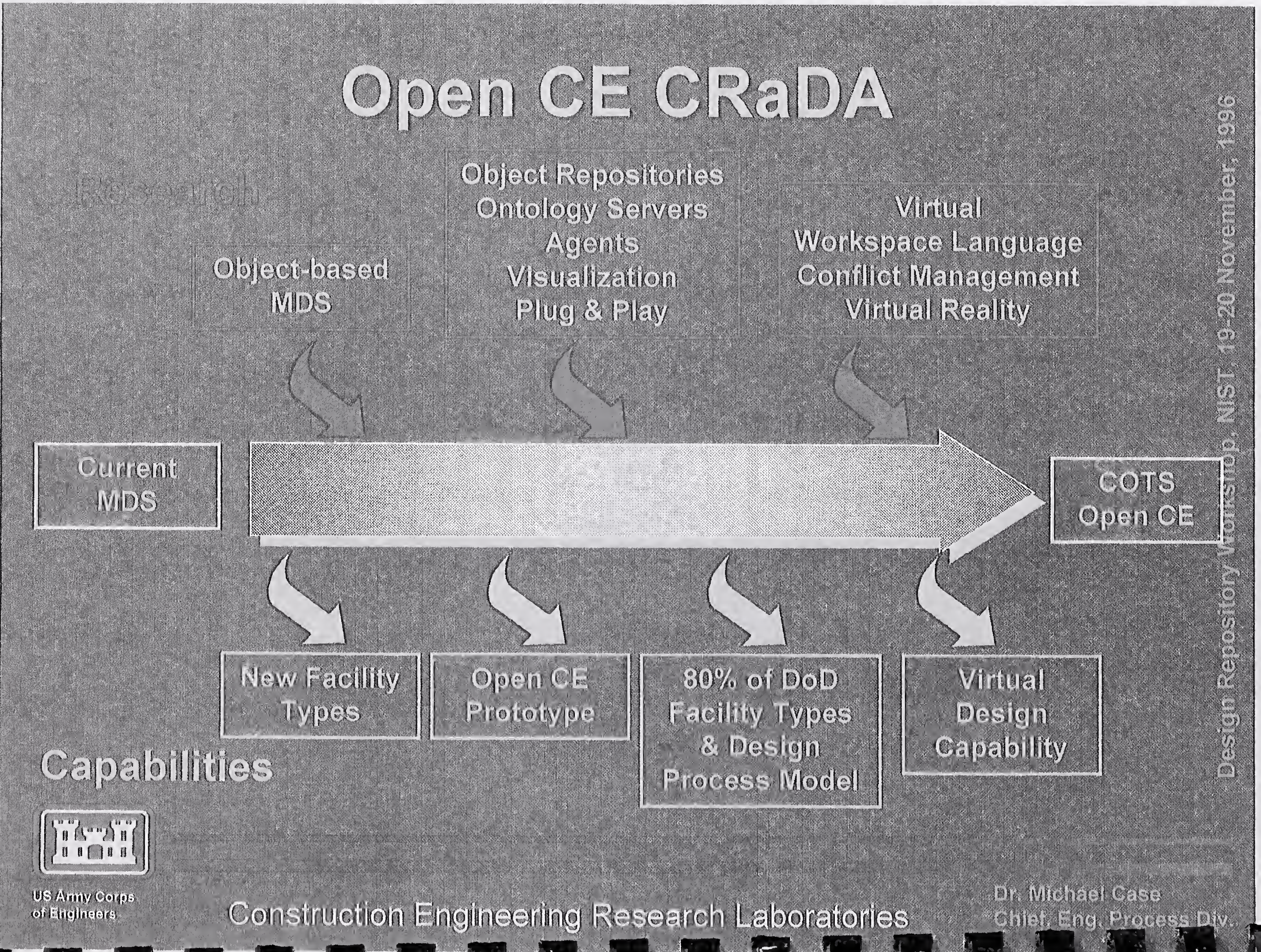
Modular Design System

- Developed by Louisville District (COE) as the Center of Standardization for Army Reserves.
- MDS - A component module design approach for defined facility types. Engineered Modules are configured within an established grid. It is developed within MicroStation.
- First Facility Type - Army Reserve/ National Guard Training Centers (demonstrated 540 to 140 days)
- MDS produces construction documents, specifications and cost estimates for a facility design.

US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.


Open CE CRaDA
Research
Object-based MDS
Object Repositories
Ontology Servers
Agents
Visualization
Plug & Play
Virtual
Workspace Language
Conflict Management
Virtual Reality
Current MDS
COTS Open CE
New Facility Types
Open CE Prototype
80% of DoD Facility Types & Design Process Model
Virtual Design Capability
Capabilities
US Army Corps of Engineers
Construction Engineering Research Laboratories
Dr. Michael Case
Chief, Eng. Process Div.
Design Repository Workshop, NIST 19-20 November, 1996

Open CE Component Service Needs

- Open Software Architecture (plug and play)
- Modeling
  - Object and relationship repositories
  - Ontology (Libraries, Modeling Languages)
  - Behavior
  - Versioning, Alternatives, Chronology
  - Retrieval strategies
  - Ownership and security
  - Rationale
  - Long Term Issues - Archival, schema evolution

Open CE Component Service Needs (cont.)

- Manipulation
  - Agent
  - Visualization (interpretation)
    - Reporting
    - Form-based
    - Decomposition
    - VR

Open CE Component Service Needs (cont.)

Collaboration (virtual meetings)
- Voice/Video
- Interest
- Replication
- Messaging
- Translation
- Conflict Management

US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

Design Repository Workshop, NIST, 19-20 November, 1996

Research and Development

- What is the agenda?
- What's here now?
- What's here but not useful yet?
- Where does the research need to focus?

Design Repository Workshop, NIST, 19-20 November, 1996

US Army Corps of Engineers

Construction Engineering Research Laboratories

Dr. Michael Case
Chief, Eng. Process Div.

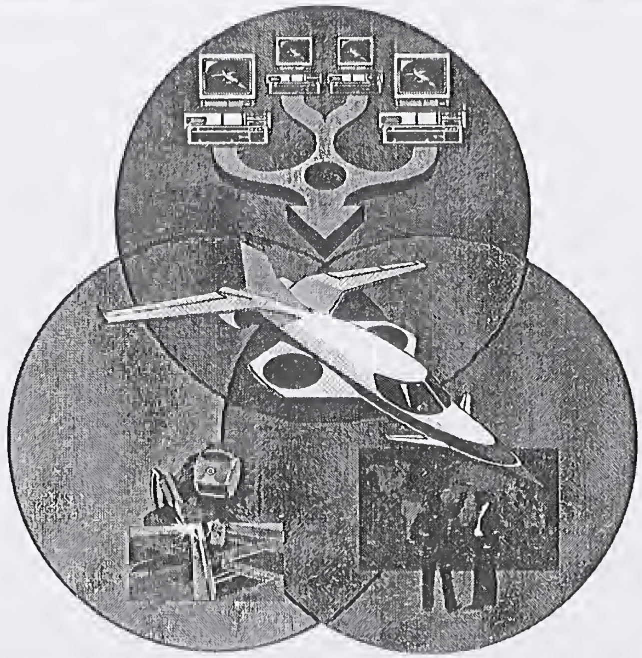
Integrated Design Systems: Aeronautics Design Process Improvement

Dr. David E. Thompson, NASA Ames Research Center,
Computational Sciences Division (dethompson@mail.arc.nasa.gov)

The summary for this presentation can be found on page 33
20 slides follow

Integrated Design Systems

- Aeronautics Design Process Improvement -



Dr. David E. Thompson
dethompson@mail.arc.nasa.gov

Integrated Design System
- Overarching Technology Goals -

Integrated Design Systems will coalesce a variety of technologies to address requirements for design process improvements that arise in aero and space:

- Immersive Environments and Visualization
- Infrastructure for Secure Collaborative Computing
- Systems Integration Tools and Databases
- Intelligent Computational Tools and Modeling Methods

Integrated Design System
- Overarching Technology Goals -

Immersive Environments and Visualization

adapt or develop tools that support multi-sensory synthetic environments and scientific visualization and exploration capabilities for design, test, manufacturing, and operations

Infrastructure for Secure Collaborative Computing

extend commercial tools to support collaborative research, engineering, and project management across widely distributed, secure, paltform-independent, standards-based systems

Integrated Design System
- Overarching Technology Goals -

Systems Integration Tools and Databases

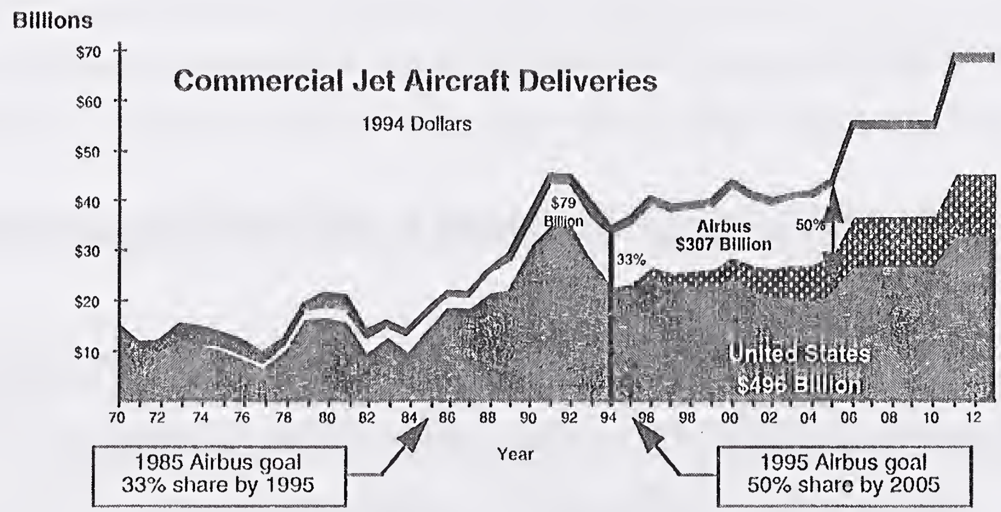
intelligent knowledge extraction and data mining technologies that support database management and navigation tools integration across a distributed, virtual computing space

Intelligent Computational Tools and Modeling Methods

develop or enable use of a variety of computationally intelligent tools and architectures that support intensive simulations for design, test, manufacturing, and operations, but that also deploy agent-mediators and intelligent wrappers to launch deep codes and retrieve appropriate data

Objective

To provide validated tools and infrastructure technologies that enable the U.S. Aerospace Industry to revolutionize <u>their</u> design processes

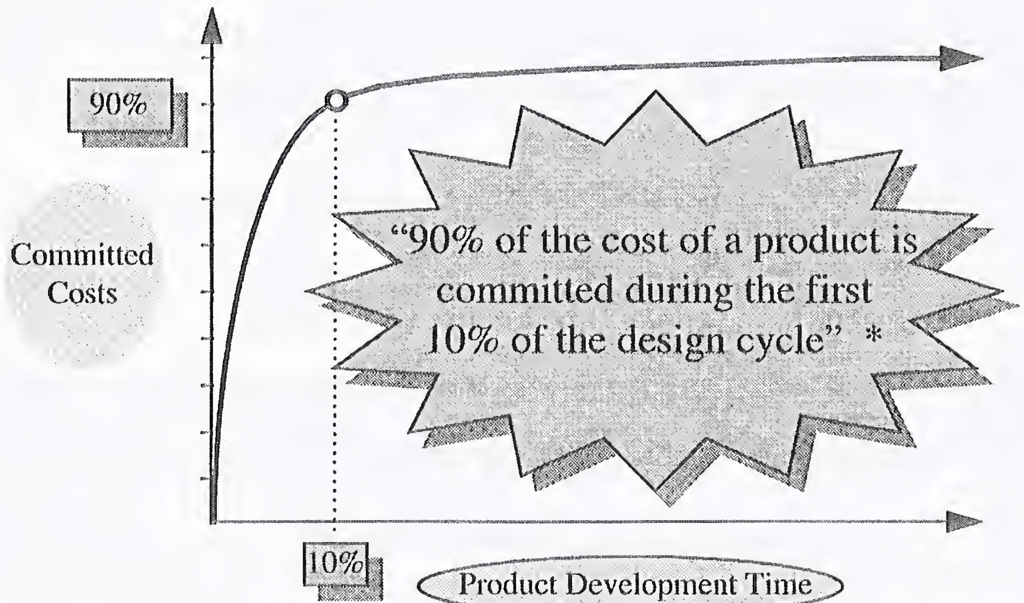


International competition demands a more agile design process

Program Focus

Enable the design revolution through exploitation of IT revolution

- Acquiring and sharing better design data, sooner
- Capturing and communicating requirements/constraints, sooner



* Industrial Strength Optimization at Boeing, by Paul Davis, SIAM News, Jan./Feb. 1996

Program Strategy

Shift from Defense to Offense

Abandon evolutionary strategy

1980's

Chief Designer/Individual Discipline Teams

1990's

Chief Designer Teams/Mildly Multi-Discipline Teams working with a Common Geometry Database

Pursue revolutionary strategy

2000

Integrated Design Environment including seamless mining of product data enabling mold-breaking design capabilities

Design Process Targets:

Single Thread Acceleration

Multiple Thread Acceleration

Product Thread Unification

Single Thread Acceleration (An Example)

24 Month Design Process (three 8 month cycles)

8 months	8	8

conf. design & analysis

WT model design & fab

test

analysis & re-design

Ongoing Efforts

- Elimination of One Cycle
- Reduction in Time/Cost of CFD
- Reduction in WT Model Design and Fab

12 months

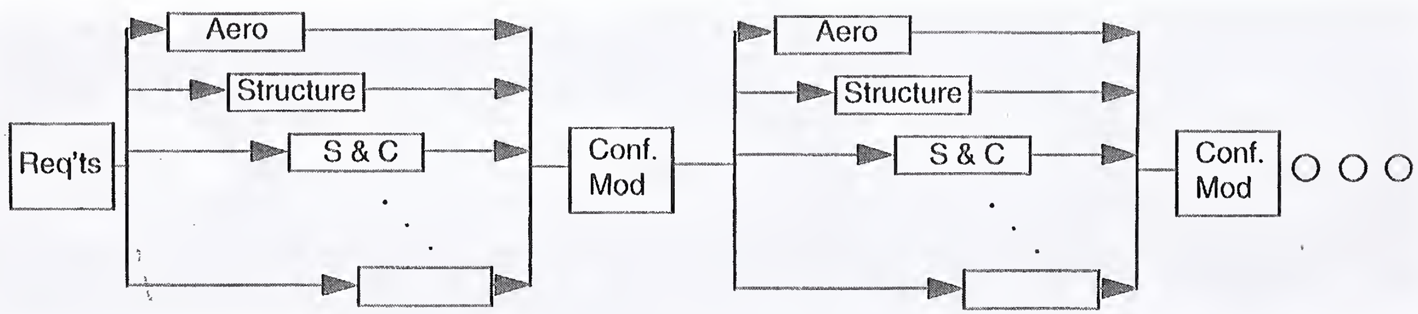
New Initiative

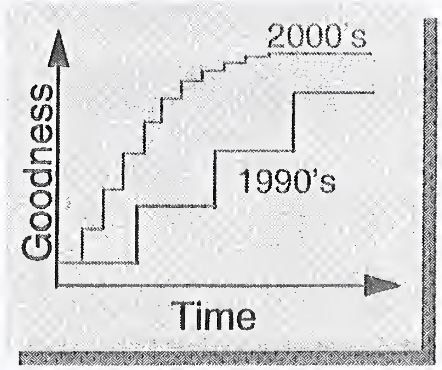
- Communications Infrastructure
- Intelligent Tools and Methods
- Knowledge Extraction

6 months

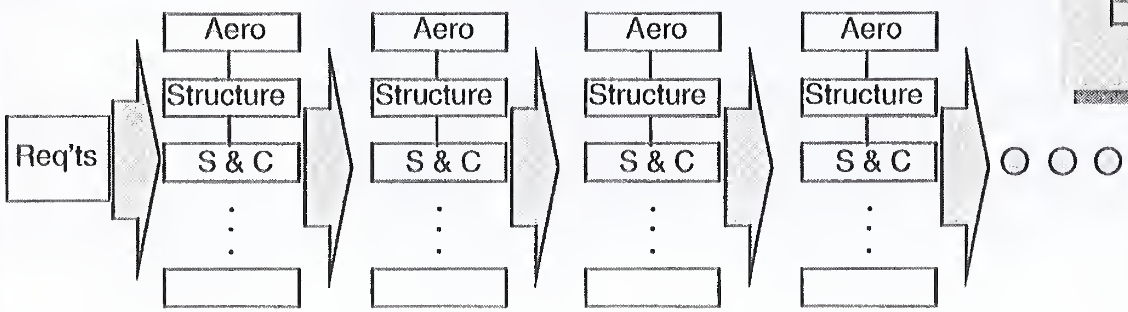
Multiple Thread Acceleration

1990's Semi-serial low-frequency cycles with discrete communications

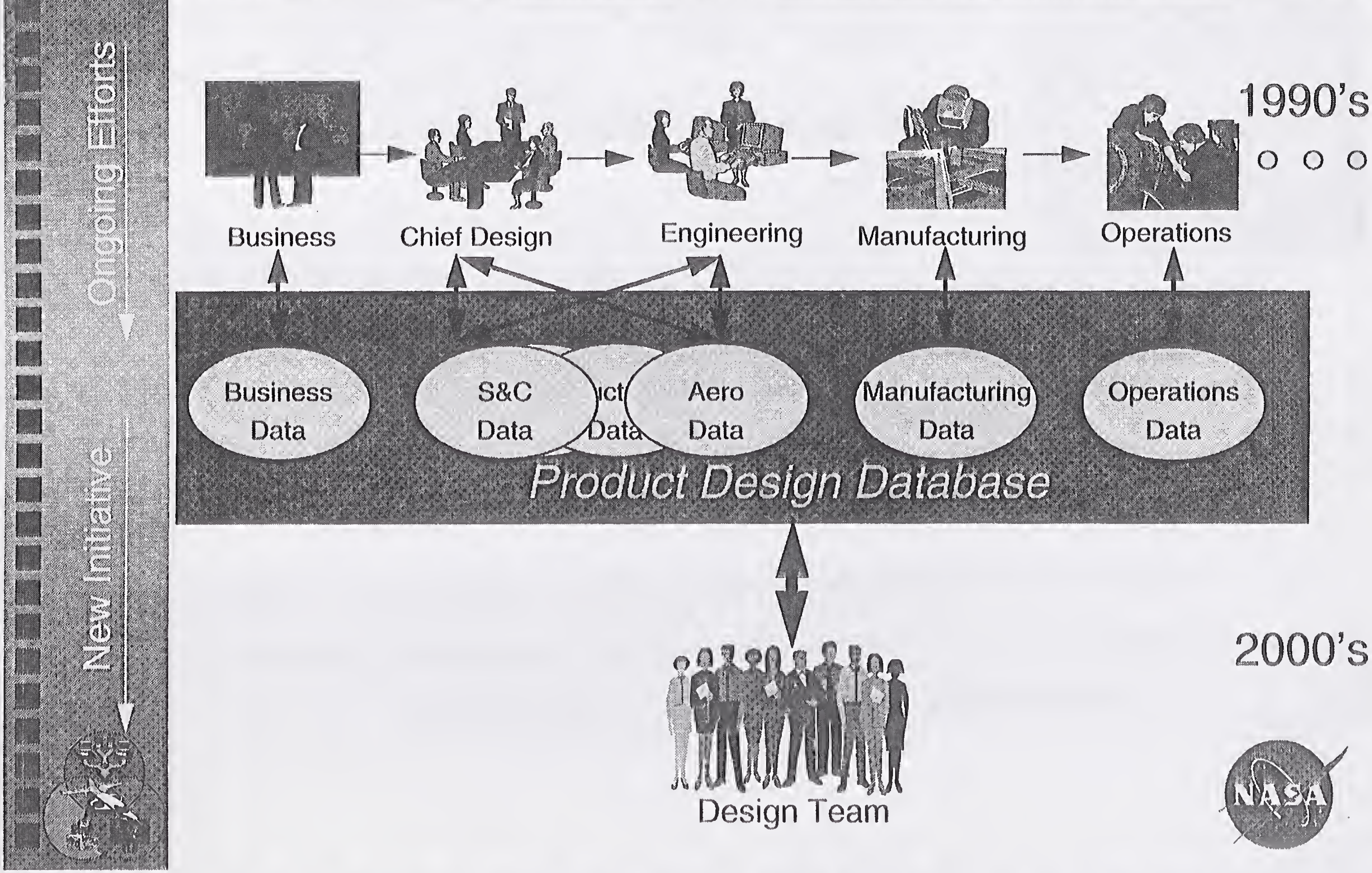




2000's Concurrent high-frequency cycles with continuous communications



Product Thread Unification



A Representation of the Design Domain

Design Process

	Single Thread	Multiple Thread	Product Thread
Data Acquisition	Aerodynamics or Structures or ...	Aeroelastics or ...	Engineering and Manufacturing and Business and ...
Data Organization & Mgmt.	Homogeneous Data		Severely Heterogeneous Data
Knowledge Extraction & Presentation	C_D or Weight or ...	C_D as a function of Weight	Profit as a function of launch date
Actions (Example)	Refine Product Geometry	Update Discipline Requirements	Alter Design Goal

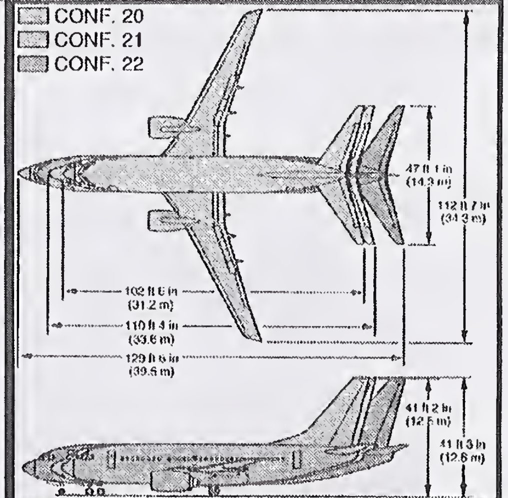
Current NASA Activities

New Initiative Activities

MB-797-11, CONFIGURATION 22

RADAR SELECTION

Mary S.
Designer
5/2/04-10:34 AM



PROCESS:

1: Acquire & present data & constraints

2: Collaborative evaluation of design alternatives with real-time advising

3: Implement design up-date

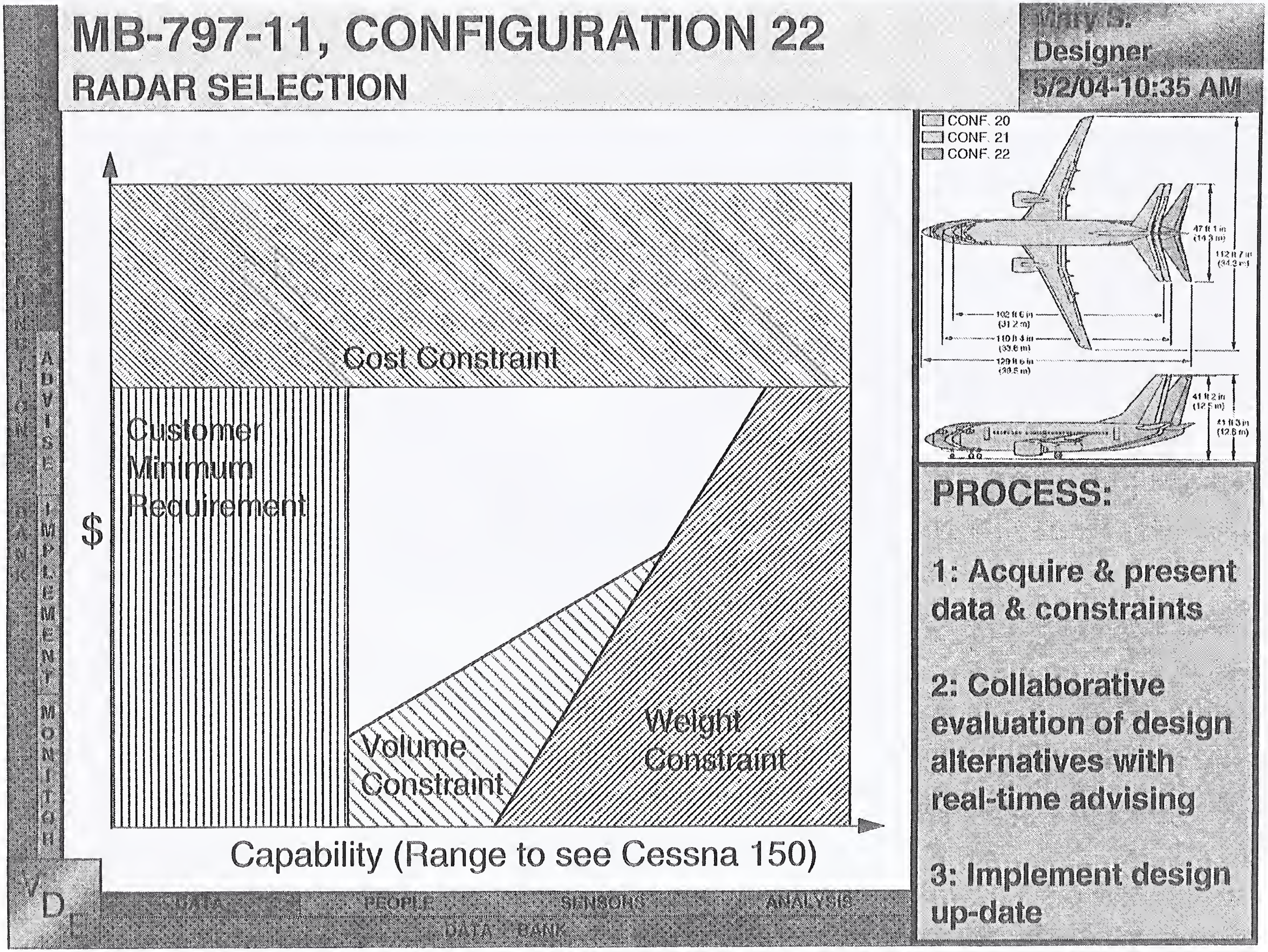
ACQUIRE | PRESENT | ADVISE | IMPLEMENT | MONITOR

FUNCTION BANK

VDE

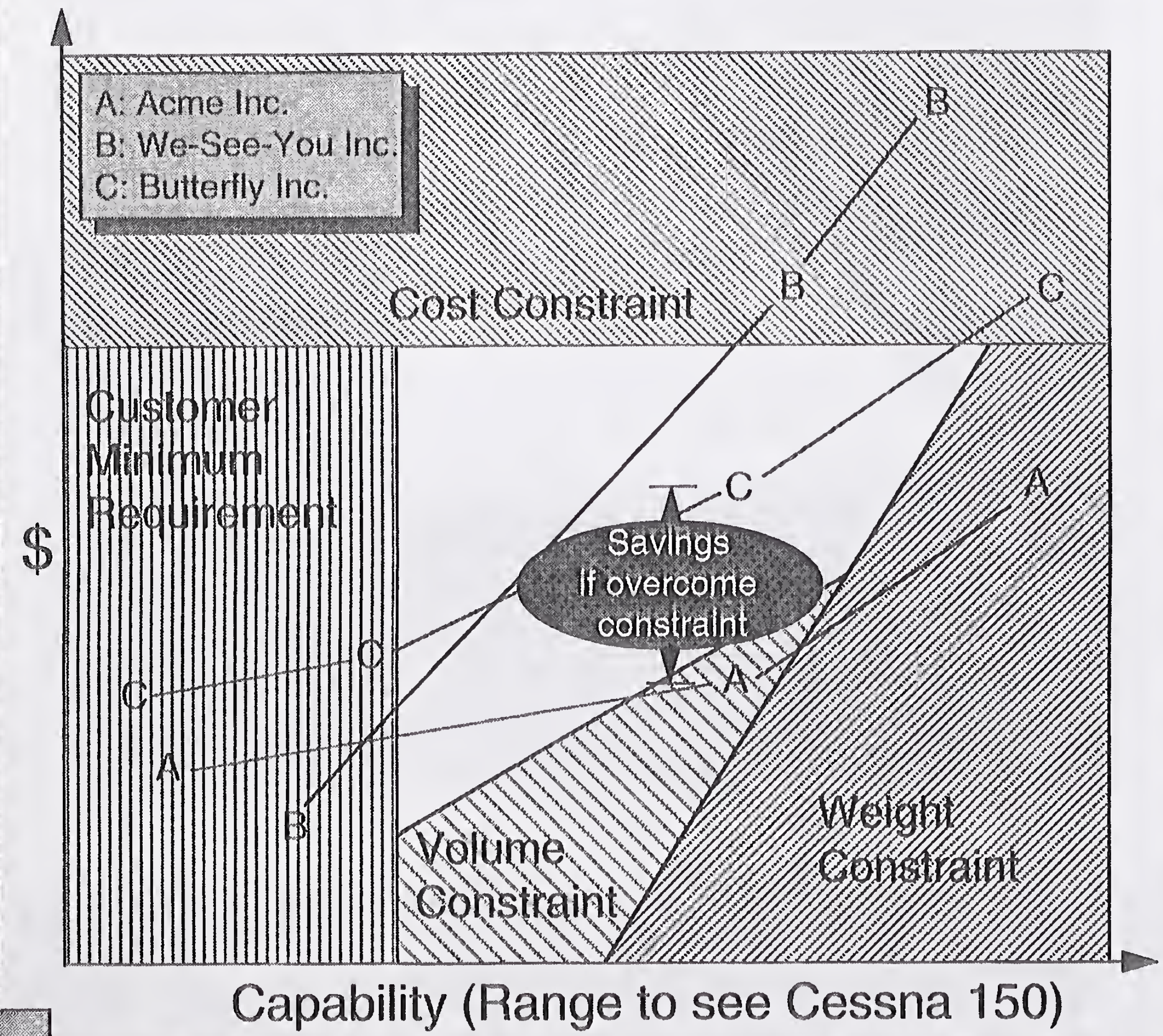
DATA | PEOPLE | SENSORS | ANALYSIS

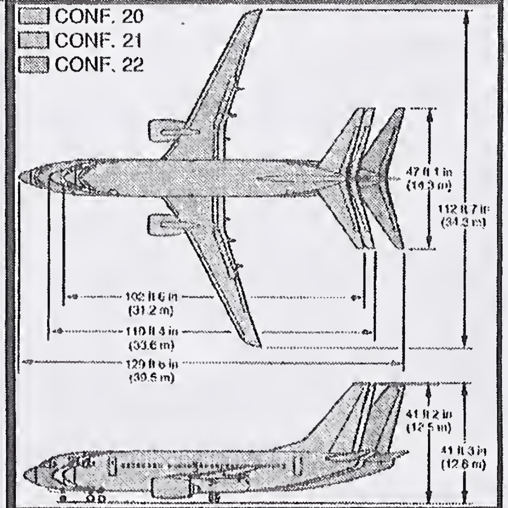
DATA BANK


MB-797-11, CONFIGURATION 22
RADAR SELECTION
Mary S.
Designer
5/2/04-10:35 AM
$
Capability (Range to see Cessna 150)
Cost Constraint
Customer Minimum Requirement
Volume Constraint
Weight Constraint
CONF. 20
CONF. 21
CONF. 22
47 ft 1 in (14.3 m)
112 ft 7 in (34.3 m)
102 ft 6 in (31.2 m)
110 ft 4 in (33.6 m)
129 ft 6 in (39.5 m)
41 ft 2 in (12.5 m)
41 ft 3 in (12.6 m)
PROCESS:
1: Acquire & present data & constraints
2: Collaborative evaluation of design alternatives with real-time advising
3: Implement design up-date
ADVISE
IMPLEMENT
MONITOR
DATA
PEOPLE
SENSORS
ANALYSIS
DATA BANK
VDL

MB-797-11, CONFIGURATION 22
RADAR SELECTION

Mary S.
Designer
5/2/04-10:37 AM





PROCESS:

1: Acquire & present data & constraints

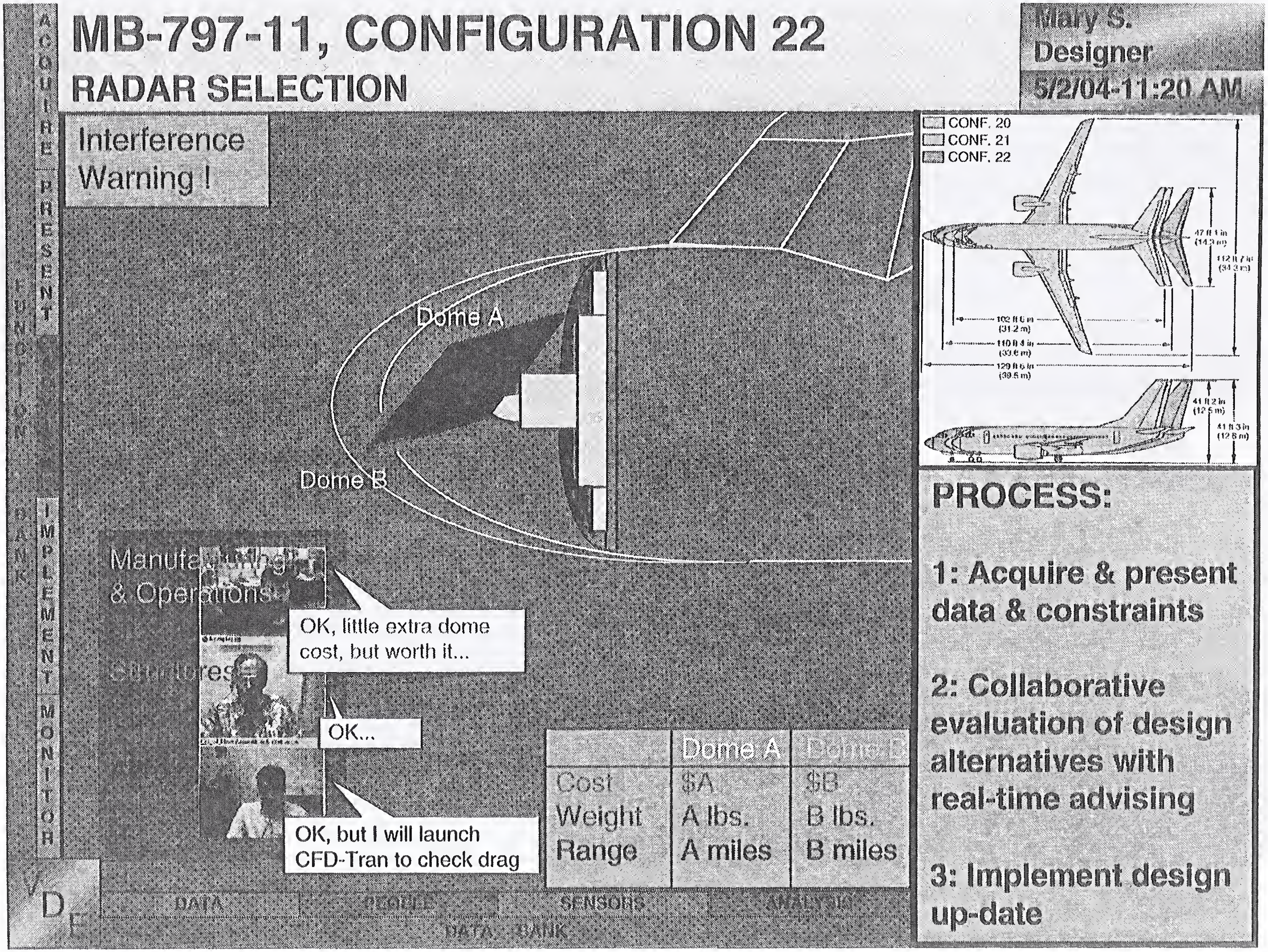
2: Collaborative evaluation of design alternatives with real-time advising

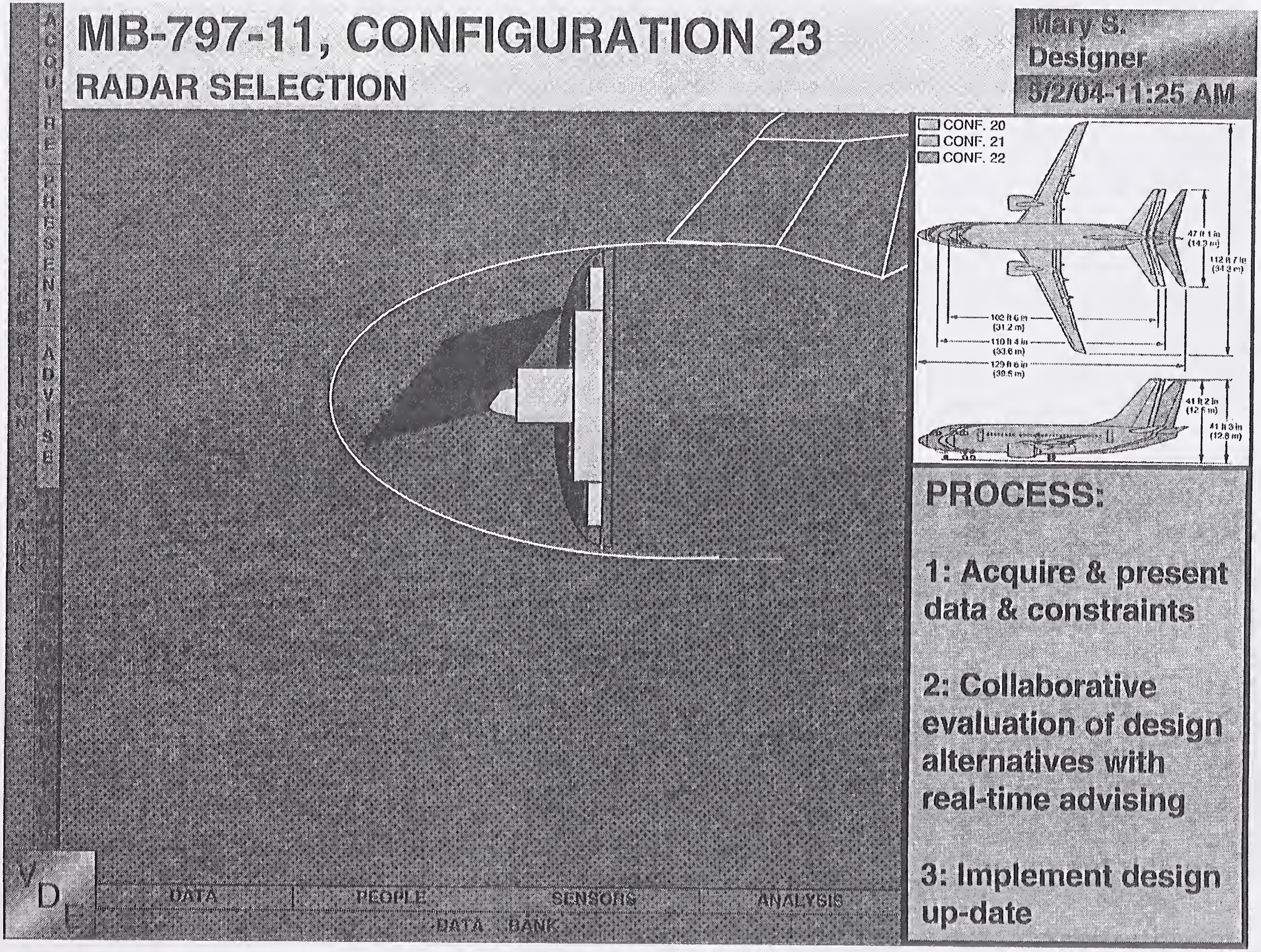
3: Implement design up-date

ADVISE IMPLEMENT MONITOR

DATA PEOPLE SENSORS ANALYSIS

DATA BANK


MB-797-11, CONFIGURATION 22
RADAR SELECTION
Mary S.
Designer
5/2/04-11:20 AM
Interference Warning !
ACQUIRE
PRESENT
IMPLEMENT
MONITOR
Dome A
Dome B
CONF. 20
CONF. 21
CONF. 22
Manufacturing & Operations
Structures
OK, little extra dome cost, but worth it...
OK...
OK, but I will launch CFD-Tran to check drag
Dome A
Dome B
Cost $A $B
Weight A lbs. B lbs.
Range A miles B miles
PROCESS:
1: Acquire & present data & constraints
2: Collaborative evaluation of design alternatives with real-time advising
3: Implement design up-date
DATA
PEOPLE
SENSORS
ANALYSIS
DATA BANK



MB-797-11, CONFIGURATION 23
RADAR SELECTION
Mary S.
Designer
8/2/04-11:25 AM
ACQUIRE
PRESENT
ADVISE
FUNCTION
BANK
CONF. 20
CONF. 21
CONF. 22
47 ft 1 in (14.3 m)
112 ft 7 in (34.3 m)
102 ft 6 in (31.2 m)
110 ft 4 in (33.6 m)
129 ft 6 in (39.5 m)
41 ft 2 in
41 ft 3 in (12.6 m)
PROCESS:
1: Acquire & present data & constraints
2: Collaborative evaluation of design alternatives with real-time advising
3: Implement design up-date
VDE
DATA
PEOPLE
SENSORS
ANALYSIS
DATA BANK

Investment Areas & *Example Products*

- Intelligent Database and Knowledge Extraction Techniques
  - *WHAT: Real-time data search, extraction, and information/knowledge presentation*
  - *WHY: Access to all stored data/information/knowledge during entire design process*
- Intelligent Tools and Modeling Representations
  - *WHAT: Configurable, automated deep-analysis (e.g., CFD, FEM, simulation...)*
  - *WHY: Allow early use of high-quality computational physics data*
- Smart Systems Integration Agents and Data Fusion
  - *WHAT: Configurable, automated multi-thread deep-analysis (e.g., aero-elastic)*
  - *WHY: Allow early use of high-quality multi-thread data*
- Collaborative Communication Tools and Infrastructure
  - *WHAT: Anytime video-con with white boarding, data/analysis presentation, parametric design manipulation, and complete configuration control*
  - *WHY: Real-time design review & update, with team awareness of configuration status*
- Immersive Environments and Navigation Capabilities
  - *WHAT: "Virtual Reality" data browsing and system control/monitoring capabilities*
  - *WHY: Improved "situational awareness" and knowledge extraction*
- Distributed, High Performance Simulation Design Environment
  - *WHAT: All data/processes distributed across heterogeneous computer systems*
  - *WHY: Only cost-effective, low-risk approach to support design process*

Program Plan

- Model the current aircraft design process, and quantify process improvement opportunities
- Acquire, adapt, and customize commercial tools/infrastructure technology
- Develop intelligent tool, search, data-fusion, knowledge extraction methodologies and capabilities, as required
- Deploy with industry a series of prototypes leading to robust, distributed, fully-immersive and collaborative design space analysis and advisory technologies

Technology Transfer Plan

Co-Development Ensures Technology Utilization

- Industry utilization planned from inception
  - 80% of net effort performed by U.S. industry/academia
  - Aerospace industry requirements define program goals
  - IT industry capabilities/plans map most appropriate program content
- Aerospace industry co-development
  - Real-time adaptation to evolving industry needs
  - Technology absorption accelerated through continuous industry development and prototyping technology
- IT industry co-development
  - Ensures state-of-the-industry technology utilization
  - Gives IT industry a window into design process customer base
- Procurement strategies
  - Utilize Space Act Authority for all technology development partnerships (JSRA, etc.)
  - Utilize NASA contracting vehicles for all services and goods (Task-order contracts, etc.)

Model-Based Support for Collaborative Engineering—Focus on Ontologies: How Things Work Project

Dr. Yumi Iwasaki, Stanford University (iwasaki@ksl.stanford.edu)

The summary for this presentation can be found on page 35
6 slides follow on three pages
A paper on this topic begins after page 375

Model-Based Support for Collaborative Engineering

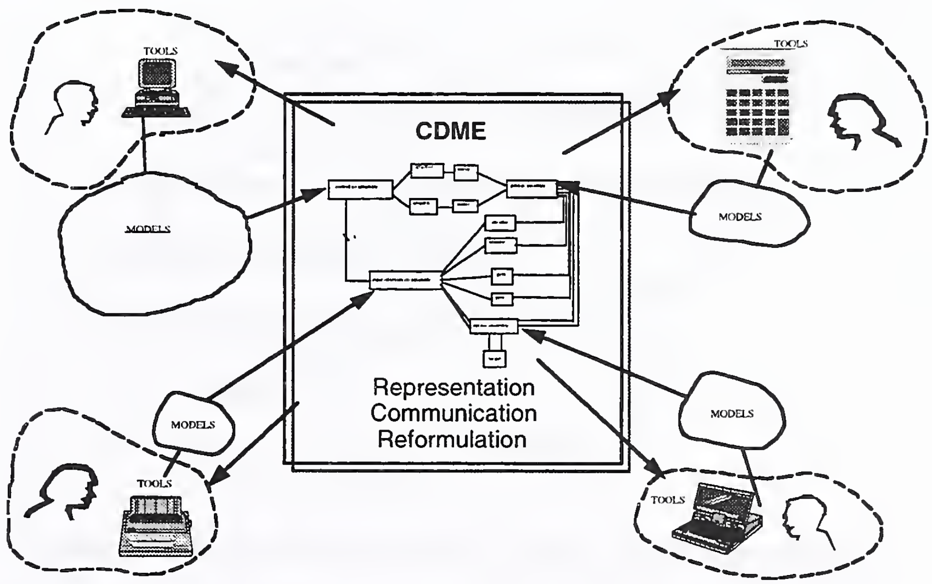
Focus on Ontologies

How Things Work project

Richard Fikes
Yumi Iwasaki
James Rice
Robert Engelmore
Todd Neller
Tony Loeser
Dana Clarke
Kentaro Oguchi

Knowledge Systems Laboratory
Stanford University

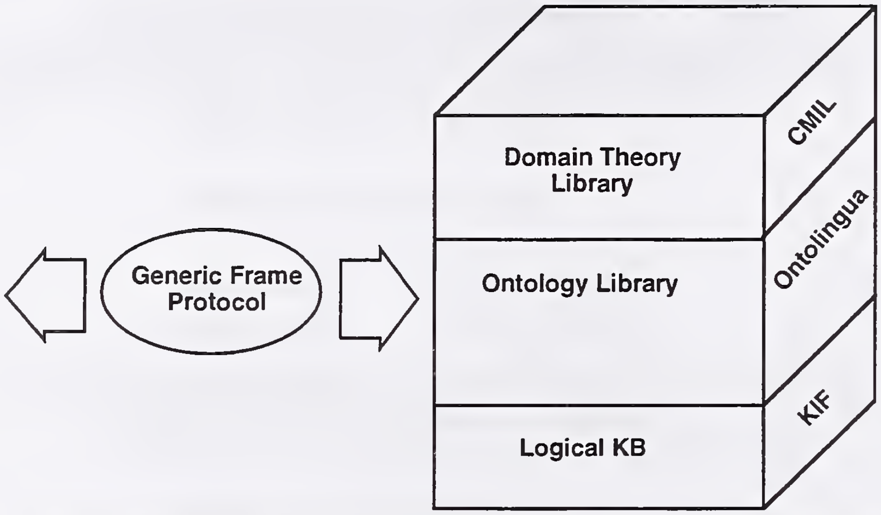
Distributed Collaborative Design with CDME



Knowledge Systems Laboratory, Stanford University

Yumi Iwasaki
Knowledge Systems Laboratory

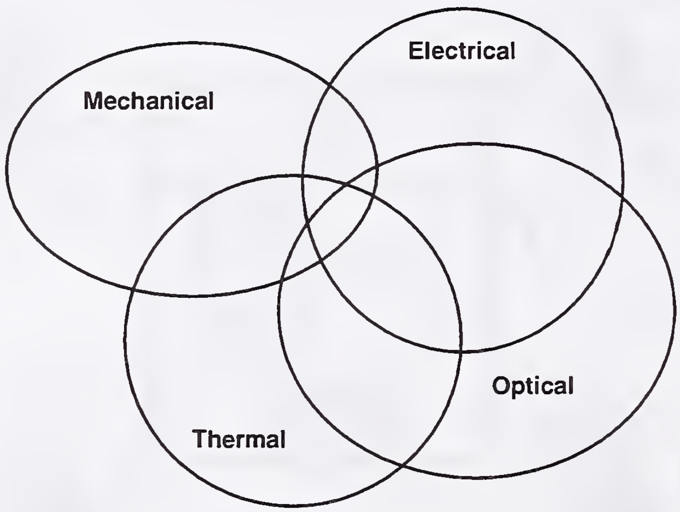
Representation Layers



12/2/96

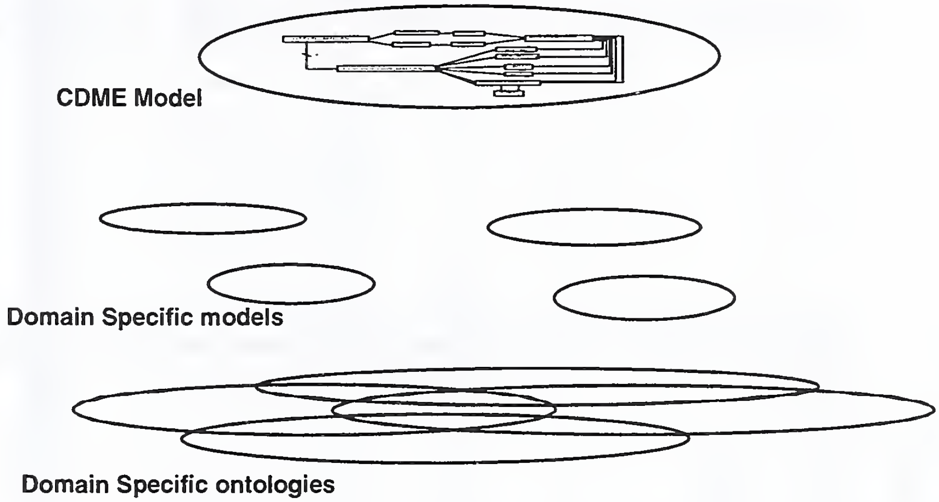
Knowledge Systems Laboratory, Stanford University

Domain Specific Ontologies



Knowledge Systems Laboratory, Stanford University

Domain Specific Ontologies



Knowledge Systems Laboratory, Stanford University

Summary

- Support collaborative modeling
- Declarative domain specific ontologies
- Global, high-level model
- Communication, coordination

12/2/96

Knowledge Systems Laboratory, Stanford University

The Shared Design Manager: A Repository for Integrated Product Data

Dr. Susan Urban, Arizona State University (s.urban@asu.edu)

The summary for this presentation can be found on page 36
7 slides follow

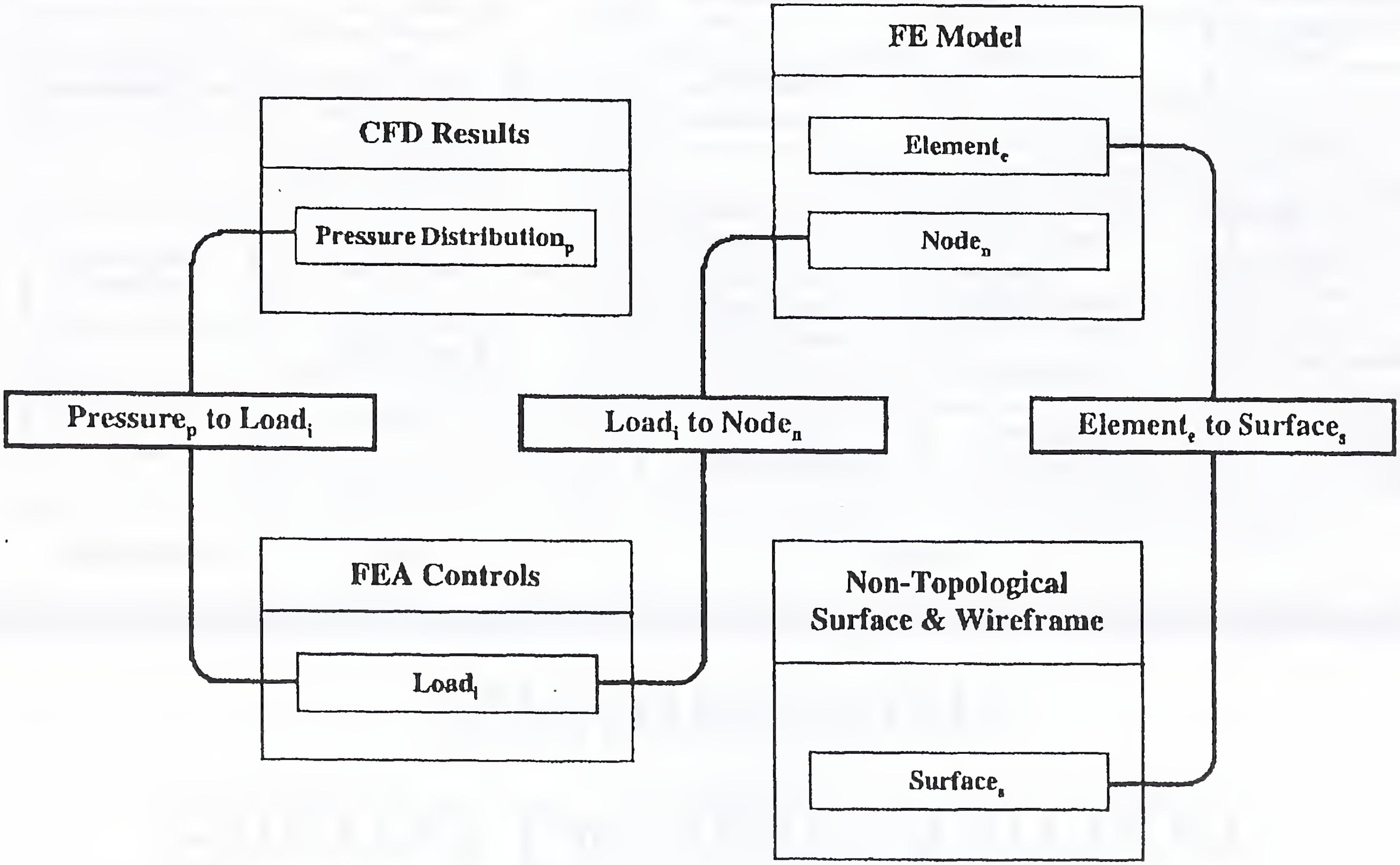
THE SHARED DESIGN MANAGER: A REPOSITORY FOR INTEGRATED PRODUCT DATA

Susan D. Urban
Arizona State University
s.urban@asu.edu

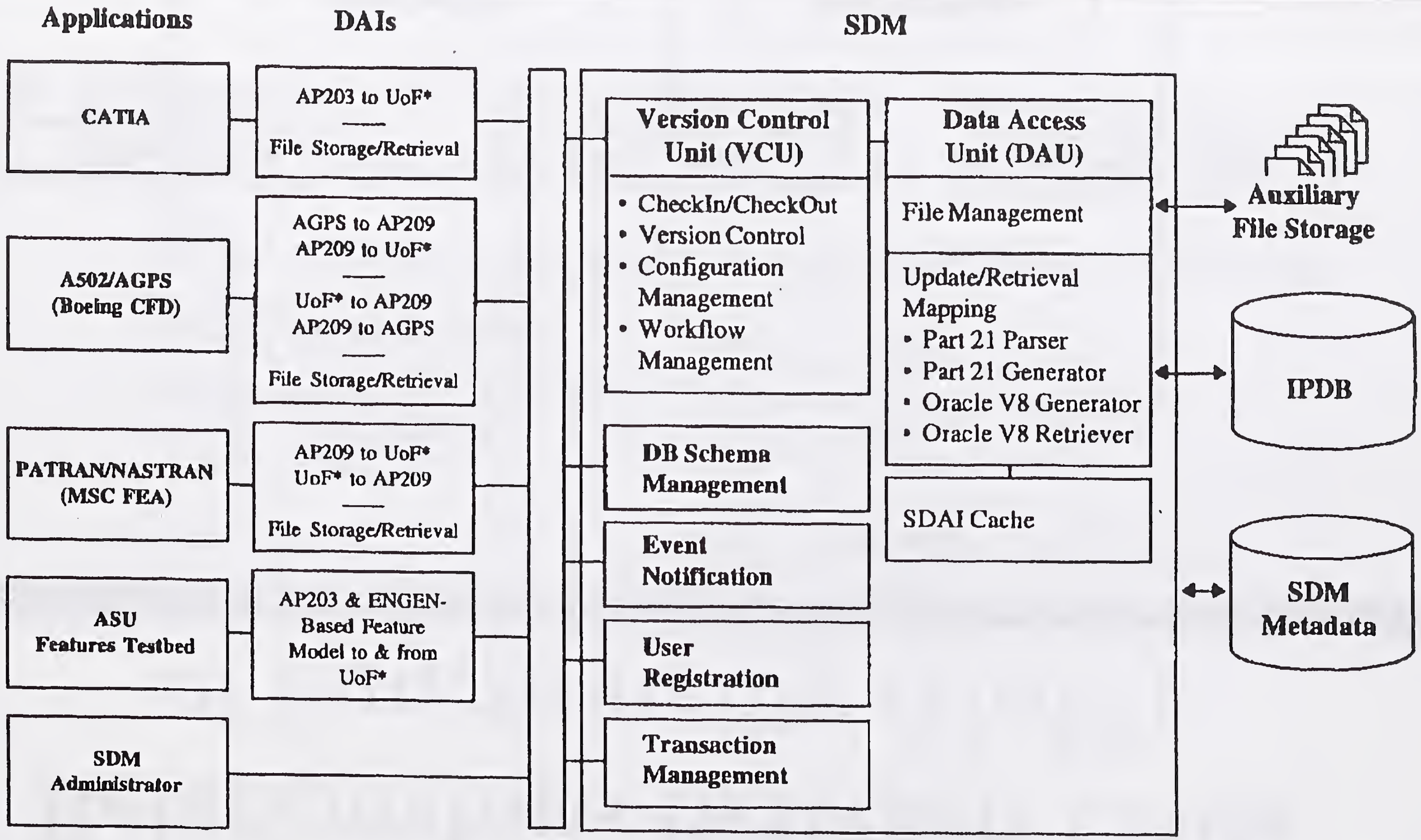
Objectives

- Integration of Technical Data: Develop a Common Information Model to Enable:
  - » STEP-Based Database Implementation of Design and Analysis Data
  - » Data Sharing Among Design and Analysis Disciplines
  - » Configuration Management of “In-Work” Data Throughout the Design/Analysis Cycle

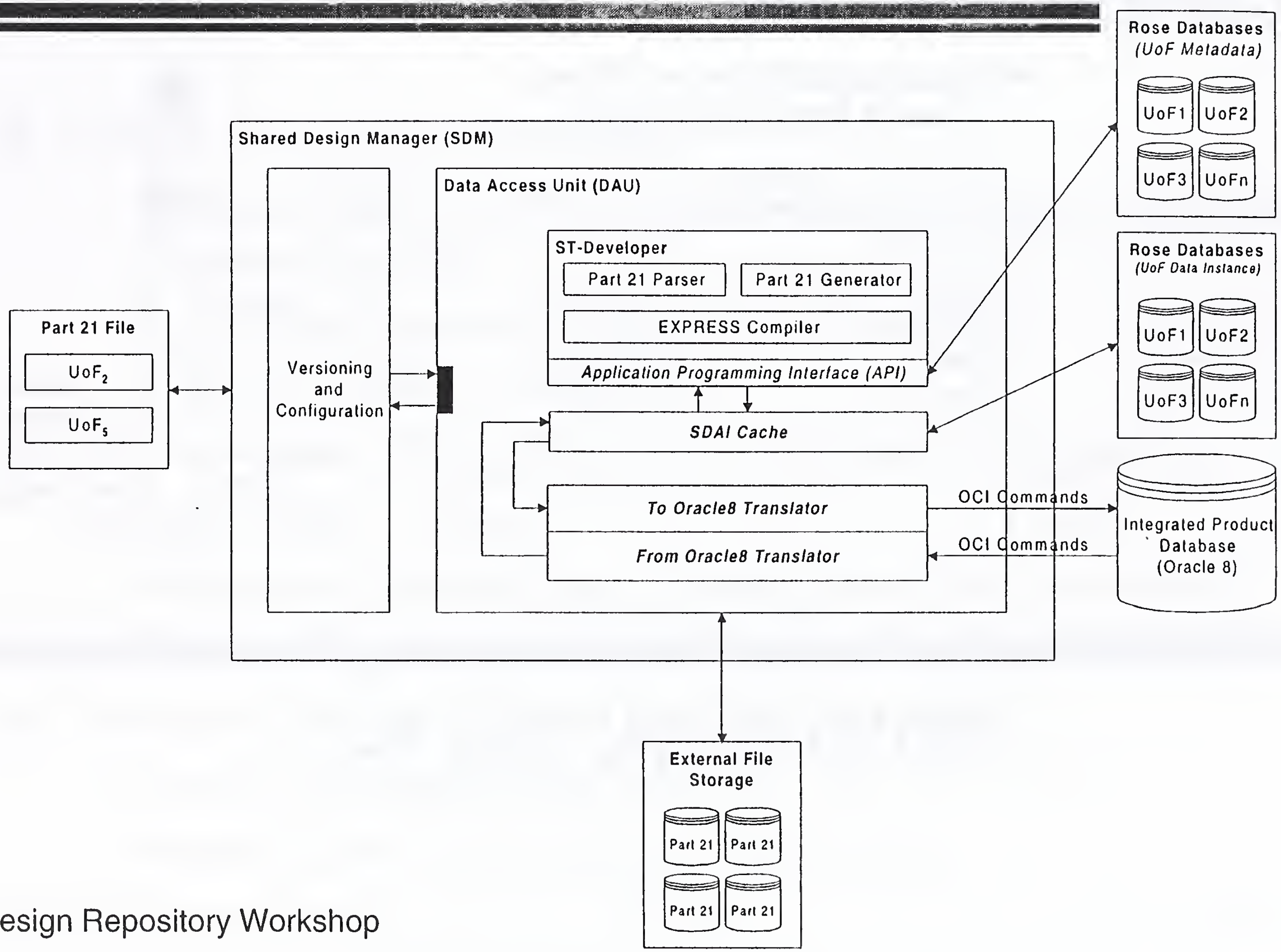
Relationships Between Units of Functionality (UoF*)



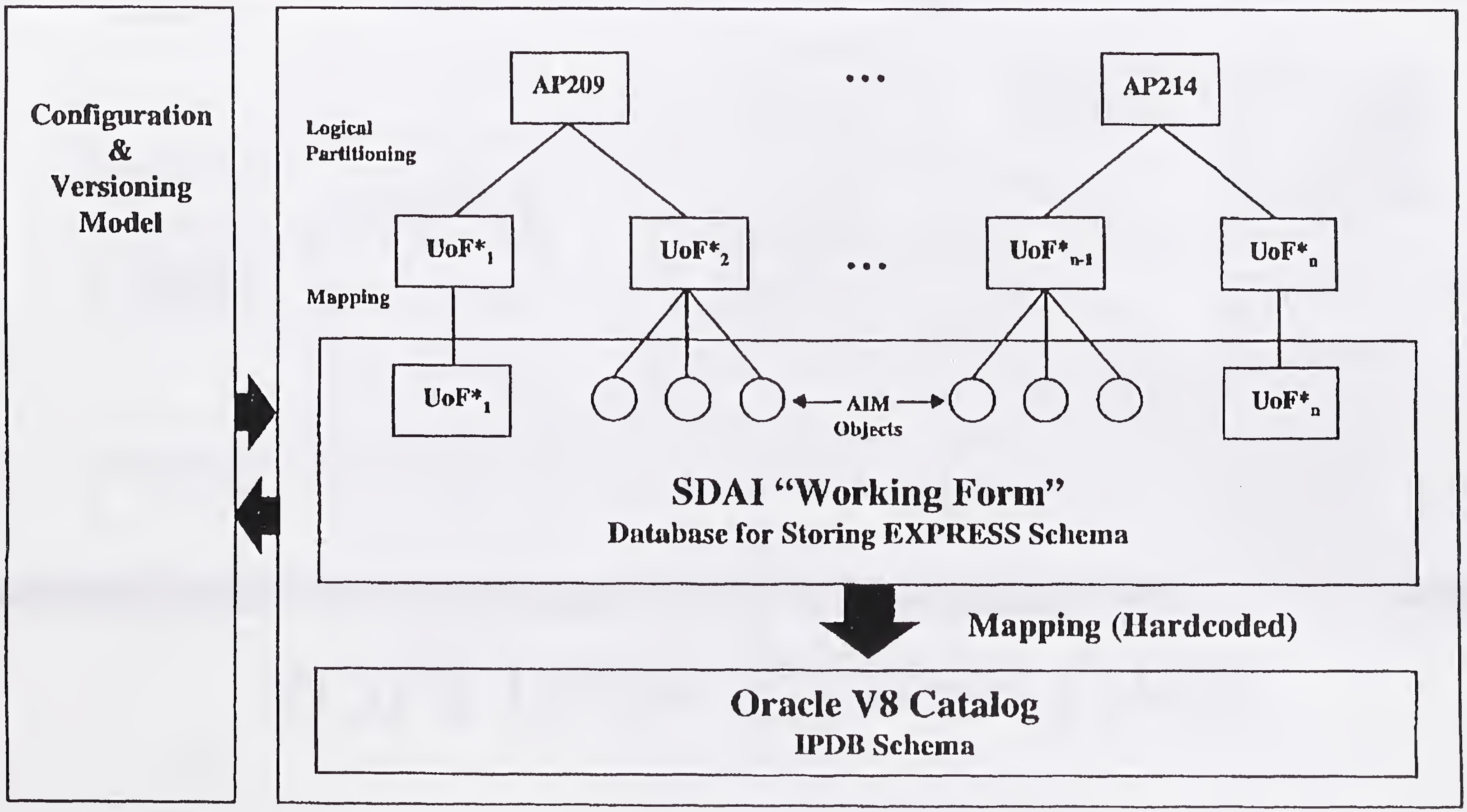
Shared Design Manager Architecture



SDM Data Access Unit



SDM Metadata



NIST Design Repository Workshop 11/15/96

SDM Technology Issues

- Define DAI/SDM Interface
  - » Manage UoF*s and Part 21 Files as the Primary Unit of Data Transfer for IPDB
  - » Coordinate Version/Configuration Data with Checkin/Checkout Operations
- Define Version/Configuration Model for Evolving Design Data
  - » Understand Industry (e.g., Boeing) Practices and Needs
  - » Maintain Control Without Stiffling the Process
- Perform Internal SDAI Caching of UoF*s
  - » Extract UoF*s from Part 21 Files
  - » Resolve UoF* Cross Domain References

Active Catalogs: A Designers Aid

Dr. Peter Will, University of Southern California (will@isi.edu)

The summary for this presentation can be found on page 37
17 slides follow

Active Catalogs: A Designers Aid

Peter Will
USC/ISI

NIST November 1996

Work performed under DARPA support

Outline

- Design Issues
- Current Catalog Technology
- Our Vision
- SBD Example
- Future Work
- Conclusion

Concurrent Design Activities

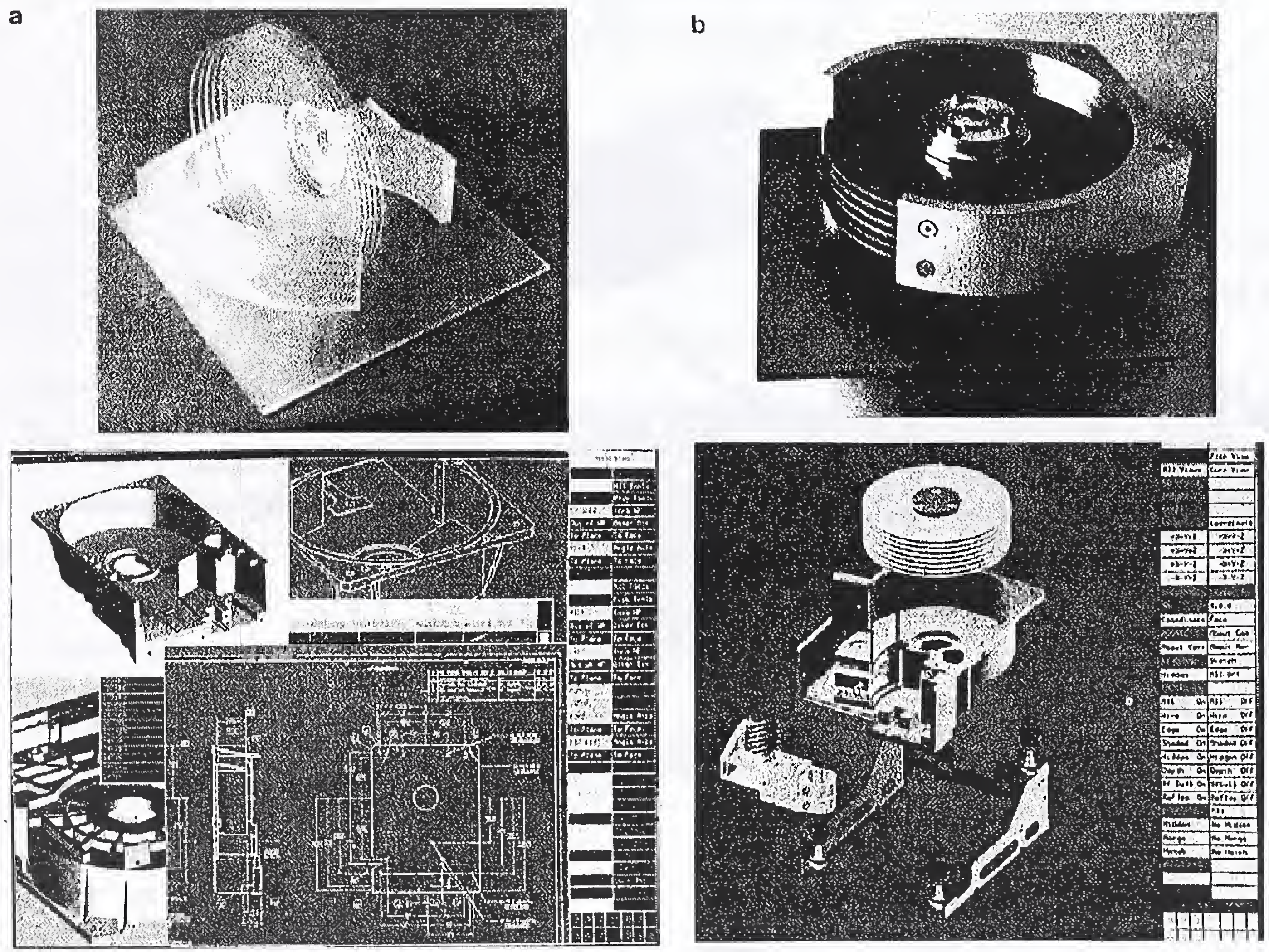


Fig. 7. (a) Disk drive models: top, product conception; bottom, detail layout of frame. (b) Disk drive models: top, final drive component; bottom, simulated exploded assembly.

No behavioral information !

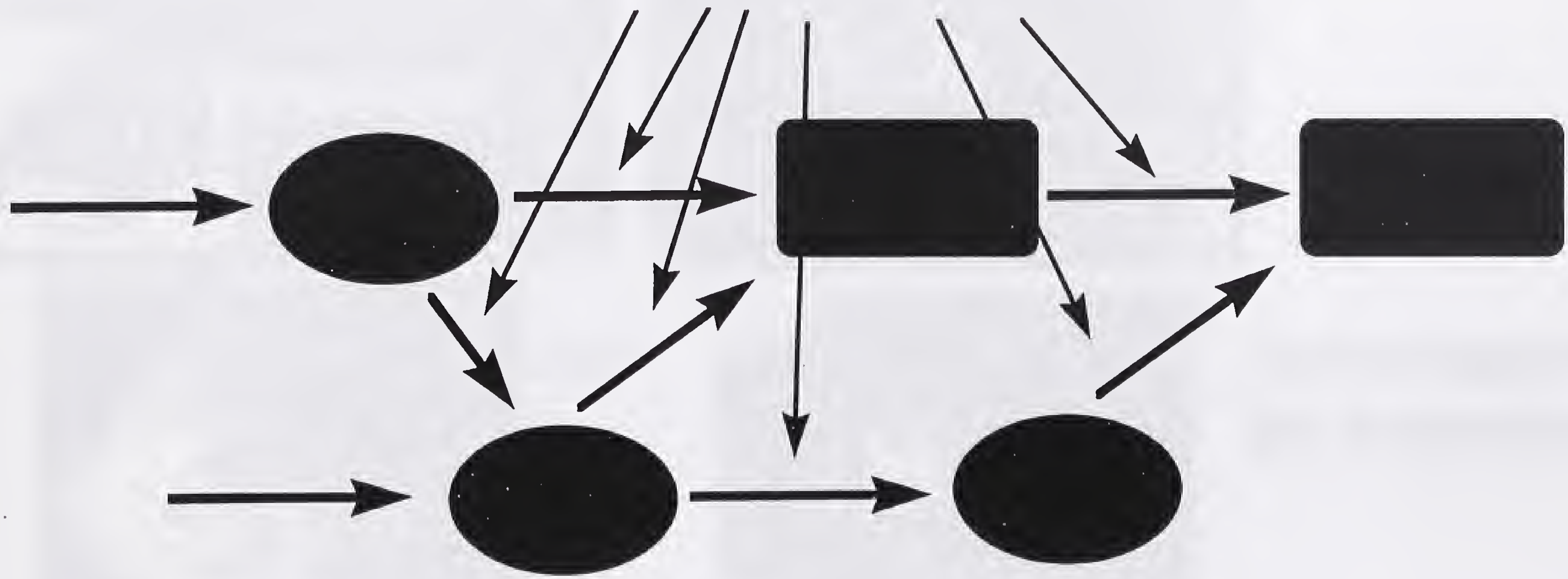
From Simulation and Modeling in Early Concept Design: An Industrial Perspective by P. Will
Res. Eng. Des. (1991) 3:1-13

Active Catalogs, KIC, Sept. 96

P. Luo, USC/ISI

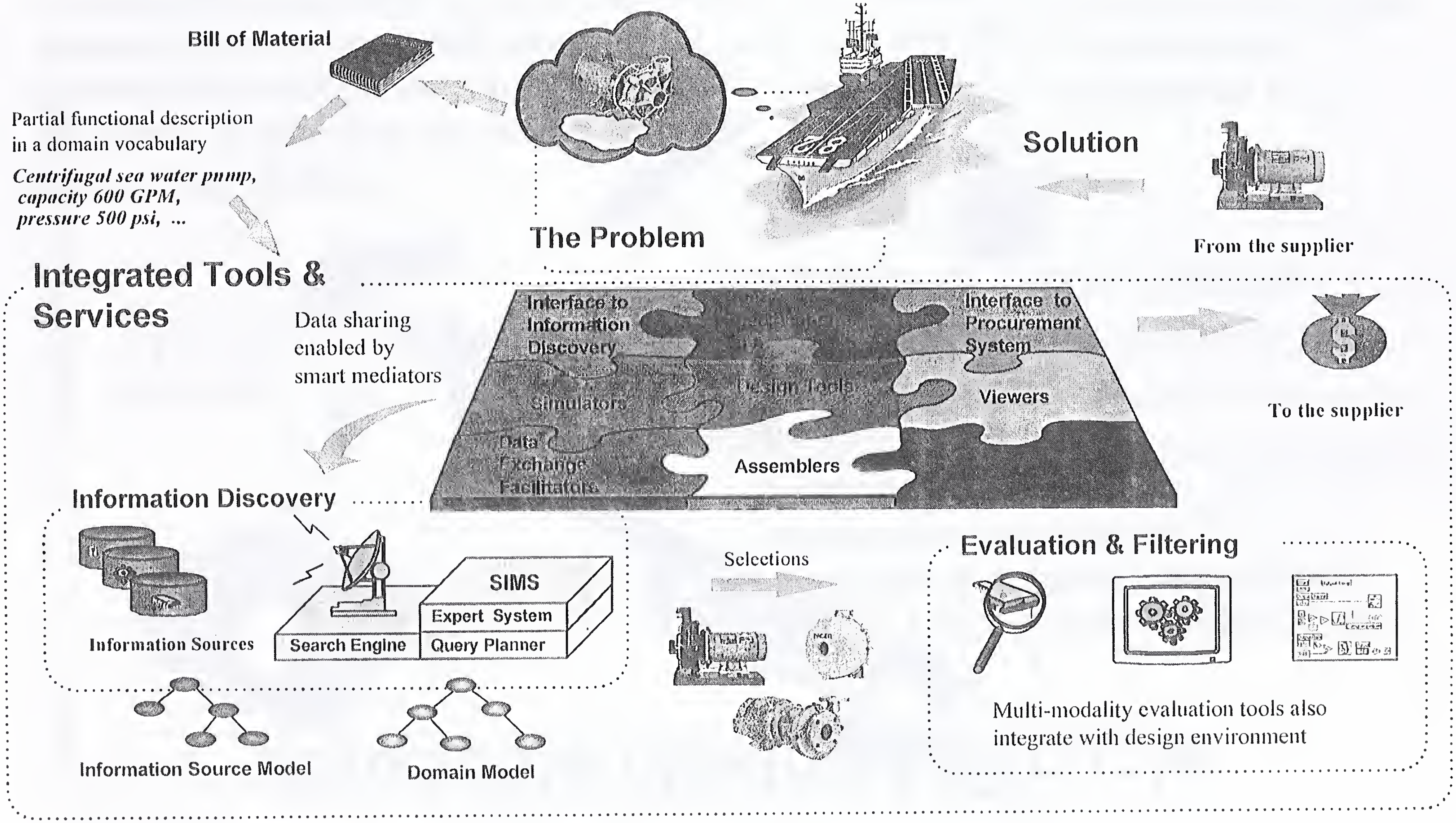
Reuse

Component Interactions

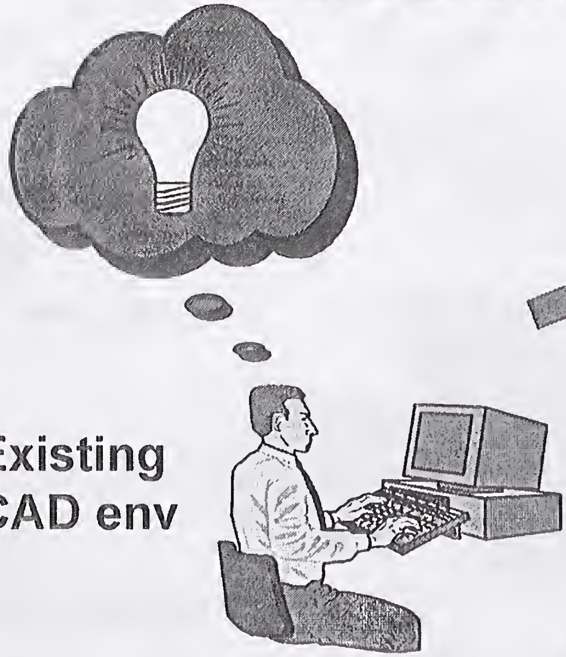


Key to Reuse:
- Rich modalities of models
- Easy access
- Seamless information flow

Active Catalogs Scenario



Reusable Design Lifecycle



Existing
CAD env

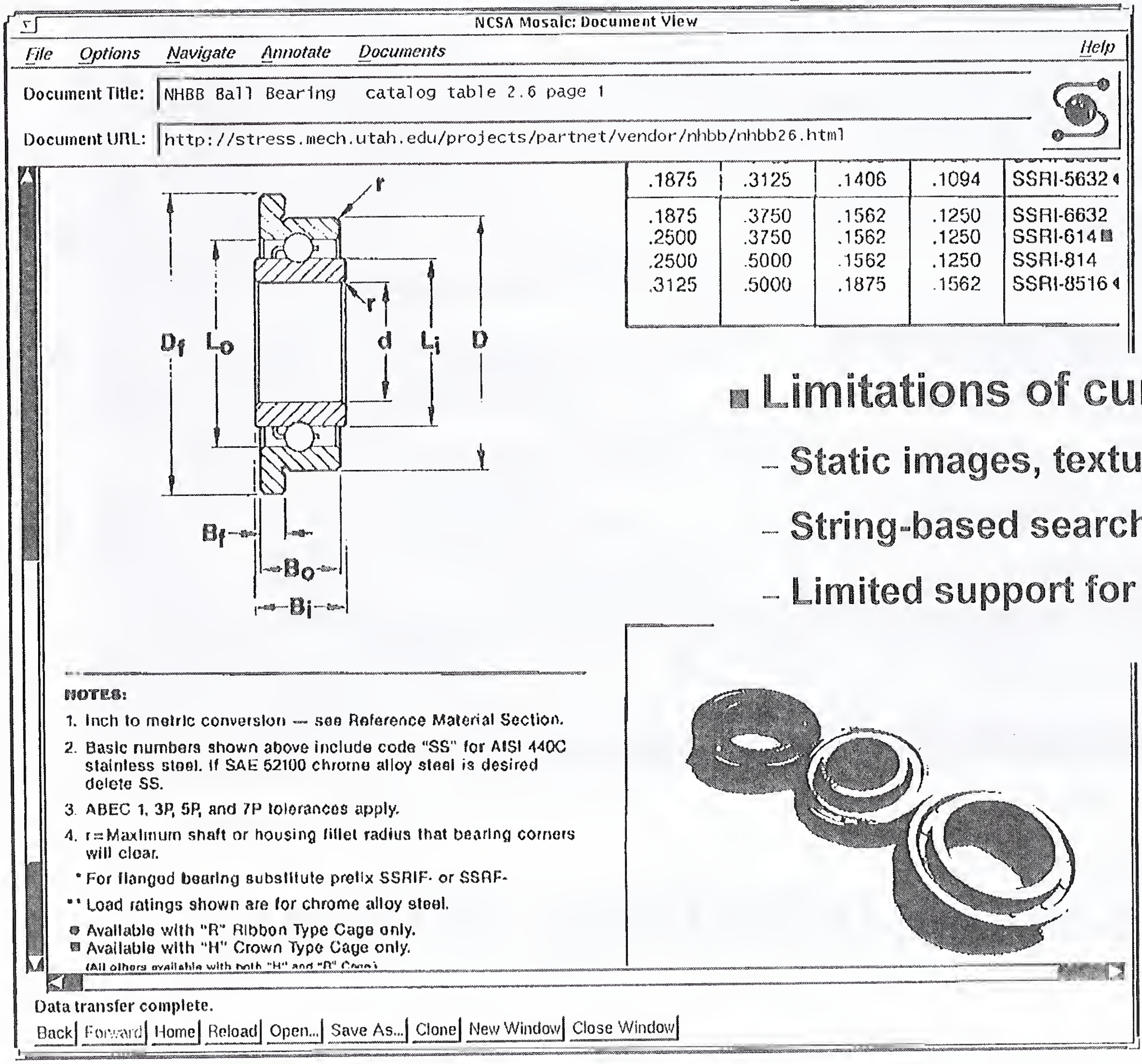
Catalog Tools & helper applications:
- Ontology & semantic description
- Behavioral information
- Input & output specs
- ...

Repository of
components
& systems for reuse

Helper applications:
- Ontology & semantic navigator
- Knowledge-based info retrieval engine
- Viewers for rich modality information
- Knowledge-facilitated consumption env
- ...

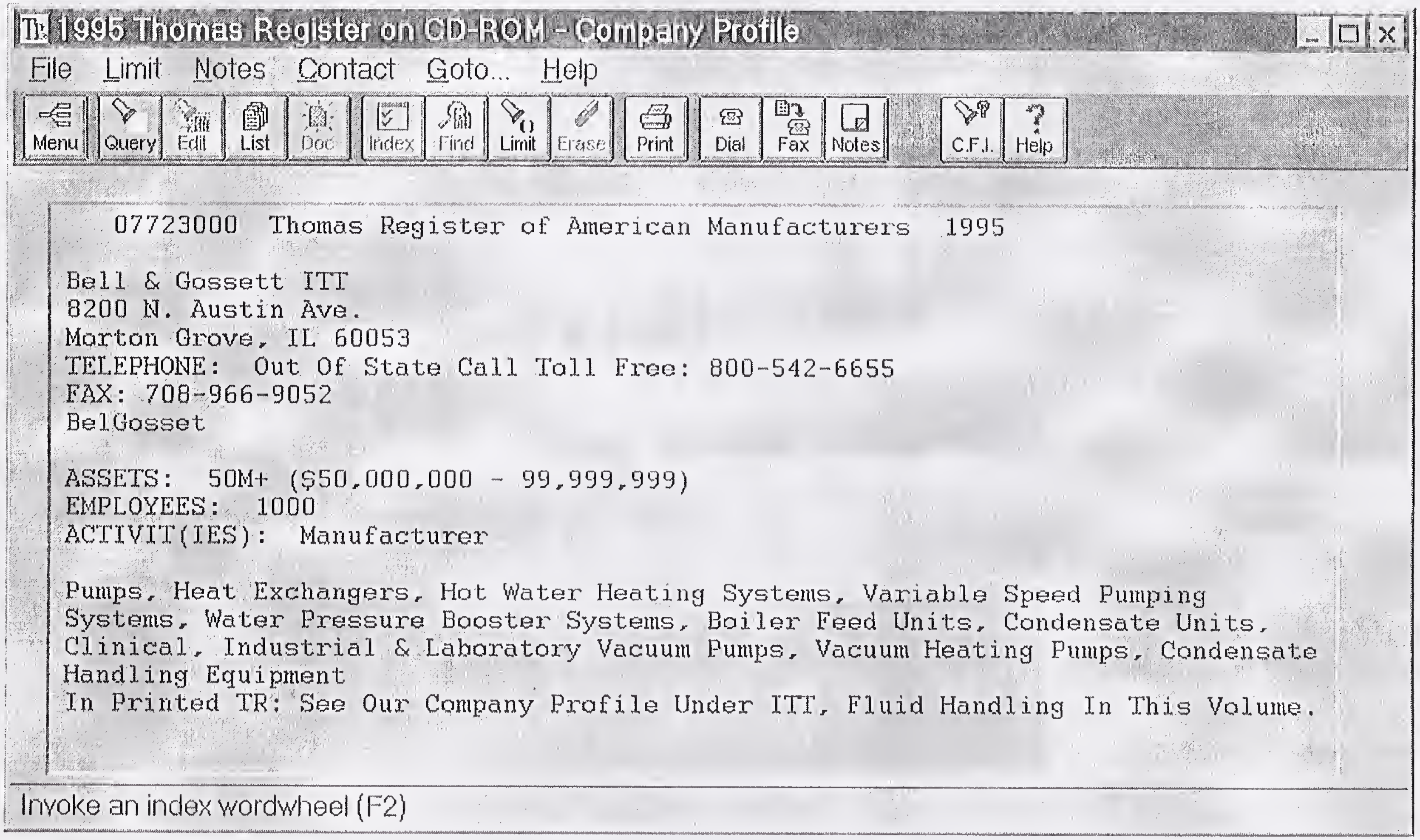
Knowledge about design, components, & information consumption is critical for reuse of designs & components

On-Line Catalogs: Using Computer to Imitate Paper



- Limitations of current catalogs:
  - Static images, textual descriptions
  - String-based search on textual information
  - Limited support for information consumption

Thomas Register Line Card



Cataloging Technology

State of the Art — Extended by — Active Catalogs

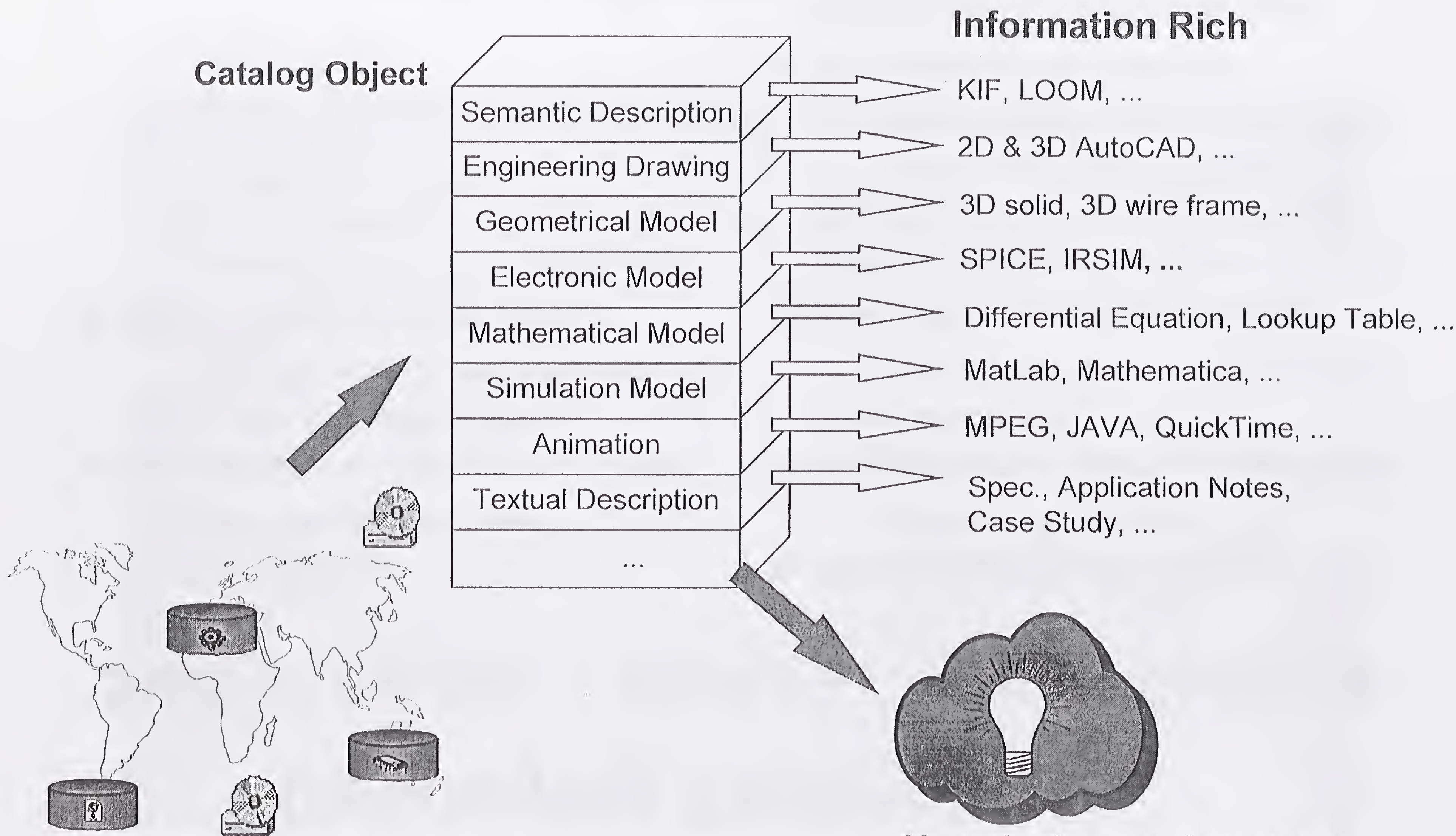
State of the Art

- Textual search
  - keywords, attribute names
- Information is mainly for human eyes and redistribution
  - Text, Bitmaps and MPEG Movies
- Taxonomy for procurement
  - Not suitable for machine reasoning
  - Thomas Register
  - Oil business
  - Auto industry
  - NATO,
  - State of Texas, ...

Active Catalogs

- Knowledge-based search
  - problems, applications
- Information is readily usable and computational
  - Mathematical and simulation models
- Semantic network for domain experts of wide range of applications
  - Classification in great details
  - Can be reasoned upon by computers
- Knowledge-facilitated information consumption environment

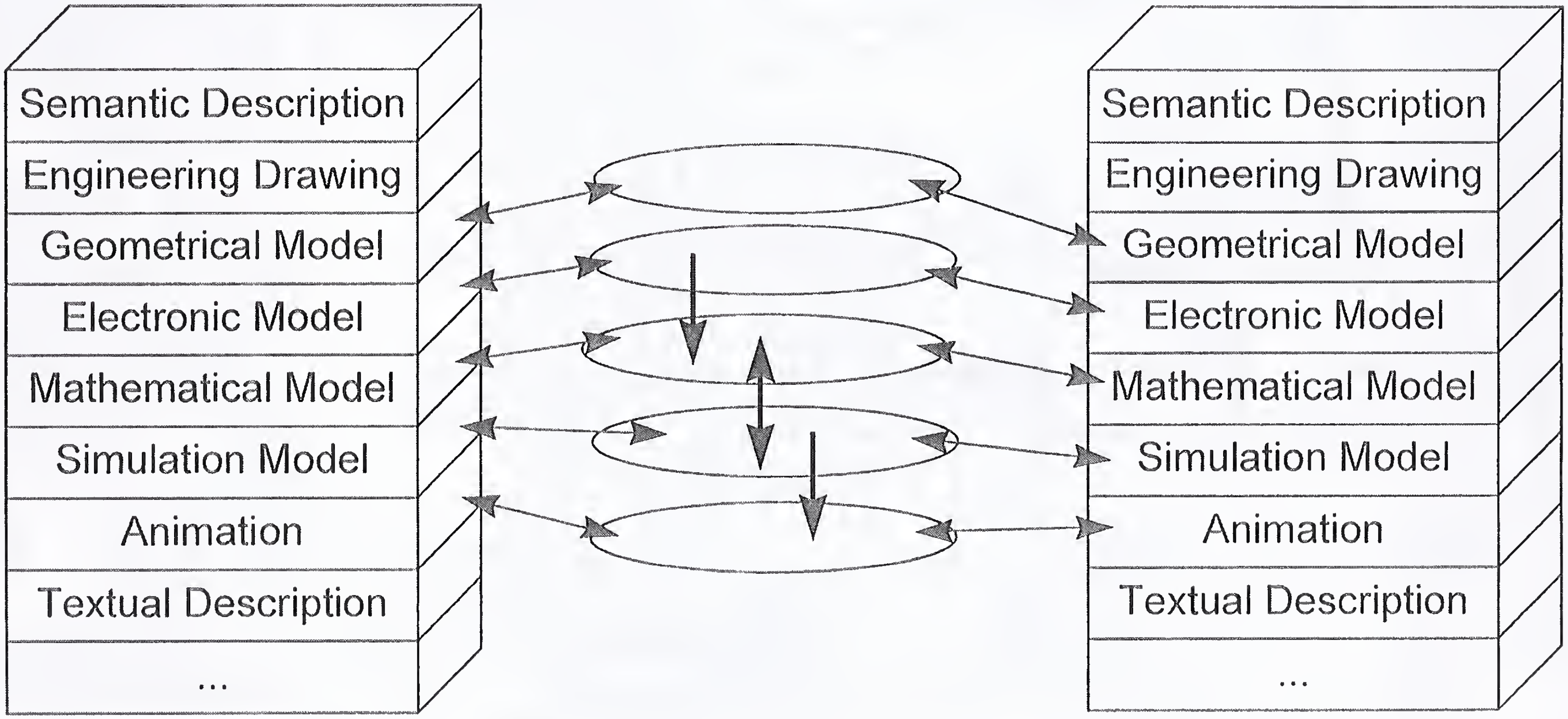
Objects in Active Catalogs



Distributed computational Objects

New design environment: "Try Before You Buy"

Active Catalogs' Contents Are Interoperable

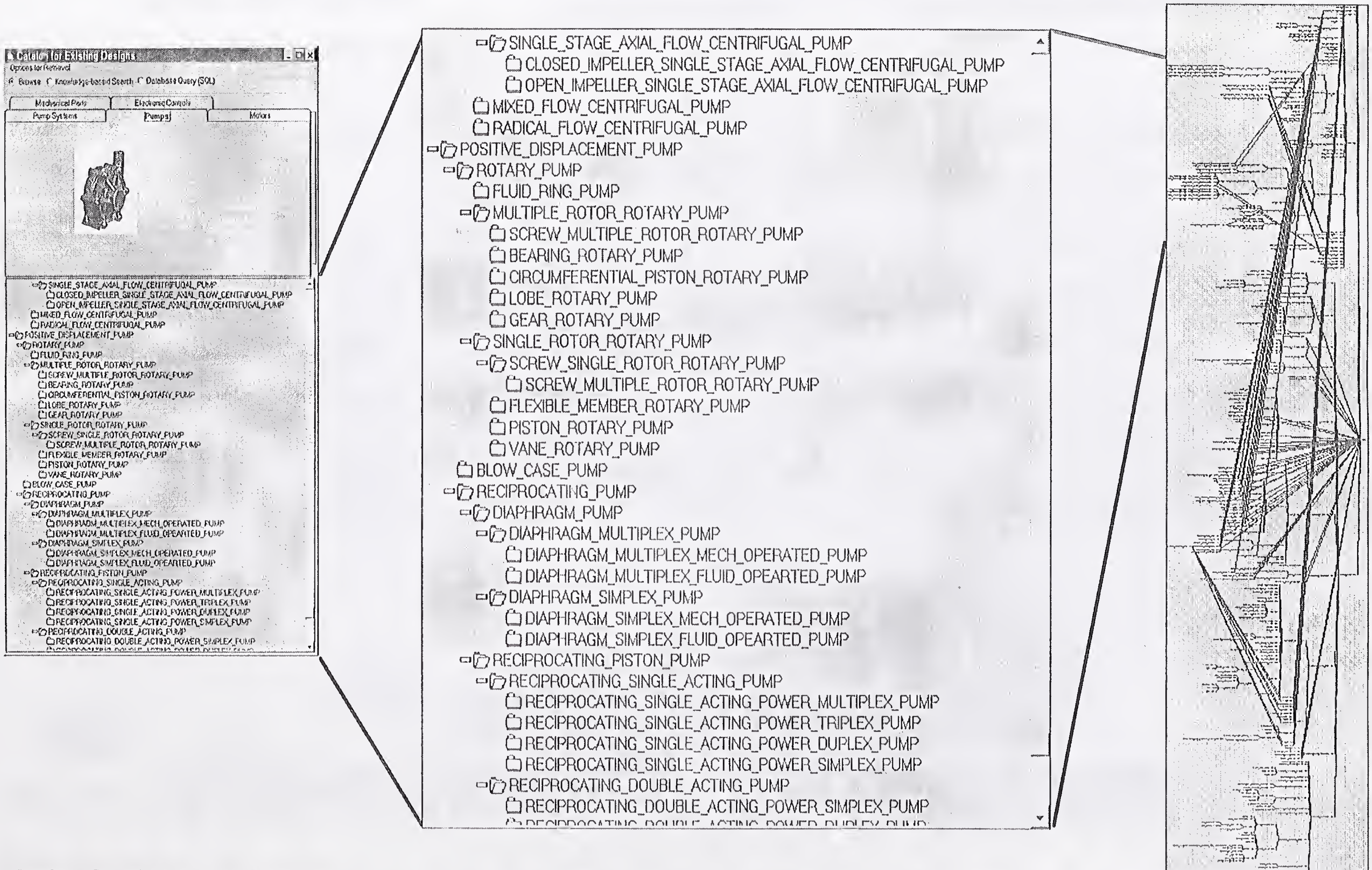


Pump Object Model

Interface wrappers governing data exchange

Motor Object Model

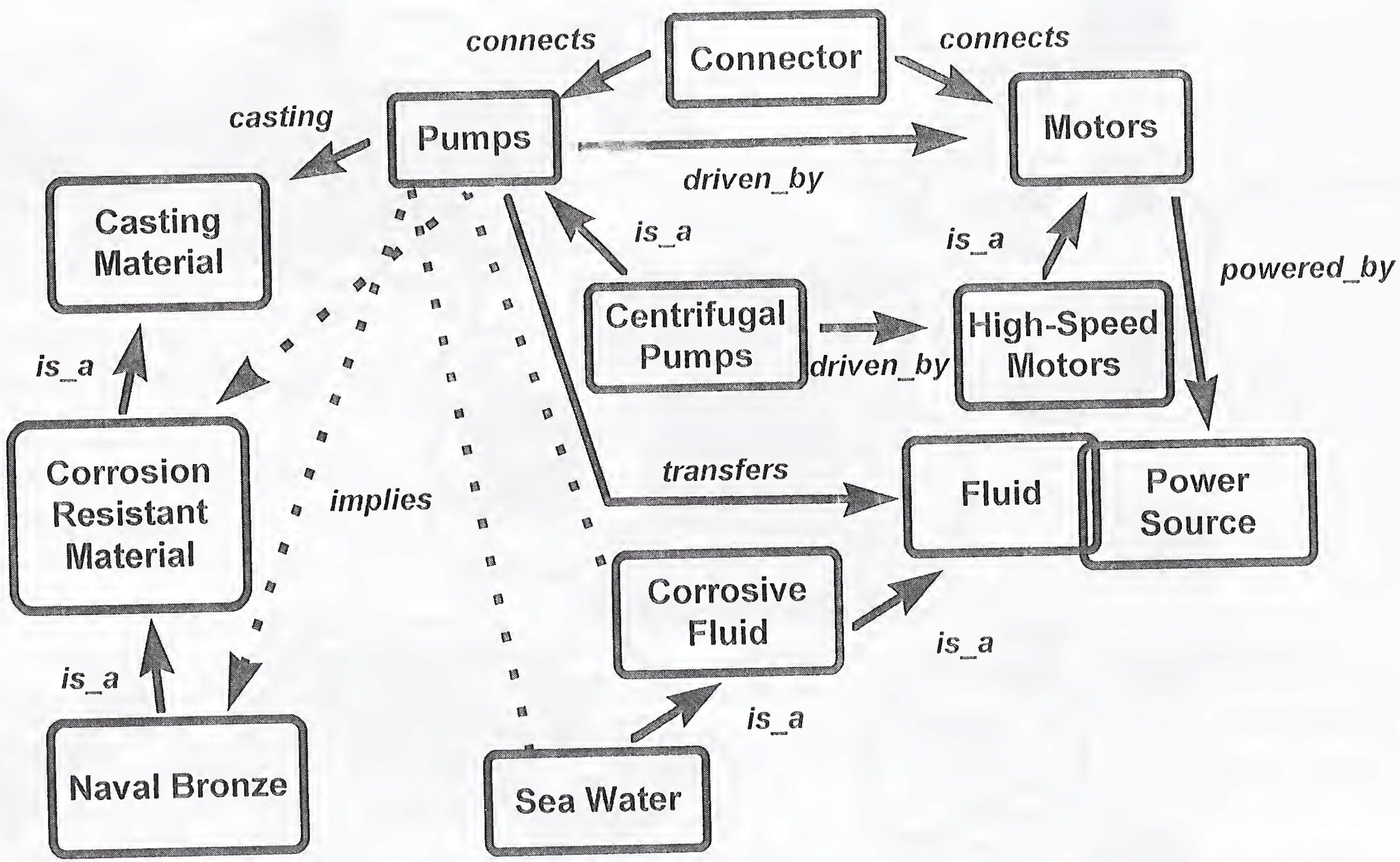
Active Catalogs' Domain Ontology



Active Catalogs, KIC, Sept. 96

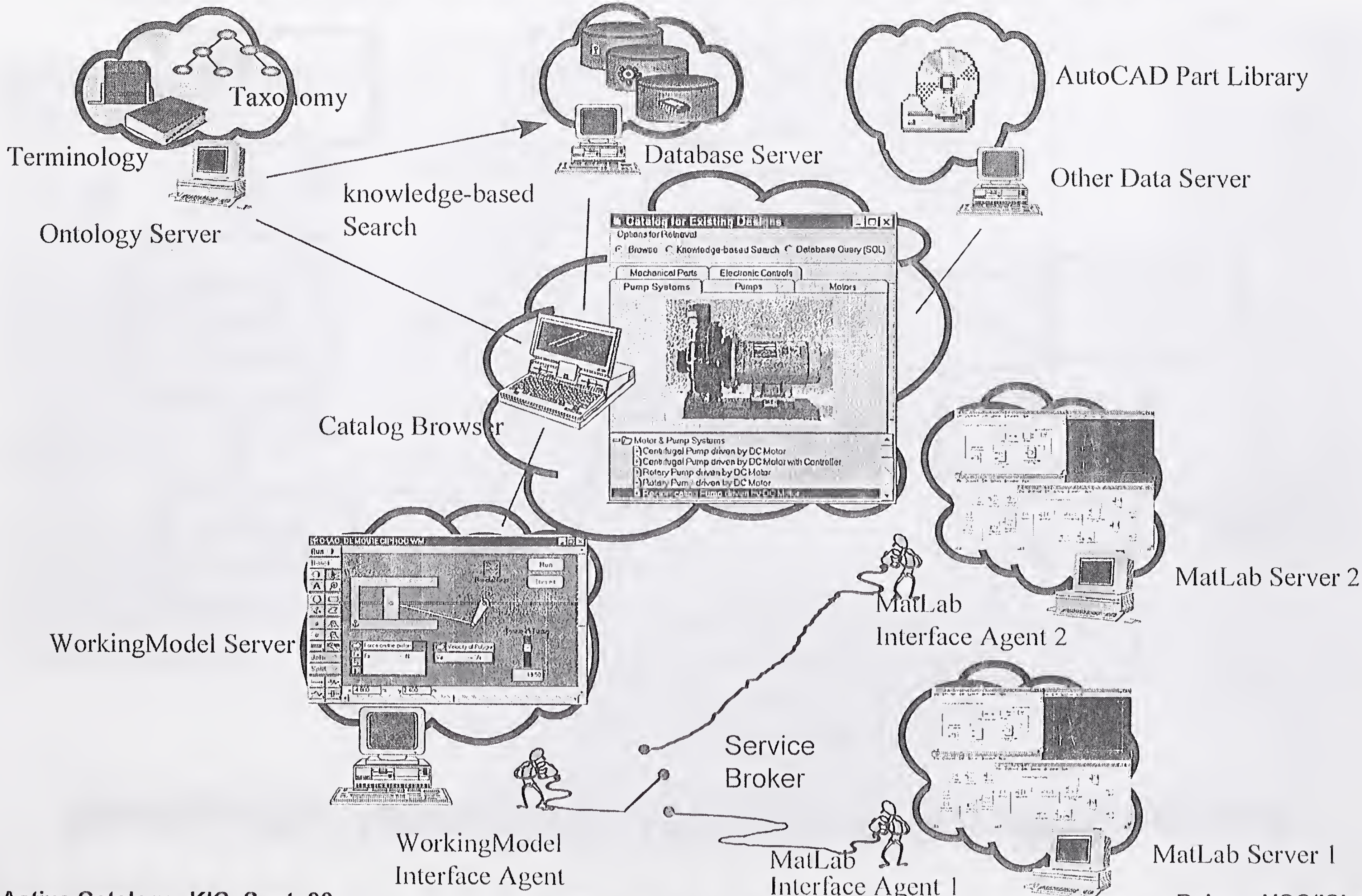
P. Luo, USC/ISI

Semantic Network: Usage & Environment

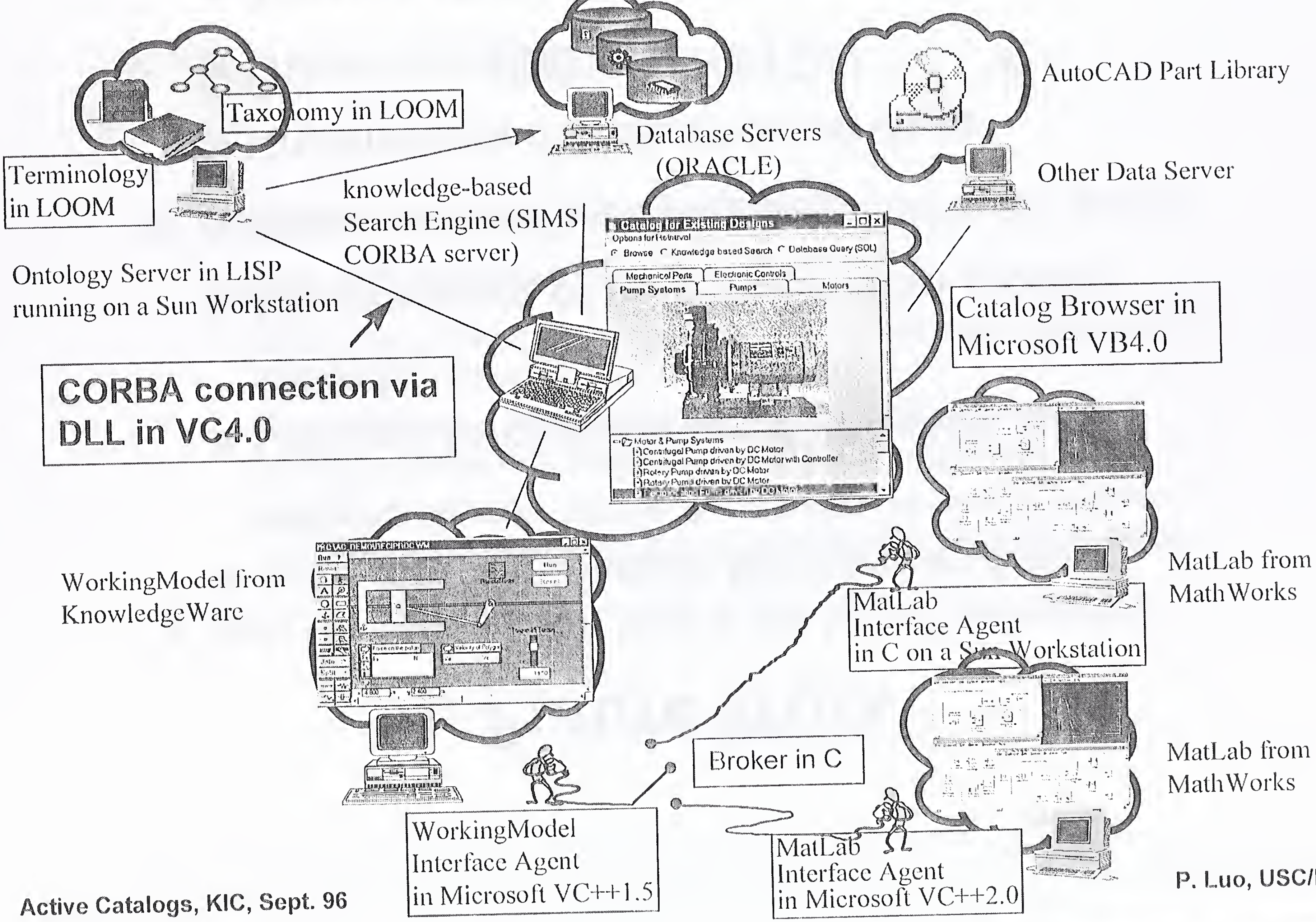


- Query under KB-facilitated search: pump capable of transfer sea water
- Query under traditional DB search: pump with bronze casting

Active Catalogs Architecture



Implementation Detail



P. Luo, USC/ISI

Future Work

- Java implementation for front-end and middle-ware
- Linkage to commercial databases and models
- Management of time scales and clocks
- Other modalities of models
- Sophisticated broker for services
- Other domains and
- Tons of other things to do

Conclusion

- Critical to reuse of existing designs are three types of knowledge
  - Describing designs & applications of parts & components (semantic network, theory of design)
  - Enabling ease of consumption of existing reusable designs (behavior models and I/O descriptions)
  - Facilitating consumption of reusable designs (runtime environment that provides a seamless information flow)
- Many open issues and a lot yet to be done!

Engineering Design-Related Database Research at Georgia Tech

Dr. Sham Navathe, Georgia Institute of Technology (sham@cc.gatech.edu)

The summary for this presentation can be found on page 38
21 slides follow
A paper on this topic begins after page 384

Engineering Design related Database Research at Georgia Tech

Prof Shamkant B. Navathe
College of Computing,
Georgia Institute of Technology,
Atlanta, GA 30332-0280.
(404) 894 0537
sham@cc.gatech.edu

Home of the 1996 Olympic Village
Georgia Tech
Database Group

HIPED
Heterogeneous Intelligent Processing for Engineering Design

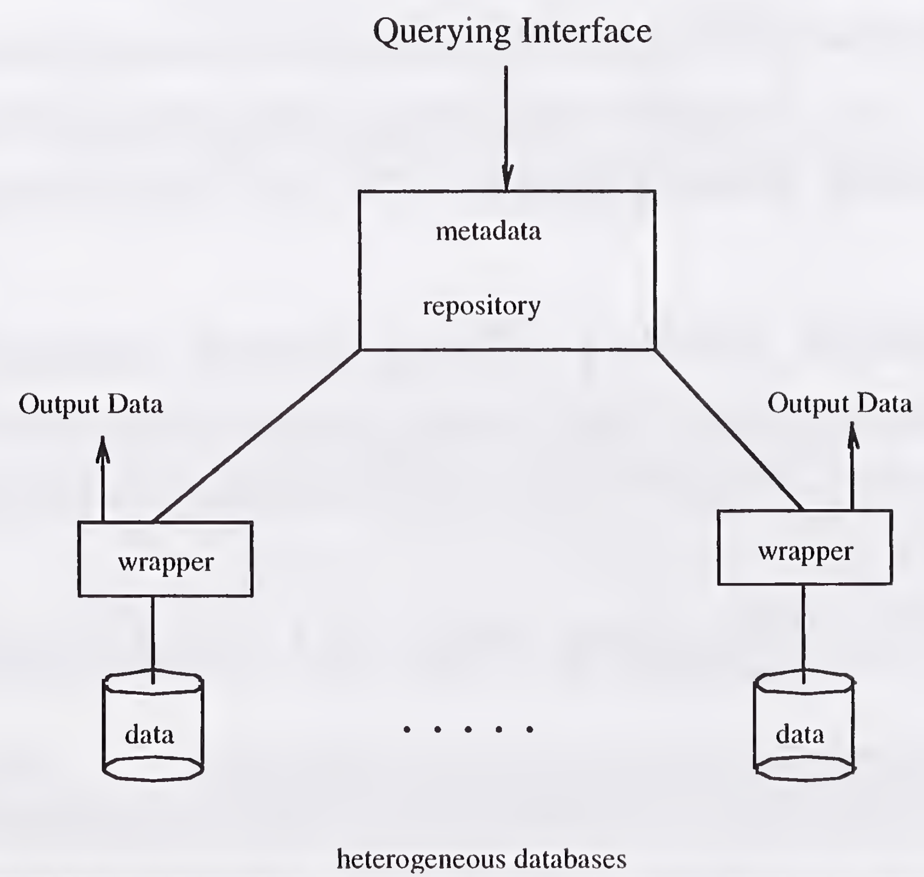
P.I. : SHAM NAVATHE
Co-P.I.s : ED OMIECINSKI
ASHOK GOEL

Highlights of the HIPED project

- intelligent querying interface to a set of information sources (like structured databases, knowledge bases, unstructured data)
- combination of an intelligent front-end for engineering design and a heterogeneous database back-end
- information retrieval through common interface
- project supported by **DARPA** (in the I3-"intelligent integration of information" program).

Home of the 1996 Olympic Village
Georgia Tech Database Group

The System Overview



Querying Interface

- (Prototype <proto_name>)
- (Property <prop_name>)
- (Prototype <proto_name>)
  (Property <prop_name>)
- (Prototype <proto_name>)
  (Property <prop_name>)
  (Value <value>) (Rel-op <op>)
- (Property <prop_name>)
  (Value <value>) (Rel-op <op>)

Home of the 1996 Olympic Village
Georgia Tech Database Group

Query Examples

- (Prototype battery)
- (Property voltage)
- (Prototype battery) (Property voltage)
- (Prototype battery) (Property voltage)
  (Value 10) (Rel-op ==)
- (Property voltage) (Value 24) (Rel-op >)

Heterogeneous Backend : Database Integration using Rules

- **Use of two types of RULES**
  - **Database correspondence Rules**
  - **Query mapping Rules**

- **Use of CORAL deductive DBMS for Rule Processing to generate appropriate queries for the DBMS wrappers**

Home of the 1996 Olympic Village
Georgia Tech Database Group

Rules for Interpretation of Queries

- **(Prototype Battery) [Type 1]**
  - » **Prototype Tables having Battery, Property and Value columns**
  - » **Battery Tables having Property and Value columns**
  - » **Tables having Prototype, Property and Value columns (check for Prototype == Battery)**
- **(Property Voltage) [Type 2]**
  - » **Property Tables having Voltage and Prototype columns**
  - » **Voltage Tables with Prototype columns**
  - » **Tables having Property and Prototype columns (check for Property == Voltage)**

- (Prototype Battery)(Property Voltage) [Type 3]
  - » Prototype Tables having
    - Battery, Property and Value columns (check for Property == Voltage)
    - Battery and Voltage columns
  - » Battery Tables having
    - Property and Value columns (check for Property == Voltage)
    - Voltage column
  - » Property Tables having
    - Voltage, Prototype and Value columns (check for Prototype == Battery)
    - Voltage and Battery columns

- Voltage Tables having
  - Prototype and Value columns (check for Prototype == Battery)
  - Battery and Value columns
- Tables having Prototype, Property and Value columns (check for Prototype == Battery AND Property == Voltage)

- **(Prototype Battery)(Property Voltage) (Value 10) (Rel-op ==) [Type 4]**
  - Prototype Tables having
    - Battery, Property and Value columns (check for Property == Voltage AND Value == 10)
    - Battery and Voltage columns (check for Voltage = = 10)

Rules for Interpretation of Queries (contd.)

- Battery Tables having
  - Property and Value columns (check for Property == Voltage AND Value == 10)
  - Voltage column (check for Voltage == 10)
- Property Tables having
  - Voltage, Prototype and Value columns (check for Prototype == Battery AND Value == 10)
  - Voltage and Battery columns (check for Voltage == 10)
- Voltage Tables having
  - Prototype and Value columns (check for Prototype == Battery AND Value == 10)
  - Battery and Value columns (check for Value == 10)

Home of the 1996 Olympic Village
Georgia Tech | Database Group

Rules for Interpretation of Queries (contd.)

- **(Property Voltage) (Value 10) (Rel-op ==) [Type 5]**
  - » **Property Tables having Voltage and Prototype columns (check for Voltage == 10)**
  - » **Voltage Tables having Value and Prototype columns (check for Value == 10)**
  - » **tables having Property, Value and Prototype columns (check for Property == Voltage AND Value = = 10)**

Home of the 1996 Olympic Village
Georgia Tech | Database Group

Query Mapping Algorithm

- It maps the user queries to the appropriate set of subqueries against wrappers of backend databases.
- It consults the facts and rules represented in CORAL and deduces the applicable queries.
- Equivalence of attribute and table names within and across databases is automatically considered.

An Example

db1

Components Table

CompNo	Prototype	Property	Value

Battery Table

bNo	Voltage

db2

Supplier Table

BatteryNo	SupplierNo

Battery Table

bNo	Current

CORAL Facts and Rules

- belongsTo(components, db1).
- belongsTo(battery, db1).
  - hasAttribs(db1, components, [compNo, prototype, property, value]).
  - hasAttribs(db1, battery, [bNo, voltage]).
- belongsTo(battery, db2).
- belongsTo(supplier, db2).
  - hasAttribs(db2, battery, [bNo, current]).
  - hasAttribs(db2, supplier, [batteryNo, supplierNo]).
- isAttrib(Db, Table, Attri) :-

      hasAttribs(Db, Table, Attribs),
      iselem (Attri, Attribs).

Example Query

- (Prototype battery) (Property current) the user wants to see all the prototypes identified as "Battery" with a special interest in their "Current" Property. It is implicit that he is interested in viewing their respective values.

- This is a query of Type 3 according to our scheme. Thus we would apply the corresponding rules to check for the relevant tables.

Result for the example

- The example query can be run against Components table in db1 and Battery table in db2.
- The corresponding SQL queries,

  for Components table in db1 :

  ```
  SELECT *
  FROM components
  WHERE prototype == battery AND property == current
  ```

  for Battery table in db2:

  ```
  SELECT *
  FROM battery
  ```

Conclusions and Future Work

- Approach illustrates use of facts and rules in CORAL, a deductive DBMS, to achieve integration of heterogeneous databases.
- The five generic types of queries can be further modified and new types can be added to cater to a wide range of design problems.
- The generic query types were derived from Engineering Design perspective but can be extended to any application domain.

Future Work (contd.)

- **Current implementation deals only with equivalence among attribute and table names. It can be extened to other types of relations.**
- **The database back ends can be extended to non -SQL types (other data models).**
- **Multiple tables at a time can be considered for satisfying a given query by incorporating joins within and across databases.**

Previous Work

- 1989 : Two PhDs
  - » Rafi Ahmed : Versioning Specification and Version anagement in CAD Databases
    - OO Model of CAD Objects
    - Restricting version proliferation
    - Efficient Storage
    - Notification Schemes
  - » Aloysius Cornelio : Structure Function Paradigm for modeling and simulation of Physical Systems.
    - Engineering Systems
    - Complexity of function, structure
    - Application to simulation
    - Proposal for cataloging

Additional Relevant Work

- **Aravindan Veerasamy (PhD 1997)**
  - » **Visualization as an aid to improve retrieval performance from large document databases**
    - **USER - SYSTEM Interaction**
    - **Free Form Querying**
    - **Extensive User Performance Studies**
- **Magdi Morsi (PhD 1992)**
  - » **An OO Database System for schema evolution**
    - **Developed a flexible OODBMS**
    - **Evolution of classes automatically impacts evolution of instances**

Rapid Design Exploration and Optimization (RaDEO)

Mr. Kevin Lyons, DARPA (klyons@darpa.mil)

The summary for this presentation can be found on page 50
14 slides follow

Rapid Design Exploration and Optimization (RaDEO)

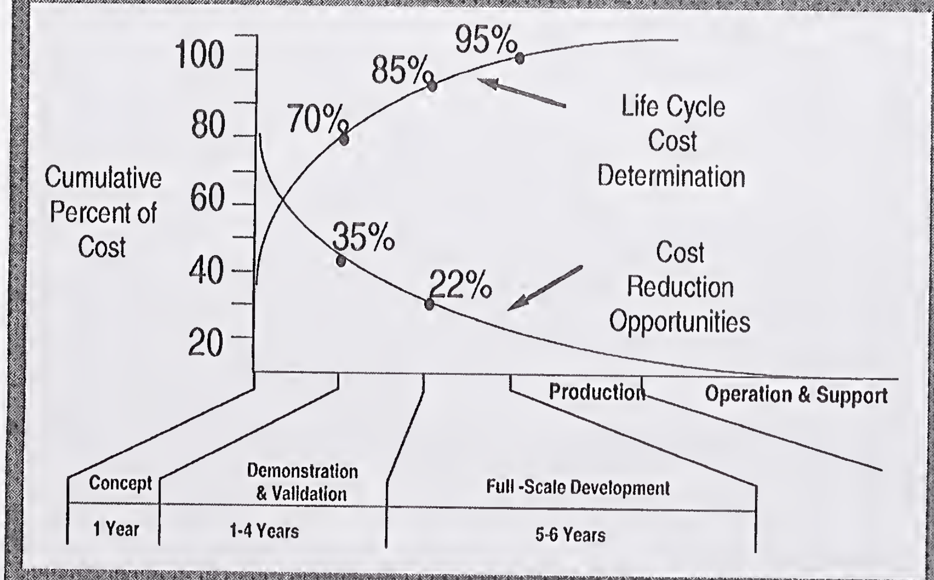
Kevin W. Lyons, DSO

NIST Large Scale Design Repository Workshop

November 20, 1996

http://radeo.nist.gov/radeo

Changing Marketplace for Defense Contractors



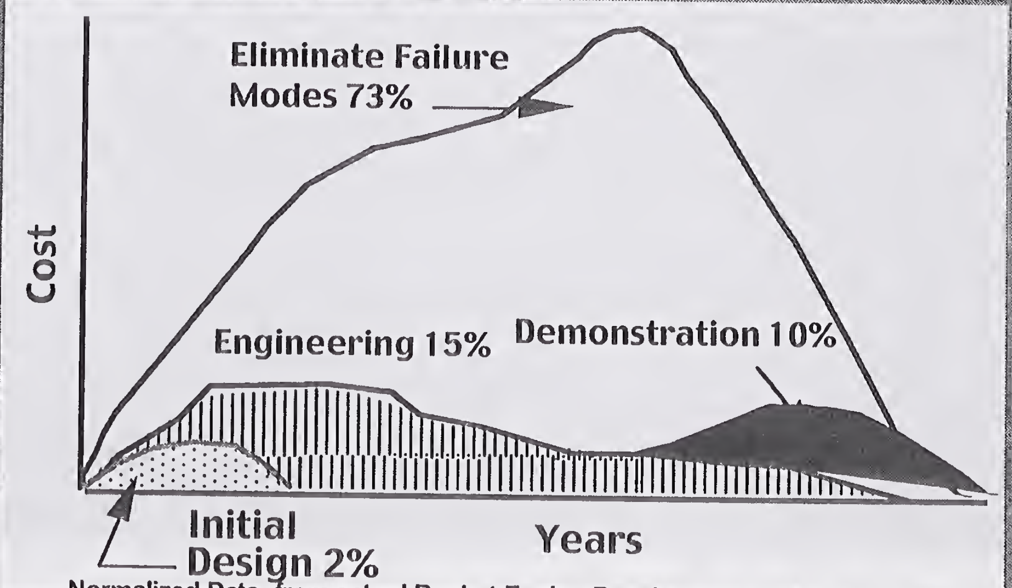
- Teaming encouraged or required to compete
- Supplier base becoming computer knowledgeable
- Mergers of major corporations
- Contracting changes
- Emphasis on total life cycle costs

The Problem ....

Inadequate tools for early product development trade-offs

Long cycle times to evaluate a design alternative

Integrated Product and Process teams are/will be distributed

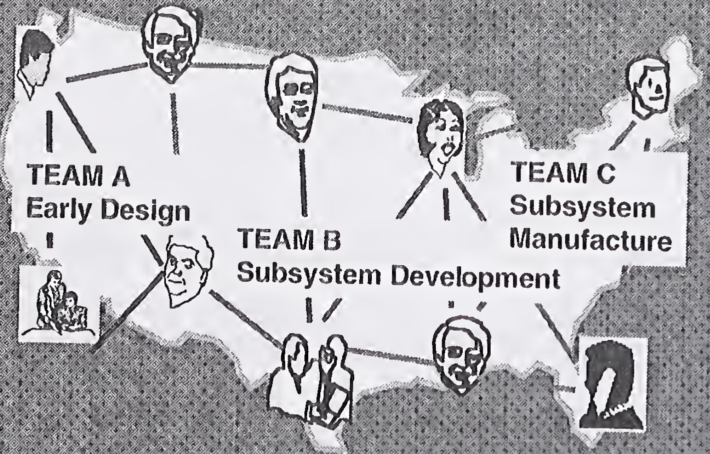


Normalized Data from actual Rocket Engine Development Project

Program Goal

The goal of the RaDEO program is to develop enabling technologies and tools to provide cognitive support to engineers by vastly improving their ability to explore, generate, track, store, and analyze design alternatives.

Technical Objectives

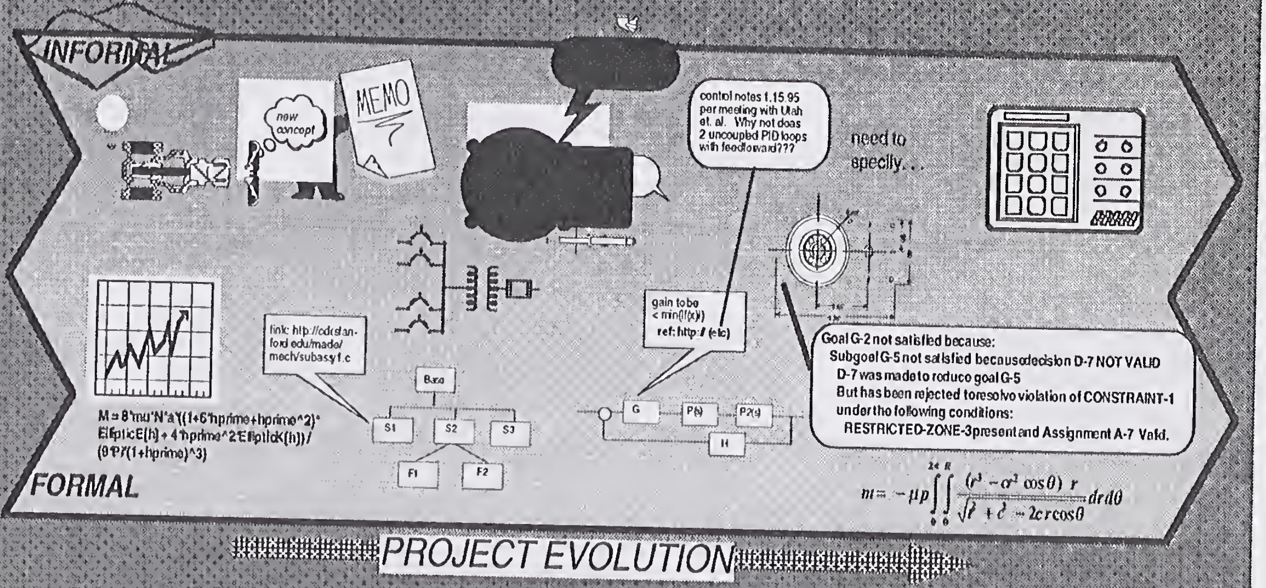


Design Web

Develop and Demonstrate enabling technologies and tools for an Information Infrastructure to support Personal, Organizational, and Artifact Webs for Collaborative Design

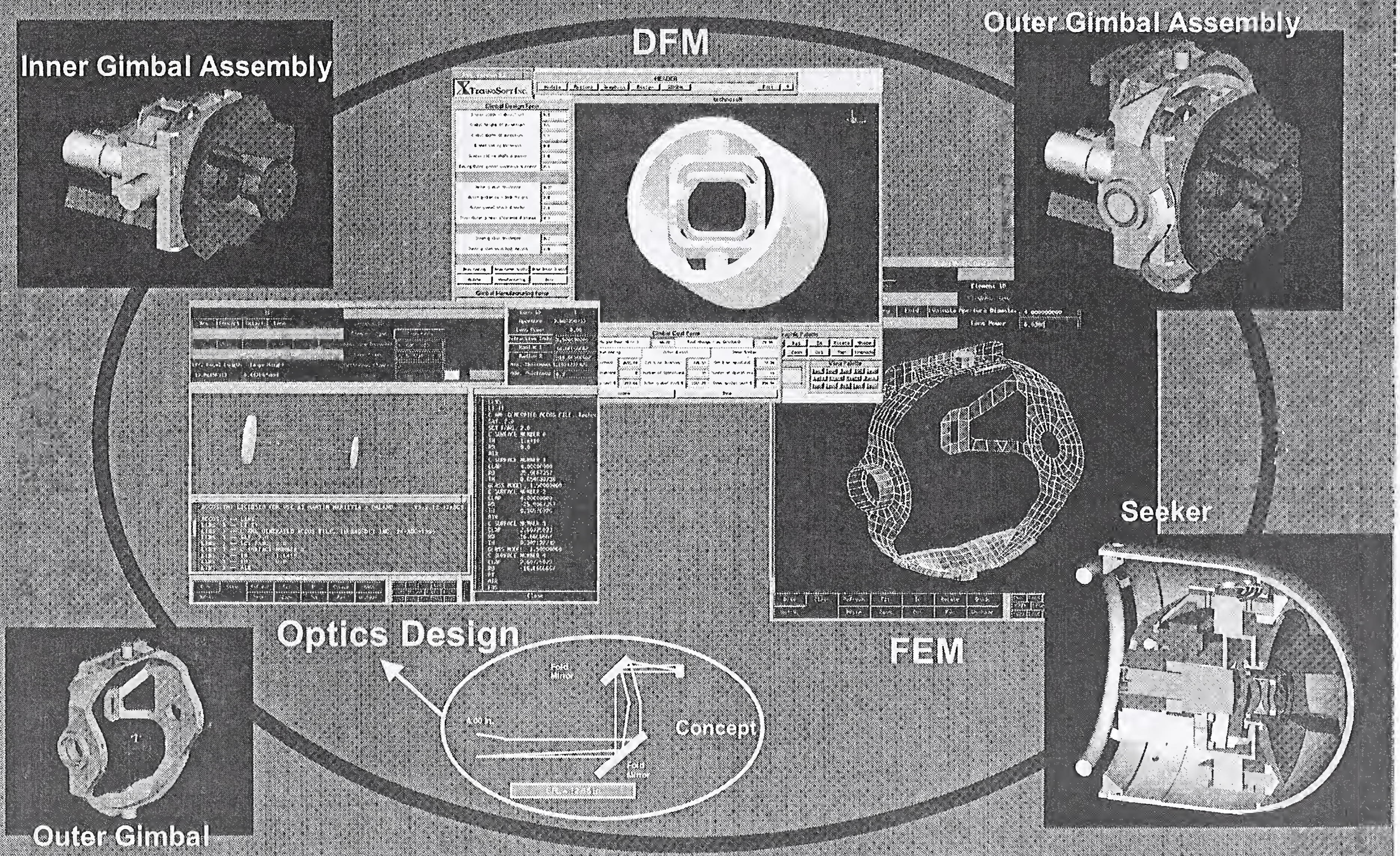
Comprehensive Design

Develop, Integrate, and Demonstrate enabling technologies and tools that significantly improve the capability to generate, store, retrieve, track, and analyze alternatives during all phases of a design

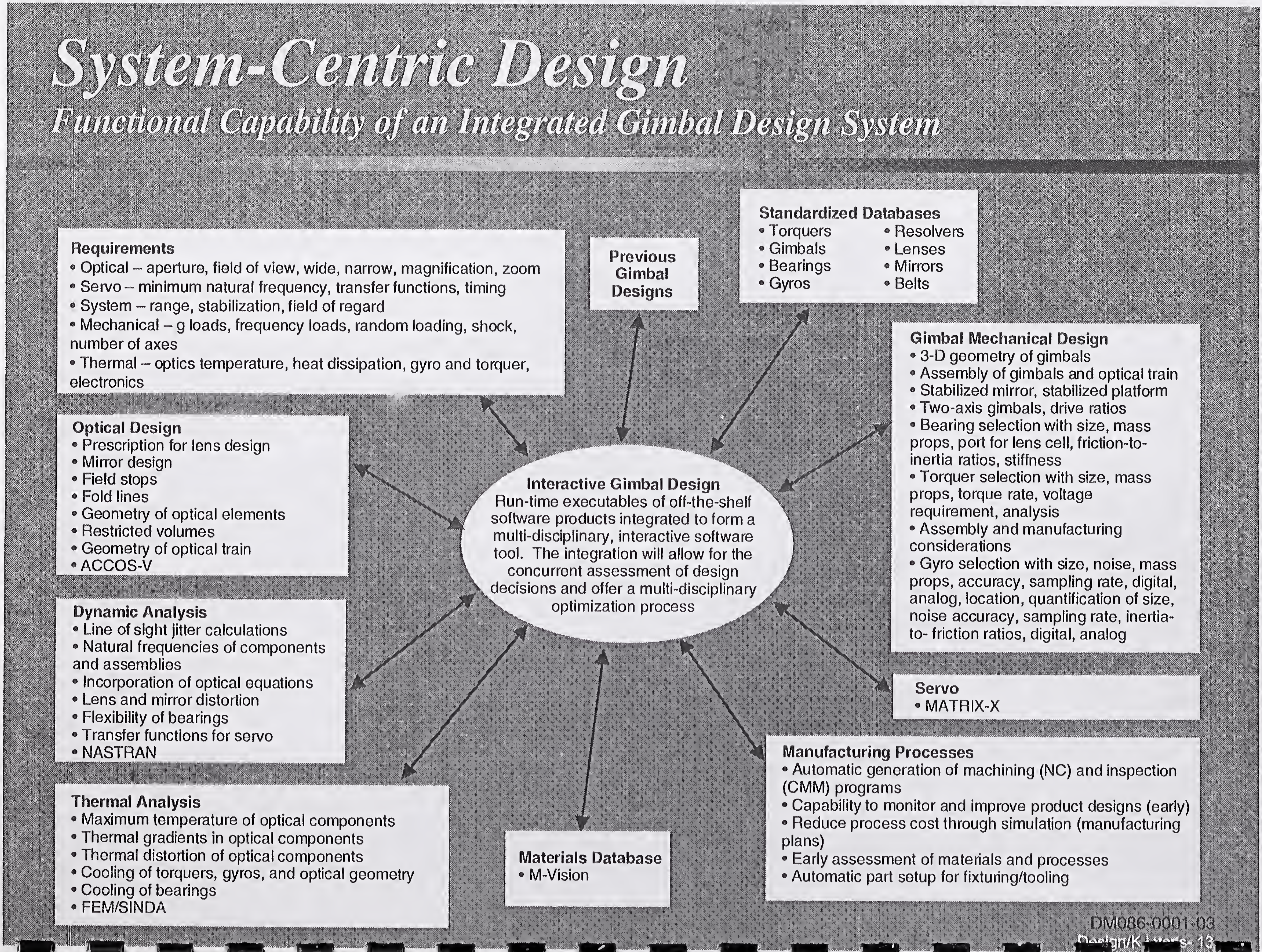


Enable Affordable Acquisition of Complex Electro-Mechanical Systems such as Weapon Systems and Platforms

Realization of a Seeker System



Design/K.Lyons- 17


System-Centric Design
Functional Capability of an Integrated Gimbal Design System
Requirements
• Optical – aperture, field of view, wide, narrow, magnification, zoom
• Servo – minimum natural frequency, transfer functions, timing
• System – range, stabilization, field of regard
• Mechanical – g loads, frequency loads, random loading, shock, number of axes
• Thermal – optics temperature, heat dissipation, gyro and torquer, electronics
Previous Gimbal Designs
Standardized Databases
• Torquers • Resolvers
• Gimbals • Lenses
• Bearings • Mirrors
• Gyros • Belts
Gimbal Mechanical Design
• 3-D geometry of gimbals
• Assembly of gimbals and optical train
• Stabilized mirror, stabilized platform
• Two-axis gimbals, drive ratios
• Bearing selection with size, mass props, port for lens cell, friction-to-inertia ratios, stiffness
• Torquer selection with size, mass props, torque rate, voltage requirement, analysis
• Assembly and manufacturing considerations
• Gyro selection with size, noise, mass props, accuracy, sampling rate, digital, analog, location, quantification of size, noise accuracy, sampling rate, inertia-to- friction ratios, digital, analog
Optical Design
• Prescription for lens design
• Mirror design
• Field stops
• Fold lines
• Geometry of optical elements
• Restricted volumes
• Geometry of optical train
• ACCOS-V
Interactive Gimbal Design
Run-time executables of off-the-shelf software products integrated to form a multi-disciplinary, interactive software tool. The integration will allow for the concurrent assessment of design decisions and offer a multi-disciplinary optimization process
Dynamic Analysis
• Line of sight jitter calculations
• Natural frequencies of components and assemblies
• Incorporation of optical equations
• Lens and mirror distortion
• Flexibility of bearings
• Transfer functions for servo
• NASTRAN
Servo
• MATRIX-X
Manufacturing Processes
• Automatic generation of machining (NC) and inspection (CMM) programs
• Capability to monitor and improve product designs (early)
• Reduce process cost through simulation (manufacturing plans)
• Early assessment of materials and processes
• Automatic part setup for fixturing/tooling
Thermal Analysis
• Maximum temperature of optical components
• Thermal gradients in optical components
• Thermal distortion of optical components
• Cooling of torquers, gyros, and optical geometry
• Cooling of bearings
• FEM/SINDA
Materials Database
• M-Vision
DM086-0001-03

Program Structure

Program is comprised of four focus areas

- Design Exploration and Advanced Design Representation
- Multi-Disciplinary Optimization and Simulations
- Integration Frameworks
- Designer's Interface

Major Contracts include teams Industry, Tool Vendors, and Universities

Carnegie Mellon, MIT, Stanford, Ohio State, Cornell Univ., Univ of Utah, Arizona State Univ., Univ. of Wisconsin- Madison, Rutgers Univ., ISI-USC, ITI

Lockheed-Martin, Rocketdyne, Ford, McDonnell Douglas, Rockwell, Texas Instruments, Boeing Defense, Boeing Computer Services, General Electric, Boeing Helicopters, JPL, SRI, Raytheon, Coleman Research

Optimum Tech, BEAM Tech, McNeal Schwendler, Deneb, Virage, EIT/Verifone

- Industry is highly committed as evidenced by significant matching funds

DARPA -- \$40.3M Industry Match -- \$11.2M

Focus Area: **Design Exploration and Advanced Design Representation**

- Store, Retrieve, and Index design information
  - Textual, Schematics, 3D CAD Drawings, Images
- Documentation of Design and Design Process (Rationale capture)
  - Active & Passive
- Knowledge based engineering for Multi-disc integration
- Create Formal Models from Sketches

- Design for the Future - MIT
- Design Information Retrieval - Virage
- Functional and Diagram Representation - OSU
- Design Space Colonization - Stanford
- Automated Capture of Design Rationale - SRI
- Context Integrated Design - Lockheed Martin
- MADESmart - Boeing
- Knowledge based Eng. - Lockheed Martin

Focus Area: ***Multi-Disciplinary Optimization and Simulation***

- Trade off analysis
- Multi-disciplinary Optimization
- Multilevel Optimization
- Intelligent Product Models
- Probabilistic Models and Analysis for Design
- Integration of CAD / Simulation
  - Composable Simulation Models
  - AI technologies

- Design Sheet - Rockwell
- Collaborative Optimiz.. - GE CRD/Engen.
- Multi-discipl./level simulators - BEAM
- Robust Design Computational System - Rocketdyne/MSC
- Active models for collab. design - Cornell
- AI-Augmented Optimiz. Envir. - Rutgers

Focus Area : ***Integration Frameworks***

- Innovative Frameworks for rapidly configuring design environments
- Intelligent Agents for Collaboration amongst Tools
- WWW Applications and Utilities
- Database technologies for Integrating and Managing Design information

- Collaborative Open Design - CMU
- Agents for Product/Process Integ. Design - ITI
- System Design Advisor - TI
- Integrated Product Definition Environment - Boeing
- Multiphase Integrated Engineering Design - Utah

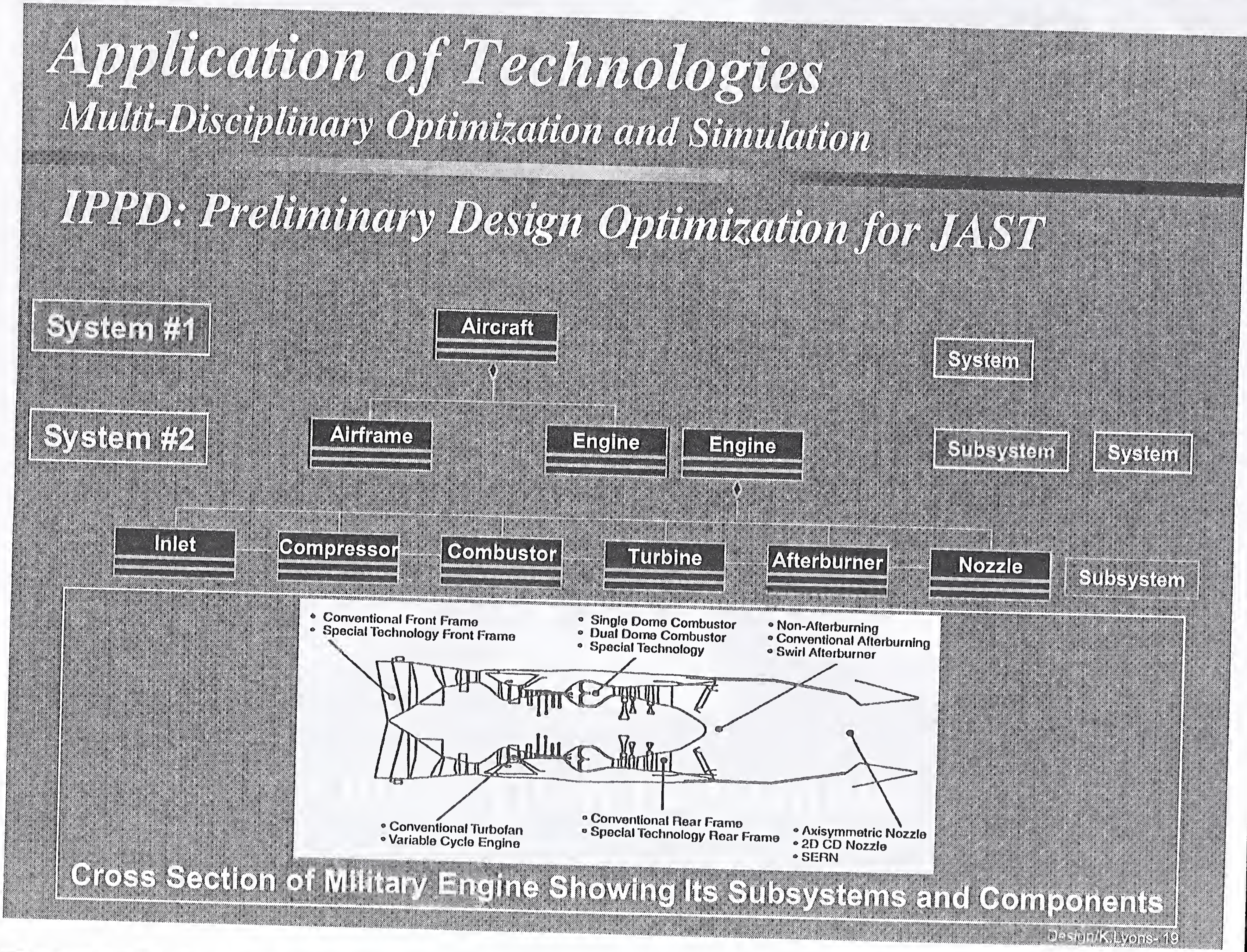
Focus Area : *Designers Interface*

- Electronics Catalogs
- User interface to other design applications
- Incorporate the characteristics of novel / advanced manufacturing processes

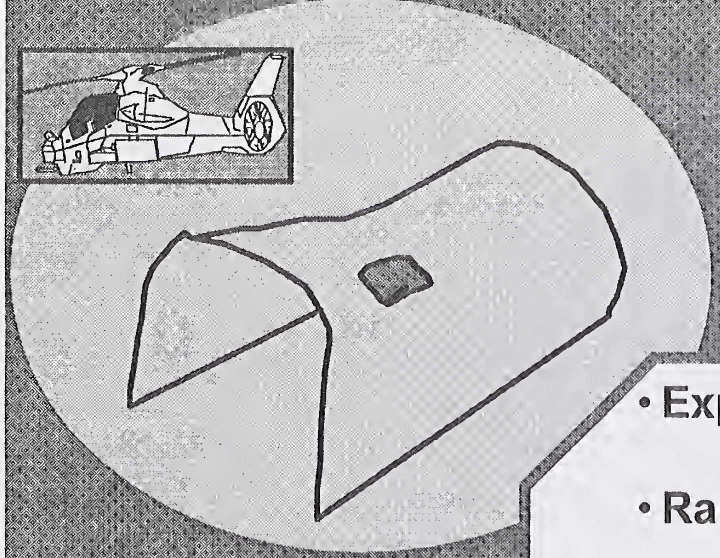
- Manufacturing Simulation Driver - Raytheon
- Virtual Design Workplaces - EITech
- Active Catalogs - ISI
- Interpretive Product Design Environment - JPL
- Planning for Manufacturing - Stanford
- Conceptual Process Design - MSU

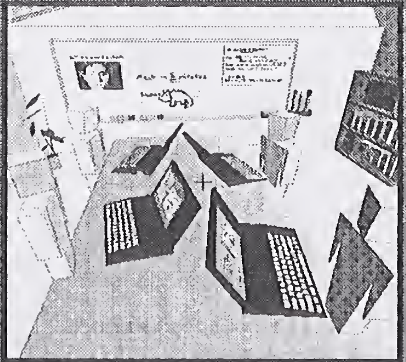
Information Use

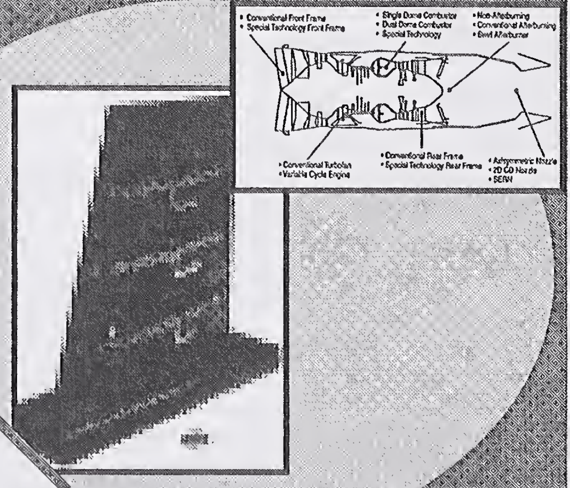
- Current Information
  - *What do we need to share?*
- New information
  - *What is important to capture?*
- Retrieval of information
  - *How can we find what we want?*
- Viewing of information
  - *Now that I have it, can I make sense of it?*


Application of Technologies
Multi-Disciplinary Optimization and Simulation
IPPD: Preliminary Design Optimization for JAST
System #1
System #2
Aircraft
Airframe
Engine
Engine
System
Subsystem
System
Inlet
Compressor
Combustor
Turbine
Afterburner
Nozzle
Subsystem
• Conventional Front Frame
• Special Technology Front Frame
• Single Dome Combustor
• Dual Dome Combustor
• Special Technology
• Non-Afterburning
• Conventional Afterburning
• Swirl Afterburner
• Conventional Turbofan
• Variable Cycle Engine
• Conventional Rear Frame
• Special Technology Rear Frame
• Axisymmetric Nozzle
• 2D CD Nozzle
• SERN
Cross Section of Military Engine Showing Its Subsystems and Components
Design/K.Lyons- 19

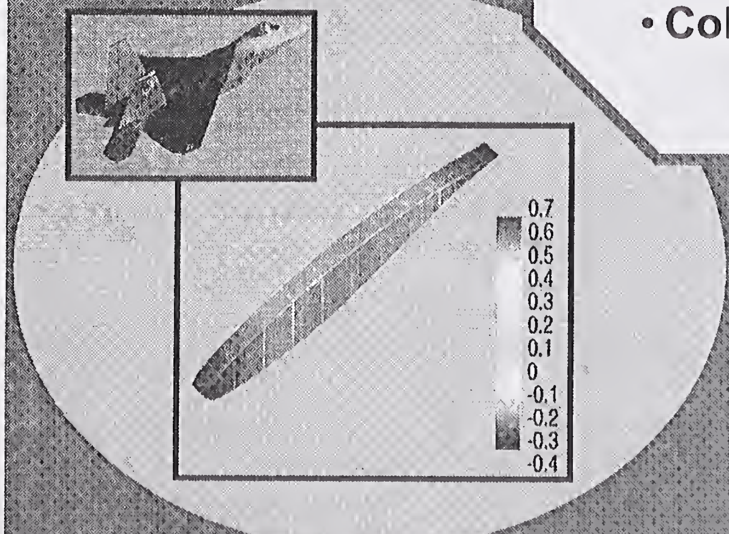
System-Centric Demonstrations

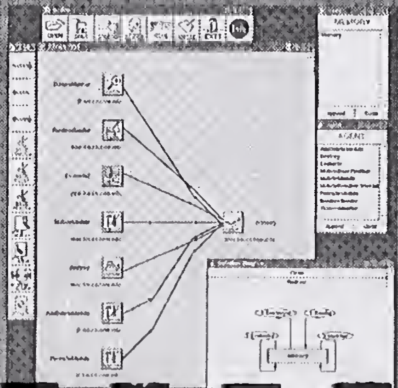


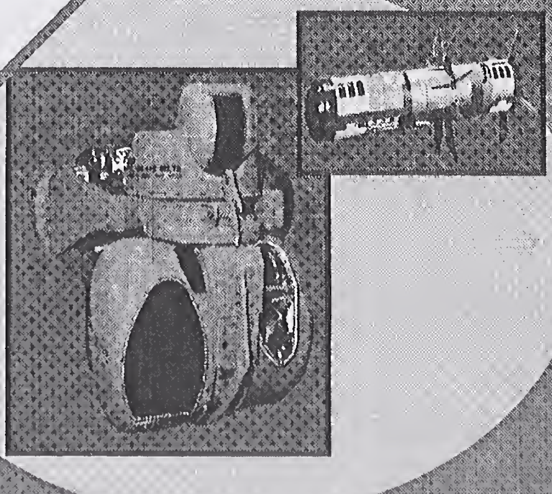




- Exploration of Design Space
  - Optimization across levels
- Rapid Evaluation of Alternatives
  - Rapid config. of design environ.
  - Optimization across disciplines
- Collaborative tools
  - Searchable data
  - Design intent







High-Performance Knowledge Bases (HPKB)

Dr. David Gunning, DARPA (dgunning@darpa.mil)

The summary for this presentation can be found on page 51
16 slides follow

DARPA

High-Performance Knowledge Bases (HPKB)
BAA 96-43

David Gunning
DARPA/ISO

October 21, 1996

High-Performance Knowledge Bases: Program Goal

- The goal of the High-Performance Knowledge Base (HPKB) program is to produce the technology needed to enable system developers to rapidly (within months) construct large (10K-100K axiom/rule/frame) knowledge-bases that provide comprehensive domain coverage, are reusable by multiple applications, and are maintainable in rapidly changing environments.

High-Performance Knowledge Bases: What is a Knowledge Base?

DARPA

Knowledge Base	Conventional Software
Facts (DATA) about the situation	**DATA**
Knowledge: •Ontologies •Domain Theories •Problem Solving Strategies	**Procedural Code:** • Step-by-Step Instructions • Optimized for computational efficiency
Inference Procedure	

Domain knowledge is explicitly stated in concept definitions, axioms & constraints.

Domain knowledge is implicit in the programmer's step-by-step procedural instructions.

What is a Knowledge Base?
Domain Theories

DARPA

Domain Theory

Axioms:

Concept Definitions

```
vehicle( v ) <=>
   physical_object( v ) and
   self_propelled( v ) and
   can( move( v, pt_A, pt_B ) ).

transport( v ) <=>
   vehicle( v ) and can( hold( v, cargo ) ).
```

A vehicle is a self-propelled physical object which can move from one location to another.

A transport is a vehicle which can hold cargo.

Laws Constraints Relations

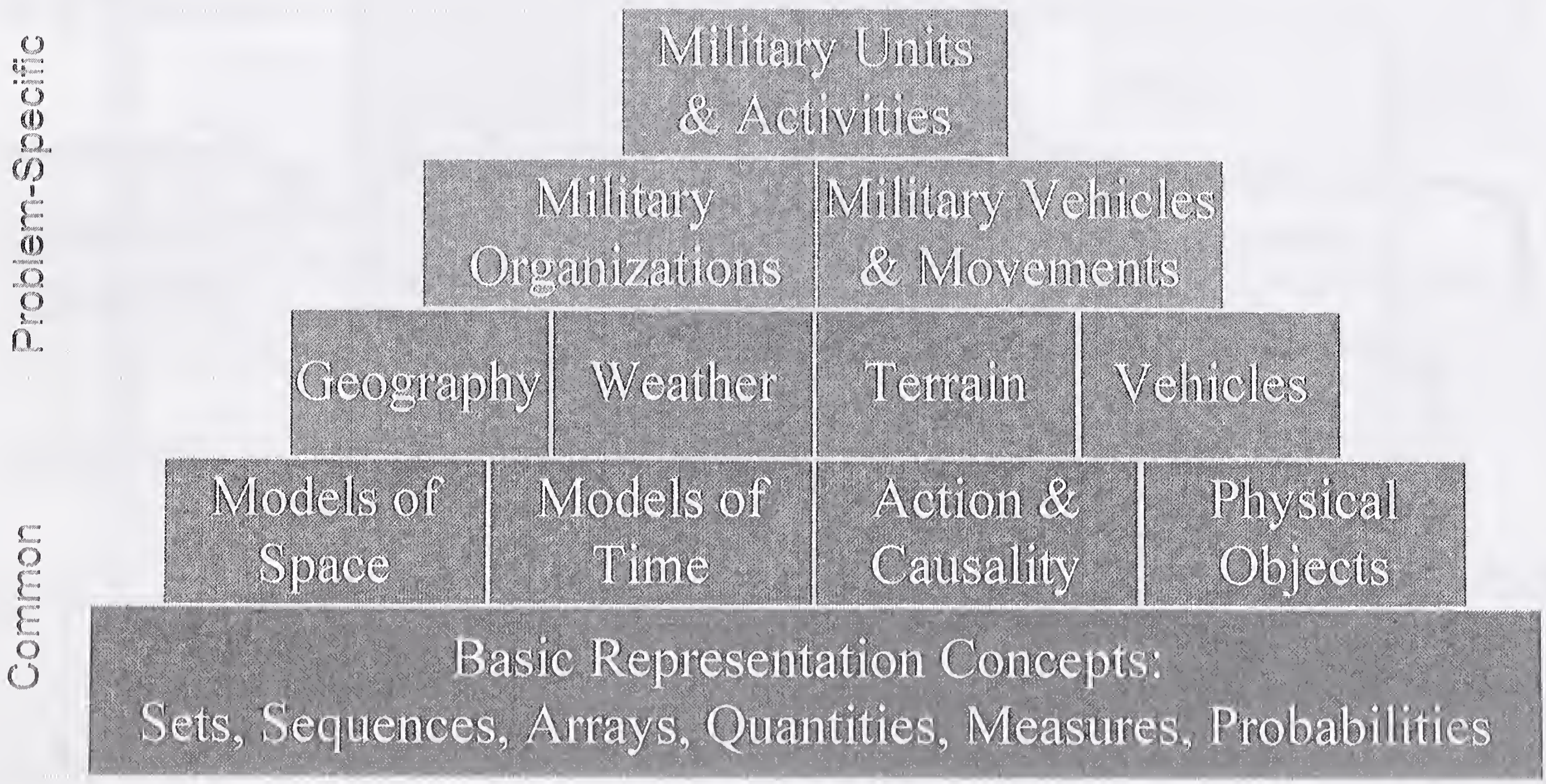
```
location( v, s_1 ) = pt_A and
   exists( p )::p = path( pt_A, pt_B, s_1 ) a
   can( transverse( v, p ), s_1 )
   =>
   result( move(v, pt_A, pt_B ), s_1 ) = s_2 and
   location( v, s_2 ) = pt_B.

transport( v ) and holding( v, cargo )
   =>
location( v, s ) = location( cargo, s ).
```

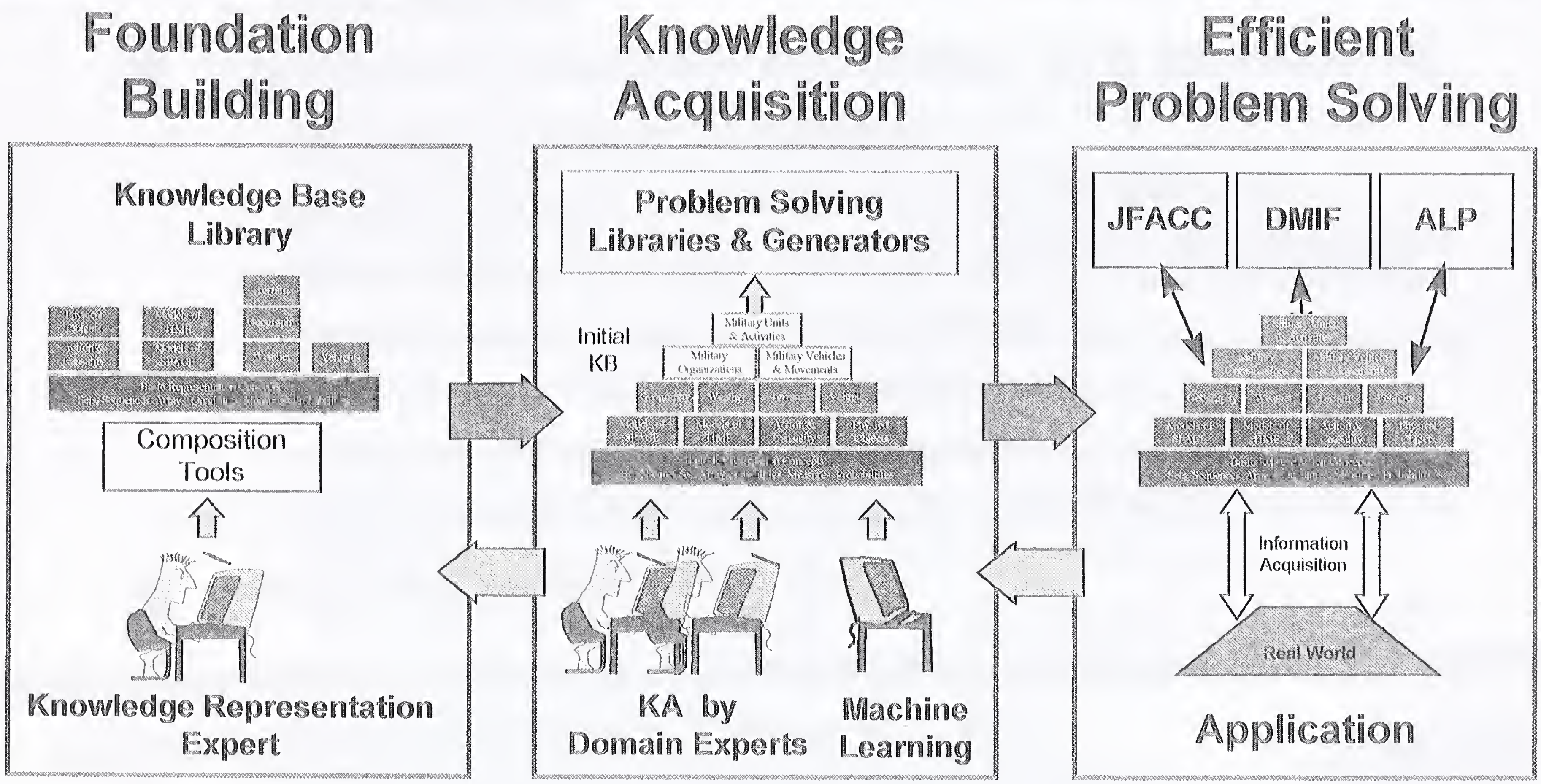
If a vehicle is in location A and there exists a path to location B which the vehicle can traverse, then the vehicle can move from A to B.

If a vehicle is holding cargo, the cargo is in the same location as the vehicle.

What is a Knowledge Base?
Composable Domain Theories



High-Performance Knowledge Bases: Envisioned KB Development Process



High-Performance Knowledge Bases: Foundation-Building Technology

- **Tools and techniques to:**
  - define, modify, test, and verify KB components
  - extend and specialize KB components
  - compare and compose KB components
  - translate among different representations
  - import data dictionaries, schemas, lexicons, and ES rules
  - visualize, browse, and understand KB components
  - represent foundation knowledge
- **Libraries of common ontologies and theories of:**
  - space and time
  - actions and processes
  - belief and uncertainty
  - physical objects and behavior
  - other commonly used terms and domains

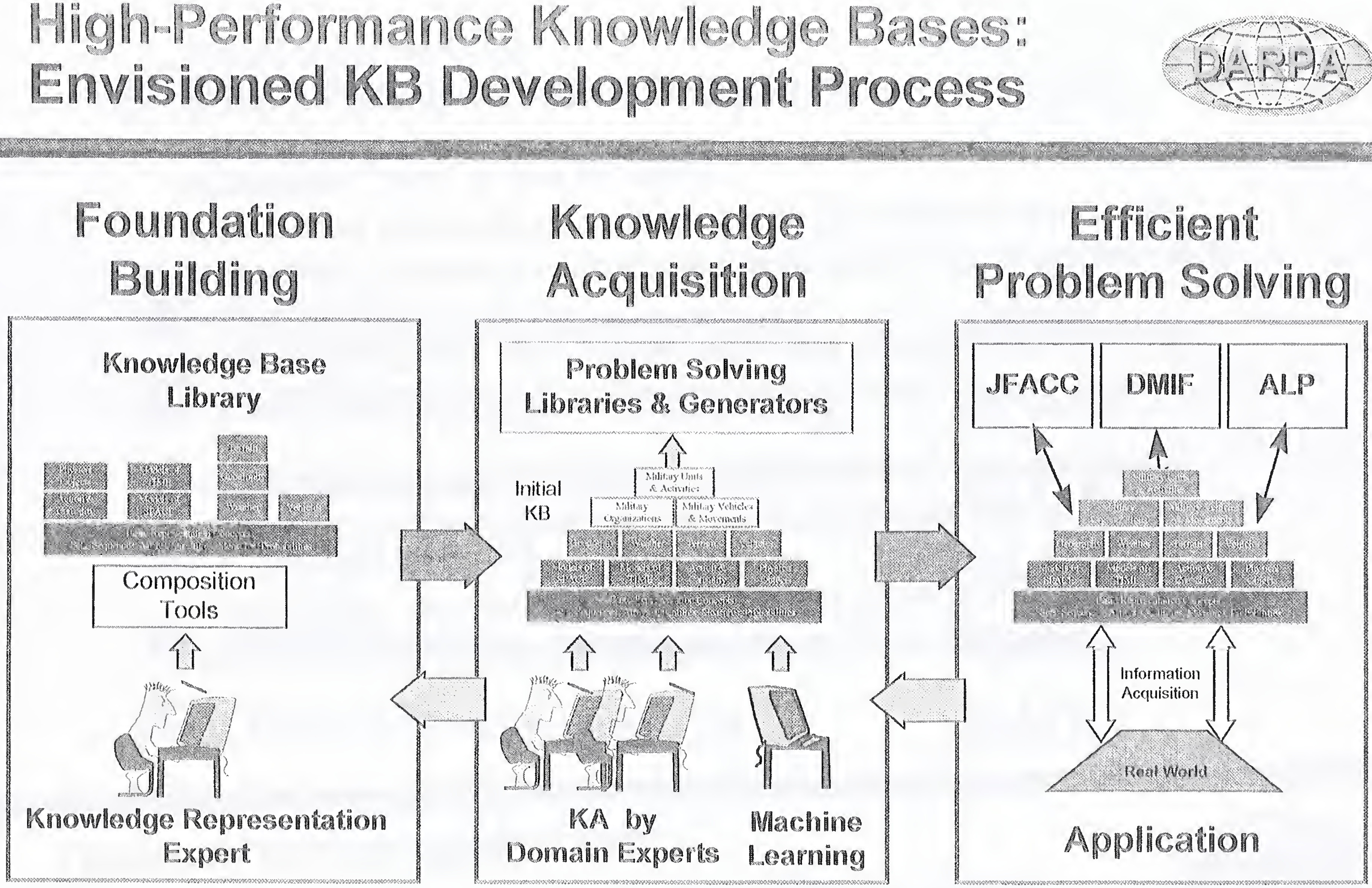
High-Performance Knowledge Bases: Knowledge-Acquisition Technology

- **Tools and techniques to:**
  - automatically generate domain-specific KB editors and visualization tools from foundation knowledge;
  - automatically and semi-automatically learn new knowledge from a variety of human and electronic information sources;
  - automatically discover and extract new knowledge from text documents or from natural language dialogs;
  - assist collaborating teams of domain experts define new knowledge, compare and merge independently developed knowledge into coherent and consistent theories;
  - assist collaborating teams of domain experts compare and merge independently developed knowledge into coherent and consistent theories;
  - provide continuous knowledge acquisition, error checking, anomaly detection, and knowledge maintenance over the life-cycle of a knowledge base;

High-Performance Knowledge Bases: Efficient Problem-Solving Technology

- **Tools and techniques to:**
  - provide efficient reasoning and inference procedures for large knowledge bases;
  - select, extract, transform, and compile knowledge into optimized, application specific problem solving modules;
  - enable the translation, packaging, and integration of problem solving modules into diverse application environments.

High-Performance Knowledge Bases: Envisioned KB Development Process



High-Performance Knowledge Bases: Development Approach

(1) Develop individual foundation-building, knowledge-acquisition, and problem-solving technologies;

(2) Integrate those individual technologies into two or more integrated knowledge-base development environments;

(3) Develop a set of test "challenge problems" which reflect the knowledge-base requirements of the application projects;

(4) Evaluate the alternative knowledge-base technologies and integrated development environments against those test problems; and, in the process

(5) Produce knowledge-base components for use by the application projects.

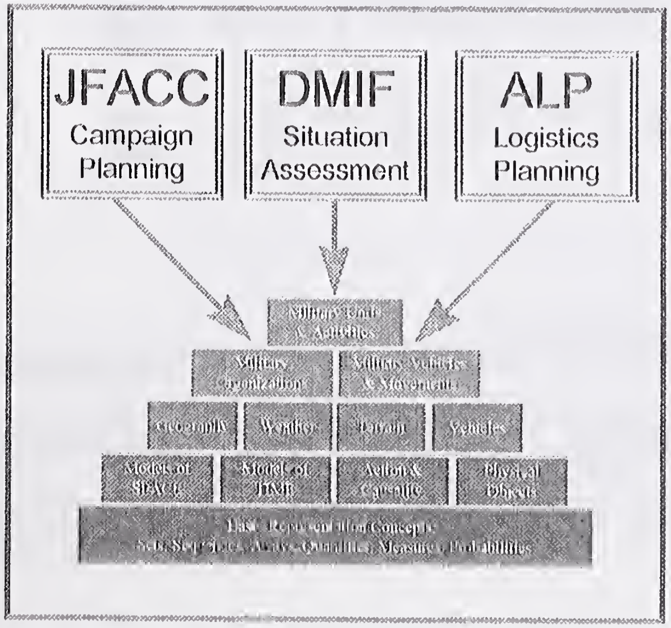
High-Performance Knowledge Bases: Candidate Application Projects

DARPA

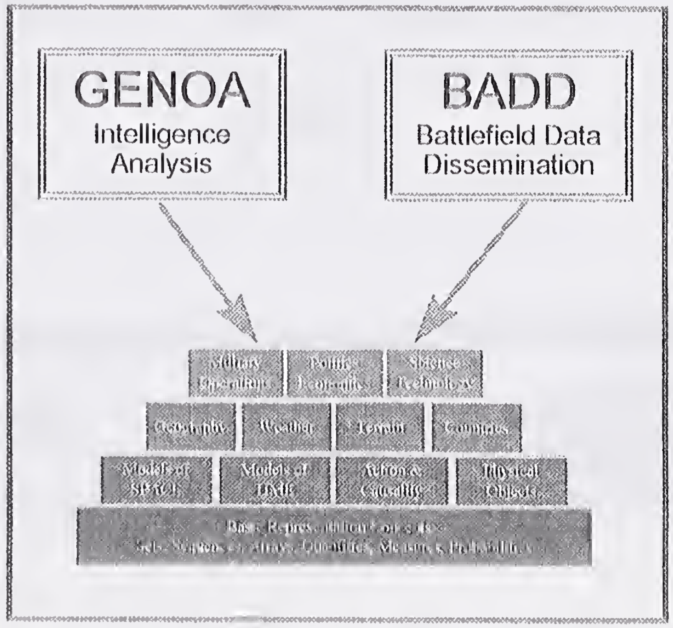
Situation Assessment	DMIF
Air Campaign Planning	JFACC
Command & Control	JTF ATD
Logistics Planning	ALP
Intelligence Analysis	Genoa
Data Dissemination	BADD

High-Performance Knowledge Bases: Challenge Problems

Build in-depth knowledge bases of military situations, operations, and logistics to support a multiple applications.



Build broad-but-shallow knowledge bases of general knowledge to support information retrieval, integration, and dissemination.



High-Performance Knowledge Bases: Challenge Problem Process

- **Challenge Problem Specification:**
  - description of the domain to be covered
  - definition of the terms (in a formal ontology) to be used in specifying input requests or test questions;
  - a set of sample questions and training data if appropriate.
- **Evaluation Process:**
  - build a knowledge base to problem specifications
  - record development time
  - issue test questions and record % correct
  - modify the knowledge base and re-evaluate

High-Performance Knowledge Bases: BAA 96-43 Topics of Interest

DARPA

- **Knowledge-Base Technology Development and Integration:**
  - develop the individual foundation-building, knowledge-acquisition, and problem-solving technologies, as well as combining those technologies into integrated knowledge-base development environments.
- **Challenge Problem Development and Evaluation Management:**
  - define, develop, and maintain the challenge problem test sets, as well as defining and managing the annual evaluation process and evaluation conferences.

High-Performance Knowledge Bases: Approximate Level of Effort

- **Knowledge-Base Technology Development and Integration: $6-7M per year**
  - Individual Technology Development Efforts:
    - 10-30 efforts
    - 200K-600K per year each
  - Integrated KB Development Environments:
    - 2-4 efforts
    - $1M-2M per year each
- **Challenge Problem Development and Evaluation Management: $1-2M per year**
  - 1 prime plus supporting efforts

How Does a Standard Become a Standard?

Ms. Sharon Kemmerer, NIST (kemmerer@cme.nist.gov)

The summary for this presentation can be found on page 53
2 slides follow

NWI (New Work Item)
- requires 5 countries to actively participate
- 3 month ballot cycle to subcommittee members

CD (Committee Draft)
- 4 month ballot cycle to subcommittee members

DIS (Draft International Standard)
- 5 month ballot cycle to ALL ISO* members (currently around 85 countries)

FDIS (Final Draft International Standard)
- 2 month ballot cycle to all ISO members

IS (International Standard)

* ISO - Organization for Standardization ... all non-electrotechnical standards (IEC - International Electrotechnical Commission responsible for all electrotechnical standards)

STEP on a Page

STEP on a Page provides a graphic summary of the progress of STEP, Standard for the Exchange of Product Model Data, the familiar name for ISO 10303. STEP is developed by ISO TC184/SC4.

Status of STEP Parts

The twelve parts that comprise the initial release of STEP are circled in the diagram.

Every part shown in the STEP on a Page has its status shown beside it. The status designators vary from "O" (the ISO preliminary stage) to "I" (international standard—the most advanced stage of standards development and acceptance). Parts designated as "E, F" (levels of draft international standard) and "I" are considered advanced enough to allow software vendors to prepare implementations.

Architecture of STEP and STEP on a Page

STEP on a Page attempts to reflect the STEP architecture by grouping the STEP parts into six main groupings--description methods, integrated-information resources, application protocols, implementation methods, and conformance methodology.

From an architectural perspective, the description methods group forms the underpinning of the STEP standard. This includes part 1, Overview, that also contains definitions that are universal to the STEP. Also in that group, part 11, EXPRESS Language Reference Manual, describes the data-modeling language that is employed in STEP. Parts in the descriptive-methods group are numbered from 1 to 19.

At the next level is the integrated-information-resources group, the parts that contain the actual STEP data models. These data models can be considered the building blocks of STEP. Integrated-information resources are subgrouped into generic resources, application resources, and application-interpreted constructs or AICs. Integrated-generic resources are generic entities that are used as needed by application protocols (APs). Parts within generic resources are numbered in the 40s and are used across the entire spectrum of STEP APs. The integrated-application resources contain entities that have slightly more context than the generic entities. The parts in the integrated-application resources are numbered in the 100s. Because entities in the integrated-resources group are shareable across the application protocols that need them, they can help enable AP integration and interoperability.

The 500 series are application-interpreted constructs, AICs. These are reusable groups of information-resource entities that make it easier to express identical semantics in more than one AP.

At the top level of the STEP hierarchy are the more complex data models used to describe specific product-data applications. These parts are known as application protocols and describe not only what data is to be used in describing a product, but how the data is to be used in the model. The APs use the integrated-information resources in well-defined combinations and configurations to represent a particular data model of some phase of product life. APs are numbered in the 200s. APs currently in use are the Explicit Draughting AP 201 and the Configuration Controlled Design AP 203.

Implementation & Conformance

The STEP implementation-methods group, the 20s, describe the mapping from STEP formal specifications to a representation used to implement STEP.

The conformance-testing-methodology framework group, the 30s, provides: information on methods for testing of software-product conformance to the STEP standard, guidance for creating abstract-test suites, and the responsibilities of testing laboratories. The diagram shows that part 31, which describes the methodology to perform conformance testing, has been approved as an international standard. The STEP standard is unique in that it places a very high emphasis on testing, and places these methods in the actual standard itself.

Abstract Test Suites

The 300 series of parts, abstract-test suites, consists of test data and criteria that are used to assess the conformance of a STEP software product to the associated AP. SC4 requires that every AP contain or be associated with an abstract-test suite. The numbers assigned to ATSs exceed the AP numbers by exactly 100. Therefore, ATS 303 applies to AP203.

ooOOoo

STEP on a Page was conceived and implemented by Jim Nell, National Institute of Standards and Technology.

STEP on a Page

ISO TC184 SC4 — ISO 10303

APPLICATION PROTOCOLS AND ASSOCIATED ABSTRACT TEST SUITES

I 201 Explicit draughting (W)*
I 202 Associative draughting (C)
I 203 Configuration-controlled design (C)
C 204 Mechanical design using boundary rep (C)
C 205 Mechanical design using surface rep (W)
X 206 Mechanical design using wireframe (X)
E 207 Sheet metal die planning and design (C)
W 208 Life-cycle product change process (W)
C 209 Compos & metal struct, anal, & related dgn (W)
C 210 Electronic P-C assy: product-design data (W)

X 211 Electronic P-C assy: test, diag, & remanuf (W)
C 212 Electrotechnical design and installation (W)
E 213 Num contr (NC) process plans for mach'd parts (W)
C 214 Core data for automotive mech dgn processes (W)
W 215 Ship arrangements (W)
W 216 Ship moulded forms (W)
W 217 Ship piping (W)
W 218 Ship structures (W)
X 219 Dimension inspection (X)
O 220 Printed-circuit assemblies: mfg planning (O)

C 221 Process plant functional data & its schem rep (W)
W 222 Design-manuf for composite structures (W)
W 223 Exc of dgn & mfg product info for cast parts (W)
E 224 Mech parts def for p. plg using mach'n'g feat (W)
E 225 Structural bldg elem using explicit shape rep (W)

W 226 Ship's mechanical systems (W)
E 227 Plant spatial configuration (W)
O 228 Building services: HVAC (O)
W 229 Forged parts (W)
W 230 Building structural frame: steelwork (W)

W 231 Process-engineering data (W)
W 232 Technical data packaging: core info & exch (W)
O Neutral optical-data-interchange format (O)
O Product life-cycle support--NATO (O)
O SGML and industrial data (O)

DESCRIPTION METHODS

I 1 Overview and fund. principles (Amend. 1=O)
I 11 EXPRESS lang ref man. (Ed 2=W)
X 12 EXPRESS I lang ref man (Type 2 tech report=W)

INTEGRATED-INFORMATION RESOURCES

INTEGRATED-APPLICATION RESOURCES

I 101 Draughting
X 102 Ship structures
X 103 E/E connectivity

C 104 Finite element analysis
I 105 Kinematics
W 106 Building core model
A 107 Engineering anal core

INTEGRATED-GENERIC RESOURCES

I 41 Fund of prdct descr & spt (ed2=A)
I 42 Geom & topol rep (Amd 1=W)
I 43 Repres specialization (ed 2=A)
I 44 Product struct config (ed 2=A)
F 45 Materials

I 46 Visual presentation
F 47 Tolerances
X 48 Form features
F 49 Process structure & properties

APPLICATION-INTERPRETED CONSTRUCTS

C 501 Edge-based wireframe
C 502 Shell-based wireframe
C 503 Geom-bounded 2D wireframe
C 504 Draughting annotation
C 505 Drawing structure & admin.

C 506 Draughting elements
C 507 Geom-bounded surface
C 508 Non-manifold surface
C 509 Manifold surface
C 510 Geom-bounded wireframe

C 511 Topol-bounded surface
C 512 Faceted B-representation
C 513 Elementary B-rep
C 514 Advanced B-rep
C 515 Constructive solid geometry

X 516 Mechanical-design context
C 517 Mech-design geom present'n
C 518 Mech-design shaded present'n
C 519 Geometric tolerances
C 520 Assoc draughting elements

IMPLEMENTATION METHODS

I 21 Clear text encoding of exch str.
E 22 Standard data access interface
E 23 Early C++ (binding for #22)

C 24 Late C (binding for #22)
X 25 Late FORTRAN
E 26 IDL (binding for #22)

CONFORMANCE TESTING METHODOLOGY FRAMEWORK

I 31 General concepts
E 32 Requirements on testing labs and clients
X 33 Abstract test suites
C 34 Abstract test methods for Part 21 impl.
W 35 Abstract test methods for Part 22 impl. (Approved for new scope)

jgnell, 89-Oct 23; rev 98-Mar-16; Origin: ISO 10303 Editing Committee

E, F, I safe to implement

*W = status of separate abstract test suite, e.g. ATS 3xx=W

Legend: Part Status
O = Preliminary Stage (Proposal-->approve NP ballot)
A = Proposal Stage (NP circulation-->NP approval)
W = Preparatory Stage (Working Draft devel.-->CD regis)
C = Committee Stage (CD circulation-->DIS regis)
E = Enquiry Stage (DIS circ.--FDIS circulation)
F = Approval Stage (FDIS circ-->Int'l Std regis)
I = Publication Stg (Int'l Std approved & published)
X = Discontinued

Presentation: Breakout Session #1—Industry Perspective

Dr. Michael Barbieri, Lockheed Martin (MPBarbieri@lmtas.lmco.com)

The summary for this presentation can be found on page 55
2 slides* follow

*These slides were originally handwritten and have been retyped.

Industry Needs

1. NIST Material Database
   - Supplier input material data with minimum requirements, and with pedigree to ASTM standards
2. C.O.T.S. (Commercial off the Shelf) Database
   - NIST D/B with supplier input minimal specifications
   - Ability for end user to query supplier for specific information
3. Information/Design Broker
   - Act as a go-between for suppliers and end users
4. Design Intent/Diary Format
   - Determine what information should be recorded on designs to create archive of information

Industry Needs

5. National Research Database
   - Catalog with one-page summary of government and university research
   - Points-of-contact
   - Interactive to request research or assistance

6. Improve STEP
   - Speed up process
   - Follow up with additional information – build up

7. More Research/Commercialization of 3D Packaging Tools

Presentation: Breakout Session #3—Standards for Information Modeling/Knowledge Representation

Dr. Simon Szykman (szykman@cme.nist.gov)

The summary for this presentation can be found on page 57
3 slides* follow

* These slides were originally handwritten and have been retyped.

Standards/Information Modeling and Knowledge Representation

Current Practices

- Use of existing standards
  - Idef, EXPRESS and STEP
  - Some good translators are available
  - Limited to geometric information
  - Standards are used as exchange mechanisms, not for data sharing
  - Typical loss of information
  - Database issues
  - Numerous barriers to use of standards
  - STEP development is slow

Standards/Information Modeling and Knowledge Representation

Needs and Recommendations

- Immediate Needs
  - Parametric information
  - Tolerances
  - Features
  - Constraints and assembly
- Priorities
  - DFM (for artifact and process)
  - Geometric precision problems
  - Design rationale and intent
  - Database issues
  - Integration issues
  - Design retrieval/redesign

Standards/Information Modeling and Knowledge Representation

- Long-Term and Other Issues
  - DFM (for artifact and process)
  - Conceptual design issues
  - Representation of function, system design
  - Multidisciplinary issues
  - Ontologies (part/material libraries)
  - Relating measurement data to design and process data
  - Building architectural standards
  - Internet issues

Papers

Function-Based Engineering Part Retrieval

Yumi Iwasaki, Richard Fikes, Adam Farquhar, and Robert Engelmore

The summary for this presentation can be found on page 35
The slides for the presentation start after page 278
8 pages follow

Function-Based Engineering Part Retrieval

Yumi Iwasaki, Richard Fikes, Adam Farquhar, and Robert Engelmore

Knowledge Systems Laboratory
Department of Computer Science
Gates Bldg. 2A, m/c 9020
Stanford University
Stanford, CA 94305
{iwasaki, fikes, axf, rse}@ksl.stanford.edu

Abstract

The Internet provides dramatic new opportunities for gathering information from multiple, distributed, heterogeneous information sources. However, this distributed environment poses difficult technical problems for the information-seeking client, including finding the information sources relevant to an interest, formulating questions in the forms that the sources understand, interpreting the retrieved information, and assembling the information retrieved from several servers into a coherent answer. This paper describes a function-based approach to address this problem in the context of engineering part retrieval. In particular, this paper addresses the problems specific to the task of searching for device components in product catalogs based on a description of a desired function. We plan to implement the proposed approach as part of the general information broker architecture being developed at Knowledge Systems Laboratory.

1. Introduction

Engineers need ready access to a broad range of information in order to do their work, and finding the right information is often cited as the #1 technical problem in engineering design. It is reported that designers in industry spend over 50% of their time, retrieving, organizing and handling information [Court, Culley et al. 1993; Baya and Leifer 1995]. The figure is expected to well exceed 50% during early stages of novel design projects [Leifer 1995].

Designers need many kinds of information, including descriptions and models of previous designs that satisfy functional requirements similar to those of a current design task and of parts and devices available for purchase that are candidate components of a new design. The designer's task of obtaining such information is made significantly more difficult because of the multiple forms of descriptive criteria used in specifying the queries, including desired structural, behavioral, and/or functional characteristics.

The amount of time required to search for device components in product catalogs, even if they are computerized, is often prohibitive. The result is the well-known tendency for engineers to reuse previous solutions that they are familiar with and to take advantage of the information that is on their bookshelves within arm's reach.

Tools for automatically searching and retrieving information on relevant device components would be a powerful and welcome tool to engineers. The Internet provides dramatic new opportunities for developing such tools that can gather information from multiple, distributed sources. However, the task is not easy for several reasons:

- The volume of information is enormous. As an example, the Thomas Register [Thomas Publishing Company][1] contains approximately 190,000 suppliers of parts.
- The information is inherently distributed. It resides with the producers of the parts and systems and is stored in many forms. The Thomas Register provides the list of suppliers' addresses and phone numbers under each class of products but not detailed information about each product. Many suppliers also describe their products in separate "catalog pages" in the Thomas Register, but those pages are essentially advertisements and rarely include the detailed product specifications needed by an engineer.

[1] It is the most commonly used reference on engineering products and suppliers.

- No single taxonomy effectively covers the majority of intended applications. For example, an engineer might need a mechanism that will convert rotary to linear motion with torque around one Newton-meter and length of travel around two meters. Solutions include rack-and-pinion gearing, cable drives, shaft drives with skewed rollers, recirculating ball screws, and linkages. No standard taxonomy would include such a varied set of mechanisms.

We plan to use functional descriptions as well as taxonomic information about parts and systems to find potential solutions and weed out inappropriate choices. Our approach will be embodied in an information broker which can take advantage of explicitly encoded ontology of functional knowledge.

A competent broker of engineering products can use information such as functional, behavioral and physical descriptions, the context of use, and other general characteristics including size and cost, to quickly find available products that best match a client's needs. A broker should also be able to suggest several alternative methods to achieve a functional goal and be able to ask effective questions in helping the client decide among them. Like human brokers, effective computer-based information brokers must take advantage of domain-specific knowledge, such as:

- The terminology used to describe a product, including functional terminology as well as taxonomic terminology along multiple dimensions;
- Typical ways for achieving functions (i.e., the types of components and the ways they are used to achieve certain functions);
- Characteristics of components that are relevant to achieving different functions.

While effective brokering requires much specialized domain knowledge, building such brokers as ad hoc, monolithic applications for each domain will not scale. What is needed here is a general system architecture for information brokers that can make use of domain-specific ontologies of products and functions to perform effective brokering in the domain.

In this paper, we propose approaches to enable building of such domain-specific information brokers of engineering components. In particular, we propose:

(1) An information retrieval scheme using functional specifications as well as taxonomic information along multiple dimensions.

(2) A methodology for codifying domain-specific functional and taxonomic ontologies, and computational tools to assist in development and use of such ontologies.

2. Retrieval Based on Function

A salient feature of the information retrieval task in the engineering domain, not shared by some other domains (e.g., a bibliographic retrieval service or a travel agent), is the importance of a functional goal. A functional goal is the specification of the function that the client hopes to achieve with the component being sought. The importance of functional specification is not an incidental fact about the task but is due to the very nature of the discipline: Engineering products are artifacts designed to achieve some functions and engineering design is always undertaken to meet some functional specification.

Thus, it is critical that the client is able to specify the query in terms of functions to be achieved. Existing taxonomies, as found in the Thomas Register or PartNet[2][Crandall 1993] directories, are organized around the words that appear in the names of devices and not organized around the functions that devices perform. Such organization makes it difficult to retrieve parts based on a description of what the user is seeking to achieve with the part because (1) names may not be a good indicator of the function, and (2) it requires one to know the name of the class of parts that can achieve the function before searching for them.

Although the name of a component or a device in some cases may suggest its function, as in the case of "generator" or "fastener", many names, such as "gear" or "belt", do not. Furthermore, relying on a name, such as "generator" to search for parts to achieve a function "to generate" is not sufficient for several reasons:

(1) The name of a class is too general to be a functional specification by itself. For example, the index pages of the Thomas Register reveal that there are over 70 classes of devices under "generator". They include

[2] PartNet, developed at the University of Utah , is a project to provide direct, interactive on-line access to parts catalogs. This access relies on the Internet to provide an efficient communications medium for transferring parts information from vendors to customers.

everything from "Acetylene Gas Generator" to "Television Synchronizing Generator".

(2) The class name is too specific to be a functional specification by itself. Many devices that serve the function of generating something are not called "generators". For example, while an "alternator" generates electricity and a room "humidifier" generates steam, neither of them are indexed under "generator".

(3) The name of a class does not necessarily indicate the function implied by the name. Despite the name, "generator" does not necessarily mean that the function is to generate. For example, the function of "tachometer generator" is to measure angular velocity.

(4) There is generally not a one-to-one correspondence between a class in a standard taxonomy and a function. One class of devices can be used to achieve multiple functions and one function can be achieved in many different ways.

An even more fundamental problem is that a taxonomic hierarchy, such as employed by the Thomas Register or PartNet, is useless when searching for parts if one does not already know what parts to use to achieve one's goal or if one is looking for alternative ways to achieve a known function. An effective broker must be able to understand the user's description of what he/she needs to achieve, to suggest a variety of methods for achieving it, and to retrieve appropriate parts.

We propose the general scheme shown in Figure 1 for going from a user-provided functional specification to available parts.

The functional specification schema represents the functionality desired by the user. The device ontology is a library of "devices" indexed by function. Each type of device specifies a class or classes of components in the information source (e.g., the PartNet hierarchy of parts) that is commonly used to achieve the function. The device ontology is the body of knowledge that allows mapping from the user-specified functionality to available parts. The following subsections discuss representational and inferential elements that are needed to realize this scheme, but before doing so, we describe our view of how the user may provide the functional specification through interactions with the broker.

Formally, the problem of finding parts that achieve a function can be viewed as a mapping from the input functional specification to a set of parts. If one could expect the user to start with a clear idea of the desired functionality, including the constraints to be satisfied by the parts, the task of a part broker would reduce to simply finding such mapping. In reality, however, a user is not likely to start with a complete, well-articulated functional specification. A user is likely to start with some high-level specification of the desired functionality, such as "generate steam", and he/she may willy-nilly provide further constraints such as "portable" or "electrical" upon seeing the range of possibilities the first query produces.

An important part of helping the user select a part is enabling him/her to refine the functional specification by suggesting additional information that might narrow the search. The process of going from functional specification to parts should not be expected to happen in one

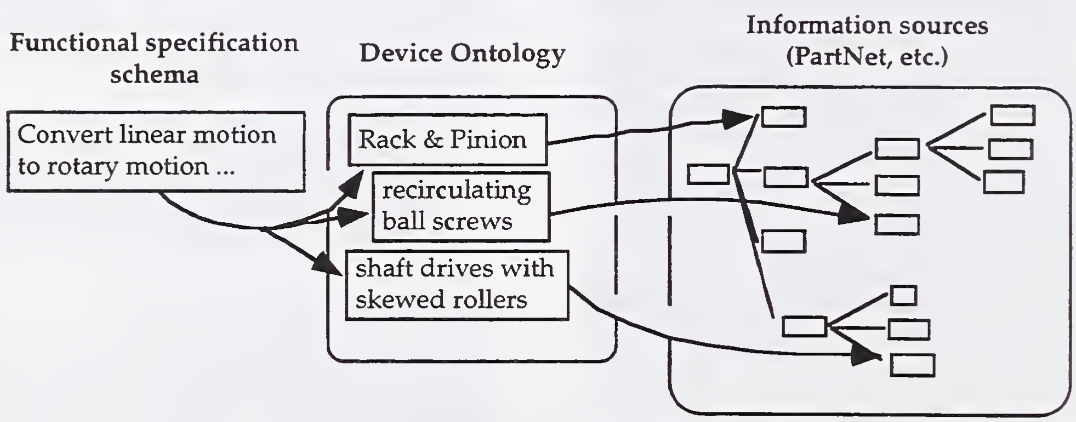


Figure 1: The general scheme for going from function to parts

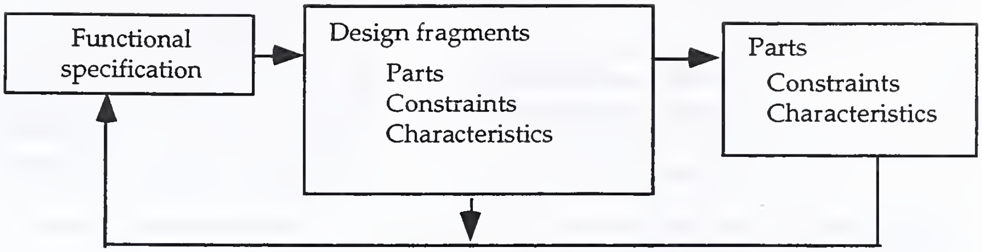


Figure 2: Incremental functional specification refinement-part retrieval process

step but proceed in many iterations, as shown in Figure 2. Each iteration results in a more refined functional specification, a smaller set of candidates, and additional constraints that may be solicited from the user to narrow down the search even further.

2.1 Functional Specification Schema

A functional specification formalism should satisfy the following requirements to be practical for engineering part retrieval:

(1) It should be easy for a practicing engineer to write and understand functional specifications without special training. Therefore, it should use the common vocabulary of the domain and employ a combination of ordinary English words and domain-specific vocabulary.

(2) It should allow a large variety of functions to be represented at different levels of details. It should also allow a large variety of information to be included such as cost, size, weight, context of use, etc., depending on the type of function.

There have been a number of proposals for representing knowledge of functions. One important approach is to specify function by specifying the input/output behavior of a device. This is simple and uniform, and may be quite appropriate for some domains such as digital circuits. In many domains, however, devices do not have well-defined inputs or outputs, or if they do, people do not think of them in these terms. For instance, devices such as fasteners and belts do not have well-defined inputs and outputs.

The functional representations proposed by Sembugamurthy [Sembugamoorthy 1986] and Iwasaki [Iwasaki, Vescovi et al. 1995]employ an abstract specification of behavior and the expected causal interactions among components. These representations allow hierarchical decomposition of functions so that one can represent the function of a device at many levels of detail. The Causal Functional Representation Language (CFRL) [Iwasaki, Vescovi et al. 1995] makes the structure of a device and the context of its use explicit. Furthermore, CFRL has a clear semantics defined by the dynamic behavior of the device; as a consequence, a functional specification can be evaluated against an actual (simulated or observed) behavior of a device. However, for the purpose of part retrieval, those representations are likely to be overly elaborate and cumbersome.

A more fundamental problem, however, in using functional representations such as CFRL to represent a desired function for the purpose of part retrieval is that those representations are geared towards representing the functionality intended by the designer of a device. On the one hand, a CFRL representation of a function is an abstract representation of the behavior the designer expects a given structure to exhibit to achieve some effect, and, as such, it includes a specification of the structure as well as the specification of the conditions under which it is to be operated. On the other hand, what we need to represent for retrieval on functions is the concept of a functionality that is desired but not yet associated with any particular structure (or part). The distinction is subtle but important for our purpose. We will call the former type of function, designed function, and the latter target function in the rest of the document.

For our present purpose of part retrieval, we propose to represent target functions as a combination of a verb, objects, and a set of qualifiers. Since this form of specifying a function as "to do something" is how people normally describe a function in English, it should be easy for an engineer to use the form. We call this form a functional specification schema. Each functional specification schema represents a function in the form of "to do something", as in "to generate electricity" and "to heat water".

To do*: generate **What?*:** electricity **Energy source:** gasoline **Power output:** 100~200 watts **Portable?:** **Emergency?:** ...	**To do*:** convert **From:** rotary motion **to:** linear motion **Torque:** 100~200 watts **Length of travel:** ...

Figure 3: Examples of functional specification schemata

Each schema has a required slot for a verb to specify the action, and zero or more slots for the objects to be acted upon. A schema also can have any number of optional slots to elaborate the action further. The exact set of slots depends on the verb and the objects.

Figure 3 shows some examples of functional specification schemata. The slots marked with * are required slots, while other are optional slots. The level of indentation shows the dependency among slots. For example, "to generate" requires the slot for specifying what is to be generated. "To convert" requires "from" and "to". Finally, depending on the content of the action (to do) and object slots (What?, From, and To), the schema may have any number of additional slots to further qualify the function.

Formally, the functional schema reifies the abstract concept of a target function. Without reification, we can represent that a certain class of devices generates electricity by introducing a relation *Generates* and stating

Generates(device-instance-1, Electricity).

This representation makes it difficult to state additional properties about the function of generating electricity such as the power output, the fuel used, and so on. These properties are characteristics of this instance of the generates relation. A function schema reifies the relation and allows it to be treated as an individual. In practical terms, the functional schema allows the user to specify the desired functionality of a part in a simple form that is easy to write and to understand.

2.2 An Ontology of Function

The system must have a vocabulary of terms that can be used in a functional schema. An ontology names and describes the entities that are assumed to exist in a domain and the predicates that are used to represent relationships among those entities. In other words, an ontology not only provides a vocabulary for representing and communicating knowledge about the domain but also makes explicit the relationships that are assumed to hold among the terms of that vocabulary. For functional schemata, this means that the ontology defines the slots that are relevant for an action and the vocabulary that can fill the slots.

Functional specification schemata include the required slots for the action, the objects, and types of the qualifiers. Thus, the ontology must define the terms that correspond to actions (e.g., to generate, to convert, to fasten, to support, etc.), and also the terms that are objects of the actions. In addition, functional specifications can include a variety of qualifying information that can help to narrow the search. For example, consider a functional specification for a mechanism that will convert rotary motion to linear motion. The specification "to convert rotary motion to linear motion" can be further qualified by the desired range for torques and lengths of travel. Thus, the ontology must also include a set of predicates and relations that are meaningful for further elaborating the functional specification. The vocabulary of terms used for such qualification can be large, but we conjecture that for any given combination of verb and object (or objects), there is a fairly standard set of qualifications that make sense.

For the purpose of part retrieval based on function, it is important for the system not only to have a vocabulary of words as mere symbols attached to products but also to know their meanings (i.e., definitions). Not all products that fit a desired qualification (e.g., portable) may be explicitly marked as such, and the system must be able to judge whether a given candidate part fits the qualification using the definition.

Note that the ontology must allow polymorphic definitions of terms to enable context-sensitive interpretation of terms in functional specifications. There are many terms, whose meaning depends on the type of function or the type of device in question. For example, "portable" is a term used to describe many different types of devices. However, its meaning in terms of the actual size, weight, etc., of the device varies depending on the type of device. A camera that weighs 100 pounds is hardly portable, while an electric generator that weighs the same amount may be called "portable". Likewise, the meaning of the qualifier "emergency" depends on whether it is used in reference to electric generators or vehicles.

2.3 An Ontology for Devices

Given a specification of a desired function, an information broker must now find parts that can achieve the function. Our approach is to build an ontology of devices to enable information retrieval from functions. A device ontology defines classes of devices and their properties; it will be indexed both by function and by a standard taxonomy such as that found in PartNet, the Thomas Register, or the Federal Classification System [Defense Logistics Agency]. Here, we are using the word "device" in a broad sense. We define a device to be "something that has a function" [Keuneke 1991; Iwasaki, Vescovi et al. 1995]. In most cases, a device in our ontology will correspond to a class in PartNet or the Thomas Register. However, in some cases, a device may consist of a configuration of several classes of parts that can together achieve a function. In either case, a device in our ontology represents a design fragment, which is an abstract design of a functional unit to be instantiated by the user's choice of particular instances of the class(es).

In any established engineering domain, there is a set of standard engineering techniques that many engineers know to achieve an often-needed function. For example, to achieve the function "to convert rotary to linear motion with torques in the range of one Newton-meter and lengths of travel about two meters", an experienced engineer can list solutions including "rack-and-pinion gearing", "cable drives", "shaft drives with skewed rollers", "recirculating ball screws", and "linkages". We conjecture that the number of such techniques of typical functions is relatively small (on the order of 10s, not 100s per function).

The devices in the ontology will be indexed along a standard taxonomic hierarchy as well as functions. Functional indexing is accomplished by associating each device class with one or more functional schema template to be matched by the user-provided functional specification. The template includes not only the action and the set of objects to be matched, but also relevant constraints on the rest of the specification. For example, the device class of "photovoltaic cells" can match the functional specification "Generate electricity" but has constraints on desired power output and the energy source.

To allow part retrieval, each device class also has a pointer to a class in PartNet hierarchy. There may also be a set of further constraints for filtering the instances in the class. Thus, the retrieval process proceeds as follows:

(1) Given a functional specification schema S, the system retrieves a set D of devices whose template matches S.

(2) For each element in D, the system retrieves the set P of PartNet classes each with associated filtering constraints.

(3) For each element of P, the system filters its instances using the filtering constraints and constraints from S. The parts that remain are presented to the user.

This approach has the following advantages:

(1) Functional specification templates provide a way to organize devices into functional taxonomies with any degree of specificity. In general, the number of possible functional specifications is very large, and there is not likely to be a way to organize devices into a simple class-subclass hierarchy. The functional templates associated with each class provide a way to classify devices along the functional dimension and serve as a clear definition of each class.

(2) Having a device ontology of its own distinct from the information sources allows the system to retrieve parts based on functions even if there is no clear mapping from a function desired to a class of parts in one information source. Some functions may even require parts from totally separate classes of parts or from different information sources (catalogs).

(3) The device ontology allows decomposition of functions. When a function can be decomposed into sub-functions, the device class representing the overall function can point to other classes that achieve the subfunctions. This promotes modularization of the device ontology and avoids duplication of information within the ontology by enabling sharing of subfunctions.

(4) The constraints that are part of the templates as well as the filtering constraints directly suggest the types of additional information to be solicited from the user to refine the functional specification.

There is a wide range of approaches one can take to this problem of mapping from function to parts differing on the level of difficulty. The simplest is to attach a functional specification to each class of parts in an information source. This approach does not work well if the parts in the information source are not organized around functions, as is typically the case. The most sophisticated approach is to automatically design a configuration of parts to achieve the function. Automatic design is a subject that has been researched extensively but that remains impractical except in limited cases requiring only parametric design. The approach proposed here aims for a middle ground between the two extremes that will be most practical and useful for practicing engineers. One can start from a basic set of design fragments, and as the device ontology grows, the information broker can move along the spectrum of sophistication.

3. Related Work

There is a considerable amount of related work in both the areas of information seeking agents and functional representation.

3.1 Information seeking agents

Three research efforts on information gathering from heterogeneous information sources are closely related to the information brokering aspects of the work described here, although they do not make use of functional representations. These are: research at ISI on the SIMS project [Knoblock, Arens et al. 1994], research at MCC on the Carnot project [Collet, Huhns et al. 1991], and research at AT&T's Bell Laboratories on information gathering agents [Levy, Sagiv et al. 1994]. The Carnot project relies on the large common sense knowledge base of Cyc [Lenat and Guha 1990], assuming that it can be used for all domains. In contrast, we intend to use smaller domain models so that (1) it will be easier for the broker to maintain its own small domain-dependent ontology than to incorporate its ontology into the very large ontology used in Cyc, and (2) it will be easier to write articulation axioms relating the vocabulary of each information source to the relatively small broker vocabulary than to the huge global vocabulary of Cyc. The research on SIMS contributes most to the area of query planning. SIMS focuses on query optimization through learning agents. Its agents learn efficient ways to access multiple information sources for well-defined queries and try to improve efficiency by caching frequently retrieved or difficult to retrieve information. The research at AT&T's Bell Labs on information gathering agents also focuses on query optimization. Their main contribution is in providing a method for determining the minimal set of the relevant information sources needed to answer a given query.

The Information Brokers Project at the Knowledge Systems Laboratory is also developing technologies to enable a marketplace of network-based information brokers that retrieve information about services and products via the Internet from multiple vendor catalogs and data bases. This approach to information brokering does not assume that clients are easily able to articulate an exact query. This is critical in the Internet environment because the pool of clients and information sources is potentially enormous, varied, and dynamic. Therefore, even very sophisticated clients will find it impossible to know the full range of relevant information sources and vocabulary available. Clients also may wish to know about information that is relevant to their query even though they did not explicitly request that information. Consequently, it is essential to offer explanations to the clients. This project differs from other research efforts by focusing on the difficulties in formulating queries, explaining retrieved information, and designing tools for developing and maintaining information brokers. As these considerations are also important for part retrieval, we intend to make full use of the technologies developed by the Information Brokers Project.

3.2 Functional representation

There has been significant previous work on both representing function and using function to reason about physical devices. CFRL is based on the work on Functional Representation [Sembugamoorthy 1986], and it is a further extension of the work presented in [Iwasaki 1992]. Bradshaw and Young [Bradshaw and Young 1991] also represent the intended function in a manner similar to Functional Representation, and Franke [Franke 1991] proposes a representation of an abstract pattern on behavior as the function of a design modification. These functional representations are inappropriate for part retrieval — they are geared towards representing the designed functions and not target functions. The specifications are too detailed, and presume some knowledge of the device that achieves the function. For part retrieval, one needs a specification that is much simpler and that does not presume knowledge of the name or the structure of the device, as we have proposed here.

4. Summary and Conclusion

In this paper, we have proposed techniques for addressing problems inherent in information brokering of engineering components. In particular, we have described

(1) A representation formalism for functions such that the user can specify the functional goals using ordinary vocabulary of the domain with a varying degree of specificity, and

(2) Techniques for relating functional specifications provided by the user to classes of components and component characteristics to allow parts retrieval on functions.

Our next task is to actually implement an information broker for parts. The implementation will use the general-purpose architecture being developed by the Information Brokers Project.

The applicability of the general scheme we have described in this paper is not necessarily limited to physical part retrieval. Software, including both pieces of code to be embedded in a physical device as well as various types of analysis programs needed during the design process, can also be broadly viewed as parts with intended functionality. Large engineering firms have an extensive library of simulation and analysis programs, which pose the same kind of retrieval problems as part catalogs. In the future, we intend to investigate an extension of this approach to software domains as well.

Bibliography

Baya, V. and Leifer, L. (1995). *Understanding Design Information Handling Behavior Using Time and Information Measure.* ASME Symposium on Design Theory and Methodology,

Bradshaw, J. A. and Young, R. M. (1991). Evaluation Design Using Knowledge of Purpose and Knowledge of Structure. IEEE Expert.

Collet, C., Huhns, M. and Shen, W. (1991). Resource Integration Using a Large Knowledge Base in Carnot. IEEE Computer. 55-62.

Court, A. W., Culley, S. J. and McMahon, C. A. (1993). *The Information Requirements of Engineering Designers.* the International Conferences on Engineering Design,

Crandall, D. (1993). A Query Server Agent for PartNet: A component Selection Design Aid. M.S. thesis. University of Utah.

Defense Logistics Agency. *Federal Item Identification Guide.* Defense Logistics Services Center, Battle Creek, MI:

Franke, D. W. (1991). Deriving and Using Descriptions of Purpose. IEEE Expert.

Iwasaki, Y., and Chandrasekaran, B. (1992). *Design verification through function and behavior-oriented representations: Bridging the gap between function and behavior.* The Second International Conference on Artificial Intelligence in Design., Pittsburgh, PA, Kluwer Academic Publishers.

Iwasaki, Y., Vescovi, M., Fikes, R. and Chandrasekaran, B. (1995). Causal Functional Representation Language with Behavior-Based Semantics. *Applied Artificial Intelligence* **9**(1): 5-31.

Keuneke, A. (1991). Device Representation: The Significance of Functional Knowledge. IEEE Expert. 22-25.

Knoblock, C., Arens, Y. and Hsu, C. (1994). *Cooperating Agents for Information Retrieval.* the Second International Conference on Cooperative Information Systems,

Leifer, L. (1995). Personal communication.

Lenat, D. and Guha, R. V. (1990). *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project.* Reading, Mass., Addison-Wesley.

Levy, A., Sagiv, Y. and Srivastava, D. (1994). *Towards Efficient Information Gathering Agents.* AAAI Spring Symposium on Intelligent Agents, Stanford, CA,

Sembugamoorthy, V., and Chandrasekaran, B. (1986). Functional Representation of Devices and Compilation of Diagnostic Problem-Solving Systems. *Experience, Memory, and Reasoning.* Hillsdale, NJ, B.Lawrence Erlbaum Associates.

Thomas Register of American Manufacturers and Thomas Register Catalog File. Thomas Publishing Company: New York, N. Y.

Rule Based Database Integration in HIPED: Heterogeneous Intelligent Processing in Engineering Design

Shamkant B. Navathe, Sameer Mahajan, and Edward Omiecinski

The summary for this presentation can be found on page 38
The slides for the presentation start after page 308
8 pages follow

Rule Based Database Integration in HIPED : Heterogeneous Intelligent Processing in Engineering Design

Shamkant B. Navathe Sameer Mahajan Edward Omiecinski

College of Computing,
Georgia Institute of Technology,
Atlanta, GA 30332-0280, USA.
{sham,sameer,edwardo}@cc.gatech.edu

Abstract

In this paper [1] we describe one aspect of our research in the project called HIPED, which addressed the problem of performing design of engineering devices by accessing heterogeneous databases. The front end of the HIPED system consisted of interactive KRITIK, a multimodal reasoning system that combined case based and model based reasoning to solve a design problem. This paper focuses on the backend processing where five types of queries received from the front end are evaluated by mapping them appropriately using the "facts" about the schemas of the underlying databases and "rules" that establish the correspondance among the data in these databases in terms of relationships such as equivalence, overlap and set containment. The uniqueness of our approach stems from the fact that the mapping process is very forgiving in that the query received from the front end is evaluated with respect to a large number of possibilities. These possibilities are encoded in the form of rules that consider various ways in which the tokens in the given query may match relation names, attrribute names, or values in the underlying tables. The approach has been implemented using CORAL deductive database system as the rule processing engine.

1 Introduction

Heterogeneity of databases is becoming a necessary factor to contend with in the design of new applications because of the proliferation of database management systems that used diverse data models over the last three decades. Among widely implemented data models we have the hierarchical, network, relational and object oriented data models. A large body of work exists that deals with the mapping of these models among one another (e.g. see the mapping of models using the entity relationship model as an intermediate model in [1] [3]. While vendors are also providing middleware solutions to draw data from these legacy systems, the semantic problems of resolving, naming, scale, structure etc. that were pointed out several years ago [5] [6] still remain. The purpose of the present research was to develop a technique to dealing with the semantic differences in data by taking a flexible rule based approach. Another goal of the project was to tie a set of heterogeneous databases to an "intelligent front end application" which would make requests for data without any knowledge of the schemas of the target databases. To limit the degree of difficulty we assume that we are dealing with data in relational databases only. This assumption is reasonable in the sense that of the data is coming from a hierarchical or a network DBMS, we can first convert the schema to a relational one before treating it for purposes of integration.

The database integration problem we discuss here is couched in the context of engineering design which, like any other design application, relies on extracting data from existing databases containing material data, components, existing designs etc. The exact context and the application scenario will be explained in the next section.

We assume that relevant data for the design application is stored in relations (tables) whose schemas are available at "design time" to construct a rule-base. It is conceivable that to support large scale engineering designs, data from a variety of databases, i.e., from multiple schemas would be required. To facilitate integration of data among these databases we assume that the "correspondances", i.e., the similarities and differences among the (meaning of) attributes is encoded in the form of rules. Furthermore, for our application context, the front end of HIPED issues certain queries looking for relevant design information. We show in this paper how a query may have several interpretations, each one of which is encoded in the form of rules again.

Because of these two kinds of rules involved in the integration approach we have termed our approach a rule based approach to database integration. The present approach is an improvement over previous approaches where we handled integration by using the correspondance information to derive the process [2] [6] [7] [8].

2 Application Context

In this section we will provide the overall architecture of the HIPED system and point out the need for heterogeneous database processing which will be de-

[1] To appear in the Proceedings of International Symposium on Cooperative Database Systems for Advanced Applications, Heian Shrine, Kyoto, Japan, World Scientific Press, 1996.

scribed and illustrated in the next two sections.

2.1 Overall Architecture of HIPED

Our main objective in the HIPED project is to build and demonstrate an intelligent interface to a set of (possibly autonomous) information sources including structured databases, knowledge bases, and unstructured data. The approach we have selected involves the development of a mediator which utilizes meta-knowledge of the underlying information stores to aid a user in browsing data or to enable an application front-end to retrieve specific relevant information for problem solving.

The overall architecture of HIPED is described in Figure 1. The data is organized at two levels namely, (1) the metadata repository : consisting of information about various databases and tables in them and (2) the actual data which is distributed in various heterogeneous databases. This organization reduces the data to be dealt with at the first level to get to the appropriate database(s) and table(s). It also allows heterogeneity in the various databases involved. The Querying Interface is as described in section 3.1. The "data" together with its "wrapper" forms a database system. "Wrapper" simply defines the access methods to the data for reading purposes. A wrapper can be designed for each target database management system. A user query would be translated into the corresponding query, as understood by the corresponding "wrapper", for each of the relevant tables. This query would then be routed to the corresponding database, that contains this table. The metadata repository is consulted in determining these relevant tables and finding the corresponding database. The user would get the result, obtained after running the query against the table through the concerned "Output Data" channel(s).

2.2 Interactive KRITIK Front End

We developed the HIPED architecture by assuming a frontend system called Interactive Kritik [4]. This system is a multimode reasoning system which works like a design assistant for the design of devices such as acid coolers, electrical devices. In its current form the system uses "hard-wired" knowledge in the form of LISP data structures. The goal was to extend the capability of interactive Kritik to make it scalable to real-life design problems by incorporating databases of relevant design data as the back end. We therefore abstracted different forms of generic query types which would be used as requests to the back end. By coupling an intelligent front end application to a set of heterogeneous databases, we can thus extend the scope of problem solving by a large measure. For engineering device design, the above front end generates a number of requests for data from the underlying design databases such as design prototypes, properties of devices and components, material data, design specifications and tolerances etc. For illustrative purposes we have chosen five generic types of queries that are most commonly presented by the front end. They will be explained in detail in the following section.

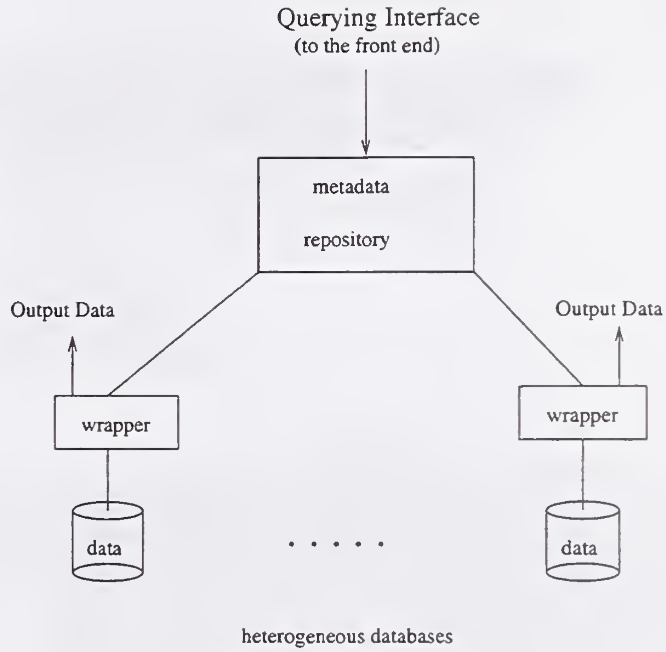


Figure 1: The High Level View of HIPED back end

3 Rule Based Approach to Database Integration

As explained earlier the main contribution of this research is the use of the two types of rules to accomplish access to the underlying heterogeneous information sources. The first set of rules deals with establishing various types of relationships among relation names and among attribute names across databases. The second set deals with the interpretation of queries from the front end so that various possible mappings to the interface of underlying target databases may be considered. We will explain both these types of rules when we discuss the generic queries and their mappings.

3.1 Five generic types of queries

The user is assumed to use this system as an Engineering Database for device design. Let us limit the application domain for illustrative purposes. We assume that during the design process, he would typically like to find components that satisfy his requirements (e.g. batteries with voltage rating higher than 10V and cheaper than $10). Keeping this user's perspective in mind, the Engineering data is thought to be made up of various "Prototypes". Each Prototype has various "Properties". Each Property takes up some "Value" for every Prototype. We can compare the Values of various properties using the relations : $==, <, >, <=, >=, <>$ etc. The queries can be classified into the following five generic types,

1. (Prototype <proto_name>) : here the user is looking for all the prototypes identified by "proto_name". It is implicit that the user wants

to see the various values for various properties (attributes) of these prototypes.

2. (Property <prop_name>) : the user is interested in all the prototypes having the specific Property identified by "prop_name". It is implicit that the user wants to see the values taken by this property for the various prototypes, that would be listed.

3. (Prototype <proto_name>)
(Property <prop_name>) : the user wants to see all the prototypes identified by "proto_name" and having property identified by "prop_name". It is implicit that the user also wants to see the corresponding value that the property takes for the particular prototype.

4. (Prototype <proto_name>)
(Property <prop_name>)
(Value<value>) (Rel-op <op>) : the user is interested in prototypes identified by "proto_name" having a property identified by "prop_name". In addition to this he wants only those prototypes for which the property takes a value which is related to the given "value" or a constant in the query by the operator "op" (i.e. it is equal to "value" or greater than "value" etc.)

5. (Property <prop_name>) (Value <value>)
(Rel-op <op>) : the user is interested in all the prototypes for which the property identified by "prop_name" takes a value which is related to the given "value" by the operator "op.

Data is distributed among various databases and various tables in each of those databases. The only assumption that we make about any database system is that it has an SQL access method. It is a reasonable assumption and is made to contain the complexity of the problem.

The system needs to find out which databases and which tables in these databases have the relevant data to answer a particular query. It then translates the query into a corresponding SQL query for every table. This SQL query is run against that table to get an answer. As we made an assumption of a uniform SQL interface to all the databases, we can simply translate a request for data into a set of SQL queries in each of these cases.

3.2 Rules for Interpretation of Queries

For better understanding of the following discussion, let us take up an example query. Let the four components of the query be,
(Prototype Battery) (Property Voltage)
(Value 10) (Relation ==).
As there can be various tables with different schema, we need to run this query with only those tables that might give meaningful results for the query. We can easily observe that any of "Prototype", "Battery", "Property" and "Voltage" can be a table or a column of a table. The "Battery" and "Voltage" can also be values in the columns (e.g. those labeled as "Prototype" and "Property" respectively). Of course there are a lot of dependencies amongst these components - e.g. if "Prototype" is a table then "Battery" has to be a column of this table. On the other hand if there is a table called the "Battery", then we are looking for values in the column "voltage" or "volts" - so that the query would generate meaningful results with the table. Now we take up an example query for each of the five types listed above. For every query we list the possible interpretations according to our scheme.

1. (Prototype Battery). The user typically means that he wants all the batteries with their properties and their corresponding values. Hence we will have to run this query against all the tables which,

   - are equivalent to "Prototype Table" and have a column equivalent to "Battery" or
   - are equivalent to "Battery Table"
   - have a column equivalent to "Prototype" (and only the tuples with Prototype as "Battery" would be considered).

   if and only if these tables have columns equivalent to "Property" and "Value" each.

2. (Property Voltage). The user is interested in listing all the Prototypes having "Voltage" as their one of the Properties. The Values of these Properties would also be significant from his standpoint. Hence we consider all the tables which,

   - are equivalent to "Property Table" and have a column equivalent to "Voltage" or
   - are equivalent to "Voltage Table"
   - have a column equivalent to "Property" (and only the tuples with Property as "Voltage" would be considered).

   if and only if they have "Prototype" equivalent column.

3. (Prototype Battery) (Property Voltage). The user wants all the batteries with special interest in their voltages. Hence we will run the query against all the tables which,

   - are equivalent to "Prototype Table" and have "Battery", "Property" and "Value" equivalent columns and we would be interested only in the tuples having an entry of "Voltage" in the "Property" equivalent column or
   - are equivalent to "Prototype Table" and have "Battery", "Voltage" equivalent columns or
   - are equivalent to "Battery Table" and have "Property" and "Value" equivalent columns. We would be interested only in those tuples having Property "Voltage" or
   - are equivalent to "Battery Table" and have a column equivalent to "Voltage" or

- are equivalent to "Property Table" and have columns equivalent to "Voltage", "Prototype" and "Value". We would be interested in tuples with Prototype as "Battery".
- are equivalent to "Property Table" and have "Voltage" and "Battery" equivalent columns.
- are equivalent to "Voltage Tables" and have "Prototype" and "Value" equivalent columns. We would look for only tuples with Prototype as "Battery".
- are equivalent to "Voltage Table" with "Battery" and "Value" equivalent columns.
- have "Prototype" and "Property" equivalent columns as far as they have "Value" equivalent column. Only the tuples with Prototype as "Battery" and Property as "Voltage" would be considered.

4. (Prototype Battery) (Property Voltage) (Value 10) (Relation ==). Here the interest is indicated in all the batteries having Voltage as "10". The query can be run with all the tables as indicated as above with an added constraint that only those tuples which have an entry of "10" in the "Voltage" or "Value" column - whichever is applicable - (Note the table can have only one of these columns at a time) will be considered.

5. (Property Voltage) (Value 10) (Relation ==). All the Prototypes having voltage of "10" are being considered. Thus all the tables that,

    - are equivalent to "Property Table" and have a column equivalent to "Voltage"
    - are equivalent to "Voltage Table" and have a column equivalent to "Value"
    - have "Property" and "Value" equivalent columns along with "Prototype" column. (only tuples with Property "Voltage" and Value "10" would be taken into consideration).

    would be considered if and only if they have a column equivalent to "Prototype". All the tuples with "Voltage" or "Value" being 10 would be taken into account.

3.3 Rules to establish Data Correspondance

We need to relate various attributes and tables, within and across databases. The relationship could be of equivalence, subsumption, overlap, disjointness or containment. The relationship between attributes needs to be supplied by the schema developer. e.g. Attributes called "volt" and "voltage" in different tables are actually equivalent. The relationship between tables can either be supplied or can be deduced by the relationships of their individual attributes. A simple deduction rule can be that two tables are equivalent if all their attributes are equivalent.

4 Use of CORAL for rule representation and query processing

The metadata is stored in the form of CORAL [10] [11] facts and rules. CORAL is a deductive database system which stores data as facts and rules, and allows for that data to be queried. By using CORAL the mediator can decide which database(s) and table(s) are useful in answering any given query. In particular, CORAL is used in deriving relationships like equivalence; between attributes, tables and databases. Any creation, deletion or modification of a table results in a change in the metadata repository. This dynamic behavior can be easily captured by CORAL. In essence, CORAL provides us with the facility for database integration through the facts and rules specified about tables and databases. However, this integration can be considered implicit rather than explicit since no global conceptual schema is explicitly formed. Also the C++ interface provided by CORAL makes writing general purpose programs easy.

We explain the implementation with the help of an example. One more sample system for a single database environment is given in Table 5. Some sample input queries and the corresponding output SQL queries are shown in Tables 6 and 7 respectively.

4.1 A Simple Example

Consider the query,

```
(Prototype Battery) (Property Voltage).
```

Let us assume that there are two databases - db1 and db2. Let db1 have tables : Table 1 and Table 2. and

CompNo	Prototype	Property	Value
B101	Battery	Voltage	10
M101	Motor	Voltage	10
B110	Battery	Voltage	100
B111	Battery	Current	100
	⋮		

Table 1: "Components" Table in db1

let db2 have Table 3 and Table 4.

We observe that according to the discussion in section 3.2 only the tables in db1 would produce meaningful results with the query under consideration.

BatteryNo	Voltage
B101	15
B102	30
⋮	

Table 2: "Battery" Table in db1

BatteryNo	Current
B101	15
B102	30
⋮	

Table 3: "Battery" Table in db2

BatteryNo	Supplier No
B101	4567
B102	4568
⋮	

Table 4: "Supplier" Table in db2

4.2 Schema Representation

It is stored as CORAL facts and rules. The advantage of such a storage is that we can utilize the strong deductive power of CORAL (e.g. deducing equivalence of attributes, equivalence of tables etc.). The various components of the repository are described below.

- First we list all the tables in all the databases as CORAL facts :

```
% For the first database, db1.
belongsTo(components,db1).
belongsTo(battery,db1).

% For the second database, db2.
belongsTo(battery,db2).
belongsTo(supplier,db2).
```

- Then we list attributes of individual tables as CORAL facts. The first argument of these predicates is the database name. It is so because the same table may have different attributes in different databases. e.g. the "battery" table in the two databases "db1" and "db2" as shown below.

```
% for db1
hasAttribs(db1,components,
   [compName,prototype,property,value]).
hasAttribs(db1,battery,[bName,voltage]).

% for db2
hasAttribs(db2,battery,[bName,current]).
hasAttribs(db2,supplier,[bName,sName]).
```

- We also need facts to list what attributes are equivalent. The equivalence of tables can be either given by facts or can be deduced by the rules (e.g. two tables with equivalent attributes are equivalent). But we do not need them in this particular example.

- To find whether a table has a particular attribute in a given database we define a CORAL rule as,

```
module isAttrib.
export isAttrib(bff).
isAttrib (Db,Table,Attri) :-
        hasAttribs(Db,Table,Attribs),
        iselem (Attri,Attribs).
end_module.

% Module ''iselem'' is defined for the
% sake of completeness.
module iselem.
export iselem(bb).
@pipelining+.    % Solve in a
                 % top-down fashion

iselem(X, [X|_]).
iselem(X, [_|Z]) :- iselem(X,Z).
end_module.
```

4.3 Sample Query Mapping Algorithm

The mapping of input requests into SQL queries is done according to the scheme suggested in section 3.2. We use the C++ interface of CORAL for this matter. In fact, an imperative interface (e.g. in C) would have been enough for the purpose. We check for the various conditions given in the scheme and generate the appropriate SQL queries for the existing tables. We run through the algorithm for the example query under consideration,

```
begin
For every ''table'' equivalent to
''prototype table''
  for every attribute equivalent
  to ''battery'', say attrib1
    for every attribute equivalent
    to ''property, say attrib2
      if ''table'' has ''attrib1''
      as well as ''attrib2''
        for every attribute
        equivalent to ''value'',
        say attrib3
          if ''table'' has attrib3
            select the corresponding
            database and fire SQL query,
            SELECT * FROM table
            WHERE attrib2 == voltage or
                  some equivalent value.
            goto next table
        for every attribute equivalent
        to ''voltage'', say attrib4
          if ''table'' has attrib4
            select the corresponding
            database and fire SQL query,
            SELECT * FROM table
```

```
For every ''table'' equivalent to
''battery table''
  for every attribute equivalent
  to ''voltage'', say attrib1
    if ''table'' has attrib1
        select the corresponding
        database and fire SQL query,
        SELECT * FROM table
        goto next table
  for every attribute equivalent
  to ''property'', say attrib2
    for every attribute equivalent
    to ''value'', say attrib3
      if ''table'' has attrib2
      and attrib3
        select the corresponding
        database and fire SQL query,
        SELECT * FROM table
        WHERE attrib2 == voltage or
              some equivalent value.

For every ''table'' equivalent
to ''property table''
  for every attribute equivalent
  to ''voltage, say attrib1
    for every attribute equivalent
    to ''prototype'', say attrib2
      if ''table'' has ''attrib1''
      as well as ''attrib2''
        for every attribute
        equivalent to ''value'',
        say attrib3
          if ''table'' has attrib3
            select the corresponding
            database and fire SQL query,
            SELECT * FROM table
            WHERE attrib2 == battery or
                  some equivalent value.
            goto next table
        for every attribute equivalent
        to ''battery, say attrib4
          if ''table'' has attrib4
            select the corresponding
            database and fire SQL query,
            SELECT * FROM table

For every table equivalent to
''voltage table''
  for every attribute equivalent
  to ''battery'', say attrib1
    if ''table'' has attrib1
      select the corresponding
      database and fire SQL query,
      SELECT * FROM table
      goto next table
  for every attribute equivalent
  to ''prototype'', say attrib2
    for every attribute equivalent
    to ''value'', say attrib3
      if ''table'' has attrib2
      and attrib3
        select the corresponding
        database and fire SQL query,
        SELECT * FROM table
        WHERE attrib2 == battery or
              some equivalent value.

For every table having columns
equivalent to each of
prototype, property and value
  select the corresponding
  database and fire SQL query
  SELECT * FROM table
  WHERE prototype equivalent column
          == battery equivalent value
        AND
        property equivalent column
          == voltage equivalent value
end
```

4.4 The Result

Let us say that the wrapper of db1 can handle SQL queries. In that case we first select that database and then simply run a query,

```
SELECT *
FROM components
WHERE prototype == ''battery''
        AND property == ''voltage''
```

against the first ("components") table in the database. We take similar actions for the other table in (possibly various) databases. The other query in this case would be,

```
SELECT *
FROM battery
```

again with the same database namely, db1. The result is presented to the user as displayed by the corresponding "wrapper".

5 Conclusions and Future Work

In this paper we illustrated the implementation of a rule-based database integration scheme by considering two types of rules : (1) Rules to establish the "correspondence" among underlying component databases and (2) Rules to interpret data requests in an "open-ended" fashion where no knowledge of the component database schemas is expected from the application front end. We also described an interface to heterogeneous databases in which a user may directly access the back end data by making use of the rules of data correspondance and an SQL-like syntax for the queries.

The system makes an assumption that all the databases involved provide an SQL interface. This condition can be relaxed. In this case we need to generate different queries, as understood by each of the databases involved. This work was predicated on the assumption that the data relevant to our application was stored in relational tables. An extension of the present work involves relaxing this assumption and illustrating the utility of the approach by actually providing wrappers for hierarchical and network databases and sequential files. That would establish

```
% CORAL facts
isTable(battery).
hasAttribs(battery,
   [bname,voltage,current,life]).

isTable(compTable).
hasAttribs(compTable,
   [no,prototype,property,value]).

isTable(dummy).
hasAttrib(dummy,[prototype,property]).

isTable(prototype).
hasAttribs(prototype,
   [motor,property,value]).

isTable(motor).
hasAttribs(motor,[property,value]).

isTable(property).
hasAttribs(property,
              [rps,prototype,value]).

isTable(rps).
hasAttribs(rps,[prototype,value]).

isTable(voltage).
hasAttribs(voltage,[battery,value]).

% CORAL rules

module isAttrib.
export isAttrib(ff).
isAttrib (X,Y) :- hasAttribs(X,Z),
                  iselem (Y,Z).
end_module.
```

Table 5: A Single Database System

```
prototype battery property voltage
prototype battery property current
prototype motor property rps
prototype sheet property size
```

Table 6: Sample Input Queries

```
++++++++ for the first data request ++++++
SELECT * FROM battery;
SELECT * FROM voltage;
SELECT * FROM compTable
WHERE prototype == battery
        AND property == voltage;
++++++++ for the second data request +++++
SELECT * FROM battery;
SELECT * FROM compTable
WHERE prototype == battery
        AND property == current;
++++++++ for the third data request ++++++
SELECT * FROM prototype
WHERE property == rps;
SELECT * FROM motor
WHERE property == rps;
SELECT * FROM property
WHERE prototype == motor;
SELECT * FROM rps
WHERE prototype == motor;
SELECT * FROM compTable
WHERE prototype == motor
        AND property == rps;
+++++++ for the fourth data request +++++
SELECT * FROM compTable
WHERE prototype == sheet
        AND property == size;
```

Table 7: The corresponding SQL queries

the practical utility of the approach in a significant way. The next step would be to work on a query optimization by introducing a stage after the query interpretation phase to evaluate possible orderings of sub queries and cross subquery reduction of redundant processing.

From the engineering design standpoint, the problem horizon can be extended to include additional types of design problems. The current implementation can be initially enhanced by considering additional types of design queries.

Currently only the individual tables are checked to see whether they provide satisfactory data to answer a particular query. But it is possible that two or more tables taken separately do not have enough information to answer a query. At the same time, when taken together (e.g. their join), they provide data to answer the query. Consider that there are two tables - which might be in the same database or in different databases - one with columns "Component Number" and "Prototype". The other with columns "Component Number" and "Voltage". Then neither of them provides enough information for the query,

(Prototype Battery) (Property Voltage)

But their equijoin with the additional condition of "Prototype == Battery" for the tuples is of interest to us. The extended solution can exhaustively take care of all such cases.

In essence, the overall rule based approach appears promising in the context of Navathe's long standing investigation of the database integration problem [5] [6] [7] [8] [9].

Acknowledgement

We would like to thank Ashok Goel and William Murdock for their work on Interactive KRITIK. The work crystallized the HIPED front end to our system. We are also grateful to Ashok Savasere and Byong-soo Jeong for initializing the implementation work. The idea of using CORAL as a deductive database system was generated by Ed Omiecinski. The implementation was completed by Sameer Mahajan. Support from ARPA grant no. F33615-93-1-1338 is greatly appreciated.

References

[1] C. Batini, S. Ceri, and S.B. Navathe, **Conceptual Database Design: An Entity Relationship Approach**, Benjamin Cummings, August 1991, 470 pp.

[2] C. Batini, M. Lenzerini and S.B. Navathe, A Comparative Analysis of Methodologies for Database Schema Integration, *ACM Computing Surveys*, 18, 4, December 1986, pp. 323-364.

[3] R. Elmasri, S Navathe, **Fundamentals of Database Systems**, Addision Wesley Computer Science, 2nd Edition, 1994.

[4] Ashok Goel, Andres Gomez and Nathalie Grue, Task-Method Structures and Design Cases. *Proceedings of the AI in Design Workshop on Case-Based Design Systems, (Mary Maher and Pearl Pu: editors)*, August 1994, Lausanne, Switzerland.

[5] J. Larson, S. B. Navathe, and R. Elmasri A Theory of Attribute Equivalence in Databases with Application to Schema Integration, *IEEE Transactions on Software Engineering*, Vol. 15, No. 4, April 1989.

[6] S.B. Navathe, R. Elmasri and J.A. Larson, Integrating User Views in Database Design, *IEEE Computer*, Vol. 19, No. 1, January 1986, pp. 50-62.

[7] S. B. Navathe and A. Savasere, "A Practical Schema Integration Facility using an Object Oriented Approach," *Multidatabase Systems (A. Elmagarmid and O.Bukhres, Eds.)*, Prentice Hall, 1996.

[8] S. Prabhakar, J. Srivastava, S.B. Navathe, et al., Federated Autonomous Databases: Project Overview, *Proceedings of the International Conference on Interoperability in Multidatabase Systems (IMS'93)*, Vienna, Austria, April 19-20, 1993.

[9] A. Sheth, S.K. Gala, S.B. Navathe, *On Automatic Reasoning for Schema Integration, Int. Journal of Intelligent Co-operative Information Systems*, Vol. 2, No.1, March 1993.

[10] R. Ramakrishnan, D. Srivastava and S. Sudarshan, CORAL: Control, Relations and Logic, *Proceedings of the International Conference on Very Large Databases*, 1992.

[11] R. Ramakrishnan, D. Srivastava, S. Sudarshan and P. Seshadri, Implementation of the CORAL deductive database system, *Proceedings of the ACM SIGMOD Conference on Management of Data*, 1993.