Contract MDA972-88-C-0047

AD-A219 638

DARPA INITIATIVE IN CONCURRENT ENGINEERING (DICE)

FINAL TECHNICAL REPORT
PHASE 1
(July 1, 1988 - September 30, 1989)

Prepared by:
Kamar J. Singh
Program Manager

Sponsored by:
Defense Advanced Projects Agency
Defense Sciences Office
DARPA Initiative in Concurrent Engineering
ARPA Order No. 6511
Issued by DARPA/CMO under Contract No. MDA972-88-C-0047

GE Aircraft Engines

Concurrent Engineering Programs
Cincinnati, Ohio 45215-6301

Contract MDA972-88-C-0047

DARPA INITIATIVE IN CONCURRENT ENGINEERING (DICE)

QUARTERLY R&D STATUS REPORT NO. 3

For the Period
1 January 1989 Through 31 March 1989

This Document Contains
Missing Page/s That Are
Unavailable In The
Original Document

OR are
Blank pgs
that have
Been Removed

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: Unclassified	1b. RESTRICTIVE MARKINGS: None
2a. SECURITY CLASSIFICATION AUTHORITY: -----	3. DISTRIBUTION/AVAILABILITY OF REPORT: Unrestricted
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE: -----	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION: GE Aircraft Engines 6b. OFFICE SYMBOL (If applicable): GEAE	7a. NAME OF MONITORING ORGANIZATION: Defense Advanced Research Projects Agency
6c. ADDRESS (City, State, and ZIP Code): 1 Neumann Way, Cincinnati, Ohio 45215-6303	7b. ADDRESS (City, State and ZIP Code): 1400 Wilson Blvd., Arlington, VA 22209-2308
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: ----- 8b. OFFICE SYMBOL (If applicable): DARPA	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: MDA972-88-C-0047
8c. ADDRESS (City, State and ZIP Code)	10. SOURCE OF FUNDING NOS.: PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.

11. TITLE (Include Security Classification): DARPA Initiative in Concurrent Engineering (DICE) - Phase 1 (Unclassified)

12. PERSONAL AUTHOR(S): Singh, Kamar J.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Final	FROM 1988,07,01 To: 1989,09,30	1990,02,09	

16. SUPPLEMENTARY NOTATION: Prepared in cooperation with: GE-CRD, West Virginia University, Carnegie Mellon University, Resselaer Polytechnic Institute, NC State University, Cimflex Teknowledge,

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	Concurrent Engineering; CE; Computer Architecture; CE Research Center; CALS; TQM

19. ABSTRACT (Continue on reverse if necessary by block number)

Abstract:

The mission of the DICE program is to create a Concurrent Engineering environment that will result in reduced time to market, improved total quality and lower cost for products or systems developed and supported by large organizations. This environment will enable all disciplines important in life cycle of a product or system to cooperate interactively in its definition, planning, design, manufacture, maintainence, refinement and retirement from service.

The DICE Concurrent Engineering environment will emulate for large organizations the human tiger-team approach to concurrent engineering successfully employed by small groups.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION: Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL: Larry Pendleton, Jr.	22b. TELEPHONE NUMBER (Include Area Code): 202-694-1771	22c. OFFICE SYMBOL: DARPA

DD Form 1473, JUN 86 EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

16. Supplementary Notation (Continued)

Howmet Corporation, Martin Marietta Labs, Textron Specialty Materials, University of California at Santa Barbara

19. Abstract (Continued)

This environment will leverage prior experience; it will emphasize early, high-quality decisions; and it will support the propagation of requirements, the feedback of constraints, and the efficient management of risk and change as the product or system is progressively refined from concept to realization.

The DICE mission is to provide an open, computer-assisted environment that implements this concurrent engineering vision. The DICE environment will consist of i) a shared information model that captures complete descriptions of the product or system and all associated process activities and organizational resources; ii) a global object framework that enables the use of the shared information model by a network of cooperating, computer-based clients; and iii) services, methods, tools and advisors that assist concept evaluation, analysis and intelligent decision making.

The DICE mission encompasses a series of rapidly-prototyped demonstrations involving high-valued, complex products in multiple domains and focussing research and development on the validation of the design, performance and scalability of the evolving DICE environment.

The DICE mission will be fulfilled through the establishment of a Concurrent Engineering Research Center to promote education, training, and research in concurrent engineering, to facilitate the transfer of DICE technology to U.S. industrial users and suppliers, and to support the implementation of national initiatives related to concurrent engineering.

The mission of DICE program is to create a Concurrent Engineering environment that will result in reduced time-to-market, improved quality and lower costs for products or systems developed and supported by large organizations. The environment will enable all disciplines important in the life cycle or product or system to cooperate interactively in its definition, planning, design, manufacture, maintenance, refinement, and retirement from service. The DICE environment will emulate for large organizations the human "tiger-team" approach to concurrent engineering successfully employed by small groups.

INDEX

PAGE

1.0 Program Overview

Present government and industrial programs are conducted with a sequential process moving from research and development of materials and processes to design and then to manufacturing of components and assemblies. This serial approach has resulted in very long periods of time to introduce new technology, often exceeding beyond 20 years. In addition to the extensive development time required, the serial approach results in insufficient design iterations that lock in suboptimal design, late discovery of problems, and relatively low productivity. The constraints and requirements of downstream disciplines are not accounted for in the early stages of the design cycle leading to many avoidable time-consuming problems and delays. These problems translate into long lead time to production.

Concurrent Engineering is a revolutionary approach to simultaneously conduct research and development, design, and manufacturing of components and assemblies resulting in a relatively short introduction time for new and advanced high technology materials and processes. It will be possible to include the downstream constraints and requirements at the conceptual stages. By exploiting the parallel processing methodology now common in the field of computer science, concurrent engineering promises to reduce the introduction time for advanced systems from concept to actual production by one-third to one-half.

DARPA Initiative in Concurrent Engineering (DICE) program is the first attempt of its kind to make available concurrent engineering technology to simultaneously conduct research and development, design, and manufacturing in the areas of structural components and electronics. This novel approach will permit examination of multiple options quickly to achieve optimum design and provide active updates to keep all disciplines aware of any changes/modifications. Consequently, options to make changes will be left open to a much later stage in the design cycle.

The DICE program will provide Architecture; Methods, Tools and Advisors; the Manufacturing Demonstrations; a University Research Center; and means to transfer technology to industry to achieve the benefits of concurrent engineering.

To achieve the goals for concurrent engineering, a university/industry consortium has been formed to research concurrent engineering issues. This consortium, under the program management of General Electric Aircraft Engines (GEAE), will demonstrate and validate concurrent engineering tools for:

- critical parts/structures from advanced materials
- advanced electronic assemblies

Within the scope of the program, a Concurrent Engineering Research Center (CERC) will be established at West Virginia University, Morgantown, WV. The research results will be "showcased" at CERC and transferred to the industry through a comprehensive technology transfer programs.

CERC will conduct symposia and hold workshops in concurrent engineering to dissipate the technology developed/integrated under the DICE program.

Phase 1 of the DICE program commenced on July 1, 1988 and finished on September 30, 1989 for a total of 15 months. The following participants were involved in Phase 1.

• GE Aircraft Engines	GEAE
• West Virginia University	WVU
• Cimflex Teknowledge Corp.	Cimflex
• GE Corporate Research and Development	GE-CRD
• Carnegie Mellon University	CMU
• Martin Marietta Laboratories	MML
• Howmet Corporation	Howmet
• Rensselaer Polytechnic Institute	RPI
• University of California at Santa Barbara	UCSB
• North Carolina State University	NCSU

2.0 Summary of Tasks

The four tasks being pursued under the DICE program are:

- **Task 1 -** Concurrent Engineering Research Center (CERC)

  A Research Center is being established at West Virginia University to teach/train the discipline of concurrent engineering. It will be a "showcase" of CE technology for the nation. The research carried out under the DICE program will be integrated, validated and demonstrated at CERC for further transfer to the U.S. industry.

- **Task 2 -** Architecture for CE

  A generic computer architecture will be developed for design and manufacture of both structural components and electronic assemblies. It will be an innovative architecture based on parallel information/computing concepts incorporating existing (both VMS and UNIX) software and hardware "point solutions" in a manner that is relatively transparent to end users.

- **Task 3 -** Methods, Tools & Advisors for CE

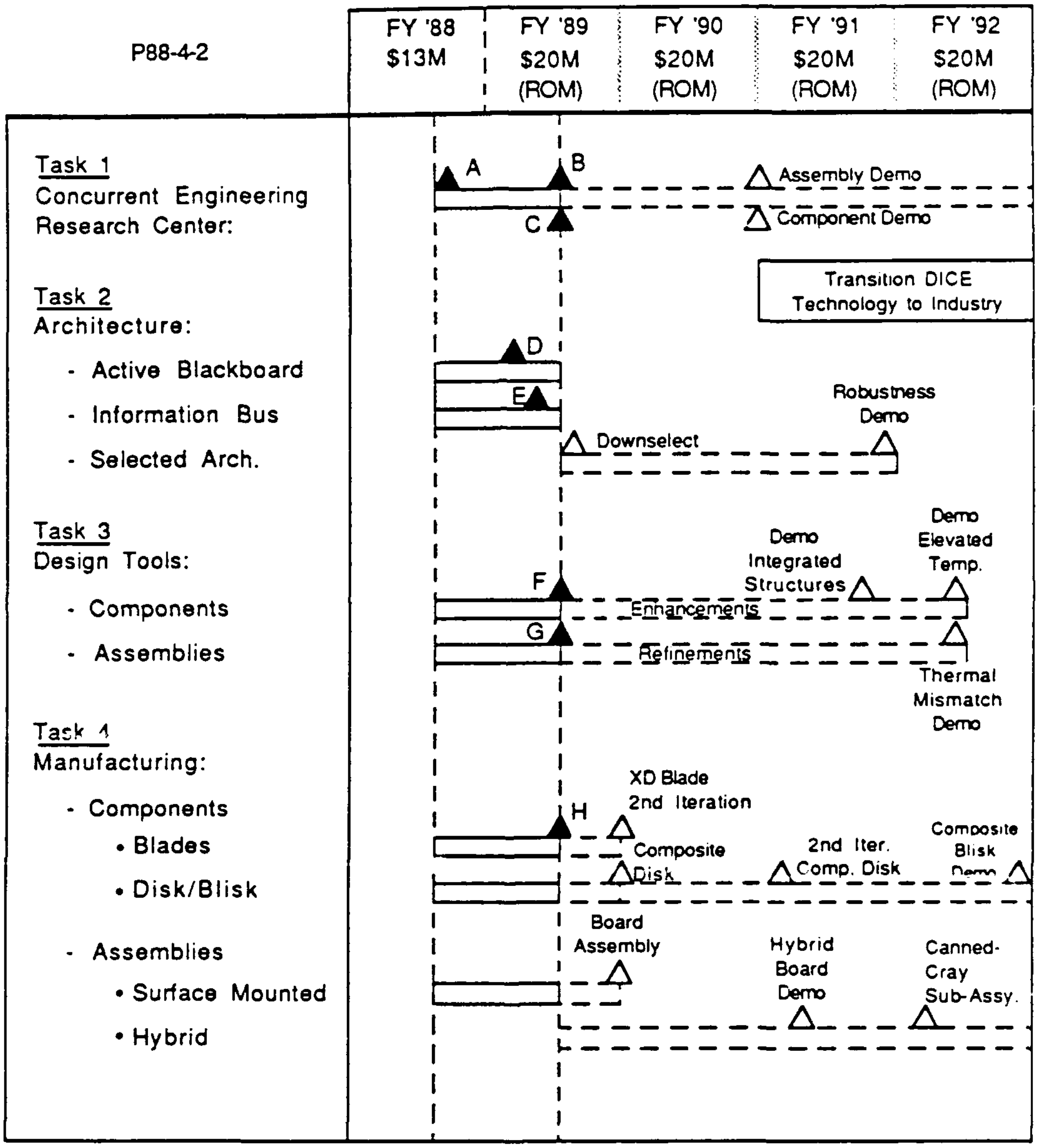
  New design methods, tools and advisors will be developed and along with those that already exist will then be integrated for both structural components and electronic assemblies. These tools will be domain-dependent having "plug in" option with the architecture. As a demonstration, design tools will be developed for new advanced composite materials and advanced electronic components.

- **Task 4** - Demonstrations of CE

  Concurrent Engineering technology will be demonstrated by manufacturing structural components from composite materials and electronic assemblies utilizing DICE architecture and design tools developed/integrated in the program. These demonstrations will move from simple to complex parts as the architecture and design tools become more sophisticated during this program.

A detailed Work Breakdown Structure of the program is as follows:

DICE Milestones/Schedule
Phase 1



A = Research Center Definition (Equipment, Materials, Parts)

B = Initial Sub-Assembly Demo

C = Initial Pseudo-Part Demo

D = Active Blackboard Demo

E = Information Bus Demo

F = Demo for Component Design Tools

G = Demo for Assemblies Design Tool

H = Demo of XD Blade-1st Iteration

3.0 Technical Accomplishments

This section contains the description of technical progress accomplished during Phase 1 of the DICE program, sequenced according to the WBS numbers.

Since there are a number of participants involved in the program and their effort is not integrated at this phase of the program, this report is compiled according to the WBS numbers rather than according to technical effort. However, all the technologies are integrated at the time of the annual demonstration for the customer (Phase 1 demo was held on July 25, 1989). These demonstrations are a means of testing and validating the computer architecture and methods being developed under the DICE program and are used to show reviewable progress.

PHASE 1

Work Breakdown Structure

WBS	Description	Participant
3.2	**CERC:**	
3.2.1.1	Capital Equipment	
3.2.1.1	Computing Environment	WVU
3.2.1.2	Assembly Cell	Cimflex
3.2.1.3	Materials Characterization Lab	WVU
3.2.2	CERC Administration & Technical Services	WVU
3.2.3	Curriculum in CE	WVU
3.2.4	Faculty/Graduate Students	WVU
3.3	**ARCHITECTURE:**	
3.3.1	Information Content and Flow	GEAE
3.3.2	Data Representation	
3.3.2.1	Design Knowledge Representation	CMU
3.3.2.2	Constraint-Based Models	CMU
3.3.2.3	Information Management Data Base	RPI
3.3.2.4	Graphics Interface & X-Windows	RPI
3.3.2.5	Architecture Definition	WVU
3.3.3	Information Architecture Prototype	
3.3.3.1	VMS Thread Components	CRD
3.3.3.2	UNIX Thread Components	CRD
3.3.3.3	Workstation Node Prototype	CRD
3.3.3.4	Fileserver Node Prototype	CRD
3.3.3.5	Info. Management Node	CRD
3.3.3.6	Integrated Operating System Thread	CRD
3.3.3.7	PaLS	NCSU
3.3.3.8	Local Architecture	CMU
3.3.3.9	File/Database Translator Gen.	RPI
3.3.3.10	Knowledge Server	WVU
3.3.3.11	Information Modeling	WVU
3.3.4	Hardware Study	WVU
3.3.5	Integration and Implementation	
3.3.5.1	Integration at CERC	CRD
3.3.5.2	Integration at GEAE	CRD
3.3.5.3	Integration of PaLS	NCSU
3.3.5.4	User Interfaces & Graphics Support Systems	WVU
3.3.5.5	Operating System Interfaces	WVU

3.4 DESIGN TOOLS:

3.4.1	Enhanced CAD Tools	
3.4.1.1	Process Planning Advisor	WVU
3.4.1.2	Geometric Modeling	WVU
3.4.3	Macro Models	GEAE
3.4.4	Microscopic/Micromechanics Models	
3.4.4.2	Mechanistics Models	WVU
3.4.5	Material Properties	
3.4.5.1	XD Material	MML
3.4.5.2	RSPD Material Properties	GEAE CRD
3.4.5.3	Material Properties & Reasoning	CMU
3.4.5.4	Material Characterization	WVU
3.4.6	Design Rules	GEAE
3.4.7	Design for Assembly	
3.4.7.1	High Density PCBs	Cimflex
3.4.7.2	Design for Assembly Advisor	WVU
3.4.8	Cost Models	
3.4.8.1	LCC Analysis Model	GEAE
3.4.8.2	Cost Modeler	WVU
3.4.9	Optimization Methods	
3.4.9.1	Engineous Software	CRD
3.4.9.2	Optimization of Composites	CRD

3.5 DEMONSRATION OF CE:

3.5.4	Component Manufacturing Demonstration	
3.5.4.1	Concurrent Mfg. of XD Airfoils	Howmet
3.5.4.2	Plasma Spray Disk	GEAE
3.5.5	Component Testing	GEAE
3.5.8	NDE/QA	
3.5.8.1	NDE/QA	GEAE
3.5.8.2	Quality Control Advisor	WVU

DICE PROGRAM

Task 3.2.1.1 Computing Environment West Virginia University

A. **Objectives:**

The objective of this task was to design, purchase and install the computing environment for the Concurrent Engineering Research Center (CERC). The CERC computing environment is to facilitate the development of software at West Virginia University and to provide a demonstration site for the software developed in the DICE project.

B. **Approach:**

The criteria for selecting the equipment and software in the CERC computing environment included:

- discussions with other DICE participants;
- discussions with the task leaders;
- requirements for requested software;
- budget considerations, and
- discussions with vendors.

C. **Technical Results:**

It was determined that CERC requires a multi-vendor computing environment to:

- support the wide variety of proposed software;
- respond to user requirements, and

- allow for the development of software that will run on a variety of workstations and machines found in today's technical computing environment.

D. Conclusions:

The CERC computing environment for Phase I of the project was adequate for most users. There were some needs not satisfied in the initial configuration and the increasing use of the system resulted in some additional needed requirements:

- disk storage;
- software;
- workstations;
- local disk space on workstations, and
- output devices.

E. Recommendations:

In the next phase of the project, the CERC computing environment will be enhanced through:

- additional disk space;
- additional local disks for swapping and paging to improve diskless workstations performance;
- acquisition of more output devices (i.e. printers, plotters, etc.), and
- acquisition of more software.

F. Publications:

NONE

G. Hardware:

The following hardware and software was acquired for the computing environment.

- DISK SERVERS
  - 1 SUN 4/280 with 1.7 GB of disk space, and
  - 2 Digital VAXServer 3600s with 1.2 GB of disk space.
- WORKSTATIONS
  - 5 SUN 4/110s;
  - 1 Silicon Graphics 4D/240GTX;
  - 4 Silicone Graphics 4D/20s;
  - 5 Digital VAXstation 2000s;
  - 5 Digital VAXstation 3200s, and
  - 2 Hewlett Packard 835 TurboSRXs.
- PERSONAL COMPUTERS
  - 3 Apple Macintosh SEs, and
  - 6 Apple Macintosh II computers.
- PRINTERS
  - 2 Digital LN03 Scriptprinters;
  - 2 Apple Laserwriters;
  - 2 Apple Imagewriters, and
  - 2 Digital LA75s.
- TERMINALS
  - 5 Digital VT230 terminals
- COMMUNICATIONS EQUIPMENT
  - 2 Vitalink Translan IIIs (used for Local Area Network (LAN));
  - 5 Digital Link Modems (used for LAN);

- 1 Digital DEMPR (used for LAN);
- 1 Cayman GaterBox (used for Appletalk to Ethernet gateway);
- 1 Farallon Star Controller (used for Appletalk), and
- 1 Encore Annex II terminal server.

• SOFTWARE

- Operation systems, and C language, Fortran and support software for all workstations, including:
- ICAD;
- Objective C;
- IDEAS;
- Experimental Design;
- LASER;
-TRUCE, and
- CHIDE.

DICE PROGRAM

Task 3.2.1.2 Assembly Cell **West Virginia University**

A. Objectives:

The objective of this task was to monitor American Cimflex's efforts in constructing a printed circuit board assembly workcell and provide a communications channel between American Cimflex and West Virginia University staff working in related areas. This effort included coordinating the Cimflex efforts with parallel activities at West Virginia University in the design and assembly of a printed circuit board workcell.

B. Approach:

The work in this task to monitor the work at American Cimflex on the assembly workcell and coordinate the Cimflex efforts with West Virginia University researh was accomplished through:

- Direct meetings with the personnel from American Cimflex and from West Virginia University;
- Providing a contact person and a gateway for Cimflex and West Virginia University;
- Organizing a series of monthly meetings between WVU and Cimflex staff, and
- Providing technical assessments of the efforts at American Cimflex to the Concurrent Engineering Research Center (CERC).

C. **Technical Results:**

The work cell is being constructed by American Cimflex and will be delivered in late, 1989 or early, 1990. The project is behind the schedule proposed by Cimflex in their original proposal. There have been delays: Initially, there were some delays caused by administrative difficulties in the definition of the contract between West Virginia University and American Cimflex. Since then however, an excellent line of communication between the staff at American Cimflex and the research faculty at West Virginia University has been established. American Cimflex has supplied technical information on the workcell, sample circuit board, sample surface mount components as well as examples of typical component feeders. Moreover, there was an exchange of information about the software interface between the workcell and the design assembly CAD package. All in all, the coordination, in the design and assembly task at West Virginia University and the related work at American Cimflex, has been good.

As the work at American Cimflex and West Virginia University proceeded, it became apparent that face-to-face meetings between WVU and Cimflex were needed. A series of meetings were held in early, 1989, at West Virginia University and at the American Cimflex facility in Pittsburgh. These meetings provided needed insight and understanding of the mission of the assembly cell task.

A demonstration of the DFA/PCB tasks was given in July, 1989, at the West Virginia University CERC. The demonstration illustrated the WVU-

Cimflex workcell interface. A planned demonstration of the workcell and the WVU software interface is scheduled for September 1989.

D. **Conclusions:**

This task led to a close working relationship between West Virginia University and American Cimflex. The coordination of task work has been very good. In particular, the face-to-face meetings have benefited both groups in understanding DICE goals and related tasks.

E. **Recommendations:**

The monitoring of American Cimflex, as a sub-contractor, has not been as successful as would be desired and it is recommended that the lines of communication between the CERC staff at West Virginia University and the sub-contractor be more open. In particular, parallel lines of communications between the administrative staff at WVU CERC and the sub-contractor be opened and the monitor be better informed of these exchanges. It is also recommended that the duties and responsibilities of the task be better defined.

F. **Publications:**

See attached report from Cimflex Teknowledge.

G. **Hardware:**

None

FINAL REPORT - PHASE 1

CIMFLEX TEKNOWLEDGE CORPORATION

Table of Contents

1.0 Introduction

This final report is written to document the tasks and results associated with the development of a advanced workcell for high density electronics (HDE) assembly. Cimflex Teknowledge Corporation, located in Franklin, Massachusetts, was assigned the responsibility of developing the workcell as part of the DICE program to validate the capabilities of the Concurrent Engineering environment. Cimflex was a sub-contractor during DICE Phase 1 to West Virginia University's Concurrent Engineering Research Center (CERC) located in Morgantown, West Virginia.

The period of performance of this contract was initially from July 12, 1989, to July 31, 1989. The contract was later extended to August 31, 1989. The workcell was successfully demonstrated to General Electric and CERC personnel on September 12, 1989 to confirm the completion of all contract deliverables. A second demonstration of the workcell was given to General Electric, CERC, and DARPA personnel on October 11, 1989. The workcell will remain at Cimflex during most of DICE Phase 2 before being shipped to the CERC on January 30, 1990.

2.0 Overview and Goals of the Advanced Workcell Initiative

Manufacturers of complex electronic products face significant challenges in circuit design and production. Miniaturization and high density electronic packaging are rapidly advancing the capability and power of commercial and military electronics systems. The continual evolution of HDE technology offers far greater flexibility in circuit design and functionality than current technologies. The impact of this sophisticated HDE technology on the production environment is far reaching, and will require the development of new supporting processes and production methods for HDE to reach its final product form.

While the manufacturing technology base must be upgraded to handle the latest advances in electronics technology, such as HDE, the needs of providing a greater variety of end products, shorter development lead times, and lower unit costs remain as high operational objectives. Defense and aerospace suppliers are confronted with the most stringent of these requirements, and must simultaneously satisfy exacting standards for product quality, reliability and environmental stability. The combined effect of these factors results in greater costs and development cycles for weapon systems used today, and promises to increase as HDE and other technologies come to the forefront. Much of the development cost and time is associated with the iterative design, prototyping and testing of new printed wiring board assemblies (PWAs) and high density modules.

The mission of this Assembly Workcell Project is to develop fundamental manufacturing system technology to facilitate high flexibility and rapid prototyping and testing of current and emerging HDE circuits. The primary goal is to minimize the iterative circuit and process development times by providing capability for new designs to be assembled and tested within 24 hours. This capability is expected to provide significant benefits to the defense electronics industry.

3.1 Workcell Approach and Technical Strategy

The following sections identify some of the key problems in the defense electronics manufacturing domain and the technical approach taken to resolve them.

3.1.1 Nature of Defense Electronics Industry

Electronics manufacturing operations vary greatly in size and complexity, but there are several underlying characteristics that are common to defense electronics production environments. First, they tend to produce a large number of assembly types in relatively small quantities. A typical production profile would be two hundred board styles, fifty boards per lot, and over one thousand part numbers. This high mix, low volume scenario results in numerous equipment changeovers which can cause substantial delays in the production flow. The two most time intensive tasks during equipment setup are: 1.) changing the software programs that run the machines, and 2.) changing the mechanical fixtures that are unique to different board styles. Furthermore, the equipment used today is somewhat application specific in that only a limited number of tasks can be performed. For example, a fluid dispensing system can only dispense, a placement system can only place, and a test system can only test. Most equipment manufacturers design their systems in this manner since they serve a large customer base of high volume producers. For the defense electronics producer, the net result is a combination of highly specialized machines with different programming languages, different interfaces, and usually a high price. The inflexibility and cost of current equipment has impeded the use of automation in this environment and has left human operators as the only viable substitute in many cases.

Humans are extremely flexible, but they do not achieve the same level of consistency as automated equipment. The average first pass yield of circuit boards assembled manually is less than seventy percent. Furthermore, fifty percent of the field failures are attributed to manual rework performed in the factory. The use of manual methods in the production of high density electronics is intractable, if not impossible. The precision and fineness required for high density electronics assembly can only be achieved reliably with automated equipment.

Another deficiency in the production of circuits boards in the defense sector today is the excessive amount of time needed to proceed from design to manufacturing. The average time currently spans sixty to one hundred days. Although much of the time is spent in component purchasing and logistics, a significant amount is consumed in translating the design documentation into a form suitable for manufacturing. In many cases, special tooling and operator training are required for new designs which can add further delays in the completion schedule. Since many operations are performed manually they do not take advantage of design data that may exist in an electronic format, such as component locations stored within a CAD file. Automated systems can respond to CAD data and other computer generated information essential to the production process. Correspondingly, these systems can be monitored and controlled to produce the desired results, usually in a much shorter period of time.

3.1.2 Assessment of Available Manufacturing Technology

As mentioned previously, the vast majority of equipment used in the production of PWAs is application specific and correspondingly inflexible. The needs of the defense electronics industry for high flexibility have largely been ignored because the customer base of the commercial sector is much greater in magnitude and less demanding technologically.

The effort required to modify existing equipment to be more flexible would be enormous. Since communication standards are just now starting to emerge, changing current equipment would require access to each vendor's proprietary control and communication software, a highly unlikely possibility. Furthermore, adapting equipment to have common electrical, mechanical, and pneumatic interfaces would also prove to be improbable.

Instead of changing existing equipment, many forward thinking defense and aerospace contractors have designed and built custom automated systems to satisfy their difficult requirements. These custom solutions have often been robotically based to improve flexibility. Unfortunately, the associated tooling and software have been designed in a manner such that the overall system has become application specific and non-portable or transferable. The net result is a set of non-integrated islands of automation.

Some defense contractors produce a few types of PWAs in sufficient quantities to justify an automated production line consisting of off-the-shelf equipment, custom systems, and manual operators. The recurring engineering costs associated with equipment changes due to design or product changes tend to be high. The viability of the line typically lasts for the duration of the mature phase of the product's life cycle, which can be less than two to three years.

Aside from the inflexibility issues surrounding equipment available today for the production of PWAs, an equally significant issue arises when dealing with HDE. The relative newness of HDE has created quite a stir recently within the surface mount equipment manufacturing community. Suppliers are clamoring to introduce products that satisfy this segment of the market. Invariably, these suppliers are making modifications to their existing pieces of equipment. Again, they tend to address the needs of the higher volume commercial sector, and fail to provide hardware that is suitable to defense and aerospace PWA production. Worse yet, these machines are incapable of placing HDE components accurately enough for high reliability and high yield requirements. Even if equipment manufacturers upgrade the performance of their machines to meet military requirements, they will probably continue to be focused at the larger commercial market and retain the inflexible attributes that affect today's equipment.

3.1.3 Technology Selection and Evaluation

To satisfy the requirements of the DICE program, a high performance assembly platform was needed, one capable of addressing the flexible automation requirements of defense electronics manufacturing and the precision demanded by HDE assembly. (See Figure 4.1 for a conceptual sketch of a generic assembly platform.) Accordingly, the platform had to be able to be reconfigured with different tooling to accomplish a wide variety of electronics assembly tasks normally performed by separate, highly specialized pieces of equipment. Equally important was the workcell's ability to operate within the Concurrent Engineering environment established by DICE. Figure 4.2 illustrates the technical strategy envisioned for the workcell.

Below are key attributes the workcell needed to retain to provide this functionality:

1.) High Performance Assembly Platform
- Accuracy and resolution of the motion/drive system
- State-of-the-art control and programming system
- Vision-guided manipulators
- Stable structural frame

2.) Rapid Setup and Reconfiguration
- Quick-change end-of-arm-tooling
- Modular, smart feeders
- Simplified CAD-driven process planning
- Adjustable circuit board transport system

3.) Capability of Incorporating a Diverse Set of Flexible Process Tooling (Examples below)
- Various vacuum pickup tools for component placement
- Programmable placement pressure through compliant wrist
- CAD-driven tool selection
- Local fiducial matching and lead coplanarity inspection
- In-place soldering (integrated reflow)
- Solder paste dispensing
- Fine pitch component in-circuit test probe
- Post-solder inspection
- Component excision for repair and reattachment

4.) Integrated Engineer'ng and Manufacturing Databases
- Physical Layout
- Component data
- Tooling data
- Assembly sequence and process control parameters
- Production data
- Material tracking

A major benefit of this generic, high performance platform approach is that it enables multiple workcells to be configured in a production line system t^ assembly HDE circuit boards. (See Figure 4.3). Another advantage with the platform is the high level of commonality among stations, such as programming interfaces, mechanical interfaces, electrical interfaces, and pneumatic interfaces. The amount of operator training, maintenance training, and spare parts inventories are likewise reduced.

Given the unavailability of a piece of commercial equipment that met the technical specification of the generic assembly platform, an alternate solution had to be considered. To be viable, this new solution must also be cost-effective for the end-user. The remainder of this section describes the basis for the platform selection.

Mechanically, the workcell had to be founded on a stable and accurate platform. Since all electronics assembly tasks are performed in a X, Y, Z, and Yaw (roll) workspace, four axes of motion are sufficient. However, the method of accomplishing this motion needed to be identified. After evaluating traditional robots and other technologies as a starting point, it was determined that a manipulator system based on sawyer motor technology was superior in many aspects.

Unlike typical robots and other assembly systems that have jointed (e.g., SCARA) or cantilevered arms, the sawyer motor based manipulator is supported overhead on a rigid superstructure in a gantry configuration. See Figure 4.4. The sawyer motors are actually planar stepping motors that can move quickly and precisely along the ceiling of the workcell on air bearings. As a result, the sawyer motors operate without friction and are capable of making extremely precise incremental moves. The ceiling, or platen, is a steel plate with a finely engraved "waffle iron" pattern to position the manipulators. The only connection between the manipulators and the platen is the powerful magnetic field of the permanent magnets in the sawyer motors. This makes it possible to have multiple robots in the workcell and to have them move about each other, within the constraints of their cabling, to share feeders, tools, and assembly tasks. Since the platen can support additional manipulators, the workload of the cell can be increased while only incurring the incremental cost of more manipulators.

Additional advantages of the gantry arm design using Sawyer motor technology over other approaches are:

1.) Structural Integrity: the rigidity of the frame and the elimination of a main load-bearing joint provide a stiff and spatially sound platform.

2.) Higher Accuracy: Articulated arms must convert rotational movements into linear transformations. This "joint interpolation" introduces inherent inaccuracies not found in gantry robots since their movements are initially in a Cartesian reference frame (i.e., X and Y coordinates).

Although sawyer motors showed enormous promise as the fundamental platform technology for the workcell, only a few companies were involved in the manufacture of commercial systems. After a lengthy evaluation and analysis period, a sawyer motor based machine was purchased for development purposes. This initial unit demonstrated the viability of using sawyer motors to achieve accuracy needed for HDE assembly. However, it failed to address several key issues important for flexible manufacturing, such as an integrated data base, a simplified programming interface, and a vision system for manipulator guidance to enable CAD downloading. The choice was to add these features to the existing system or develop a different system.

The risks associated with modifying the current piece of equipment were considered too high, particularly in light of the inaccessibility to proprietary control hardware and software. Accordingly, it was decided to develop a new system based on the best available software, control, data base, and mechanical technologies.

An a Adept Technologies controller was selected as the cornerstone of the workcell. This unit included an integrated vision system, programming language and data base system. The motion control and drive units were purchased from a vendor who specializes in linear stepper motor control products. The electrical wiring, mechanical design and packaging, tooling, application software, and overall systems integration were performed by Cimflex. See Figure 5.x for a sketch of the workcell platform.

3.2 Workcell Platform Development

3.2.1 Mechanical Hardware and Related Sub-Systems

The workcell mechanical system design was approached from the standpoint of providing a platform which would meet both the immediate needs of the program and be readily integrated with other equipment in the future as the system develops. With this in mind, the workcell structure was designed to be as generic as possible relative to interfacing with peripheral equipment such as conveyors and feeders. Plus, considerable consideration was given to the layout of the cell, both internally to optimize production through-put and externally to facilitate loading parts into feeders, minimizing floor space, and providing access for reconfiguring the cell and performing maintenance, when required.

Packaging of electronics modules, controls, and operator interface was done with the goals of minimizing space requirements while retaining the ability to expand the system functionality by adding additional arms, component testing apparatus, and/or more sophisticated end of arm tooling. The placement of drives, cell computer, I/O modules, and pneumatics was determined according to proximity to their function, access requirement, and space availability. The electronics modules are all rack mounted to provide a building block means to configure systems. The racks are slide mounted to facilitate access for maintenance and for installing additional features and enhancements. The operator interface is from the front of the machine, where all controls and operator interfaces are placed. The feeders are at the back of the machine, where parts loading and system set-up (mechanical) is performed. The sides of the machine are closed except for conveyor feed through since the machine is proposed as an in-line device with adjacent equipment which normally restricts access from the sides.

The sawyer motors require a large, rigid and extremely flat structure to support the platen tiles. Integrating a structure of this type into the cell with minimal impact on floor space requirements and overall height was accomplished by utilizing a low carbon steel plate cantilevered at the front and rear edges such that the reinforcing ribs fit inside the support frame. This optimizes floor space and operator access to the workspace. The platen structure is open at the top, which lends itself to fabrication as a casting or a weldment, plus provides space for cables and airlines. The result is an economical design which adds a minimal amount to the overall height of the workcell structure. To assure that the platen structure retains its precision flatness, it is attached to the frame at three points, using spherical washers, to assure that inaccuracies and distortion of the frame cannot be carried over into the platen.

The frame is constructed from rectangular steel tubing which has been sized to provide a rigid, strong, yet open construction. The structure is sized to have minimal deflection due to cell operation or typical external forces to assure that the manipulator accuracy will not be affected by vibration. The joints of the frame were given special design consideration to assure that strength and stiffness of the tubing was fully utilized and usable space within the frame was maximized. This was done by adding reinforcement to the inside of the tubes rather than gusseting the corners, which would take away from space inside the frame. The frame is open in the back to allow maximum flexibility in configuring feeders and other related peripherals. Fork tubes are provided for ease and safety of handling the workcell.

The feeders and rear conveyor rail are mounted on a movable table that provides adjustment for board width. As a result, the feeders are always adjacent to the conveyor, which both minimizes the length of manipulator moves and maximizes the amount of available feeder space without the need to adjust anything but the conveyor. The feeder table is mounted on a frame which in turn attaches to the rear of the main frame, which has a vertical mounting surface with provision for installing the feeder table at various heights to suit the particular needs of an application.

The manipulators were designed to provide robustness while keeping weight and cost to a minimum. The manipulator structure is designed for stiffness. Similarly the bearings have been selected to have sufficient load capacity to preclude damage due to manual handling and/or collisions, each of which result in bearing loads much larger than normal service. In addition, bearings have been chosen which provide the required strength plus precision and stiffness, yet are compact and lightweight. The manipulator drives selected are rack and pinion for the Z axis and single reduction spur gear for the Yaw axis. The Z gears are precise, although precision is of secondary importance in the Z direction. The principal advantages of the rack and pinion are that it contributes negligible inertia to the motor, it is compact and lightweight thereby adding little to the size and weight of the manipulator, plus it is economical. The gear drive for the Yaw axis is much more critical in nature because the precision of this axis is essential to the function of the system. The gears used are precision gears and the drive gear is relatively large at 2" diameter such that the gear mesh is at about or slightly beyond the radius at which parts are being placed. Backlash in the gears is minimal and the torque on the Yaw axis due to the internal cabling and the limit mechanism is such that the gears are loaded and backlash should not affect accuracy of placement. The motors used for Z and Yaw are identical. They are a hybrid stepper motor that in fact is a very simple, rugged, brushless D.C. motor which emulates a stepper. The motor uses windings in the field to provide resolver like feedback which its controller uses to close the loop. The controller uses step and direction inputs to make the motor appear as a stepper, thereby making it easier to integrate into the system, while having the added advantage that because it has feedback it does not get lost, a useful feature since the force levels of these axes and the tasks they are assigned to would require greatly reduced performance if there were no feedback.

To support end-of-arm tooling requirements envisioned for the workcell, the manipulator incorporates 25 electrical and 8 pneumatic lines in an internal festoon system which is routed through the Yaw shaft. A tool changer which is compact, lightweight, and having this degree of functionality did not exist, therefore one was designed. The unit is very light in weight and it has all electrical and pneumatic lines entering and exiting axially, thus the real diameter of the coupling is the same as the O. D. of its housing, 2.5". Similar units which were looked at did not offer the pneumatics lines and they effectively add 0.63" to 0.88" to their diameter due to side exit signal lines.

The manipulator tooling consists of commercial products in the form of a Vertical Compliance Wrist (VCW) and vacuum tips. The VCW protects the components by limiting the forces applied during placement, while enhancing performance by permitting the manipulator to be run at high speed without fear of damage to the component due to Z height variation resulting from board warp, component tolerance or other sources. The VCW has position sensing to permit measurement of compliance motion. This can be used to determine that a part has been properly placed, to determine that the part was in fact still on the tool at the time of placement, and to determine height of objects in the workcell during calibration. The vacuum tool has quick change tips to permit using different sized cups for different sized components. The tips are of a very reliable design using an elastomer cup to provide good sealing and positive pick-up and hold of the part, plus there is a rigid ring around the cup to give a firm, flat surface to register the part, align it in a horizontal plane, and provide a firm reference when seating the part in the solder paste.

3.2.2 Controls Architecture and Integration

A block diagram of the Control System architecture is shown in Figure 4.5. The architecture consists of a system control unit which provides the overall system processing and coordination functions, and a separate drive control unit responsible for low-level control for each of the two manipulators.

The system control unit is responsible for the processing, storage and maintenance of all PWB assembly instructions and database information. During actual assembly sequences, the system control unit executes the control strategies required for the complete assembly sequence based on data retrieved from the system databases.

Throughout the systems operation, whenever it is required to reposition the manipulators, the system control unit transfers position and velocity information to the drive control units for each and every motion. It is the responsibility of the drive control units to attain the required motions.

The system control unit in the DICE workcell is the Adept IC Controller. The Adept IC was chosen because of its extensive processing power, integrated grayscale vision, standard industrial I/O control capabilities, and powerful software packages.

This unit features a 32-bit 68020 based system CPU with support from a 32-bit 68020 based vision board and a 16-bit 68000 based I/O processor. It is a true multi-processing system capable of simultaneously processing system functions, vision functions and I/O functions.

In addition to the above processor boards, the controller provides the interfaces to a variety of peripheral devices including a 20 Mbyte Mass Storage Drive, a Floppy Disk Drive, 512 digital I/O, 5 RS-232C serial devices, 8 video cameras, a video monitor, manual control pendant and system console CRT.

The drive control units where engineered specifically for the DICE workcell using standard industrial control components. This integration was required to enable control of the unique motors chosen for the 4 manipulator axes. The components were integrated into a 19 inch rack mount enclosure complete with cooling, internal power supplies, fused disconnects and all required external wiring assemblies. The drive control units where designed into the architecture to be self standing so that additional manipulators could be easily integrated into the system

The major components in this unit include two Motion Science Inc. MS12004-4-PSX Microstep Driver Assemblies required to drive the 2-axis X-Y linear stepper motor and two Sieberco AIM-1011 Driver Assemblies required to drive the Sieberco Sensorimotors used for the Z and Theta manipulator axes.

The interface between the system control unit and drive control units is provided by the Motion Science Inc. MPS-A-440-PS Motion Processor system. This 8-bit processor based three board set accepts RS-232C serial input commands from the system control unit and generates the subsequent motion profile signals to the motor drivers.

3.2.3 Software Architecture and Database System

The system software architecture was developed around AIM (Assembly Information Management) database system which is a software application package provided for use on the IC Controller. The AIM system offers a complete set of database management, user interface and control strategy tools which permit users to develop a custom database oriented system for specific applications. These tools were utilized in creating the data driven DICE assembly workcell.

The DICE workcell application and AIM system tools are both coded in the V+ programming language, the native language of the IC Controller. The V+ system is a multi-tasking operating system and programming language with a robust instruction set designed for industrial control applications. Though this language is unique to Adept controllers, it has been successfully isolated from the DICE workcell user due to the data-driven design and enhanced user interfaces of the application package.

The software architecture is composed of several major subsystems. These subsystems are highly integrated through an interface of internal global data which is used to pass data between subsystems that are executing as separate concurrent tasks in the multi-tasking environment. A block diagram of the software architecture is shown in Figure 4.6.
The major subsystems of the architecture are:

A.) A menu-driven operator console interface with full database access utilities and complete system control and maintenance screens.

B.) A serial device data exchange interface permitting system control and database access services for remote Concurrent Engineering Workstations or Production Line Management systems.

C.) A database management system for data organization, storage and retrieval, and cross relational linking.

D.) A runtime scheduling strategy responsible for multiple manipulator scheduling and status monitoring during assembly operations.

E.) A complete set of control strategies that are driven by the database information to perform the required assembly operations specific to the DICE workcell.

The responsibility of the two system control and data access interfaces (subsystems A & B) is to field inputs from their respective devices (system console or remote serial data exchange device), perform the requested operator actions and in return report status information. These functions include all system database accesses for manual data entry/display or remote upload/download, production order entry and initiation for executing actual assembly sequences, production status monitoring and general workcell maintenance and setup.

The database management subsystem (C) provides the system functions required to organize, maintain and provide access to the large amounts of information used drive the assembly sequences. In addition to actual assembly oriented data, the same data management functions are used throughout the AIM system to manage information concerning internal operations themselves such as the format of the assembly statements, the display of all menu screens and operator error messages.

The runtime scheduling subsystem (D) serves as the interface between the operator front end and the multiple manipulator control strategies that actually control the assembly operation. Whenever the operator or host control system initiates a production run, the scheduling subsystem is responsible for scheduling each step of the assembly sequence into the manipulator tasks and then reporting any problems that might occur during the placement operation. When fatal problems do occur, the scheduling task logs the error condition and request an operator response from the front end interface as to the desired recovery action.

The control strategies (E) are algorithms that were developed to perform the sequence of operations required to perform the specified assembly tasks. The strategies were developed to be very generic and modular in nature so they may be invoked on an as needed basis dependent upon the actions and parameters specified by information in the databases. These are the routines which position manipulators, actuate devices and apply vision tools during assembly.

The system databases were organized in a tree-like structure as depicted in Figure 4.7. Together with the underlying control strategies, the complete application package is divided into three major levels, the PWB assembly definition level, the workcell configuration and parameter level, and finally the process execution level.

The PWB assembly definition level (top level) consists of three databases whose information form a relation between a PWB assembly and its components. From the assembly workcell perspective, the primary difference in the definition of PWAs is the physical layout of each board, and type and order of the components being placed on that board.

The three databases are:

1.) Assembly or Board Database - Description of PWB layout

    Each data record in the assembly database specifies the coordinates of a physical board location for a component. The board database has one record for each component location on the PWB. Multiple board databases can be created and stored on the workcell.

2.) Part Database - Description of package style and source

    Each record in the part database specifies the part type (package style) and feeder name associated with the given part tag.

3.) Sequence Database - Relation between parts and board locations

    The data records of the sequence database are formatted as assembly instructions forming the relation between parts and board location names. Much like a PWA Bill of Materials. The order of records determines the order of placement.

Through relational database linking, the sequence database records direct the system to the appropriate part and assembly database records since two of the sequence database record fields are the respective part and assembly database record tags.

The workcell configuration and parameter level (middle level) consists of numerous databases which serve as libraries of information specifying the details of the physical workcell makeup and the various process parameters used by the low level strategy execution routines.

The same use of relational database linking to further levels of detail as described for the PWB definition level is used throughout the database system structure for the workcell configuration and parameter level. It permits the logical grouping of detailed information concerning specific devices and processes into separate databases. These databases may then be individually accessed and modified as required for workcell setup or process modifications.

As a further example, when a sequence database assembly instruction specifies a given part is to be placed, the associated part database record specifies a package style for the particular part and the name of the feeder from which that part shall be acquired. A relational link to the proper feeder database record produces the feeder location data (workcell setup), the feeder access strategy name (routine generating the sequence of operations for part acquisition) and additional parameters concerning the operation of that feeder within the workcell.

Similarly the part record also specified the part type being placed. A link to the workcell part type database record produces information such as which workcell end-of-arm tool is used to handle the particular style of component, what processes (visual inspection, lead forming, etc.) are applied to the component prior to placement, and what is the particular placement strategy for this component, direct placement, force assisted or vision guided.

Continuing the use of this database hierarchy for the remaining aspects of the system such as Tooling, Cameras, and Vision Processes you can begin to see that all the information concerning the assembly of boards is maintained in the system databases in some form or another. The higher database levels being more general and generic in nature and the lower levels being more device or process specific.

The process execution level consists of a library of subroutines which implement all the control strategies the workcell is capable of. Utilizing the database information and a series of standard procedure calls the algorithms are able to perform the operations required for assembly. The currently implemented control strategies are discussed in the next section.

3.2.4 Control Strategies

The control strategies are the runtime routines responsible for translating assembly data and process information into workcell commands to perform the assembly operations.

The main control strategy developed was the runtime scheduling task. This task is invoked upon initiation of a selected assembly sequence and is responsible for scheduling the efforts of the two manipulators in order to complete the requested production run.

The scheduling task parses the sequence database extracting the assembly instructions and determining which device shall, when available, execute a given instruction. Instructions may specify a particular manipulator for each instruction or utilize the default choice of next available. This task is responsible for fielding and processing all error conditions returned from the manipulator task as they execute the requested instructions.

The second layer of control strategies implemented were the manipulator control tasks. These independently executing task act as servers to the scheduling system continuously polling for their assigned assembly instructions. Depending on the requested instruction, a statement routine is invoked which begins parsing down the database tree structure extracting the relevant data to execute the statement.

Two statements have been implemented under the Phase 1 DICE program. They are the LOCATE.BOARD and TRANSFER statements. LOCATE.BOARD is a statement which positions a manipulator over three different fiducials on the PWB as it comes into the work station, visually locates these three fiducials utilizing the arm mounted cameras and object recognition strategies, and then determines the exact registration of the board as it is locked in the station. This procedure was required to support accurate placement onto CAD downloaded board locations.

The TRANSFER statement is the fundamental placement strategy routine. The basic algorithm implemented is to (1) change tools of required, (2) acquire from feeder, (3) visually inspect if required, and (4) place onto board. To support this overall process, the following low level strategies were implemented.

• Manipulator device interface routines - A complete set of interface routines to serially communicate motion commands to the drive control units had to be implemented. Standard AIM comes with such a package dedicated to the control of an Adept SCARA type robot system which utilizes backplane integrated axis cards and standard V+ commands to implement motion. This package had to be completely redeveloped into a set of functions which support the multiple Cartesian style manipulators controlled by remote drive control units interfaced via an RS-232C serial link.

• Anti-Collision - To support the fact that these multiple manipulators would be operating within the same envelope, a collision avoidance algorithm was developed that would prevent multiple devices from interfering with each others motion. The algorithm models the two-dimensional (X-Y) motion of each manipulator with two parallel lines and two points defining the envelope scribed by the motion. The algorithm continuously monitors the motion of all manipulators to assure minimum time delay before motion clearance may be granted.

Strategies to support operation of workcell peripheral hardware had to be developed. Those include:

- Feeder interfaces to support tape reel and tray type feeders.
- A Tooling interface to support part acquisition, insertion and rejection for the vacuum pickup style end-of-arm tool.
- A variable compliance wrist force sensor interface to prevent possible damage to component leads during pick-up and placement.
- A process control task to support board transfers through the conveyor stages.

3.3 Workcell Performance Capabilities

Starting with an overview of the operation of the workcell, a clearer picture can be obtained of how the various technologies discussed in the previous section are integrated to satisfy the requirements for HDE assembly. The sections that follow describe the performance of the workcell in terms normally used in evaluating the capabilities of an assembly machine.

3.3.1 Operational Scenario

One of the most powerful features of the workcell is its ability to automatically place surface mounted devices on circuit boards using design data downloaded from a CAD station. This allows the circuit board design function to be integrated directly with the manufacturing function, resulting in a substantial decrease in the design-to-assembly time for new products and products requiring changes.

The CAD station is capable of creating new circuit board designs, or of taking existing designs that were developed on external CAD systems and incorporating them into the local system environment using the Initial Graphics Exchange Specification (IGES) translator. The CAD station contains a component library of all SMD devices to assist the engineer in producing the design. This component database is also resident in the workcell database and contains geometric information such as lead count, width, and pitch. All these features are accessible through a user friendly, menu-driven operator interface that offers tremendous productivity improvements.

The workcell was designed to provide attendant-free operation by eliminating the need to teach placement positions, paths, speeds, etc., on the plant floor. A generic or "shell" program is resident within the workcell controller level that is pre-configured with all the necessary control information to direct the workcell through its tasks strictly using the data described in the CAD file. This software feature permits the system to remain in operation on a full-time basis.

Figure 4.8 shows a top view of the workcell. An edge-type conveyor is responsible for transporting bare circuit boards or pallets into the work envelope. The conveyor design consists of four stages: input buffer, workstation #1, workstation #2, and output buffer. Each station is controlled by a separate stepper motor to balance the flow of circuit boards through the workcell. The conveyor width is adjustable from 100 mm to 500 mm. Sensors along the conveyor monitor the position of the boards in transit and enable the controller to send signals to the conveyor's locate and lock mechanisms to secure the boards within the workcell.

A camera is mounted on each manipulator to identify reference marks (i.e., fiducials) on the board. This feature minimizes the need to precisely register the circuit boards on the conveyor. Once the orientation of the board is known by the vision system, CAD data can be used for accurate assembly. The manipulators then proceed to retrieve components from matrix trays located on an adjustable feeder table behind the back conveyor rail.

After component pickup, each manipulator presents a component to an up-look camera, one located on the input stage of the conveyor buffer and one on the output side. This arrangement allows the manipulators to operate independently of each other. The uplook cameras are used to inspect for lead quality, such as the absence and presence of leads, lead parallelism and defective leads. The vision system also determines the orientation of the component relative to the manipulator's vacuum pickup tool to compensate for X-Y offsets. This feature eliminates the need to mechanically align the component prior to placement and avoids damaging the leads. It also minimizes the requirement to have precise and costly component presentation mechanisms.

The manipulator has a programmable compliant unit integrated with the vacuum pickup tool to place components on the board with a wide range forces to compensate for different component geometries and sizes. This feature is particularly important when dealing with fragile component leads and circuit pads that have a fine layer of solder paste on them.

The robot controller is capable of collecting and displaying data during assembly, such as production counts, rejected parts, low feeder inventory, equipment failures and other alarm conditions.

3.3.2 Range of Circuit Board and Component Types

The conveyor is designed to accommodate circuit boards or pallets as small as 4" x 4", and as large as 20" x 20". These boards or pallets can range in width from 0.020" to 0.120", well within industry standards. The conveyor requires no more than 0.187" of clear board space at the side edges. The conveyor height from the floor is nominally set at 38" and is adjustable +/- 2 inches.

The different component package types that can be handled by the workcell are shown in Figure 4.9. These packages represent most of the typical component styles used in defense electronics manufacturing. The smallest component lead pitch tested was 0.020", although 0.015" pitch seemed highly possible during a cursory examination. Phase 2 will explore the smallest lead pitch device the workcell is capable of placing.

3.3.3 Component Placement Accuracy and Repeatability

Accuracy and repeatability are related, but distinct, properties. Accuracy is the precision with which the system moves or measures. This is an important property for a system such as this which is intended to use input in the form of CAD data and from that be able to move precisely to the required locations. Repeatability is the property that measures the precision with which the system does a given task time after time, in this case the placement of components. There is a third property which is important to both of the above, resolution. Resolution includes both the increments of manipulator motion and the pixel size of the vision system. The resolution of both must be finer than the smallest motions which the system is required to make, a rule of thumb for gaging systems is to have a resolution an order of magnitude smaller than the smallest increment to be measured.

One of the primary advantages of the sawyer motor is that is operates on an air bearing surface which is necessarily very flat. This in combination with the feature of the air bearing whereby it averages over the entire surface of the motor, thus further compensating for surface irregularities, lends a high degree of precision to the system. The surface is lapped to be flat within 0.0005" per foot, the air bearing averages any waviness that is of a shorter period that the 6" length of the bearing. The height of the manipulator, measured from the platen surface to the surface of the board where the parts are placed, is approximately three (3) times the length of the air bearing. This amplifies any non-planer motion of the sawyer motor, in this case a platen variation of 0.0005" results in a position error of 0.0015". This error is structural and can be and compensated for by several means.

Another feature of the sawyer motors is that they operate on a photo etched steel plate. The etching process produces a grid which is accurate of 0.001"/foot. It is subject to error due to thermal expansion of the steel structure, which will change in length approximately 0.001"/foot/per 10 degrees Fahrenheit. The grid has an 0.040" pitch which results in the sawyer motors having a full step size of 0.010". The controller divides steps into micro-steps that can be as small as 0.0001". Practical considerations and speed dictate that the micro-steps be 0.00025" or larger. The motors are such that they average over their full area, thus compensating for minor imperfections such as missing teeth and small dents in the platen surface. Because there are four motors driving along each axis and the tool axis is centered between the motors, the affects of localized errors are greatly attenuated, i.e. an imperfection which results in an error of 0.001" at one of the four motor poles will result in an error at the tool point of only 0.00025".

The Sawyer motors are stepper motors and as such they operate open loop in that they do not provide position information to the controller. The controller only "knows" the position of the motor to the extent that it is still in synchronism with the grid. This is measured in 0.010" full step increments, not very good resolution. However, stepper motors have relatively high position gain, in this case the position gain or spring rate is over 2000 lbs. per inch. The motor operates on a frictionless air bearing and the only static forces which it must overcome are those due to the umbilical cable. The cable is made of very flexible, lightweight wire and is mounted from an air bearing to minimize umbilical forces on the manipulator. The force is on the order of one pound, but varies less than 0.25 pound over the area of the conveyor where parts are being registered and placed. The 0.25 pound variation in combination with the 2000 lbs/in. spring rate result in a position error of only 0.00013".

The system utilizes vision to calibrate itself and to register the printed circuit boards and components. Resolution of the cameras and the type of measurement (area vs. edge) combine to give a sub pixel resolution of approximately 0.003".

The system accuracy is a function of the sum of the many parameters discussed above. In a simple, worst case scenario, the resolution of the vision system is of primary importance since it occurs during calibration of the system, board registration, and component registration. Thus the potential to contribute three or more times the resolution, or 0.0009" to placement error. The manipulator moves are affected by the accuracy over a move length on the order of the size of the printed circuit board, typically this will be less than 10", thus the manipulator error of 0.001" is representative. In addition, the errors due to bearing error, which could be as much as 0.0015", plus that attributed to compliance, 0.00013", and that due to manipulator resolution, 0.0025", must be factored in. These do not sum directly, but rather in a statistical manner, resulting in an error on the order of 0.0022".

3.3.4 Component Placement Speed

Placement speed is best measured in terms of system throughput. This is a function of all the elements of the workcell and the board being built. This should include the time required to learn a new board and to configure the system to produce it.

Speed of the manipulators will have a strong bearing on the throughput for long production runs of simple parts requiring little or no vision. However, the shorter the run and the more cell set-up and vision requirement, the less important manipulator speed becomes. However, for a typical placement, the manipulator will make an X-Y move of approximately 12", 14" and 10". At the end of two of these moves the Z axis will move up and down a distance of 1" and at the end of the third move, vision will register the part. The average time for the X-Y move is 0.67 second, that for the Z moves is 0.3 second, for total move time of 2.6 second per placement. The time for vision processing, cell control, motion planning, etc. is additional, typically doubling the placement time.

3.3.5 Reliability and Maintainability

The mechanical system has been designed with reliability as one of the primary design goals. The selection of Sawyer motors, which have no moving parts to wear out and which incorporate non-contacting air bearings for the manipulator to move about on, was in part based on their intrinsic reliability.

This was carried a step further with the use of the Seiberco hybrid stepper motor for the Z and Yaw axes. These motors are actually a brushless D. C. motor which utilizes additional windings in the armature to sense position. This provides a very rugged and reliable feedback transducer which is every bit as tough as the motor itself, unlike an optical encoder which is sensitive to shock and temperature. A further advantage of these motors is that, for their frame size, they have larger capacity bearings than a conventional stepper motor, an important consideration in light of the use of gear drives, which place radial loads on the shaft.

The motors have been manufactured with an integral drive pinion to provide the maximum of precision with minimum size weight and complexity in the drive train. These pinions are precision gears and they are made of high quality material which is nitrided to maximize wear life and load capacity. The bearings and gears on both axes have been designed with reserve capacity such that they are unlikely to be damaged by extraneous loads and more importantly, so they have virtually unlimited service life (>20,000 hours). For purpose of design and maintenance simplicity, the Z and the Yaw were made identical.

The reliability of the wiring with respect to its ability to flex across the different axes of the manipulator was an area of concern during the design of the manipulators. This was particularly so for the Z and Yaw axes. The Z axis service loop is a simple rolling flexloop, divided into two parts. The first part incorporates the EOAT and the manipulator limit ribbon cables plus the pneumatic ribbon tubing. The second loop is for the camera and the Z and Yaw motor cables which require a larger flex radius. Both of these have been designed to minimize stress in the cables within the small space permitted by the manipulators small size. The Yaw axis flex loop was a much more difficult problem due to the desire to have 360 degrees of rotation and to keep the exterior of the manipulator free of cables and tubing. An internal flex loop was designed which provides controlled flexure to maximize cable life in a compact package which provides 25 electric and 8 pneumatic lines for the EOAT.

Maintainability of the system has been optimized by minimizing the components in individual modules, which, in conjunction with the modular design of both the electronics and the mechanical system, allows for maintenance by replacement of modules (electronics racks or manipulators) in addition to providing for easy removal of modules for bench top repair. Periodic maintenance is minimal, consisting of routine application of rust inhibitor to the platen and lubrication of the gears and bearings.

DICE Workcell Program - Generic Platform

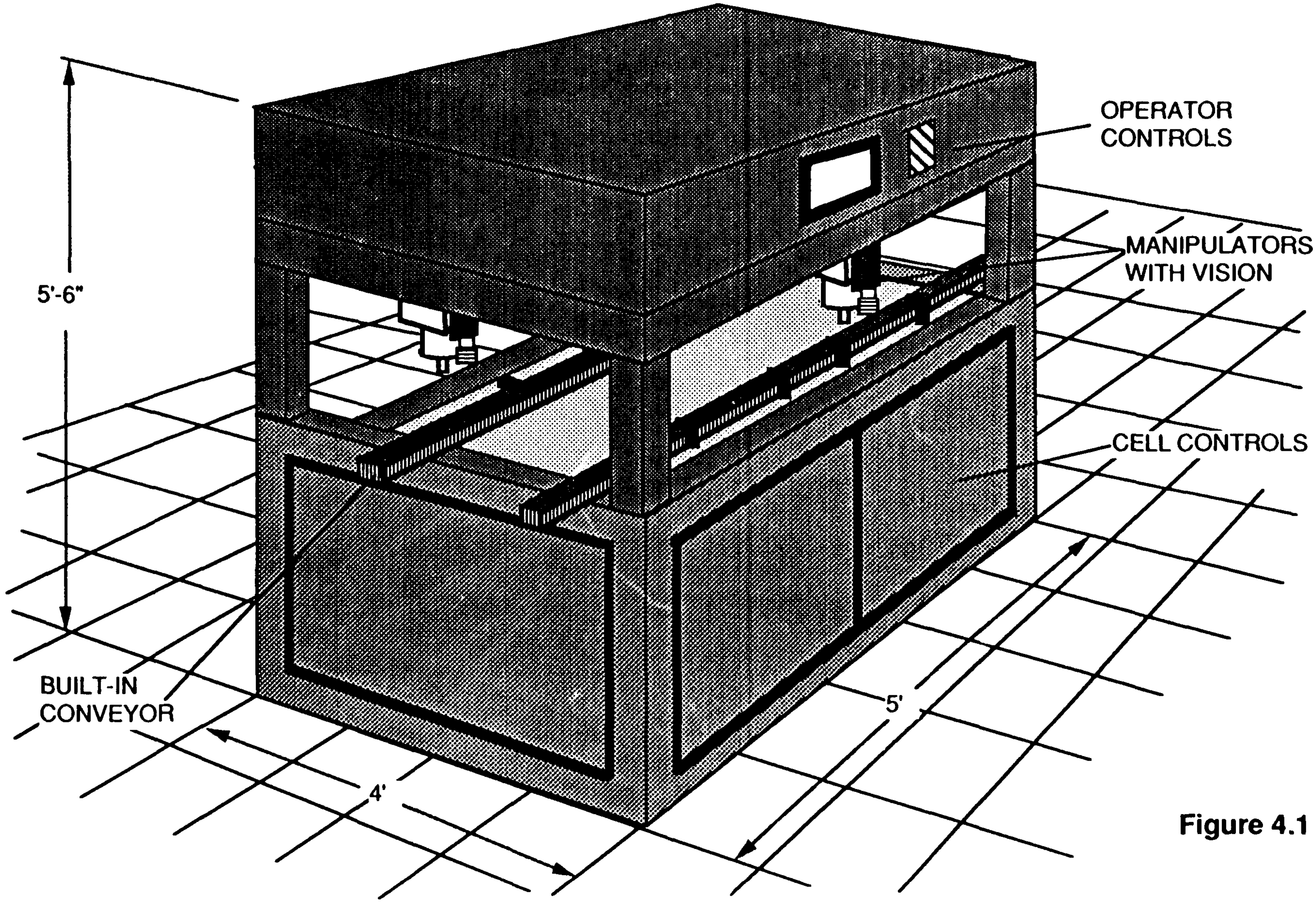


Figure 4.1

DICE Workcell Program - Overall Technical Strategy

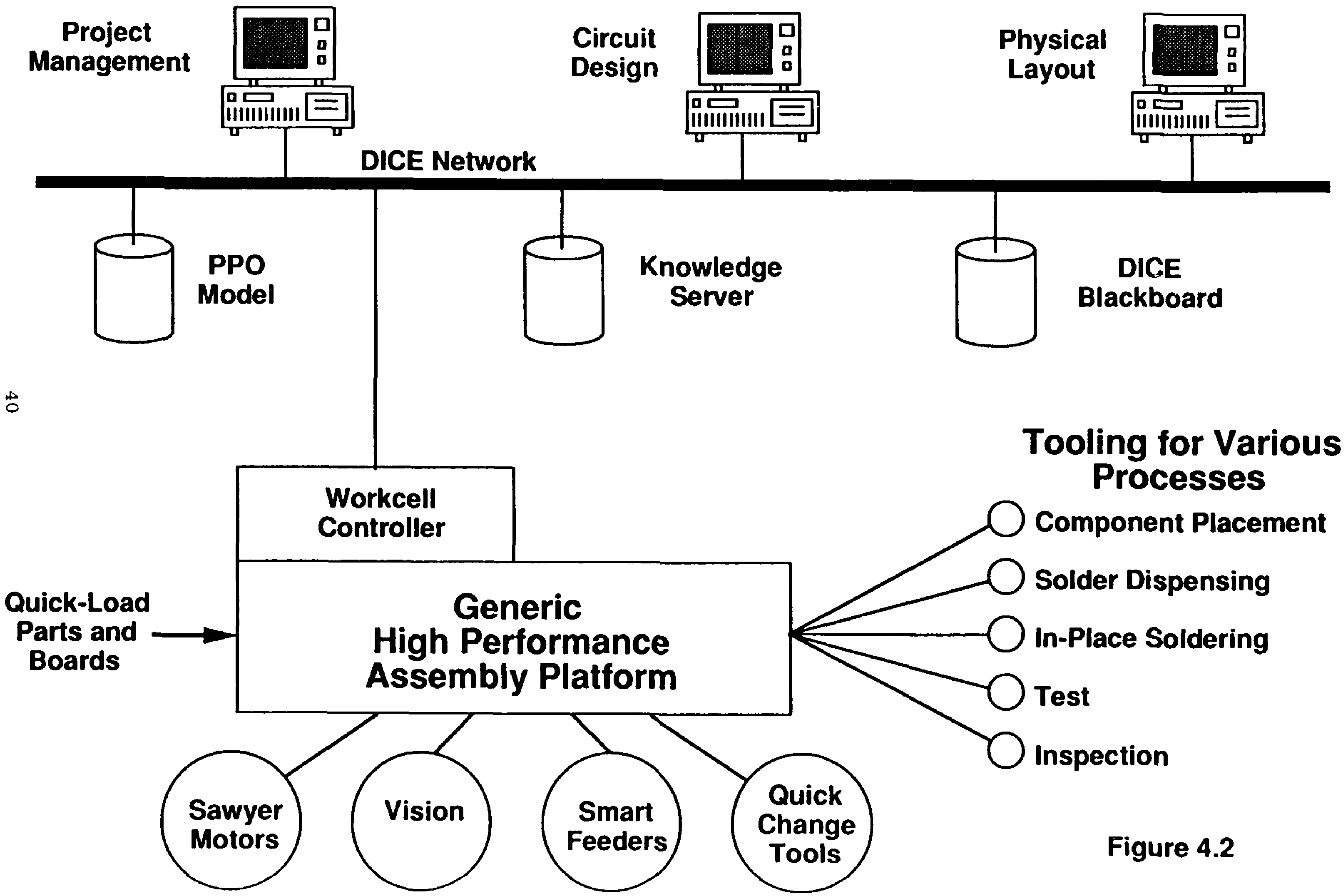


Figure 4.2

DICE Workcell Program - Example of Integrated Workcells

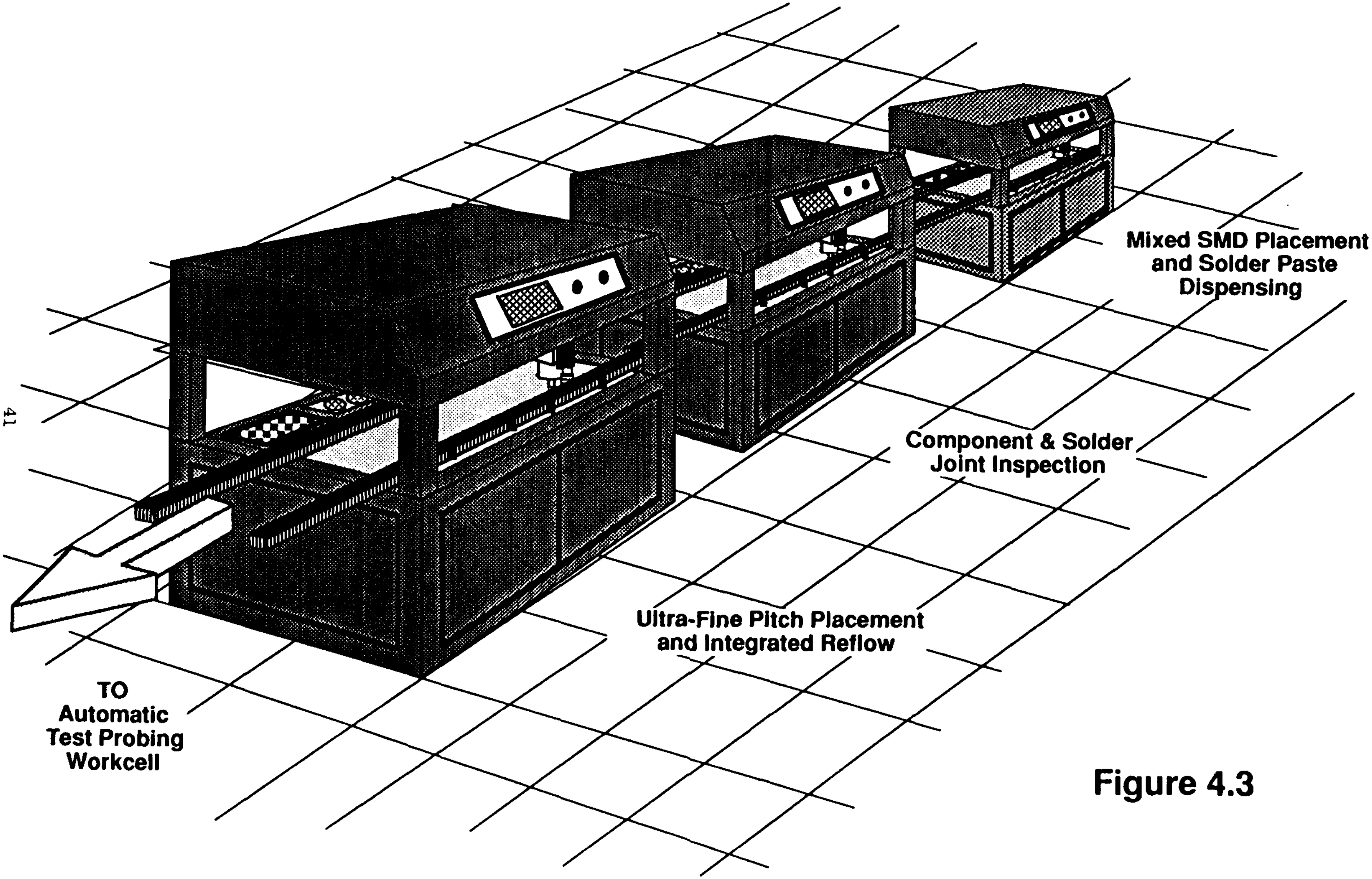


Figure 4.3

DICE Workcell Program - Platform Mechanical Hardware

END VIEW ELEVATION

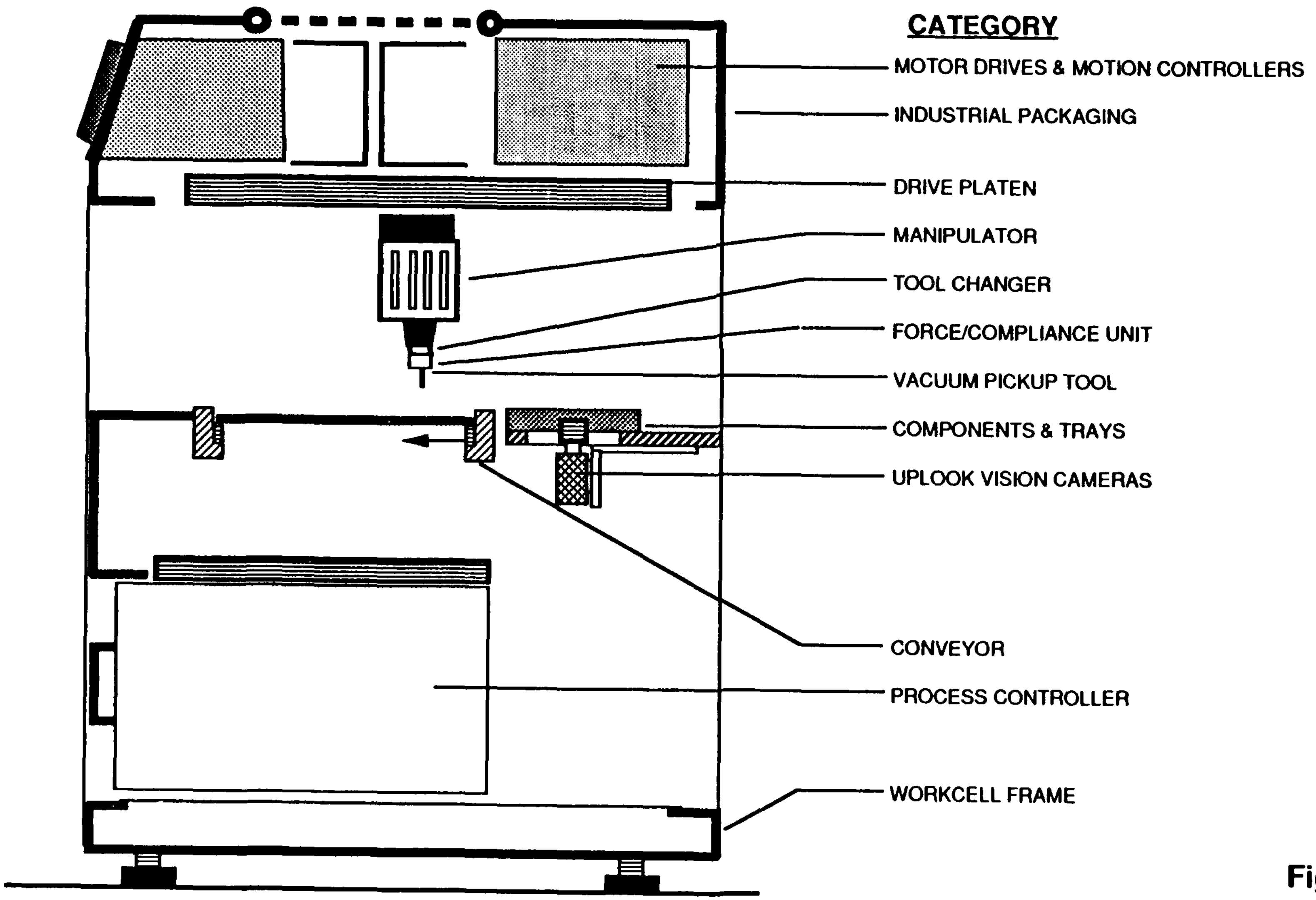


Figure 4.4

DICE Workcell Program - System Control Unit Status

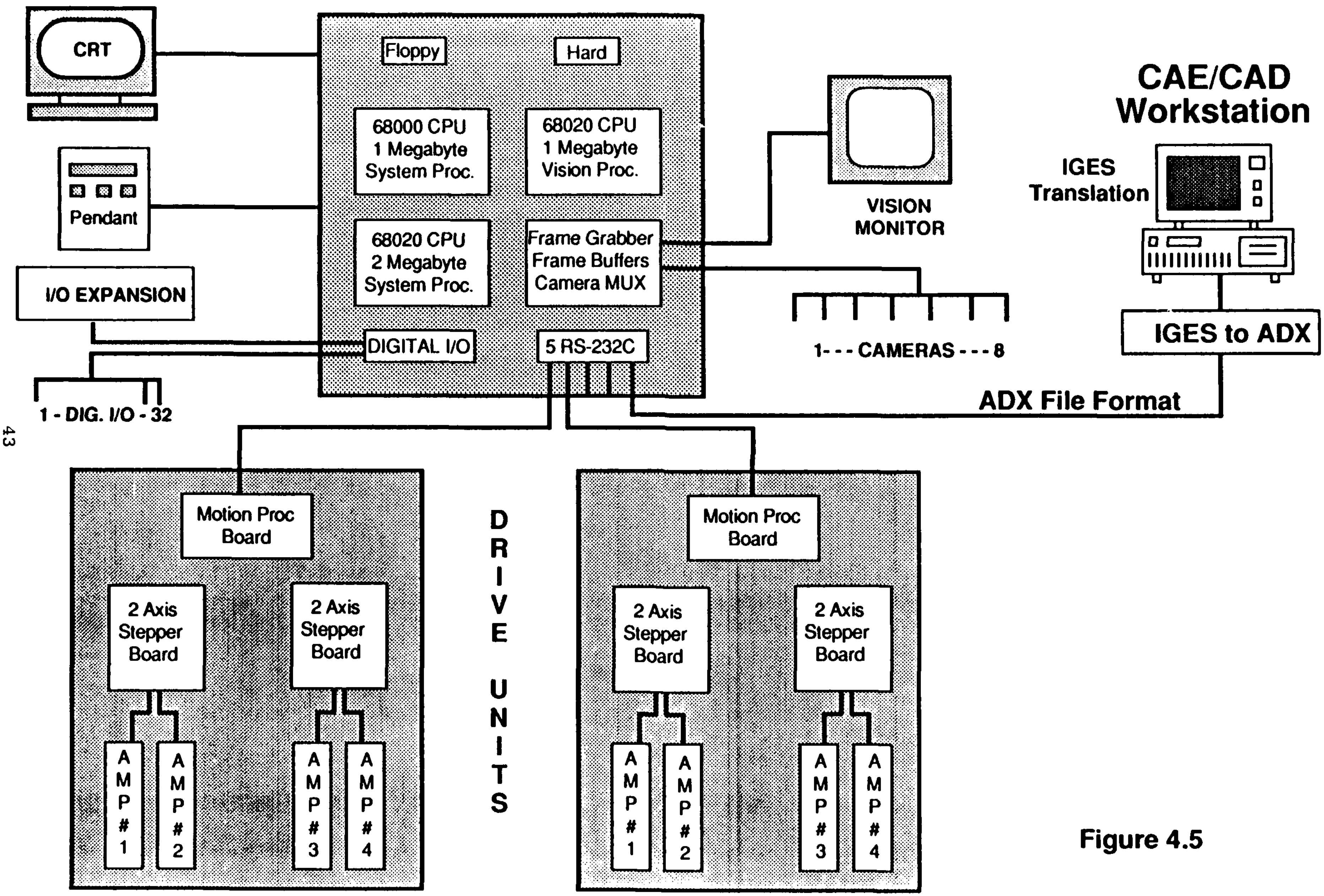


Figure 4.5

DICE Workcell Program - Software Architecture for Assembly Control

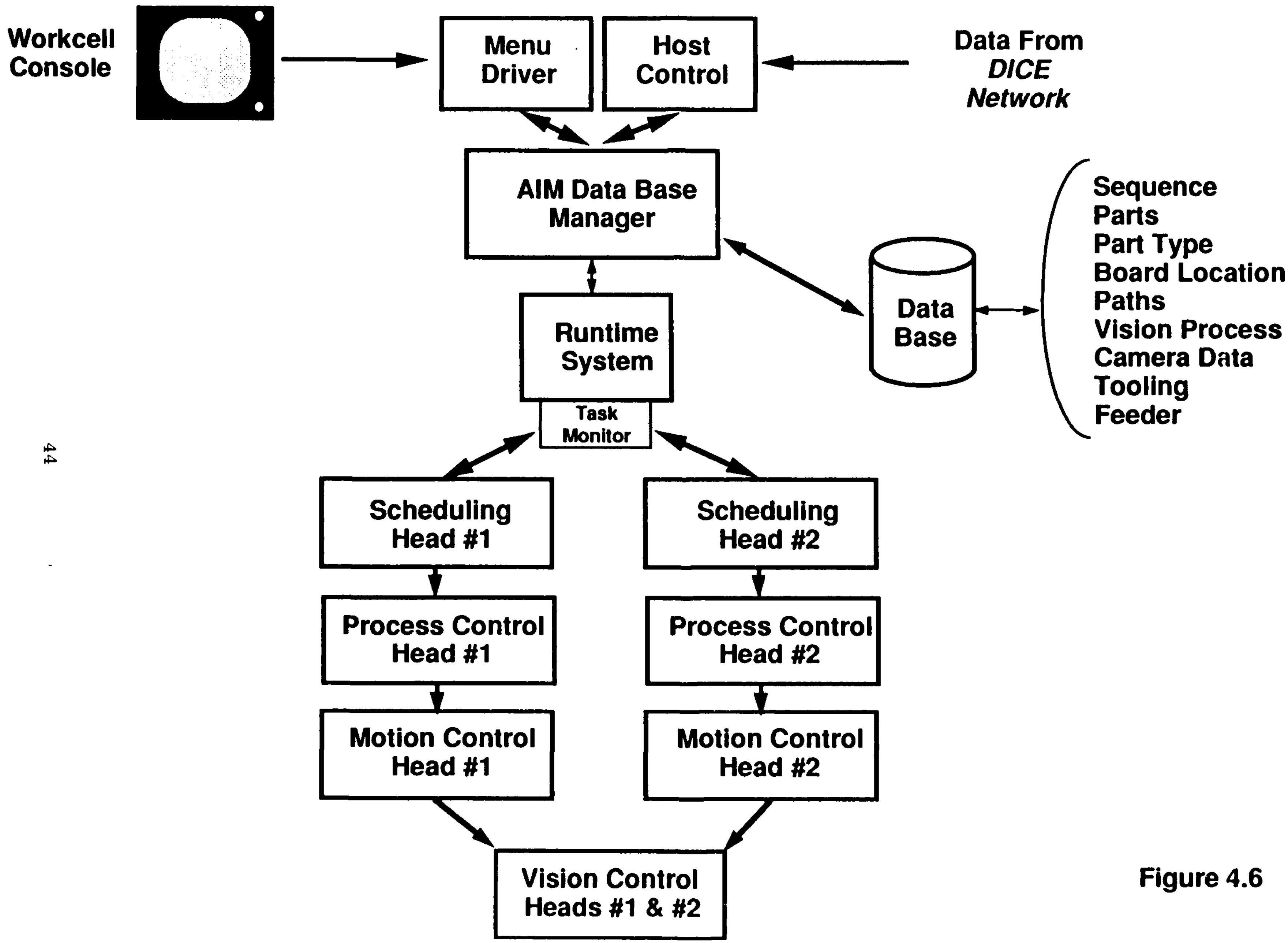
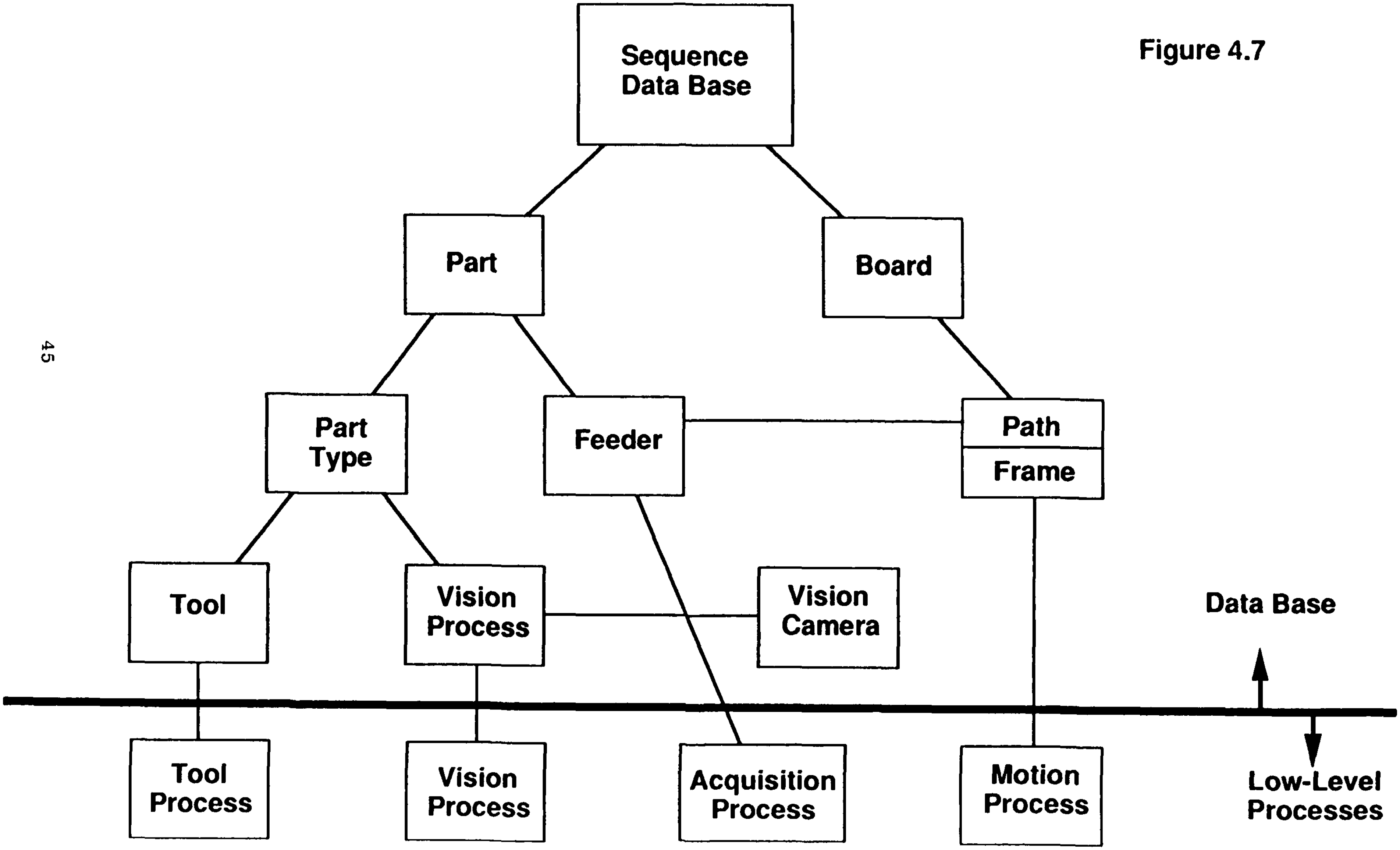


Figure 4.6

DICE Workcell Program - Data Base Structure for Assembly Control

Figure 4.7



DICE PHASE 1 WORKCELL LAYOUT - PLAN VIEW

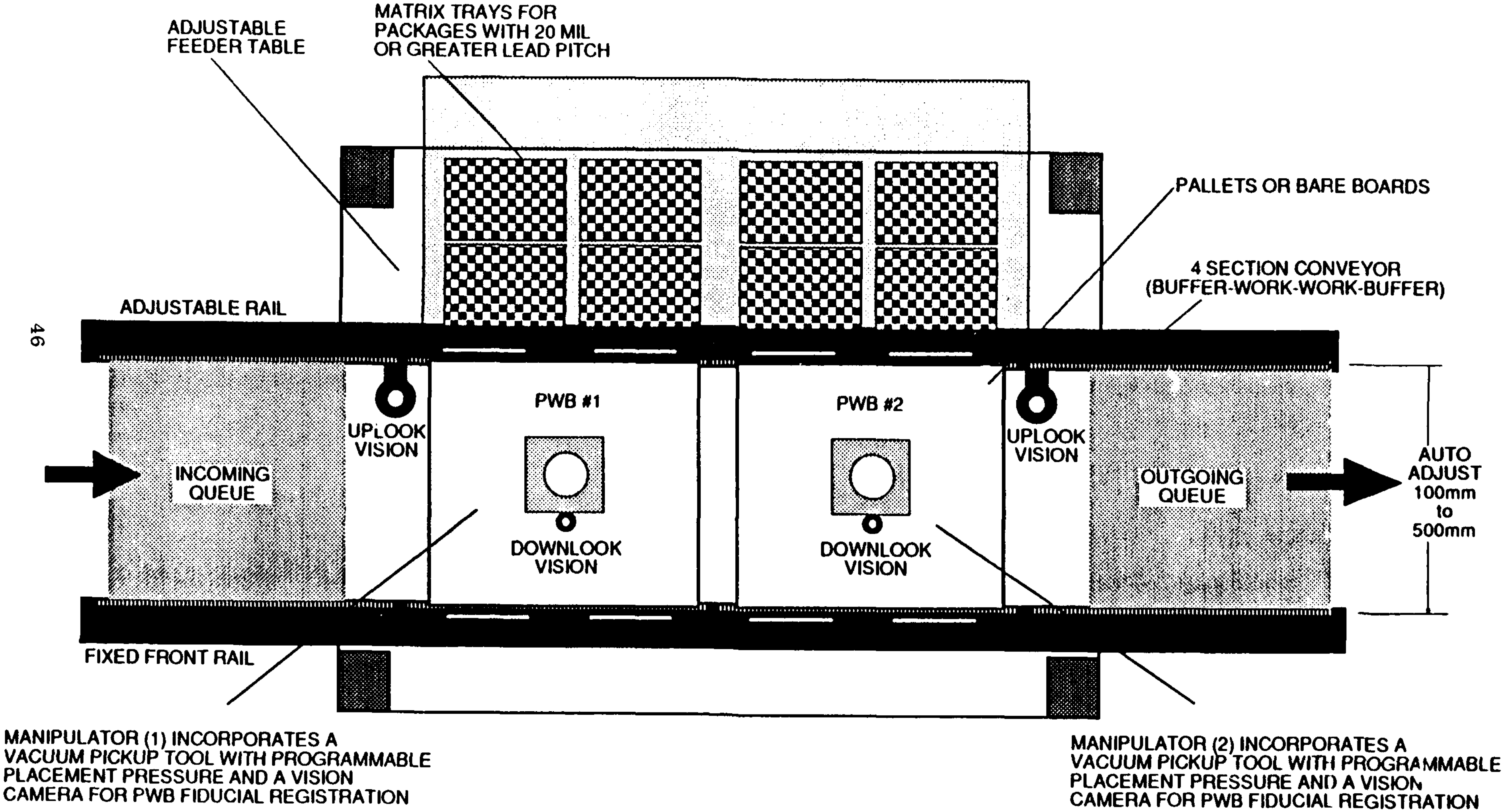


Figure 4.8

DICE Workcell Program - Typical Surface Mounted Devices

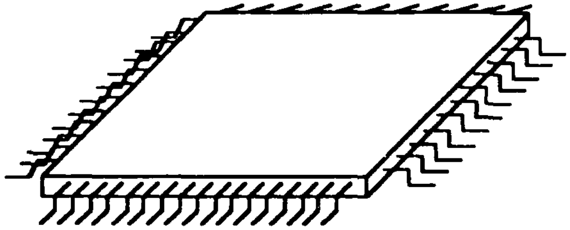
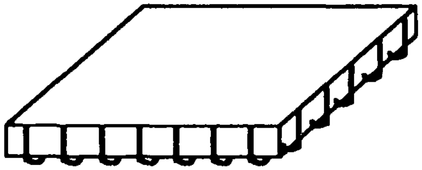
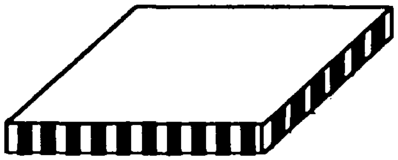
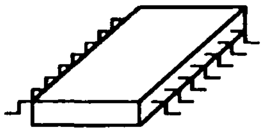
Illustration	Type	# of Leads	Pitch	Feeder
	QUAD PACK	54 - 172	0.020 -0.050"	Tray or TapePak™
	J-PACK	44 - 68	0.050"	Tube or Reel
	LEADLESS CHIP CARRIER	68	0.050"	Tube or Reel
	SOIC	24 - 28	0.050"	Tube or Reel
	CHIP	N/A	N/A	Reel or Track

Figure 4.9

DICE PROGRAM

Task 3.2.1.3 Materials Characterization Laboratory **West Virginia University**

A. Objectives:

One of the objectives of this project is to strengthen the research facilities at West Virginia University for the synthesis and characterization of advanced materials. These advanced materials include stable compounds and intermetallics, artifically stabilized structures, composites and long-lived metastable structures. Specific task objectives include developing a research program for designing advanced materials with superior properties (low density, high temperature strength, creep resistance, and environmental stability) in order to tailor the properties of these materials for a variety of applications.

B. Approach:

An experimental program was designed for both the synthesis and characterization of materials. During Phase I, the goal was to prepare TiAl alloys using existing facilities (powder x-ray diffraction, thermal expansion and specific heat measurements) and to add newly acquired equipment, including:

- A modern x-ray diffractometer with low and high temperature capabilities;
- A SQUID (Superconducting Quantum Interference Device) for measuring electronic and magnetic properties;

- A TMA (Thermomechanical Analyzer) for measuring temperature dependence of the thermal expansion and mechanical properties;
- A hardness tester for measuring the mechanical strength of materials, and
- A Rotating Anode X-Ray System for the analysis of single crystals, superlattices and multilayer systems.

During Phase II, an optical microscope for materials characterization and a sputtering unit for the preparation of films and multilayers will be acquired. The X-ray diffratometer, the SQUID, the TMA and hardness tester have been installed and are operational. Rotating Anode System components are being received and the system will be assembled by September 15, 1989. Some site preparation were necessary for installation and operation of these systems, including the acquisition of several rooms in Hodges Hall, air-conditioning these rooms and upgrading the electrical wiring and plumbing in these rooms.

Cooperation from several sources was essential to accomplish these objectives in such a short period. The Physics Department was cooperative in agreeing to provide the necessary space; the Office of The Provost was instrumental in accelerating the room renovations; the College of Arts and Sciences provided the necessary funds for five air-conditioners; The Physics Machine Shop provided invaluable assistance in the renovation of the rooms and installation of the equipment, and Physics Department faculty provided important advice in the selection and acquisition of the various pieces of equipment. These pieces of equipment are now being operated by graduate assistants and postdoctoral associates.

C. Technical Results:

The equipment is being used for the study of TiAl intermetallics and other materials. Other results are included in the Final Report under Task 3.4.5.4; Material Characterizaiion.

D. Conclusions:

Not applicable.

E. Recommendations:

See Final Technical Report, Task 3.4.5.4.

F. Publications:

See Final Technical Report.

G. Hardware:

The folllowing hardware was purchased and installed during Phase I:

• X-Ray Diffractometer; Rigaku model D/Max with computer control, and accessories for low temperature and high temperature studies; it has been installed in Room B-17, Hodges Hall; the cost was $136,654;

• SQUID, Model MPMS by Quantum Design Inc., with accessories for high temperature studies and hysteresis measurements; Half of the cost ($60,000) of this instrument was provided by West Virginia University for a laboratory; it is installed in Room B-13, Hodges Hall; the cost was $128,050;

• Hardness Tester (Model 940-142); it is installed in Room B-05, Hodges Hall; The cost was $6,343;

A Rotating Anode System with a 18 kW Generator by Rigaku, four cycle Goniometers by Huber, and detectors by EG&G and Tennelec. The system, along with computers by Apple , and other accessories, were installed in Room G-40, Hodges Hall. The system is expected to be assembled by September 15, 1989. The system cost was $250, 000, and

• The Department of Chemistry has received equipment for the measurement of variable frequency microwave dielectric loss and surface resistivity. The unit has been assembled from various components (10 MHz-20 GHz Synthesized Sweeper, Sclaar Network Analyzer, Directional Bridge, AC/DC Detector etc.) purchased from Hewlett Packard. A microwave cavity with a center frequency of 15 GHz is being fabricated for use with the system. The system is installed in the Chemistry Research Laboratory. The system cost was $50,000.

DICE PROGRAM

Task 3.2.2 CERC Administration & Technical Services WVU

A. Objectives:

This task has focused on the development of a preliminary organizational structure for the CERC and the coordination of its research activities, both within the center and between the center and the other participant institutions in the DICE program.

B. Approach:

Three basic levels of research coordination have been identified and pursued by the CERC:

1. Coordination among the various research tasks within CERC. This is an important level of coordination for the coherence of the CERC team and to assure that the efforts of all its members follow a common, integrated path. Such coordination has been implemented through regular faculty meetings, frequent discussions between researchers involved in different tasks, review and planning meetings between the technical management of CERC and the personnel of various research tasks;
2. Coordination among the various organizations involved in the DICE program. This type of coordination has been pursued both formally, through official coordinators who have been appointed by each organization, and informally, through individual researchers who share common interests and objectives, and

3. Coordination within the Special Interest Groups (SIG). These groups were established in Phase I of the DICE program in order to facilitate and formalize the interactions among all the program participants in certain technical and administrative areas. This type of coordination has been implemented primarily within two SIG's; DICE System Architecture and Design Methods. Most research activities that fall into one of the other SIG have been defined at the start of the program and have been subsequently incorporated into one of the two SIGS.

C. Technical Results:

A preliminary organizational structure was developed for the technical and administrative operations of the CERC and is illustrated in the attached diagram. It reflects three major types of effort that the CERC pursued in the first year of its existence:

1. Technical - planning, undertaking and reporting a broad spectrum of research tasks, developing strategies for technology transfer, training research personnel and designing a concurrent engineering curriculum;
2 Administrative - financial planning and management, acquisition of software, hardware and information material, hiring of research and support personnel, and developing promotional activities at both the local and the national levels, and
3. Facilities - planning and contracting the construction of a permanent building for the CERC, developing a suitable and fully-networked computing environment, defining specifications for prototyping,

testing equipment, communications, conferencing, and information storage and management.

An integrated demonstration of a blade design scenario was presented at the end of Phase I. As a result, there was a successfully close research coordination within the System Architecture SIG, between the researchers from CERC and GE/CRD and between the "Architecture" and "Engineering Modeling" teams within CERC.

On May 23-24, 1989, the CERC hosted its First National Symposium in Concurrent Engineering. The event, combined with an official dedication ceremony and an "open house" of the center, was attended by Senator Byrd and about 100 invited guest from government agencies, industrial companies and academic institutions. The National Symposium provided the DICE program visibility in the national arena and informed the "Concurrent Engineering community" of the main objectives and approaches being pursued by all the participants in the program.

In addition, the CERC Outlook, a "newsletter" published by CERC on the DICE program, is disseminated to nearly 500 subscribers all over the nation and provides an effective means of communication for the DICE program and with outside organizations interested in Concurrent Engineering. Currently, special attention is being given to enhancing the quality of the "newsletter," ranging from graphic appearance to sustantative technical articles and news items.

Finally, a series of technical seminars well conducted at CERC during Phase I to update faculty, staff and students on emerging trends and national issues in concurrent engineering.

D. Conclusions:

In a multi-participant and multi-task research program such as DICE, close coordination at all levels, from researchers to institutions, is essential for the success of the project. This is especially true for the DICE program, which is expected to evolve integrated computer environment concurrency and effective communications between the various tools and experts involved in product development. While enhancing individual tool capability is extremely important it is equally important to realize that, tools are better utilized and lead to more impressive benefits when they are integrated in a proper computer framework.

E. Recommendations:

Major emphasis in Phase II of the program must be placed on the integration of the separate software modules developed in Phase I into a coherent and transparent environment that can maximize the efficiency of these tools in a broad range of applications. The role of close research coordination is apparent. Through regularly scheduled events (meetings, technical reports, newsletters, symposia) and informal frequent contracts among individual researchers, coordination must be increased. Likewise, the volume of results expected to be generated in the near future must be prompt and effectively integrated within the DICE system. The electronic communications network among all the participants in the program and the planned video conferencing facilities must be completed in Phase II to enhance extensive, continuous technical coordination.

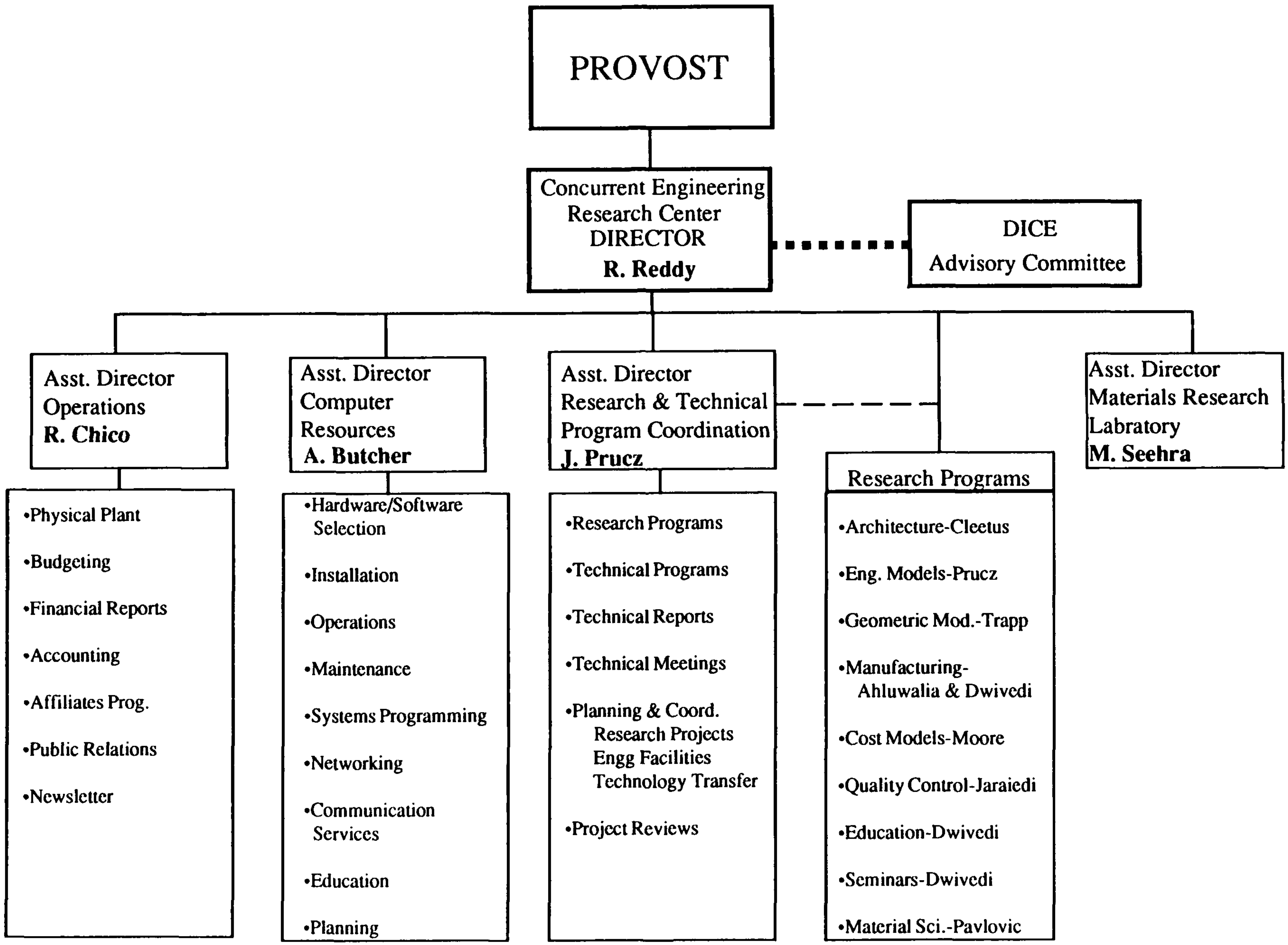
F. **Publications:**

Concurrent Engineering Research Center, West Virginia University, "Proceeding of the First National Symposium on Concurrent Engineering", May 23-24, 1989, Morgantown, WV.

G. **Hardware:**

None

CERC Organizational Chart



DICE PROGRAM

Task 3.2.3 Curriculum in Concurrent Engineering **West Virginia University**

A. Objectives:

The manufacturing world is extremely competitive and in order to be competitive, America must make products that are high in quality and flexibility, responsive to the market and manufactured in less time. To achieve this goal, attention must be focused on downstream conflicts during the early stages of manufacturing. Certainly, concurrent engineering is the most effective way to achieve this objective and produce these results.

In achieving this goal, designing and implementing concurrent engineering curricula is essential to teaching future engineers these concepts. Nationally, there is still a lack of emphasis in higher education in the general area of manufacturing. In fact, there is no university now offering formal courses on Concurrent Engineering. West Virginia University is the first university to take the initiative in developing a series of undergraduate and graduate courses in concurrent engineering. For the first time this past Spring, a graduate course on concurrent engineering was offered as part of this task in the DARPA Initiative in Concurrent Engineering (DICE).

B. Approach:

The current concurrent engineering course offered speakers, from different manufacturing areas with varying backgrounds, to explore major topics in the field. Course content included:

- An Introduction;
- Concurrent Engineering: A Concept;
- Design for Manufacturability/Assembly;
- Data Representation;
- Information Architecture;
- Models;
- Quality;
- Function;
- *Deployment*;
- Process Planning;
- Cost Models;
- Implementation of Concurrent Engineering;
- Individual and Group Projects;
- Case Studies, and Research Paper Discussions.

Design for manufacturability/assembly and other such related topics were taught by Dr. S.N. Dwivedi. Data Representation and Information Architecture Prototypes was taught by Dr. Joseph Cleetus; Process Planning was taught by Dr. Gopalkrishna; Dr. Robert Creese discussed cost models; and Dr. M. Jaraiedi lectured on Quality Function Deployment.

Guest speakers from the industrial sector and federal agencies included:

Lada Zajicek, Manager of Apple Computer, who discussed several case studies related to design for manufacturability and assembly; Clark

Preston, Manager, IBM Flexible Automation, who outlined real world implementation of Concurrent Engineering and Flexible Automation and Dr. Ranga Komanduri, Deputy Director, Division of Computer Integrated Engineering at the National Science Foundation (NSF), who presented recent research in the areas of machining and metal forming, with a special emphasis on the material characteristics. The course also involved individual and group projects related to DFM/A and Concurrent Engineering.

Other areas in Concurrent Engineering were also discussed in group sessions, including the different roles of product designer, manufacturing engineer, marketing personnel, etc. in Concurrent Engineering.

C. Technical Results:

As a result of the course, students received insight into the production of better quality products within a shorter production cycle using concurrent engineering concepts. In the present educational system, design, manufacturing, and material quality control are taught as different courses in different departments. As a result, most students do not experience these concepts simultaneously. They do not acquire the integrated approach so crucial to considering these factors concurrently. This course provided an opportunity to students to observe and experience such an integrated approach.

D. Conclusions:

This course provided an interdisciplinary interaction among design engineers, process engineers, quality managers and marketing personnel. With this course, a young graduating engineer should be in a better position to visualize how design and manufacturing involve other important factors such as materials, quality, reliability, serviceability, and testability. As a result, these engineers learned the limitations associated with each of these factors and; therefore, became more well-rounded and more of a "renaissance engineer".

E. **Recommendations:**

This concurrent engineering course was taught here for the first time in the Spring of 1989. It is hoped that it will be taught again during the third phase of the DICE Program. Hopefully, the course will be more refined and ready for distribution as a model for other universities interested in offering courses of this nature.

Moreover, a workshop is planned during the third phase of the program at the CERC for faculty members interested in teaching Concurrent Engineering. The workshop will be organized in a course sequence and readied as hard copy in the form of class notes to interested faculty members. In addition, a brief description of the course content, course materials, and methods of instruction will be published in professional journals for additional dissemination and to enhance its development. A new course in the "Design of Manufacturability" is being taught at West Virginia University in the Fall Semester, 1989.

F. **Publications:**

None

G. Hardware:

Since the course was not a research-oriented task, no hardware and/or software was purchased or developed.

Task 3.2.4 Faculty/Graduate Students West Virginia University

A. **Objectives:**

This task was initiated to attract experts in Concurrent Engineering from industry, government agencies and academia. These professionals are expected to reside at CERC for various periods of time in order to present seminars, conduct research and provide technical advice to CERC graduate students, faculty and full-time staff.

B. **Approach:**

A comprehensive list is being compiled of technical experts who are currently active in the field of Concurrent Engineering. This directory will focus especially on those individuals who have expertise related to the issues and directions addressed in the DICE program. Such professionals will learn about CERC's objectives, work plans and results, as invited guests to national symposia, seminars, technical meetings, demonstrations or tours. Close cooperation and involvement with CERC will subsequently evolve from direct discussions between outside experts and appropriate CERC personnel when common research needs, interests and expertise are identified and agreed upon. Arrangements will need to be worked out to the mutual benefit of CERC's research programs and its guests.

C. Technical Results:

Professional relationships have been established with scientists specialized in research and development areas related to Concurrent Engineering. Both informal exchanges of information between individual researchers and more formal invitations to these scientists to be guests or consultants have been extended. These contacts represent a broad spectrum of specialties and professional organizations either formally affiliated with the DICE program or who are potential CERC customers, sponsors, advisors or research associates. There have been a number of mechanisms utilized to initiate and expand such professional contacts:

1) Invitations have been extended to outside experts to give technical presentations as guest speakers of CERC's weekly seminar series. A chronological list of these speakers, their affiliations and presentation topics is given below:

GUEST SPEAKERS TO
CERC's TECHICAL SEMINAR SERIES IN PHASE I

DATE	NAME	AFFILIATION	TOPIC
10/18/88	Mr. Yacavone	GE/AE	"EngineeringTomorrow's Propulsion Today"
2/14/89	Dr. Komanduri	NSF	"Interdisciplinary Aspects in Manufacturing"
2/21/89	Dr. Upadhyay	G.E. Medical Systems	"Plasma Processing of Ceramic and Composite Materials"
2/28/89	Mr. Buenzli	American Cimflex	"Concurrent Engineering for the High Density Electronics Domain"
3/14/89	Dr. Mistree	University of Houston	"Concurrent Engineering: A Decision-Based Approach"

3/21/89	Mr. Zajicek	Apple Computer	"DFM Support for Simultaneous Engineering Electro Mechanical Design"
3/28/89	Dr. Prasad	Electronic Data Systems	"Parametric Programming for Software Prototyping"
4/11/89	Mr. Preston	I.B.M.	"CIM - A Plant Management Perspective"
4/18/89	Dr. Meyrowitz	Office of Naval Research	"ONR-Research in Intelligent Systems and Distributed Artificial Intelligence"
4/24/89	Dr. Das	Alabama A&M University	"Knowledge Based Systems- A Metalevel Reasoning Architecture for Explanation Generation"
5/16/89	Mr. Buenzli	Cimflex Teknowledge	"Concurrent Engineering For Electronic Assembly"
7/10/89	Dr. Raj Singh	GE/CRD	"Development of High Temperature Ceramic Metrix Composites"
7/17/89	Dr. Fabrycky	Virginia Institute of Technology	"Computer Aided Concurrent Engineering Design"

2) Invitations were extended to scientists from government, industry and universities to attend the First National Symposium on Concurrent Engineering on May 23-24, 1989, to interact with DICE researchers and be advised on CERC's current research and plans, and

3) Bilateral exchanges of visits and technical information between CERC and outside researchers have explore opportunities of collaboration in specific areas of interest to the DICE program. Such professional contacts have included exchanges with Dr. Sriram from MIT, Dr.Bloom from NIST, Dr. Ling from Columbia University and others.

D. **Recommendations:**

The CERC continues to actively pursue the development of close professional interactions with outside researchers in order that the Center be recognized as the National Research Center on Concurrent Engineering. It is imperative to utilize the expertise currently available elsewhere in the nation in a systematic and effective manner.

E. **Publications:**

None

F **Hardware:**

None

ARCHITECTURE FOR CONCURRENT ENGINEERING

DICE PROGRAM

Task 3.3.1 Information Content and Flow GEAE

Objectives:

A needs analysis was performed that defined the information required by the Concurrent Engineering concept and its flow sequence during the conceptual design phase of the product development cycle. The intent of this work was to allow more designs to be considered, in less time, during the conceptualization phase of design and prior to the down selection process for the detailed design phase. These needs were also developed with key inputs identified for materials, manufacturing, processing, and quality technologies. Both the "as-is" and "to be" (with Concurrent Engineering methods in place) were developed and documented.

Approach:

Several joint meetings were held with both design engineers and computer systems personnel at which time the information content and flow required to perform Concurrent Engineering was storyboarded. In addition, the IDEFO process was used to further define the detailed information content and flow requirements. Knowledgeable individuals from design, manufacturing, materials, and computer systems were consulted. Information was also provided to other DICE Program groups as requested.

Technical Results:

Information Flow and Content for Concurrent Engineering has been defined for the initial stages of conceptual design. This has been accomplished with very early inputs being required from materials, manufacturing, and quality personnel in order to develop concurrency. This information has been documented and is believed to be generic enough in nature that the architecture developed to meet these needs could be applied to other industries and products. However, minor changes or additions may be needed to exchange data among different systems and different applications that may exist and be specific to a particular industry.

Conclusions:

This Task is one of the most important ones of the DICE Program since an architecture that does not meet the specific needs for required information flow and content is of little use to anyone, especially the end-user. Having the end-user's input in this phase of the Program is absolutely necessary and most valuable to the overall success of the architecture. In addition, it increases the chances for acceptance by the end-user and makes him an early contributor to the development of a concept (i.e., Concurrent Engineering) and architecture. The fact that this is done by using real product development problems also lends credibility to the work that is accomplished by the computer systems/architecture personnel. End-user involvement must continue and increase as DICE matures.

Recommendations:

The use of real product development problems must continue and even expand as the architecture for Concurrent Engineering continues to evolve. As this DICE effort moves from the conceptual design phase and into the detailed design/product development phase, the information content and flow required will increase in volume and complexity. For this reason, the continued use of real product development problems must continue in subsequent stages of the DICE Program. In addition, input from the potential end-users of the computer architecture must continue in order to make the architecture that is developed meaningful and readily accepted (by the user) if Concurrent Engineering is to succeed.

Publications:

N/A

Hardware:

N/A

Task 3.3.1 Information Content and Flow GEAE

Objectives:

The goals for this task are to understand the information requirements of the engineering users attempting to manage engineering information in a concurrent environment. The authors approached this task from their respective experiences in managing the data associated with a large engineering computing environment.

Approach: Three approaches were used to carry out the above goals:

1. Meetings were held with other DICE team members to understand the approach promoted by the team along with the computing systems planned to support the added information management made necessary by concurrent engineering activities.

2. The technical literature was surveyed to understand the various requirements reported by other research groups concerned with concurrent engineering.

3. Meetings were held with engineering users to determine the practical needs of the practicing engineers along with the possible cultural and technical impact of concurrent engineering on designers.

Technical Results:

1. The Redbook of Functional Specifications for the DICE Architecture was published, which included our contribution of the GEAE Long Term Functional Requirements for Concurrent Engineering Summary.

2. Assisted GE-CRD, GEAE Lynn, and WVU in defining the functional requirements for the DICE demonstrations.

3. Used the planning of the DICE demonstration to expose engineering users to the core concepts of engineering data management and concurrent engineering.

4. Determined that the description of the data and computing environment which had been designed into the GENIE Catalog at GEAE before the beginning of the DICE contract was a suitable means for achieving several key objectives of the DICE program.

Conclusions:

1. The meetings held with the engineers showed a general lack of engineering data management facilities and tools.

2. The same meetings pointed to the time consuming nature of data management in engineering. This suggests that the automation of the data management activities will have a large pay off for the engineering users.

3. At present, the lack of data management tools and practices have the following serious shortcomings:
   a. The ability to retrieve the work of others is linked to the personal knowledge of the author.
   b. The retrieval of the work of other contributors is difficult, time consuming, and in general, requires manual transcription of the data from one format to another.
   c. The comparison of the data generated across time and individuals is at best tedious, very much time consuming. Nonetheless, such comparisons are essential to the reuse of the Corporate Engineering know-how.
   d. On occasion, the wrong data is used in studies.

4. In general, engineering users have difficulties in defining the engineering process used to carry out specific design tasks.

Recommendations:

1. Involve engineering users in engineering process definition: The definition of the engineering process allows users to develop a greater appreciation for the discipline required to manage engineering data. This education effort also drives home to the engineering users the pay off of such a discipline.

2. Engineering data management systems must be flexible: Engineering data management systems must have the flexibility required to accommodate the ever changing nature of the design process. Changes in the design problem (redesign versus new design) imply different sequences of design steps with different data retrieval capabilities.

3. Data definitions must be available: As data is shared among users, users must be able to access exact definitions of the data exchanged as well as they must be able to point quickly to the proper data set containing the information they seek.

4. <u>Random access of engineering data</u>: Providing random access to engineering data is one way of maximizing the value of the data to the engineer. Such organization supports all data access needs.

<u>Publications</u>:

None

<u>Hardware</u>:

None

DICE Programs

Task 3.3.2.1 Design Knowledge Representation **Carnegie Mellon University**
Task 3.3.2.2 Constraint-Based Models
Task 3.3.3.8 Local Architecture and Designer's Workstation

Objectives

The long-term goal of the Design Fusion project is to develop a design system to assist designers in creating designs that meet their function, material, cost, and quality requirements while simultaneously meeting the constraints imposed on the design throughout its life-cycle: planning, production, distribution, field service, etc. Our approach is to fuse the functional requirements of material and mechanical design with the life-cycle constraints.

For the representation task, our primary objectives are to define and create the central representation for the design based on the Noodles geometric modeller and to create the system for feature definition and extraction. For the constraint task, our objective is to construct and manipulate design models based on an underlying constraint and feature-based representation. In the context of engineering design, large numbers of complex constraints must be satisfied to complete a design task. Since no general solution strategy for solving arbitrarily complex constraints exist, our goal is to provide the designer with insights about the design problem at hand. Our research on constraints-based models for concurrent engineering has been concentrated in two major areas, the identification of critical constraints and the problem of *constraint solution planning.*

Finally, the objective of the local architecture task is to develop a problem-solving architecture that will opportunistically refocus attention of the design process onto the perspective whose knowledge currently dominates. In particular, our effort has been directed towards creating an architecture where the issues of focus could be investigated.

Approach:

Design Fusion uses a blackboard architecture to provide a heterarchical organization for multiple life-cycle perspectives to view and comment on a shared representation of the design. It is based on three underlying concepts:

1. Integrating life-cycle concerns through the use of views from multiple perspectives, where each perspective represents a different life-cycle concern such as manufacturing, structures, materials, *etc*;
2. Representing the design at various levels of abstraction and granularity through the use of features, where features are the attributes that characterize a design from the viewpoint of any perspective;
3. Generating and pruning the design space through the use of constraints.

Figure 1 depicts the organization of the system. The designer and four life-cycle perspectives are represented: materials, manufacturing, structures and aerodynamics. Each perspectives encapsulates expertise for the relevant specialty. They act as proxies, providing feedback to the designer based upon qualitative and quantitative models. The perspectives monitor the design through the blackboard, which represents the design in terms of geometry (and topology), constraints, and a database of qualitative design features and quantitative parameter values derivable from the geometry.

A non-manifold representation scheme for geometry provides a neutral representation of the design. The perspectives extract features from the geometry, and calculate parameter values from the features. Pinilla et al. describe a formal method to describe shapes and features that relate closely to very low-level models of objects. This provides an interlingua for the perspectives to communicate about an artifacts form.

Multi-Layered Architecture

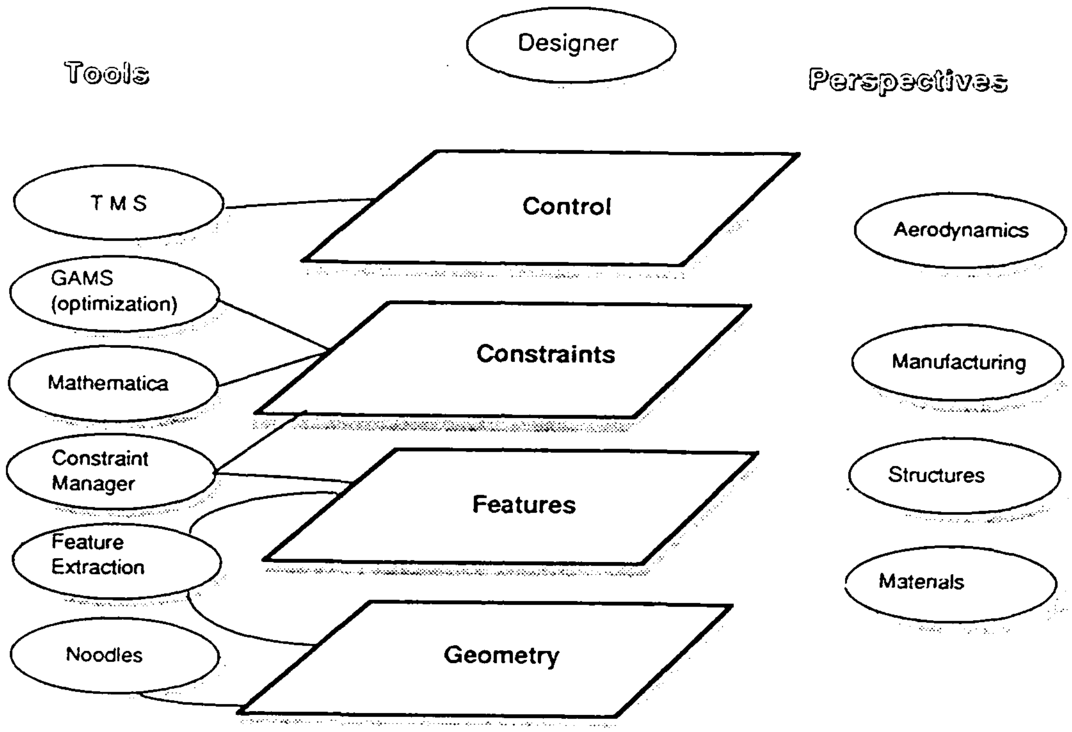


Figure 1: Flow of control in Design Fusion

Parameters extracted from the design are recorded in the database. A truth-maintenance system (TMS) tracks the dependency between the data in the database and the geometry from which it was derived. It is used to detect when assertions in the database are no longer supported by the current geometry. Revisions are also explicitly represented in the database as relations between assertions.

Perspectives monitor database changes, and are invoked to comment on the design at appropriate times. For example, in Figure 1 the structural perspective is monitoring the design for sharp root radii. Likewise, the aerodynamic perspective is monitoring that no other perspective comments on sharp root radii.

When a monitoring condition is satisfied, the life-cycle perspective is invoked. It may express preferences about the design by commenting to the designer or imposing constraints on the design. For example, when the structural perspective recognizes a blending surface in the design, it can impose a constraint on the sharpness of the surface. When this constraint is violated, it can comment to the designer its concern about radii sharpness. Likewise, when the aerodynamic perspective detects an airfoil in the design, it can express its preferences on the radii of blending surfaces by imposing constraints on the design. It is important to note that aerodynamics notion of *sharpness* and structures notion of *sharpness* need not be the same. Each perspective defines its own *sharpness*-feature through the feature extraction language. These perspective-specific features are extracted from the neutral geometric representation.

A constraint manager coordinates constraints imposed by the designer and perspectives. It has three functions. First, it detects violated constraints. When a constraint violation occurs, the perspective that imposed the constraint is invoked to process the violation. Second, it can propagate known values to determine consistent assignments for other design variables. Lastly, the constraint manager facilitates opportunistic reasoning. There are times during the design process when the designer desires to perform some analysis, but all the inputs are not known or have not been specified. Through constraints, these inputs can be identified, and plans generated to acquire the necessary design parameters.

Together, these components form a problem-solving architecture that meets the first objective described above. It permits opportunism through a shared representation of design assertions and constraints. Multiple perspectives encapsulate qualitative and quantitative models to provide the designer feedback from life-cycle proxies. The system remains focused by observing changes to a database of facts about a design coordinated by the designer.

Technical Results

There are two types of results for the architecture task. First, it must be noted that the architecture is both the glue of the design fusion system and a testbed. As such, a version must exist before any experimentation can be performed. We spent the first year putting together the first version and integrated the perspectives and designer's interface. We are now ready to experiment with the architecture itself.

The second result is that the architecture integrates, for the first time, a four plane representation of design knowledge:

1. Control.
2. Constraints.
3. Features.
4. Geometry.

CAD systems are either limited to a geometric representation of a design (e.g., Computervision, etc.) or to a constraint-based parametric representation (e.g., ICAD, Wisdom). Our architecture provides a full integration of geometry, features, constraints, and control.

For the representation task, we have written a preliminary, feature-based interface for turbine blade design in which the designer can directly manipulate the design based on features and parameters. The designer interface is linked to the Noodles representation, so that, as the design is created and modified from the designer's point of view, the underlying Noodles geometric representation, interpretable by all perspectives, is generated.

We have also written the specifications for a general feature recognition algorithm based on graph grammars and incremental pattern matching. This algorithm has not been implemented in the current software, but will be in the next version. In the current version, feature extraction is implemented as parameter extraction.

We have tested the interval-monotonicity approach with a set of aerodynamic constraints and have successfully identified critical constraints. The approach was used to find which variables go to their bounds for the given optimization criteria. This information can help the designer eliminate the need to run a full-blown optimization every time a change is made to the design. This result, however, is valid only within the intervals of the relevant specifications.

The approach also provides the designer with insights about how the constraints affect the main objective. Armed with this information, the designer is in a better position to make tradeoffs between optimizing the objective (aerodynamic efficiency in the test case) and relaxing constraints. As the algorithm does not require global monotonicity, it is applicable to realistic design problems.

There are two parts to the solution planning research. The first part deals with solution planning to find shortest paths through constraint graphs. The algorithm tries to reduce the requirement for simultaneous solution of variables. The second part is aimed at identifying, for the designer, those variables which are the best ones to guess during problem solving. The algorithm is aimed at finding variables that can be guessed in order to solve the entire constraint graph serially.

We have identified a heuristic approach to finding variables which, when guessed, can render the constraint graph serially decomposable. The basic idea is to find the "most connected" variable in the constraint graph. This is done by finding the innermost strong components in the graph and then picking the variable with the maximum number of dependent variables which lie outside the strong component. This is a heuristic measure of how critical it is to know the value of the variable before other parts of the design can be determined.

Conclusions

One conclusion to date is that a multi-planar approach to the integration of design representations is feasible, as long as the appropriate mappings among planes can be performed and maintained. Feature extraction is a necessary part of the mapping from the geometry plane to the feature plane, while truth maintenance is necessary to propagate the effects of design changes.

The main idea is that constraints provide a uniform representation for a variety of downstream concerns. This idea, coupled with solution planning and the propagation of intervals makes it possible to find potential violations, critical constraints and redundancies even before the design is complete. We express constraints as algebraic relations among feature parameters (e.g. hole diameter, wall thickness, stress level). Before the design process begins, an upper and lower bound is assigned to each parameter. As the designer makes decisions, the intervals on the parameters are narrowed to smaller intervals or down to single valued points. As these changes occur, the intervals are propagated through the constraints to other parts of the design. During this process, if any constraint is violated it is reported to the designer. As we use an uniform representation for constraints among all the life-cycle concerns, the system does not differentiate between up-stream and down-stream constraints. Algorithms for finding inconsistencies and redundancies are applied uniformly among all constraints and consequently, the system is equally likely to find violations among the up-stream constraints as among the down-stream constraints. It is for this very reason that our approach achieves concurrency.

By combining interval based methods with monotonicity and convexity, we have developed an useful approach for

- propagating Interval values between constraints.
- the identification of conflicts before all the variables have been set to point values.
- the recognition of critical constraints for the given optimization criteria.
- providing the designer with feedback about critical variables.

In addition, we have found that strong components in the constraint graph represent inter-dependence among variables. A highly cyclic constraint set can be reduced to a serially decomposable set by simply identifying those variables which are most strongly connected. The designer can decide to pick values for such variables, thereby making the constraints serially solvable. This makes it possible for us to solve constraints sets which are too complex for even the best known solution packages.

Recommendations

The availability of the architecture now permits us to explore the objectives described earlier. Consequently, our recommendations are obvious: that we now explore methods of control in design. Our first approach will be to use knowledge of constraints to identify portions of the design where attention should first be placed. Recent results provide methods of how to measure where in a constraint network a decision should be made next. We will use this knowledge and the knowledge generated in the constraint task to focus the attention of the design fusion system.

Our development of a robust algorithm for incremental feature extraction is very promising, but further work on efficiency and feature definition is necessary. In addition, more research is needed to define the paradigms for human-computer interaction for the user of the design fusion system.

In our constraint work, the following research areas need to be investigated:

1. We need to unify (a) interval propagation, (b) interval refinement, (c) constraint relaxation, and (d) the identification of critical constraints and variables.
2. Solution planning of constraints has to be extended to both uni-directional and bi-directional constraints. This will unify solution planning of equations and black-box functions (e.g. FEM analysis).
3. In constraint graphs it is sometimes possible to transform variables algebraically to make a cyclic constraint graph serially decomposable. These transformations will be have to be based on algebraic transformations of variable. A network of constraints, for example, relating fan pressure, flow, and blade stress to hub diameter, tip diameter, and rotational speed can be made acyclic by expressing the constraints in terms of disc area, tip speed and effective diameter. Transformations of variable of this sort have important advantages not only on the structure of constraint networks but also on a the establishment of a nominal design point.
4. Strategies for constraint negotiation among life-cycle perspectives need to investigated. How will the negotiation take place? Will constraints be relaxed? If so, will the relaxations be based on utilities?

Publications

Navinchandra, D., *Constraint Management in Concurrent Design*, in the working notes of a workshop on concurrent engineering held at the International Joint Conference of Artificial Intelligence organized by D. Navinchandra, August 1989.

Navinchandra, D. and Rinderle, J. R., *Interval Approaches to Concurrent Evaluation of Design Constraints*, To appear in the proceedings of the Concurrent Engineering Symposium, ASME Winter Annual Meeting, December 1989.

Pinilla, J. M., Finger, S. and Prinz, F. B., "Shape Feature Description and Recognition Using an Augmented Topology Graph Grammar," *Proceedings of the 1989 NSF Engineering Design Research Conference*, University of Massachusetts, Amherst MA, June 11-14, 1989, pp 285-300.

Safier, S. and Fox, M.S., "The Role of Architecture in Computer-Assisted Design Systems", *Proceedings of the 1989 NSF Engineering Design Conference*, University of Massachusetts, Amherst MA, July 11-14, 1989.

Hardware

No hardware or software was purchased. The prototype software system for the Design fusion workstation was written. This software includes the blackboard and the truth maintenance system, the designer interface, geometric modeling interface, feature recognition, aerodynamic analysis and optimization, and the constraint management system.

DICE PROGRAM

Task	3.3.2.3	Information Management Data Base	RPI
Task	3.3.2.4	Graphics Interface & X-Windows	RPI
Task	3.3.3.9	File/Database Translator Generator	RPI

1. Overview

This final report describes the work accomplished by RPI during Phase I of the DARPA Initiative in Concurrent Engineering. During this phase RPI had responsibility for implementing an object oriented database system to support concurrent engineering. This responsibility is to continue in Phase II.

The report given here has been divided into five sections. The second section contains a descriptive summary that outlines the progress made by RPI during Phase I. The third section contains a summary by task number (as allocated by the RPI/DICE contract) of our achievements within each task. The fourth section describes the papers and reports produced by RPI during Phase I. The last section describes the software modules produced by RPI for the DICE architecture. These modules were used in the DICE architecture demonstration given at the Concurrent Engineering Research Center (CERC) on July 25th.

2. Descriptive Summary

RPI was selected as a participant in DICE because it had previously produced an object oriented engineering database system called ROSE. This database system had been tested at General Electric Corporate Research and Development and used to produce the CHIDE user interface management system.

The version of ROSE that existed at the start of the DICE contract had a serious weakness, however. This weakness was that it had to be programmed using a language that was unique to ROSE. While this language had some good properties with respect to rapid prototyping, it was unlikely to be widely adopted by groups outside of DICE. Therefore, in October of 1988, the dramatic (for RPI) decision was made to not use ROSE as it then existed as the database system for DICE. Instead, Objective-C was chosen as the "backbone" language for DICE, and RPI was asked to produce a ROSE like database system in Objective-C.

RPI responded to this decision in two ways. First, it produced a database system that is language independent. Second, it took the opportunity to improve the concepts within ROSE so that it could give better support to Concurrent Engineering. As a consequence of the first decision, ROSE is now a database system with a powerful Objective-C programming environment, and it has the potential to have powerful programming environments in C++, CLOS and other object oriented programming languages. As a consequence of the second decision, ROSE now has some powerful constructs for concurrent engineering as detailed below.

The new version of ROSE continues to have good performance because it represents designs as sets of objects stored in files. These files can be down-loaded into a users workstation at the start of a transaction and up-loaded to a file server at the end of a transaction. Therefore, during a transaction (engineering design session) the data in a ROSE database can be processed at almost main memory speeds. The result is a database system that is dramatically faster than the traditional (relational) database systems that process data at secondary memory speeds.

Traditional database systems process data at secondary memory speeds because applications such as banking require the result of an operation to be immediately visible to all users. This requirement is necessary because otherwise two users might make debits to the same bank account simultaneously that would not be allowed if they occurred serially. Fortunately, such immediate data sharing is usually not required in engineering applications. Instead, in these applications each user can edit and perform experiments on a private copy of the database. If a set of experiments are successful, then the results can be "published" by putting the designs produced back into the global shared database.

The new version of ROSE gives good support to concurrent engineering because it has the ability to allow multiple engineers to operate on a design concurrently. It can do this because it captures the changes that an engineer makes to a design as a script of editing commands. This script is used to make applications reliable in case of a system failure, and it is used to communicate the changes made by one engineer to other engineers. In particular, if two users check out a design concurrently, ROSE can capture the changes they make, communicate those changes to others, and identify any conflicts between the changes. It cannot, however, resolve any conflicts that occur. The potential for creating conflicts is the penalty that must be paid for editing designs concurrently. RPI is currently developing various protocols for controlling the scope of conflicts including: forcing users to edit designs serially; enforcing territorial agreements between users with security measures (so that they can only change a subset of the data); and treating all changes as tentative unless they have been approved by a lead engineer.

In essence, the scripting feature of ROSE gives the DICE architecture the ability to support "fine grained" concurrency where users edit the same design concurrently, as well as "coarse grained" concurrency where users edit different designs concurrently. Fine grained concurrency is not supported in traditional engineering systems. Therefore, research will be needed to determine when it is most effective, and how it can best be controlled. Since it gives DICE users the ability to perform in parallel operations that previously had to be done serially, there should be occasions (such as meeting tight deadlines) when it is exceedingly useful.

3. Task Summaries

3.1. Task 1.0 Concurrent Engineering Research Center

All of the software modules described in the last section of this report, except for the ROSE++ module (C++ version of ROSE), are available for delivery to the Concurrent Engineering Research Center. The ROSE++ module will be available as soon as its first version has been debugged.

On July 25th, the modules were used in the DICE architecture demonstration given to DARPA. In this demonstration, General Electric Corporate Research and Development showed how the DICE architecture could be used to shell several existing engineering applications including an engineering design system, an engineering analysis system and a spread sheet. The ROSE system was used to construct and control these shells. Specifically, ROSE was used to read and write data in the FORTRAN name list format to communicate with the engineering design system and the analysis system, and its code was integrated with that of the spread sheet to produce a tightly coupled system.

In the demonstration, the ROSE data translation tool was demonstrated by showing how data could be read and written from the PPO database produced by GE into the FORTRAN name list format. The Concurrent Editor for Objects (CEO) produced by RPI as a programmers tool was used to browse the PPO database. Concurrency and scripting was demonstrated by showing how changes could be communicated between a user using the CEO to browse the PPO, and a user using the spread sheet to run an analysis system.

The relational database interface was not demonstrated on July 25th because of the non-availability of a relational database system on the necessary machines at CERC.

3.2. Task 2.2.3 Information Management Database

The requirements of the Information Management Database are being filled by the development of ROSE. As previously explained, because it has been developed specifically for concurrent engineering, ROSE is faster and more flexible with respect to concurrent engineering applications than traditional database systems. The ROSE database system as it exists at the end of Phase I has the following abilities:

1. It is fast.
2. It has a powerful Objective-C programming environment and it has the potential to have powerful programming environments in C++, CLOS and other object oriented programming languages.
3. It has the ability to support fine grained concurrency where multiple users edit the same design concurrently.
4. Its applications can be reliable in the sense that they need not loose data in the event of a system failure.
5. It has an interface to a relational database. (In the DICE architecture, this database is to be used as a dictionary for the design files stored in ROSE. A user will be able to code a high level query in SQL to find a design, and the relational database will return the names of the ROSE files containing designs that satisfy the query.)
5. Tools exist to edit and manipulate its databases.

In the future phases of DICE, work is needed to realize the other programming environments, and to add features for security and conflict resolution (between designs and design versions). In addition, work will continue to be needed to make ROSE an easier to use, better system. The volume of this work should not be underestimated. As any system grows larger it becomes more difficult to modify.

3.3. Task 2.2.4 Graphics Interface and X Window Standard

The original intent of this task was that RPI should develop a powerful easy to use X windows interface for the original ROSE system. However, when the direction of ROSE was changed to Objective-C, this goal became redundant because Objective-C already had a good interface to X windows. Therefore, this task was modified and RPI has used it to produce easy to use, graphics based tools for the ROSE system. Specifically, RPI has produced the Concurrent Editor for Objects (CEO), and an AND/OR tree editor for ROSE data structures.

The Concurrent Editor for Objects is a browser like tool that can be used by programmers that are knowledgeable about ROSE to browse and edit ROSE databases. The tool is concurrent because it uses the ROSE scripting mechanism to allow multiple users to edit the same design concurrently. As it is currently configured, the CEO can be used to browse through the objects in a ROSE database editing their values. A powerful graphics interface exists to help the user with this editing. In addition, the tool contains an interpreter that allows the user to send any message to any object managed by ROSE. (It cannot be used to send messages to Objective-C objects not managed by ROSE, however.)

The AND/OR tree editor is a tool that allows ROSE users to define ROSE data structures graphically. The tool is called the AND/OR tree editor for historical reasons, but the name is now a poor one because it implies that ROSE can only represent hierarchical, tree-like data structures. In fact, ROSE can represent any recursive or non recursive data structure that can be represented using the records, unions and lists found in most programming languages. The AND/OR tree editor allows users to describe ROSE data structures. The result of the editor is a ROSE file that describes these structures. An application program can use the data structures by reading the file.

3.4. Task 2.2.5 File/Database Translator Generator

The File/Database Translator generator is an ambitious tool that will allow a user to convert ROSE databases between data structures. In DICE, the intent is for this tool to be used to convert designs between the formats needed for different tools. For example, an analysis tool might produce data that can be used by a process tool provided it is reformatted to include slots for new attributes. The File/Database Translator generator will allow the user to describe which attributes should be added to the data graphically, and it will produce a program that when applied to the output of the analysis tool will produce the input required by the process tool.

A first version of the File/Database Translator was produced for the original ROSE system. Producing a translator for this system enabled us to get valuable experience while the concepts for the new ROSE system were still being developed. A new translator is now being produced for the new ROSE system. An early version of this translator was used in the DICE architecture demonstration on July 25th to format data for the engineering analysis and design tools linked to ROSE (and the spread sheet) in that demonstration.

4. Papers and Reports Produced in Phase I

1. M. Hardwick, D. Spooner, B. Downie, E. Hvannberg, D. Loffredo and A. Mehta, "The ROSE-IC Library: A Database for Cooperative Work and Concurrent Engineering in Objective-C", Version 1.0, Technical Report, Rensselaer Design Research Center, June 24, 1989.
2. M. Hardwick and G. Samaras, "Using a Relational Database as an Index to a Distributed Object Database in Engineering Design Systems", *2nd International Conference on Data and Knowledge Systems in Manufacturing and Engineering,*, IEEE Computer Society, October 1989, to be published.
3. D. Spooner, D. Sanderson and G. Charalambous, "A Data Translation Tool for Design Databases", *2nd International Conference on Data and Knowledge Systems in Manufacturing and Engineering,*, IEEE Computer Society, October 1989, to be published.

5. Software Modules Produced in Phase I

These modules are documented in a report produced for the DICE architecture demonstration of July 25th and attached here as an Appendix.

DEMONSTRATION

ROSE-IC

Martin Hardwick, Blair Downie, Ebba Hvannberg, Dave Loffredo, Alok Mehta, David Spooner

Rensselaer Polytechnic Institute
Troy, New York 12180

Description

ROSE is an object-oriented engineering database system supporting concurrent editing of objects, and providing support for cooperative engineering work environments. Its current implementation, ROSE-IC, is a set of classes in the Objective-C programming language. As a result, it provides persistent objects for Objective-C programs, along with version control for persistent objects, scripting of changes made to persistent objects, and other functions related to managing concurrent editing and cooperative work environments for persistent objects.

While the first version of ROSE is implemented in Objective-C, the ROSE concepts are portable to other object-oriented programming languages. Hence future versions of ROSE for other languages such as C++ and the Common Lisp Object System (CLOS) are expected.

ROSE-IC will be demonstrated via several tools built using ROSE-IC and Objective-C. One of these tools manages scripts of changes made to ROSE-IC persistent objects. A second tool allows browsing of ROSE-IC persistent objects. A third tool converts ROSE-IC persistent objects into Namelist and Laser formats for exchange of data with applications programs using these data formats.

Contribution

The major contribution of ROSE-IC is that it provides a flexible, high-performance database management tool with features satisfying many of the requirements of a concurrent engineering environment. No other existing database system currently supports concurrent editing of objects and cooperative engineering work environments as done in ROSE-IC. A second advantage of ROSE-IC is the many tools for facilitating data management that can be added to it because of its open architecture. This includes such tools as a graphics management system, relational finder, and data translation tool, all of which will be useful in DICE. A third advantage of ROSE-IC is the portability of the system concepts across multiple workstations and systems.

Users within DICE will benefit from ROSE-IC and its tools because it provides a rich data management environment including many concurrent engineering tools for persistent Objective-C objects -- the DICE development language. Users outside DICE will benefit from ROSE-IC and its tools because of the portability of ROSE-IC and the tools for integrating foreign applications with a ROSE-IC database.

Technical Approach

ROSE-IC is implemented as a set of classes in the Objective-C object-oriented programming language. These classes allow application programs to create, manipulate, read, and write persistent objects. The persistent objects are paged between main memory and secondary storage on demand, achieving high performance and ease of use for the ROSE-IC system.

Persistent objects in ROSE-IC can be subclassed so that behavior specific to a particular type of persistent object can be defined. ROSE-IC is responsible for maintaining the subclass information as objects are moved back and forth between main memory and secondary storage. A rich set of methods is provided for searching and manipulating persistent objects.

Status

The first version of ROSE-IC is in beta test and available to all DICE participants. This version includes all the Objective-C classes for creation and manipulation of persistent objects. Advanced functionality of ROSE-IC is under development, including scripting of changes made to persistent objects, control of parallel editing of objects, and version control. Future releases of the system will include these features.

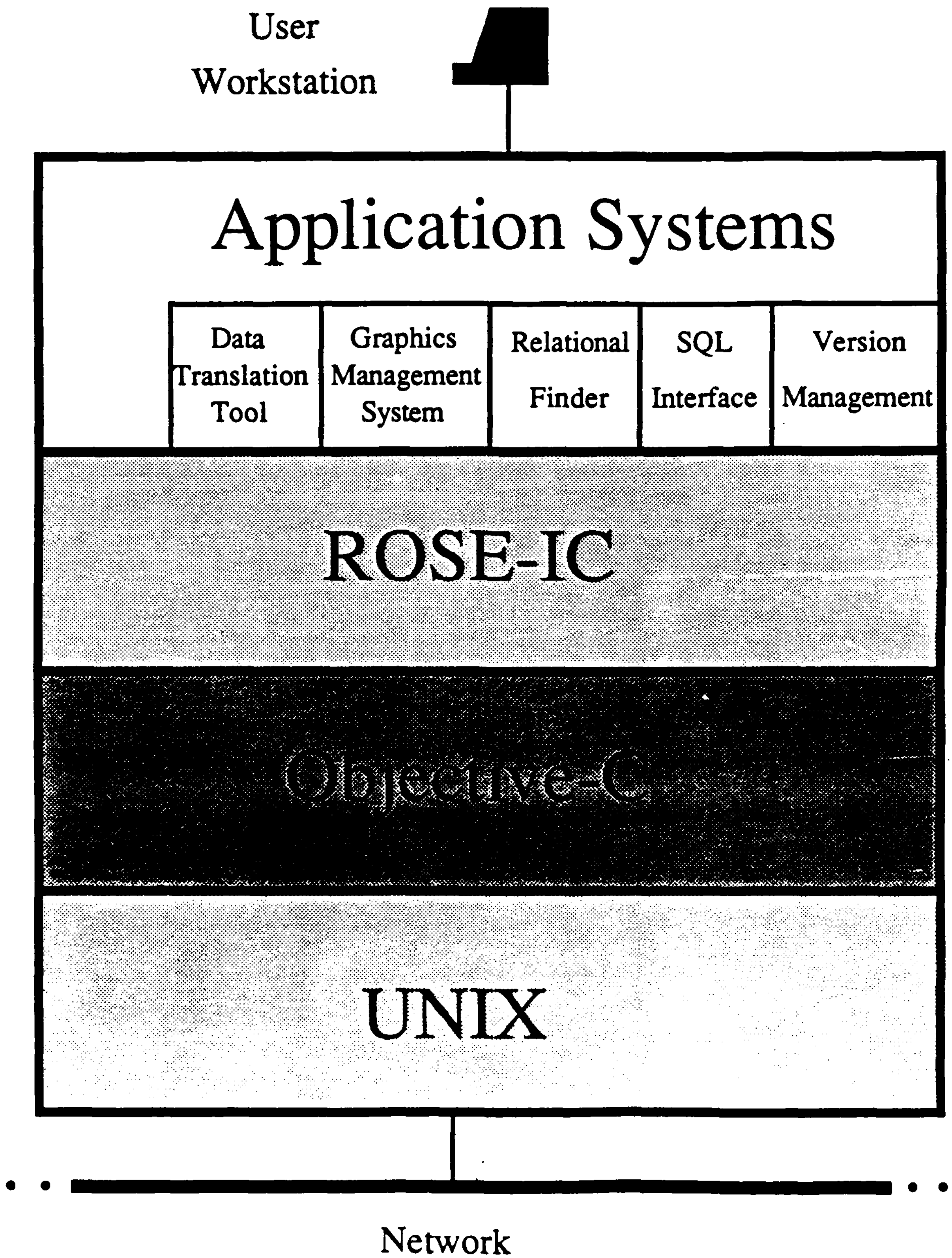
Future Work

Work for the immediate future will be devoted to implementing the advanced functionality of ROSE-IC. This includes scripting of changes made to objects, version control, and parallel editing of objects. It is also expected that versions of ROSE will be created for other object-oriented programming languages such as C++ and CLOS.

Environment

ROSE-IC currently requires only Objective-C 4.0. Its object I/O facilities are optimized for the Unix operating system, but should work (less optimally) on other systems. It has no specific hardware requirements.

ROSE-IC SYSTEM ARCHITECTURE



DEMONSTRATION

Concurrency Support for ROSE

Ebba Hvannberg and Martin Hardwick

Rensselaer Polytechnic Institute
Troy, New York 12180

Description

The scripting utility supports concurrent editing of objects and cooperative work in ROSE. Scripts provide a means to keep track of the creation and modification of objects in ROSE, and thus can capture the changes that are performed on a design. Scripts have several uses. Users on different workstations can exchange scripts to see each others changes. A user may choose to automatically make his changes public, or he may choose to publish them explicitly. Another user can retrieve the changes any time after they are made public.

Scripts can be compared to locate conflicts between two designs. Thus, users can see which objects in common between the two designs have been modified. Scripts can be used to alert users of particular changes in the database. The differences between two designs can be computed and stored as a script. Scripts can be used to integrate two designs into one. Finally, scripts provide reliability as a log of modifications made to a design in the event of software or hardware failure.

The scripting utility can be used with any application program, but will be demonstrated via the Concurrent Editor for Objects.

Contribution

The scripting utility allows multiple users to edit a design in parallel. In particular, it allows users to incorporate updates made by other users into their versions of a design, to compare changes, and to integrate two versions into one.

Technical Approach

The scripting utility is implemented as set of classes that are added to ROSE. It is implemented in such a way that the application programmer can choose to use it or omit it (i.e., generation of scripts is transparent). Scripts are communicated between users on different workstations via disk files or sockets. Sockets provide the means for instantaneous communication of scripts between users.

Status

A prototype implementation of the scripting utility has been completed. Scripts can be produced by one user and interpreted by other users. The scripting utility keeps track of what script records a producer has published, and what script records a consumer has read. Thus, a consumer of scripts can check for new changes from a producer. Scripts can be published on request by the user, or automatically when the script buffer becomes full.

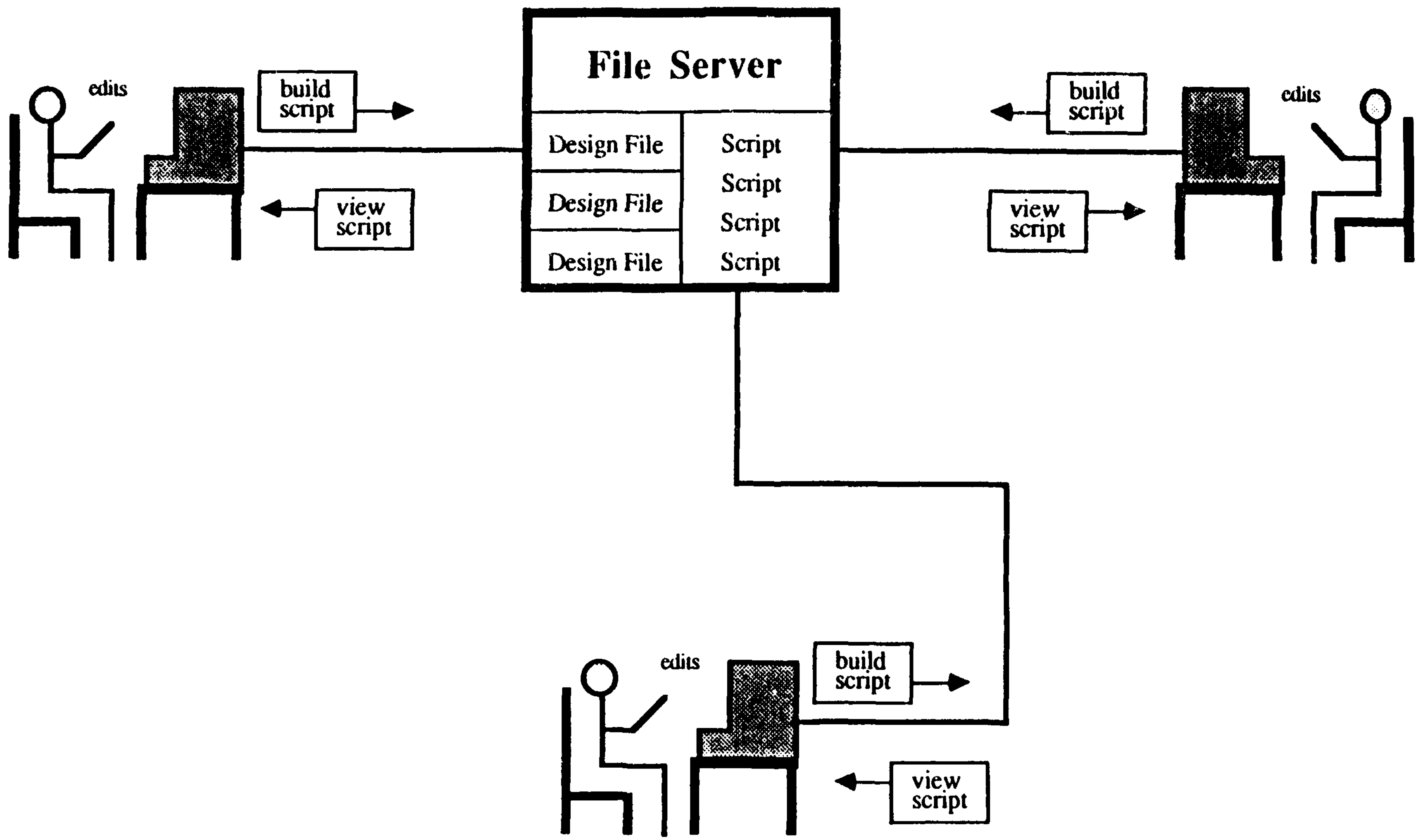
Future Work

Work is underway to implement other capabilities of the scripting utility, such as detecting conflicts and finding the difference between two versions of a design. Work is planned for integration of two design versions using scripts.

Environment

The scripting utility is implemented in Objective-C 4.0 under the UNIX operating system. It has no specific hardware requirements.

USING SCRIPT FILES TO RECORD CHANGES



DEMONSTRATION

Concurrent Editor for Objects

Dave Loffredo and Alok Mehta

Rensselaer Polytechnic Institute
Troy, New York 12180

Description

The concurrent editor is a ROSE-IC application which provides a graphical display of ROSE objects. It implements an easy to use interface for navigating through and modifying the complex data within these objects.

This utility will be demonstrated by using it to view the structure and contents of various ROSE objects, and it will illustrate how concurrent editing of persistent objects can be done.

Contribution

This utility gives trusted DICE users, who have some knowledge of application data, the means to explore and modify the data within persistent objects. They are able to do this in a cooperative fashion, allowing multiple users to concurrently edit the same persistent object.

DICE application writers can also use this as a powerful debugging aid. They are able to manipulate persistent object data in secondary storage. In addition, the editor may be linked into an application program, providing the ability to interactively view and modify the persistent objects within that application.

Technical Approach

The utility is written as several Objective-C classes. These classes may be used stand-alone or incorporated into other applications. The graphical features of the utility are implemented by a set of Objective-C classes that act as an interface to the X Toolkit.

In addition, the utility makes use of the ROSE-IC scripting features to capture and publish changes made to objects, thus making possible the concurrent editing of objects.

Status

An initial version of the utility is operational.

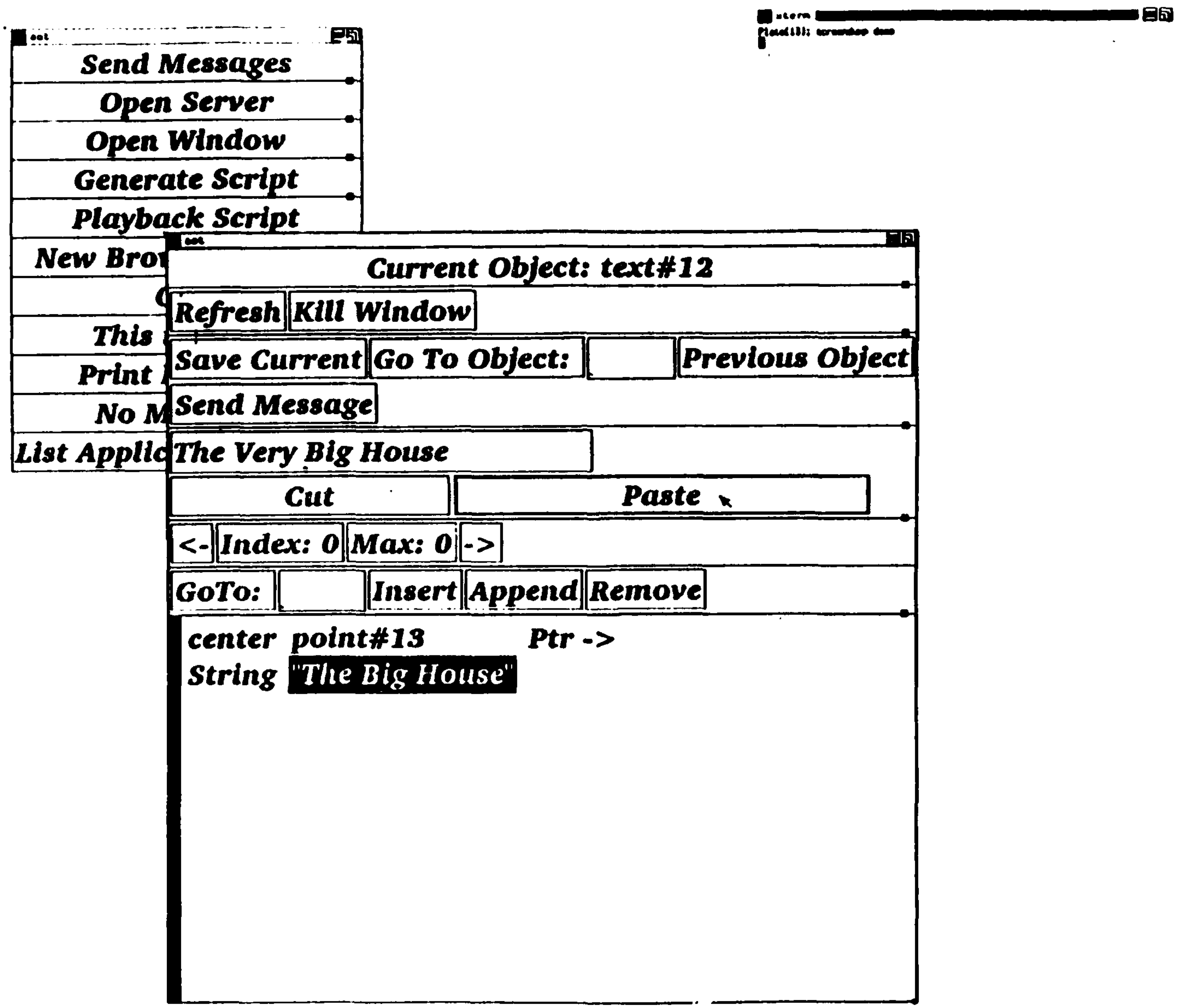
Future Work

Future work will be directed towards enhanced display capabilities, and improved data manipulation functionality.

Environment

The utility requires ROSE-IC, Objective-C 4.0, and X11R3.

CONCURRENT EDITOR for OBJECTS - Sample Session



DEMONSTRATION

SQL Interface/Relational Finder

Blair Downie, George Samaras

Rensselaer Polytechnic Institute
Troy, New York 12180

Description

The SQL interface allows application programs to use a relational database as an index for an object database.

The interface will be demonstrated as part of a VLSI design tool. The designer uses the interface to query the database to find components used in the chip design.

Contribution

For DICE users, the relational database will serve as a dictionary for objects. This will make it easier for engineers at different sites to work together.

Technical Approach

The SQL interface has been provided for the ROSE object database system. This combination provides efficient storage and retrieval for clustered objects, and fast associative searching using the relational database. The latter is accomplished by storing in the relational database only the properties of an object that are of interest to a designer. A tool has been implemented that traverses objects and creates relational tuples that describe the object.

The SQL interface is implemented as a set of classes within an object-oriented programming language such as Objective-C. These classes allow application programs to communicate with a relational database using SQL.

Status

A preliminary interface has been implemented in Objective-C using ORACLE as the relational database system.

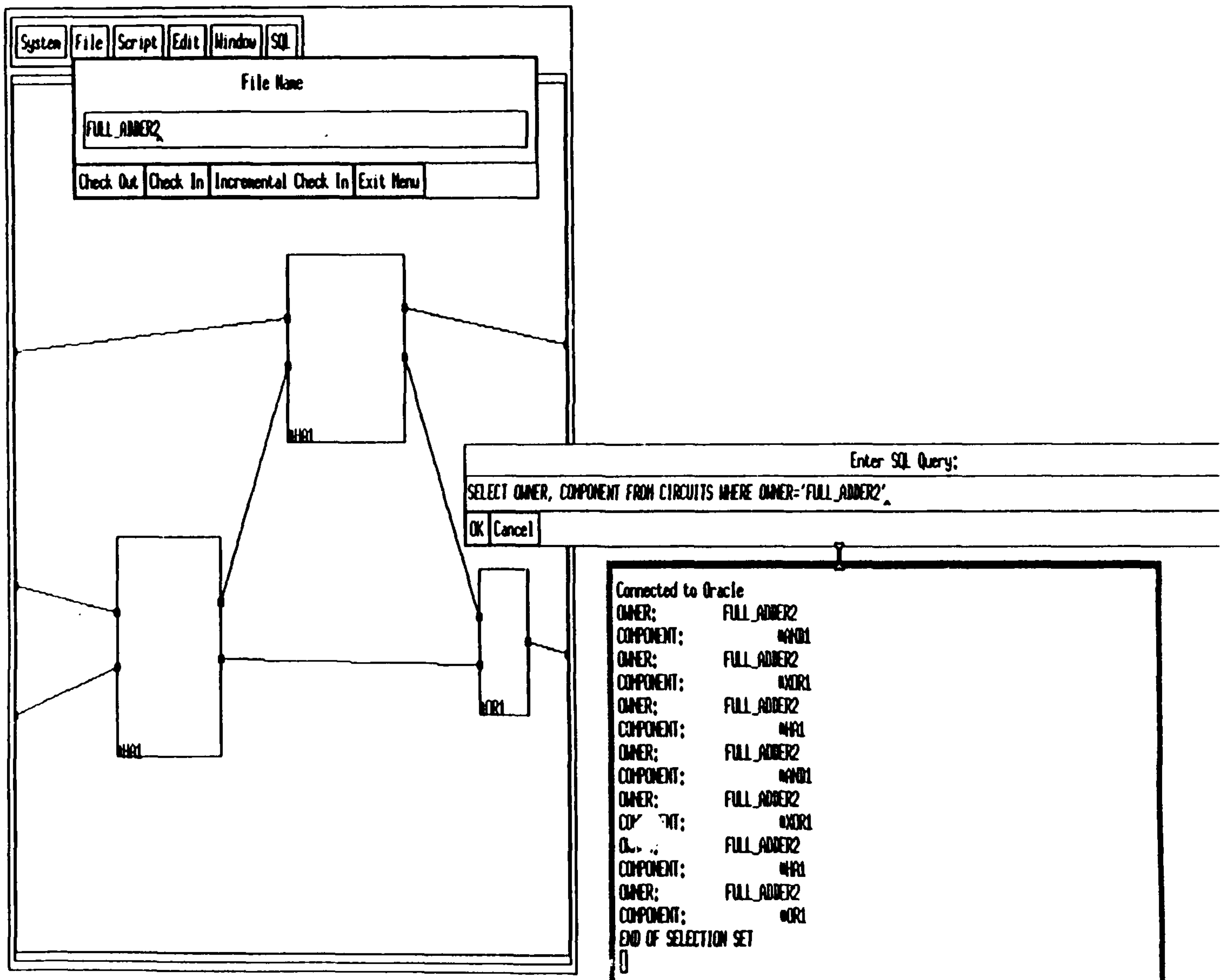
Future Work

Adapt the interface to other relational database systems.

Environment

The interface requires Objective-C 4.0 and the ORACLE relational database system.

RELATIONAL FINDER -- Sample Session



DEMONSTRATION

DATA TRANSLATION TOOL

Donald Sanderson, Daniel Jacobs, Martin Hardwick, David Spooner

Rensselaer Polytechnic Institute
Troy, New York 12180

Description

The Data Translation Tool is a tool for editing the structure of data so that it can be exchanged between application programs requiring different data formats. It is intended for use in those situations where a data exchange standard is not available. It is best viewed as a data translator generator. Using a graphical user interface, a desired data translation is specified, and the Data Translation Tool generates a program to perform this translation.

The Data Translation Tool will be demonstrated by using it to convert ROSE objects into Namelist format for potential use in FORTRAN programs and other applications requiring data in Namelist form.

Contribution

The major contribution of the Data Translation Tool is providing a mechanism for integrating heterogeneous application systems into a common data management framework. For example, it will allow existing application systems, as well as new application systems written in foreign programming languages, to be integrated into the DICE architecture. Thus, it will be useful to users of the DICE system, as well as to users outside the DICE system, wishing to integrate a foreign application into the DICE architecture.

Technical Approach

The Data Translation Tool is being implemented as part of ROSE using Objective-C and ROSE-IC. Its major component is a ROSE Data Translator for editing the structure of objects in a ROSE database. If the source data for a translation task is not a ROSE object, it is first converted to a ROSE object using an import tool. The ROSE Data Translator is then applied to convert the structure of the ROSE object into the desired form. If the target data for the translation task is not in the form of a ROSE object, the final ROSE object is converted to the required form by an export tool. Portions of the import and export tools must be application-specific. The goal is to minimize the amount of application-specific code required.

Status

The first version of the ROSE Data Translator for a pre-DICE version of the ROSE system has been completed. A new version of this tool is under development for the current ROSE system used in DICE. This version of the ROSE Data Translator is a complete redesign with significant enhancements in capabilities. It is also expected that the performance will be significantly improved. The import and export tools needed to complete the Data Translation Tool will be developed concurrently with the enhanced version of the ROSE Data Translator.

Future Work

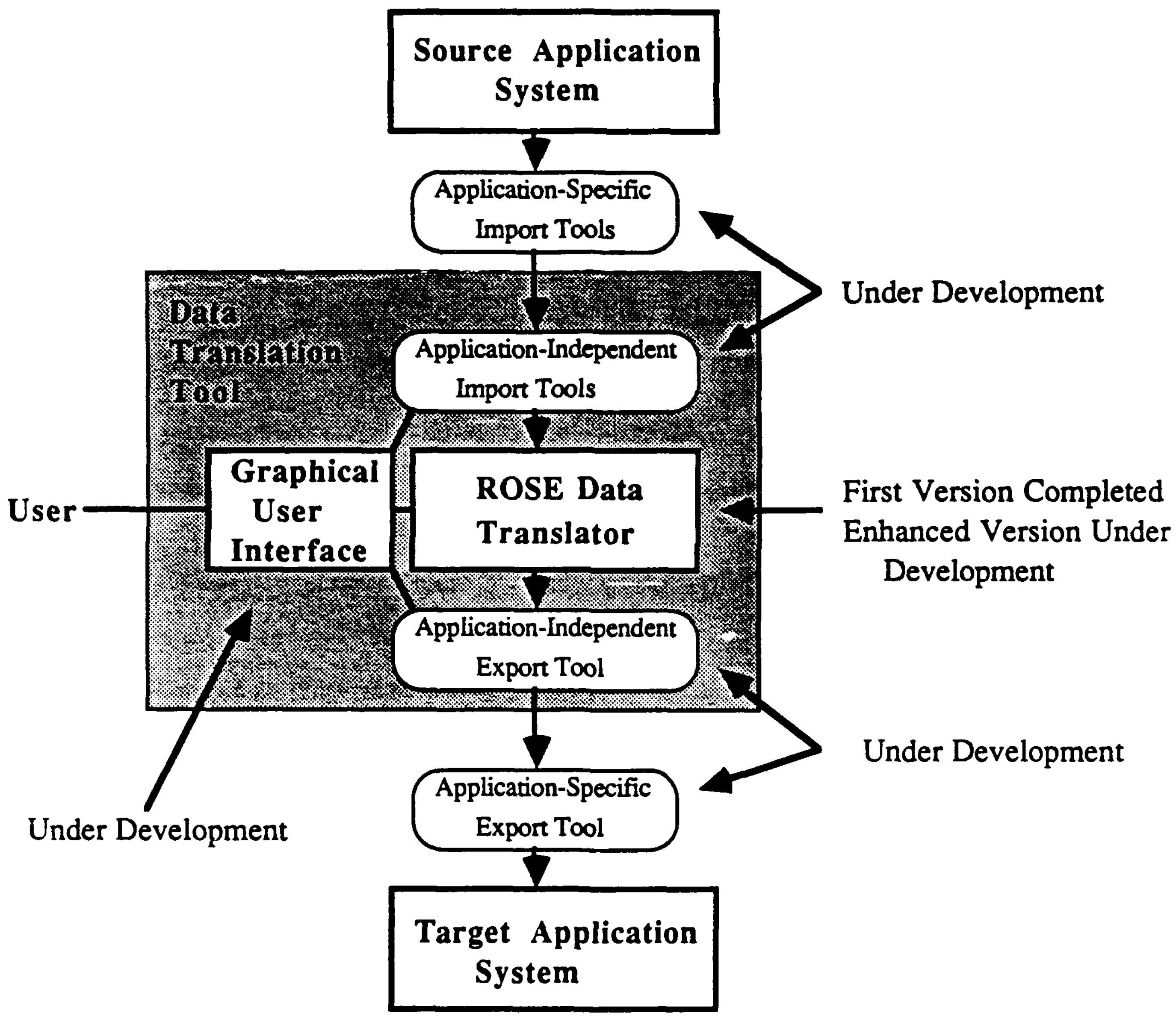
Work for the immediate future will be devoted to implementing the operators needed for the enhanced version of the ROSE Data Translator, the import and export tools, and the graphical front-end for the resulting Data Translation Tool. After this, work will focus on integrating the Data Translation Tool with the scripting facilities and other related facilities of the ROSE system to support concurrent editing of ROSE objects and cooperative work environments.

Environment

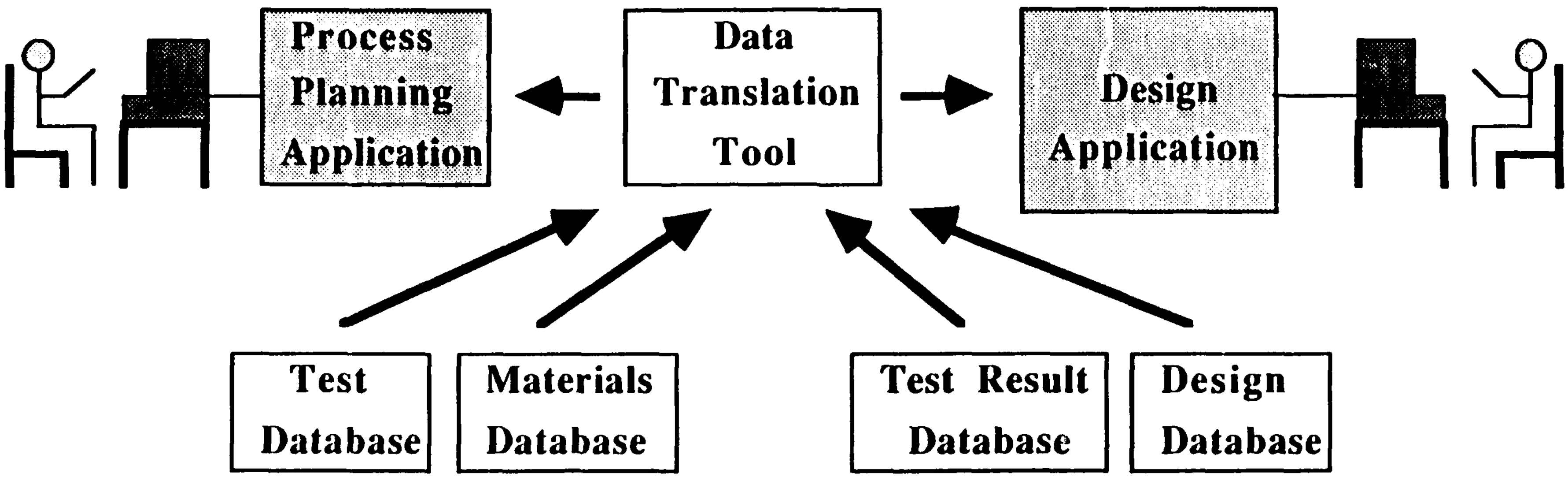
The Data Translation Tool requires Objective-C and ROSE-IC. It should run on any hardware supporting these systems.

DATA TRANSLATION TOOL

Architecture and Status



USING THE DATA TRANSLATION TOOL



(Future) DEMONSTRATION

ROSE++

J. Alyce Faulstich, Martin Hardwick

Rensselaer Polytechnic Institute
Troy, New York 12180

Description

ROSE is an object-oriented engineering database system supporting concurrent editing of objects, and providing support for cooperative engineering work environments. It has previously been implemented in the Objective-C programming language as a library of database functions which support application programs also written in Objective-C. The current task is to provide an implementation of ROSE in C++ (ROSE++) to support application programs written in that language.

ROSE++ is not yet available for demonstrations.

Contribution

ROSE++ will provide the functionality and portability of ROSE and its associated tools (such as a graphics management system, relational finder, and data translation tool) to users who prefer to use the C++ programming language. Since the Objective-C and C++ versions of ROSE will support the same database functionality, programmers in cooperative work environments will not need to use the same programming language to share a database.

Technical Approach

ROSE++ is being implemented as a library of classes within the C++ object-oriented programming language. These classes provide the same functionality as the Objective-C version (ROSE-IC) with a very similar user interface.

Status

ROSE++ is still under development. Several ROSE-IC classes have already been ported to C++. The first version of ROSE++ will be based on the version of ROSE-IC now in beta test.

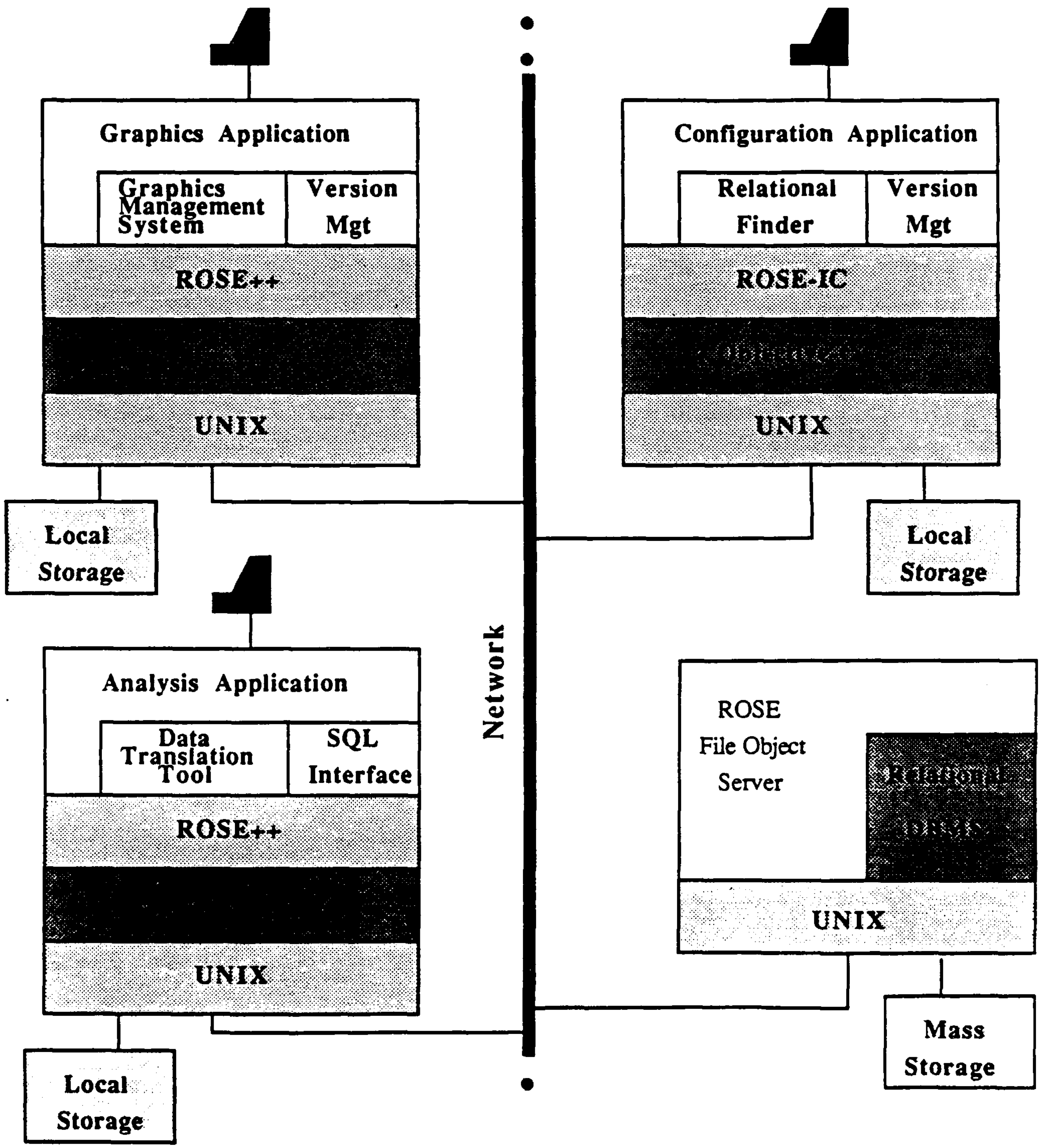
Future Work

The initial port of ROSE-IC to C++ must be completed. Then the tools associated with ROSE-IC will be ported to the C++ programming environment, as will future versions of ROSE.

Environment

ROSE++ currently requires only AT&T's Release 1.1 of C++. This will soon be moved to Release 2.0. It has no specific hardware requirements.

ROSE-IC AND ROSE++



Task 3.3.2.5 Architecture Definition   West Virginia University

A. Objectives:

The DICE Communication Channel (DCC) is seen as the connecting network that can bring together designers of different domains to forge a cooperative enterprise. The DCC is the underlying layer of communication software enabling and facilitating the cooperation and coordination involved in Concurrent Engineering.

The CONCURRENCY MANAGER, (embedded in each node as a module called the Local Concurrency Manager or LCM), eliminates the need for users or applications to manage remote process communication themselves. Instead, the LCMs taken together, *implement a complete* virtual network of connected workstations, nodes, servers and resources. These capabilities to execute applications are known collectively to the LCMs and available transparently to all users and applications.

The BLACKBOARD COOPERATION scheme enables DICE users to share all or part of their evolving designs with local or remote development team collaborators. This facility encourages designers,who may have completed their sub-tasks, to publish relevant results on a blackboard. All affected designers are, then, notified on a "need-to-know" basis thus providing an easy mechanism for catching downstream conflicts early.

B. Approach:

The DICE approach to Concurrent Engineering design visualizes several design groups working with their individual CAD Tools on connected workstations linked in a network. In such a system, there is a very large contribution to the efficiency of the concurrent design process from the computer assistance present for the three Cs: Communication, Cooperation and Coordination.

In such a system, there is a layer of software operating on every networked computer to enable the engineers to take advantage of cooperating experts and their tools on the network. This software performs the following chief functions:

Communication:

File transfer and messaging are supplemented by new utilities to enchance cooperative communication for problem solving. This software, called *Cooperate* connects a group of engineers and organizes a virtual "meeting" on the network. Yet another utility, called *View* , enables the communication of graphics between networked designers to buttress the exhange of design information during the meeting. Object-oriented communicaiton is supported by the development of methods to enable objects represented in different computers to invoke each others, behavior;

Cooperation:

Communication between processes and executing in different nodes is at the core of sharing expertise over the network. In permanent process-to-process communication, remote process communication is simlified and generalized through a base layer of systems software in each workstation, called the Concurrency Manager (CM). Any application or designer can invoke the CM in their workstation to communicate with any other application or designer in the network. Therefore, a complete protocol is defined.

The CM also facilitates the capability to have a local menu and window interface to another designer's application without needing login authority or special expertise. This is called *Designer Pctency* , sharing the capabilities of any desinger's applications with others.

Third and an equally important capability is running a job consisting of numerous programs in different computers in a defined precedence relationship. Designers, who can invoke such an execution capability, leave the CM to manage the details, and have a powerful tool to perform network task scheduling,;

Coordination:

The third component of the systems software is aimed at making the activities of the designers and the Project Leader (PL) visible to each other. A modified blackboard architecture implements the coordination between designers. The PL can assign tasks, access the engineering data base and bring down parts of it to the globally visible blackboard workspace. The blackboard maintains the current state of the design and designers can assert any proposed changes to the design on the blackboard. Conflicts are detected by the blackboard. The blackboard has a base of design rules, constraints and dependencies to pilot the design evolution, but its main purpose is to allow designers and the PL to exploit the global visibility of the blackboard as a coordination mechanism. Task management is, then, a specific function performed on behalf of the PL.

In summary, the net effect of implementing the three Cs is to increase the level of concurrent activity in the network.

C. Technical Results:

Under the communication task, the first versions of the Cooperate and View utilities were implemented and demonstrated at the July 25, 1989, Demonstration. The Network Traffic Monitor and the Diagnostic Evaluation Monitor were also implemented in a base-level version and the capabilities illustrated as a stand-alone demo.

The Concurrency Manager was also shown in a stand-alone demo on July 25, 1989, having the functionality of managing the resources of the network including users, applications, and machines.

The DICE Blackboard for Design Evolution also played a signal role in the demonstration of July 25, 1989, affording the basic mechanism of common visiblity and coordination of tasks between perspectives.

All these efforts are to be fully described in a Technical Report of some 150 pages to be submitted to GE and DARPA in September, 1989.

D. Conclusions:

This approach has been shown to be useful and further work is needed to deepen the scope of the modules and to port it to 3 other UNIX platforms in addition to the Sun 4.

The possibility of integrating this approach with other modules, in many ways, is also proven. However, the actual smooth interchange of data is yet to be achieved. It is that interchange which will be attempted at the December 1989 Demonstration.

E. Recommendations:

The tasks undertaken must be continued and supported. In their full versions (June, 1990), these tasks will form the core of an extremely useful set of DICE services. Further tasks emanating from current developments are proposed for Phase II. These are quite important to the ultimate transitioning of DICE to industry. These tasks need to be authorized and fully funded.

F. **Publications:**

Two papers from the Architecture Group were submitted to Conference workshops at IJCAI in August 1989:

-- **A Blackboard Scheme for Cooperative Problem-solving by Human Experts,**

F. Londono, K. J. Cleetus, Y. V. Reddy, and

-- **Software to Facilitate Concurrent Engineering,**

K. J. Cleetus, R. Kannan, F Londono, Y. V. Reddy

G. **Hardware:**

Software for this task included:

Purchased:	LASER on Sun 4		
	Objective C on Ultrix		
Developed:	*Cooperate*	}	
	View	}	DCC
	Network Traffic Monitor	}	
	Diagnostic Evaluation Monitor	}	
	Blackboard for Design Evaluation (DBB)		
	Concurrency Manager (CM)		

The developed software is not yet in a form releasable to beta sites

DICE Program

Task 3.3.3.1 VMS Thread Components GE-CRD

Objectives
The objectives of this task are to develop VMS software components supporting the DICE CE environment, and the Phase I demonstration.

Approach
The overall development plan is to carry out most of the generic software development in Ultrix, and to port completed components to the VMS environment.

Technical Results
Since it appears that many of the UNIX services which Ultrix provides, (and which the DICE development leverages), may soon be available under VMS, little effort was spent in re-creating those services. Resources were expended however in providing interfaces, and translators and in general integrating the VMS based manufacturing tool suite, specifically TRUCE, and NC Verify.

Conclusions
Many necessary standard services for full reliable inter-connectivity and inter-operability are not yet available under VMS.

Recommendations
Wait for UNIX-compatible supported standards.

DICE Program

Task 3.3.3.2 UNIX Thread Components GE-CRD

Objectives
The objectives of this task are to develop UNIX software components supporting the DICE CE environment, and the Phase 1 demonstration.

Approach
The overall development plan is to carry out most of the generic software development in Ultrix, and ꞷ port completed components to other UNIX environments.

Technical Results
Many Ultrix code components were developed/modified/integrated as part of the Phase 1 effort; functional descriptions appear later in this document in the those sections where the components are integrated to provide overall functionality. Specific Codes developed/modified/integrated are:

- XS spreadsheet tool
- Q-Calc spreadsheet tool [wrapped and evaluated,but not used in the Phase 1 demonstration]
- TAE+ outer wrapper for XS
- Data wrappers for
  - ADS optimization code
  - ROSE database code
  - BEAM namelist FORTRAN program
- Change Management System (not employed in the Phase 1 demonstration)
- "genSchema" automated schema generation tool
- "Executable Mockup" tool

Conclusions
The development, modification and/or integration of these modules went smoothly in a uniform, common single operating system environment, however when the software was ported to other platforms inconsistencies were discovered which were difficult and time consuming to locate and resolve. The fact that CRD did not have local access to some of the hardware platforms used at CERC

posed serious porting and integration problems not encountered until final integration at CERC.

Recommendations

CRD should facilitate a stand-alone DICE CE Testbed environment which has representative samples of the platforms supported in the demonstrations; the facility will be of great help in detecting and resolving software interactivity/interference problems before software modules are distributed to the rest of the DICE team.

DICE Program

Task 3.3.3.3 Workstation Node Prototype GE-CRD

Objectives

Develop and/or integrate the basic technical underpinnings needed by user's application programs running on a workstation platform to "connect into" the DICE information architecture framework; these basic services consist of:

- "wrappers" whose function is to encapsulate the applications that the user wishes to run, providing inter-connectivity, inter-operability and transparent access to the global object space, and
- the ROSE "client" process, whose function is to provide version-controlled access to persistent global objects.

Approach

The work associated with this task was initiated in of a series of studies to define and constrain the problem and to determine the strengths, weaknesses and overall applicability of existing software tools to achieve the desired functionality. Next, modules providing the needed functionality were acquired, and/or developed through rapid-prototyping. Finally, these modules were integrated to provide a suite of existing old applications with the desired new properties of inter-connectivity, inter-operability, shared global object space and object persistence.

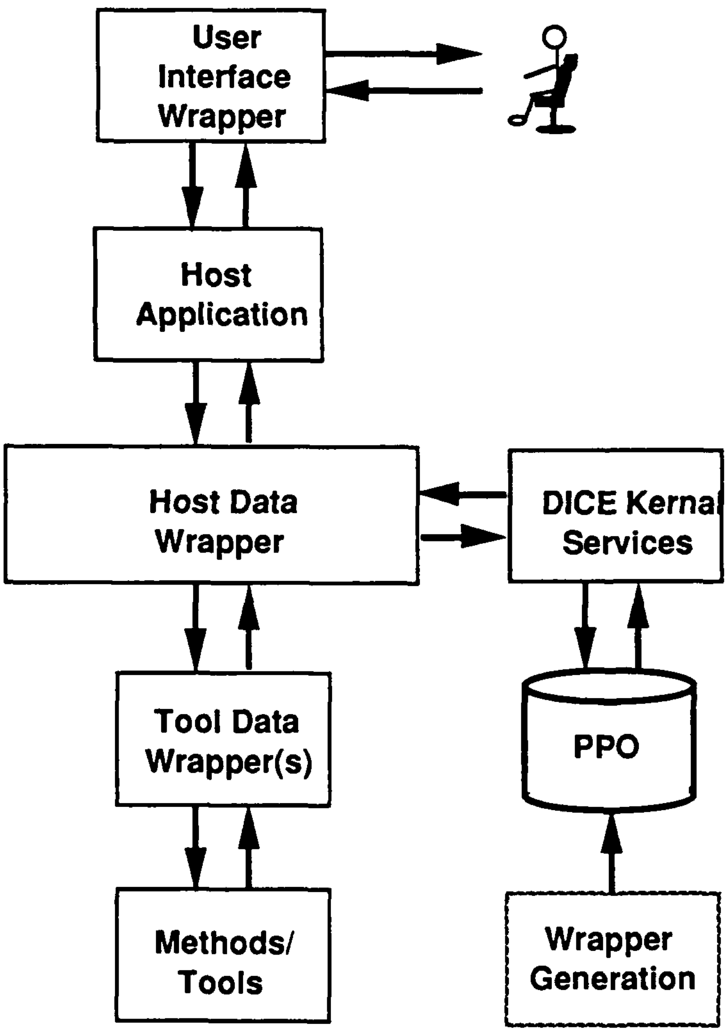
Technical Results

Study of Prior Work

Most currently "integrated" applications are *vertically* integrated where each application defines a different environment with its own private versions of each of these facilities. Standardized applications architectures such as the IBM Systems Applications Architecture (SAA), the Apple Macintosh Inter Applications Communications (IAC) facilities, and the OSF Distributed Computing Environment (DCE) are emerging, but these architectures are different from each other inform, content, and implementation. Moreover, for various reasons, product developers and their organizations have standardized on one or more host applications and are unlikely to change very quickly or very completely to others. Inter-connectivity between these environments has not been addressed.

To solve this dilemma applications must be integrated with "wrappers" which enable non-DICE application software packages to access data and data services within the DICE environment in an integrated manner. A wrapper must provide a standardized interface to be used for integrating applications into the DICE environment. At the same time it must surround an application in such a way that the interface to the application itself remains unchanged, allowing the application to operate in its normal manner. More specifically wrappers must:

- Facilitate the bi-directional transfer of relevant data (including meta-data) between the application(s) and the DICE object space.
- Facilitate the transfer of directives to and status from applications.
- Present data to the user in a form which is easy to comprehend.
- Hide the internal workings of applications from the outside world.
- Provide "seamless" access to DICE kernel services.
- Provide a mapping between the application data and the DICE object space (and the host environment in the case of a host wrapper).



Application Integration Model

The *outer* or *user interface wrapper* provides a standardized graphic user interface to existing applications and converts all user interactions into the command streams specific to the individual application(s). The *host* or *inner wrapper* is a collection of interface modules within a host application which presents local and foreign functions and data to the user as if they all were local to the host. *Tool wrappers* are the interface modules which connect existing procedures, libraries,and applications to the remote call protocols or procedure interfaces, which is what makes their services available to other applications.

The figure above illustrates the major classes of wrappers and their relationship to the user, the PPO and the DICE Kernel Services. Note that in addition to these run-time wrapper features, there must be tools to facilitate the generation, integration, and modification of wrappers.

As a result, the DICE system itself is insulated from most application specific details since they are handled by the application and its wrapper. The application is insulated from most DICE specific implementation details since they are handled by DICE and the application's wrapper.

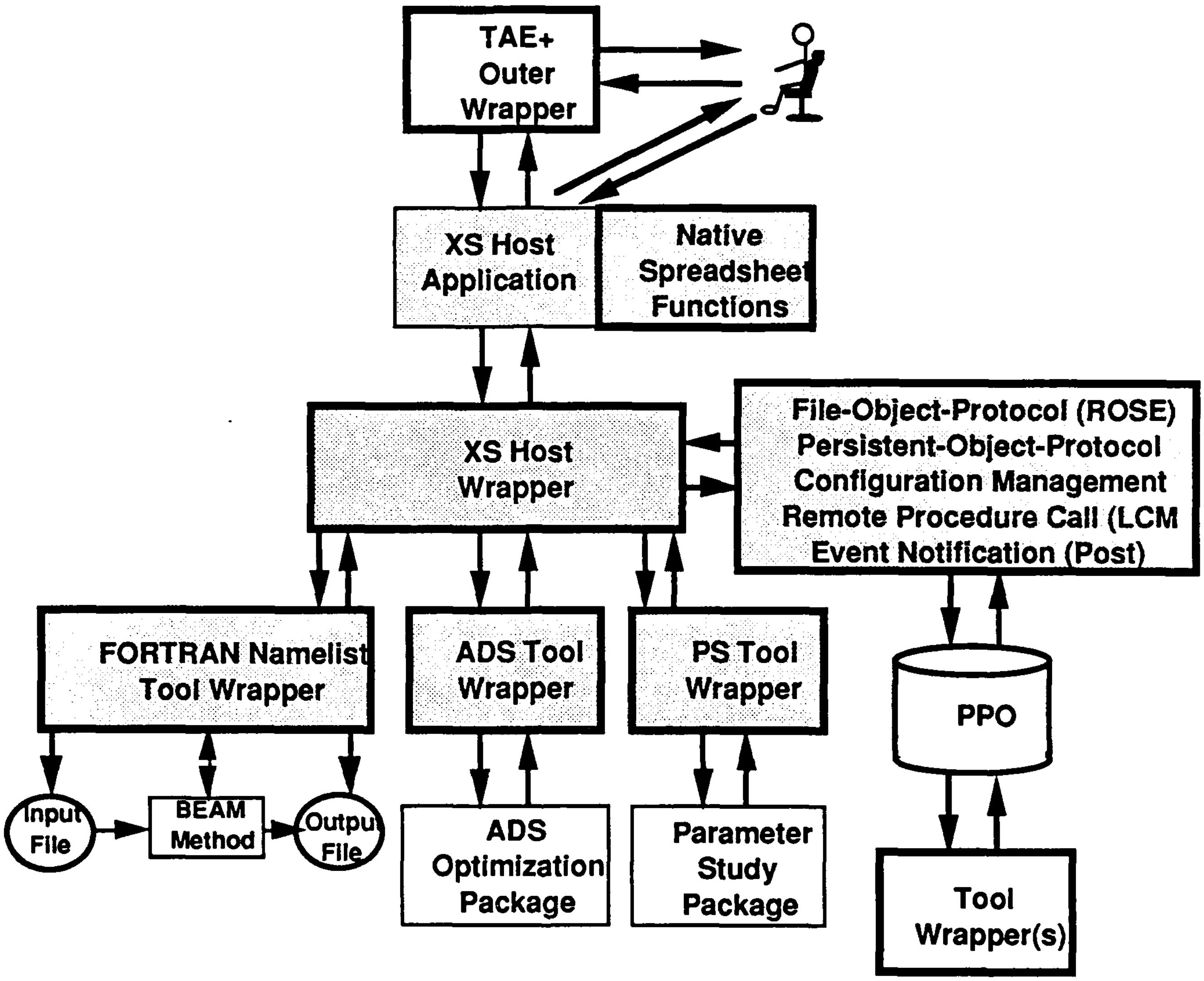
Module Development

XS Integration Environment

A prototype of the wrapper concepts described to this point have been implemented as part of the DICE Concurrent Engineering Demo. The prototype wrapper concept was initially developed for a spreadsheet integration environment, using an X-Windows based extensible Lotus 1-2-3 emulation called XS, and was (initially) targeted for Ultrix systems.

The integration environment includes a spreadsheet (XS) with access to PPO data, an optimization package (ADS), a FORTRAN namelist oriented method and DICE kernel services. The spreadsheet interface is used to reference PPO data, experiment with data, and optionally publish results. User experimentation can include conventional spreadsheet operations, custom XS functions, namelist methods, and both optimizations and parameter studies with dynamic display of intermediate results.

This integrated environment consists of several applications packages integrated and coordinated by their corresponding wrappers. A hostwrapper allows XS (a Lotus-compatible, X-windows based, spreadsheet) to be the user's view into the system. An outer wrapper allows the user to optionally access XS commands from a menu-oriented interface. Separate tool wrappers allow access to beam stiffness (namelist FORTRAN) routines, to optimization routines, and to parameter study routines that specify loop control for the selected namelist method.



XS Integration Environment

This environment provides the interfaces between user interfaces, methods and the DICE kernel services. The XS Wrapper environment has the following interfaces to the outside world:

- TAE Plus "outerwrapper" provides a menu, icon user interface which sends X events to XS spreadsheet window, supplementing XS's Lotus 1-2-3 style keystroke interface.
- Access to the PPO is through direct ROSE invocations.
- A file based persistent object protocol implements hot-links of value updates from the spreadsheet to the Project leads host environment (WINGZ and the Macintosh).
- Methods are executed locally by subprocess invocation.

- XS worksheet files may be read or written at any time. This effectively saves a users current analysis data at any given time.
- The BEAM (namelist) wrapper tool will automatically read and write the namelist files, and export the results to the XS spreadsheet.
- XS spreadsheet cells may be "hot linked" to send data to the information manager.

Conclusions

On the basis of experiences in developing and demonstrating these concepts, several conclusions were drawn.

- The spreadsheet paradigm for users is a familiar one for development engineers; they are comfortable with the interface, and the programming concepts provided. Extensions of these programming facilities to include invocation of external applications, and global persistent storage was very natural, and well accepted by the user community. In general, the "Macintosh"-style menu/icon interfaces employed were viewed somewhat suspiciously as "gimmicky" by line engineers.
- The wrapper concepts described above were relatively simple to implement; they appear to offer the potential for cost-effective generation of wrappers for the application classes encountered in the demonstration. They support adequately the overall system requirements of inter-connectivity and inter-operability.

Recommendations

In light of the development and demonstration experience and conclusions, several recommendations are made:

- Continue development of the wrapper concepts described above
- Extend the spreadsheet integration paradigm, to further leverage end-user familiarity with spreadsheets.
- Wrap additional examples of current application classes, as well as new ones; verify the cost-effectiveness of the wrapper concept by demonstrating auto-wrapping for some classes. Generalize the approach.
- Generalize the global object access methods

DICE Program

Task 3.3.3.4 Fileserver Node Prototype **GE-CRD**

Objectives

To enable product development groups to achieve a high degree of concurrency, a shared model of the Product (both form and function), Process (activities in all life cycle phases), and Organization (resources of all types) is required. Eventually these PPO models might be stored using a uniform set of engineering models in a uniform representation scheme and a common database, but that degree of standardization is certainly impractical for most organizations in the next ten years. Consequently, DICE PPO services are implemented as a heterogeneous collection of files, relational databases, languages, and object oriented databases. DICE must provide an information system which supports this heterogeneity.

Commonality is achieved in this heterogeneous environment in several ways, the most fundamental of which is through a common, system-wide schema. Ultimately, the DICE CE data model must be able to accommodate *dynamic changes* in the PPO schema which emerge from this pipeline.

Approach

The PPO schema defines an information model describing the product (form and function), process (activities of all sorts), and organization (resources of all kinds). This schema specifies both the attributes and behavior of all shared objects system-wide. The structured files support a heterogeneous set of interfaces. PPO objects must be accessed as objects in a number of languages including C++ and Objective-C, CLOS and Laser, (although in Phase 1 only objective-C was supported). In the Phase 1 prototype, the experimental ROSE object-oriented database system from RPI, together with some specialized DICE utilities, support this capability in Objective-C.

In addition to the *core model* which replaces the blueprint in defining the product in detail, the PPO should contain a collection of auxiliary *models* which:

- convey design intent downstream to production and support,
- convey constraints and tradeoffs upstream from production and support to design, and
- communicate anticipated progress and response to change among groups working concurrently.

Theoretically, a complete model of the entire manufacturing enterprise could be generated and exploited for concurrent engineering. In particular, for simple, stable products, both the core product definition and auxiliary models can be captured as parametric models. *For more complex products and organizations, the core models are too complex to be translated easily from the forms in which they are generated into integrated parametric databases.* In this case, the core models are left in their original forms (files or databases) and the principal relationships among design parameters captured through rules of thumb, linearization, data fitting, and reduced order models.

Common system-wide schema are defined using a pipeline in which raw information is first collected as an informal scrapbook, a formal structured model is then created, and finally a data definition leads directly to an implementation.

The task of generating and maintaining PPO schema breaks down into three concurrent processes. First, a scrapbook of programs, data definitions, papers, and other data is collected. Second, that scrapbook is structured using data flows, entity-relationship diagrams, and other structured analysis tools. Finally, the structured document is converted into a computer-executable specification from which schema for the various languages and databases can be generated.

Technical Results

PPO Scrapbook

The mechanism chosen for reducing the time and labor involved in data model creation was to start with existing documentation. The term "scrapbook" is used to denote the fact that all available documentation and information is collected and organized into categories, without yet imposing structure or format on the information. Until all appropriate information is collected, the appropriate structure and organization may be difficult to determine. The task of structuring is described next.

The PPO Scrapbook is intended to:

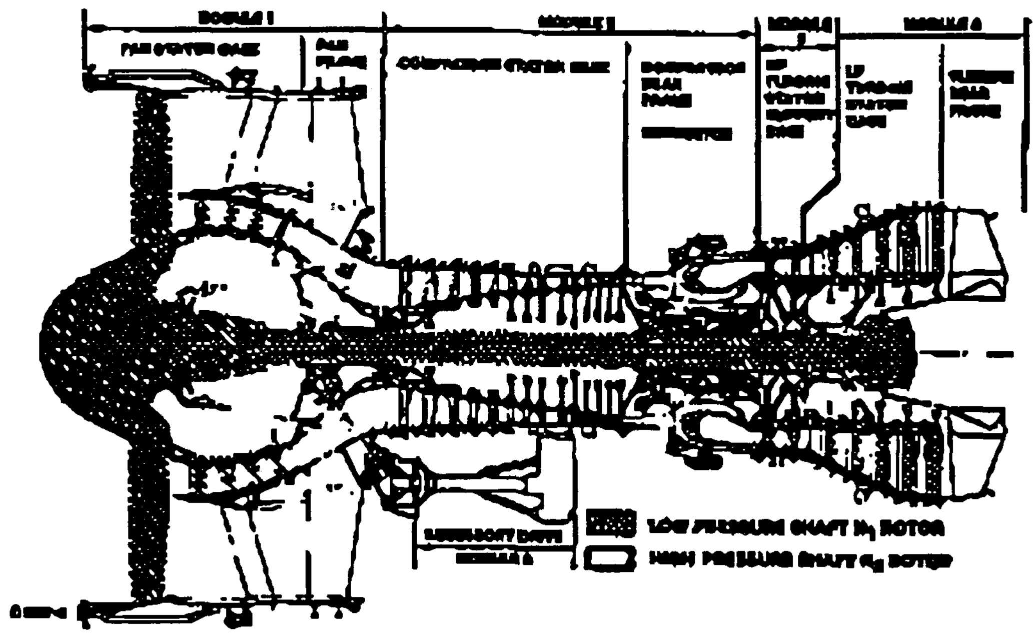
- Provide the raw material from which a detailed, structured PPO schema can be formulated,
- Determine a first candidate list of important parameters or features which play a large role in determining the characteristics of the design. Typically these parameters are shared between designers of different disciplines.
- Provide a reference for the item being designed, including pointers to the existing literature and documentation. This can form the basis of an electronic, hypertext documentation system.

For the CE structures testbed, a scrapbook was created which describes jet engines. Information was taken from various sources, including:

- textbooks on engine design,

- training documentation from GE Aircraft Engines,
- other documentation from GEAE,
- papers from West Virginia University, and
- conversations with experts at GE CRD.

The scrapbook contains diagrams and explanations of the jet engine form and function. The major sections are identified, as are the major design goals, such as low weight and high efficiency. The scrapbook attempts to capture the important concepts involved in creating the design. For example, in jet engine design, global properties, such as thrust, specific impulse, and efficiency are related to the temperature and pressure changes across the major engine sections. (see figure below) These temperature and pressure changes are related to changes in energy and momentum of the fluid (air) flow through the engine sections.



The major sections of a jet engine

These changes are broken down into smaller changes across the stages which make up a section. For example, the compressor is actually composed of a number of fan-like stages. Finally, detailed part characteristics, such as blade shape, must be designed to bring about the desired fluid flow properties.
The characteristics of jet engines are governed by a number of complex, interrelated equations and heuristics (rules-of-thumb). All relevant equations and design rules should be captured in the PPO Scrapbook.

These equations are based on many parameters whose values will ultimately be stored in the PPO. There is another dimension of complexity in that the values of these parameters depend in some cases on the operating condition of the engine. For example, efficiency may be a function of velocity, air pressure, temperature, and other factors.

In fact, different designers may be concerned with essentially different models of the same part. The mechanical engineer may be interested in a blade which is at operating temperature and under stress, while the manufacturing engineer is concerned with the production and assembly of parts under static conditions. The characteristics of a part differ greatly depending on its temperature, stress loading, and so forth. How to assimilate these different views of a single entity within the information model is an area for research.

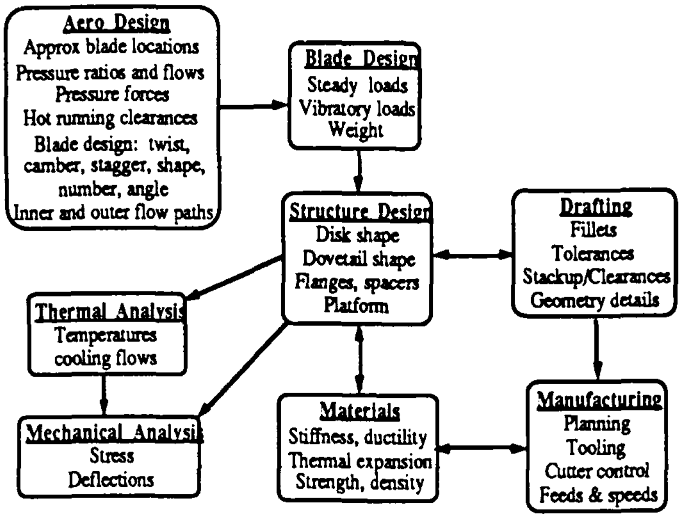
The scrapbook contains a set of pages which contain descriptions of attributes. An example attribute description page is shown in the figure below. The scrapbook may also contain photocopies of documentation which provides further explanation for an attribute. For the case where attribute values depend on dynamic operating conditions, it is appropriate to maintain tables of values.

Attribute Description

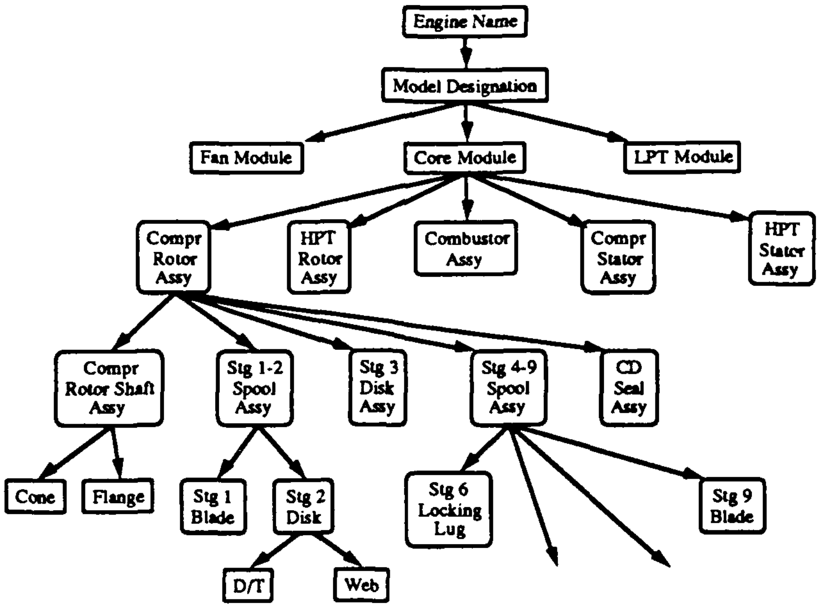
Attribute: Burst Speed.
Symbol:
Definition: The speed at which a disk will separate into pieces and fly apart.
Attribute of: Disk.
Formulas:
Tradeoffs:
Changes Dynamically? No.
Units:
Typical Values: 1.3 to 1.4 times the design speed.
Notes: The fragments would have so much energy that the loss of the aircraft is likely to result.

Sample attribute description page from the PPO Scrapbook

Also contained are such diagrammatic aids as design flow graphs and product assembly trees shown below. The scrapbook therefore contains information relating to both the product and the process of a design. This content will carry over into the PPO itself.



Informal turbine blade design data flow



Aircraft engine structure (informal and partial)

Equally important to the description of the product itself is a description of the process by which the product is designed and manufactured. This *activity model* can be used to either guide (prescribe) or monitor (describe) the design process. Activity models are appropriate for projects which focus on incremental design. Incremental design is where the general structures being designed are well understood, and only incremental changes in parameter values need to be determined. For original design, it is much more difficult to create an activity model in advance. Jet engine design generally falls in the category of incremental design, and is therefore conducive to the use of activity models.

Data Modeling and Formal Representations

From the PPO Scrapbook, a formal information model was created. This information model must be unambiguous and internally consistent. In other words, the meaning of the information model should be clear, and not subject to individual interpretation. The model should also translate easily into different possible implementations, without being limited to any particular implementation. The modularity between the data model formalism and any particular implementation will facilitate the simultaneous development of the information model and the implementation. Substantial effort was put into studying available modeling techniques and their applicability to this problem.

Product Models

Information modeling techniques have generally been derived as aids to database design. As such they tend to support the modeling of *entities*, and the *relationships* between entities. An entity is defined as being some thing which exists and is distinguishable and which has *attributes*. Relationships may also possess attributes which properly belong to the relationship itself, rather than to any of the associated entities. This terminology is general enough to be applied conceptually to any of the information modeling techniques.

The important point is that an information model should be a complete and unambiguous description which does not constrain the type of information system which will ultimately be used to implement the model (although entities, attributes, and relationships generally map quite naturally into relational databases).

Since products generally consist of tangible items, it is generally straightforward to create at least an initial model of a product. Intangible items, such as control, activities, states, and so forth are generally more difficult to model. The standards community has defined a variety of data transfer standards, and traditional (sequential) manufacturing and design work has developed models for specific products, which are in many cases proprietary.

Activity Models

There are many techniques for representing activities to be found in the modeling community, the AI community, traditional business-oriented project management techniques, the engineering design community, and so forth. These techniques may be concerned with the dynamic behavior of objects in real-time systems, or with the robust representation of project activities for the purpose of generating, scheduling, and chronicling activities.

Some modeling techniques provide specific support for representing the life-cycles and data flows of objects (i.e. Shlaer-Mellor notation), while other techniques provide very general modeling notation which allows any type of information to be modeled, but with no special constructs for modeling processes (i.e. E-R Notation).

Graphical vs. Text-Based Formalisms

Modeling techniques can be either text-based or graphical (diagrammatic). Formalisms of either type can fulfill the requirement of providing a consistent, unambiguous notation for representing information models. Graphical formalisms are generally easy to comprehend on inspection, assuming that the conventions of the formalisms are understood. For example, in Shlaer-Mellor notation, an arc with a small cross line represents an inheritance relationship. Also, arbitrary networks of relationships can easily be created in a diagram. Text-based formalisms tend to have the opposite problems and virtues. Existing word processing and outline processing software provide convenient editors for text-based notations, and are generally extremely fast to use. Text-based formalisms allow a very high density of information to fit on a page. However, arbitrary networks of relationships are difficult to represent using a text-based notation. Hierarchies can be easily represented in an outline form, but complex networks can only be represented in a rather clumsy way by using multiple hierarchies.

Modeling Tools

The remainder of this section lists the available set of data modeling tools evaluated for use in DICE, and describes briefly the tools actually used for schema modeling in Phase 1. MORE II, PDES/Express, ER Notation, Shlaer-Mellor Notation, Object Modeling Technique, and Logic-Based Modeling were all studied as potential schema creation tools.

MORE II is the tool which has been chosen for the *initial* data modeling activity in DICE. MORE II is a commercial software package available through Symantec Corporation, runs on the AppleMacintosh, and is therefore readily available and commercially supported. MORE II is a text-based tool, with the associated benefits described above.
MORE II provides a convenient means to edit hierarchies in outline form. A style sheet has been developed for DICE data modeling which provides a convenient template for creating relationship hierarchies such as inheritance trees, assembly trees, and so forth. Features which are supported include collapsing/expanding of hierarchies, and cloning of attributes. Since MORE II is a generic outline editor, any entities can be modeled. However, no special constructs or notations are provided for modeling activities. MORE II is being

used both as a simple abstract data modeling tool, and, through the generation utilities, as a data definition language.

Automated Generation from Schema

As described above, MORE II is currently used as the tool by which the formal PPO information model is represented. This model contains entities, parameters, data types, and the relationships between the various pieces of the model including inheritance relationships, as well as assembly and functional relationships. MORE II is capable of exporting an ASCII text file as input to automatic generation utilities.

Class Definitions

One such tool, known as "genSchema", was developed to generate Objective-C source code class definitions, along with data access methods. This process has been called "schema generation". Once compiled, the class definitions can be used to view, manipulate, and update associated data files. The Objective-C class methods actually create ROSE schema objects and manipulate all data internally as ROSE application objects. This has the benefit of providing the application developer with a higher-level object-oriented interface that is more tailored to the specific object classes and their attributes, while supporting object persistency through ROSE.

File Management

The DICE information system provides a common organization for a heterogeneous collection of files. The directory structure may contain files of any type. Application-specific files are those which would be used even in a non-CE environment. Structured files contain summaries of important parameters, especially those which may be of interest to multiple disciplines, together with references to tool-specific files.

Structured Files

The structured files used by the CE Testbed are those supported by the ROSE database system. The structure definitions are automatically generated from the schema specification by genSchema, as described above. ROSE supports several formats of its own, including "ROSE Format" (ASCII files in pretty-print format which are easily read and understood), "DICE Format" (ASCII files which are not generally human-readable, but which are more compact than ROSE Format), and several additional external formats, including FORTRAN *Namelist*, and *Laser* formats.

Activities

As discussed above, there are many ways of representing activity information. It is assumed that several formats will become standard parts of the PPO model for an enterprise. Several types of activity models have been used in existing DICE prototypes and demonstrations. These include:

- A network of activities which form a PERT diagram.
- A "dynamic" GANTT chart in which posted events dynamically create the chart, filling in entries at each time increment.
- A "dynamic" flow graph in which posted events cause the currently active activities to be indicated.

The "dynamic" models mentioned above run on a Macintosh and generally serve as demonstration aids for tracking the progress of the concurrent scenario. These models, and their associated tools, are not yet integrated into the Phase 1 unified PPO, but instead have their own local activity data.

Conclusions

The overall approach seems sound:

- The "information pipeline" approach for PPO instantiation seems to work well.
- MORE II is an extremely efficient tool for schema creation, and was used to quickly create a large (XXXX object) database; on the other hand, it cannot be used simply to create complex relationships .
- "genSchema" works well for the automated creation of Objective-C source code definitions and access methods for ROSE.
- Execution time response for the PPO was satisfactory.
- No insuperable problems were found with the basic PPO approach, and it seems capable of addressing basic system requirements.

Recommendations

Continue and extend the basic PPO approach:

- Capture the "information pipeline" in an integrated hypertext document so that all definitions are traceable back to the original materials.
- Since the schemas which define these structures evolve iteratively, the CE environment inherently requires a data system which supports flexible, dynamically changing schema and dynamic migration of instance structures between schema in order to support these iterative changes.
- Wrap and integrate the commercial project planner, MicroPlanner™ as a low-cost activity model editor. This will allow the status monitor tools access the same activity model as does the rest of the DICE system.
- Explore the use of integrating a cooperating graphical editing tool for defining more complex schema than those addressed by the MORE II tool.

- Extend PPO schema to include records of the events which are generated during the execution of the CE Testbed, as well as transcripts of Configuration Management transactions and assertions which are made to the DICE Blackboard.
- Prototype examples of key features of the PPO "auxiliary" models including constraints and tradeoffs.
- Support a fast "binary format" (very small and efficient files which are specific to different machine architectures) for the PPO.
- Integrate a Configuration Management System supporting the publication and retrieval of any type of file, regardless of its format.
- Support object exchange for other languages in addition to Objective-C, such as FORTRAN, C++, Laser, and CLOS.
- Leverage operating system capabilities by providing automated directory structure generation directly from object models.

DICE Program

Task 3.3.3.5 Information Manager Node **GE-CRD**

Objectives

The purpose of the DICE Information Manager (IM) is to give an overview of what is happening within the DICE environment, and to provide tools to help determine what could be and should be done within the environment. The IM therefore provides activity management tools for project planning and tracking, as well as tools for running "what-if" scenarios, which help determine the best paths to pursue during design evolution. In short, the Information Manager is a tool which provides a "road map" into the concurrent project. The information provided by the IM is available to all project participants. The tools for project planning may be available only to the project manager.

The IM also must provide tools for browsing PPO schema and data, (not implemented in Phase 1), and management tools for generating PPO schema (implemented in Phase 1).

The IM activity management tools must track the various versions which are under development within the concurrent engineering environment. It would be difficult to keep track of which people or groups are performing which tasks, using which versions, without project tracking tools to facilitate this. The project plans and status which are generated and tracked by the activity manager will be stored in the PPO, but this has not been implemented in phase 1.

Tools to run what-if tests against analytical models provide an invaluable tool for choosing the correct design paths which should be pursued. Analytical models provide a first-order approximation of the characteristics that a given design will have. Running analytical models against a range of possible parameter values suggests the parameter values which hold the most promise. By integrating the what-if tools with the PPO, first-order evaluations can be determined against the archived library of existing designs.

Finally, the IM should demonstrate an *electronic "design notebook"*; the EDN has the following goals:

- Support hypermedia documentation
- Capture design intent
- Support cut and paste from applications
- Preserve linkages back to the PPO for pasted objects

Approach

A rapid prototype of the Information Manager was implemented on a Macintosh SE:

- WingZ was actually used as an application prototyping environment.
- Communication links to UNIX workstations were established by means of shared directories between the Macintosh and UNIX. This is possible through TOPS (for Sun workstations) or through AppleShare (for Ultrix workstations) .
- Prototype EDN as a WingZ application.
- Spreadsheets provided a convenient mechanism for organizing data, allowing functions and dependencies between data to be easily implemented, and for providing a user interface over tables of data. WingZ was chosen because it provides this spreadsheet capability, along with graphics, Hypercard-like links, and programmable event handlers, all of which can be controlled and programmed using the Hyperscript programming language.

Technical Results

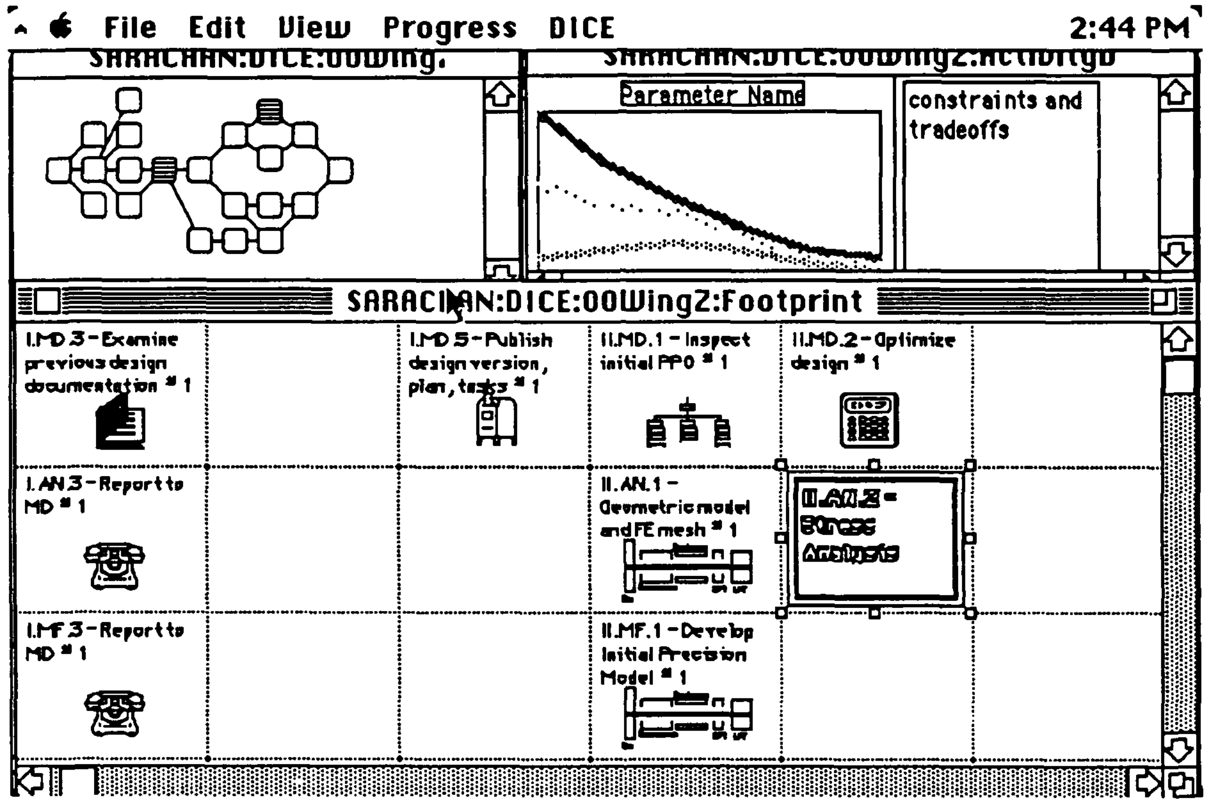
Views

There are several ways in which activity diagrams can be represented. The IM prototype implemented several different views. Possible views include the following:

- A flow graph which represents the data dependencies between the tasks of a project.
- A PERT chart which represents the plan for a project in the form of a graph, but where adjacent nodes are not necessarily related by data dependency.
- A GANTT chart which represents a plan as a set of parallel time lines.

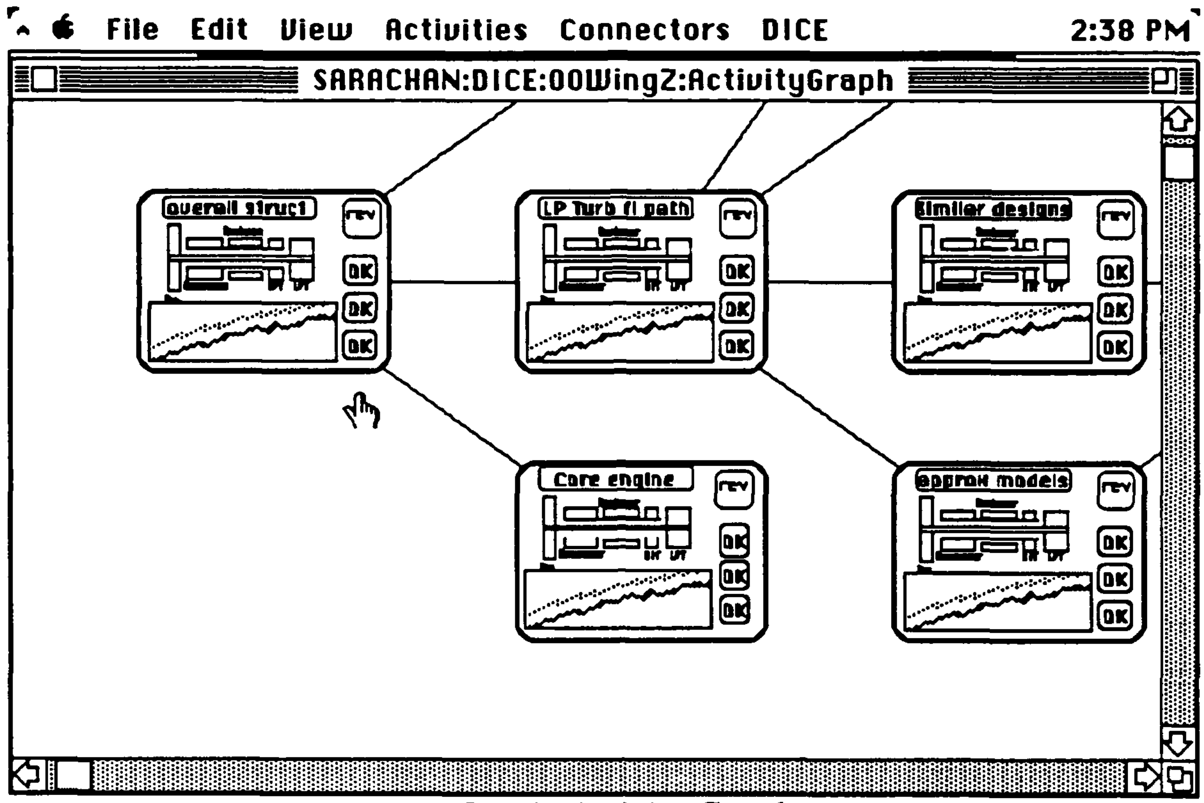
For the demo, several different activity views were implemented, corresponding to the various ways in which one may consider activities within a concurrent engineering project. The main display consisted of a "growing footprint" of project status. This is like a GANTT chart which is created dynamically to reflect actual project status rather than being created in advance to represent a plan. This view differs from a flow chart in that iterations are represented explicitly as new tasks. In contrast, a flow chart represents one complete iteration through a project. Iterations in a flow chart are represented by multiple visits to the same nodes.

In the footprint view, three parallel tracks represent the project leader / mechanical designer, analysis engineer, and manufacturing engineer. Each cell indicates the activity (through text and an icon) and the iteration number. A heavy border indicates a task which is currently in progress. An outline font indicates an activity in which an alert has occurred.



Concurrent Scenario

The IM also contains a small iconic view of the flow graph of the project. Shaded nodes represent those that are currently in progress. Flashing nodes indicate activities which have called for alerts. In contrast to the footprint view, the flow graph allows nodes to be visited more than once in consecutive iterations, so that one may judge project status in terms of distance from the end result, or goal state. The dynamic display is intended to be iconic. For more detail about the nodes, there is another, more detailed flow graph view.



Iconic Activity Graph

In the more detailed version of the flow graph, nodes are labeled, and have icons and charts. (These charts take their data from tables within the WingZ spreadsheets.) This view actually provides a flow graph editor in which nodes and arcs can be created and deleted, nodes can be moved, and so forth. The iconic flow graph shown previously is an iconic representation of the flow graph created using this graph editor.

Object-Oriented Environment

In order to facilitate the development of the IM in WingZ, a WingZ Object Manager was implemented. The purpose was to provide a programming environment in which the WingZ Hyperscript code could be well-structured and readable.

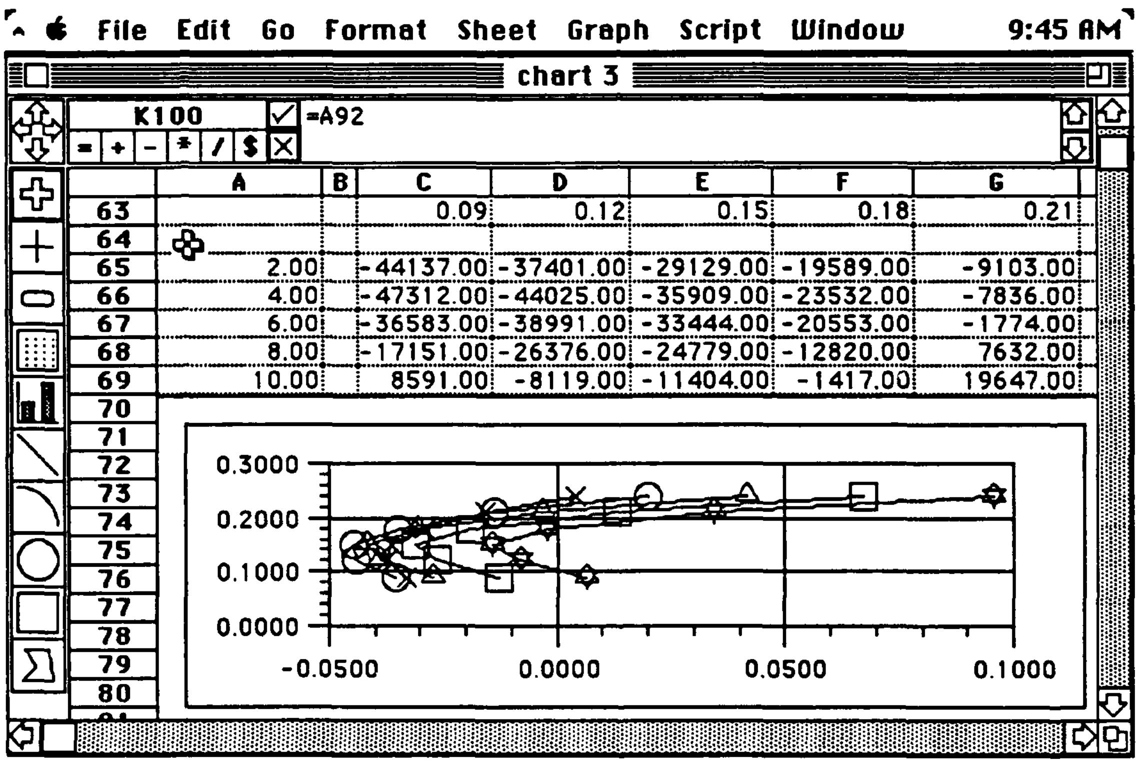
Stored Models

The data which defines the activity models (both the footprint and the flow graph) needs to be stored. The WingZ spreadsheets provide a convenient mechanism for the storage of data in the Phase Implementation, (although, ultimately, activity information should be stored in the PPO). The activity flow graph editor supports the creation and modification of activity (node) and connection (arc) objects.

The footprint view has its own storage schema, and again uses a spreadsheet table rather than the object system. The IM prototype is configured for a given demo scenario by creating the appropriate flow graph objects (which can be done using the graphical editor which has been implemented), and by loading the footprint table with the appropriate information. Ultimately, this information should all reside within the PPO. There must be tools for easily specifying plan and replan information.

Trade-off Spreadsheet using Parametrized Models

Besides activity management, the IM prototype showed several other tools which are useful in determining future design paths. One such tool is a set of curves which show how the sensitivity of stress and vibration characteristics change with the geometric properties of a turbine blade.



	A	B	C	D	E	F	G
63			0.09	0.12	0.15	0.18	0.21
64							
65	2.00		-44137.00	-37401.00	-29129.00	-19589.00	-9103.00
66	4.00		-47312.00	-44025.00	-35909.00	-23532.00	-7836.00
67	6.00		-36583.00	-38991.00	-33444.00	-20553.00	-1774.00
68	8.00		-17151.00	-26376.00	-24779.00	-12820.00	7632.00
69	10.00		8591.00	-8119.00	-11404.00	-1417.00	19647.00

Trade-off Spreadsheet

"What-If" / "Hot" Spreadsheet

Another tool used the data from the trade-off curves shown above. This is a "what-if" spreadsheet which calculates values of stress and vibration sensitivities given a table of geometric dimensions.

File Edit Go Format Sheet Graph Script Window 9:46 AM

WhatIfSpreadsheet

Hot spreadsheet: Approximate blade model

Chord	Thickness	Length	Δ Stress / Aspect	Δ Stress / Thick	Δ Freq
1.00	0.09	2.00	0.14807	-0.03310	-
1.10	0.13	3.80	0.24965	-0.04163	-
1.20	0.17	5.60	0.33800	-0.04376	-
1.30	0.21	7.40	0.41682	-0.02950	-
1.40	0.25	9.20	0.49519	-0.01639	-
1.50	0.29	11.00	0.56074	0.00275	-
1.60	0.32	12.80	0.60955	0.00425	-
1.70	0.36	14.60	0.66227	0.02542	-
1.80	0.40	3.00	0.05000	0.05000	
1.90	0.44	18.20	0.75560	0.07251	-
2.00	0.48	20.00	1.00000	1.00000	

"What-If" / "Hot" Spreadsheet

This spreadsheet is also "hot" in that the input table can be modified based on geometric parameters returned by a wrapped method code to the XS spreadsheet, which runs on an Ultrix workstation. Updates to the XS spreadsheet should therefore vary the input parameters of interest in the IM "what-if" spreadsheet. These input parameters in turn map through the trade-off curves to arrive at values for stress and vibrational sensitivity.

Inspect PPO Schema

During the demo it was possible to open a text window containing a copy of the MORE II PPO schema file which has been created as part of the DICE PPO kit. Ultimately, the IM may provide tools to create/modify schema, and to browse PPO objects.

EDN

Many hypertext "notebook" projects have been reported on in the literature. What principally distinguishes the DICE EDN from prior systems is its integration with user's applications and with the shared PPO.

Overview

A study of the requirements for the EDN revealed The paper-based paradigms for this triplet of EDNs are *the pad of paper, the notebook*, and the *patent/project book.* These paradigms differ not so much in functionality, but in controls policy: the amount of control given to the user who is creating the entry into the EDN, access control for readers, distribution, and so on.

The Pad of Paper

The *Pad of Paper* paradigm implies that users can use the EDN exactly like a pad of paper. They must be able to record anything from anywhere, including any application or database, and must dynamically re-arrange the links to fit the users' present view of the data. In essence, at this level, the EDN must be extremely dynamic and mutable/malleable. The user must be able to dynamically create entries, link them in any manner, save or throw them away and read/write data from many different applications and databases. Everything must be considered under the control of the user whether it is versioned, temporarily saved, saved permanently, or thrown away. At this level the dynamic capability is more important than the presentation because only in rare instances will this be seen by anyone other than the user who created the entries. There must be a strong link to the applications that created the data and the system must be able to rerun the application with either the same data or new data. Under the discretion of the user, the old results must be capable of being either temporarily saved or over written by the new results.

The Lab Notebook

The second level/paradigm is that of the Lab Notebook in which personal ideas and results are stored in more permanent form. The level of control is greater here. The user can add information, add notations to information and make references/link to other parts of the notebook and to other databases. The user is not allowed to erase anything and everything is versioned. Other readers must get the owner's permission. The data is kept and versioned in the PPO under the creating user's ownership and control. If the user desires, the system can rerun the application with either the same data or new data. Unlike, the Pad of Paper, the old results are not overwritten at the discretion of the user but rather a new version and new entry in the Lab Notebook is created. The presentation aspects are emphasized over the "Pad of Paper", because eventually some of the results will be seen by others.

The Patent/Project Book

The paradigm/level is called the Patent/Project Notebook. The user places information into the Patent/Project Notebook with permission from the Project Leader. This information is now a permanent record which other individuals can view with appropriate permission, and can not be changed by anyone. Other entries may reference it and indicate newer results and because of the bidirectional links, they are referenced by the older entry. Basically, from this point on, no user can alter or delete information/data in this EDN. In essence, the user has given up control of this particular entry of information. This control factor is similar to those employed for Patent Books or Legal Documents. At this level/paradigm, there is still a link to the application that created the information but it is not as strong as prior instances. The application and the application environment is indicated but the application can only be rerun in a read-only mode from this point and the information can not be over written. The new result can not be added without the Project Leader's approval. The normal pattern would be to copy the information to one of the other two layers and run the application from that level. The information resides in the PPO under a different owner other than the individual who created. In the prior levels/paradigms, the user is indicated by the owner. This is not the case at this level/paradigm and a link is created to the original user generator. The presentation of the data assumes major importance rather than the dynamic links that were so important in the other levels/paradigms.

Implementation

In the Phase 1 effort, the spreadsheet package WingZ was used to implement a crude "Pad of Paper" level EDN on a Macintosh. With the WingZ programming language, we were able to read files and launch other applications. The data from these applications were then brought into the spreadsheet with filters and saved in local files. The data was easily displayed by WingZ. A local database was maintained to aid in maintaining the links and link reconstruction when links were added or deleted. The WingZ approach was also used to prototype the "Lab Notebook" but was found lacking principally in its presentation and presentation control capabilities. In summary, the WingZ implementation provided some of the needed functionality, but did not appear to be an ideal vehicle.

Integration

The communication between the Information Manager running on the Macintosh using WingZ is done using TOPS/AppleShare to create a shared directory between the Macintosh and the Ultrix file server. Since WingZ supports user-supplied "on-idle" event handlers, it was possible to create a WingZ Hyperscript function which looks for files in the shared directory whenever the Information Manager is not otherwise engaged.

In the demo, a UNIX script containing the status update messages for the demo scenario was stepped through from a remote terminal in synchronization with the demo. Ultimately, of course, the concurrent designers and the wrapped applications should themselves send the status updates. The inverse communication path has also been implemented. WingZ can write a text file to the shared directory. A different Ultrix server then executes this text file as a UNIX script.

Demo Scenario
In advance of the Phase 1 demo, the IM prototype was initialized on the Macintosh. (This also initialized the EDN which is integrated with the IM). Throughout the duration of the demo, the script is stepped through in synchronization with the actual flow of the demo.

The result is that the "footprint" view grows; nodes are added as new tasks are initiated. Newly added tasks, which are still in progress, are highlighted with a heavy border. When a task completes, the border returns to normal. Any activity which calls an alert (i.e. a stress or vibration problem), causes the font style within the task description to change to outline. Clicking on an alerted node causes the detailed description for that node to appear. In addition, the appropriate activity nodes in the iconic flow graph highlight when active. The iterative nature of the scenario can be observed by noting that control sometimes returns to earlier nodes in the flow graph. An activity which calls an alert causes the corresponding node in the flow graph to blink. In addition to the activity management tools, appropriate tools including the trade-off curves; the "hot", what-if spreadsheet; and the PPO schema browser are displayed at the appropriate times in the demo scenario. The schema browser is simply a text editor containing the MORE II schema file. The trade-off curves were prepared in advance on a fixed spreadsheet. The what-if spreadsheet uses interpolation techniques into the trade-off curves, and receives input values dynamically from the XS spreadsheet.

Conclusions
The Information Manager Prototype illustrated many of the capabilities that DICE will provide later in the program, and was well received by users. The implementation did reveal weaknesses however, where changes are appropriate. For example, given the current tools, customizing the WingZ IM prototype for a given demo scenario is a rather tedious task. Performance of the Phase 1 prototype was not yet at acceptable levels.

Recommendations
The basic ideas are sound and should be pursued through more detailed prototypes. Several specifics are:

- The Information Manager should be linked more directly to the PPO in ensuing versions. Private databases stored in spreadsheet memory in the current version should be moved to the PPO and versioned controlled.
- Successive prototypes of the Information Manager's Activity Manager should be capable of taking a more active role in the DICE environment:

issuing task assignments, autonomous execution of well-behaved, deterministic application processes, etc.

- The issue of whether graphical descriptions of the objects being edited are stored with the objects themselves or in a separate, graphical, database must be studied and resolved
- Alternate implementation vehicles for the EDN other than WingZ should be evaluated.
- In order to promote tighter application linkages for the EDN, future versions should better leverage the DICE Wrapper effort.
- more robust communication mechanisms are needed between communicating modules.

DICE Program

Task 3.3.3.6 Integrated Operating System Thread GE-CRD

Objectives

The major objectives of this task were to pull together the separate components and "nodes" of the DICE CE architecture into an integrated system, prior to moving it to Morgantown WV for final integration with the CERC developed architecture modules prior to the Phase 1 demonstration.

Approach

Since the internal development environment at CRD is Ultrix-based, the bulk of early development was carried out in that environment. Subsequently, CRD developed tools were ported to the actual target platforms used in the demonstrations.

Technical Results

Integration of a functional subset of the DICE CE architecture was achieved in the CRD Ultrix environment. Other components, principally the Information Manager Node and the Manufacturing Workstation, were integrated later. The Manufacturing Workstation running under VMS was also coupled, but more loosely in the Phase 1 demonstration than will be the case in subsequent demonstrations. Since the DICE Information Manager node runs on a Macintosh, that integration was carried out in the Macintosh environment, and was also more loosely coupled.

Conclusions

The integration went smoothly in a uniform, common single operating system environment, however when the software was ported to other platforms inconsistencies were discovered which were difficult and time consuming to locate and resolve.

The fact that CRD did not have local access to some of the hardware platforms used at CERC posed serious porting and integration problems not encountered until final integration at CERC.

Recommendations

Put in place at CRD and at CERC similar hardware and software, so that software porting problems can be identified and corrected earlier in the development cycle.

DICE PROGRAM

Task	3.3.3.7	PaLS	NCSU
Task	3.3.5.3	Integration of PaLS	NCSU

1 Objectives

Traditional methodologies of product development are based on a sequential flow from specifications to a detailed design which is then manufactured, tested, and delivered to the customer. In reality this almost never happens. For example, a product specification may progress to the production phase before it is determined that it is not manufacturable, or that manufacturing costs for the product as designed are prohibitive. In the worst case, the project goes all the way back to the R&D phase. Such long feedback paths result in long product development times.

An alternative approach to the traditional methods described above is to take a more fine grained view of the operations and interactions required to go from product concept to delivery. The process may be represented as a directed graph, where the nodes of the graph represent primitive operations such as *evaluate the thermal properties of material Z*, or *design widget A*, or *inspect assembly B*. The edges coming into a node then represent the information required to perform the operation, for example the *functional specification for assembly B*, and so on. This approach tends to elucidate the explicit dependencies among all of the operations required in the product development cycle. The problem then becomes one of mapping the nodes of the graph onto the available resources (people and machines) and scheduling the operations assigned to each resource so as to achieve maximum concurrency. By so doing, the product development cycle will be minimized. This approach to product development is one aspect of the current DARPA Initiative in Concurrent Engineering (**DICE**).

When maximizing the concurrency of the design process within given cost constraints, the following questions must be answered (among others):

- What types and amounts of resources will be needed to complete the design on time?
- How will the workers be organized to minimize communication delays and errors?
- to whom will the tasks be assigned?
- In what order will the tasks be performed?

The **DICE** scheduling problem is difficult to solve due to its combinatorial nature - i.e., the quality of a solution is affected by a large number (possibly millions) of interacting decisions. Expressed mathematically, we must find a vector $\mathbf{s} = \{s_1, s_2, \cdots, s_N\}$ which minimizes some function of interest, $H(\mathbf{s})$, that depends on $\mathbf{s}$ in some complex, non-linear way. Compounding the difficulty, the problem is *discrete* in that the decisions may assume only one of a limited number of values (e.g., $s_i \in \{0,1\}, \forall i$). Typically, the decisions are made based upon a combination of prior experience and guesswork. Unfortunately, experience can bias a schedule away from an original and advantageous configuration, and a single poor guess can distort an entire system due to interactions between decisions.

2 Approach

At North Carolina State University, we are examining the use of simulated annealing and neural networks for solving optimization problems without bias. Simulated annealing is a gradient descent technique incorporating a random process which allows the escape from local minima such that the globally optimum solution can be found. Neural networks contain many simple processing elements which are highly interconnected such that they can rapidly solve large problems in a cooperative manner. We have incorporated the randomness of simulated annealing into neural networks to create the Mean Field Annealing (**MFA**) algorithm. The controlled randomness improves the solutions found by the neural network, while the cooperative and continuous nature of the network increases the speed and parallelism of the annealing process. Thus, **MFA**can rapidly find near-optimal solutions to a wide variety of problems.

We have modified and repackaged the **PaLS** optimizer for task planning in the **DICE** environment. This new tool is called **MFTP** (Mean Field Task Planner). **MFTP** is a tool to be used by the **DICE** Project Lead (PL) to plan and evaluate the distribution and scheduling of task assignments over the available organizational resources. The **MFTP**uses **MFA** to manipulate the underlying decision variables

$$s_{ijk} = \begin{cases} 1 & \text{if task } i \text{ is executed on resource } j \text{ at time } k \\ 0 & \text{otherwise} \end{cases}$$

so as to arrive at a near-optimal schedule. The variables are updated according to a normalized Boltzmann distribution as follows:

$$s_{ijk} = \frac{\exp(-\Phi_{ijk}/T)}{\sum_{l,m} \exp(-\Phi_{ilm}/T)}$$

where Φ_{ijk} is merely the cost incurred by executing task i on resource j at time k (i.e. $s_{ijk} = 1$). Lowering the control parameter T (often called the *temperature*) forces the task to be converge to the resource and time slot having the lowest cost. This cost is of course dependent upon many of the other task assignments, thus the tasks cooperate and compete for desirable resources and time slots and eventually reach a near-optimal global solution.

3 Technical Results

The first task to be accomplished was the determination of the cost function, $H(s)$. This included determining not only the cost criteria, but also formalizing the constraints on the problem solutions. For example, a task is constrained from being executed before all of its predecessors are complete. Such constraints are handled by placing them into the cost function using Lagrange multipliers.

Using the cost function and a simple concurrent engineering problem, a demonstration of the **MFTP** was constructed. The problem was represented as a directed graph whose nodes are the set tasks which must be performed, and a set of resources which are available to perform those tasks. The arcs in the graph represent the dependencies between the tasks. A cost is associated with each task executed on a particular resource at a particular time. The **MFTP** then attempts to find an optimal partitioning and scheduling of the task graph onto the available resources, such that the cost of completing the entire set of tasks is minimized. The output is a pert chart which shows the schedule of tasks for each available resource.

The initial version of **MFTP** did not take into account many of the realistic constraints that will be encountered in a concurrent engineering problem. These included characterizing the properties of the engineering and/or manufacturing resources and adding more constraints and time-dependent parameters. These features were added to the second version of the **MFTP** and it was used to develop schedules for a small concurrent engineering problem involving the scheduling of tasks to designers at geographically separated work sites. The man-hour cost of each designer is different and varies with time as does the cost of shipping material or data between the design centers. The objective is to assign tasks to designers at specific times so as to minimize the combined man-hour and shipping costs while still completing the design before a given deadline. The **MFTP** solution technique was also combined with an X10-based spreadsheet (developed at **NCSU**) which made it easy to change the scheduling parameters and test the performance of **MFA**over a set of trial problems.

In solving the small demonstration problem, it was found that the convergence time for the **MFTP** is polynomial in the product of the size of the task graph, the number of available organizational resources, and the length of the scheduling time horizon. This was a significant problem, particularly when attempting to scale up to real problems of interest in the context of **DICE**. Therefore, a new version of **MFTP** was created which progressively halved the scheduling horizon into two intervals and then decided in which half each task was to be executed. Thus, for a horizon of T time units the new algorithm required $\log_2(T)$ iterations to achieve the same precision as the original version of **MFTP**. However, the execution time of each iteration was reduced to a polynomial of only the size of the task graph and the number of available organizational resources. As an additional benefit, the smaller number of decision variables needed by the successive refinement technique made it practical to share actual cost information between the decisions and eliminate the arbitrary Lagrangean multipliers used to incorporate hard constraints into the cost function. These improvements allowed **MFTP** to be used for interactively solving more realistic problems during the DARPA review of the **DICE**effort in July.

4 Conclusions

From our experience with using the **MFTP** on the **DICE** scheduling problem, we have found

1. **MFA** can effectively and rapidly optimize a wide range of forms of cost functions;

2. the scheduling cost function can be expressed as a function of simple binary decisions;

3. constraints on the problem solutions can be expressed in the cost function using Lagrangean multipliers;

4. the large number of binary decisions slows the **MFTP** algorithm;

5. it is difficult to set the Lagrangean multipliers in order to insure that the hard constraints are satisfied;

6. using successive refinement on the time dimension of the scheduling problem greatly reduces the number of decision variables;

7. successive refinement simplifies the problem and allows actual cost information to be shared between the decision variables, thus eliminating the need for Lagrangean multipliers;

8. using **MFTP** through a spreadsheet information is very difficult due to the volume of data needed to characterize the problem.

5 Recommendations

Based upon the previous conclusions, the following areas of research are recommended:

1. as was done with the time dimension, the underlying structure of the task and resource hierarchies must be used to reduce the dimensionality of the scheduling problem;

2. improved methods must be found for enforcing hard constraints on the schedules created by **MFTP**;

3. an interface must be created to easily feed scheduling data into the **MFTP** from the **PPO** (by way of the **ROSE** database).

6 Publications

"Graph Partitioning Using Annealed Neural Networks", D. E. Van den Bout and T. K. Miller III, published in the *Proceedings of the IEEE International Conference on Neural Networks*, Volume 1, pp. 521–528, 1989.

7 Hardware/Software

	Hardware Purchased	Qty
1.	**PM227-ER** DS3100S Server, 24MB, 3 RZ55 (1GB)	1
2.	**PM201-CH** DS3100, 8 plane color, 16MB, RZ23 (104MB)	2
3.	**PM200-AC** DS3100 1 plane mono, 8MB, diskless	2
4.	**RZ23-FF** 104MB disk for DS3100, field install	3
5.	**TK50Z-FA** TK50 drive for DS3100	1
6.	**QA-VV1AA-H5** ULTRIX documentation	1
7.	**LN03R-AA** PostScript laser printer	2
8.	**DELNI-BA** Local network interconnect	1
9.	**H4005** Ethernet transceiver	6

Software	Description
MFTP	the Mean Field Task Planner
GanttView	an X-11 based graphics program for viewing the schedule output by **MFTP**.

DICE PROGRAM

Task 3.3.3.10 Knowledge Server **West Virginia University**

A. Objectives:

The Knowledge Server (KS) is a tool to browse through sources of information (called Knowledge Centers) on the DICE system. These sources of information include the DICE Blackboard (DBB) which contains data reflecting the current state of the design and the Product-Process-Organization Model (PPO) which provides archived data of past designs. Just as a design engineer refers to design handbooks or other reference material to clarify points of interest, a DICE user can invoke the services of the Knowledge Server to ascertain information related to the task at hand. The KS requirements include the means to view both textual and pictorial information, easy-to-use interface, and it must be "smartless".

B. Approach:

In the context of the DICE environment, the basic requirements for the KS are translated to mean its capability to:

- display specific information (e.g. charts, graphs, pictures, parts and tables;
- view specific reference information (e.g. cost, process or design operations)
- search through large portions of data via queries in an SQL-like language;

and examining the consequences;

- provide data validation to prevent the entry of erroneous data;
- provide data isolation between users (so that such experiments are selfcontained), and
- employ user interest profiles to screen out undesired information, thereby reducing visual clutter.

Knowledge Server interactions will be via a user-interface process invokable from the user's workstation. It is expected that the user's workstation will be capable of running X-11 Release 3 software. Since the process of browsing is a typically human activity, there is no provision for a function-level interface to the Knowledge Server. During Phase I of the DICE project, a version of the Knowledge Server was developed that operates on an internal Knowledge Center populated with representative information. It allows multiple users to access it simultaneously and utilize its services. Each user, who accesses the KS is provided with a separate user-interface process to facilitate a speedy response.

C. Technical Results:

The primary effort in the development of the Knowledge Server was in the following areas: architecture, knowledge-base, user-interface, query management and data organization and validation. The archtiecture of the Knowledge Server was designed as a multiple process system consisting of one large Knowledge Server process interacting in a "round-robin" fashion with a number of user-interface processes. Each user-interface process interacts with the user to elicit the command, then transmitted it to the Knowledge Server for processing. The results of the command are then, transmits from the Knowledge Server to the user via the user-interface process. A communications protocol was developed to handle the data transfer between the Knowledge Server and the user-interface processes. In order to allow users on remote workstations to connect to the Knowledge Server, a invocation program was developed which initiates a request to the Knowledge Server. The invocation evokes the Knowledge Server menu on the user's workstation. Even though the Knowledge Server is deployed on a remote computer, for all practical purposes, the Knowledge Server appears to be located on the user's workstation. Moreover, a data representation format was developed to support browsing operations. It provides a framework on which to base the functionality of the KS and with which a knowledge base of information can be assembled. Information about materials, blades, etc. are assembled using this data representation format.

Provisions were made for a simple "what-if" type mechanism which allows a user to make changes to data and to determine the effects of that change. Data validation routines were developed which ensures that only appropriate data are entered. Checks for data type, range, cardinality, etc. were developed. A formula analyzer utility was incorporated with a generic demon to propagate the effects of any change to other affected data. Likewise, isolation procedures were employed by the KS architecture permitting a user to experiment with the data being browsed.

An SQL-based query language utility was developed to operate on the contents of a knowledge center. Such a utility allows a user to sift through the contents of a knowledge base for those entries which are of particular interest . Once identified, the relevant data can be examined. A mouse-based user interface, geared towards information browsing, was also developed. It provides a means to print text information, displays pictures, and processes queries. Each request is handled in a separate window and the results of the queries remain on the user's screen as long as required. The print window was wired in with the "what-if" mechanism and the data-validation constructs. The user can print the data and then experiment with it. The picture display utility causes bit-map images to be drawn on the screen and buttons on the display window allow different views to be accessed. For example, when an aircraft engine blade is shown, the user can also examine its dovetail, airfoil or other such images. In order to reduce visual clutter, a user-interest profile mechanism was developed causing only information that is of interest to the user to be displayed.

D. Conclusions:

The primary objectives of the Phase I deployment of the Knowledge Server (KS) have been achieved. Its functionality, as a tool by which users may browse through information through an easy-to-use interface, was successfully demonstrated. It was tested with a number of simultaneous users connected to it without any noticeable degradation in performance. The Knowledge Server can now be deployed in a distributed computing environment and be a valuable tool in the DICE environment.

E. Recommendations:

The Knowledge Server software should be enhanced to integrate it with the other components of the DICE architecture. The primary requirement is for the Knowledge Server to operate on multiple sources of information that may be available in the DICE environment.

There are three primary integration issues which need to be addressed during Phase II of DICE: data integration, user-interface integration and architecture integration. User interface integration is the goal: to provide a uniform "look-and-feel" to DICE software. This objective can be achieved by ensuring that all user interfaces developed under DICE employ a common X-windows toolkit library. When the DICE Architecture Group decided to adopt the TAE-Plus toolkit software and to modify current interfaces, this endeavor was enhanced. In order to accomplish data integration (the process by which data can be moved from one system to another in the DICE system), there will need to be a definition of the DICE data exchange standard. The definition of the exchange standard is critical for the proper development of software for Phase II tasks.

F. Publications:

There were no papers published during the Phase I period. However, there are plans for publications relative to the Knowledge Server in the next phase.

G. **Hardware and Software:**

No computer hardware was either purchased or developed during the course of performing this task. However, one license of Objective-C software was purchased from Stepstone Corporation. The Knowledge Server software was developed in Phase I of DICE.

DICE PROGRAM

Task 3.3.3.11 Information Modeling West Virginia University

A. **Objectives:**

To develop a tentative design of a Data Model for supporting engineering design, to allow concurrent development of the Architecture modules at WVU, without waiting for the ultimate PPO.

B. **Approach:**

Important modules, like the Blackboard and the Concurrency Manager, require data base services which will be ultimately provided by the PPO. In the interim, before the PPO is ready, the approach was to model these data bases for limited purposes within those modules.

For example the Concurrency Manager modules used the Unix data base manager, (N)DBM, to model the data base of Users, Applications, and Machines in the DICE network and the Blackboard used LASER knowledge bases created with schemas defined with that tool, for the Tasks, Designs, and Assertions.

C. **Results:**

Using this approach, it was possible to show demonstrations in July, 1989, which allowed the look up Users and send messages to machines via the Concurrency Manager with a degree of transparency. Similarly, remote applications could be looked up and invoked with registered messages.

The second approach allowed the loading of initial designs in the July demonstration into the Blackboard. LASER knowledge bases also had the information necessary to hold Tasks and disseminate them. Sending of Alert messages based on the dependency among perspectives was another demonstration of the use of previously stored information for assistance in the design process.

D. **Conclusions:**

This approach has been shown to be useful and work in this direction must continue in order to permit the development and validation of the Architecture tasks independently.

E. **Recommendations:**

The tasks undertaken must continue to be supported. They embody many of the concepts of rapid prototyping and incremental development. The data model of the Concurrency Manager must be fully ported to various platforms. At some point, the Blackboard needs to shed its dependence on LASER knowledge bases for design intelligence and acquire it from the PPO, via object exchange messages, yet to be defined.

F. **Publications**

None

G. **Software**

None

DICE PROGRAM

Task 3.3.4 Hardware Study **West Virginia University**

A. **Objectives:**

The purpose of this task was to evaluate and recommend hardware for the Concurrent Engineering Research Center (CERC) computing environment.

B. **Approach:**

Equipment selections included in the computing environment were *determined by:*

- discussions with other participants in the DICE project;
- discussions with the task leaders;
- requirements of the software;
- budget restrictions, and
- discussions with vendors.

C. **Technical Results:**

None

D. **Conclusions:**

The conclusion of the study is that the use of workstations and personal computers with local disk servers is an appropriate configuration for the CERC.

E. **Recommendations:**

As new hardware becomes available, it will be necessary to evaluate it for possible inclusion in the CERC computing environment.

F. **Publications:**

None

G. **Hardware:**

None

Task 3.3.5.1 CERC **GE-CRD**

Objectives

The major objectives for this activity revolved around the preparation for and execution of concurrent demonstrations at CERC. Specifically, the task objectives were to:

- Reach consensus with the CERC (and other DICE program contributors) on CE architecture goals, directions, and demo-specific implementations
- Integrate GE-developed architecture modules and GE Aircraft Engine supplied application methods, tools and advisors with CERC architecture methods and CERC-developed application methods, tools and advisors into a functional system.
- Carry out a concurrent demonstration of the DICE-developed CE system, illustrating its architecture, methods, tools and advisors for DARPA and other selected reviewers.

Approach

A multi-phase approach was pursued in Phase 1. It consisted of:

- Team building exercises
- "Redbook" preparation
- Concurrent scenario finalization/demo planning
- System integration
- Concurrent Demonstration

Technical Results

Team building, scenario finalization and demo planning were important and successful activities, but since they did not *directly* produce tangible results not covered elsewhere in this document, they have not been reported on. Specifically, listed below are the concrete results of activities associated with this task.

"Redbook" Preparation

A "Redbook" of Functional Specifications was prepared in November of 1988, four months after the start of the DICE program. The"Redbook" was a very early attempt to convey the DICE vision, the methodologies the DICE team planned to follow, and descriptions of architecture modules to be produced under DICE.

System Integration

Integration of architecture and application methods software produced by GE and CERC/WVU was initiated prior to the July 1989 Phase 1 DICE demonstration. In spite of the extremely tight initial development schedule, limited integration between GE and CERC software was achieved in Phase 1, and demonstrated in the "Concurrent Demonstration" described below.

Concurrent Demonstrations

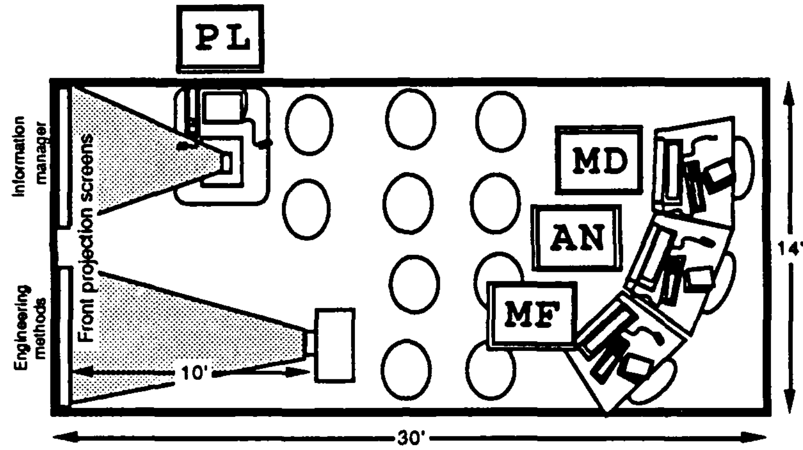
The objectives of the concurrent demonstrations were (1) to show some of the different ways in which product development can be made concurrent; (2) to work through transactions which show how the information systems framework expedites concurrent product development, and; (3) to serve as a starting point for organizations developing concurrent product development systems. However, the interactions among roles, transactions, applications tools, and systems services are very complex in real concurrent product development activities. As a result, concurrency is hard to understand in the usual linear viewgraph presentations, in conventional written specifications, or in sequential computer demonstrations. The major challenge in a concurrent demonstration is presenting the multitude of parallel activities and underlying transactions without confusing the participants. The focus of the Phase 1 demonstration was a one-hour program tailored for senior technologists and management. Major features of the demonstrations included:

Methods: While some engineering analyses had to be precomputed due to the time constraints of a compressed-time demonstration, two or three iterations of a valid product development process were shown.

Services: At key points, some of the details of the underlying system services were illustrated by examining underlying schema, files, data flows, and data structures.

Products: The Phase 1 demonstration simulated the production of a part or parts derived from design parameters defined during the demonstration.

In the present one-hour concurrent demonstration, the four members of the demonstration team (e.g., MD, AN, MF, and PL for structures) work through a compressed-time execution of the concurrent scenario (with some variability) using four workstations and their supporting servers. In the physical layout shown in the figure below, the project lead (PL) workstation and its activity monitor are always projected on one screen in order to help the reviewers track the scenario as it unfolds. Either the activity diagram or the concurrent scenario, (they are both described in the section on Task 3.3.3.5.2), were displayed on the activity monitor.



A physical layout for structures demonstration at CERC

Conclusions

Much was learned during Phase 1 about how to integrate the activities of geographically distributed development groups, but nevertheless consensus was difficult to achieve on architecture requirements early in the program,. The "Redbook" shows evidence of this problem; it has many inconsistencies, and does not distinguish clearly enough between short term and long term goals.

Integration activities were not initiated early enough in the development cycle, and fewer elements were integrated than expected.

The CE prototype demonstrated in Phase 1 proved to be a valuable vehicle both for developing an understanding of the concurrent engineering process, for testing an information systems framework which expedites concurrent product development in large manufacturing organizations, and for communicating the DICE vision.

Recommendations

- Distinguish clearly in future documentation between "future" architecture, methods ideas and extensions, and what is actually being developed *and demonstrated* in the current phase.
- Publish a "demonstration book" describing the modules which make up the current demonstration.
- Revise the demonstration planning schedule to reflect more realistic estimates of the time needed to achieve full integration.
- Future activities should involve expanding the pilot processes, extending the demonstrations, and enabling additional flexibility in design choices.

DICE Program

Task 3.3.5.2 GE Aircraft Engine Integration GE-CRD

Objectives

The major objective in this task for Phase 1 was to provide realistic problem focus, requirements and specifications for the CE architecture development and subsequent demonstrations.

Approach

The objectives were pursued through a sequence of activities which provided the framework for later developments. The activities were:

- Acquire a representative pilot or model (subset) problem, broad enough to encompass the key CE issues, and small enough to be credibly addressable with the resources at hand.
- Collect descriptions of the product/process activities used in the model problem
- Capture these major transactions for the model problem in the form of an as-is storyboard.
- Evolve a to-be storyboard, incorporating concurrency and addressing future user needs.
- Create a concurrent scenario.
- Identify appropriate application methods tools and advisors to support the scenario.

The outputs of this activity (concurrent storyboard and concurrent scenario) are *deliberately* user-oriented, rather than architecture-oriented, and express primarily user requirements rather than architecture requirements. Early in the program, this scenario served as a "flight simulator" for CE architecture concepts: it provided a basis for assessing the roles, functionality and overall utility of proposed architectural modules, without predisposing the CE architecture developers toward one architectural implementation solution or another.

Technical Results

The results of these technical efforts are organized by activity, and are listed below.

Model Problem Specification

The key requirement in identifying a suitable model problem is that the activities in the specific model problem be representative of product development in the organization as a whole. If this requirement is met, a system designed for the pilot activities will scale up smoothly into a solution for the whole organization. For an aircraft design example, the aircraft itself is too complex; a major subsystem such as an aircraft engine is too complex; even engine subsystems such as the fan, compressor, combustor, and turbines (HPT and LPT) are too complex. A good choice for a model part is a single turbine blade.

Product/Process Descriptions

Once candidate parts were selected, descriptions of the activities involved in developing, producing, and supporting the part were captured and documented. Careful selection and analysis of these activities was more important to the ultimate success than the choice of the part itself. Because the full collection of activities is too complex to be captured in detail, informal data flow and structure diagrams were most helpful in getting started. Developing these diagrams was surprisingly difficult because the process understanding is usually spread over a large number of people within multiple groups and extensive formal documentation of the product development process is usually not maintained.

Concurrent Storyboards

After the target product development activity was defined and an initial structured analysis completed, the next step was the construction of a concurrent storyboard. The objective in this activity was to define the major transactions in the product development activity so that a smaller and more detailed concurrent scenario could be selected and documented in detail. Most useful are an *as-is* scenario for understanding the current design process (which tends to be a long strung-out sequence of steps) and the concurrent *to-be* scenario for the CE testbed (which usually involves three to five parallel tracks).

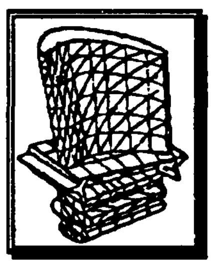
The DICE conventions for concurrency and dependencies are compatible with most storyboard formats. In particular, the DICE concurrent scenario for structures is organized on a long wall as an array of sheets. Each row of the concurrent storyboard array is a left-to-right time sequence of sheets representing the transactions for the particular person, group, mechanism, process, or organization. Each column of the array represents a particular time interval. Dependencies, major dataflows, and significant events can be indicated by lines, markers, and notes superimposed on the array.

The first task in building a concurrent storyboard was to identify roles. The initial choice for the turbine blade design example was: project leader, aerodynamics, mechanical design, and materials (materials includes pilot process development at GE Aircraft Engines). Then each of the major transactions was captured as a letter-size sheet. Each sheet describes a small group of related transactions. The sheet is composed of a state image in the upper half (a graph, terminal screen, data structure, or diagram indicating the current system state or activity) and descriptive text in the lower half (indicating *how who* did *what* to *whom* using *which* tools *when*). Storyboard or presentation graphics programs such as PowerPoint™ are helpful in preparing the large number of sheets required (four rows of forty sheets for the blade example). Two representative sheets are illustrated below.

Integrate/Mesh/FEM Analysis Results

STRESS ANALYSIS RESULTS TABLE

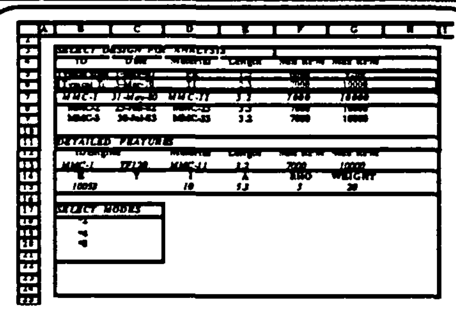
Node	Stress	Deviation	Rating
...			
2074	3.89	+2.5%	Nom.
2075	2.71	-8.2%	Under
2076	3.06	-0.2%	Nom.
2077	0.77	-11.2%	Under
2078	2.28	+1.1%	Nom.
2079	1.52	+1.3%	Nom.
2080	2.28	+1.8%	Nom.
2081	3.13	+2.2%	Nom.
2082	2.47	+1.3%	Nom.
2083	1.83	+2.8%	Nom.
2084	2.09	+2.2%	Nom.
2085	1.62	-11.4%	Under
2086	2.43	-0.0%	Nom.
...			



Description:
AERO has redesigned the airfoil and made it thicker in order to overcome the stress problem at the airfoil/platfor interface. Once the initial, redesigned, streamline model fr AERO became available, MECHANICAL had it converted to a CAD model and joined to his dovetail/platform model. He then generated a mesh and ran the FEM analyzer program in order to obtain the above results.

Action:
MECHANICAL observes that the airfoil redesign has significantly reduced the stress at the airfoil/platform interface. He will now make several modifications to the interface and reanalyze until the blade is fully compliant wi the interface stress constraint.

Frequency Analysis



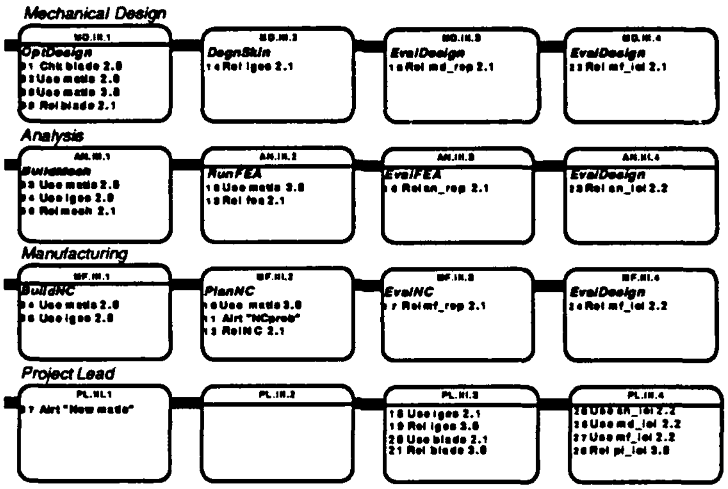
Description:
MECHANICAL has successfully modified the airfoil/platform interface to meet the existing constraints a now continues to perform further analyses. This is a frequency analysis package used to help identify any blade excitation frequencies that correspond to expected vibrator loads.

Action:
MECHANICAL specifies the RPM harmonics of interest and the design under current consideration. The system the generates input files through tool wrappers which include tr user specified modes as well as the design data from the da base. The system collects the output files from the analysi tools and imports them into the spreadsheet for plotting anc further analysis.

Two sheets from the turbine blade concurrent scenario

Concurrent Scenario

Once the target product development activity was defined and the initial structured analysis (data flows, data structure, and concurrent storyboard) completed, a small subset of the sheets was selected and the transactions worked out in detail. Much like the storyboard, each row of the concurrent scenario is a left-to-right time sequence of boxes representing the tasks for the particular role. Each column of the array represents a particular time interval. Again dependencies, major data flows, and significant events can be indicated by lines, markers, and notes superimposed on the array. Prior reviews of the storyboard will usually generate changes in roles and focus from the storyboards. For example, in the concurrent scenario shown below, in response to reviews indicating a need for stronger manufacturing emphasis, the roles were changed to: Mechanical Design (MD), Analysis (AN), Manufacturing (MF), and Project Lead (PL).

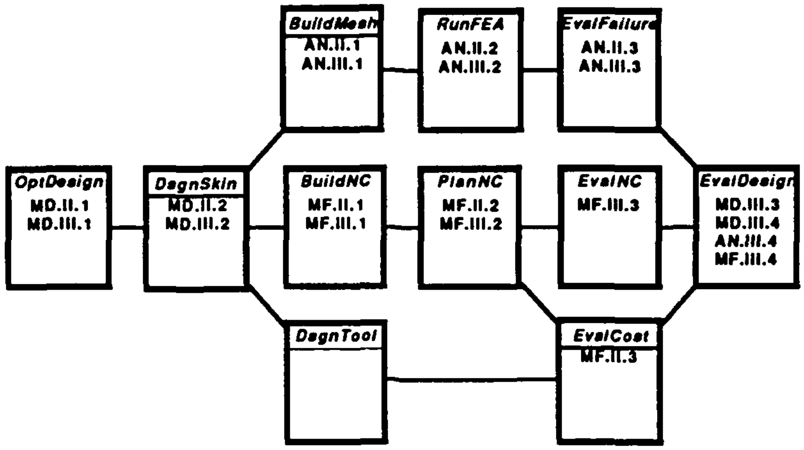


A concurrent scenario fragment

The concurrent scenario is divided into several phases (usually three to five) representing initialization, a few design iterations, and termination. Each phase contains perhaps three to five tasks. In DICE, the concurrent scenario is captured in two different forms. The summary form of the concurrent scenario diagram (shown above) is displayed on the activity monitor during demonstrations and used as a management planning aid while the DICE system runs. Each of the blocks includes the task identifier (e.g., AN.III.3), an identifier indicating the activity being performed (e.g.,Eval FEA), and a list of events posted by that task. In the abbreviated form the focus is on transactions to and from the Product-Process-Organization (PPO) model (which is described in more detail later). The events are described in a simple structured English. Typical events include:

Rel	Release	Release file into the PPO; notify interested parties, and distribute.
Chk	Check	Check out files from the PPO in order to modify the design
Use	Use	Check out files from the PPO; subscribe to change notices
Alrt	Alert	Post an alert concerning and even. or problem

Another useful documentation tool is the activity diagram shown below. It illustrates the major activities in the testbed and the specific tasks in the concurrent scenario which will be implementing them. As in the case of the concurrent scenario described above, the abbreviated form is illustrated here: it appears on the activity monitor screen during demonstrations and can be used by management for planning the detailed interactions. Both the activity and concurrent scenario diagrams are also maintained in more detailed forms for written documentation.



Activity diagram (fragment)

Conclusions

The sequence of activities in this task turned out to be an excellent way to begin addressing the objectives. Storyboards and concurrent scenarios are intuitive, easily understood, and elicited feedback from a much broader spectrum of end users than we would have obtained had we adopted a more formal methodology as the communications medium.

Recommendations

Continue using the approach outlined above, but supplement it with more conventional specifications and requirements in the next program phase.

DICE PROGRAM

Task 3.3.5.4 User Interfaces and Graphics Support System — **West Virginia University**

A. Objectives:

The objective of this task was to build interfaces to the software of Task 3.3.2.5 (Architecture Definition) using an interface-building tool based on standards and following guidelines for consistency.

B. Approach:

From the beginning of the task, the approach focused on developing a mockup of some of the interfaces using an in-house tool built on Sun's proprietary windowing system.Later, the approach selectively built interfaces using the X-Window *software directly. At the demonstrations* July, 1989, Demonstration, this approach was used for several modules exclusively.

C. Results:

As a result of the research efforts, the following efforts were made:

- The Blackboard module was demonstrated in July using a number of user interfaces and menus directly with X-lib;
- The Concurrency Manager was shown as a standalone using the same approach for the remote application invocation module; however, the interface was custom-built for a sample application only;
- The Traffic Monitor was also demonstrated with X-lib using some higher primitives, including the Athena widgets. The interface was reasonably sophisticated with mouse selection from lists, with dynamically changing histograms showing plots of on-line data;

- The *Cooperate* and *View* utilities were also developed with user interfaces built on X-lib and the Athena widgets, and
- The Diagnostic Monitor was shown bare, without any attempt to organize data in a pleasing manner.

D. Conclusions:

While the approach has been shown to be successful, the further work is needed to develop all the interfaces yet missing and move to a building tool like TAE-Plus, itself based on X-windows.

E. Recommendations:

Tasks undertaken must continue to be supported. The adoption of TAE-Plus will simplify development, but there are a number of problems associated with providing it sufficiently flexible, using the latest version depends on C++.

The task intends to expand its scope of the user interface tool to enhance consistency by using of rigorous guidelines and templates.

F. Publications:

None

G. Hardware:

The following hardware was acquired:

OUIB (a tool built at West Virginia University), and

TAE-Plus (no-charge software in beta test).

DICE PROGRAM

TASK 3.3.5.5 Operating System Interfaces. West Virginia University

A. Objectives:

The primary focus of this task was to study various operating systems and determine the features of these systems and their impact on the integration and implementation of software developed in the project.

B. Approach:

Initially, three systems were the focus of this task: Unix, VAS/VMS and Mach. This list has now been expanded to include the new multiprocessor systems by BiiN and Silicon Graphics.

C. Technical Results:

This task was redefined to develop and maintain the computing environment. Recently, the CERC employed two highly qualified professionals to direct task activities required for the project. Moreover, this task, (its money and personnel), was incorporated into the Administrative and Support Services Task (3.2.2). This is an "ongoing" task that will be a part of Phase II and have a continual need for personnel to develop, manage and maintain a diverse computing environment. The consolidation of this task into the Administrative and Support Services Task places all full time CERC personnel performing service, not research-related tasks, into a single unit whose efforts will impact on the center directly.

D. **Recommendations:**

There is a need to establish an additional staff employee position to maintain the maximum level of effort required to adequately perform a full range of responsibilities and requirements for the CERC computing Systems.

E. **Publications:**

None

F. **Hardware:**

See hardware list under Task 3.2.1.1 - The Computing Environment.

This task is still using the facilities of the existing Artificial Intelligence Laboratory. Courses are being held on the X Window system providing operating system independence for the project.

METHODS, TOOLS, AND ADVISORS FOR CE

DICE PROGRAM

Task 3.4.1.1 Process Planning Advisor West Virginia University

A. Objectives:

Four major research objectives were undertaken under this task in Phase I:

1. Develop a knowledge base system for top level process selection. During the first phase, investment casting, powder metallurgy, closed die forging and milling processes were considered;
2. Define the parameters of a turbine blade in order to generate a Numerical Control (NC) program from it;
3. Analyze the turning and milling processes and select the optimal combination of machining parameters; and
4. Develop an integrated methodology to evaluate forging used to produce components made of advanced materials.

B. Approach:

Approaches to these objectives involved a number of strategies:

1. The process selection procedure is mainly based on design information, manufacturing experience and process capabilities. In addition, most of the top level-process selection problems are not quantitative in nature. Hence, an AI approach to process selection was adapted. A rule based expert system shell, VP-EXPERT, was utilized to handle the process knowledge. Process requirements and part geometry were the primary inputs to the system within process knowledge, described in terms of rules;

2. The machining parameters of a turbine blade were specified and Fortran-based software was written to process the parameter file. The parameter file was generated via a text editor, although It could have been produced by another software which extracts geometric features from a CAD database. The complex airfoil surface requires a parametric approach to NC programming. The NC program generator attempts to minimize production cost and production time while satisfying the specified tolerance constraints. Production cost and production time information are provided to the designers in order for them to evaluate the manufacturing process;
3. The analysis of turning and milling processes relies on a combination of knowledge based expert systems, data base management systems, and algorithms. The machining process evaluation and the selection of the machine and cutting tools were accomplished using expert systems and data bases. The machinability of various materials is based on properties including material hardness, yield strength, and heat treated conditions. During this phase, fundamental research was completed to develop the large knowledge bases which contain information about conditions resulting from the machining of any material with a given set of properties; and
4. The evaluation of the forging process was achieved through demonstration of the process modeling concept. Forging was found to be a versatile method of producing net shaped components. Finite element methods and various material and friction modeling techniques were also employed in this evaluation. Short term tasks included simulations of the selected technical process, while the long range tasks included development of new computational procedures

to model complicated processes that occur while processing material.

C. **Results:**

1. One of the paramount findings resulting from this task was that an appropriate process can be selected based on manufacturing process parameters such as weight, roughness, hardness, materials, tolerance, strength and geometry features such as length, section thickness, corner radius, fillet radius. The characteristics and capabilities of manufacturing processes, such as investments casting, closed die forging, surface milling and powder metallurgy were reviewed with respect to parameters and related rules. A final expert system software was developed as an advisor to the designer.

2. The blade definition for NC machining has also been completed in terms of geometric parameters specified in a parameter file. The turbine blade surface was decomposed and several parameters were identified.

3. A software module was also developed and demonstrated for efficient analysis of turning and machining processes as applied to a particular example of the turbine blade. This selection of the manufacturing activity, from among the menu items, leads to the specification requirement of a particular turbine blade feature. The selection is made possible by declaring "hot spots" on the computer screen corresponding to the blade features, clicking on any feature of the blade image, and the system can, then, understand which feature

was selected. Once the feature is selected, the system can delve into the PPO to obtain information about the blade material, design features and feature parameters. The information obtained is used by the rule-based expert system to decide the feasibility of the machining process with respect to the materials properties.

4. On July 25, 1989, a general methodology was demonstrated to simulate the isothermal closed-die forging process. A special purpose finite element code, meant for large deformation analysis and available in **ADDS-Forming**, was used to conduct a 2-D simulation of the forging of a L-section part. The material chosen for this demonstration was Al2024. An appropriate model for this material was interfaced with the finite element package to ensure that behavior of the material under deformation was realistically represented.

D. Conclusions:

1. From these results, a number of conclusions can be made. The "process selection" technique enables early consideration of downstream manufacturing operations. The approach can also be applied during detailed design; however, its limitation is in extracting relevant reometric features from a CAD database. Feature extraction is being pursued elsewhere. An alternative to feature extraction is to design by features;
2. Several features of turbine blades were also defined. A Fortran program was written to read the geometric parameters and automatically generate the NC program for the EMCO CNC F1 milling

machine. The program was verified by actually milling the airfoil out of machinable wax;

3. The fundamental research on machinability, tool material selection, and machine selection resulted in some important rules that can be generally applied with suitable modification to any other domain in manufacturing. The attempt here is to find the underlying basic factors and reasons for these aspects which give reasonably good results which can be published on their own, and
4. Forging simulation is a useful tool in the concurrent engineering environment because it allows early consideration of forging feasibility and constraints upstream in the design process.

E. Recommendations:

1. Future work should be directed towards integrating feature based design and expert systems as well as the expansion of the knowledge base to parts other than turbine blades.

   A reasoning mechanism for recommending the most appropriate (optimal) process in terms of cost, production rate, production capability, available equipment, operating experience, geometric features, lead time, and specific user requirements should be developed;
2. Feature-based part design could facilitate detailed process planning activity. The extraction of features from a CAD database is cumbersome. As information on PDES becomes available, it should be utilized in the process planning activity. In the meantime, the IGES file

format should be followed for the purpose of interfacing with a CAD system;

3. The major focus of future research in the machining analysis area should be the integration of a CAD system with the DICE architecture. The cost model should be made more design-sensitive to substantially help the evolution of the design process. The development of cost models could facilitate the analysis of cost considerations for machining an entire object, instead of one operation at a time.
4. With the finite element package, the forging analysis work conducted in Phase I is essentially restricted to 2-D simulations. Further work should be pursued in developing methodologies to extend this work to 3-D simulations. State-of-the-art work in developing of computational procedures will help solve highly non-linear problems. In the realm of experimental work, two remaining areas of interest are the characterization of materials under deformation at high temperatures and the generation of databases on interfacial friction between workpiece materials and tool surfaces.

F. Publications:

None

G. Software:

During Phase I, VP-EXPERT, an expert system shell on IBM PCs, was utilized to handle the manufacturing knowledge. The machining analysis software developed in this task included the MPSS (Machining Parameter Selection System) and MCM (Machining Cost Models). The

MPSS was developed using VP-EXPERT, a commercially available expert system shell and MCM was developed using FORTRAN 77.

A comprehensive die design package for die design applications in forming, called **ADDS-Forming,** was used for this research task. This software creates the required material model and the required deformation model to simulate the process.

DICE PROGRAM

Task 3.4.1.2 Geometric Modeling West Virginia University

A. Objectives:

The research undertaken in Phase I under this task had four major objectives:

1. To develop the PARFEM module for the DICE environment to synthesize solid modeling strategies into a simple logical procedure based on an engineering parameter definition of the part being modeled (e.g. turbine blades);
2. To develop the MODFEM module for the DICE environment to model and characterize families of parts (e.g.. turbine blade segments) through modal features and substructuring techniques based on the parametric definition obtained through the PARFEM module;
3. To facilitate the efficient and accurate definition, representation and transfer of geometric data among various software packages including CAD systems, and
4. To develop free-form curvature-continuous surface representations such that control points have local not global effects.

B. Approach:

1) PARFEM: In the development of this DICE module, several techniques have been used and integrated to synthesize both the

solid modeling and the finite element modeling strategies into one simple (user friendly) logical procedure through:

A) Parametric Approaches. The engineering parameters used to define a part (e.g.. turbine blade) are grouped into subsectors, which describe the configuration of a series of segments that can be individually defined.

B) Constructive Solid Geometry. The geometric representation of the series of parameters (above) are created through a combination of Boolean operations, skinning operations and extrusion of profiles through which the solid objects for each of the segments and their assembly can be generated.

C) Finite Element Mesh Generation. Discretization of the solid volumes into a finite element mesh is accomplished by breaking down the segments into subvolumes which, in turn, are individually meshed prior to overall assembly.

2) MODFEM: The technical approaches implemented in the development of this module involved;

A) Substructuring Techniques. Through these techniques, it is possible to create a substructure data base which contains geometry, finite element mesh, loads and boundary conditions and the matrices which characterize the substructure.

B) Guyan Condensation Techniques. For dynamic modal analysis of structures with many degrees of freedom, it is necessary to reduce the order of the matrices involved in the operations. This can be done through the definition of "Master Degrees of Freedom" which, in turn, allow the main modal characteristics to be extracted through a reduced set of modes.

C) Component Mode Synthesis. Once the modal characteristics of each substructure are extracted, the modal characteristics of the assembly can be obtained through "component mode synthesis" techniques, which involves the modal matrices of each substructure.

3) In order to transfer the geometric data representation of an object from the I-DEAS package to the TRUCE package, the IGES format was used . IGES (Initial Graphics Exchange Specification) is a man-machine (as opposed to a machine-machine) standard for CAD data exchange.

4) The approach for surface representation is based on the use of box- spline surfaces and their associated theory which has been developed over the last six years. Box splines are a generalization of one-dimensional splines in several dimensions in a formulation that includes, but is more general than, the widely used bicubic splines. For instance, a curvature continuous box-spline surface can be achieved with triangular patches of degree-four, while bicubic patches require degree-six patches. The approach used for curve approximation and curve offset generation is based on the theory of polar-splines, a new and efficient method of approximating curves. Two important properties of polar splines are:

   1. They avoid unwanted oscillations that sometimes appear when conventional spline approximation is used, and
   2. They are independent of the coordinate system.

C. Technical Results:

1. The major technical results associated with the PARFEM and the MODFEM modules are summarized below:

- A turbine blade configuration defined by 50 engineering parameters can be now created by the PARFEM module;
- A finite element model, using solid elements for the assembly of a blade, shank and root with boundary conditions, can now be created with PARFEM;
- Any parameter change can be made and the models revised to reflect these changes and "updated" models can be made available to other tasks in the form of IGES and finite element files;
- Analysis results can be displayed with either I-DEAS™ or ANSYS™ solver. Results may include stress analysis and/or dynamic modes of vibration performed with the MODFEM module;
- A turbine blade model created with PARFEM can be decomposed into three substructures in such a way that if any of them is modified only that segment need be revised Thus, CPU time can be saved in iterative blade redesign stages;
- A substructuring approach can be devised through the package ANSYS ™ to characterize a substructure through its modal characteristics; and
- Any substructure generated through MODFEM can be subsequently used for assembling more complex systems or superstructures (e.g.. disk with blades).

2. The July 25, 1989, demonstration showed that the IGES file transfer is usable for geometric data translation in a production environment. Currently, IGES is not widely accepted nor widely used in industry. The CAD geometric data transfers performed during the July demonstration illustrated the power and utility of IGES. This implementation of an IGES

transfer of CAD data begins the effort to make CERC a National Technical IGES Center.

3. In this task, an entirely new and general framework was developed for the construction of local interpolation methods; that is, interpolation methods in which each piece of data has only local influence. This new method of surface and offset generation was implemented in two demonstration software packages on the Silicon Graphics Iris. One package illustrates the interactive modification of a functional surface z=f(x,y), by changing prescribed z-values at the integers (x,y)-(i,j), and continually updating the graph of the surface. The other package reads an array of data points in three-dimensions and construct an interpolating surface, as well as an offset surface, with the magnitude of the offset interactively modifiable.

   The Polar-spline effort has resulted in new and efficient methods of curve approximation and offset curve generation. The results obtained include the now known existence of the polar-spline approximation, the algorithm to calculate the approximation, error estimates of the approximation and important properties of polar splines. Two of the more important properties are the avoidance of unwanted oscillations and the independence of the frame of reference. This generation of offset curves is an important part of the manufacturing process. Although, this work is preliminary and only partial results have been obtained, a number or publications were completed during this Phase. Moreover, the extension of polar spline approximation to 3 space dimensions was begun.

D. Conclusions:

1. The PARFEM and MODFEM modules are aimed at the concurrent engineering environment. They are not currently available through commercial CAD packages. Moreover, the generation of a solid model, and a finite element model, the extraction of modal characteristics, and the capability of repetitive use substructures for assembling superstructures are engineering tasks very much needed in the concurrent engineering environment . The major contribution of PARFEM and MODFEM resides in the reduced effort required to perform the above tasks for both the analyst and for the computer.
2. The geometrics modeling work performed so far clearly indicates that IGES translation, geometric data representation, more general product data representation and Box-splines and Polar-splines are essential components of Concurrent Engineering, in general, and the CERC effort, in particular. The new cardinal spline representation of surfaces, developed as part of the task, has met the Phase I objectives, as described above. Likewise, the polar spline work has proven a useful tool for curve approximation. Finally, the seminal work on offset curves and 3 space polar splines appears promising.

E. Recommendations:

More work is required to enhance and expand the present capabilities of both PARFEM and MODFEM, primarily in regard with their interfacing to other modules and utilities of the DICE framework. Further development efforts to improve surface and curve representations are better recommended methods for local subdivision, modeling closed surfaces, surface intersections, interfacing, offset curves and surfaces.

F. Publications:

(1) C.K. Chui and H. Diamond, "A general framework for local interpolation", Texas A&M University Center for Approximation Theory, Technical Report, 1989, currently submitted for publication.

(2) C.Q. Zhang, Polar spline approximation, Technical Report TR88-10, Department of Statistics and Computer Science, West Virginia University, 1988.

(3) C.Q. Zhang, Error Analysis of Polar Spline Approximation, Technical Report TR89-2, Department of Statistics and Computer Science, West Virginia University, 1989.

(4) C.Q. Zhang, Polar-Splines and Offsets, Technical Report TR89-3, Department of Statistics and Computer Science, West Virginia University, 1989.

(5) C.Q. Zhang, Polar-Splines Approximation, submitted to SIAM, 1989.

(6) C.Q.. Zhang, Existence of Offset, Technical Report, CERC. 1989.

(7) C.Q. Zhang, Polar Splines in 3-Dimensional Space, preprint.

(8) Ruth, Mary K. Parametric Design of Turbine Blades for Concurrent Engineering Environment , Masters Thesis, Department of Mechanical and Aerospace Engineering, West Virginia University, May 1989.

G. Hardware:

The hardware used in this task was CERC hardware and is described in a different section of this final report. The software used in this task included Commercial CAD packages, General Electric CAD and IGES representation and translation codes, and systems written by task members.

DICE PROGRAM

Task 3.4.2.4 Physical Models GEAE

Objectives:

The long term objective of this task is to construct and demonstrate an integrated material behavior simulator which links process history, composite mechanics and life analysis. The final simulator will be compatible with the overall DICE system architecture and will be linked to the other major elements of the overall system. It will be accessible to material developers and design engineers and will integrate the most current predictive models and material properties.

The specific objective for Phase I was to design and construct a prototype simulator.

Approach:

The construction and development of the prototype simulator was subcontracted to Theta Systems Inc, located at Woburn, MA. The prototype simulator will integrate at least two models and will feature a menu system, file management and graphics subroutines for generating plots. Flexibility will be assured by allowing each of the models to be run separately or in combination. The prototype simulator design will also facilitate revision and insertion of additional models in the model sequence as well as future enhancements and refinements to be incorporated in the final simulator.

The final simulator will consist of three modules, viz. a Process Module, a Mechanics Module, and a Strength and Life Module:

1. The Process Module will consist of models which transform constituent (such as fiber and matrix) properties and processing parameters into a description, both micro- and macroscopically, of the composite structure.

2. The Mechanics Module takes structural and constituent materials data and predicts internal and overall stress and deformation response to all fundamental loading states.

3. The Strength and Life module takes stress-strain information combined with externally applied loads (mechanical and thermal) and environmental conditions and predicts material strength and life.

Technical Results:

Two documents describing the Design Requirements and Functional Specifications for the simulator were provided by the subcontractor. Construction of the prototype simulator software architecture began in Phase I and will continue under Phase II.

Conclusions:

Recommendations:

Publications:

None.

Hardware:

None purchased against DICE funding.

DICE PROGRAM

TASK 3.4.3 MACRO MODELS GEAE
3.4.6 DESIGN RULES

Objectives:

The design and development of an advanced component can involve the consideration of a large number of factors. The factors that must be considered run the spectrum of predicting material properties on a micro-mechanical level all the way to predicting static, dynamic, and thermal responses of a component on a macromechanical level. In addition, a significant element of the design approach involves the application of appropriate design rules and constraints. When considering Concurrent Engineering methods, these rules and constraints must include input from manufacturing, materials, and quality to be successful. Since manufacturing, materials, and quality are not normally embodied in the present design rules and constraints as trade-off criteria, the challenge was to bring these considerations to the front of the product design cycle.

Approach:

The design practice appropriate to the component is followed to ensure that all component design requirements are met. In addition, criteria specific to advanced materials, manufacturing techniques and processes, and quality considerations are imposed and evaluated at the very early stages of the product development cycle.

Trade-offs in the design are considered with respect to mandatory design and evaluation criteria, design approaches, analytical methods, reference sources, limits, past field experience, failure modes, and other related information. All of the above are derived from the documentation of design approaches based upon proven experience within GE Aircraft Engines.

Technical Results:

The Concurrent Engineering scenario for a turbine blade was developed and used in the July, 1989 demo at the Concurrent Engineering Research Center at West Virginia University. This scenario included interactions between Design, Manufacturing, and Materials personnel in the very early stages of component design/development. In order to accomplish the above, the interactions between these groups were documented so that the interactions (currently sequential) could be developed into parallel or concurrent activities. This concurrency required new methods that met required design rules while being accomplished in a new (concurrent) way. For this reason, design rules and macro models were developed and programmed for real product development problems. The two components selected to work on following the turbine blade were a shaft and outer duct.

These components were selected due to their generic applications in products other than aircraft engines and because the development problems they present represent a wide and varied range of technologies and disciplines.

Conclusions:

Several individual/stand-alone models have been successfully tested for the duct and shaft problem during Phase I. This work will be continued and interfaced with the architecture during Phase II. Some models that existed on PC's had to be ported to the Workstation environment so that they would be more compatible with the interfacing activities required by the architecture.

The need to continually interface product development efforts (i.e. design, manufacturing, quality, etc.) with the computer system and programming personnel continued to be emphasized throughout this work. To attempt Concurrent Engineering without the above interfacing activities is a guaranteed formula for failure.

Recommendations:

Continue this work into Phase II and demonstrate the shaft and duct concurrency during the scheduled February, 1990 manufacturing demonstration.

The involvement of design, manufacturing, materials, and quality personnel must continue to expand as the architecture develops to handle more complex product development problems. If the architecture is developed with early involvement and inputs from the groups mentioned above, it will prove to be of great use to the end-users and will be more readily accepted/implemented in industry.

Publications:
N/A

Hardware:
N/A

DICE PROGRAM

Task 3.4.4.2 Mechanistic Models West Virginia University

A. Objectives:

The primary objective of this task is to develop practical analysis models that quantify relationships between design parameters, constraints, and product performance metrics in order to expedite the design process and improve its quality. Although the initial focus was directed towards analysis methods of composite materials, it was expanded, subsequently, to a broader range of design parameters, ranked in accordance with their effect on product performance in service.

B. Approach:

Three major directions of research were pursued simultaneously under this task:

1. Theoretical models were developed for predicting expediently the thermomechanical properties of engineering materials, especially particulate and fiber-reinforced composites. Phenomenological relationships between these properties and microstructural characteristics of particulate composites were established compiling an extensive database of experimental results and investigating their trends using curve fitting and parametric analysis techniques. Moreover, a new computational method, based on integrated numerical and experimental analysis, was developed to reduce the amount of CPU time required for the elastic-plastic analysis of fracture

propagation in metals. This method permits a significant reduction in the number of degrees of freedom by using experimentally measured displacements to simulate the loading conditions along crack boundaries. A new, closed-form analytical model was developed for predicting degradation effects that local debonding between matrix and the reinforcement may have on the macromechanical properties of a composite structure. This model is based on an "element mechanics" approach that provides a link between micro and macromechanical analysis of composite materials;

2. Software modules were developed that can demonstrate the benefits of employing parametric and simplified engineering models in the early design phases of the product development cycle. The parameter selection covers the main stages of the product development cycle from material tailoring to manufacturing.
   An extensive set of parametric relationships were also developed for stress and dynamic analysis of a turbine blade made of conventional metal alloys or particulate metal-matrix composites. Functional dependencies were derived from these results and subsequently utilized for the development of design rules based on maps, ranges and ranking of sensitivity coefficients. In addition, an expedient method for stress and dynamic analysis of turbine blades was developed combining simplified finite element modeling with feature-based descriptions of design configurations. This method relies on a one-dimensional finite element model that yields all the important modal characteristics of a rotating blade (including the effects of centrifugal stiffening and twist), and

3. An experimental set-up for non-destructive characterization and inspection of composite materials was designed and constructed. This experimentation involved both fracture toughness low-cycle fatigue. The set-up relies on two optical methods, (holographic and moire interferometry) that provide whole-field high sensitivity measurement (typically 0.4 µm gage sensitivity) with versatile capabilities, like static and dynamic measurements, on-site flaw detection or NDI part inspection.

C. <u>Technical Results:</u>

1. An example of phenomenological sensitivity plots for tailoring the microstructural composition of an aluminum-based metal-matrix composite material to specific strength requirements is shown in Figure 1:
   1. *The design rules that may be extracted from such plots are* illustrated in the narratives associated with this figure.
   2. The computer program for enhancing the computational efficiency of elastic-plastic fracture mechanics analysis was also validated against experimental results. Excellent agreement was observed. This agreement is illustrated in Figure 2 for the displacement field normal to the crack.
   3. Sample results of simplified parametric analysis of turbine blades are shown in Table 1 and Figure 3. Table 1 illustrates that the change of the thickness to chord ratio has little effect on the rotating natural frequencies of the blade. Figure 3 indicates that the sensitivity of the maximum stress at the blade root increases significantly with the aspect ratio, especially for tip twist angles larger than 15 degrees.

4. To demonstrate the capability of the photomechanical laboratory, (established in Phase I), preliminary moire interferometry tests were conducted to study crack growth behavior of a 5052-H32 aluminum single edge-notched specimen. Figure 4 shows typical u and v displacement moire fringe patterns of a 5052-H32 Aluminum SEN specimen under increasing load. As shown in Figure 4, each moire fringe represents a contour line of equal displacement and differs with its adjacent fringe with 0.8 μm displacement, (i.e. sensitivity is 0.8 μm/fringe).

D. Conclusions:

A first set of parametric design rules was developed for feature-based specifications of turbine blades made of conventional metal alloys or particulate metal-matrix composites (MMC). These rules rely on functional relationships and sensitivity studies of a comprehensive database compiled from experimental material properties, theoretical predictions of finite element software (ANSYS) and micromechanical models results for composites.

The displacement-input method for elastic-plastic fracture mechanics fulfills time and money saving demands. 35% CPU time was saved for the example considered. Since the displacement-input case is less sensitive, fewer iterations are required when the material is in the nonlinear range of the stress-strain curve.

E. Recommendations:

The parametric and simplified design capabilities need to be expanded. Those designs demonstrated in Phase I for metal turbine blades, should be extended to other structural components like ducts/shafts and continuous fiber-reinforced composites based on either metallic or polymeric matrices.

The experimental facilities for non-destructive inspection and on-line quality control also need to be expanded in accordance with the "rapid prototyping" strategy adopted by the DICE program. These facilities must go beyond the photomechanics laboratory established at CERC in Phase I.

F. Publications:

None

G. Hardware:

Equipment for the photomechanical laboratory consists of three types of components:

1. Components for material testing
   a. One clip-on displacement gage, and
   b. One 1000 lb. fatigue-rated load cell.

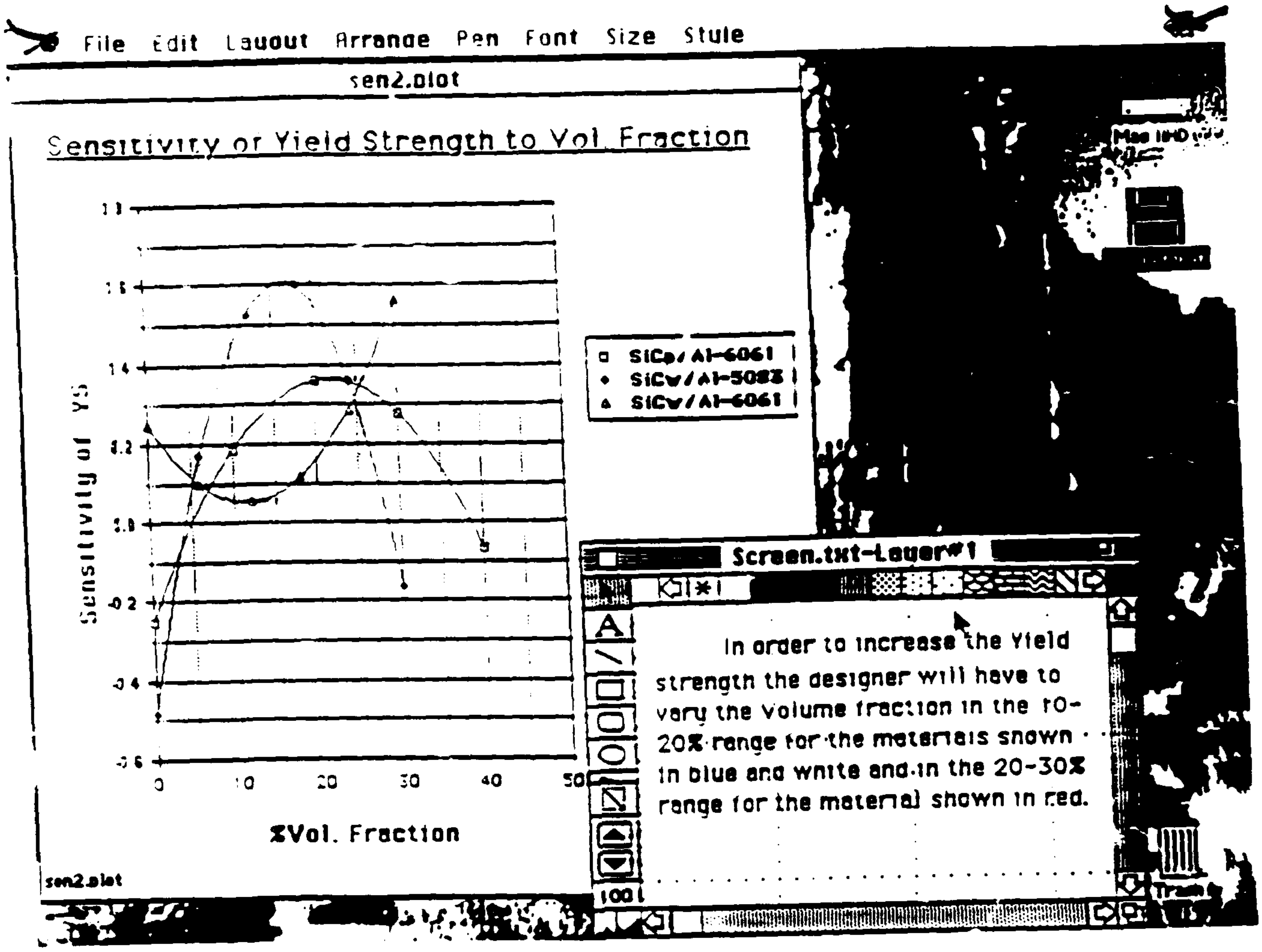
2. Components for digital image processing
   a. One MVP-AT real-time image digitizer board for IBM PC/AT, and
   b. One high-resolution solid charge injection device (CID) camera; and

3. Optical components for a moire interferometry and holographic interferometry setup:

A typical moire interferometry or holographic interferometry optical setup consists of an optical isolated table system, laser, assorted mirrors, spatial filters, adjustable tilting, rotation and translation stages and several other small optical components.

Table 1 Comparison Of Rotating Natural Frequencies (rad/s) for Two Different Thickness to Chord Ratios

	NACA 0009	NACA 0021
Mode 1	1074.7	1089.2
Mode 2	2642.2	2741.1
Mode 3	4319.5	4790.3



The figure above shows the sensitivity of Yield Strength of some whisker/particulate reinforced metal matrix composites to their Volume fraction. These curves have been obtained by taking the first derivative of the equation that describes the relationship between the two variables. From the graph it can be observed that changing the Volume fraction in the range where there is a peak in the curve, will result in the most significant change in the value of Yield strength.

Fig.1 - Example of Sensitivity Plots for Material Properties

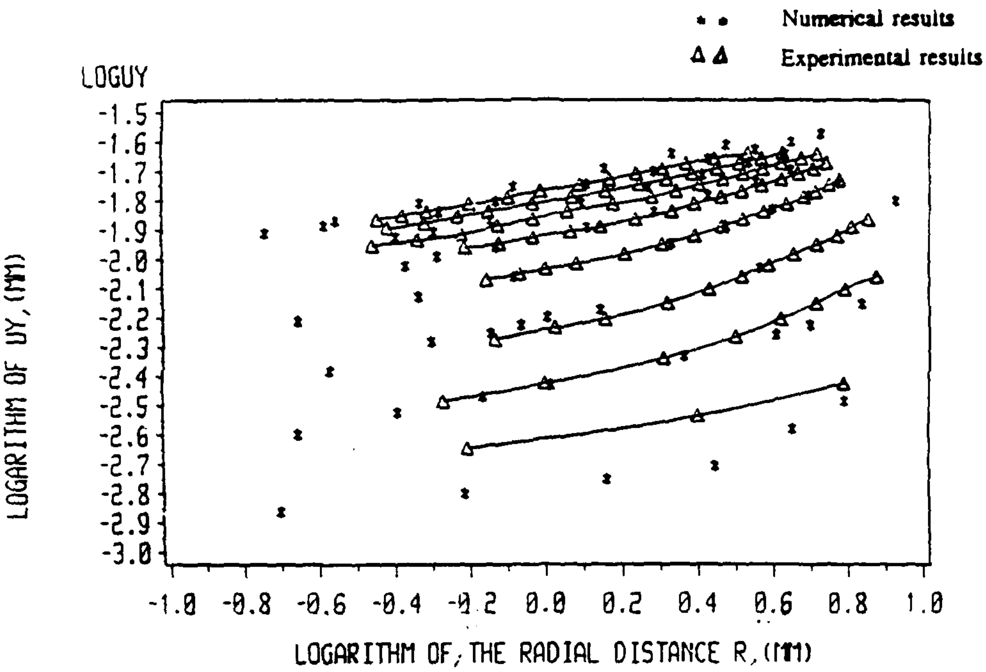


Fig. 2 - u_{yy} displacement fields of Kang's experimental and FEM results at L.F. = 1.16

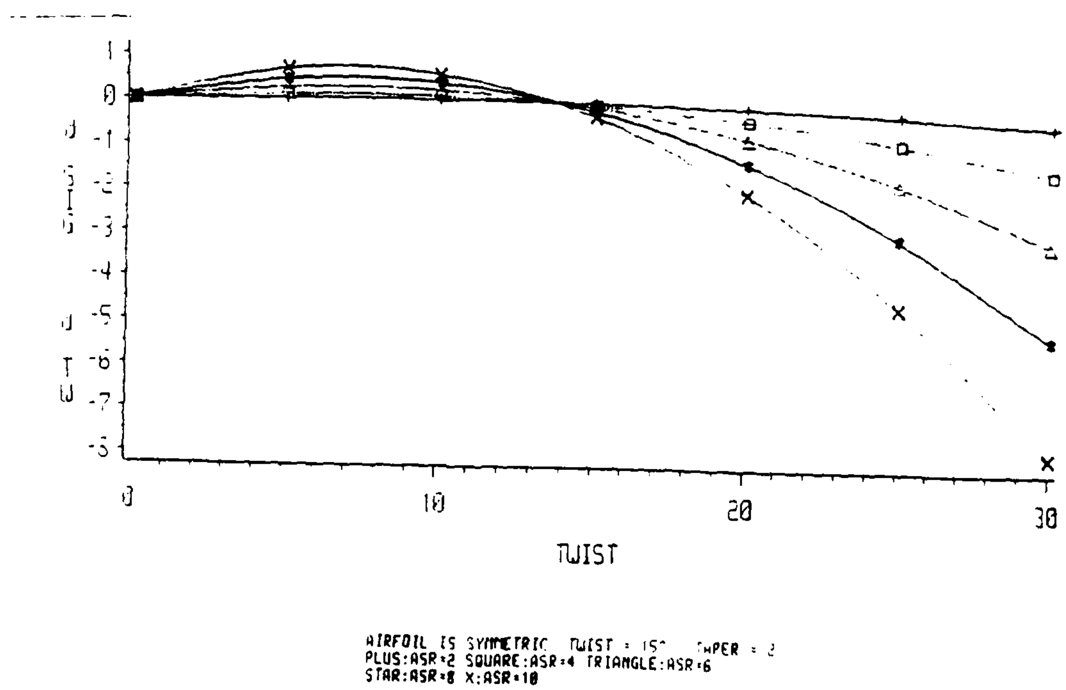
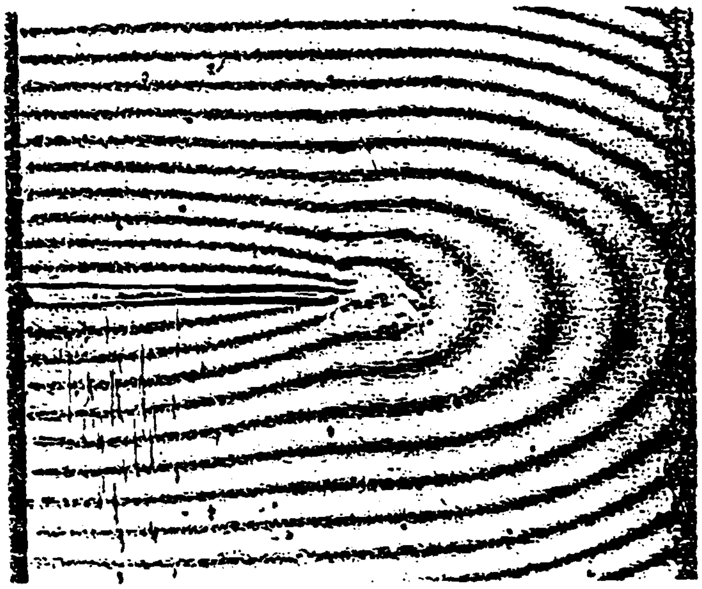
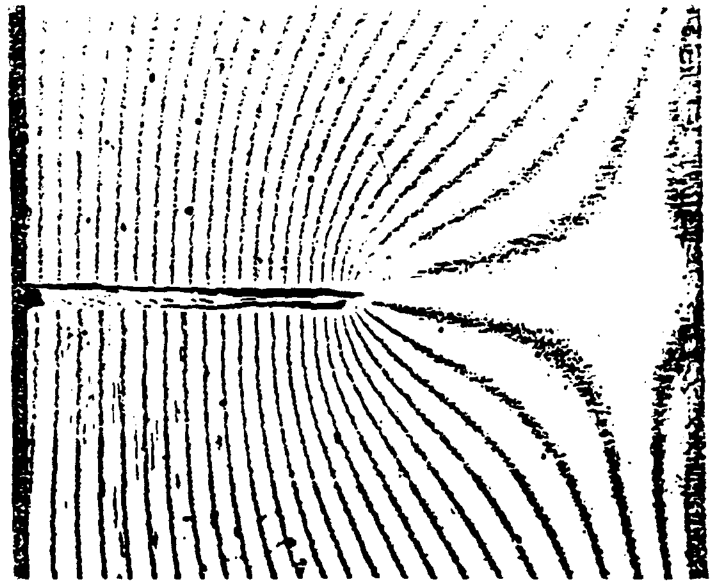


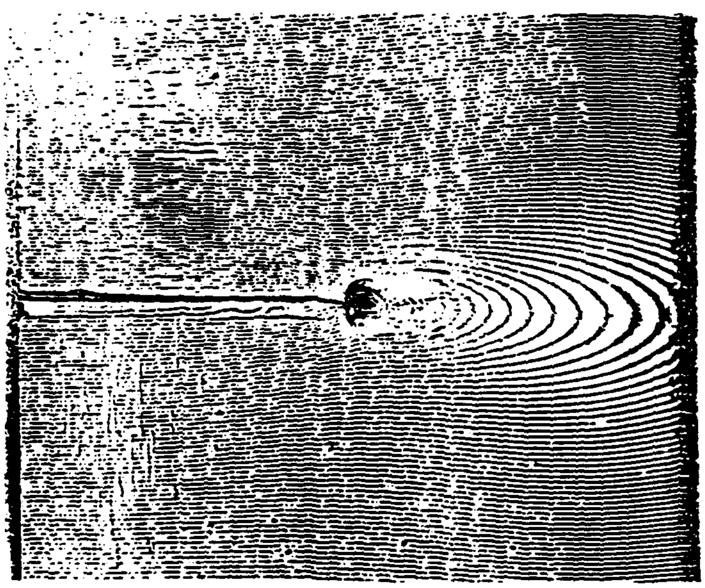
Fig. 3 Sensitivity Coefficients of Maximum Stress to Tip Twist Angle



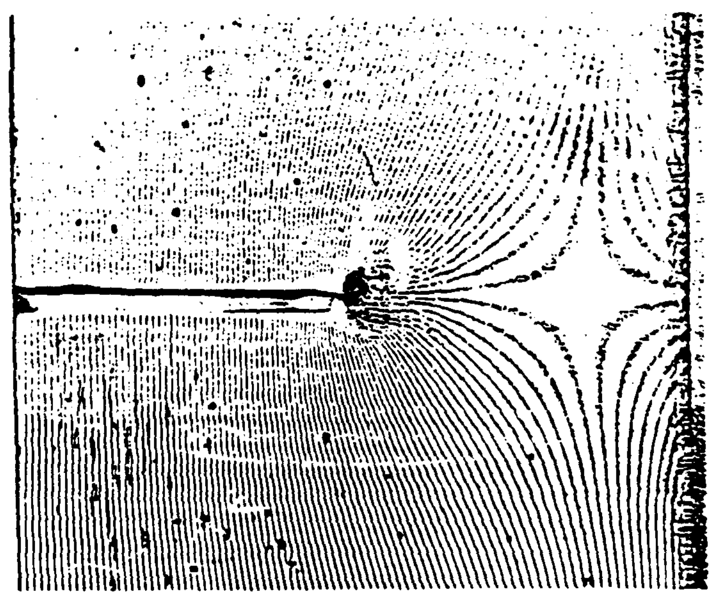
u-displacement
Applied load: 65 lb



v-displacement
Applied load: 65 lb



u-displacement
Applied load: 135 lb



v-displacement
Applied load: 135 lb

Fig. 4 Moire fringe patterns of a 5052-H32 aluminum SEN specimen under increasing load.

DICE Program

Task 3.4.5.1 XD™ Material — **Martin Marietta Laboratories**

Objectives

Material selection for a given application can proceed concurrently during component and manufacturing design, provided that the alloy's performance and the evolution of its microstructure following all processing options have been characterized. Discontinuously reinforced composites combine the improved performance benefits of metal matrix composites (MMC's) with the flexibility and convenience of conventional metal production and fabrication processes. This class of composites provides an interface between the monolithic, unreinforced matrices and the high-performance continuous-fiber composites currently under development.

In the present study, the synergism among processing, alloy microstructure, and properties has been examined through the evaluation of XD™ particulate- and short-fiber-reinforced gamma-based titanium aluminide composites. These materials have been investment cast, forged, and directionally solidified, yielding a wide range of microstructures from which materials with varying performance attributes have been produced. Material of each processing and/or microstructural state has been evaluated through the measurement of room temperature and 800°C tensile properties (yield and tensile strength, elongation, and modulus), strain-rate sensitivity, creep, high-cycle fatigue, short-rod fracture toughness, coefficient of thermal expansion, and ultrasonic modulus.

It is anticipated that this study will establish a data base for engineered discontinuously-reinforced titanium aluminide composites, and delineate the range of processing options available within the concurrent engineering environment.

Approach

Both matrix microstructure and reinforcement variants were produced for evaluation to explore the range of material properties attainable. The combinations explored include:

MATRIX CHARACTER	REINFORCEMENT	PROCESSING
Equiaxed	Particulate	Investment casting
Lamellar	Short Fiber	Wrought (forging)

Reinforcement shape is varied via XD^{TM} processing and by the proper selection of reinforcement chemistry. The alloys selected for this study are (nominal compositions, atomic percent):

Particulate	Ti-48Al-2V + 7.5 v% TiB_2
Short Fiber	Ti-48Al-2Nb + 7.5 v% NbB_2

The above two-phase, near-gamma titanium aluminides (based on the intermetallic α_2-Ti_3Al + γ-TiAl compounds) can exist in either a lamellar or equiaxed microstructural morphology, depending upon processing and/or heat treatment. As-cast ingots or investment castings exhibit an extremely stable lamellar morphology that remains essentially unchanged with subsequent heat treatment. Wrought processing (e.g. forging) of ingot produces an alloy that is of primarily equiaxed-grain morphology. The extent of the lamellar-to-equiaxed transformation during wrought processing can depend upon several processing variants, including working temperature, deformation strain, deformation strain rate, and the temperature and time of subsequent heat treatments.

Four 20 lb ingots of each composition were produced at ALTA Group (Fombell, PA). These were subsequently forged at Wyman-Gordon Company (Worcester, MA). The majority of the wrought characterization was performed on material forged from a 3 in. high billet into 0.5 in. pancakes. To investigate the effect of processing strain on the evolution of the equiaxed microstructure, variable-height preforms were utilized to produce a strain gradient in the forged product. In addition, segments of these pancake forgings were further reduced to 0.2 in. for property measurement. In each instance, material was produced by forging at both 1100°C and 1180°C.

One 250 lb ingot of each composition was produced at Timet Corporation (Henderson, NV) for subsequent investment casting of test bars at Howmet Corporation (Whitehall, MI).

Following delivery of both the wrought and cast products to Martin Marietta Laboratories, heat treatments were initiated to evolve the intended matrix microstructure and/or to optimize the properties. The heat treatments and rationale for each are as follows:

HEAT TREATMENT	RATIONALE
900°C / 16 h	Stabilizes microstructure through restoration of equilibrium proportions of α_2-Ti_3Al microconstituents.
1200°C/16 h/900°C/5 h	Converts wrought lamellar to an equi-axed matrix morphology, and restores equilibrium proportions of α_2 and γ microconstituents.
1400°C / 2 h	Restores the wrought mixed lamellar/ equiaxed microstructure to a stable, fully lamellar morphology.

Directional solidification techniques have also been used in an effort to produce in situ, aligned short-fiber metal matrix composites. The degree of alignment has been qualitatively assessed using metallographic techniques; mechanical properties (tensile and compression) were also measured.

Technical Results

All compositional and processing variants were successfully produced, with the exception of the investment-cast, short-fiber reinforced alloy which failed during during remelt via variable arc remelt (VAR) for investment casting into test bars. Subsequent analysis of the ingot revealed the presence of several large cracks, which were present in the initial ingot as-received from Timet. Hot isostatic pressing (HIP) prior to remelt was insufficient to heal these cracks. Two approaches were attempted to provide a substitute for this material. Twelve test bars were produced by Howmet Applied Research and Development group by remelting segments of the failed ingot in a laboratory induction skull remelt (ISR) unit, and in some instances HIPped ingot sections were used for mechanical property evaluation.

Figure 1 illustrates examples of the particulate- or short-fiber-reinforced microstructures obtained through processing and/or heat treatment. Reinforced lamellar structures are obtained through casting or by heat treatment at very high temperatures (e.g., 1400°C) of wrought materials. Equiaxed structures are obtained only with heat treatment of wrought products at temperatures in the approximate range 1130-1350°C.

Table I lists a portion of the properties attained for each alloy and processing variant studied. A full discussion of properties obtained in this study are beyond the scope of this summary; however, a detailed report containing all the data obtained in this program is available upon request.

Figure 2 shows an example of a short-fiber-reinforced gamma titanium aluminide that was processed via directional solidification, using techniques developed under the present program. As shown, a general alignment of high-aspect-ratio short-fibers (as high as 200:1) has evolved longitudinally, perpendicular to the solidification interface. Depending upon specific processing conditions, the mechanism of alignment is either by mechanical rearrangement of the existing fibers or through the in-situ solidification of new fibers oriented perpendicular to the solid/liquid interface. The degree of alignment, fiber length, and aspect ratio of the resulting reinforcement can now (to a limited extent) be varied with variations in melt superheat, solidification rate, and temperature gradient. Tensile and compression tests performed on these materials confirm the non-random alignment of the reinforcement, as manifested by an increased longitudinal strength combined with a reduction in transverse strength relative to the base (random) ingot, and an increase in the tensile/compression asymmetry in these materials.

In addition to fiber alignment, directional solidification techniques were used to demonstrate the production of "graded" particulate-reinforced composites. Particles can be swept along a solidification interface in a controlled manner or deposited at specific locations within an ingot by imposing variations in solidification rate.

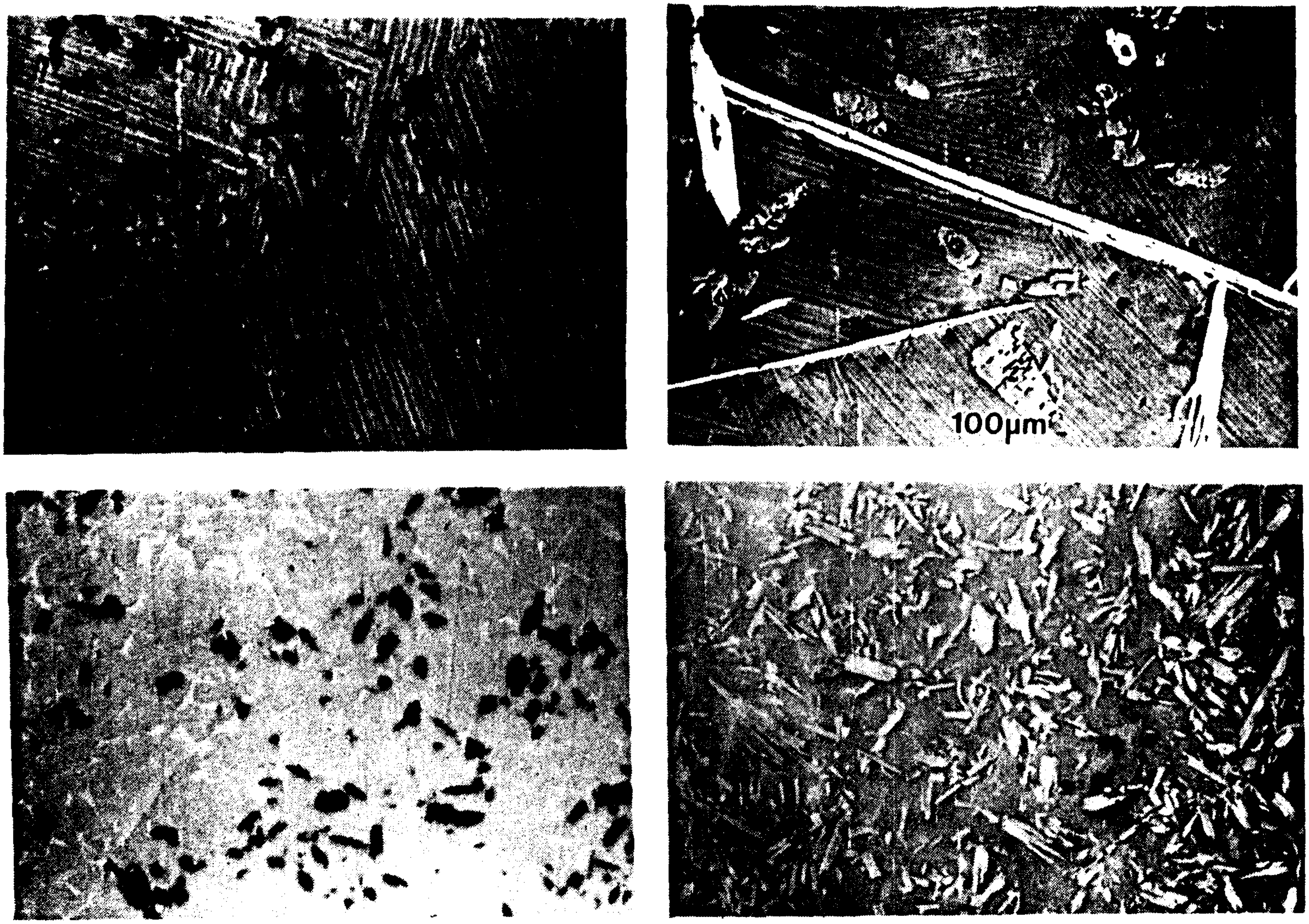


FIGURE 1. Examples of microstructure and reinforcement variants examined in the present study. a) TiB_2-particulate (black) in a lamellar, α_2 (light) + γ (gray) matrix, b) NbB_2-short-fiber (white) in a lamellar, $\alpha_2 + \gamma$ matrix, c) TiB_2-particulate in an equiaxed, $\alpha_2+\gamma$ matrix, and d) NbB_2 short fiber in an equiaxed, $\alpha_2+\gamma$ matrix. Backscattered electron images at 500X.

TABLE I. Summary of processing variants and example properties measured.

Product	Heat Treatment[1]	$\sigma_y^{25°C}$ (ksi)	$\sigma_u^{25°C}$ (ksi)	(% el)$^{25°C}$	$\sigma_y^{800°C}$ (ksi)	K_{Ic} (ksi√in)	Creep[2] (h)	HCF[3] (N_c)	Modulus[4] (Msi)
Wrought Particulate	#1	102.4	105.0	0.50	48.6	13.1	4.4		26.5
(Forged at 1100°C)	#2	79.9	90.4	0.64	49.4	9.5	11.6	10.6×10^6+	27.0
	#3	78.8	90.4	0.31	53.4	13.1	83.0	6.4×10^6+	26.4
Wrought Particulate	#1	103.1	108.7	0.74	55.6	TBR	4.6		26.2
(Forged at 1180°C)	#2	80.1	89.7	1.0	53.0	12.1	9.2	3.5×10^6	26.3
	#3	73.4	86.4	0.70	48.9	13.6	16.0	9.6×10^6	26.5
Wrought Short-fiber	#1	---	113.9	0.0	67.8	9.6	6.4		26.2
(Forged at 1180°C)	#2	---	80.6	0.0	65.6	10.1	19.5	7.3×10^6	27.3
	#3	94.1	96.2	0.42	65.5	15.1	121.5	10.0×10^6+	26.3
Cast Particulate	#1	74.0	90.4	0.71	58.4	17.1	TBR		TBR
	#2	TBR	TBR	TBR	58.4	15.4	TBR	TBR	TBR
	#3	74.8	98.4	0.65	53.6	16.3	TBR	TBR	TBR
Cast Short-fiber	#1	TBR	TBR	TBR	76.6	TBR	TBR		TBR
	#2	TBR	TBR	TBR	75.8	TBR	TBR		TBR
	#3	TBR	TBR	TBR	86.0	TBR	TBR		TBR

Explanation to notes: TBR = To Be Reported

All tensile testing conducted at strain-rate = $3 \times 10^{-4}\ s^{-1}$

1 Heat treatments: #1 = 900°C/16h; #2 = 1200°C/16h/900°C/5h; #3 = 1400°C/2h

2 Creep: time to 0.5% strain at 800°C, σ_o = 20 ksi

3 HCF: Number of cycles at 800°C, σ_{max} = 40 ksi, R = 0.1; '+' indicates sample removed prior to fracture.

4 Modulus: Determined by ultrasonic techniques

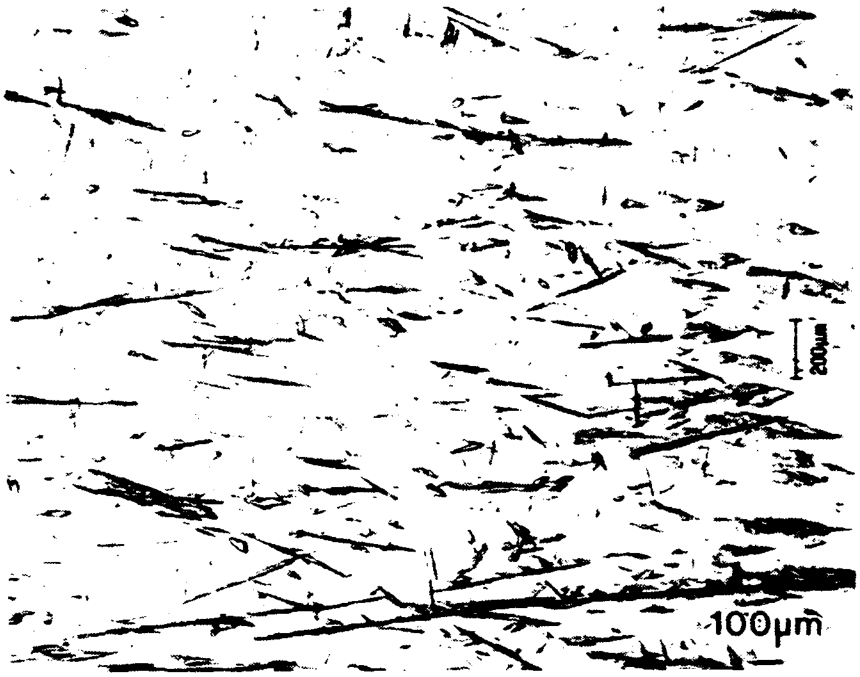


FIGURE 2. Ti-40Al + 4.0 v% NbB_2 ingot which was directionally solidified to produce a general alignment of short-fiber reinforcement.

Conclusions

The following general conclusions were reached regarding the microstructural, reinforcement, and processing variants explored as part of the present study:

- PARTICULATE REINFORCEMENT leads to maximum room temperature ductility and excellent processability; good fracture toughness, strength, and high-cycle fatigue.

- SHORT-FIBER REINFORCEMENT leads to maximum strength, both at room and elevated temperatures; good creep resistance, fracture toughness, modulus, and high-cycle fatigue. However, room temperature ductility is extremely low and successfully processing this material is challenging.

- EQUIAXED MATRICES are required for maximum room temperature ductility. However, fracture toughness is poor. Creep resistance is moderate, but could be improved with anneals designed to promote grain growth.

- LAMELLAR MATRICES give maximum fracture toughness; however, ductility is low. If microstructurally stable, good creep resistance and high-temperature strength are observed. Unstable lamellar matrices lead to maximum creep rates.

- WROUGHT PROCESSING provides maximum microstructural flexibility; suitable post-working heat treatments can produce either equiaxed or lamellar microstructures. Short-fiber reinforcements tend to fracture during wrought processing, the extent of which depends on processing temperature and strain.

- INVESTMENT CASTING produces a stable, high-fracture-toughness, near net-shape material without the need for subsequent, complex heat treatments. Minor improvements in properties are attainable with heat treatment.

- DIRECTIONAL SOLIDIFICATION of XD™ short-fiber-reinforced ingot results in partial alignment of the boride fibers. As predicted, strength is maximum in the direction of alignment, and reduced in the transverse direction. Graded structures can be produced by varying particulate loading through the cross-section of the directionally solidified ingot. The ability to engineer and produce anisotropic structures, such as those produced via directional solidification in the present study, is a powerful tool for designing and tailoring materials for a specific application in a concurrent engineering environment.

Recommendations

Isotropic, discontinuously reinforced composites are one of several material options available to the materials and design engineer. The information obtained in the present study gains significance if evaluated in parallel with data on other material classes, e.g., unreinforced and continuous-reinforced alloys.

In the present study, only two alloy compositions were selected in an effort to model the relative performance of two morphologies of discontinuous reinforcement (particulate and short fiber) over a range of processing options. Very likely, there are further opportunities to optimize material performance by extending the present results to advanced titanium-aluminide matrices and to broader ranges of reinforcement volume fraction than those studied here.

The ability to produce engineered anisotropic structures creates an ideal opportunity for concurrent engineering design strategies. Further development of processing schemes to produce such structures, such as those resulting from the directional solidification effort reported here, will provide an important and advanced capability in present and future materials technology.

DICE PROGRAM

Task 3.4.5.2 RSPD Material Properties GEAE

Objectives:

A materials properties data base for titanium aluminide composite materials that use the RSPD process is being generated for use in the microscopic/micromechanical models and design rules being developed under this contract. In particular, temperature dependent properties are being examined in order to study the feasibility of using these materials in components of aircraft engines.

Approach:

Phase I of this task covered machining and testing of mechanical test specimens cut from Ti-14-21/SCS-6 panels supplied by CR&DC according to the test matrix outlined in the statement of work. Test specimens were evaluated to ascertain failure modes, and results are analyzed and reported.

Four 4-ply Ti-14Al-21Nb/SCS-6 composite panels (designated A,B,C, and D) were fabricated by GE CR&DC. The drums were wound to 112 filaments per inch. This material was plasma sprayed to result in high beta content matrix monotapes. Fibers were extracted from the plasma sprayed foils in order to measure fiber tensile strengths The panels were laid up with four unidirectional composite foils and three neat foils, one in the middle and one on each side. This procedure resulted in panels with relatively low volume fractions (19-21%). The panels were HIPed at 1830°F (1000°C) for 3 hours at 15 ksi (103 MPa). The panels were inspected by c-scan and FPI, and machined into specimens for tensile, impact, fatigue, shear and crack growth tests. Two blanks from each panel were returned to CR&D to determine the strength of extracted fibers after HIPing and to tensile test. Interstitial contents of the panels were also determined. A fifth composite panel, 6"x15" in size, was also received from CRDC and has been inspected and submitted for specimen machining.

Technical Results:

Initial results from the testing program showed somewhat lower longitudinal material properties in comparison with rule of mixtures expectations, although the transverse tensile strength of the panels was quite good in comparison with higher fiber volume fraction composites. Results from tensile, fatigue, and impact tests are summarized in Table 3.4.5.2-1.

Figure 3.4.5.1 shows a specimen edge montage starting from the tensile failure of longitudinal specimen B-4. In this micrograph, it is possible to see that there were several cracks in the specimen in addition to the main tensile failure. The inhomogenity of the microstructure is also evident, as well as a white-etching surface layer which is thought to be caused by molybdenum diffusion into the matrix during HIPing.

Cracks were observed in specimen fatigue specimen A-2 after 121559 cycles at a maximum stress of 75 ksi. A replica taken at 5016 cycles from specimen A-6 (cycled at a maximum stress of 85 ksi) revealed no cracks, even in SEM. At a maximum stress of 95 ksi, specimen A-7 was heard to crack during the initial application of load, and failed after only 616 cycles.

Figure 3.4.5.2 shows the front and back surfaces of specimen D-7, which had been impacted at room temperature with an energy of 0.54 ft-lbs. A surface crack is visible on the back side of the specimen. Specimen B-2, impacted at a higher energy (1.08 ft-lbs) at 1200^{o}F, had no visible cracks on the back side (Figure 3.4.5.3).

Crack growth and Iosipescu shear tests were also conducted on this material, with marginal success. The two Iosipescu shear specimens did not fail in the gage section, so the test results are invalid. Two remaining Iosipescu shear specimens will be modified by thickening the ends with tabs. This should provide better alignment and a successful test. A room temperature potential drop crack growth test conducted at 50 ksi nominal stress displayed secondary cracks in addition to the cracks emanating from the precrack.

Interstitial analysis of the panels was done at CR&DC. The analyses revealed high nitrogen interstitial content, as well as somewhat high oxygen content. Since the four panels were made from the same plasma spray run (RF1171), all four panels were affected. The larger panel was made from plasma spray run RF1174. It is now believed that there was a leak in the plasma spray system that accounted for the high interstitial content of these materials. The high interstitial content may well have degraded composite strength by embrittling the matrix, but the relatively good transverse strength obtained is surprising if this occurred.

Further analysis by x-ray diffraction revealed that the "contaminated" layer on specimens from panel B was beta phase. Two pieces from panel B were examined: one piece was taken from a specimen which had been tested at room temperature, and the second piece was taken from a specimen which had been exposed to a temperature of 1200°F for a short period of time (approximately 10 minutes) during an unsuccessful crack growth test. Interestingly, the piece that had been exposed to the higher testing temperature had beta peaks which were not so strong; additional peaks were beginning to form in this alloy but it is not clear what phase they constituted. Neither piece showed elemental molybdenum. However, molybdenum was detected in the beta phase using energy dispersive spectroscopy (EDS). The next step in the x-ray diffraction work will be to examine the matrix structure of panel B by grinding off the surface of the specimen.

In view of the possibility of surface contamination (the white-etching surface layer), the test plan was modified to investigate the possibility of improving strength by surface removal. Two tensile tests (A-5 and B-6) were conducted at room temperature in which the specimen surface was replicated after each 5-10 ksi increment in stress. This procedure was discontinued when cracks were observed on the specimens in an optical microscope. Specimens with two different surface finishes were examined. The surface of specimen A-5 had been diamond ground until approximately 0.003 inches of material had been removed. After grinding, the specimen was carefully examined in an optical microscope to determine if the fibers had been exposed. No evidence of this was seen. Specimen B-6 had an as-received finish. Cracks were observed in the optical microscope after a stress of about 130 ksi in the specimen with the diamond ground surface, and after about 75 ksi in the as received surface specimen. The replicas taken from these specimens will also be examined by SEM. Specimen A-5 was tested for its ultimate tensile strength after replication, which was found to be 157 ksi. This was about 10 % higher than specimen A-1, which was from the same panel, only had not had the surface grinding treatment. However, because some of the matrix was removed, specimen A-5 also had a slightly higher volume fraction of fibers.

Also, two specimens from panel D were loaded in tension to 100 and 130 ksi, respectively, for metallographic examination. These specimens will be cut up to determine the mode of tensile fracture. The same procedure will be used on tensile specimens from the large panel, only with variations of surface finish in order to see how the surface finish is affecting the failure.

Conclusions:

Mechanical testing of materials received to date has revealed relatively low longitudinal strengths but good transverse properties. High interstitial content is the suspected reason for the low longitudinal properties, but the good transverse properties are surprising in light of this.

Metallography reveals a contaminated layer on the surface of these panels which has been identified as beta phase. This phase appears to cause cracks to form in tensile tests at lower stress amplitudes than they do in specimens in which the layer has been removed. Methods for removing this layer (diamond grinding, chem milling, etc.) are being investigated.

Recommendations:

The source(s) of interstitial contamination in the plasma spray sytem must be found and eliminated. The influence of processing conditions (especially consolidation) on alpha-2 composites must be determined so that parameters with improved properties and consistency can be defined.

A study of the effects of surface condition on tensile properties of these materials should be conducted so that such effects can be eliminated and the intrinsic composite strength attained. Metallographic characterization should be performed earlier in the process to identify the extent of any surface contamination.

Alternate alpha-2 matrix alloys should be investigated to improve composite properties by acheiving higher matrix strength, ductility, or tolerance to interstitial contamination.

SPECIMEN NUMBER	TEMPERATURE ^{o}F (^{o}C)	TEST ORIENTATION	UTS KSI (MPa)	E MSI (GPa)	e_f (%)	VOLUME FRACTION	FAILURE LOCATION
A-1	70 (21)	0[4]	140.9 (971)	23.8 (164)	(2)	0.22	gage
A-5	70 (21)	0[4]	157.0 (1082)	20.4 (141)	(1,3)	0.23	gage
B-4	70 (21)	0[4]	138.4 (954)	22.8 (157)	0.73	0.21	gage
B-6	70 (21)	0[4]	(1,4)	22.1 (154)	(1,4)		(1,4)
A-3	600 (316)	0[4]	174.3 (1200)	27.7 (191)	(2)	0.21	gage
D-6	1400 (760)	0[4]	104.6 (721)	13.8 (95)	0.95	0.19	gage
D-9	70 (21)	90[4]	46.8 (323)	15.9 (110)	0.40	0.21	radius
C-7	1400 (760)	90[4]	31.0 (214)	7.6 (52)	1.98	0.21	gage

(1) Specimen was incrementally loaded and replicas taken every 5-10 ksi in order to observe surface cracking before maximum load.
(2) Could not detect
(3) Specimen had diamond ground surface.
(4) Specimen was not tested to failure.

Table 3.4.5.2-1a Tensile Properties of High Beta Content Ti-14-21/SCS-6 Unidirectional Composite HIPed at 1830^{o}F (1000^{o}C) for 3 hours at 15 ksi (103 MPa)

SPECIMEN NUMBER	TEMPERATURE ^oF (^oC)	TEST ORIENTATION	STRESS RANGE KSI (MPa)	CYCLES TO FAILURE	VOLUME FRACTION	FAILURE LOCATION
A-2	70 (21)	0[4]	71.3 (492)	166713		(1)
A-6	70 (21)	0[4]	80.8 (557)	113000	0.21	(1)
A-7	70 (21)	0[4]	90.3 (623)	616	0.20	gage
D-1	1400 (760)	0[4]	61.8 (426)	5919	0.21	gage
D-3	1400 (760)	0[4]	52.3 (361)	30928	0.22	radius

(1) Run-out

Table 3.4.5.2-1b Fatigue Properties of High Beta Content Ti-14-21/SCS-6 Unidirectional Composite HIPed at 1830^oF (1000^oC) for 3 hours at 15 ksi (103 MPa)

SPECIMEN NUMBER	IMPACT TEMPERATURE ^{o}F (^{o}C)	SPECIMEN WIDTH inches	ENERGY ft-lbs	External visual appearance
B-2	1200 (649)	0.400	1.08	Large indent, no surface cracks
B-5	1200 (649)	0.399	0.502	Slight indent
D-7	70 (21)	0.399	0.54	Crack on back of specimen
C-1	1200 (649)	0.900	0.502	Slight indent
C-2	1200 (649)	0.899	1.07	Large indent

Table 3.4.5.2-1c Ballistic Impact Test Results for High Beta Content Ti-14-21/SCS-6 Unidirectional Composite HIPed at 1830^{o}F (1000^{o}C) for 3 hours at 15 ksi (103 MPa)

Figure 3.4.5.2-1 Etched micrograph (50x) of tensile specimen B-4 showing contaminated surface layer, inhomogeneous microstructure and secondary cracking.

Figure 3.4.5.2-2 Front (top) and back (bottom) views (10x) of specimen D-7, which had been impacted at room temperature with an energy of 0.54 ft-lbs. Front surface shows an indentation and back surface shows a large lor_'tudinal crack.

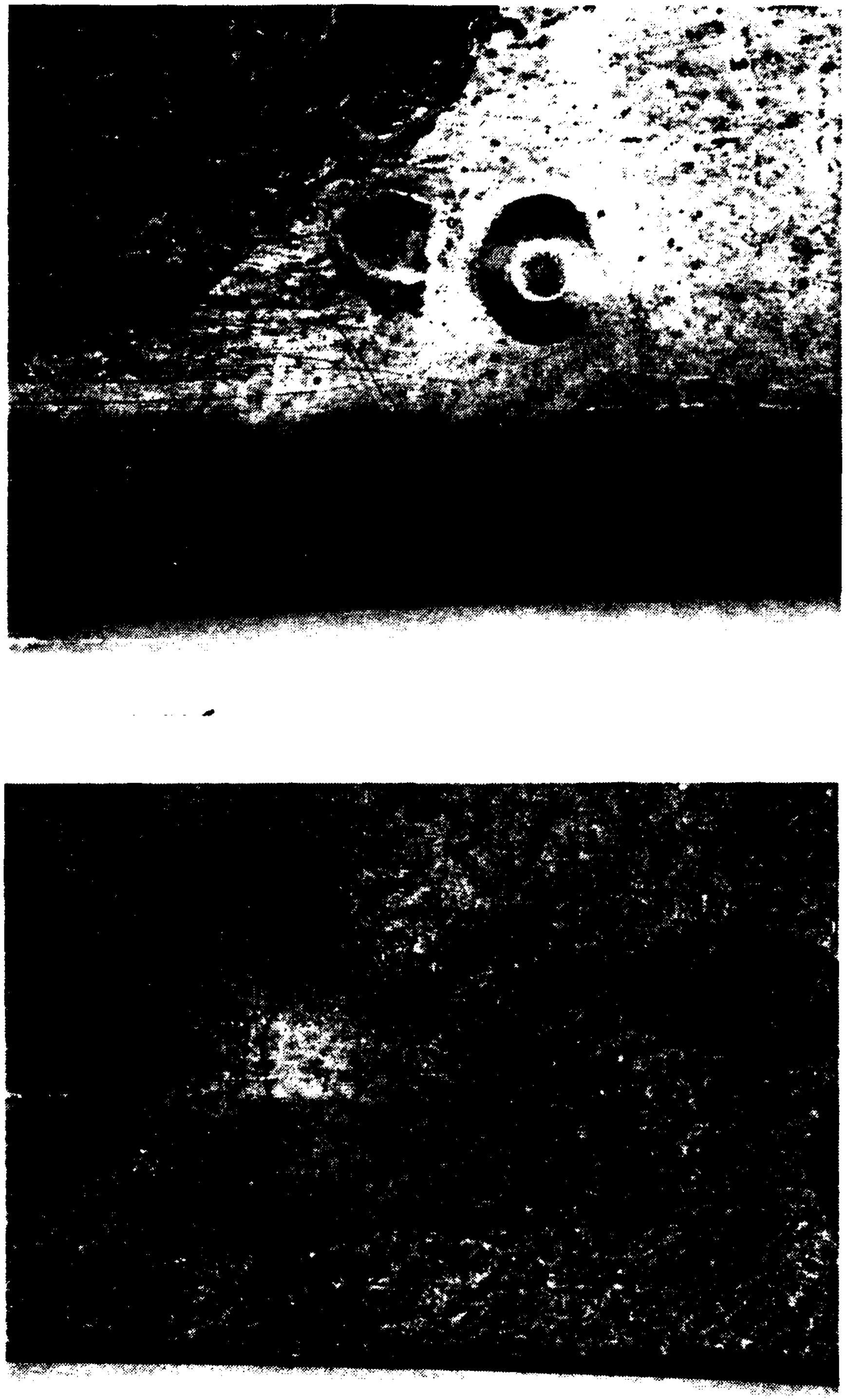


Figure 3.4.5.2-3 Front (top) and back (bottom) views (10x) of specimen B-2, which had been impacted at 1200°F with an energy of 1.08 ft-lbs. Front surface shows an indentation, but back surface shows no evidence of cracking.

DICE Programs

Task 3.4.5.3 Materials Properties and Reasoning **Carnegie Mellon University**

Objectives:

The objective of this work is to provide the mechanical designer with early feedback about the stresses present in a given blade-geometry and the implications for material selection.

Approach:

The approach used during the first year of this work has been to develop a simplified structural perspective that can be integrated into the concurrent engineering Design Fusion work station and to develop the concept of life-maps.

Simplified structural models are required because you want to be able to investigate interactions and trade-offs since it is necessary to be able to consider a large number of alternatives quickly. Consequently, it is important to have computationally efficient algorithms for calculating the structural response of different designs. Simplified structural models based on analytical solutions provide an efficient means for computing structural response. In addition, the simplified models are useful because they often identify the higher level features of the design that are most influential in actually controlling the response of the part.

The life-map concept also provides a means for integrating viewpoints from different perspectives. For example, the low cycle fatigue (LCF) life of a material can be calculated as a function of its temperature, and stress history. Since the temperature and stress history vary from point to point in a part, the fatigue life gives thought of as varying. A life-map of the LCF life of a part would be contour plots of constant life. The life-map concept is important because it provides the relevant information in a form which is easy to assimilate and also integrates the thermal, structural and materials perspectives. By comparing the life-map with temperature and stress contour maps the design engineer can decide on whether the fatigue life is mostly influenced by high stresses or by a hot spot or, perhaps, by a more complex interaction of the stress and temperature fields. The resulting information will guide the designer in determining how best to remedy the problem.

Technical Results:

Simplified models of fan blade steady stress and fatigue where developed using beam theory. Computer subroutines for calculating the steady stress have been developed and integrated with other perspectives in the Design Fusion work station. The steady stress model includes the following loading mechanisms: centrifugal pulling force, gas bending loads, and bending moments induced by blade tilt. The blade geometry is given in terms of a series of blade sections which describe an airfoil, platform, and neck. The sections are defined by their radial location and a set of coordinates of the boundary points. Section properties (area, moments of inertia, the location and orientation of the principal axes -- which can vary due to blade twist) are calculated from the boundary point description. Stresses are calculated using a beam theory which takes blade twist into account when calculating the gas bending moments. Stress concentration effects due to sharp blend radii have also been simulated. The steady stresses are used to calculate fatigue life at the boundary points using information from the materials perspective The stress and life information is then passed to the blackboard where it may be viewed by the designer as stress contour plots and life-maps.

In addition, the structural algorithm can participate in a respond to queries. For example, how does the fatigue life change if the blade length varies over a certain range of values? The results are actually quite complex since the cross section of the airfoil must also change if this is to meet aerodynamic constraints. The results of life versus blade length, in effect, integrates the aerodynamic, structural, and materials perspectives in providing its final answer.

Conclusions:

1. Simplified structural and fatigue models have been developed to represent the structural and materials perspectives in the Design Fusion work station. Because of their computational efficiency they more readily lend themselves to design optimization trade-off studies involving multiple perspectives.
2. Simplified structural models provide an understanding of which features are most important in controlling the response of the system.
3. Life-maps provide an integrated thermal, structural, and materials view of the design.

Recommendations:

1. Extend the structural models to include creep and vibration.
2. Develop simplified structural models based on a finite element approach which have a wider range of application.
3. Attempt to find approaches for simultaneously considering multiple failure mechanisms, such as creep and fatigue (both LCF and HCF) to permit optimal material selection.

Publications

None.

Hardware

No hardware or software was purchased for this task. The software to produce stress and life maps was developed.

DICE PROGRAM

Task 3.4.5.4 Materials Characterization West Virginia University

A. **Objectives:**

The overall task objective is to develop tools for the design of advanced materials for a wide variety of applications, using a coordinated effort of experimental results and modern theoretical calculations. The initial approach was to select a TiAl alloy system which had the desirable characteristics of high strength, high melting temperature and low density and the undesirable characteristic of brittleness. The problem was to investigate the possibility of modifying the alloy in such a way as to preserve the good characteristics and decrease the brittleness (i.e. make alloy more ductile).

B. **Approach:**

The initial task was to direct attention to the TiAl alloy system. Therefore, the initial theoretical studies were directed toward understanding the crystallographic stability of the y - TiAl system and understanding its elastic behavior. The crystallographic stability of a material gives a good indication of the validity of the bonding assumptions made in the modeling process. It is also the first step in calculating the elastic constants which characterize the plastic deformation behavior of a material. This knowledge indicates the validity of the initial calculations and gives confidence to advance the calculations to consider the effects of small additions of first transition series elements to y - TiAl on the stability and the elastic constants.

The experimental approach was to prepare a number of Ti_xAl_{1-x} (x = 0.40 to 0.65) alloys in the a-yfield of the TiAl phase diagram by arc-melting under an Argon atmosphere. A variety of physical properties (thermal expansion, x-ray diffraction, hardness, specific heat and metallographic characterization) were then determined. This phase of the experimental work served as a baseline to compare the alloys having additives. Subsequently, alloys with small additives (2 to 5%) of manganese, chromium and vanadium were prepared and their properties determined. Finally, the calculated results were compared with the experimental results. It was expected that the comparison would assist in improving theoretical modeling and result in the formulation of some empirical rules for the design of materials for specific applications.

C. Results:

1. The major theoretical results are listed below:
   - For pure y- TiAl, the calculated lattice parameters are within 1% of experiment for the c-value and within 3% of the experimental values for the a-value confirming the inherently high quality results that can be obtained using the methodology;
   - The calculated bulk modulus of y- TiAl is 1.4 Mbar and Young's Modulus is 1.94 Mbar to be compared with the experimental value of 1.74 Mbar for the Young's Modulus;
   - The results of the *calculations*, which model the addition of 2% Mn in y- TiAl indicate a 2% or less change in the lattice parameters and a decrease of 10% to 20% of the bulk modulus, indicating an improvement in ductility. These low concentration calculations; however, had to be done using a crude "pseudo-atom" model, and
   - Finally, the more accurate supercell calculations completed for 25% concentrations show an energy advantage in the selection of Ti sites for V, Cr and Mn additives but with widely differing energies ranging from 6500 K for V to 1900 K for Mn. This is consistent with metallurgical evidence that Mn is distributed between both the Ti and Al sites. This also indicates possible high-temperature instability of the Mn alloys at high concentration.

2. On the experimental side of this project, the following accomplishments have resulted:
   - Samples of Ti_xAl_{1-x} (x = 0.40, .45, .50, .55, .60, .65) have been prepared by arc-melting followed by an anneal at 1000^o C in a purified Argon atmosphere;

- Metallographic and structural characteristics analysis of these samples have been made and shown to be in conformity with existing phase diagrams;
- The high temperature thermal expansion measurements show a gradual increase of the thermal expansion coefficient a from room temperature to 1000° C for Ti_xAl_{1-x} (x = 0.50, .55, .60, .65) which indicates a softening with increase in temperature. For (x = 0.40 and .45) the thermal expansion is constant across the temperature range. The $Ti_{50}AL_{50}$ alloy has the highest a;
- For $Ti_{45-x}Mn_xAl_{55}$, a decreases as Mn concentration increases while $Ti_{50-x}Mn_xAl_{50}$, aloy increases with Mn concentration;
- Additions of Mn for $Ti_{50-x}Mn_xAl_{50}$ (x = 1 and 2) alloys caused an increase in hardness while for $Ti_{45-x}Mn_xAl_{55}$ (x = 1) resulted in a slight decrease in hardness, and
- Specific heat measurements indicate possible phase transformations at elevated temperatures.

D. Conclusions:

- Modeling calculations of the lattice parameters were performed. Young's Modulus and the Bulk Modulus of y - TiAl produced values close to the experimental values. This result indicates that good modeling assumptions have been used;
- Modeling calculations, incorporating additives of Mn, Cr and V in y - TiAl indicates only slight changes in lattice parameters but larger changes in Bulk Modulus;
- Calculations show that V, Cr and Mn have a site preference for Ti sites although Mn could occupy either Ti or Al sites, and
- Experimental hardness determinations indicate an increase in hardness with an increase in Mn for $Ti_{50-x}Mn_xAl_{50}$ and a slight decrease in hardness for $Ti_{45-x}Mn_xAl_{55}$. This appears to conform with the same variation in thermal expansion.

E. Recommendations:

- Continue modeling calculations of the lattice stability and physical moduli of y- TiAL and additives to y- TiAl using large super-cell calculations treating low additive concentrations (necessitating supercomputer access);

- Continue modeling calculations of the site preference of additives in the y- TiAl alloy using large super-cell calculations to treat low concentration of additives (requiring supercomputer access). These results need to be used to judge high-temperature stability, especially for Mn as an additive;
- Experimentally investigate the effect of additions of Mn, V, Cr. . . to the TiAL system on the thermal expansion and elastic properties of these alloys at high temperatures;
- Determine the site location of Mn, V, Cr in the TiAl system using anomalous x-ray scattering for comparison with modeling calculations;
- Prepare multi-layer composites of the TiAl system by sputtering techniques and determine whether their thermoelastic properties are superior to the arc-melted or cast alloys;
- Measure the electronic properties of the TiAl system by SQUID magnetometry and correlate these findings with thermoelastic measurements, and
- Collect the above information to develop empirical rules for preparing the TiAl system with the best thermo-elastic properties for high temperature application.

F. **Publications:**

Submitted paper to Physical Review Letters entitled, "Predication of Site Selection for Additives to Intermetallic Compounds", P. K. Khowash, D. L. Price, and B. R. Cooper.

G **Hardware:**

See Phase I Final Report: Task 3.2.1.3

1 Design for Testability

This section discusses the prototype Design for Testability (DFT) Application developed by Cimflex Teknowledge. DFT is the first of a series of applications for the High Density Electronics (HDE) Workstation under development for the Darpa Initiative on Concurrent Engineering (DICE).

2 Problem

Test development for digital systems is becoming an increasingly difficult and expensive task. Difficulties in test development significantly delay initial production and adversely affect maintainability and logistics. A test consists of a series of 1's and 0's to be applied to input pins and a measurement of the response at output pins. Any difference between the signals at the output pins and the expected response indicate a failure and a defective circuit. A test set can easily consist of thousands or tens of thousands of such test vectors. Test development has the following problems:

- developing a comprehensive test set that can detect all physical faults is a time-consuming process.
- determining how good the test set is (test grading) can only be done after detailed gate-level design is completed.
- fault simulation, the process of determining the outcome of applying tests under varying physical fault assumptions, requires considerable computing resources, and
- design changes to improve fault detection, at such a late stage of development, are very expensive.

These problems arise because testability concerns typically are not addressed early in the design phase. Current testability analysis occurs late in the design process, i.e. at the logic level. If any testability problems are identified at this point, it is usually too expensive to do a redesign to improve testability. Typically, some compromises are made which result in the fielding of an inadequately tested system.

3 Objectives

The main objective is to support design for testability concurrently with design for functionality and performance. In order to do that and provide useful advice early in the design phase, we need to develop:

- design representations to support testability analysis concurrently with design for functionality,
- techniques which allow early identification of testability problems,
- measures which quantify and characterize testability problems early in the design phase,
- knowledge bases of techniques to improve testability of incomplete designs, and
- techniques to intelligently trade-off testability against other considerations.

4 Approach

Our approach to achieving the objectives listed above mirrors the methodology of an experienced designer. An experienced designer has a top down and bottom up approach to design. The design usually starts in a top-down fashion, with the design progressing at different levels of abstraction. At various times, the designer typically identifies physical components (islands of solutions) reflecting a bottom-up approach to design.

In order to support this hybrid top-down-and-bottom-up approach, the design representation should allow hierarchical decomposition. We are basing our representation on VHDL's hierarchical structuring. This representation supports different levels of signal abstractions as well as module abstractions. Our representation also integrates test information into the design representation.

To support the bottom-up approach, we are developing a library of prototypical circuits. These circuits are designs which serve as exemplars for a class of designs which provide a specific functionality. The testability of these circuits would be analyzed prior to putting them in the library. When the designer uses these designs (as islands of solutions), the prior analysis will provide meaningful, quantitative analysis of testability. At one extreme, new designs may be based completely on compositions of existing designs, which exploits the library and its associated information to the maximum possible extent.

More likely than not, new designs will combine old designs and new arrangements. The question we are addressing here is how to leverage knowledge about testability in general to help the designer in the early stages working in a top down fashion. Knowledge about testability is usually represented at very general levels, such as "Memory elements must be initializable before testing can be done," or by examples, as in, "Make Clear/Reset a primary input or controllable by primary input." It is a difficult task to represent testability know'edge of this type in a form usable during high level design.

Our approach relies on developing a knowledge representation to capture the intent behind rules shown above, the "deep knowledge" of testing. We relate functions of components with test plans that detail how that function can be tested. Note that a component can perform many functions, a function can be tested in several ways, the same function can be associated with different components, and a given test can be associated with the testing of several different functions. Therefore, the relationships among components, functions and test plans result in a tangled hierarchy rather than a tree, thereby reducing storage and search requirements compared to a tree or a flat rule base. Also, by representing these in terms familiar to both design and test engineers, communication between them will be enhanced. In particular, test knowledge can be maintained by test engineers and made available to design engineers trying to analyze new designs for testability problems.

5 Technical Results

Our Phase I application demonstrates two principal results: (1) Test Coverage Estimators for designs composed of functional components modeled above the gate level; and (2) design for testability guidelines for functional block diagrams. These results required the implementation of several basic technologies. First, we developed a hierarchical design representation integrating design functionality and testability data, supported by a graphical design editor. Second, we developed a demonstration library of components. Test sets and fault dictionaries for the library components were computed using the Lasar fault simulation program, accessed via an application wrapper. We developed algorithms for propagating test patterns and estimating fault coverage for the library components from this pre-analyzed data. Third, we developed preliminary knowledge bases of component functions and functional tests. We implemented a prototype functional test advisor for declaring and tracking functional test requirements for designs expressed as functional block diagrams.

The following sections describes these developments in more detail.

5.1 Design Representation and Design Editor

As expressed previously, our design representation is based on VHDL's hierarchical representation. For this phase, we use only a structural description, deferring till a later date timing and behavioral information. For each library component, a VHDL-type representation is maintained at the class level. This acts as a generic template, describing how the component is put together from other existing library components. Whenever the user wants to put the component into a design, a unique instance is created for that purpose.

Figure 1 shows a class-level VHDL description of a full adder together with one instantiation.

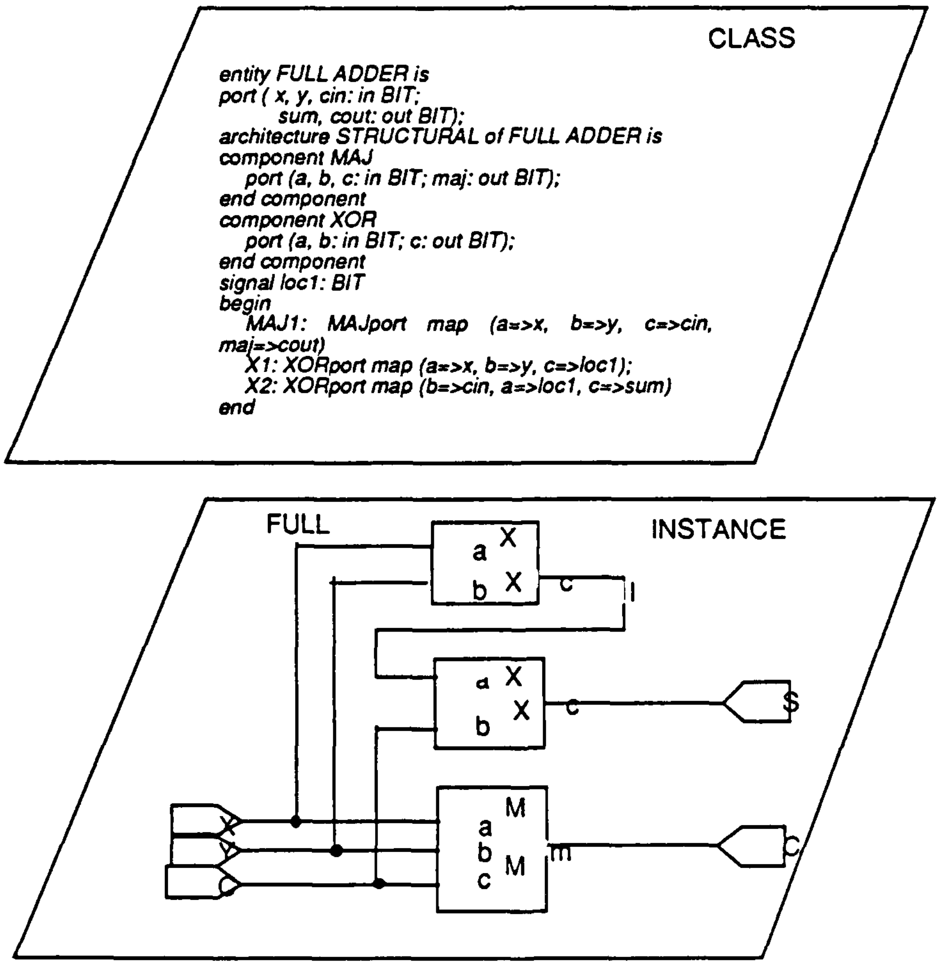
The class-level description says that the full adder is composed of MAJ and XOR components and specifies the interconnections between the subcomponents. All of the necessary information to create the instance from the class is shown here, except for the graphical information. VHDL is useful to us in several other ways:

- It separates a module's functional specification interface from its implementation.
- The port information is highly structured, allowing us to define ports at different levels of abstraction.

To handle interfaces at different levels of abstraction, we have adopted the notion of a translator. Translators are pseudo-modules which take input signals at one level of abstraction and map them to signals at a different level of abstraction.

A graphic interface consisting of icons, menus, and dialog boxes is provided to create a new design, modify an evolving design and save a design.

As the user incrementally builds up a design, the design information is translated to a VHDL-type description incrementally.



5.2 Test Coverage Estimation

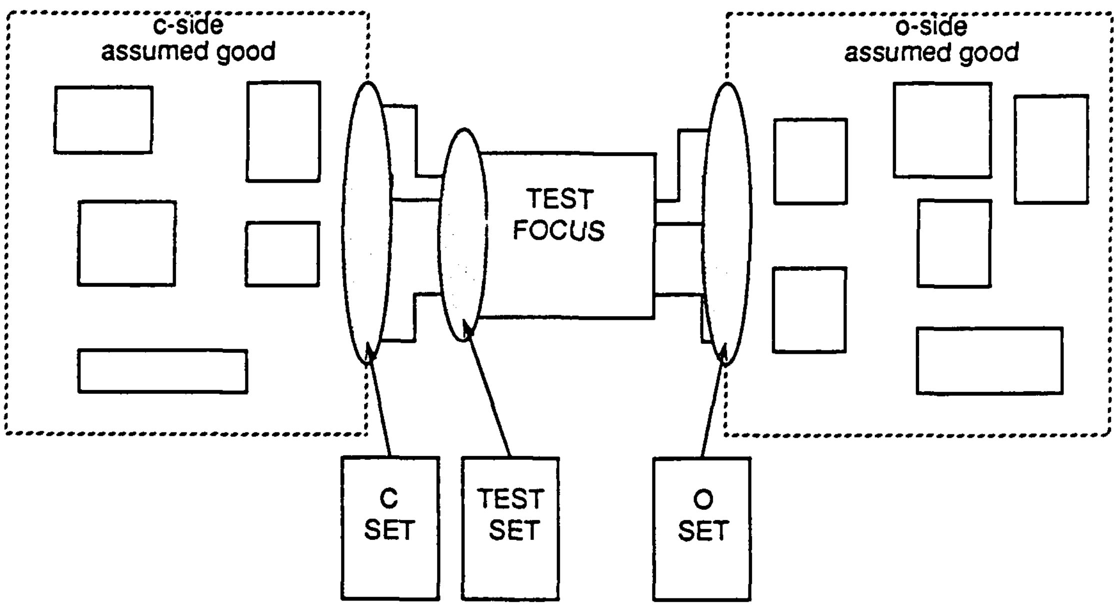
A straightforward method for developing a set of test patterns for a circuit and determining its fault coverage consists of the following steps:

1. Generate a test set for the circuit, either manually or using an automatic test pattern generation (ATPG) program,
2. Use a fault simulation program to calculate the resulting fault coverage, and
3. Generate more tests until fault coverage is acceptable (or time runs out).

This approach unfortunately is affected by several problems:

- running the ATPG and fault simulation programs can each require many hours*
- we may not know how to improve a circuit with poor testability
- the design must be detailed down to the gate level.

Several programs have been reported in past which estimate testability of a circuit without running an ATPG program. These measures have not been successful at providing accurate estimates, and they are implemented at the logic level, requiring the circuit design to be (essentially) complete.



The Test Coverage Estimation demonstration in Phase I shows the feasibility of computing testability measures of a design consisting of pre-analyzed library components. Controllability and fault coverage calculation algorithms were implemented for the Phase 1 demonstration. The observability calculation will be developed in Phase 2 as a modification of the controllability algorithm. The advantage of this approach is that the computationally expensive fault simulation is done once for the library components, and stored in the test profile. When the design uses library components, these data can be retrieved for the testability estimation algorithm.

Testability knowledge required for each library component consists primarily of a **test set** developed for the component and a list of the faults detected by each pattern of the test set. The testability estimate of a design is determined by considering each component one at a time as the

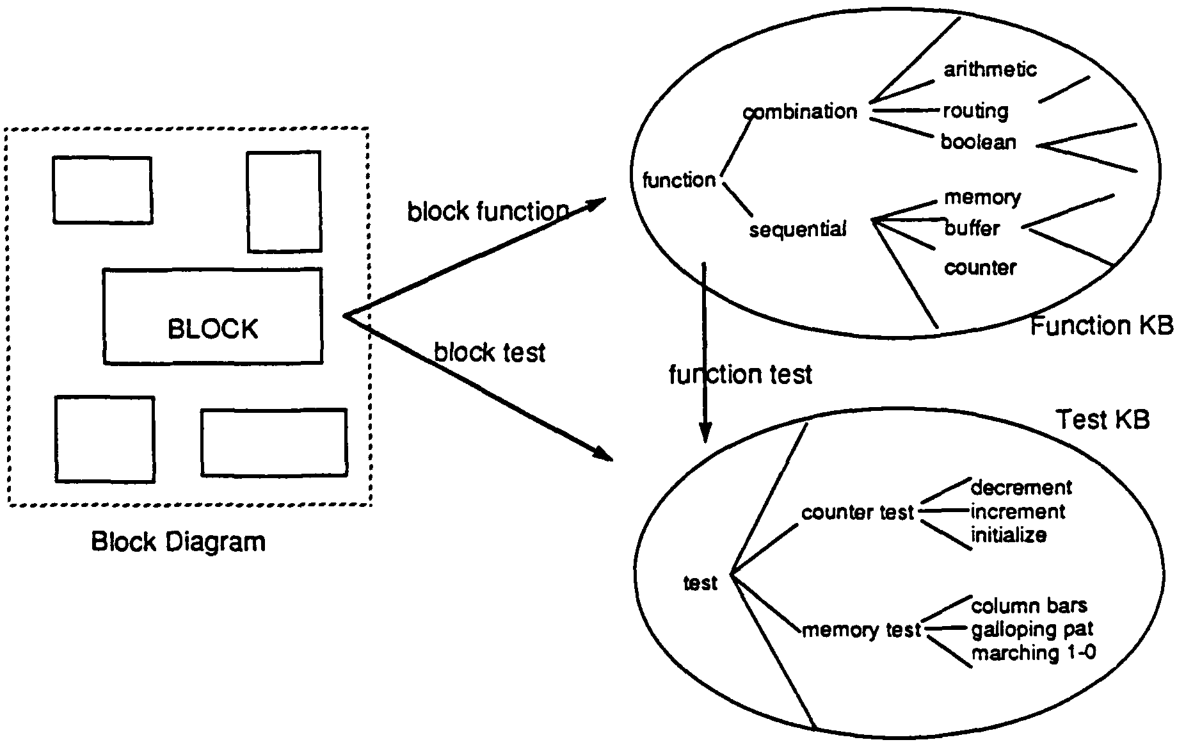
* Sometimes this is even impossible. A recent article in Computer Design estimated the running time of a fault simulation program to be 38 years for a 68000 based micro-processor!!

test focus. A coverage estimate for the test focus is calculated by determining the set of tests which can be propagated from primary inputs to the test focus (this is called the controllability set, or **c-set**), comparing the c-set to the component test-set and calculating the percentage of faults detected, and propagating the tests through the circuit to primary outputs (the difference patterns which can propagate form the observability set, or **o-set**). The calculated c-set and o-set for the component in the design are stored in its test-profile to save recomputation during design modification

Our Phase I system provides early estimates of test coverage, before gate level design, by using pre-analyzed test data and caching computed data for faster calculation of fault coverage for incremental design modifications.

5.3 Functional Test Advisor

The Functional Test Advisor gets the description of the functions of blocks in a block diagram by having the designer traverse the Functional Hierarchy knowledge base and then select functions. Test advice at the block-design level requires the system to have enough knowledge about the design to know what needs to be tested. The prototype advisor gets the functional definition from the designer. An investigation of the feasibility of inferring block diagram functionality from similar designs or module and signal names will be addressed in later phases.



The DFT advisor retrieves functional test alternatives from the Functional Test knowledge base. The functions in the Function Hierarchy have links to functional tests. After the designer chooses test alternatives, he or she selects structural requirements to access signals needed for applying the tests. Tests in the knowledge base are stored with a general definition of signal access requirements. The designer selects the test requirements and uses the mouse to select particular signals in the design. They are stored in the testability checklist for verification by the advisor. Controllability test requirements are considered satisfied if there is a direct test route from primary inputs through blocks which have an identity mode (functions in the function knowledge base having an identity subfunction) to the control signals. Observability requirements are similarly considered satisfied if there is an identity path to primary outputs. If the system cannot find identity paths, the requirement is not known to be satisfied.

The designer will either provide more specific functional design input, go on to more detailed design, or perform testability analysis after getting to the library module level.

5.4 Fault Simulator Interface

As part of the phase 1 demonstration we developed a wrapper for a commercial fault simulation software package, the Lasar system from Teradyne. The fault simulation was used to analyze library modules. The fault simulation needed the following inputs: detailed netlist, test patterns and types of faults to be covered (stuck-at-1 and/or stuck-at-0 at all the pins). Based on these, the faults detected by the test patterns and fault coverage information were obtained. Most of the inputs needed were generated automatically. When the user clicked on specific modules, the interface allowed for remotely executing the fault simulation on another processor, and displaying the results to the user.

6 Conclusions

One critical problem for concurrent engineering is getting early feedback to designers about the effects of design decisions on downstream processes. Automated tools can provide designers substantive assistance in identifying and tracking areas of concern from perspectives other than design for functionality. Such tools can handle routine design review and allow human reviewers to concentrate on difficult, novel problems. This will serve to reduce the amount of communication required for concurrent engineering by shielding human engineers from handling repetitive, routine analysis and review. The Design for Testability prototype is an initial investigation of the feasibility of and requirements for a design environment which facilitates concurrent design for testability with functionality during early stages of design. The principal foci of research were (1) integrated or unified data and knowledge representation of design data; (2) techniques for giving knowledge-based assistance to the designer from an independent perspective, in this case testability; and (3) data and knowledge requirements for giving knowledge-based assistance. These are discussed in the following subsections.

6.1 Integrated design representation

Tools that give designers feedback from the perspective of other engineering disciplines must be able to manipulate design data and give meaningful results to the designer. The design representation must be capable of maintaining functional design information, and handling information required by other perspectives. The design representation developed for the DFT application is a unified, frame-based representation. The Design for Function perspective is based on the VHDL standard. The Design for Testability perspective is included in a test profile for each module.

Our model for integrating data is the hub model. The "data hub" is a unified representation of design data, where functional, structural, and test data are stored in a single intermediate representation. Tools view design data from a specific perspective, which is mapped into the unified representation. For example, the design editor directly manipulates components using a structural perspective. Individual tools are not concerned with viewing data from different perspectives unless there is a purpose to do so. The model allows for incremental and evolutionary addition of perspectives.

6.2 Techniques for DFT

The Design for Testability capability demonstrated in Phase 1 utilizes two complementary approaches. First, high-level test guidelines are established by user interaction with the functional test knowledge base. Designers can select testability concerns for tracking by the system. Second, fault coverage estimation is implemented for designs which were composed of pre-analyzed components. This gives the designer a testability measure before final design decisions are made. Problem identification and design suggestions are planned enhancements to these advisory modes.

The knowledge-based test advisor demonstrates the feasibility of providing initial testability guidelines on a block-level design. The designer can indicate design intent by selecting functions for the design components. Functions are linked to functional tests in the knowledge base. Structural test requirements are selected by the designer and recorded in the test guidelines for the design. This system demonstrates the ability to represent functional test knowledge from a testing perspective, and make it available to designers who are using a functional perspective.

Testability measures are indicators of the ease of testing designs. Useful measures would tell the designer of potential test problems as early as possible in a computationally efficient manner. The approach taken in Phase 1 is to provide fault coverage estimates for designs composed of pre-analyzed components. The goal of this work is to provide coverage estimates earlier than is currently possible using standard tools. The Phase 1 DFT fault coverage prototype demonstrates the feasibility of providing rapid calculation of estimated fault coverage earlier than detailed design. An example set of pre-analyzed components is used to build designs. Each library component is stored with its test set and fault dictionary. A component is controllable if the set of signals which can be propagated from the circuit primary inputs stimulates a high (user-specified) percentage of faults. The design is considered controllable for testing if each component is controllable. If estimated fault coverage is not as high as desired, the user can modify the design and quickly recalculate the coverage estimate.

6.3 Data and Knowledge Requirements for DFT

Our Phase 1 effort has helped us identify the data and knowledge needed to develop a full functional DFT advisor. These requirements can be classified on the basis of the design stage the information is needed.

In the conceptual stage, the advisor needs to help the designer pick an appropriate test strategy which is consistent with the test requirements. The system needs a broad view of applicable testability techniques and the cost associated with applying them to help the designer make informed decisions and trade off functionality, performance and test cost.

In the functional design stage, the system needs a taxonomy of commonly used functions and the associated test information. The Phase 1 effort resulted in identifying the structure of the knowledge base.

In the structural/functional stage for designs that are based on pre-analyzed modules, the Phase 1 effort identified the information that needs to be stored with each module and how that information could be utilized efficiently.

6.4 Summary

In summary, we made significant progress on several fronts to enable concurrent design of testability with functionality and performance. We developed and identified,

- a VHDL-compatible, hierarchical design representation which also keeps track of test-related information,
- mechanisms which identify testability problems early in the design phase, based on intended functionality or on library modules,
- the data and knowledge requirements for developing a functional system that allows the concurrent design of testability with functionality and performance.

7 Future Work

Based on our Phase 1 experience, we plan to extend the initial DFT system to handle more complex designs. This will help in the validation of the methodology and allow us to assess the usefulness and accuracy of the advice provided by the DFT system in the various design phases.

We will develop mechanisms to support the analysis and maintenance of library modules, improve methods to analyze the testability of designs based on library modules, expand the knowledge-base of test plans and methods, and develop functions to help in planning test strategy.

8 Publications

8.1 Workshops

A. S. Matsumoto, V. Jagannathan, C. Buenzli, V. Saks, "Concurrent Engineering Workstation for High Density Electronics," presented at the IEEE Workshop on Design for Testability, April 1989, Vail Colorado.

A. S. Matsumoto, V. Jagannathan, C. Buenzli, V. Saks, "Concurrent Design for Testability," presented to the Concurrent Engineering Design workshop, IJCAI 1989.

8.2 DICE reports

L. Erman (editor), "Comments on DICE Red Book" report prepared by Cimflex Teknowledge, June 15, 1989.

V. Jagannathan (editor), "HDE Architecture Requirements Analysis" report prepared by Cimflex Teknowledge, Jan 17 1989.

9 Hardware

The DFT prototype system was demonstrated on a Sun 3/60 platform. The Lasar software ran on a MicroVax workstation and was interfaced to the Sun via an Ethernet connection.

10 References

Abadir, M. S., and Breuer, M. A., "A knowledge-based system for designing testable VLSI chips," IEEE design & test of Computers, August 1985, pp 56-68.

Birmingham, W. P., and Sieworek, D. P. "Single board computer synthesis," in *Expert Systems for Engineering Design,* edited by Rychener, M. D., Academic press, 1988, pp 113-140.

Buelow, Fred and Porter, Ed, "The Need for Fault Simulation," VLSI Systems Design, 1986.

Carter, Donald and DenBeste, William, "Managing the Design to Test Process," ATE and Instrumentation Conference East, 1987.

Chaing, Michael, and Bloom, Michael, "Timing Verification for System-Level Designs," VLSI Systems Design, Dec. 1987.

Cortner, M. J., *Digital Test Engineering,* John Wiley & Sons, New York, 1987.

Huisman, Leendert M., "The reliability of approximate testability measures," IEEE Design and Test of Computers, December, 1988.

Kirkland, Tom and Mercer, M. Ray, "Algorithms for Automatic Test Pattern Generation," IEEE Design and Test of Computers, June 1988.

Knopf, R., and Trischler, E., "CERBERUS: A hierarchical DFT Rule Checker," *Proceedings of the 1st European Test Conference*, April 1989, pp 58-61.

Marlett, Ralph, "The Risp Methodology", HHB Systems, September 1988.

McCluskey, Edward J., *Logic design principles with emphasis on testable semicustom circuits*, Prentice Hall publishers, Englewood Cliffs, 1986.

McCluskey, Edward J., "A Survey of Design for Testability Scan Techniques," Semicustom Design Guide, Summer 1986.

Miczo, A. *Digital Logic testing and simulation*, Harper & Row, publishers, New York, 1986.

Mott, Geoffrey and Newkirk, John, "Eliminating Failures in the Field," VLSI Systems Design, October 1986.

Muehldorf, Eugen I., and Savkar, Anil D., "LSI Logic Testing -- An Overview," IEEE Transactions on Computers, January 1981.

Parker, Ken P., *Integrating Design and Test*, Computer Society Press, 1987.

Parker, Ken P. "The impact of boundary scan on board test," IEEE design & test of Computers, August 1989, pp 18-31.

Rizzatti, Lauro and Wasilewski, Mary, "Worst-case Timing Analysis Ensures Board Reliability," Computer Design, Nov. 15, 1987.

Singh, N. *An artificial intelligence approach to test generation*, Kluwer Academic Publishers, 1987.

Teradyne, Inc., "Automatic Generation of Functional Tests for PLDs", VLSI Systems Design, April 1986.

Teradyne, Inc., "Design for Testability: Its Impact on Testing at the Board Level," Dec. 1988.

Tulloss, Rodham E., "Automated board testing: Coping with complex circuits," IEEE Spectrum, July 1983.

Turino, J., *Testability Advisor*, Logical Solutions Technology, Inc., 1985.

Wang, Laung-Terng, Hoover, N., Porter, E., and Zasio, J., "SSIM: A Software Levelized Compiled-Code Simulator," Proceedings of 24th IEEE/ACM Design Automation Conference, 1987.

Zhu, X., "A knowledge-based system for testable design methodology selection," *Technical report CRI-86-23*, Unversity of Southern California, August 1986.

DICE PROGRAM

Task 3.4.7.2 Design for Assembly Advisor **West Virginia University**

A. Objectives:

The research efforts under this task have focused on the following five primary objectives:

1. To define the rules and guidelines for the manufacturability an assemblability of electronic components and synthesize them into an Electronic Assembly Ranking (EAR) software module;
2. To develop a computer module Manipulator PATH Machines Planner (PATHPLAN) for collison free motion of robot manipulators during automated electronic or mechanical assembly;
3. To develop a software module for simulation and optimization of an electronics assembly workcell. The optimization of assembly depends on the number and type of parts, the workcell and feeder arrangements as well as other constraints, such as part complexity;
4. To develop the interface software need to allow the transfer of printed circuit board lay-out data from a standard CAD package to the printed circuit board assembly work cell being developed by Cimflex Tecknowledge and enhance the interface software to perform all the necessary data base checks to insure complete information transfer, and
5. To define rules and guidelines for the manufacturability and assemblability of mechanical components and provide the information to the designer early in the design stage.

B. **Approach:**

1. The rules and guidelines of electronic assembly have been defined for subsequent integration in an expert system shell. These rules allow the evaluation of a PCB (printed circuit board) design with respect to automatic assembly compatibility, manual operations requirements, testability, solderability, cleanability, inspectability and reliability. The designer can use the ranking result to compare candidate PCB designs and to select the "best" design based on the available assembly workcell;
2. The path planner software applies an new obstacle avoidance algorithm to compute the collision-free assembly paths. This obstacle avoidance algorithm uses the growth of obstacle and plane-unfolding techniques to facilitate the determination of a collision free path. An A* searching algorithm, followed by a backtracking strategy, is used to actually identify all the via points along a collision-free path. A motion planner, then, computes the time for each of the collision-free, paths using the velocity and acceleration constraints that are imposed on each degree of freedom in the assembly robot. Motion synthesis procedures, that account for system constraints such as the maximum velocities and maximum accelerations of robot joints, are used to determine the time required to complete the specified workcell assembly mechanics;
3. The method used for the optimization of assembly sequence relies on the implicit enumeration of the assembly steps. In this method, an objective function is written containing the sum of all possible assembly paths. In addition, constraints on the assembly process are written. These include the workcell constraints and the path constraints. In the test case used in Phase I, there were nineteen constraint equations satisified. The

optimum assembly sequence is found by enumerating the possible assembly paths and checking each possible sequence with the objective function and constraints. If a sequence does not lower the objective function or violate one or more constraints, it is eliminated. This allows the elimination of a large number of solutions implicitly without actually enumerating them. This greatly reduces the time required to reach an optimum solution;

4. Traditionally, teaching pendants have been used to "instruct" a robotic mechanism to perform the motions specified by a user. The ability to rapidly adapt to new changes and/or new board designs dictates the use of a better system than one that requires a teaching pendant; the information should instead be encoded in an IGES file. A translator or post processor, eliminates the teaching pendant encoding, a most time-consuming operation. The requirements of the IGES post processor led to developing a software program capable of sequentially reading an IGES file created by a modern CAD package. The program has the capabilities to digest the information contained in the file and determine the meaning of each line within the IGES file. Once the line's meaning is determined, a comparison with a known workcell parameter file, (those components that the workcell can pick and place), is completed, and
5. A framework for integrating rules and guidelines for mechanical assembly into an expert system shell was established, according to various types of assembly constraints like component orientation, fastening, etc.

C. **Technical Results:**

1. The overall organization of EAR software was developed. After PCB design data are input from the common database, the first task of the ranking software is to classify the components from the standpoint of assembly. The classification is based on the following factors which affect the PCB assembly: board construction; circuit track layout; solder masking; through-hole parts; component spacing; component orientation; general layout, and component selection.

   These major factors contribute in various proportions to the assembly areas. All of these rules are stored in an expert system and each rule is assigned a relative weight. The rating can be computed from the weights. EAR can automatically show the evaluation and rating results on windows;

2. The path planning algorithm being developed for robotic assembly provides proven collision avoidance paths for the end effector among static obstacles. This algorithm has improved the efficiency of other published techniques through the use of such enhancements as the path plane-unfolding principle. The program dynamically displays the assembly sequence of the chips and the trajectory of the robot end-effector during the assembly operation. A two-dimensional collision avoidance algorithm has also been implemented in the program. While the camera and all the assembled chips are considered as obstacles, the chips in the feeders are not. The program automatically generates collision avoidance paths from feeders to the camera for inspection and from the inspection to the destinations on the PC board.

3. The optimization program developed in this task determines the optimum assembly sequence for a given workcell arrangement. The software reads a file of input data containing a list of components on the PCB and their location. This file is generated from an IGES file which will be the eventual output of the PCB designer in other modules. The workcell geometry and parts feeder data are also read from an external file.

   The constraints on the optimization problem are read from an external file or may be input during the sequence of the program. At this point, all required data has been input , and the optimization process takes place when an optimum assembly sequence is generated. In a test case, a 27% savings in assembly paths was found over an arbitrarily assigned sequence; and

4. The results of the Post Processor developed for the PCB Assembly Workcell were best depicted in the Phase I, July 25, 1989, Demonstration. The demonstration showed a sample surface mount board drawn by AutoCAD. The intermediate file generated by AutoCAD, an IGES file, was loaded by the post processing program for workcell interfacing. Along with the input file, a compare file (those components that the workcell can process) was loaded. Similarly, an output file, where the program digested the processed information, was specified.

   The translation sequentially read each of the IGES encoded lines. It then deciphered the locations of the workcells capable components and listed the cartesian coordinates of each occurrence.

D. Conclusions:

As a result of this research, there are a number of important conclusions:

1. EAR software is an efficient means to address the PCB manufacturability and assemblability issue for a given assembly workcell by providing

component classification, relationships of components and early detection of PCB design flaws;

2. The PATHPLAN program is capable of rapid assessment and/or reassessment of the minimum operational distance and time required to achieve a specified assembled arrangement of physical parts for a PCB;
3. A method of implicit enumeration, as an assembly optimization tool has been successfully demonstrated. This tool appears to be applicable to assemblies with large numbers of components due to its reduced execution time. The time required for the execution of the optimization program on the mainframe computer is approximately 30 seconds for a PCB with 15 components, and
4. The interface software used between the Design Advisor or any CAD printed circuit board design package that generates the standard IGES file and the printed circuit board assembly workcell (developed by Cimflex Teknowledge) provides the platform and format for flexible operation of the workcell.

E. Recommendations:

From this research, several recommendations can be made:

1. The EAR software should be integrated into an expert system, tested under real conditions and extended to provide a bidirectional interface with the designer;
2. Future work on the PATHPLAN software should consider the interaction between the cell components and the robot arm, especially to assure that the entire robot manipulator arm itself avoids collision, within the cell workspace, and with the workpiece or the equipment;

3. The assembly optimizations program should be expanded and refined to include larger assemblies, automatic generation of constraint equations, mechanical as well as electronic assemblies, optimization of feeder arrangements, the option of manual assembly and real time simulation of the assembly process;
4. The scope and flexibility of the CAD-Assembly Workcell translator should be expanded to interpret other intermediate file types and other robotic workcells. This enhancement could produce a package that could interpret any CAD file and produce the coordinates needed to place components on any robot, and
5. Selected rules and guidelines for mechanical assembly should be ranked by sending out questionaires to industry experts working in this area and get their views on the importance of these various rules as regards their effect on the assembly process. An optimal assembly plan should be developed subsequently using a CAD database.

F. **Publications:**

None

G. **Hardware:**

The EAR software runs under on IRIX operating system on a Silicon Graphics Personal Plus workstation. The software is written in C.

the hardware, utilized for the CAD-workcell interface program, is the IBM compatible Zenith PC/AT. MS-DOS used on the operating system with the Microsoft Basic version 4.5 was used as a programming language. The developed software was coined "IGES Post Processor" and the present version is 1.02. Associated hardware that will be used with the

Cimflex Teknowledge Robotic Workcell and the Daisy Boardmaster Software running on the Sun 386i. This software will be utilized in the Phase II demonstration.

DICE PROGRAM

Task 3.4.8.1 LCC Analysis Model GE Aircraft Engines

Objectives:

LCC (Life Cycle Cost) analysis at the early stages of product development cycle can influence the design of the component to minimize the overall cost of the system. Therefore a designer needs a cost analysis model to study trade of performance and cost of designs.

Approach:

LCC includes costs of the RD&T (Research , Development & Test) , Manufacturing , and O&S (Operation & Support) . During Phase 1, we developed a prototype frame for the manufacturing cost estimate system with overall LCC system objectives in mind.

For prototyping of the Manufacturing Cost Estimate system , a data modeler interviewed value process engineers and design engineers to determine the cost estimating needs and requirements. The system development team analyzed the requirement document and restructured the requirements. From the restructured requirements , costing data flow diagram was generated.

To meet the costing requirements, several commercial cost estimating packages were evaluated. One commercial package was selected as generic cost estimate frame. To simulate the value process engineer's inspection procedure of physical part dimension and shape , available CAD geometry data were selected for inspection on the computer screen.

From the analysis of costing requirements , a data modeling was performed and costing data modeling diagram was generated.

Initial prototype frame is to be developed by implementing the requirements and data modeling results. The prototype system will be incrementally developed with validation of the system performance by using the value process engineer's test case data.

Technical Results:

The requirements as a result of the interview with value process engineers and design engineers for the Manufacturing Cost Estimate system was documented. The requirements were classified into 23 categories (Table 3.4.8.1-1) and restructured into 6 different costing tasks ;

1. Find Similar Part
2. Determine Material Costs
3. Determine Process Costs
4. Calculate Estimated Costs
5. Determine Tooling/Gauging Costs
6. Document Findings

The subtasks for each of the 6 tasks are represented in the Figure 3.4.8.1-1. One view of interrelationship between the costing tasks is represented in the Figure 3.4.8.1-2. Among them , Find-Similar-Part is the first and most crucial task for rapid cost estimation. Also found were the differences of cost estimation methods between military and commercial engine parts.

Based on the requirements document , several commercial software packages were evaluated. The packages selected were personal computer based "bottom-up " part cost estimating systems. Only PC based systems were evaluated so that every participant could access to the prototype. Among the 7 packages evaluated the development team selected "E-Z Quote" system (by E-Z Systems Inc). It allows the user to customize the database of manufacturing process operations, hardware and material. 102 operations per part , 999 parts per assembly are allowed in the E-Z Quote. It operates under DOS ,OS2, UNIX and was written in "C". Therefore the system can be easily integrated with the DICE architecture and operation system. Other evaluation summary is in Table 3.4.8.1-2 .

However none of the commercial packages satisfied all of the user requirements. The following user requirements were not met by the packages;

- o Locating Similar part geometry (CAD interface) and "copying" a previously developed estimate
- o Determining part and material weight from the drawing
- o Determining casting/forging cost
- o "What-If" analysis capability to determine the effect of design changes on part cost.

Therefore the development team focussed on implementing those 4 functional requirements. To simulate the cost estimator's inspection procedure to determine dimension and shape of the physical part, available VersaCAD geometry data were brought on the computer screen during prototyping phase.

Data Modeling of the Manufacturing Cost Estimate was performed for prototype implementation. The results of the data modeling are shown in the Figures 3.4.8.1-3 and -4.

A prototype MCE (Manufacturing Cost Estimate) system frame was developed by integrating E-Z Quote , VersaCAD Part Geometry. Validation of the system performance and incremental development of the prototype need to be carried in the next phase.

Conclusions:

Prototype frame for Manufacturing Cost Estimate (MCE) system has been developed. Incremental implementation with validating test case data needs to be done.

The MCE system requirements were comprehensively documented. To develop a prototype , the scope of the MCE system was focussed on the precision parts and components such as aircraft engines.

During the data modeling and prototype frame development for the MCE system, the value process engineer's more active and timely participation was needed to complete the project on schedule. For further incremental development of the MCE and LCC system, the value process engineers' time should be firmly committed during the entire phase of the system development.

Costing information is business sensitive matter and the system should allow the user to set up " Proprietary Input Data" for costing process and then to save or easily remove the data when cost estimate task is completed.

Because of frequent change of the participating team members the continuity of the system development was not effective. Both Principal and System implementor left company.

Recommendations:

Continue to develop the MCE system by validating the system performance with known test case data , and implementing more functions defined in the requirements incrementally .

The more advanced functions are ;

- .Comprehensive What-If analysis capability
- .Automated CAD Geometry Data retrieval for costing methods determination
- .Flexible Historic Database availability for Similar part cost scaling

Value process engineers' full commitment to participate in the system development should be made upfront.

Initiate to determine the RD&T , O&S cost estimate requirements and LCC costing architecture.

Prototype Software used:

E-Z Quote : A loan copy was obtained from the vendor for prototype evaluation.

VersaCAD : GE Site-licensed copy and sample engine part geometry data was evaluated for prototype MCE development.

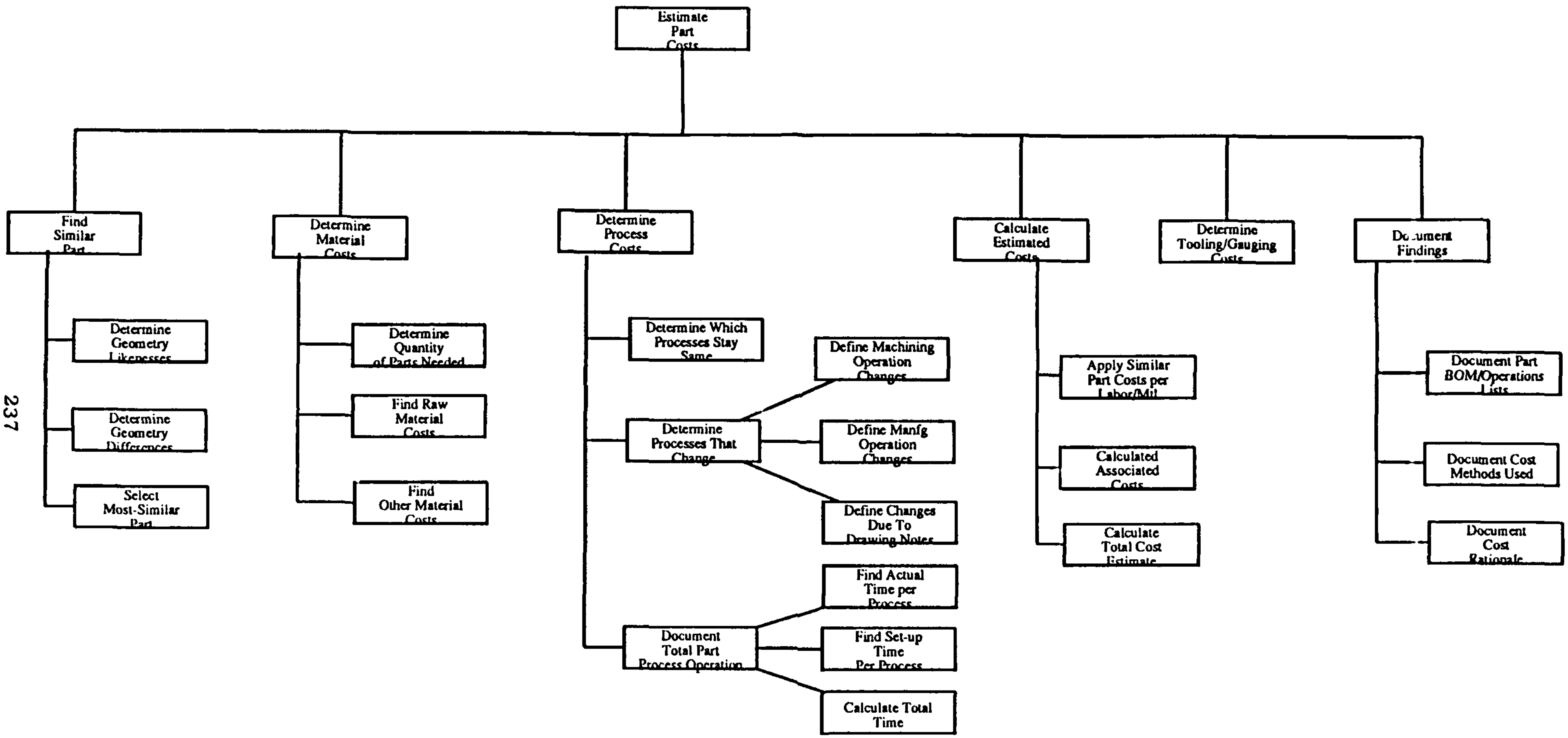


Figure 3.4.8.1-1 Costing Tasks

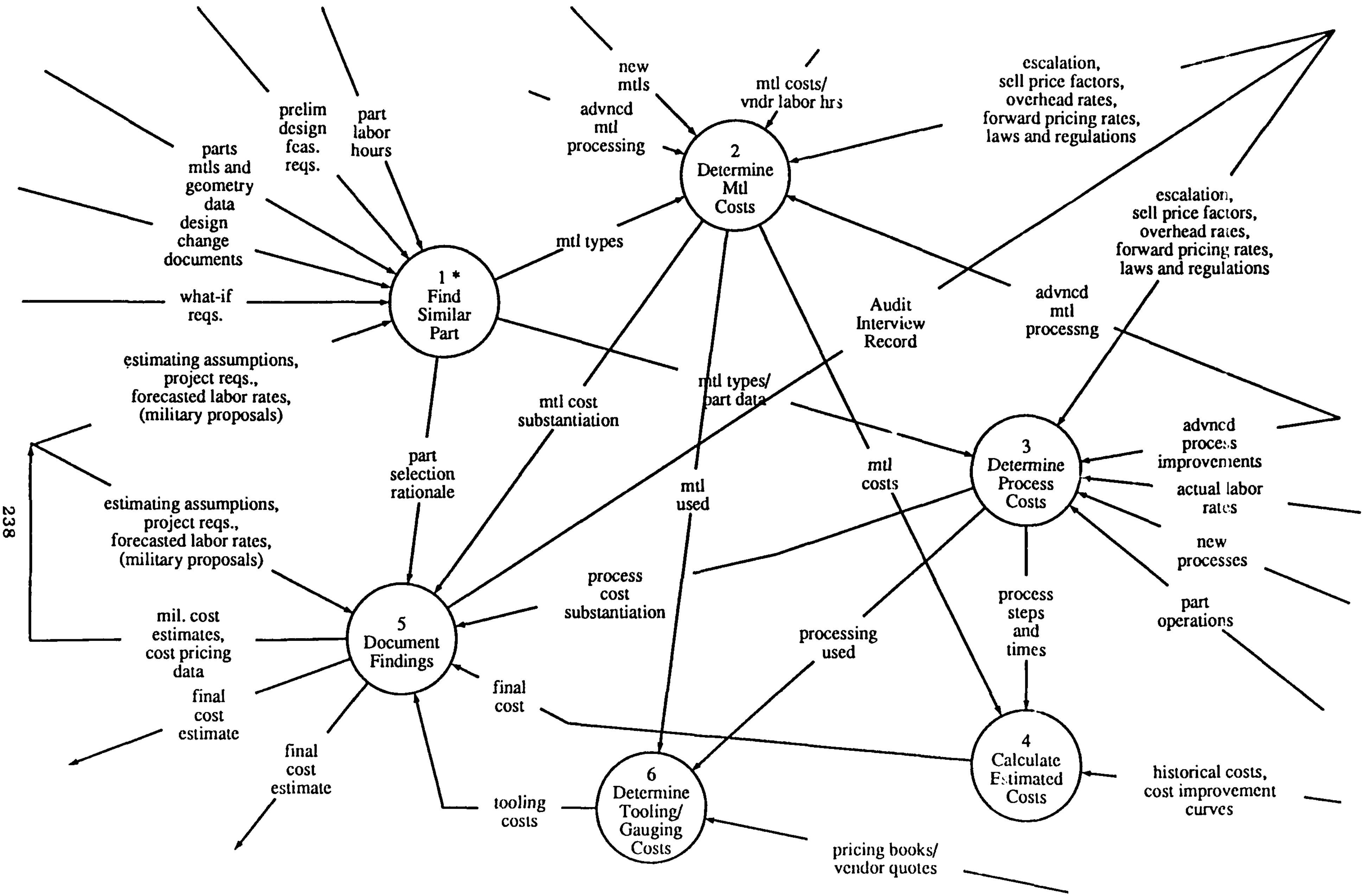


Figure 3.4.8.1.-2 A View of Inter-relationship Between Costing Tasks

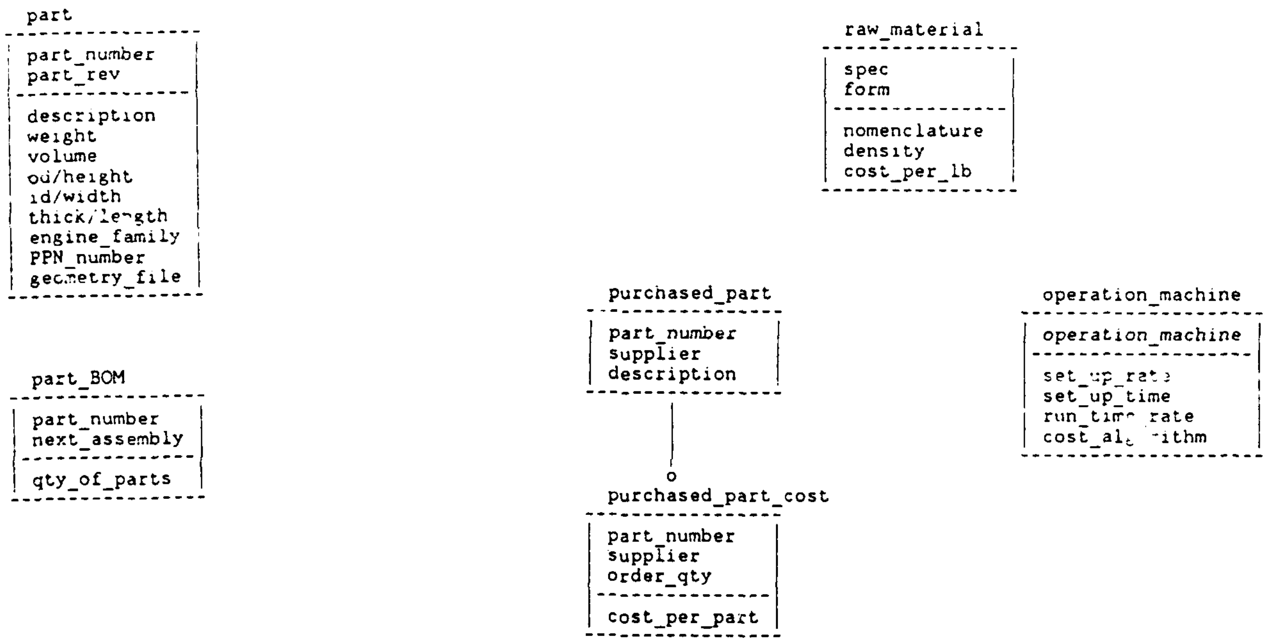


Figure 3.4.8.1-3 Part Cost Estimate Data Model 1

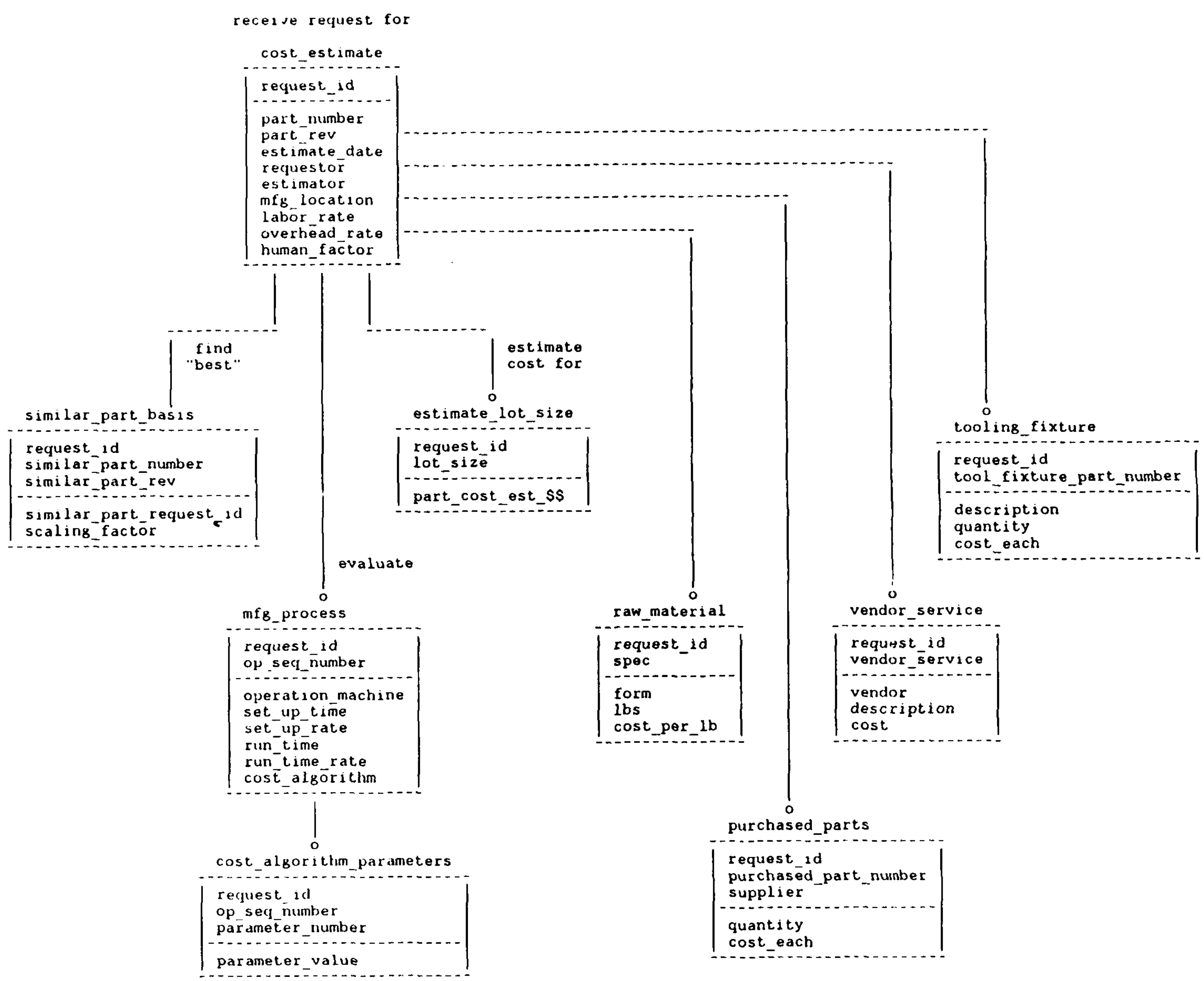


Figure 3.4.8.1-4 Part Cost Estimating Data Model 2

Table 3.4.8.1-1 MCE Requirement Documents

LIFE CYCLE COST SYSTEM

Requirements Document

R1. Compare geometry of PD part number and similar Part(s)

- C1. Call up part geometry of PD parts
- C2. Call up drafting part(s) geometry
- C3. Indexing - Lists of parts by: engine family, part type, component type, material type, ...

R2. Automatic calculations of process costs, material costs, tooling costs.

- C1. Algorithms
- C2. Historical parts (Drafting)
- C3. Manufacturing processes
- C4. Pricing Books

R3. Display any drawing notes requested.

- C1. Specifications and directions

R4. *Display any specs requested.*

- C1. Input any spec number
- C2. Relate specs to drawing notes

R5. Input/retrieve specs for the part to cost.

R6. Query the similar part(s)'s process plan and modify to accommodate the part to cost.

R7. Input new tooling requirements.

- C1. Reference by process, or tooling part number

R8. Electronically send estimate and substantiation to whomever needs it.

- C1. Calculations
- C2. Assumptions
- C3. Sources

R9. Calculate, report and graph expenditure/liability curves for management.

R10. Input and report on program plan information.

R11. Secure the total system so that individual pieces of information are available to those who need it to do costing.

****

R12. Interface with existing systems.

C1. CAD database - CALMA, ComputerVision - for engine cross sections
C2. GDSS - Honeywell - engine cross sections
C3. IGES - geometry format
C4. Purchasing - to get PO numbers to know the basis

R13. Display/modify a set of operations.

C1. Suggest list
C2. Modify/add/delete operations

R14. Way to go back to a similar part

C1. PPN number's associations to part numbers
C2. Part description groupings
C3. Tied to PO number of shop cost

R15. Know how to modify base cost to get present cost.

C1. Material changes
C2. Scaling factors

R16. Maintain learning curves by source.

C1. Outside vendor vs. in-house cost

R17. Standardize PPN numbers across drawings.

C1. Generic number over all drawings for the position in an engine

R18. See parts list for engine size and family.

C1. Modify list

R19. Maintain engine production schedule for past and present engines.

C1. Number made, number of parts made
C2. To determine position on learning curve
C3. If basing cost on another part
C4. If starting from scratch

R20. Interpret in-house cost figures to PPN's.

R21. Interpret Revenue Sharing figures.

C1. Offsets - so much guaranteed to be made overseas

R22. Integrate with ALEST tool.

R23. Maintain raw material cost tracking and automate material costing.

C1. Processes for raw material
C2. Methods of production

Table 3.4.8.1-1 Requirement Documents

C3. Weight of finished part and raw material
C4. Cost of raw material per pound
C5. Material types

Table 3.4.8.1-1 Requirement Documents

Package	E-Z Quote
Vendor	E-Z System Inc, Addison, IL Dan Ermenchuq (312-953-1555)
Price	$2,995.00 (for $99 can use package on a 30 day trial basis)
Hardware/ Software	IBM PC compatible. Operates with DOS, OS2, UNIX, XENIX and networking systems.
Data Entry	"fill in the blanks" as the user "pages" through data entry screens
Geometry Interface	key-in part dimensions
History Data Collected	customized databases up to 504 operations/workcenters up to 2400 hardware items up to 2520 material types
Material	calculated by volume or by weight (optimizes stock utilization). cost of material is entered as a factor of cost per pound.
Tooling / Rework / Scrap Cost	allows input of cost markup (profit, G&A, labor burden factor), vendor costs (amortized fixed, each and lot charges) and additional charges not amortized listed separately.
Machine / Workstation / Routing	process operations are selected from the customized database and the user is prompted for set-up time and information relating to the formula used to calculate machine time per part. formulas are stored in the database. up to 102 operations allowed per part.
Output	detailed in-house estimate sheet, routing sheet and a customer quote letter. quotes can be stored for later recall/modification.
Govt Audit	
Miscellaneous	estimates part and/or assembly using BOM (up to 999 parts per assembly allowed) on-line help is available at any time takes 2 minutes to install - "easiest to use package" output may be used by other programs (Lotus 123 interface)

Table 3.4.8.1-2 Evaluation Summary of Commercial Costing Packages

Package	**Computer Aided Manufacturing/Cost Estimating**
Vendor	**Tedaci Co, Willowick, OH** **Donald W. Ginther (216-946-0293)**
Price	**$ 3,500.00 CAMCE - standard package, in which, using menus, user selects the machine, sequence of operations, type of cut and the cutting tool material** **$10,000.00 CAMCEP - includes CAMCE package user enters the blueprint dimensions baseline, material and material size (stock size) and the software determines how to process the part, select all feeds and speeds, determine the number of cuts (rough and finished), select the tooling, determine tool life, choose the machine and cost the part. (NOT available until March '89)** $ 175.00 per day for on sight demo
Hardware/ Software	IBM PC compatible Written in BASIC, compiled, source is not made available to purchaser.
Data Entry	CAMCE - MENU DRIVEN CAMCEP - input dimensions, material and material size
Geometry Interface	Note, the blueprint data input to the CAMCEP software includes such items as tolerance, pitch, major od, number of cutting teeth; items which are not generally held in a geometry file.
History Data Collected	machine tool : slide velocity, tool change time, max spindle speed, labor rate, overhead rate, standard data allowance, obsolescence factor, etc. (24 factors) material : grade/class, price per pound, machinability rating, brinell hardness and material overhead (5 factors) user may add machine and material data
Tooling / Rework / Scrap Cost	
Machine / Workstation / Routing	does not include broaching, honing, hobbing, stamping, non-conventional and miscellaneous operations. software modifications required to add conventional machining operations not covered (i.e. broaching) could be made (most changes would require our paying for the modifications)
Output	detailed quote analysis, mfg time estimating summary, tooling summary, part cost estimate
Govt Audit	
Miscellaneous	

REVIEW NOTATION : For those operations not covered a "miscellaneous" cost module would be used to add operations.

Table 3.4.8.1-2 (Continued)

Package	E-Z Quote
Vendor	E-Z System Inc, Addison, IL
Price	(for $99 can use package on a 30 day trial basis)
Hardware/ Software	IBM PC compatible. Operates with DOS, OS2, UNIX, XENIX and networking systems.
Data Entry	"fill in the blanks" as the user "pages" through data entry screens
Geometry Interface	key-in part dimensions
History Data Collected	customized databases up to 504 operations/workcenters up to 2400 hardware items up to 2520 material types
Material	calculated by volume or by weight (optimizes stock utilization). cost of material is entered as a factor of cost per pound.
Tooling / Rework / Scrap Cost	allows input of cost markup (profit, G&A, labor burden factor), vendor costs (amortized fixed, each and lot charges) and additional charges not amortized listed separately.
Machine / Workstation / Routing	process operations are selected from the customized database and the user is prompted for set-up time and information relating to the formula used to calculate machine time per part. formulas are stored in the database. up to 102 operations allowed per part.
Output	detailed in-house estimate sheet, routing sheet and a customer quote letter. quotes can be stored for later recall/modification.
Govt Audit	
Miscellaneous	estimates part and/or assembly using BOM (up to 999 parts per assembly allowed) on-line help is available at any time takes 2 minutes to install - "easiest to use package" output may be used by other programs (Lotus 123 interface)

Table 3.4.8.1-2 (Continued)

Package	Computer Aided Manufacturing/Cost Estimating
Vendor	Tedaci Co, Willowick, OH
Price	$ 3,500.00 CAMCE - standard package, in which, using menus, user selects the machine, sequence of operations, type of cut and the cutting tool material $10,000.00 CAMCEP - user enters the blueprint dimensions baseline and the software determines how to process the part, select all feeds and speeds, determine the number of cuts (rough and finished), select the tooling, determine tool life, choose the machine and cost the part.
Hardware/ Software	IBM PC compatible
Data Entry	CAMCE - MENU DRIVEN CAMCEP - key in dimensions
Geometry Interface	
History Data Collected	machine tool : slide velocity, tool change time, max spindle speed, labor rate, overhead rate, standard data allowance, obsolescence factor, etc. material : grade/class, price per pound, machinability rating, brinell hardness and material overhead
Material	
Tooling / Rework / Scrap Cost	
Machine / Workstation / Routing	
Output	detailed quote analysis, mfg time estimating summary, tooling summary, part cost estimate
Govt Audit	
Miscellaneous	

REVIEW NOTATION : The literature provided on the CAMCE/CAMCEP software is limited because the vendor is updating to a new system.

Table 3.4.8.1-2 (Continued)

Package	MiCapp Cost Estimating System (designed for machine shop)
Vendor	MiCapp Inc, Big Rapids, MI
Price	$1,995.00 (single user) $900.00 (multi user license)
Hardware/ Software	IBM PC compatible. BASIC (code is not compiled and may be modifed)
Data Entry	MENU DRIVEN
Geometry Interface	
History Data Collected	
Material	MENU of 34 major groups of materials from AISI 1212 up to 303 stainless (includes two grades of cast iron aluminum and cast zinc). material costs such as forgings or castings may be entered as a cost per piece.
Tooling / Rework / Scrap Cost	markup, scrap and efficiency factors may be entered. allows inclusion of tooling, engineering, packing and delivery costs as costs per piece.
Machine / Workstation / Routing	using a menu, select operation, machine (80 included) and machine function. speeds, feeds and time values are calculated. may include up to 25 inside and 10 outside operations per part.
Output	estimate, routing and customer quote quotes can be stored for later recall/modification. summarizes up to 25 individual parts per estimate.
Govt Audit	
Miscellaneous	allows several individual part quotes to be summarized up to 9 different lot sizes per part supports learning curves for re-pricing

Table 3.4.8.1-2 (Continued)

Package	Quote IV
Vendor	Anderson O'Brien, St Paul MN
Price	
Hardware/ Software	IBM PC Compatible
Data Entry	MENU DRIVEN with user prompts
Geometry Interface	user prompted for part dimensions and tool paths
History Data Collected	material, speed and feed tables (composites of reference tables and experience data) which may be modified to agree with own tooling, fixturing and machining requirements. customer name and address file.
Material	
Tooling / Rework / Scrap Cost	accounts for shop efficiency, scrap rate, machine rate for each operation, material usage, costs and markups, outside vending costs and markups, expendable tooling and one time tool costs
Machine / Workstation / Routing	generates machine time calculations for turning, drilling, milling and flame/laser cutting. allows input of secondary operations with standard times. allows input/default for setup, run and handling times.
Output	Summary Sheet which includes the operations and material requirements for up to five quoted quantities Summary Sheets can be stored for later recall/modification.
Govt Audit	
Miscellaneous	menu of conversion/calculator options for interpreting prints

REVIEW NOTATIONS : severely limited in "menu" process operations (turning, drilling, milling and flame/laser cutting)

Table 3.4.8.1-2 (Continued)

Package	Quotemaster
Vendor	Randall Data Systems, Livonia, MI
Price	
Hardware/ Software	IBM PC compatible
Data Entry	MENU DRIVEN - "fill in the blanks"
Geometry Interface	
History Data Collected	No data is currently being stored (i.e. material costs, setup/run times). A new software version is to be released approximately May '89 which will include material, outside services, customer and workcenter (labor rate) data storage. Currently the package uses default values (i.e. one labor rate) which are a part of the company profile.
Material	
Tooling / Rework / Scrap / Costs	allows input of material scrap, manufacturing burden, administrative overhead, profit, labor efficiency and commission percentages. allows input of material, outside service (piece and flat fees) and tooling per piece costs. amortizes fixed tooling costs and allows for perishables.
Machine / Workstation /	user enters up to 15 operations with separate setup and runtimes (a routing template may be created)
Routing	
Output	quote summary which may be stored and retrieved by part number, quote number, quote status, customer name, date received or date due.
Govt Audit	
Miscellaneous	allows up to 4 input quantities.

Table 3.4.8.1-2 (Continued)

DICE PROGRAM

Task 3.4.8.2. Cost Modeler West Virginia University

A. Objectives:

This task has both general and specific objectives. The general requirements include: designing a cost accounting system adapted to the cost estimating needs of concurrent engineering (CE) manufacturing operations; developing cost collection methods appropriate to a CE-specific information system; specifying activities that have cost estimating functions developed for them and documenting all aspects of cost modeling within the CE structure for the purposes of technological reporting and transfer. Descending from these general goals, there are specific cost modeling objectives: implementing the developed cost accounting system; developing data exchange protocols and attendant software methods; deriving, finding and adapting cost functions for identified activities and obtaining operational data from the demonstration facility to illustrate the veracity of the system's documentation.

During Phase I of the project, both overall and phase-specific objectives were pursued. For example, particular short-term goals included:

1. Developing a method for estimating costs from principles of learning curves and traditional multiple regression analysis, and
2. Estimating manufacturing costs for investment casting of turbine blades.

B. Approach:

The software needs of this task consisted of being able to :

- import data;
- export data;
- respond to queries;
- manage databases;
- perform computations and table lookups;
- execute branching logic;
- make access user-friendly;
- produce reports, and
- allow for easy expansion.

These objectives were compared with the software planned for inclusion in the DICE architecture and the Cost Modeling staff selected spreadsheet software as naturally satisfying the above listed needs.

Within a spreadsheet language structure, modules, representing the various aspects of the cost procedures, were prepared and tested. Part of these modules were used for the Phase I Demonstration and other modules are being refined for the Phase II Demonstration.

Throughout this first phase, the project staff researched the literature on costing, computer models and cost analysis. In particular, costing procedures at GE, and within the aerospace industry, were investigated along with developed equations for specific manufacturing processes.

To estimate manufacturing costs for the investment casting of turbine blades, cost equations for various cost drivers were found. These cost equations were installed in a LOTUS spreadsheet, within a cost-estimating module, for turbine blades of this type.

Preliminary work was done on developing generic cost estimation procedures. Specifically, the task staff derived methods for predicting changes in manufacturing costs due to learning curve effects.

Because GE was listed as having responsibility for the Life Cycle Costing aspects of the project under Task 3.4.8.1, in the original project proposal, an attempt was made to integrate this task with the G.E. task. This effort was unsuccessful.

C. Technical Results:

As a result of work in this task, a set of general criteria, that CE cost modeling systems must satisfy, has been developed. These criteria are:

(1) All permissible design and manufacturing processes must be specified prior to cost system development;

(2) Cost gathering procedures and data protocols must be in place before cost systems can work;

(3) Architecture structure and procedures must be provided that facilitate the necessary data exchanges before cost systems can be effectively employed in a CE environment, and

(4) The overall CE cost accounting system must be understood throughout the CE functions.

Because not all of these criteria have been addressed in the current project, the Cost Modeling staff assumed some manufacturing processes, very simple data exchange procedures, architecture compatible with spreadsheet software and a cost accounting procedure for cost gathering and estimating. As a result, a cost model for estimating the cost of an investment casting of a turbine blade was developed. This cost model was inserted as a module in the Cost Modeling spreadsheet structure and the basic model was modified to include scrap costs. The main cost drivers included pattern preparation, mold preparation, melting and testing.

A general model for predicting learning curve effects on costs was also investigated. In developmental work, it is imperative to predict the prototype developmental standard time for first unit as well as for standard production data. Therefore, a second model was developed to predict production standards from prototype development data. The models were made more flexible by including the capability of employing multiple learning rates.

The basic template for the main modules of the Coat Modeling system were structured and installed in LOTUS spreadsheets. This template includes data input and storage sections, user query sections, calculation areas, parameter storage sites and collections of cost functions. The template can be readily expanded to encompass additional processing activities.

D. Conclusions:

The Cost Modeling staff has discovered that only very restricted models and procedures have been recorded to successfully address cost modeling problems inherent in CE systems. For instance, interactive cost accounting systems needed to make small-lot, flexible systems operate with understandable costs have not yet been published. Moreover, few process-specific and material-specific cost estimation models exist in the literature. In fact, existing cost estimation procedures have been geared primarily to large enterprises. Parametric methods also appear to be unsuited to innovative designs commonly surveyed by cost modelers. Similarly, large manufacturers employ costing methods that estimate costs only after bulk production data are available. In sum, the specific kinds of cost modeling procedures needed for a CE facility are not readily available.

The research to date has revealed:

(1) The kinds of cost drivers that need to be included in cost estimators;

(2) The characteristics of a workable cost accounting system adapted to CE needs;

(3) How to employ learning curves in cost predictions, and

(4) How to employ multiple regression analysis techniques to derive cost driver functions from raw data.

The Cost Modeling staff also has determined its expectations for data exchange procedures and protocols; however, the implementation of these protocals lies outside the task's function. With respect to these facets, there are a number of recommendations.

E. Recommendations:

It is recommended that the total project begin efforts to facilitate the continuation of the Cost Modeling task toward it rational goals. Specifically, we recommend that:

- A spreadsheet package be obtained and installed within the DICE architecture;
- The permissible design and manufacturing processes for the planned demonstration facility be identified;
- There be an effort made toward establishing the architectural structure and procedures required to facilitate information exchange between project tasks;

- The needs of the project cost accounting system be integrated into the project's information structure, and
- The essential data sets required throughout the project be identified and standardized.

Until these integrative activities are seriously addressed it will be impossible to attain the interactions necessary for simultaneity. The absence of even a rudimentary information exchange system is impeding project progress across the major tasks. Specifically, this lack of information constitutes a considerable constraint on the formulation and refinement of cost modeling activities.

With respect to the Cost Modeling task itself, continuation activities should focus primarily on extending work already in progress. These recommendations are:

(1) Development of composite shape time standards for basic shapes. Nomograms of basic relationships will be produced and regression studies will be performed to develop equations for standard time estimation. Some composite data on the T34 fan is available, and regression techniques will be applied to these data. The results of this particular study will allow the comparative analysis of variable relationships in complex and simple shapes;

(2) The learning curve model in hand needs to be extended to facilitate estimations of average unit times when multiple learning rates are assumed. The effects of different learning rates and break points need to be understood in order to model costs effectively, and

(3) <u>The concurrent engineering cost accounting model needs to be extended, implemented, and tested in parallel with its continual theoretical development.</u>

F. **Publications:**

None

G. **Hardware:**

None

Task 3.4.9.1 Engineous Software GE-CRD

1 Introduction

A key issue in the design of an aircraft engine is the existence of many islands of knowledge and developing the appropriate bridges between these islands. The bridges between these islands have been very narrow as a result of the strict division of work between aerodynamic and mechanical designers, resulting in an inefficient design process. By designing from a system or global viewpoint, engineering productivity improves due to integrated/automated analysis procedures, and a multidisciplinary approach increases knowledge transfusion among the designers. A concurrent design of flowpath and blading with simultaneous consideration of both aerodynamic and mechanical design criteria under an integrated AI design environment demonstrates the methodology.

The goals of this project is therefore to develop the methodology and tools for concurrent design of engine flowpath and blading, leading to significant increase in engineer productivity and superior product quality.

2 Design/Analysis Process for Aircraft Engine

The normal design and analysis procedures of an aircraft engine would go through three major cycles: (i) the design of engine configuration, (ii) the design of blading with aerodynamic considerations, and (iii) the design of blading with mechanical/structural criteria. A detailed design/analysis sequence and flow chart is shown in Figure 1.

The design of the engine configuratoin determines the engine performance and consists of designing the configuration profile, the number of stages, the loading distribution, etc. The design process then performs a series of aerodynamic analyses such as CAFD to determine the streamline curve and flowpath, and BLADES to define blade geometry. Once the blading designs satisfy aerodynamic requirements, they are then turned over to mechanical designers for further investigation on the structural integrity in term of vibration, stress concentration, fatigue and other mechanical criteria. If the design is not structurally sound, it will given back to aero designers for modification until both aero and structural design criteria are satisfied. Only in rare circumstances will they go back to step one for complete engine redesign.

The CF6-80C2 engine was chosen by GEAE designers, Brent Gregory (aerodynamic) and Omer Erdmann (mechanical), as the demonstration problem. The vast design experience acquired from the design of this engine has been and will continue be very useful for implementation, verification and support as the project develops.

A derivative of this engine is one of the latest engines under development at GEAE.

3 Engineous Design Methodology

Engineous is a generic software shell that can be applied to the design of many different products. Its basic philophy is to remove the tedious iterative design/analysis program work from the engineers, freeing them to perform more creative and skilled functions.

Engineous consists of a rich variety of utilities and interfaces to perform the tedious work required to iterate design analysis codes such as manipulating and editing input data, running programs, browsing output files, and applying design knowledge to modify design parameters. By delegating their work this way, engineers can realize significant improvement in their productivity, will minimize human error, and may improve their designs.

Through automation Engineous explores more design options and parameter trade-offs in a given period of time. Additionally, Engineous' own searching heuristic often turn up new answers, improves designs and resolves conflicting goals. The design rules and criteria that were acquired from the engineers will be available for future design and other designers. Thus it creates an natural way to pass and accumulate knowledge and experience. When the design knowledge is incomplete, which is often the case with rapidly changing technology or very large knowledge domains, Engineous combines its own heuristic searching scheme with the existing knowledge to solve the problem.

4 Implementation of Concurrent Engine/Blade Design in Engineous/DICE Phase I

4.1 Design Methodology of Concurrent Design in Phase I

The design methodology in phase I focuses on demonstrating a first cut coucrrent aero and mechanical design/analysis process. The abbreviated analysis programs and design procedure is shown in Figure 2. The performance analysis is predicted by the TP3 (Turbine Performance Pitch-line Prediction) program. The aerodynamics analysis will be ignored here and was substituted with a rudimentary blade geometry definition module (BG). It then go through "PSECT" to generate meshes for finite element analysis. SIESTA and ESPEC consist of several sub-modules to allow specification of external loadings, boundary conditions and material properties for the creation of ANSYS data files. It then calls ANSYS to perform both static and dynamic analyses.

First the engine performance was optimized to get flowpath geometry. Then, the blading optimization was performed without aerodynamic coupling. (The missing pieces of aerodynamics analysis codes such as CAFD and CASC will be integrated in phase II). Thus, modifications on the blade geometry has been driven only by mechanical criteria and design rules. ANSYS structural analyses are conducted on all blade rows during each design path. The design criteria of all blade rows are evaluated, taking their interactions into account. Design parameters are updated simultaneously using coumpound rules when the changes suggest an improvement in the engine design. The design methodology for concurrent engine/blade design is shown in Figure 3.

4.2 Assumptions and Limitations

The following assumptions have been made in phase I due to time restraint and scope of work:

- Preliminary blade design
- Rudimentary blade geometry definition
- No direct aerodynamic and mechanical coupling
- Aerodynamic analysis and penalty not included
- Design modification driven only by mechanical design criteria
- Some design knowledge and rules may be specific to this problem

4.3 Integration of Analysis Modules

The data flow of Engineous/DICE phase I modules is shown in Figure 4. Salient points of these modules are described as follow:

TP3 – Turbine Performance Pitchline Predictor module was adopted from an existing application to optimize engine performance. Contact Dr. Siu Tong or Dave Powell for more information.

BG – It uses seven parameter to defined the geometry of each blade section. Three sections namely tip, pitch and hub together define a blade geometry that is rudimentary, but believable. Two of the parameters, inlet angle and outlet angle are derived from the output of TP3. The other five parameters are maximum percent thickness, location of maximum thickness, thickness of leading edge and trailing edge and blade curvature parameter. The last set of parameters are normalized with respect to chord length, except for the blade curvature parameter. Because design rules are available only for blade thickness and chord length, two auxiliary parameters tmax-scalar and chord-scalar were defined so that the sectional changes will be propotional, thus maintaining the original blade shape.

BSTACK – takes the output from BG and some radii and z-axial location information from TP3 to calcuate boundary coordinates and the location of each blade section. First, the BG output and derived TP3 output have to be merged into a temporary file. The file is then split into 10 blade row files each containing 3 blade sections.

PSECT – The output from BSTACK is processed by PSECT to generate a finite element mesh.

SIESTA – is used to store finite element input and output in a random access data base. It can also prepare input files for several types of finite element programs. For our purpose, it is used to generate ANSYS data files. There are several sub-processes inside SIESTA. They are: air2uif, uif2rdb, rdb2adk and prep7.awk. The interactive prompts of these programs are put into a single file called "responses.ata" file. A C program named "ata" is then used to convert all modules from interactive mode to batch mode. Therefore, this file must be modified if other options within these modules are needed. Meshes generated by SIESTA are shown in Figures 5a-5d.

ESPEC – is a material library use by SIESTA to get material properties for finite element analysis. Because the material is referred to by a symbolic name, it can be defined as a design variable in the future. The material tables are converted into binary format, espec.dat, and can be called by SIESTA directly.

ANSYS – The ANSYS jobs are forked by a C program, "engansys". The output file from running ANSYS is renamed from "PARM.ANS" to "parmx.ans" where "x" is the run identifier in order to prevent being overwritten by later runs. All these files are concatecnated into a single file, "parms.all".

4.4 Mechanical Design Criteria and Rules

There are three type of design constraints, static stress limits, vibrational constraints and physical restraints. Figure 6 shows design rules for boundary conditions and Figure 7 shows the external pressure loading of physical restraints. Static stress limits can be modified by thickness or chord length and have no interaction with other blade rows. The vibrational constraints are more difficult to handle due to their interactions with neighboring blade rows. The Campbell diagram in Figure 8 shows their complexity.

The vibrational requirement is to place the vibrational mode of the current blade row within the feasible zone that is the region between RED-1 and IDLE-2 or RED-2 and IDLE-3 or above RED-3, as shown in Figure 9. Thus, the vibrational mode of the current blade row may want to change upward or downward depending on its relative position with respect to its neighboring blade row. When no feasible zone exists, the current blade row and its neighbor blade rows may have to be changed simultaneously in order to create a feasible zone.

Because Engineous can not handle compound rules and dynamic limits, the design criteria and rules are coded in a Lisp program called "erdman-rules.lisp" to control the variable changes.

5 Suggested Improvements of Engineous Capability for Phase II

Although Engineous provides a very good foundation for system integration and automation, several deficiencies have been identified for further improvements. Due to the complexity of this problem and some new unexpected function requirements, the current Engineous shell should be augmented in order to handle more sophisticated applications and design demands.

Compound Rules – The heuristic search scheme in ENGINEOUS only perturbs one parameter at a time as a result other blade row analyses will be performed in vain. Without compound rules smart pre- and post-processing wrappers have to be written specific for this problem. Also, a engineer would usually change several parameters when he has enough design rules and knowledge. A heuristic search that changes one parameter at a time become very inefficient for a complicated and interrelated problem like this.

Dynamic Limits – The Campbell diagram for vibration constraints required constant updates on the upper and lower bounds of these vibrational constraints. Without this capability, a big assumption to fix them must be made a priori. Ideally when the optimal result found violates these constraints, the upper and lower bounds should be updated and the whole design sequence should be repeated until the vibration constraints are satisfied.

Auxiliary Parameters – In phase I work, some auxiliary paameters are required as design variables. Since design variables can only be input parameters, this can not be done without modifying the source code.

Sequential Initialization – Engineous tries to evaluate all units in a knowledge base concurrently. This will usually require unnecessary initialization of many parameters in many kowledge bases. If the evaluation is done sequentially or the error is ignored initially, a lot of grief could be avoided.

Derive Arrays from Arrays – Currently Engineous allow parameters to be derived from other parameters, parameters associated with an array but does not allow arrays to be derived from other arrays.

Transition Hook – Some analysis modules are simply data transformation routines. They don't really require the effort of setting up a knowledge base. At present, they are handle by including them in a C script as part of procedure calls.

Symbolic Name with Blank Spaces – The material name for this problem may have blank spaces in between. These blank spaces are important as ESPEC must have identical string match to be able to get the material properties correctly.

Clear Error Message – The error handling facility is very poorly done in Engineous. The message often does not tell where and what is wrong with your setup. Sometimes, it is hard to tell which screen is active and is supposed to give you the lastest run time information. Debugging requires a person with proper training in Lisp and KEE. Another common error for a novice user is to do multiple clipping on a window selection due to the slow response of the system and/or lack of a clear indicator of whether an option has been selected.

6 Conclusion

The design methodology and analysis procedure for concurrent engine/blade design is impelmented in the Engineous AI shell environment. It started with optimization of engine configuration and performance. The aerodynamic analysis was substituted with a rudimentary blade geometry definition module in phase I. The mechanical analysis procedure, the structural criteria and mechanical design rules are mostly complete in the phase I. The design process to meet constraint is handled by a Lisp wrapper program which essentially implemented the whole design criteria and rules in itself. In phase I, the design modification is driven by mechanical design criteria and rules only. Phase II will modify the design based on aerodynamic consideration as well.

A lot of experience has been gained on how to use Engineous effectively. Many functions that cannot be handled internally by Engineous were resolved by C scripts. before wrappers and after wrappers externally. This makes a proper training course very important. Deficiencies in the current Engineous capabilities such as compound rules, dynamic limits and transition hooks, have also been identified. These new functions may be important for the success of the Phase II project.

Fig. 1 ENGINE/BLADE DESIGN PROCEDURES WITH AERO AND STRUCTURAL ANALYSES

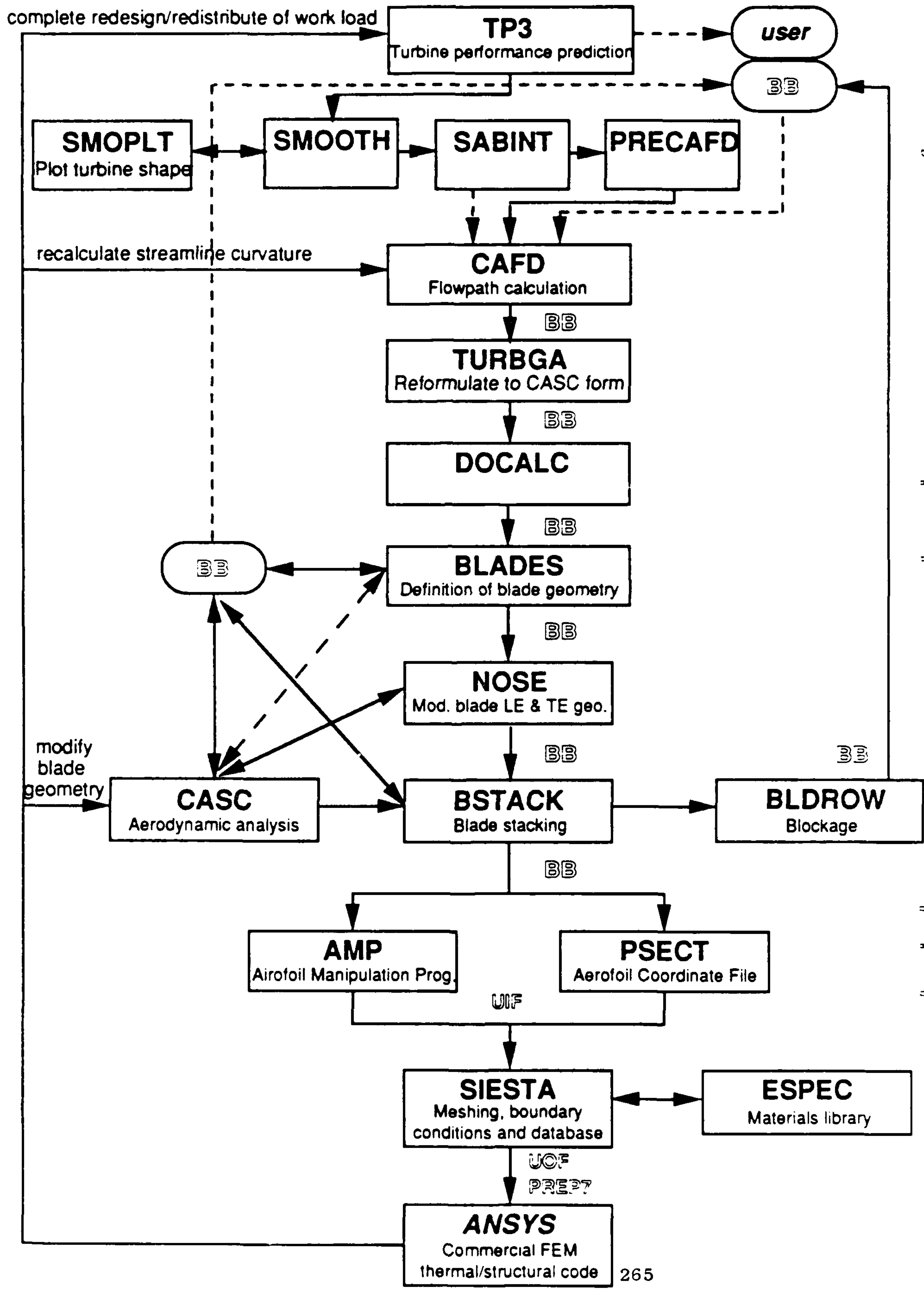


Fig. 2
ANALYSIS PROGRAMS AND DESIGN PROCEDURE

PHASE 1

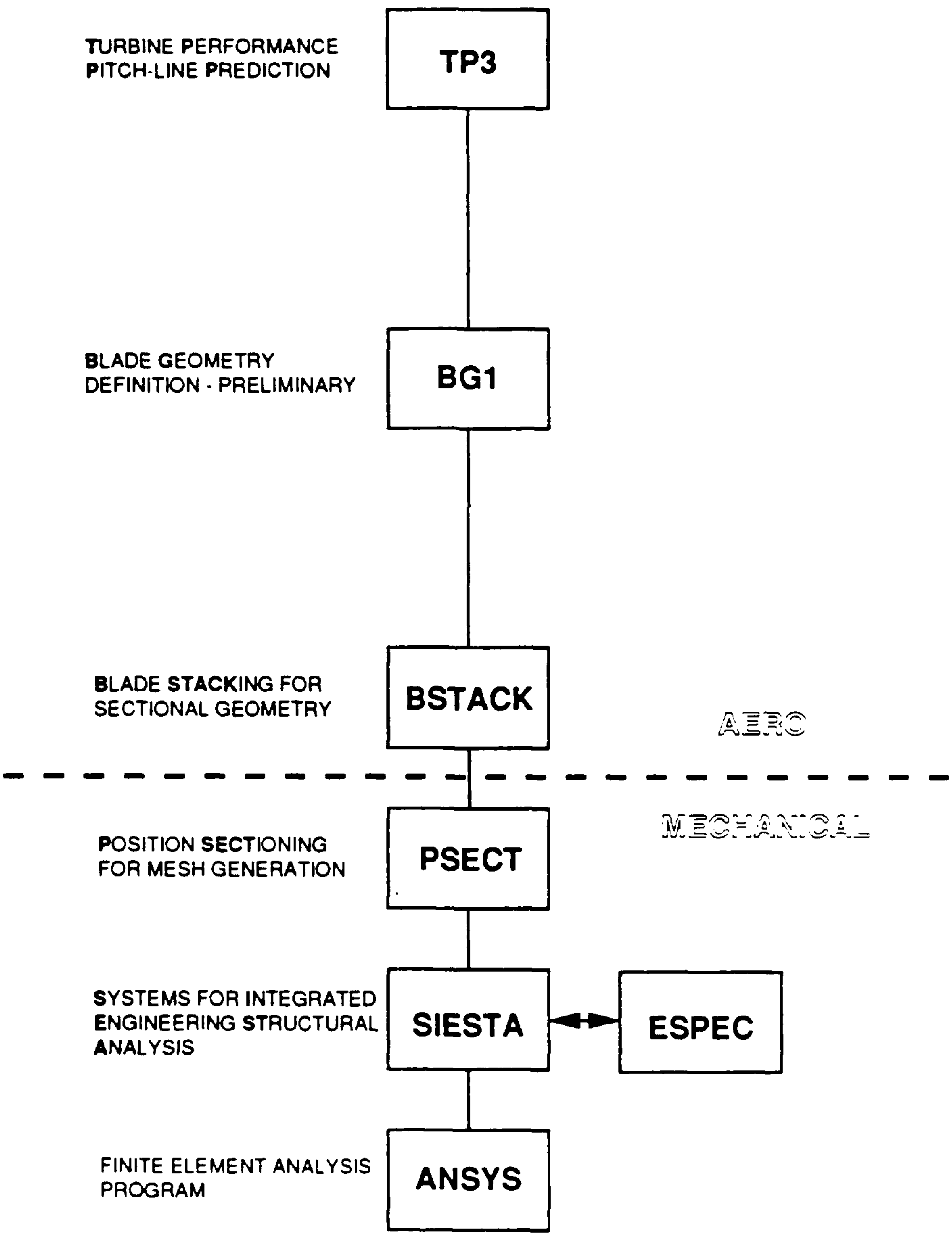


Fig. 3

DESIGN METHODOLOGY FOR CONCURRENT ENGINE/BLADE DESIGN

PHASE I

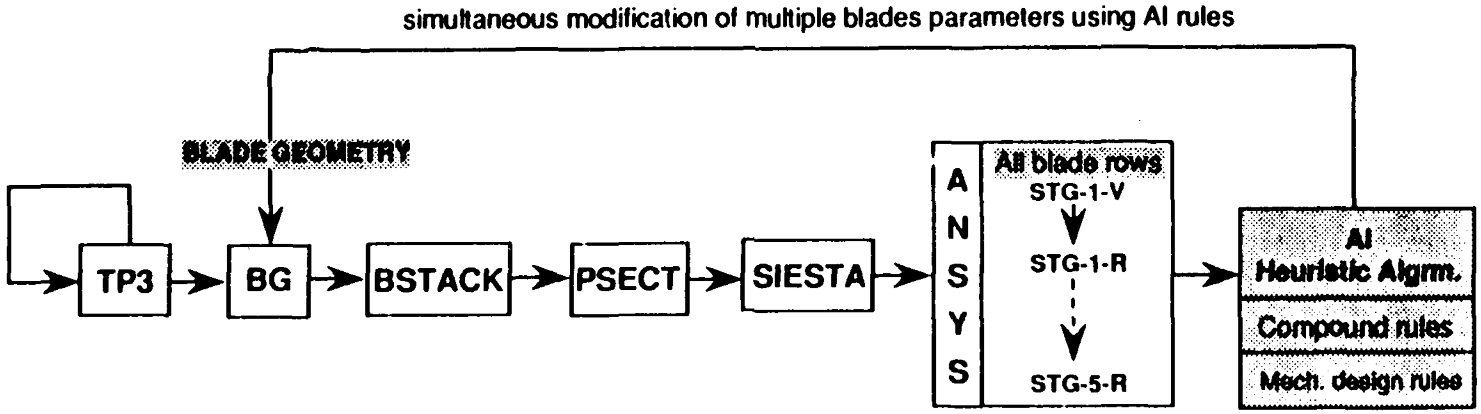
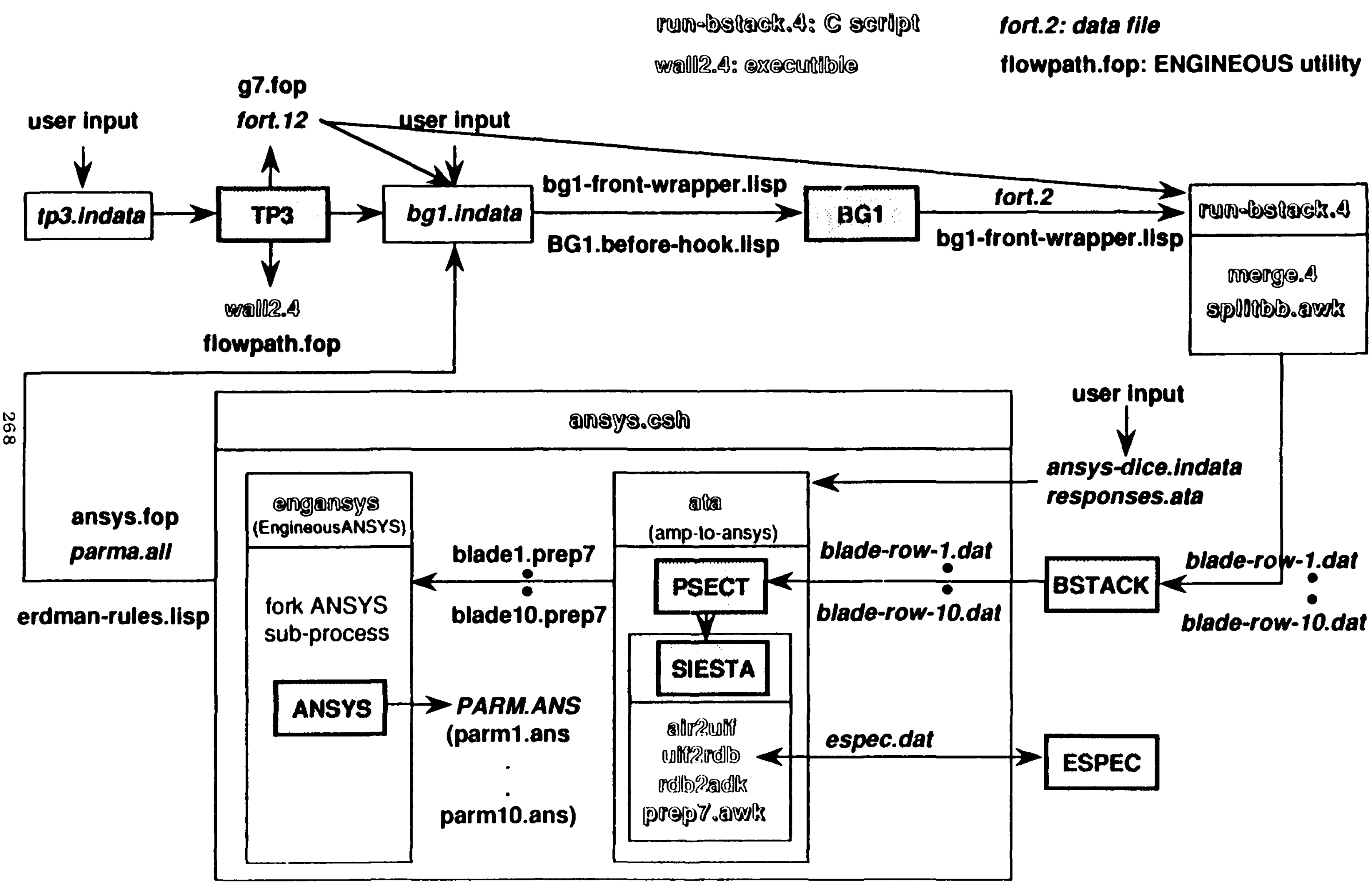


Fig. 4

DATA FLOW OF ENGINEOUS/DICE PHASE I



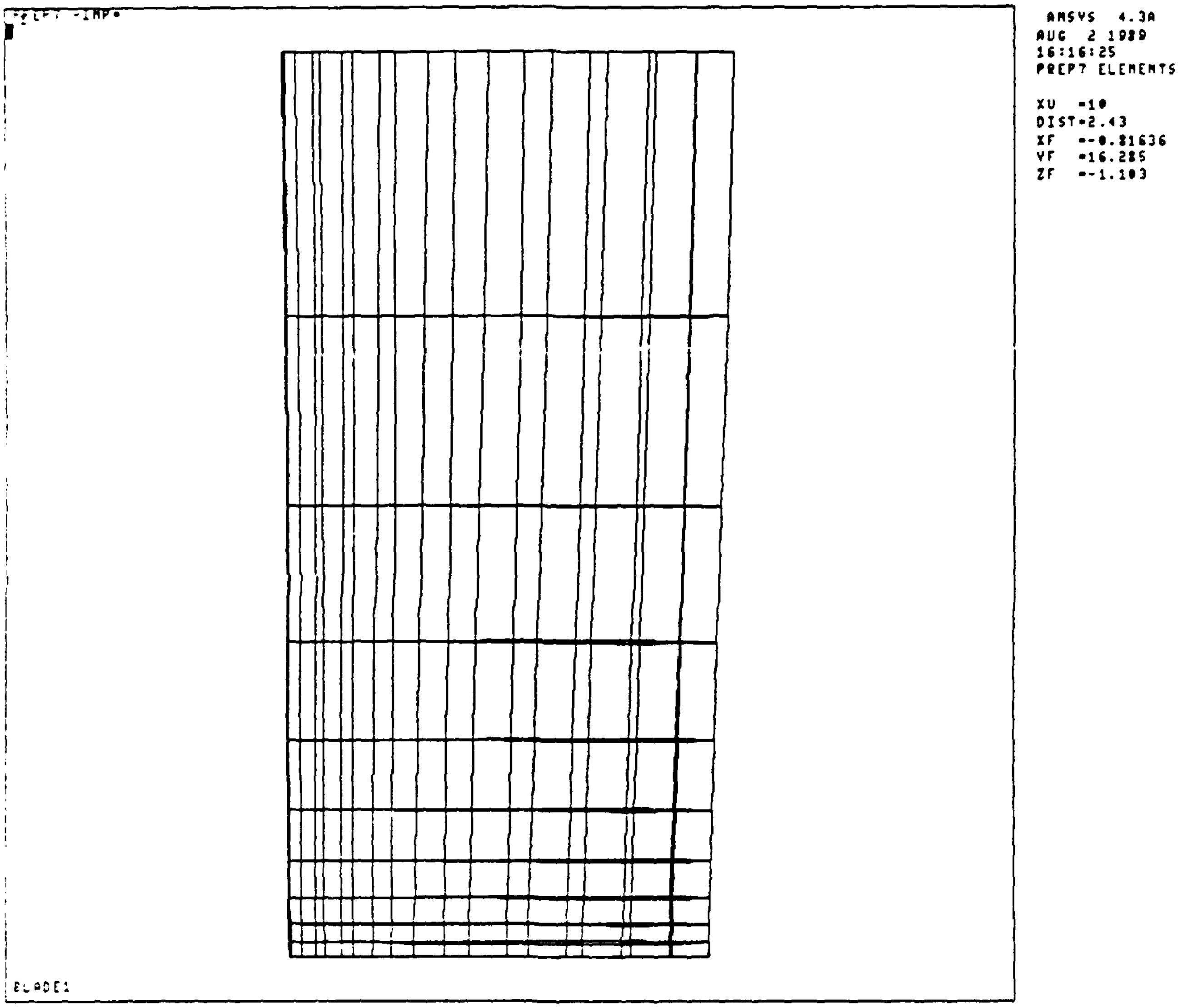


Fig. 5a ANSYS mesh for Stage 1 Stator

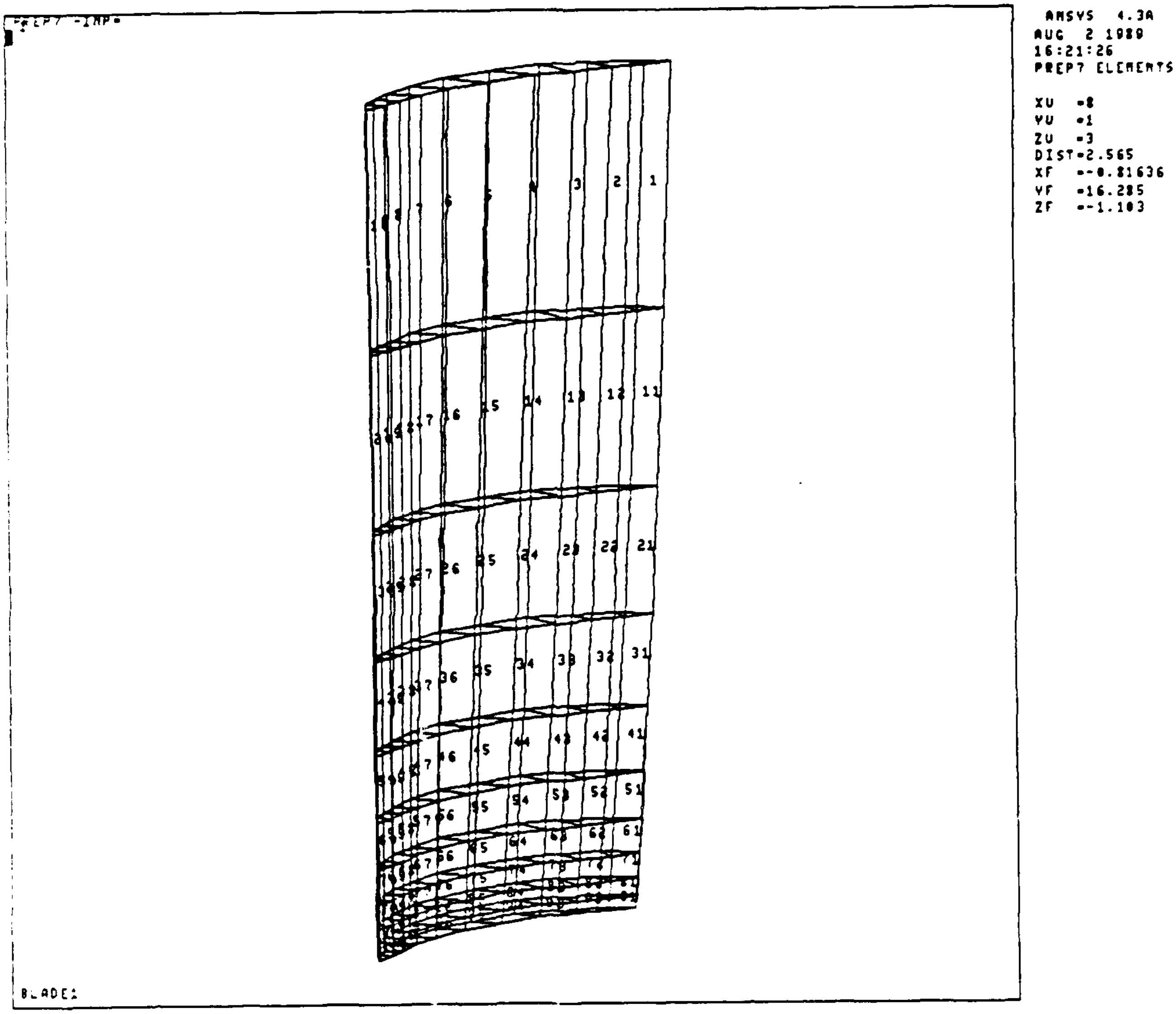


Fig. 5b ANSYS mesh for Stage 1 Stator

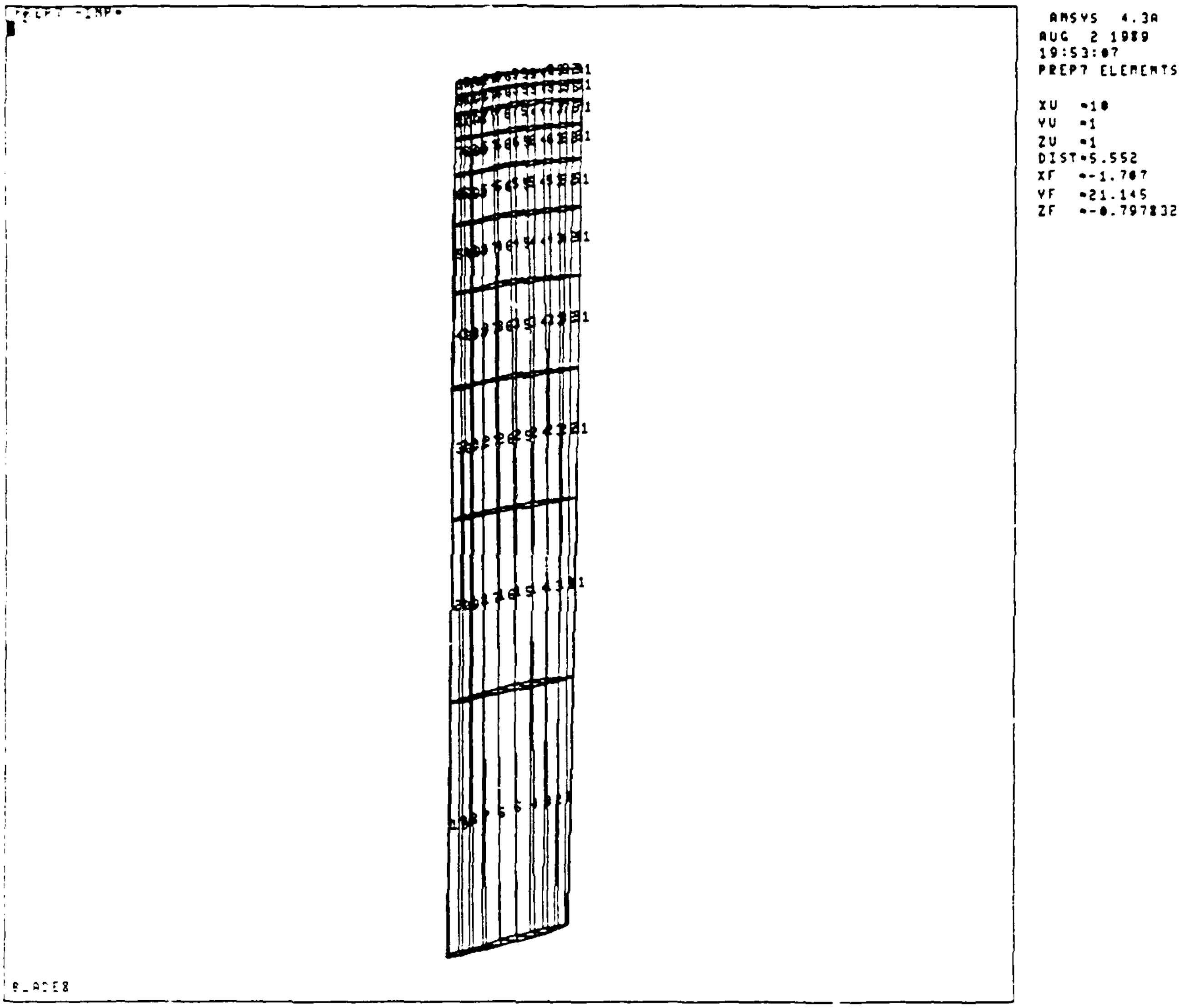


Fig. 5c ANSYS mesh for Stage 4 Rotor

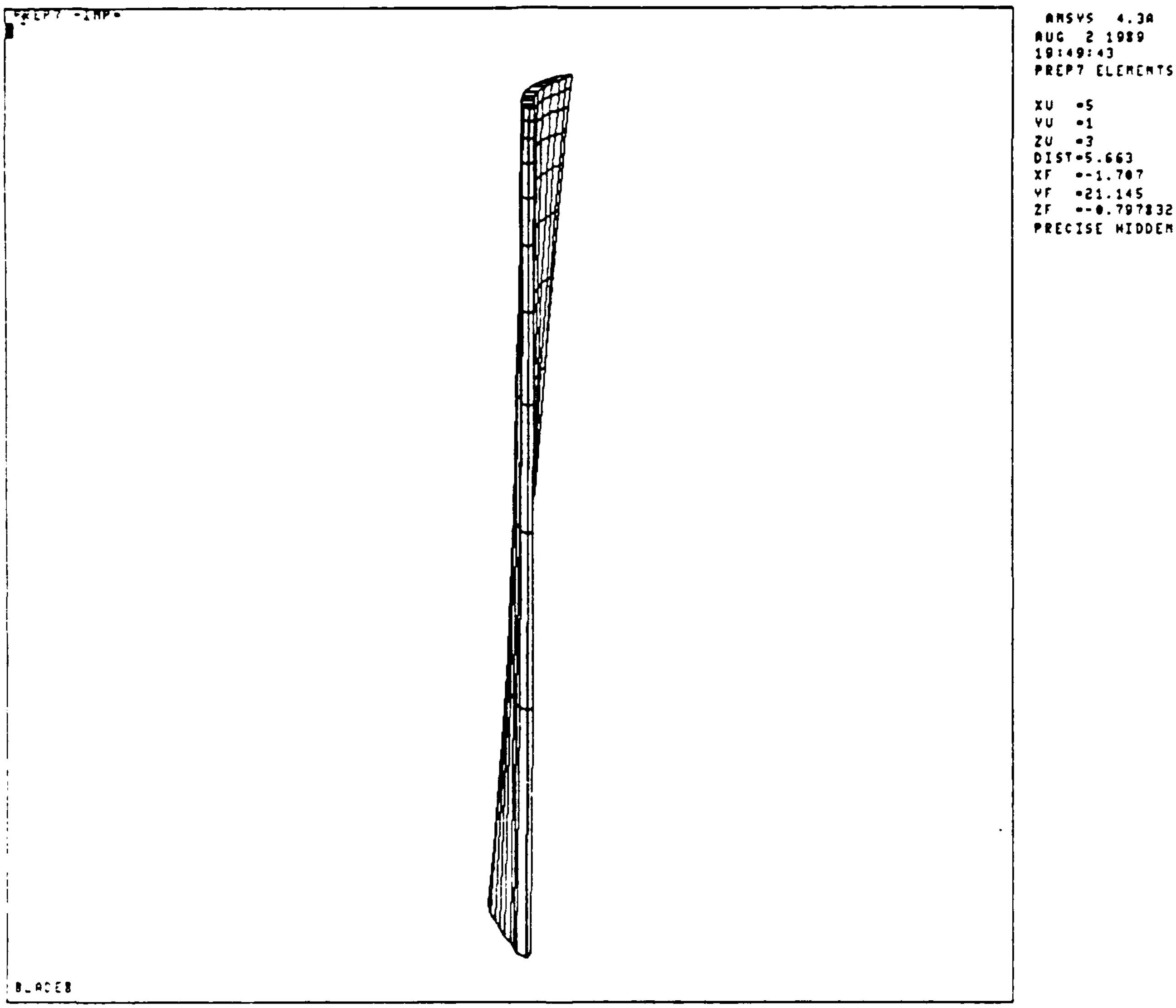


Fig. 5d ANSYS mesh for Stage 4 Rotor

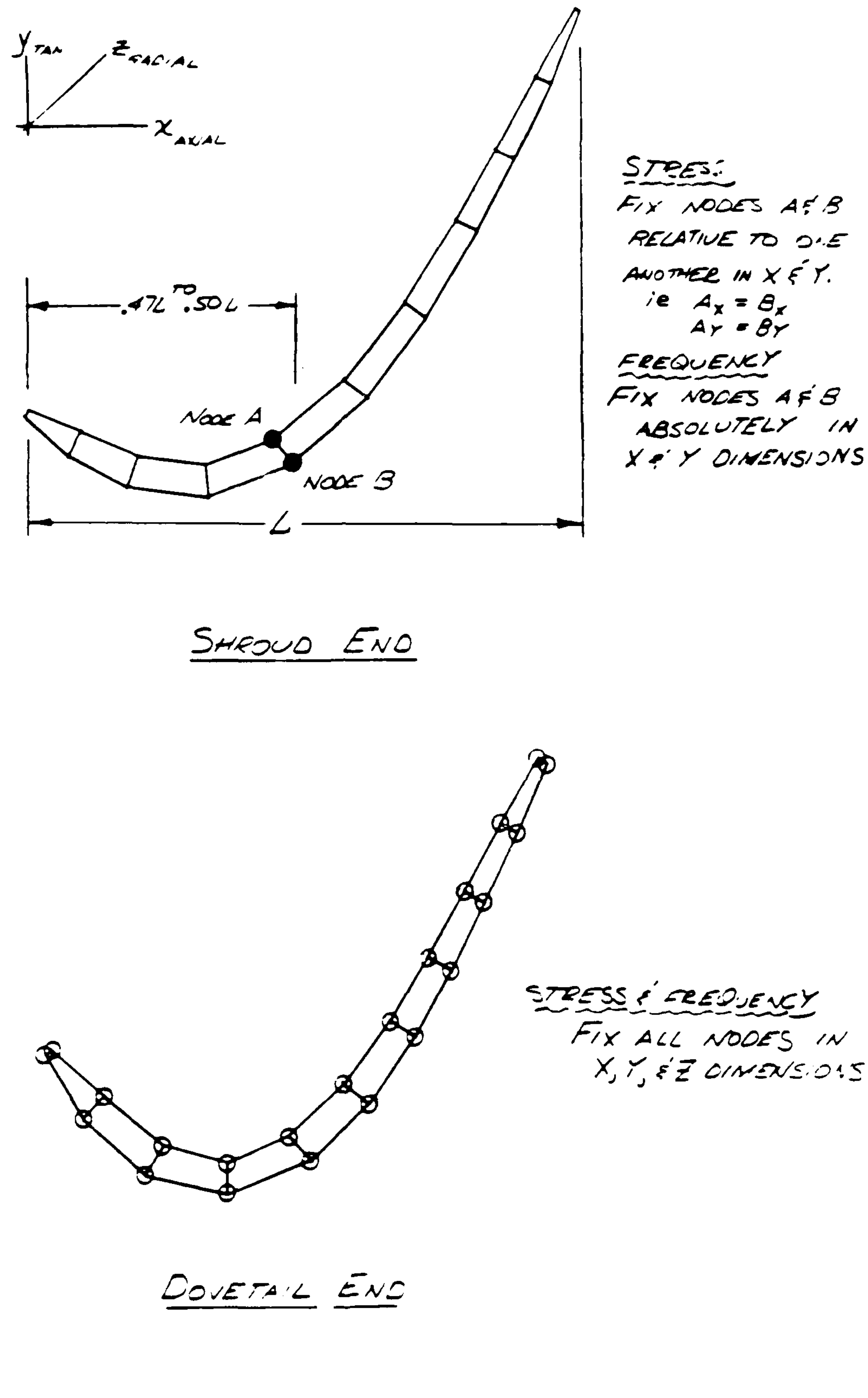


Fig. 6 Boundary Conditions on the rotors

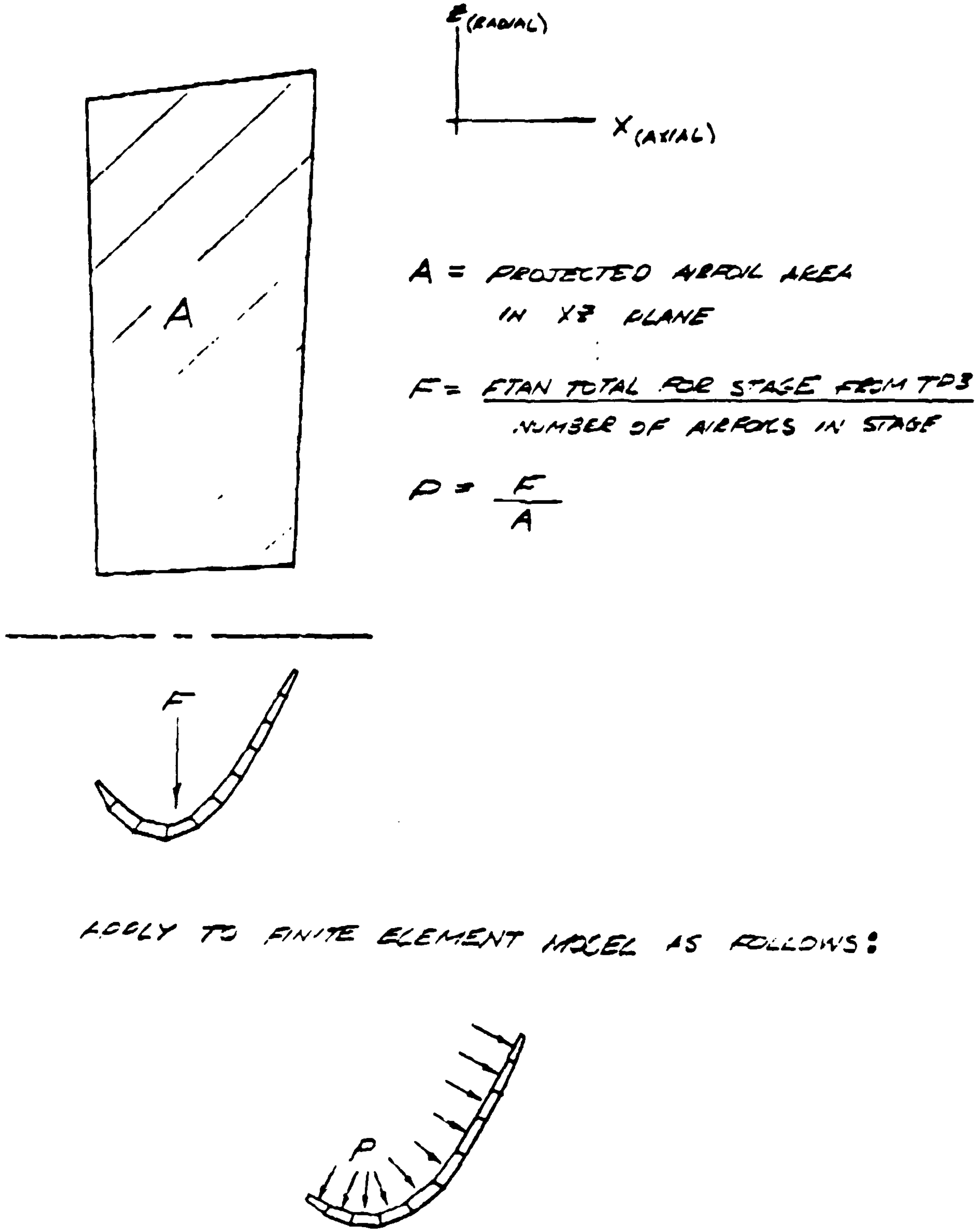


Fig. 7 Pressure Loading on Blade Row

Fig. 8

CAMPBELL DIAGRAM FOR VIBRATIONAL CONSTRAINT

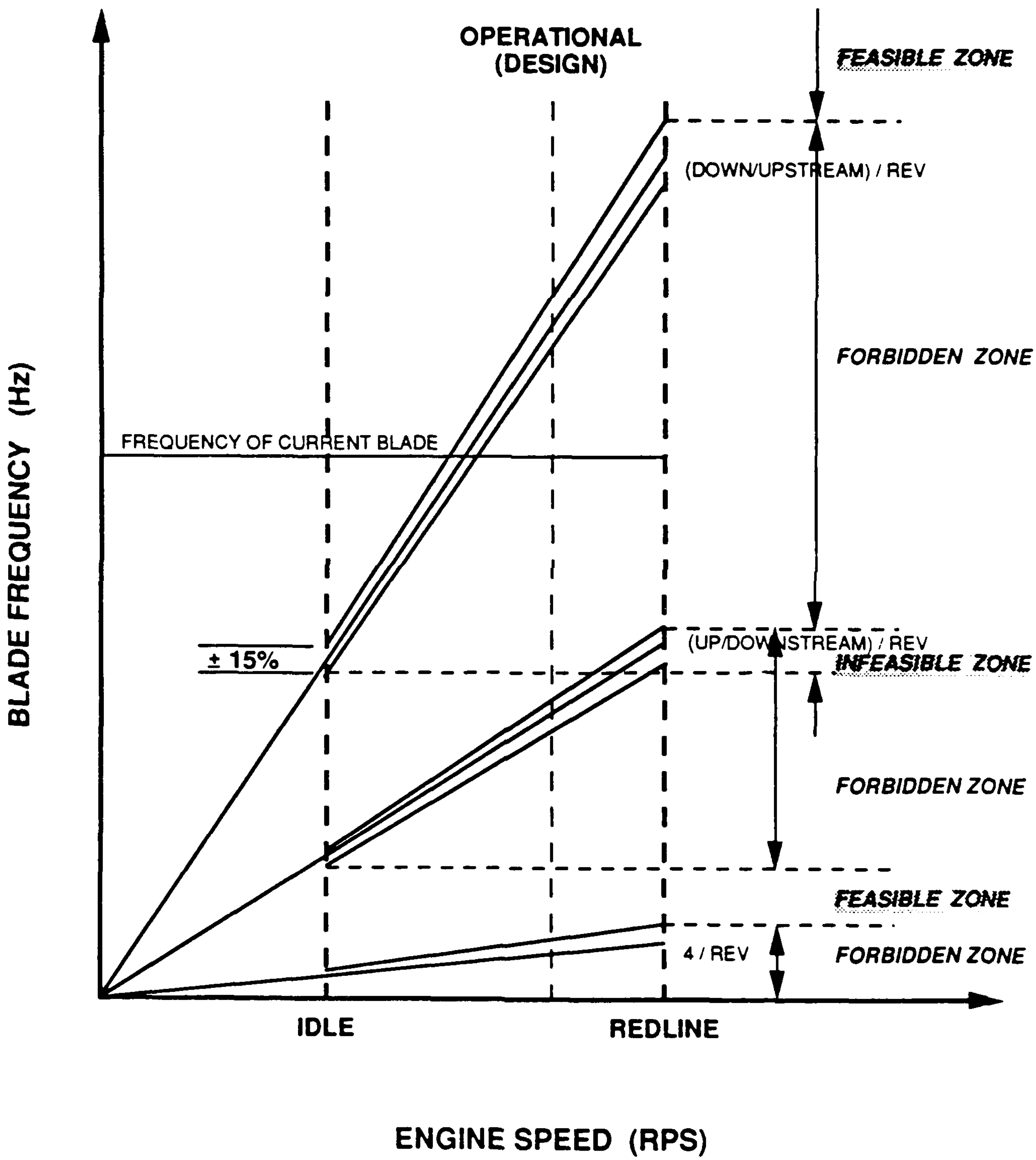
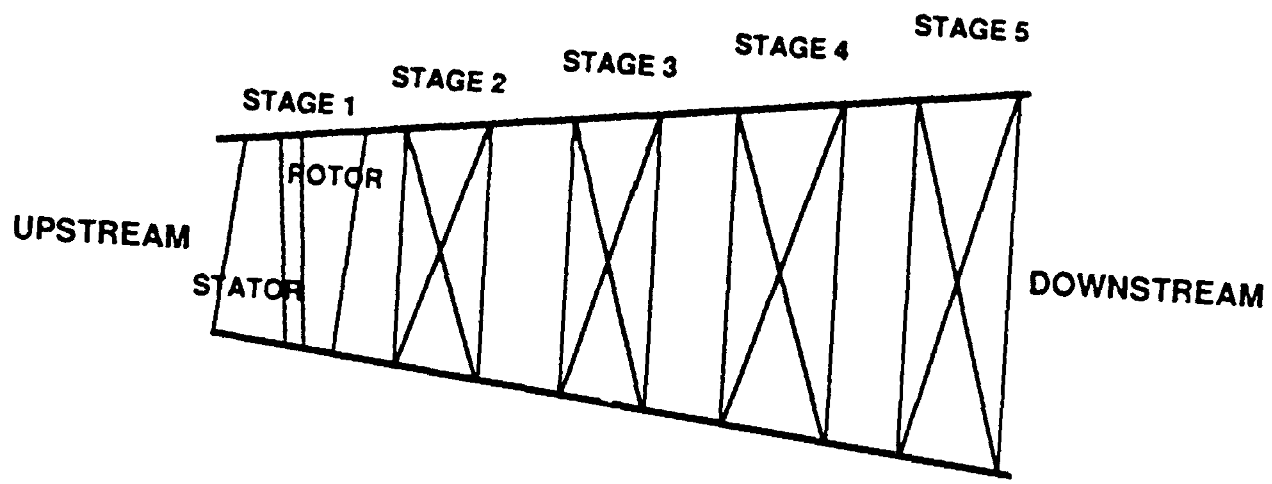
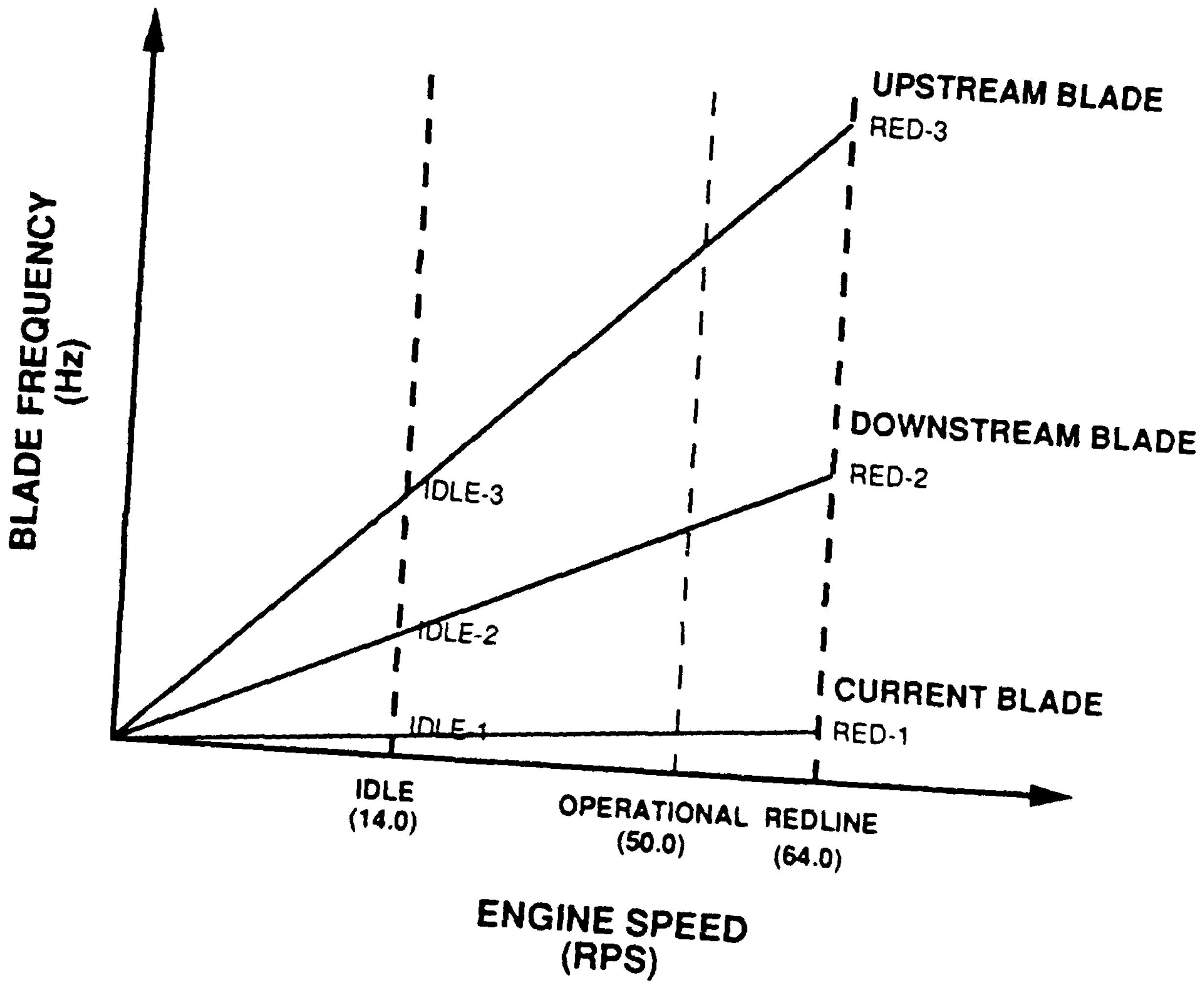


Fig. 9

IMPLEMENTATION OF VIBRATIONAL CONSTRAINT





DICE PROGRAM

Task 3.4.9.2 Optimization of Composites GE-CRD

OBJECTIVES

The objective of this effort was to develop a design methodology and a software system for structural optimization of fiber-reinforced laminated composites. Specifically the following items were addressed: (i) a state-of-the-art literature review, (ii) development and demonstration of a composites optimization software and (iii) integration of the present work with micromechanical models of composites developed in a complementary Task 3.4.5.2. The development of composites optimization capabilities was based on a structural optimization software DESIGN-OPT which is being developed at the GE R&D Center for the past few years and has been successfully applied to a variety of metallic structural components including aircraft engine discs and blades. The scope of this development was presently limited to ply optimization of 2-D composite structural components. Some illustrative composites design problems were also analyzed in order to demonstrate the software developed. The present project was interfaced with the micromechanical modeling work through a laminate analysis code which essentially transforms a given composite ply schedule into a set of orthotropic material constants.

APPROACH

The overall technical approach is based upon integrating numerical optimization methods, finite element analysis, CAE tools/software engineering and mechanics of composite materials. Mathematically, a numerical optimization problem can be stated as follows:

Find: $\underline{X} = [X_1, X_2, \cdots, X_n]$

To minimize: $F(\underline{X})$

Subject to: $\underline{g}(\underline{X}) \leq 0$

$$\underline{h}(\underline{X}) = 0$$

$$\underline{X}^l \leq \underline{X} \leq \underline{X}^u$$

Where $\underline{X}$ represents the design parameters, F the objective function, $\underline{g}$ and $\underline{h}$ the inequality and equality constraints, and $\underline{X}^l$ and $\underline{X}^u$ the lower and upper bounds on design variables. The mathematical problem stated above can be solved by using a number of available numerical methods [1] and software packages. In the present work, a state-of-the-art computer code COPES/ADS [2,3] was employed for the purposes of numerical optimization.

When using numerical optimization methods as a framework for composites design, the ply thicknesses, angles and layout are treated as design parameters. The objective function typically involves minimizing the structural weight, maximizing stiffness or strength, or maximizing frequency or stability margin. Design constraints are usually imposed on stresses, dynamic response including natural frequencies, buckling behavior, strength and stiffness, and structural failure including fiber delamination, fracture and fatigue. Thus, an example optimization formulation of a composites design problem can be expressed as follows:

Design Variables: Ply thicknesses, angles and layout

Objective Function: Minimize weight

Constraints: Stresses

Frequencies

Stiffness

Strength

The present work employs the finite element code ANSYS for structural analysis through the use of a laminate analysis code AC3. This is illustrated in Figure 1a. In essence, given the ply thicknesses, angles and schedule, the laminate analysis Code AC3 transforms a nonhomogeneous composite laminate into a homogeneous continuum with orthotropic material constants. The orthotropic material model in ANSYS is then used to carry out the stress, frequency and/or stability analysis. Relative to the role of CAE tools, a geometry-based approach is employed for formulating the problem and specifying design constraints (see Figure 1b). In this, a geometric modeler consisting of lines, arcs and splines is used for shape or boundary description, an automatic mesh generator for creating the finite element model and an attribute specification code for specifying boundary conditions at the geometry level. This eliminates the tedious and time-consuming process of specifying analysis and optimization related quantities at individual nodes and elements.

TECHNICAL RESULTS

The progress that was made on the present project can be grouped in the following categories: (i) development and application of a composite optimization software, (ii) integration with on-going work on micromechanical models for composites and (iii) literature review on composites optimization. These items are described in some detail in the following paragraphs.

Composites Optimization Software

A software package was developed for ply optimization of 2-D composite structural components using the GE CR&D design optimization DESIGN-OPT as the basis. A schematic of the software architecture is illustrated in Figure 2. Numerical optimization is carried out using a public domain/commercial computer code COPES/ADS [2,3]. It consists of a variety of commonly employed optimization algorithms including the methods of feasible direction, sequential unconstrained minimization technique and sequential linear programming. A commercial finite element code ANSYS was employed for composite structural analysis. Attention was presently restricted to 2-D components involving plane stress, plane strain or axisymmetric conditions, but both cases of stress and frequency analyses were considered. In Figure 2, OPT-AN and AN-OPT represent software modules for data interchange between the numerical optimization code ADS and the finite element analysis code ANSYS. The OPT-AN module controls the flow of the data from the optimizer to the analyzer, i.e., it maps the design parameters (ply thickness and angles) onto the orthotropic material properties part of the ANSYS input file through an intermediate computer code AC3. OPT-AN also interfaces with another module SHAPE-OPT which essentially integrates various CAE tools like geometric modeling, automatic meshing and geometry-based attribute specification. Within the context of the present development, the significance of *SHAPE-OPT lies in that it* allows the user to specify the objective function and constraints at the geometry level rather than at specific finite elements and nodes. The AN-OPT module transmits the structural analysis results of ANSYS to the optimizer through an intermediate universal format binary file BOF. The AN-OPT module and the BOF file are also interfaced with various post-processing software packages (SUPERTAB, MOVIE.BYU, PLOT10) so that the user can obtain an interactive graphics display of the stress contours, finite element model and optimization iteration histories of design variables, objective function and design constraints.

As indicated earlier, the present work was interfaced with the micromechanical modeling effort through a laminate analysis code AC3. It serves two purposes; first, given thicknesses, angles and material properties of various plies and their layout, it computes equivalent orthotropic material constants of a homogeneous medium which are subsequently used by the ANSYS code. Secondly, it calculates allowable strength values which are utilized by the ANOPT module in determining strength-related design constraints.

Software Applications to Composites Design

As an application of the software developed, we consider the ply optimization of a laminated composite plate subjected to in-plane loading as depicted in Figure 3. The laminate configuration or the ply layout is shown in Figure 3a. A plane stress condition is assumed for the ANSYS analysis. Three different cases of applied loading are considered: uniform tension in the x-direction (Figure 3b), uniform tension in the y-direction (Figure 3c) and biaxial tension (Figure 3d). Assuming the laminate to be balanced and symmetric and given the ply angles, the optimization problem is formulated as one of determining the ply thicknesses which would minimize the plate weight subjected to strength and stiffness constraints. This type of composites design problem has also been studied previously by several other researchers [4-6]. The results obtained are presented in Table 1. Since the applied loading in case (a) is along the x-direction, it is expected that the optimal design will primarily consist of 0° plies, i.e., plies having fibers along the x-axis. Consequently, we find from Table 1 that the thicknesses of 30°, 60° and 90° plies corresponding to the optimal design are very small compared to the 0° plies. For the same reason, the optimal thicknesses of 0°, 30° and 60° plies are found to be negligibly small compared to the 90° plies. For the biaxial loading case, however, all the plies are found to make relatively similar contributions to the optimal plate thickness. Since the applied stresses in x- and y-directions are equal in this case, the 0° ply is determined to have the same thickness as the 90° ply. Similarly, 30° and 60° plies are calculated to have almost the same optimal thicknesses.

As a variation of the problem shown in Figure 3d, the design parameters were changed from ply thicknesses to ply angles. In other words, the optimal thickness distribution from Table 1 was assumed to be fixed, and ply angles were permitted to vary during the optimization iterations. It is found (Table 2) that the resultant optimal ply angles are 29° and 61°, the ply angles for which the given thickness distribution was initially found to be optimal. As a final variation of this problem, the thickness distribution was assumed to be given from Table 1, the ply angles were chosen to be the design parameters, and the objective function was formulated as the maximization of the x-direction stiffness, and the design constraints consisted of strength constraints on x- and y-direction stresses. Results of this calculation are presented in the last two columns of Table 2; it is found that the ply angles which correspond to initial violation of strength constraints change significantly during the optimization iterations until the strength constraints are satisfied and the stiffness is increased by about 25%.

We next examine the effect of a hole on the optimal design of a composite plate (Figures 3e and 3f). Given the ply angles, the optimization formulation consists of ply thicknesses as the design variables and minimization of weight as the objective function with stiffness and strength limits as the design constraints. Both uniaxial ($\lambda = 0$) and biaxial loading conditions ($\lambda = 0.5$) are considered. Two different cases of initial thickness distributions are investigated. Table 3 presents the results for various cases analyzed. Several observations are made. First, because of stress concentration due to the hole the optimal plate weight is greater than the plate weight without a hole. Secondly, the increase in the optimal plate weight because of the hole is greater for the biaxial loading as compared to the uniaxial case. Finally, both starting values for ply thicknesses lead to essentially the same optimal results, particularly for the uniaxial loading case.

Literature Review on Composites Optimization

This part of the present project deals with performing a literature review on composites optimization and developing some generic optimization formulations in terms of design parameters, objective function(s) or performance measure(s) and design constraints. A large number of papers, reports and abstracts were compiled on design optimization of various types of composite structural components like shafts, plates, shells, 2D/3D solids, plates with holes, and turbine discs and blades. Computer programs that have been developed on the subject have also been reviewed to some extent including artificial intelligence/expert system applications to composites design. The design parameters in most of these problems are usually taken to be the ply angles, ply thicknesses and ply layout; the objective function being the weight, strength, stiffness or some other design criterion. Design constraints are usually placed on stresses, frequencies, buckling behavior, stability margin and impact resistance. This work is being continued into Phase II of the DICE program.

CONCLUSIONS

Several conclusions can be reached from the study described in the preceding section. First, the approach of transforming the laminated composite into an homogeneous anisotropic continuum through the use of a laminate analysis and then using the resultant material constants in a finite element analysis appears to work quite well from the structural optimization viewpoint. We note that this approach is different from using a laminated composite shell element. Secondly, optimization with respect to ply thicknesses was found to exhibit better convergence characteristics as compared to the formulation with ply angles as the design parameters. Third, interchanging the objective function with one of the constraints, i.e., maximizing stiffness with weight constraint as opposed to minimizing weight with stiffness constraint, gives different numerical behavior both in terms of optimization history and the optimal results. This observation supports the final conclusion of this study that having a well-posed optimization formulation is crucial for composites design problem, an issue which is being addressed in the work related to literature review on composites optimization.

RECOMMENDATIONS

The present software development and applications has focused on ply optimization (i.e., ply thicknesses and angles) on 2-D (plane stress, plane strain and axisymmetric) composite structural components. A major extension of this capability would be to include the shape of the component as the design parameters so that both the structural shape and the ply schedule can be changed by the optimizer in order to achieve a better optimal design. The geometry-based shape optimization approach developed in Reference [7] appears to be well-suited for this development. Simultaneous to this, it would also be important to utilize a laminated plate/shell element for finite element analysis rather than a homogeneous orthotropic material model derived from the AC3 code. Also, adding a new capability which will enable shape/ply optimization of composite shells would be of great practical significance. Some of these developments are being carried out in the DICE Phase II project.

PUBLICATIONS

None

SOFTWARE LIST

1. COPES/ADS - Commercial software package available from Engineering Design Optimization, Inc., Santa Barbara, CA.

2. ANSYS - Commercial finite element code available from Swanson Analysis Systems, Pittsburgh, PA.

3. SHAPE_OPT - A software module integrating various CAE codes, developed under GE funds.

4. DESIGN-OPT - A structural optimization software developed under GE funds.

5. COMP-OPT - Composites related enhancements of DESIGN-OPT developed under DICE funds.

REFERENCES

[1] Vanderplaats, G. N., *Numerical Optimization Techniques for Engineering Design: With Applications,* McGraw-Hill, 1984.

[2] Vanderplaats, G. N. and Sugimoto, H., "A General Purpose Optimization Program for Engineering Design," Int. J. Comp. Struct., Vol. 24, No. 1, 1986.

[3] Vanderplaats, G. N., "COPES/ADS-A FORTRAN Control Program for Engineering Synthesis Using the ADS Optimization Program: User's Manual," Engineering Design Optimization, Inc., June 1985.

[4] Schmit, L. A., Jr. and Farshi, B., "Optimum Laminate Design for Strength and Stiffness," International Journal for Numerical Methods in Engineering, Vol. 7, 1973, pp. 519-536.

[5] Schmit, L. A., Jr. and Farshi, B., "Optimum Design of Laminated Fibre Composite Plates," International Journal for Numerical Methods, Vol. 11, 1977, pp. 623-640.

[6] Dhir, S. K., "Optimization of Openings in Plates Under Plane Stress," AIAA Journal, Vol. 21, No. 10, Oct. 1983, pp. 1444-1447.

[7] Kumar, V., German, M. D. and Lee, S.-J., "A Geometry-Based 2-Dimensional Shape Optimization Methodology and A Software System with Applications," Third Int. CARS/FOF Conf. Proceed., Southfield, MI, Aug. 1988.

Table 1. Results from the ply-thickness optimization of a composite plate.

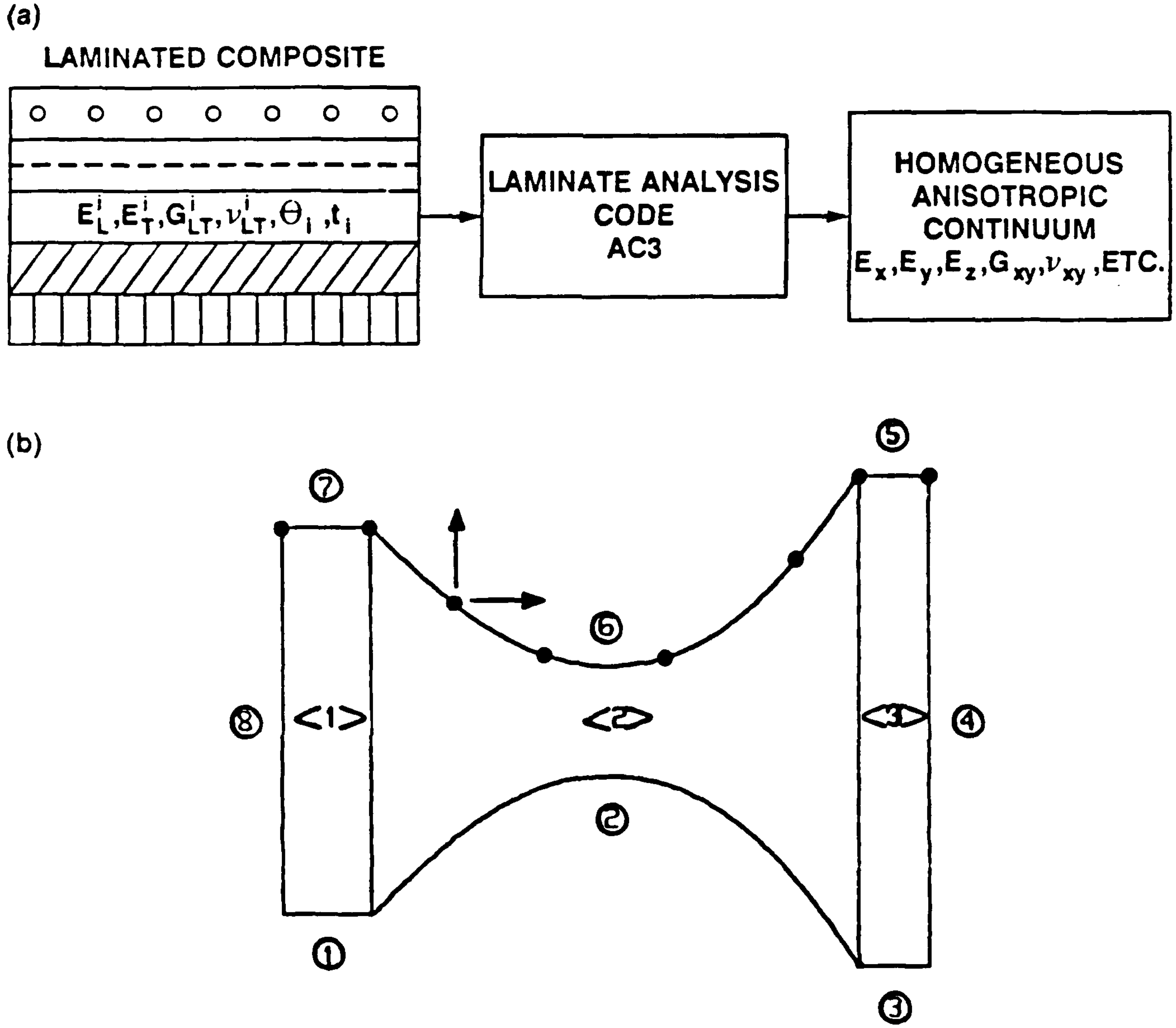
	Initial Values	Optimal Values for Figure 3b	Optimal Values for Figure 3c	Optimal Values for Figure 3d
t_1 (0°)	0.01417	0.099	0.01	0.0727
t_2 (30°)	0.01417	0.01	0.01	0.0498
t_4 (60°)	0.01417	0.01	0.01	0.0517
t_6 (90°)	0.01417	0.01	0.099	0.0712
W	1.44	0.864	0.864	2.004

Table 2. Results from the ply-angle optimization of a composite plate.

	Weight Minimization		Stiffness Maximization	
	Initial Values (Infeasible)	Optimal Values (Feasible)	Initial Values (Infeasible)	Optimal Values (Feasible)
Θ_1	0°	0°	0°	0°
Θ_2	20°	29°	25°	8.4°
Θ_4	50°	61°	50°	80°
Θ_6	90°	90°	90°	90°

Table 3. Ply-thickness optimization of a plate with and without a hole.

Case → Obj.Funct. & Des Var. ↓	Uniaxial (Fig. 3e)	Biaxial (Fig. 3e)	Uniaxial (Fig. 3f)	Biaxial (Fig. 3f)	Uniaxial (Fig. 3e)	Biaxial (Fig. 3e)	Uniaxial (Fig. 3f)	Biaxial (Fig. 3f)
	$t_i^0 = 0.417, W^0 = 1.44$				$t_i^0 = 0.2, W^0 = 6.87$			
W	1.85	2.004	3.29	6.28	1.80	2.009	3.38	6.37
t_1	0.1000	0.0727	0.1990	0.200	0.100	0.0518	0.200	0.200
t_2	0.0911	0.0498	0.1710	0.200	0.0905	0.0605	0.180	0.173
t_4	0.0110	0.0517	0.0100	0.152	0.0103	0.0623	0.010	0.186
t_6	0.0163	0.0712	0.0167	0.200	0.0100	0.0502	0.0132	0.200



- SHAPE DESCRIPTION - LINES, ARCS, SPLINES
- DESIGN VARIABLES - X, Y COORDINATES OF DESIGN POINTS
- DESIGN CONSTRAINTS - TIED TO GEOMETRY

AVERAGE HOOP STRESS ON CURVE (4) ≤ ---

HOOP STRESS IN ZONE <3> ≤ ---

MAXIMUM EFFECTIVE STRESS ON BOUNDARY EXCLUDING (4) ≤ ---

Figure 1 (a) The AC3 laminate analysis code, (b) the geometry-based approach for specifying the optimization problem.

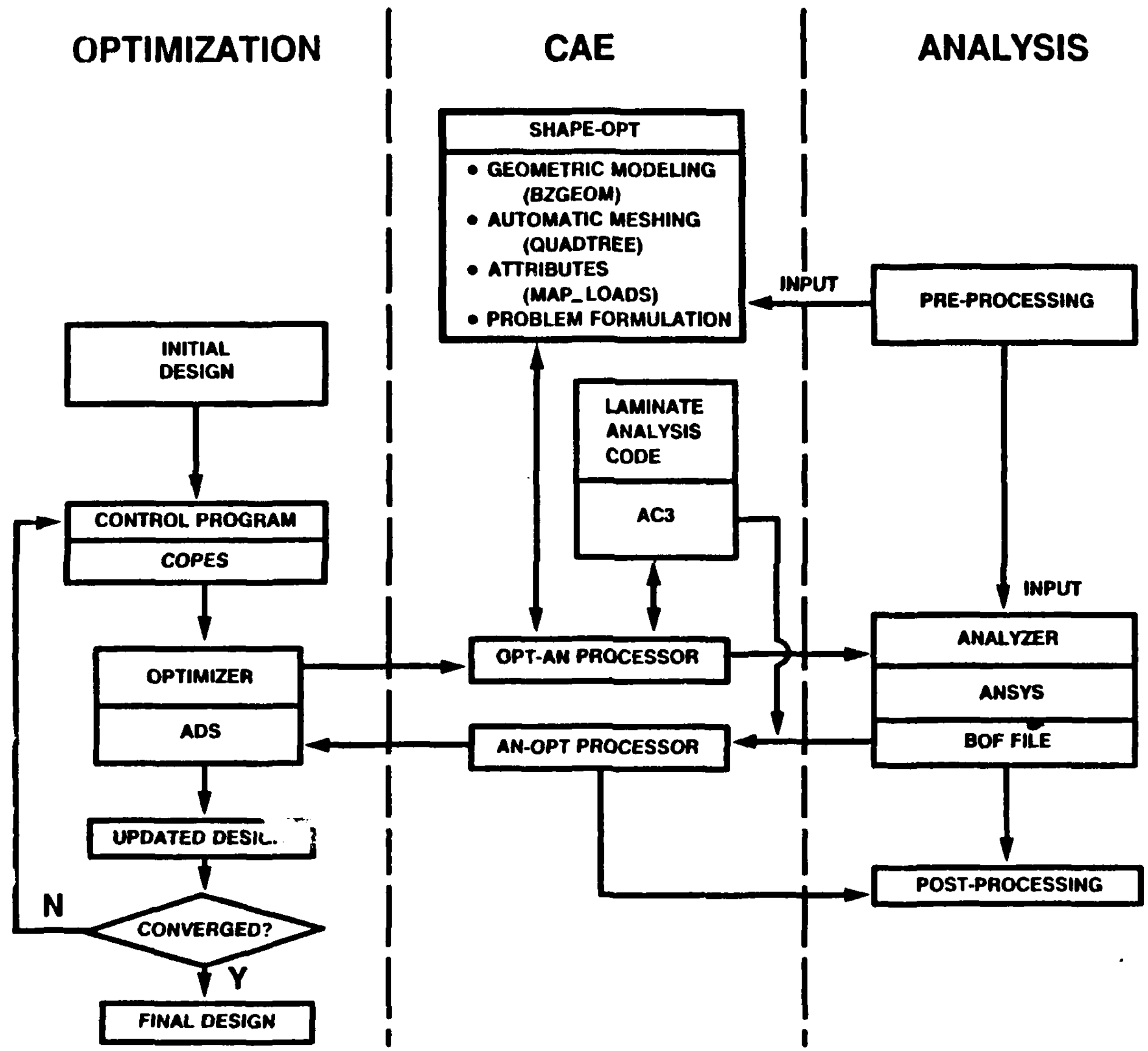


FIGURE 2. Schematic illustration of the DESIGN-OPT software architecture.

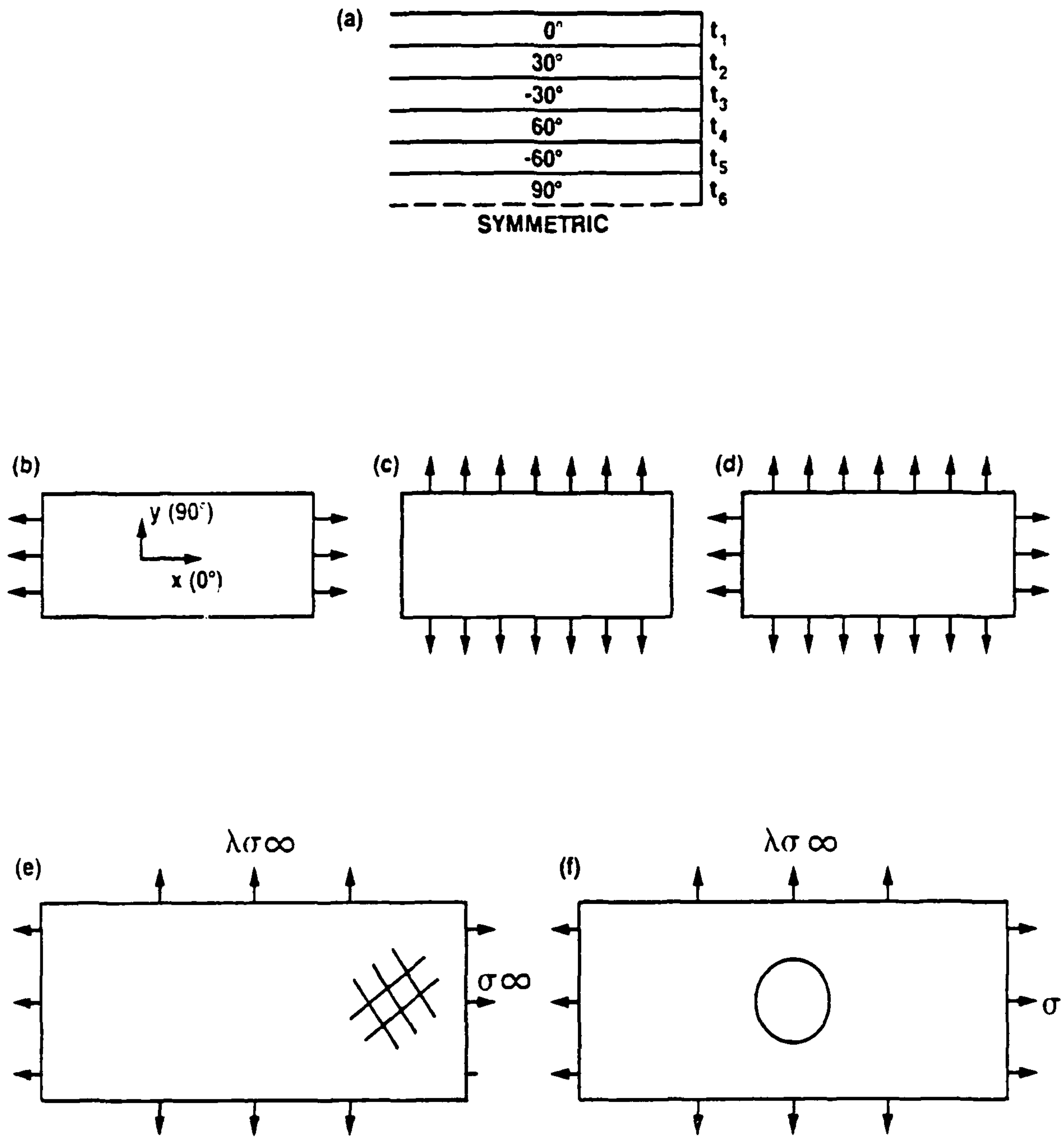


Figure 3. Schematic of a composite plate with and without a hole.

DEMONSTRATIONS OF CONCURRENT ENGINEERING

DICE PROGRAM

Task 3.5.4.1 Concurrent Manufacture of XD™ Airfoils — Howmet Corporation

OBJECTIVE:

The objective of this task was to demonstrate application of concurrent engineering principles to fabricate an advanced gas turbine engine component in a production manufacturing environment. Manufacturing processes required to produce a state-of-the-art aircraft gas turbine blade in an advanced high temperature, high strength composite material had to be established simultaneously with alloy synthesis. Through increased interdisciplinary communication among component manufacturing divisions as well as among the engine manufacturer, the alloy designer and component manufacturer, the time from alloy and component selection to delivery of the machined turbine blades was targeted for thirteen months.

Key manufacturing processes and materials, previously demonstrated on a laboratory basis, had to be scaled-up and refined to produce blueprint quality CF6-80C2 5th stage turbine blades cast in an advanced XD™ titanium aluminide alloy. Technical challenges in manufacturing quality XD™ titanium aluminide ingots of large size for use in a production casting furnace as well as challenges in vacuum arc remelt (VAR) casting, post-cast thermal processing and machining of the turbine blade castings had to be solved to achieve the task objectives.

To assure quality of the castings, ingots weighing 250 pounds (114 kg) and meeting elemental chemistry specifications with measured oxygen contents less than 1000 ppm were required. Prior to initiating this program, the largest ingot produced was 40 pounds (18 kg) with typical measured oxygen contents of 1200 ppm.

Establishing the VAR investment casting process parameters and the post-cast processing parameters and techniques best suited for manufacture of long thin turbine blade components presented challenges in wax pattern design and gating as well as fixturing and handling of the cast components. Components cast earlier in XD™ titanium aluminide alloys typically exhibited lower length-to-width aspect ratios.

Prior to this program, machining of XD™ titanium aluminide alloys was limited to test bar configurations. Establishing tooling and processing techniques both for electrochemical machining and grind machining of the more complex turbine blade presented additional challenges.

APPROACH:

Program participants included seven divisions within Howmet Corporation, four manufacturing divisions and three divisions within the research center, and five commercial subcontractors. In addition, program coordination with General Electric (GE), the engine manufacturer and prime contractor, and Martin Marietta Laboratories (MML), the alloy designer, was required.

To assure concurrence in conducting the program, several basic principles of concurrent engineering were employed:

- A team, comprised of members representing key program participants was identified and distinct responsibilities and decision making authority assigned.

- Direct communication channels between team members working on interdependent efforts were identified.

- Responsibility for problem anticipation and resolution was assigned all team members: both for problems in their specific business area as well as the overall program.

- Interdependent material and processing options were left open as long as possible to allow rapid response to unavoidable changes and to assure efforts were conducted the right way the first time.

- Independent efforts were conducted concurrently wherever practical.

The program was structured with two major tasks. In the first task, the alloy composition and specific turbine blade to be produced were identified and technical effort conducted to define the respective process parameters. In the second task, the deliverable test material and finish machined turbine blades were produced and mechanical property evaluations conducted.

Task One: Manufacturing Processes Definition

A gamma titanium aluminide matrix (Ti-48a%Al-2a%V) reinforced with 7.5 volume % TiB_2 particulate was identified as the initial alloy composition. The precursor material containing TiB_2 particulate was produced by AMAX Corporation using the MML patented XD™ process. Based on results from Howmet's internally funded research program, low oxygen precursor material was specified.

Ingot production was sequenced to allow the option of introducing a second alloy composition for manufacture of the deliverable test material and finish machined turbine blades in the second task. This option was not chosen. The initial composition was used throughout the program. Nine alloy ingots, each measuring approximately 8 inches (20 cm) diameter and 38 inches (97 cm) long and weighing approximately 250 pounds (114 kg) were produced by Timet Corporation.

A series of preliminary casting trials were conducted to verify fluidity of the XD™ titanium aluminide alloy. The chemistry, microstructure and chemical milling response of the cast alloy produced from large ingots was compared to that previously seen in laboratory scale programs. As-cast material from the preliminary trials also was used to conduct initial electrochemical and grind machinability evaluations.

A second set of cast test material was produced and used to define post-cast processing parameters and to conduct preliminary mechanical property tests. Processing parameters were defined for chemical milling, hot isostatic pressing (HIP) and heat treatment. HIP temperatures of 2200°F (1205°C), 2250°F (1232°C), and 2300°F (1260°C) at pressures of 15 ksi (103.5 MPa) and 25 ksi (172.5 MPa) were evaluated. A heat treatment of 1650°F (888°C) for 16 hours was specified. Additional ECM and grind machining evaluations were conducted on fully processed material.

The final casting series conducted in the first task produced "pseudo" blades: oversize turbine blade castings. Production of oversize blades was required to assure complete fill of the long thin airfoil section of the blade when VAR casting the XD™ titanium aluminide alloy. The amount of oversizing was influenced by the fluidity of the alloy and the starting material envelope required for ECM of the airfoil gas path flow surfaces.

Initial efforts were directed to use of a GE-36 prototype blade both as a basis for the pseudo blades and as the finish machined blade. Use of the GE-36 prototype blade wax patterns proved unacceptable and efforts were redirected to existing production blade configurations. A production wax injection tool for the CF6-80C1 4th stage blade was available for use and was selected as the basis for the pseudo blade configuration. Subsequently, the use of wax patterns produced from the production 4th stage blade wax injection tool was determined to be impractical. A low cost, temporary wax injection tool was built to produce oversize wax patterns based on the CF6-80C1 4th stage blade. The CF6-80C2 5th stage blade was selected as the deliverable machined blade configuration. This allowed use of available grind machining tooling in the second task.

Unanticipated problems were encountered early in the first task with ingot quality and ingot alloy utilization. This required a reduction in the number of molds cast to establish casting process parameters. Four molds of the oversize 4th stage blades were cast to validate the use of process parameters defined by the test material evaluations. The oversize blades were HIPed, chemically milled, heat treated and inspected to verify that quality oversize blade castings could be produced in the selected XD™ alloy composition.

Task Two: Production of Deliverable Test Bars, Pseudo Blades and Finish Machined Blades

The program plan for the second task called for separately casting a series of test bar molds, a series of deliverable oversize CF6-80C1 4th stage blade molds and a series of oversize CF6-80C2 5th stage blade molds. Due to the problems with ingot quality and ingot alloy utilization, the second task was redirected in order that overall technical objectives be met and the required number of deliverables be produced. A single mold setup, containing both test bar material and oversize CF6-80C2 5th stage blades, was designed. The oversize CF6-80C2 5th stage blades were used both as the deliverable "pseudo" blades and the blades for subsequent machining.

Based on experience gained when casting oversize CF6-80C1 4th stage blades in the first task, a wax pattern injection tool to produce patterns for the oversize CF6-80C2 5th stage blades was procured. To further define casting parameters, three oversize 5th stage blade molds were cast. The variables of static versus centrifugal casting at various mold preheats were further investigated using alternate gating techniques. The blades were processed through HIP, chemical milling, heat treatment and X-ray, fluorescent penetrant and dimensional inspections. As a result of these evaluations, the mold setup and casting parameters used to produce the deliverable oversize (pseudo) blades and the oversize blades for machining were established.

Twenty two molds of oversize CF6-80C2 5th stage blades, a total of 96 blades, were cast. All blades were processed through HIP, heat treatment, and X-ray, fluorescent penetrant and dimensional inspections. After review of the inspection results, the 80 blades that most closely met the quality requirements were submitted for ECM processing.

Tooling and fixtures required to ECM and hand blend the airfoil gas path flow surfaces of the blades were procured. The airfoil gas path flow surfaces were ECMed and hand blended to attain blueprint dimensions typical of the airfoil section of production CF6-80C2 5th stage blades. After dimensional inspection, the 40 blades with ECMed airfoil sections most closely meeting the blueprint dimensions were submitted for grind machining of the root and tip shroud details.

Tooling and fixtures used in production to grind the root and tip shroud details of CF6-80C2 5th stage blades were used in this task. Additional fixtures were required to machine oversize surfaces that typically are cast-to-size features on the production blade. The machined details of each blade were dimensionally inspected after the respective grind machining operations. In addition, after all machining operations were complete, the airfoil sections of 4 fully machined blades were dimensionally inspected.

To meet the deliverable requirements, 30 fully machined CF6-80C2 5th stage blades and 20 oversize (pseudo) 5th stage blades were shipped to General Electric. The deliverable mechanical property test bar material and the test bar material required for mechanical property tests conducted by Howmet were obtained from the twenty-two CF6-80C2 5th stage blade molds. The test material was HIPed, heat treated and fully inspected. Fifty-five sections of test bar material were delivered to General Electric. In addition to the preliminary mechanical tests conducted in the first task, fifteen tensile test bars and nine creep rupture test bars were machined and tested by Howmet.

<u>TECHNICAL RESULTS</u>:

The metallurgical structure and chemistry of the precursor material and the ingots met program requirements. Results of ingot chemistry analyses, Table 1, exhibited little variability between ingots. No cracks or other surface defects were found during visual inspection of ingot surfaces. The ingots and typical microstructures of the precursor material and ingots are shown in Figure 1.

An unanticipated problem was encountered when the first ingot cracked and broke apart during VAR casting. Although not established, suspected causes of cracking were internal stresses, internal cracks, or both. After verifying the problem in the second and third ingots, subsequent ingots were HIPed at 2300°F/4h/25ksi (1260°C/4h/172.5 MPa). The HIPed ingots did not break apart during casting.

The fluidity of the XD™ titanium aluminide alloy was less than Ti-6Al-4V alloy but comparable to other advanced titanium alloys. Using standard production process parameters, 0.100 inch (0.254 cm) thick fluidity plates filled completely. Compared to Ti-6Al-4V, less reaction layer was found on the cast surface of the XD™ alloy test material. Metal removal during chemical milling was uniform. The metal removal rate of the XD™ alloy was slower than typically experienced with Ti-6Al-4V. It was established that the standard production chemical milling bath was suitable for use with the XD™ alloy.

HIP parameters of 2300°F/4h/25ksi (1260°C/4h/172.5 MPa) were selected for use on the program. While the evaluations showed acceptable closure was obtained in 0.625 inch (1.59 cm) diameter bars at lower temperatures and pressures, the higher temperature and pressure were selected to better assure closure in the cast blades. The cast microstructure exhibited a uniformly fine grain size, a uniform dispersion of TiB_2 particulate reinforcement, and no evidence of microshrinkage after HIP and heat treatment at 1650°F (888°C) for 16 hours.

The criticality of quality test bar machining practice was highlighted when an initial set of test bars, machined at a source not previously used by Howmet, exhibited virtually no ductility. These test results were established as not valid due to unacceptable machining practice and an additional set of test bars machined. The second set of test bars yielded room temperature tensile test data similar to those measured in other programs: average results for three tests were 95.1 ksi (656.2 MPa) UTS, 83.0 ksi (572.7 MPa) YS and 0.45% plastic elongation.

Machinability evaluations showed ECM and grind machining of XD™ alloy can be accomplished using production equipment and techniques. Metallographic examinations of machined coupons showed no evidence of attack or cracking in the material at the machined surface.

Although problems with breakup of the ingot was experienced when casting the four molds of oversize CF6-80C1 4th stage blades, sufficient data was obtained to demonstrate that high aspect ratio turbine blade castings can be cast using standard VAR investment casting techniques and parameters. Unacceptable fill was experienced in all molds cast with a 600°F (316°C) preheat. Acceptable fill was demonstrated in molds preheated to 1200°F (649°C) and static cast, but heavy centerline shrink was present throughout the airfoil and tip shroud of the blades.

The oversize CF6-80C1 4th stage blades most closely meeting the quality requirements were produced using previously established casting processes, standard Mono-Shell™ materials, and centrifugal casting with a mold preheat of 1200°F (649°C). The as-cast blades exhibited slight centerline shrink in airfoil sections, relatively heavy shrink in the tip shroud and root sections and some cracks along the airfoil. HIP successfully closed the centerline shrink in the airfoil sections but only limited success was attained in eliminating the shrink in the tip shroud and root sections. Based on these results, alternate gating techniques were identified for further evaluations conducted with the oversize CF6-80C2 5th stage blades.

Complete fill was attained in each of the three molds of oversize 5th stage blades cast using the alternate gating techniques. The overall quality of the blades cast with a 600°F (316°F) mold preheat and blades static cast with a 1200°F (649°C) mold preheat was unacceptable. The quality of the blades that were centrifugally cast with a 1200°F (649°C) mold preheat was best overall. After HIP, the CF6-80C2 5th stage blades had structurally sound airfoils and outer tip shrouds. Moderate shrink was observed in the pocket area immediately below the root platform. Further process evaluations would have established additional gating enhancements to eliminate the shrink in the pocket near the root platform, but additional evaluations were beyond the scope of the current program. Dimensional data was taken on selected blades for use in setting up subsequent molds. It was established that the oversize 5th stage blades cast for machining would be centrifugal cast using a 1200°F (649°C) mold preheat. In addition, the gating on these molds would be revised further to improve quality. Cracks were observed on 5th stage blades from the first three molds after chemical milling. The cause for cracks present after chemical milling was not established. Chemical milling, to bring oversize surfaces that are not machined into tolerance, was not conducted on subsequent blades.

Ninety-six oversize CF6-80C2 5th stage blades were selected for machining. These blades exhibited defects, including evidence of shrink after HIP, typical of that seen in the oversize blades cast earlier. The microstructure of the HIPed and heat treated blades was similar to that found earlier in the preliminary evaluations: uniformly fine grain size and a uniform dispersion of TiB_2 particulate reinforcement, Figure 2. Dimensional inspection of as-cast oversize blades showed the contour, bow and displacement to be within tolerance. The twist generally exceeded the allowed blueprint tolerance. Based on an overall quality assessment of the 96 oversize blades, 47 were rated good and 19 acceptable. These blades were submitted for machining. The remaining 30 blades were rated as acceptable for establishing machining techniques and process parameters.

Eighty oversize 5th stage blades were submitted for ECM set up evaluations and finish machining. Standard procedures (i.e., voltage, feed rates, electrolyte conditions, etc.) used to ECM conventional titanium components were used to machine the airfoil gas path flow surfaces of the oversize blades. Achieving blueprint dimensional tolerances on the airfoil required hand blending after ECM. Blueprint tolerances were met with the exception of the airfoil side of the tip shroud. Likewise, achieving the required surface finish was challenging. The as-ECMed surface finish of the XD™ alloy was rougher than that of conventional titanium alloys. The requirement for hand blending to achieve dimensional and surface finish tolerances can be reduced with additional process refinement.

Forty-nine 5th stage blades, with ECMed airfoils, were submitted for grind machining setup evaluations and final machining. The grinding process parameters established earlier were used during set up evaluations. Fluorescent penetrant inspection after the initial grinding operations on the root sections showed surface cracks present on the machined surfaces. The metal removal rate was reduced by adjusting process parameters, and cracking was avoided on subsequent blades. With the exception of the radial grind on the tip shroud seals, the tip shroud details were machined to blueprint tolerances. As a result of the decision to eliminate chemical milling of the blades to be machined, the surfaces that were not machined remained approximately 0.010 inches (0.025 cm) oversize. Due to the oversize dimensions, difficulties were encountered in fixturing the parts. This resulted in an inability to meet the blueprint tolerance on the trailing edge root surface. An oversize blade casting and a finish machined blade are shown in Figure 2.

The results of tensile tests conducted by Howmet are listed in Table 2. The lack of suitable test bar fixtures precluded conducting the planned tests at 1832°F (1000°C). The ultimate strength of the XD™ alloy remained constant at test temperatures up to 1472°F (800°C). The yield strength of the alloy decreased as the test temperature increased. The measured elongation remained relatively constant over the temperature range from 392°F to 1112°F (200°C to 600°C): average values ranged from 0.83% to 0.98% plastic elongation and 1.16% to 1.36% total elongation. The average ductility of the alloy tested at 1472°F (800°C) increased to average values of 9.26% plastic elongation and 9.68% total elongation. A typical stress-strain curve denoting the measurement technique used is shown in Figure 3.

The results of the creep rupture tests, Table 3, showed limited creep deformation in bars tested at 1112°F (600°C) until applied loads resulted in stress levels approaching the yield strength. Two of the bars tested at 1472°F (800°C) and 30 ksi (207 MPa) failed at approximately 150 hours and exhibited final elongations near 30%. The third bar tested at 1472°F (800°C) and 30 ksi (207 MPa) failed at 37.3 hours and exhibited only 12.3 % final elongation. Review of the test technique and test bar revealed no assignable cause for this low result. Limited resistance to creep deformation was measured in tests conducted at 1832°F (1000°C) and 10 ksi (69 MPa). Time to failure ranged from 10.5 hours to 17.5 hours and final elongations ranged from 39.0% to 46.6%.

CONCLUSIONS:

1. The objective of application of concurrent engineering principles to fabricate an advanced gas turbine engine component in a production environment was achieved within budget.

2. Quality XD™ titanium aluminide (Ti-48a%Al-2a%V + 7.5 volume % TiB_2) ingots weighing up to 250 pounds (114 kg) can be produced. Raw material and ingot manufacturing processes must be closely controlled to assure alloy and ingot quality. Hot isostatic pressing of large ingots prevents break up of the ingot during VAR investment casting.

3. The fluidity of the XD™ alloy is less than Ti-6Al-4V alloy but comparable to other advanced titanium alloys. The general process methods and materials used for HIP, heat treatment, and nondestructive inspection of conventional titanium alloys are suitable for processing XD™ titanium aluminide alloys. Further work is required to establish process methods for chemical milling XD™ titanium aluminide alloy.

4. High length-to-width aspect ratio components, such as a CF6-80C2 5th stage turbine blade, can be VAR investment cast in XD™ titanium aluminide alloys using process methods and materials established for conventional titanium alloys. The occurrence of defects identified in the castings are typical of those found in other cast components when conducting initial casting trials. The rate of occurrence for these defects can be reduced or eliminated with further process evaluations.

5. Electrochemical and grind machining can be used to machine the complex details of investment cast turbine blades to blueprint tolerances. Deviations from blueprint tolerances experienced on this program can be eliminated with further process evaluations.

6. The measured tensile and creep rupture properties make the XD™ alloy a viable candidate for further development as an advanced particulate reinforced gamma titanium aluminide composite material.

RECOMMENDATIONS:

1. It is recommended that an expanded manufacturing demonstration of concurrent engineering be conducted. The objective of this task would be to design, manufacture and engine test a gas turbine engine component produced in an advanced titanium aluminide material. Available computer architecture and design tools and methods, developed in the current program, would be utilized.

2. It is recommended that minimum design properties and requirements for protective coatings be established for specific applications of advanced titanium aluminide materials.

PUBLICATIONS: NONE

HARDWARE:

1. Temporary Wax Injection Die
2. Single Cavity Hinged Injection Mold (DARPA DICE 7006-A)
3. Single Cavity Chill Mold (DARPA DICE 7006-B)
4. Reforming Fixture (DARPA DICE 7006-C)
5. EDM Preparation Fixture Shroud Tip (DARPA DICE 0701-T2)
6. Guillotine Inspection Gauge
7. Copper ECM Cathode
8. Platform Machining Fixture
9. Various Test Material Coupons (Scrap)
10. Test Material (55 pieces delivered)

11. Oversize CF6-80C1 4th stage blades (32 pieces scrap)
12. Oversize CF6-80C2 5th stage blades (64 pieces scrap, 20 pieces delivered)
13. Finish Machined CF6-80C2 5th stage blades (30 pieces delivered)

Table 1

Composition of XD™ Titanium Aluminide Ingots and Investment Cast Components

Component		Chemistry (Weight %)					
		Al	V	B	Fe	O	N
-	**TARGET[1]**	**34.1**	**2.7**	**2.6**	**<0.100**	**<0.100**	**<0.050**
Ingot	Average of 9 Samples	34.0	2.6	2.6	0.076	0.060	0.008
	Range	33.4 - 34.5	2.6 - 3.0	2.1 - 2.9	0.071 - 0.087	0.058 - 0.064	0.004 - 0.014
Test Material	Average of 6 Samples	36.5	2.6	2.7	0.060	0.067	0.014
	Range	36.1 - 36.8	2.5 - 2.6	2.6 - 2.9	0.055 - 0.070	0.058 - 0.085	0.009 - 0.020
CF6-80C1 4th Stage Pseudo Blades	Average of 8 Samples	35.6	2.5	3.0	0.090	0.085	0.001
	Range	33.4 - 35.0	2.5 - 2.6	2.8 - 3.2	0.050 - 0.170	0.067 - 0.109	0.001 - 0.002
CF6-80C2 5th Stage Finished Blades	Average of 10 Samples	32.2	2.4	2.3	0.070	0.081	0.008
	Range	31.8 - 32.5	2.4 - 2.5	2.2 - 2.5	0.066 - 0.093	0.075 - 0.093	0.004 - 0.009

(1) Weight % Equivalent of Ti-48a/o Al-2a/o V + 7.5 v/o TiB_2 Alloy

Table 2

Room Temperature and Elevated Temperature Tensile Test Data
From Investment Cast XD™ Titanium Aluminide Test Bars.
(Ti-48a/o Al-2a/o V + 7.5 v/o TiB_2)

TEST TEMPERATURE	TENSILE PROPERTIES(1)					TEST FACILITY
	UTS (KSI)	YS (KSI)	TOTAL EL (%)	PLASTIC EL (%)	RA (%)	
	86.0	77.9	0.78	0.88	0.9	
22°C	90.1	83.3	0.77	0.46	0.5	Howmet Corporation
(70°F)	73.8(2)	73.8(2)	0.44(2)	0.19(2)	Radius(2)	
AVERAGE	83.3	78.3	0.66	0.38	0.7	
	90.5	73.4	1.70	1.14	1.0	
200°C	86.5	72.2	1.28	0.90	1.0	Westmoreland Test
(390°F)	85.1	73.9	1.09	0.65	1.3	Laboratories Inc
AVERAGE	87.4	73.2	1.36	0.90	1.1	
	87.2	68.7	1.24	0.95	1.0	
400°C	74.1	62.7	0.86	0.47	2.0	Westmoreland Test
(752°F)	85.8	66.5	1.38	1.06	1.0	Laboratories Inc
AVERAGE	82.4	65.9	1.16	0.83	1.3	
	89.1	65.1	1.31	0.97	0.9	
600°C	81.7	55.6	1.23	0.92	Radius	Howmet Corporation
(1112°F)	96.8	64.9	1.39	1.07	0.1	
AVERAGE	89.2	61.9	1.31	0.98	0.5	
	81.8	57.2	12.32	12.30	12.9	
800°C	82.6	56.2	8.86	7.98	7.6	Howmet Corporation
(1472°F)	88.4	62.7	7.87	7.50	5.7	
AVERAGE	84.3	58.7	9.68	9.26	7.4	

(1) Technique for Determining Total Elongation and Plastic Elongation Shown in Fig. 3.

(2) Fracture Surface Anomaly Noted.

T2-89/426.5

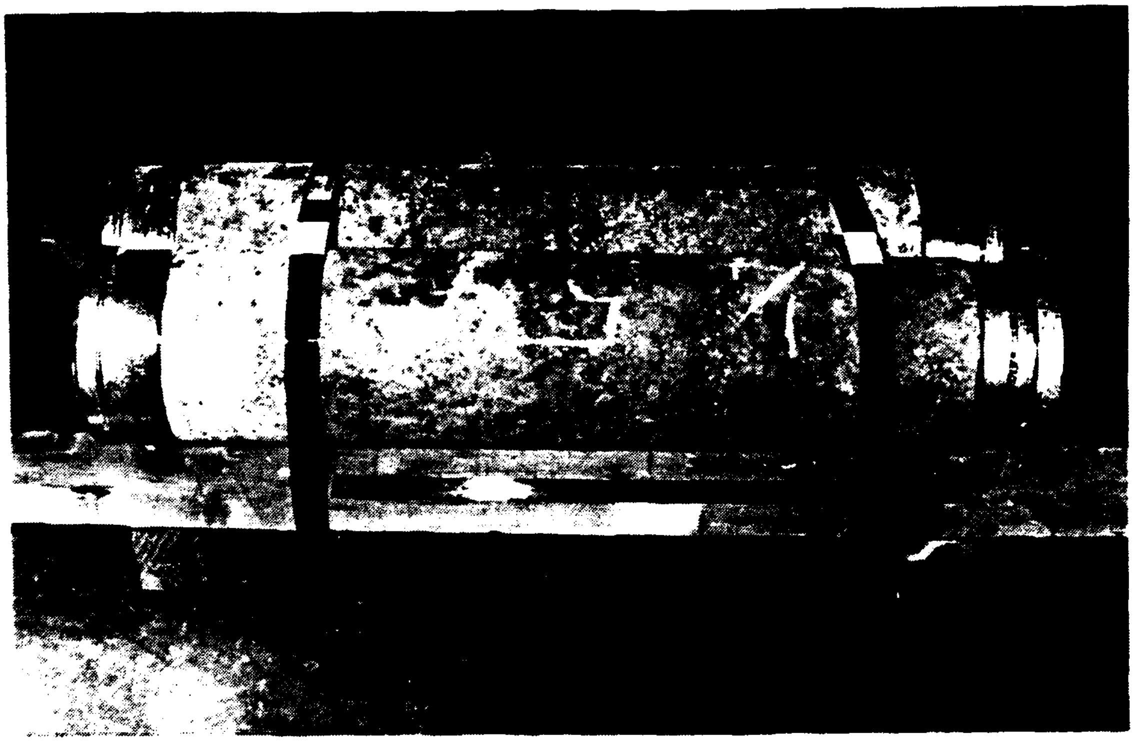
Table 3

Creep Rupture Test Data From Investment Cast XD™ Titanium Aluminide Test Bars.

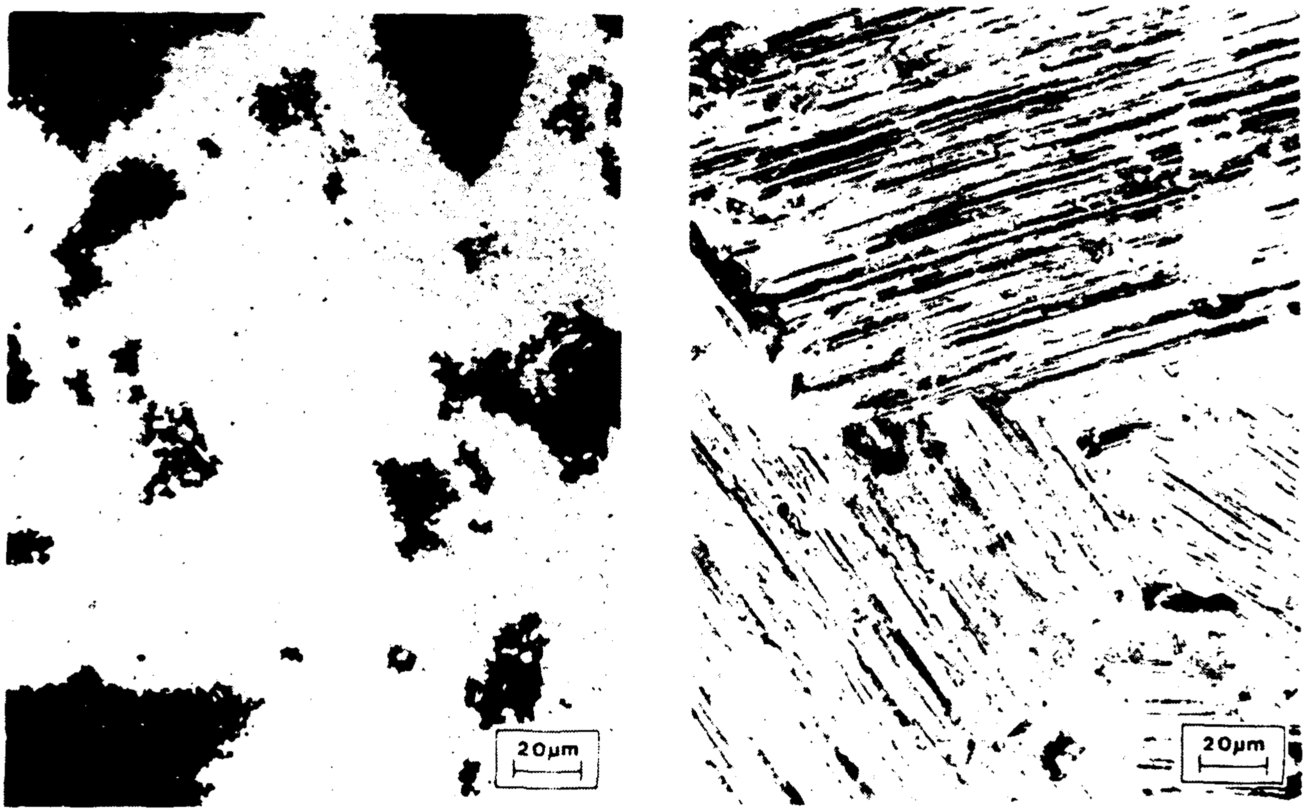
TEST TEMPERATURE	SPECIMEN ID	FINAL STRESS LEVEL (KSI)	ELONG. (%)	TIME TO CREEP (hrs.)			STEADY STATE CREEP RATE (% CREEP/h)	UPLOADED	TOTAL TIME TO FAILURE (hrs)
				0.5 %	1.0 %	2.0 %			
600°C (1120°F)	11322	85	5.2	218.0	287.3	358.8	0.0046	YES(1)	416.0
	11323	85	4.9	217.7	287.0	359.5	0.0054	YES(1)	410.0
	11324	85	6.3	217.7	246.5	310.5	0.0065	YES(1)	372.7
800 (1472°F)	11325	30	12.3	1.5	4.0	8.8	0.1176	NO	37.3
	11326	30	29.5	3.3	10.0	26.5	0.2667	NO	151.0
	11327	30	29.7	3.0	9.0	24.8	0.1120	NO	153.0
1000 (1832°F)	11328	10	39.0	0.5	0.7	1.5	2.0000	NO	12.0
	11329	10	45.3	0.3	0.7	1.5	2.7780	NO	10.5
	11330	10	46.6	0.5	1.3	2.7	1.6949	NO	17.5

(1) Uploaded Sequence: ~200 hrs, 40 KSI; ~48 hrs, 50 KSI; ~24 hrs, 55 KSI; ~24 hrs, 60 KSI; ~24 hrs, 65 KSI; ~24 hrs, 70 KSI; ~24 hrs, 75 KSI; ~24 hrs, 80 KSI; Uploaded to 85 KSI Until Failure.

T2-89/426:6

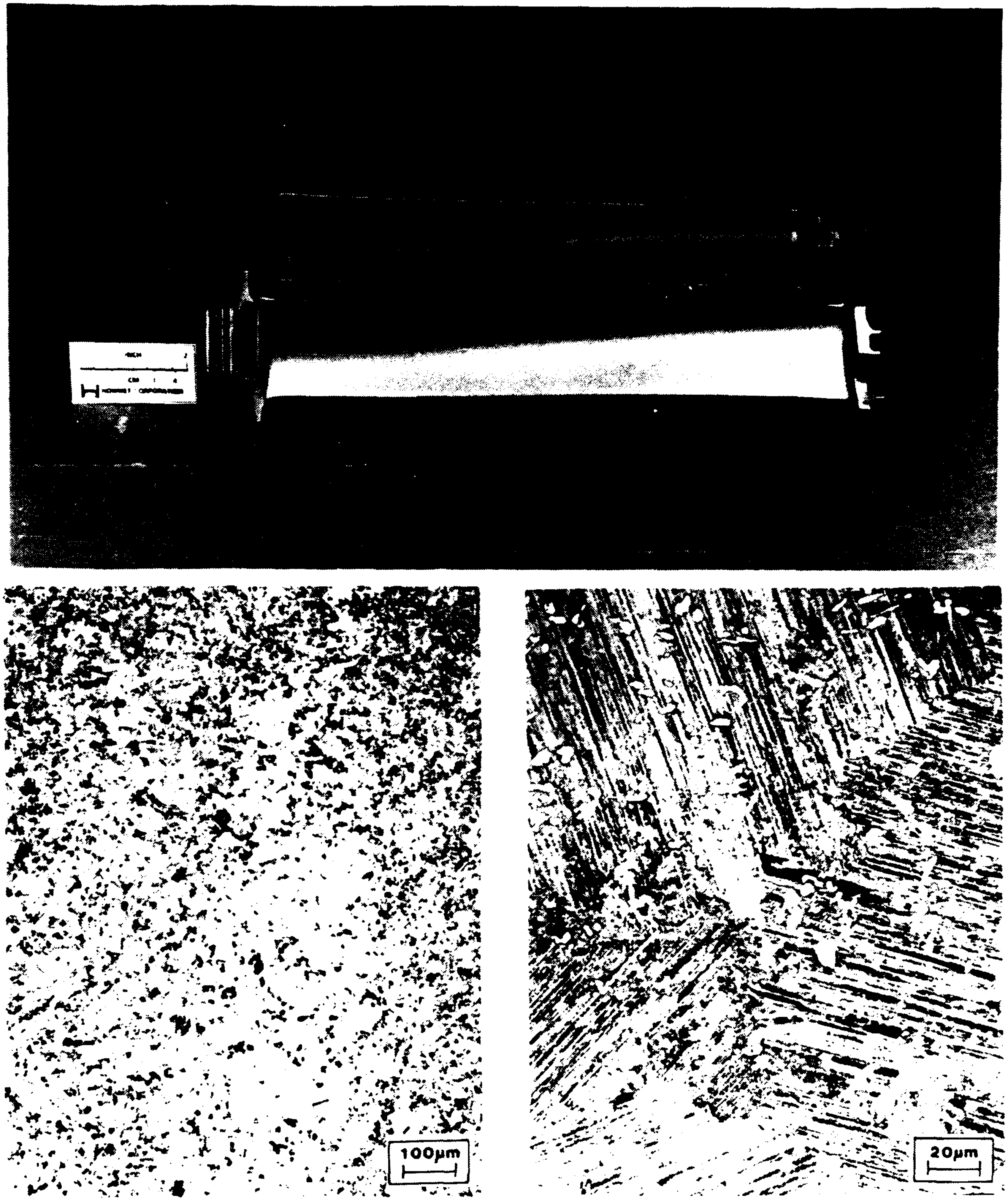


XD™ Titanium Aluminide Ingots: 8 Inches Diameter X 38 Inches Long.



TBA-70 XD™ Precursor 500X Ingot 500X

Figure 1. XD™ Titanium Aluminide Ingots and Typical Microstructure of the TiB_2 Precursor Material and the Ti-48a/o Al-2a/o V + 7.5 v/o TiB_2 Alloy Ingots.



100X 500X

Figure 2. CF6-80C2 5th Stage LPT Blade: As-Cast Oversized Blade and Fully Machined Blade and Representative Microstructure After HIP (2300°F/4h/25 KSI) and Heat Treatment (1650°F/16h).

XD™ Titanium Aluminide Alloy: Ti-48a/o Al-2a/o V + 7.5 v/o TiB_2

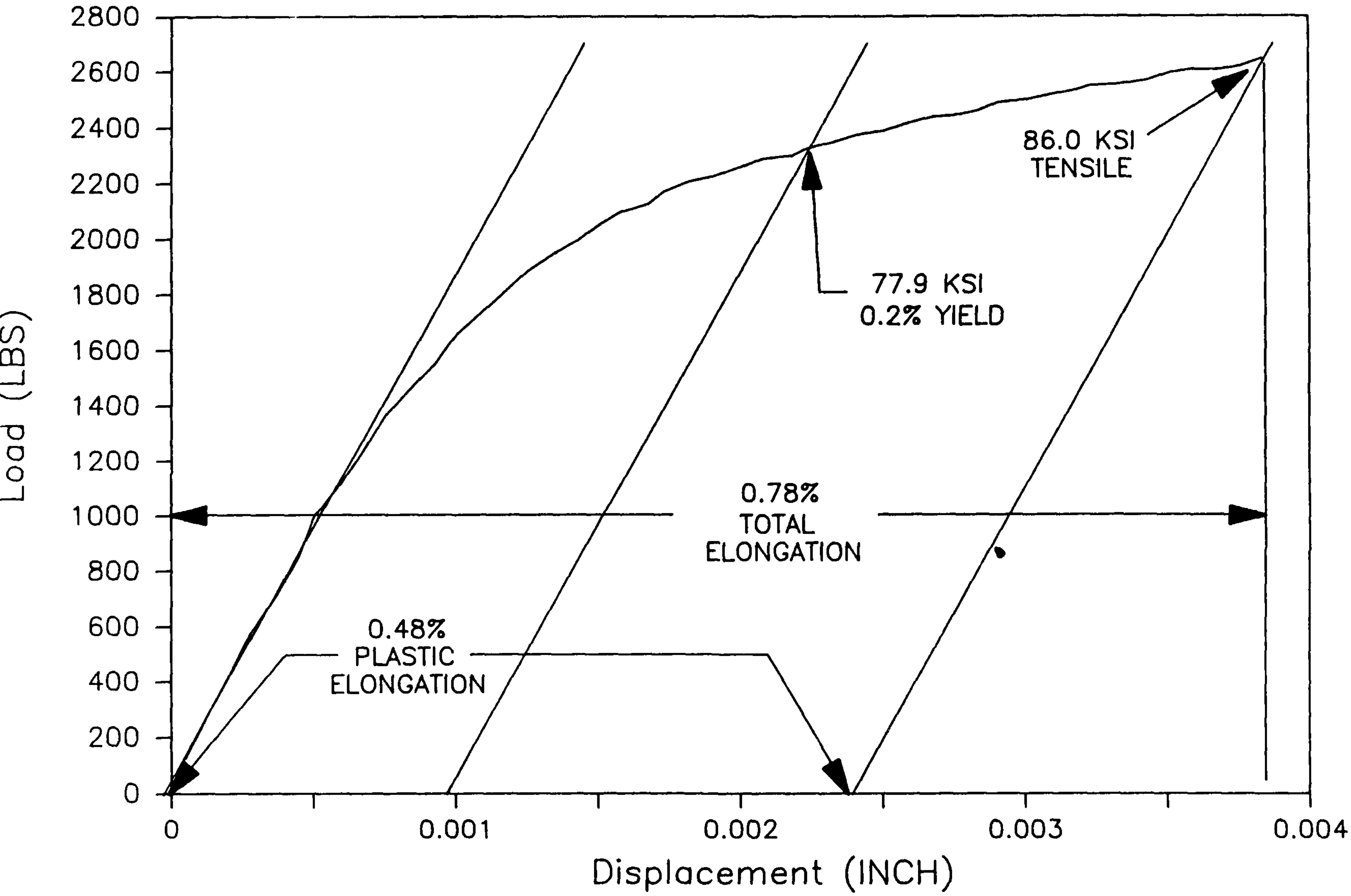


Figure 3. **Representative Stress-Strain Curve From RT Tensile Test. Note Two Methods Used to Measure % Elongation.**

XD™ Titanium Aluminide Alloy: Ti-48a/o Al-2a/o V + 7.5 v/o TiB_2

T2-89/426.3

DICE PROGRAM

Task 3.5.4.2 Plasma Spray Disk GE-Aircraft Engines

Objective:

The objective of this task was to develop and demonstrate the technical feasibility of using rapid solidification plasma deposition (RSPD) processing methods to produce advanced lightweight metal matrix composite (MMC) disk subelements for incorporation into gas turbine compressor blisks. This DARPA sponsored process technology was further developed by GEAE to enable the spraying titanium alloys by using induction plasma deposition (IPD) to produce fiber reinforced composites. The primary challenge faced in this task consisted of establishing a practical manufacturing path for the fabrication of a multilayer disk subelements which would produce a composite structure with maximum built-in quality and integrity. Accomplishing this objective with an advanced titanium aluminide matrix reinforced with a Textron SMD SCS-6 SiC fiber would represent a significant state-of-the-art advancement.

Approach:

To accomplish the objective of this task the following 2 subtasks were undertaken:

Subtask 1. Subscale MMC Disk Development and Evaluation

In Subtask 1 sufficient quantities of the titanium aluminide alloy, Ti-14Al-21Nb, and SCS-6 fiber (a 5.6 mil SiC fiber produced by Textron SMD, Lowell, MA) were procured to conduct IPD spray trials in facilities located at GE-CRD, Schenectady NY and GEAE, Lynn MA to develop a wind and spray technique for MMC fabrication. This wind and spray approach is illustrated in Figure 3.5.4.2.1 and consists of plasma spray deposition of the matrix alloy onto a drum followed by machining and grooving to accommodate fiber winding and respraying to repeat the cycle until the desired number of plies have been built up. Each step of the process requires care to avoid contamination during spraying, machining, handling and winding. The plasma spray operation is conducted in an inert (argon) atmosphere which is carefully maintained to minimize interstitial pickup. Figure 3.5.4.2.2 shows the IPD process in operation. The intent of this subtask was to fabricate 12" diameter by 32 ply MMC subelements with a fiber volume fraction of 0.30 for evaluation and potential spin test in subsequent phases of this program.

Subtask 2. Full Scale Disk Development & Evaluation

Subtask 2 efforts concentrated on tackling the manufacturing issues which are anticipated when stepping up to full scale size hardware (nominally 19-22" diameter MMC's). These evaluations included experiments to determine if several consolidated MMC hoops could be sized and nested to produce relatively thick multilayer rings which otherwise could not be fabricated due to potential fiber buckling during consolidation. A schematic of the planned manufacturing approach which includes HIP consolidation of hoop elements followed by a HIP diffusion bond of several elements is shown in Figure 3.5.4.2.1.

Technical Results:

Subtask 1. Subscale MMC Disk Development & Evaluation

a. 4" Diameter Hoop Trials

The initial wind and spray fabrication trial (RF1122) was performed using Ti_3Al powder sprayed onto a 4" diameter steel drum to a thickness of about 0.050" followed by machining a continuous groove to accept the SCS-6 fiber. The grooving was performed using a pie jaw chuck to hold the plasma sprayed base hoop. Upon winding, respraying and regrooving for the second ply attempt, separation at the machined surface/resprayed deposit interface was observed as shown in Figure 3.5.4.2.3. A second trial (RF1144) using a solid steel mandrel in an attempt to provide a more rigid structure for machining also resulted in separation at the same location. Subsequent work on thicker base hoops 4" diameter (0.125" thick) resulted in good 4 ply composites as discussed in Subtask 2.

b. 12" Diameter Hoop Trials

After an initial spray trial (LPS180) which was unsuccessful, a second trial (LPS182) was performed on a 12" diameter steel drum with the intention of doing subsequent machine-wind-spray operations on that drum. However, the 4" wide by 0.024" thick buildup separated intact from the drum on cooldown after spraying. Consequently, a third 12" diameter hoop (LPS187) was produced by first making a 5.5" wide by 0.068" thick base hoop of Ti_3Al which would serve as a self-consistent drum and then using a split drum holder made of steel for winding, spraying and machining. Unfortunately the split drum holder created a flat spot at the split which resulted in ply separation upon machining within 10 mils of the underlaying fiber winding. This hoop was sectioned to evaluate structure and possible fiber damage. As shown in Figure 3.5.4.2.4, excellent fiber spacing was achieved and no fiber damage was detected.

Based on the LPS187 results a 12" diameter trial (LPS215) was performed utilizing a thicker base (0.137") and a pie jaw holder for the machining operation. This approach proved successful for producing an 8 ply hoop which will be further processed in Phase 2 of this program. This MMC hoop is shown in Figure 3.5.4.2.5 at the 7th ply stage of processing. Minor separation of the outermost ply during machining after the 8th layer was deposited led to the decision to HIP consolidate at this stage of fabrication. Detailed manufacturing studies into the causes and prevention of this type of separation will be investigated in Phase 2 of this program. Since a total thickness of 32 plies is required for the planned MMC subcomponent testing, the fabrication of additional multilayer hoops which could be nested with LPS215 was initiated. The base layer for a second larger diameter hoop to mate with LPS215 was fabricated (LPS225) for further processing in Phase 2.

Subtask 2 Full Scale Disk Development & Evaluation

This subtask involved fabrication trials using 4" diameter by 4" wide by 4 ply hoops of Ti_3Al/SCS-6 composite. These hoops sometimes exhibited a slight amount of out-of-roundness after wind and spray processing which would be unacceptable for subsequent nesting and diffusion bonding. In order to address this issue several slices of single ply RF1122 hoop were thermally sized by placing them around a precisely machined 4" diameter mild steel cylinder (undersize by about 0.020" - just enough to slip the distorted hoop over). This assembly was then heated to 1652°F in a vacuum for 15 minutes and allowed to cool. After this cycle the MMC hoops retained the shape of the sizing cylinder with no evidence of out-of-roundness. Subsequent experiments revealed that sizing the base layer prior to wind and spray operations and post-HIP sizing of 4 ply hoops resulted in MMC cylinders which could be nested for diffusion bonding.

Inner and outer 4 ply hoops (RF1227 and RF1249) were fabricated for HIP diffusion bonding to demonstrate the nesting approach in Phase 2. Sections of these hoops were evaluated prior to HIP consolidation to determine fiber spacing and potential damage. Excellent spacing and no fiber damage was observed in these 4" MMC hoops as shown in Figure 3.5.4.2.6.

Conclusions:

1. Wind and spray as an approach for producing Ti_3Al MMC subelements for potential compressor blisks appears to be feasible and offers a unique ability to control fiber spacing.

2. Thermal sizing of MMC hoops which may be distorted from the wind and spray processing is a viable approach to achieving round subelements for possible nesting to produce thicker MMC cylinders.

3. Iterative wind-spray-machine operations to produce cylindrical MMC's can result in separation of the outermost ply when machining. The cause of this problem must be determined.

4. The wind-spray-machine process does not result in fiber breakage during processing.

Recommendations:

Manufacturing methods development should be continued to establish viable techniques for advanced Ti_3Al MMC component fabrication. These techniques should incorporate process control aspects which produce MMC parts with quality built-in to minimize reliance on post fabrication NDE which is extremely difficult in these complex materials. Such process control methods as on-line temperature, mass-flow, and environment status monitoring during plasma spray and tool temperature, forces and wear during machining should be developed. Some of these control methods are planned for development in subsequent phases of this program.

Publications:

There have been no publications resulting from the work performed in this task.

Hardware:

	Item	Purpose	Disposition
1.	LPS180 - 12"D Hoop	Initial large spray trial.	Scrapped.
2.	LPS182 - 12"D Hoop	Second base alloy trial.	Scrapped.
3.	LPS187 - 12"D Hoop	First fiber wound trial, 1 ply, delaminated.	Cut-up for evaluation.
4.	LPS197 - 12"D Hoop	Base alloy hoop to replace LPS187, too thin.	Scrapped.
5.	LPS215 - 12"D Hoop	8 ply fiber wound hoop.	To be processed in Phase 2.
6.	LPS225 - 12.5"D Hoop	Base alloy for larger hoop.	To be processed in Phase 2.
7.	RF1122 - 4"D Hoop	1 ply initial trial hoop with 50 mil base, cracked during machining.	Used for sizing trials.
8.	RF1144 - 4"D Hoop	1 ply sprayed on solid steel mandrel, unsuccessful due to CTE mismatch problems, debonded during cooldown.	Scrapped.
9.	RF1227 - 4"D Hoop	4 ply fiber wound hoop on 0.125" base.	To be processed in Phase 2.
10.	RF1243 - 4"D Hoop	2 ply fiber wound trial due, unsuccessful due to high interstitial pickup.	Scrapped.
11.	RF1249 - 4"D Hoop	4 ply fiber wound hoop on 0.125" base.	To be processed in Phase 2.

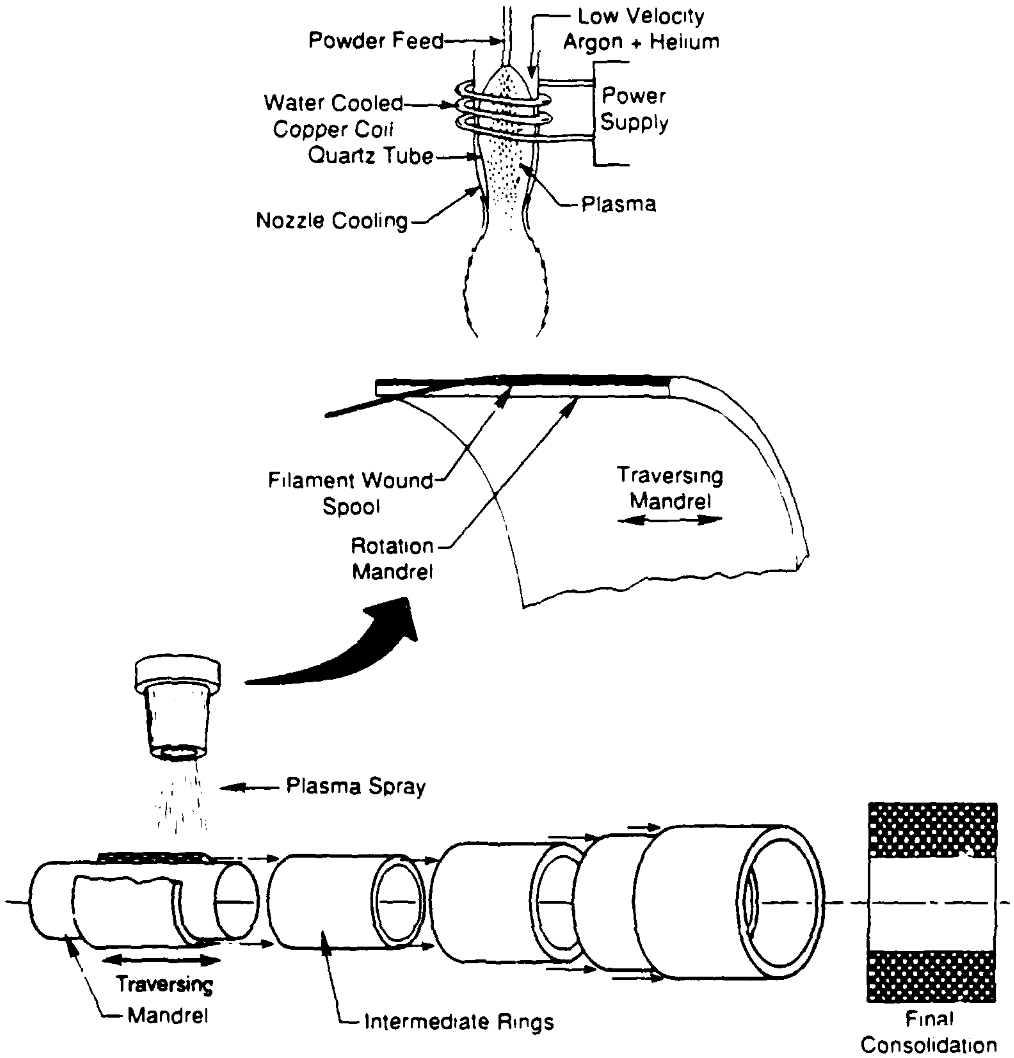


Figure 3.5.4.2.1 Wind and spray approach to MMC subelemen· fabrication.

Figure 3.5.4.2.2 Induction plasma deposition (IPD) process in operation.

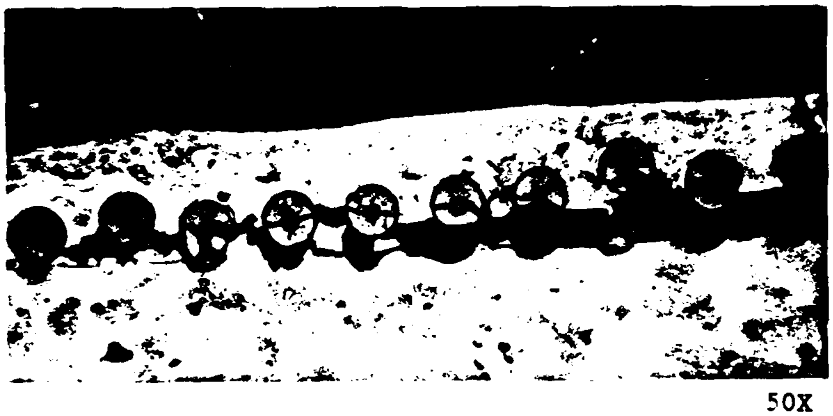


Figure 3.5.4.2.3 Ply separation during machining of 4" diameter trial RF112? for Ti_3Al/SCS-6 MMC.

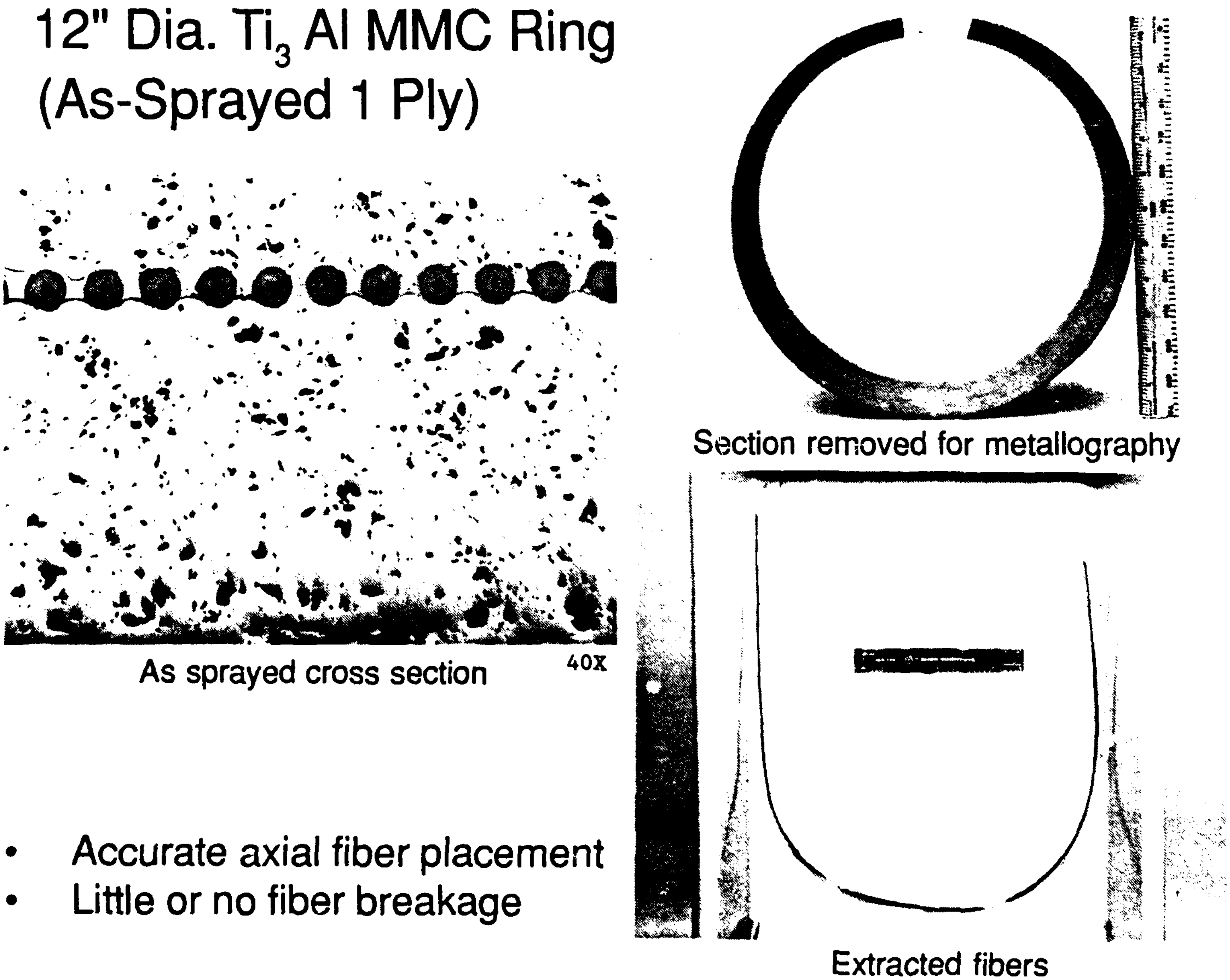


Figure 3.5.4.2.4 Microstructure and fiber damage evaluation of 12" diameter trial LPS187 for Ti_3Al/SCS-6 MMC.

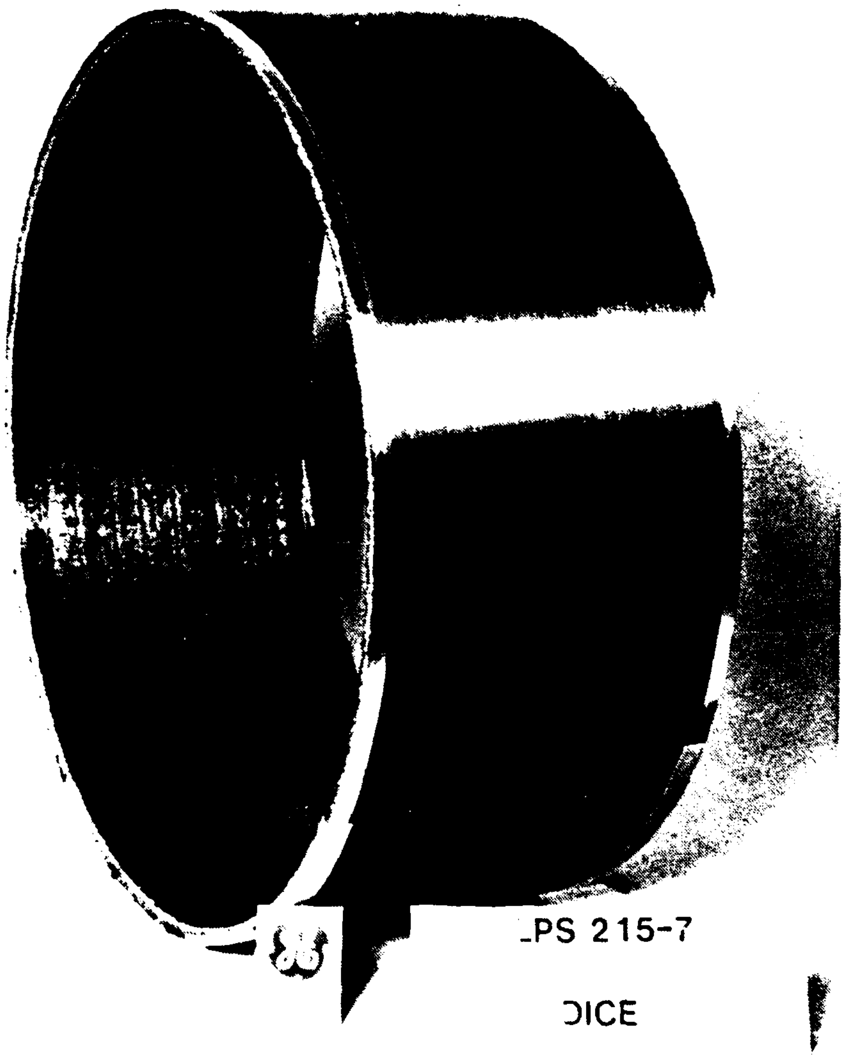
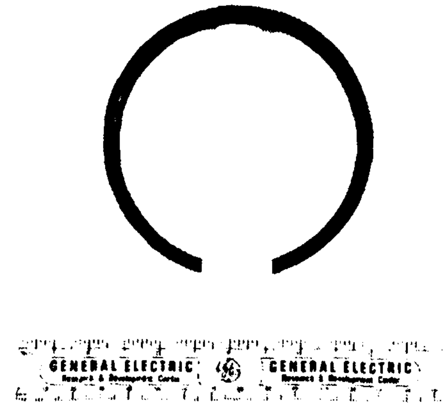
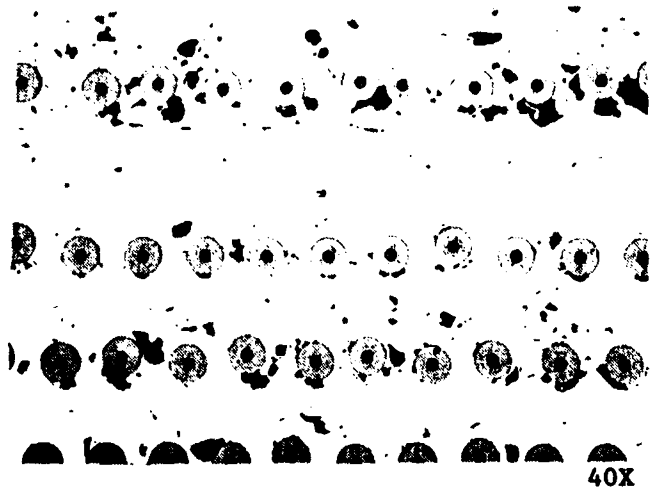


Figure 3.5.4.2.5 Plasma sprayed and fiber wound 12" diameter hoop LPS215 after 7 machine-wind-spray operations ready for 8th spray process.

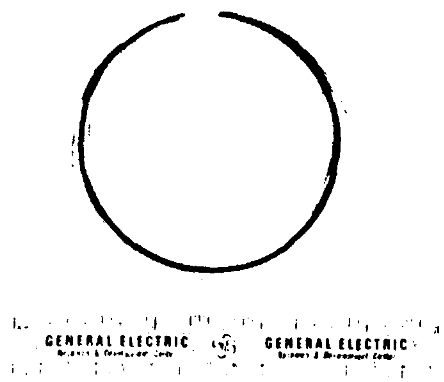


Four-Ply As-Sprayed Ring



Cross-Section of Four-Ply Ring

Unbroken Fibers from Four-Ply Ring



Unbroken Fibers from Thermally Sized Ring

Figure 3.5.4.2.6 Manufacturing development studies on 4" diameter trial RF1227 evaluated for microstructure and fiber damage.

DICE PROGRAM

Task 3.5.5 Component Testing GEAE

Under this task, Metal Matrix Composite ducts are fabricated and tested to identify manufacturing parameters and their related effects on the overall strength and quality of parts. Design, Material and manufacturing decisions will affect the strength, time to manufacture and quality of a component. A detailed investigation and identification of these parameters will help the designers to make a more informed decision with fewer design modification, therefore reducing the time from concept to market. In brief the manufacturing and testing of the MMC ducts will aid in producing a CONCURRENT Engineering DATABASE which is directly applicable to manufacturing of pressurized vessels in high temperature, pressure and bending environment. This knowledge can be further projected to design other components which have a similar geometric configuration, loading environment or material needs.

Under task 5 of the DICE program Subscale ducts were identified, designed and a vendor was selected to manufacture these components. The material chosen is continuous Silicon Carbide fiber reinforcing a Titanium alloy matrix (SiC/Ti). Due to severe interaction capability of Titanium with other materials and in particular Silicon Carbides, the fiber chosen should have a proper coating to act as buffer between the Titanium alloy and the Silicon Carbide fiber. Currently the only fiber available with such a coating is produced by TEXTRON Specialty Material and designated as SCS-6. The matrix chosen for the subscale duct is Ti-6-242 which has the desirable matrix material characteristics for a multi-axis loaded part. A university will be chosen for testing of the ducts to increase University Industry Interaction (IUI) on this project. Subcontractors final report and the quality plan developed under this contract is included as a part of this report.

The task of producing and testing of Metal Matrix Composite (MMC) ducts has been broken into five phases due to scheduling requirements. The process and tasks by which the ducts are to be designed, analyzed, manufactured and tested are as follows :

1. *Raw material will be ordered to produce items in Table I, (Phases 1 & 2).*

2. *Plasma Spraying technique will be used to produce SCS-6/Ti-6242 MMC monotapes, (Phases 1 & 2).*

3. *The MMC Monotapes will be cut and assembled on mandrels to the design specified ply stack up and consolidated using Hot Isostatic Pressing (HIP), (Phases 1, 2 & 3).*

4. *Specimens will be produced using techniques that parallel the duct fabrication process and uses the same ply stack up, (Phases 1 & 2)*

Table - I MMC components under DICE phases 1 and 2.

Phase 1
Quality Assurance Plan.
Fiber and Matrix Powder
Plasma Spray of Monotapes
One Complex plate.
Reports.
Phase 2
Quality Assurance Plan (Continued)
Duct fabrication Technical Plan.
Plasma Spray of Monotapes (Continued)
Six Simple plate.
Two Complex plate.
Two Trial Ducts.
One Simple Duct.
One Complex Duct.
Reports.

5. *Specimens will be tested for Chemistry, Micro and Macro Structure, and overall strength capability and failure modes specific to the ply stack up of the ducts, (Phases 2 & 3).*

6. *Flanges, Stiffeners, Pads and Bosses will be machined, Chem-milled or formed on the duct, (Phases 3 & 4).*

7. *The duct(s) will be evaluated using NDE technique before testing for stiffness, strength, buckling capability and modes of failure, (Phases 3, 4 & 5).*

8. *Analysis will be performed for all test configurations and conditions. Analysis results will be correlated with Test, (Phases 1, 2, 3, 4 & 5).*

9. *Design, Fabrication, Material, test, Analysis and Correlation parameters will be added to the DICE DATABASE, Phases 1, 2, 3, 4 & 5).*

Phase 1, Task 3.5.5

Under DICE phase 1, all of the raw materials except for long lead items were acquired and are at the vendor's, TEXTRON, plant. The remaining materials which will be acquired under phase 2 are the Titanium forgings for mandrels and the Titanium foil used during the stackup and assembly. Seventy percent of the SCS-6/Ti-6-2424 monotapes have been plasma sprayed. The monotapes have been cut to the proper shapes and are now being stored in Argon gas at vendor's site. A preliminary Quality Assurance plan was submitted by TEXTRON and GEAE has provided comments and additions to the plan. This plan will be finalized by the vendor under phase 2. A complex plate per sketch (2) was assembled and the HIP canister was Electron Beam welded. However, due to time constraints this part could not go through the two HIP cycles required to consolidate the monotapes and produce the plug for the penetrations. It is proposed to finish this part under phase (2). The main reason for the delay in the process has been :

1. *TEXTRON experienced a delay in receiving the large drum which is used to produced plasma sprayed monotapes.*

2. *The vendors for the Titanium forgings and foils were not able to deliver the material to TEXTRON before September 30, 1989.*

Information learned during phase (1) :

1. *Knowing the size of a plasma spray drum and the part to be manufactured, the waste factor on the monotapes can be calculated.*

2. *The amount of fiber and matrix to spray one 2' by 8-9' monotape and the associated waste factors are known.*

3. *Manufacturing hours for assembly of a 6" X 6" plate with 8 plies is known. The results can be projected for other plates with different sizes and layups.*

4. *Manufacturing hours and the time for spraying a 2' by 8' monotape is known.*

5. *Fiber splices on a plasma sprayed monotape will produce undesirable results. The fibers were spliced with acrylic spray which separated under the heat from the plasma sprayed Titanium and the fiber bending due to the drum curvature. The resulting force causes the final monotape to have cracks and splits parallel to the fibers and starting at the spliced joints. In order to avoid this problem, all of the fiber splices are placed in the cut/trim section of the monotape while the fiber is being*

wound on the drum. This requires additional labor during the fiber winding operation.

6. *When the plasma sprayed monotape is removed from the drum it tends to curl along the axis parallel to the fibers. This is due to the residual stresses in the transverse-in plane direction. The curling of the high aspect ratio monotape produces storage and handling problems. To avoid any damage to the monotape and reduce the required storage space, the monotapes are cut into the desired pieces and then stored. The storage medium is a neutral argon gas environmental.*

A report from Textron Specialty Materials is attached.

DICE PROGRAM

Task 3.5.5 Component Testing

Textron

1.0 INTRODUCTION

The objective of this program was to fabricate a Ti-6-2-4-2/SCS-6 Metal Matrix Composite (MMC) plate representative of the manufacturing techniques for several duct configurations using cross-plied composite layups. Plasma sprayed monotapes will be produced and used as the preform to the obtain the composite. This effort is in support of Phase I of the DICE program under task 4.0, manufacturing. Another objective unoer Phase I is to acquire long lead time material for the Phase II program. These materials will include matrix powder, SCS-6 fiber, tooling and a portion of the plasma sprayed monotapes. The major tasks of this program are listed as follows:

1. Definition of a Quality Assurance Plan applying to both Phase I and Phase II.

2. Fabrication of a complex plate per the Statement of Work (RFQ 981U031, Rev. A).

3. Acquisition of matrix powder, SCS-6 fiber, Tooling, and a portion of the plasma sprayed monotapes required under the Phase II SOW.

4. Identification of key parameters affecting the fabrication and manufacturing of MMC components for the DICE Duct Program.

2.0 QUALITY ASSURANCE PLAN

A preliminary QA Plan was submitted to GE for review, recommended changes were incorporated into the plan, and a final QA Plan was submitted.

3.0 FABRICATION OF A COMPLEX MMC PLATE

Fabrication of a complex plate shown in Figure 1 (Sketch 2 of the SOW) has proceeded. The following steps in this fabrication have been completed:

1. Ti-6-2-4-2 plate and sheet material to provide the monolithic material surrounding the composite were acquired.

2. Ti-6-2-4-2 powder required to produce the plasma sprayed monotapes was acquired.

3. Plasma sprayed monotapes to supply the composite preforms were produced.

4. The 3/4" thick Ti-6-2-4-2 plates were machined flat and parallel and the .065" sheets were cut into "picture frames" as shown in Figure 2.

5. +/-65 and +/-15 plies were cut from the monotapes and assembled into the machined plates and sheets.

6. The composite region was evacuated and sealed by EB (Electron Beam) welding.

7. The sealed assembly was successfully leak-checked.

The following steps need to be done to complete fabrication of this plate:

1. Dimensional inspection of the assembly.

2. HIP consolidation (Hot Isostatic Pressing at 1750 F/ 15 ksi/ 3 hour hold).

3. Dimensional inspection.

4. Drill center hole and EB weld Ti-6-2-4-2 bar in place.

5. Leak check and dimensional inspection.

6. HIP again to diffusion bond bar.

7. Dimensional inspection and machine per Sketch 2 (Figure 1).

8. X-Ray, C-Scan, Metallography, Matrix Chemistry, final inspection.

A ROM estimate has been submitted to GE to complete the fabrication of this plate under Phase II.

4.0 ACQUISITION OF MATERIAL FOR PHASE II.

The following materials have been obtained under Phase I of the DICE Duct Program to support the Phase II efforts:

- Ti-6-2-4-2 powder for both Phases.

- Ti-6-2-4-2 plate and sheet material to fabricate complex plates per Sketch 2 for both Phases.

- Steel tubing to be used as mandrels to fabricate the trial Duct under Phase II.

- 7 plasma sprayed monotapes were sprayed on a 36" diameter drum producing 2-feet by approx. 9-feet long monotapes. Of the 7, 6 sprayed well while 1 had splits due to fiber breakage leaving only approx. 5" wide strips available for use.

- The silicon carbide SCS-6 fiber needed to complete both Phase I and Phase II was acquired under Phase I.

The following long-lead materials have been ordered. but could not be obtained in time for Phase I:

- -3 and 5-mil thick Ti-6-2-4-2 foil.
- -Ti-6-2-4-2 ring forgings to be used as mandrel and monolithic material for the duct components.

5.0 KEY PROCESS PARAMETERS

The following key processing steps for producing plasma sprayed monotapes have been developed and demonstrated as applied to the DICE Duct Program:

1. A 36" diameter drum has been installed in the low pressure plasma spray chamber at Textron Specialty Materials and SCS-6 fibers have been successfully wound on the drum to produce the desired fiber spacing. Techniques have been developed to splice fiber reactor runs when necessary on the drum. These splices are placed along the same axial line so that they can be shielded from the plasma flame to avoid damage to the splices and also to locate them in the cut line so that there are no splices in the used area of the monotapes.

2. Plasma parameters used previously to spray Ti-6-4 powder were successfully implemented to spray Ti-6-2-4-2 powder onto the SCS-6 fibers to produce monotape preforms that are 2 feet wide by approximately 9 feet long. The monotapes are sprayed onto a .005" Mo foil that is wrapped around the drum which separates from the monotape after spraying due to thermal expansion differences. Processing time for each monotape is approximately 8 hours which includes 3-4 hours for drum winding (could be eliminated with second drum). 2 hours for purging. 1/2 hour for spraying, and 1 1/2 hours for cooldown. opening chamber and removing monotape.

3. Techniques have been developed to handle the monotapes after spray as they tend to curl upwards from the length edges ("stove-pipe") when they are released from the drum. This creates storing problems so a monotape is clamped to a suitably sized table as it's removed from the drum and immediately cut into plies.

4. The seven monotapes that have been sprayed for this program are currently being evaluated for post-spray matrix chemistry. extracted fiber strength (after spray and removal from monotape). extracted fiber condition. consolidated properties. amount of matrix after consolidation. and metallography after spray and consolidation.

6.0 CONCLUSIONS/RESULTS

Significant progress has been made under this program regarding fabrication of plasma sprayed Ti-6-2-4-2/SCS-6 monotapes and acquisition of materials for the DICE Duct Phase II Program. These efforts will continue under Phase II.

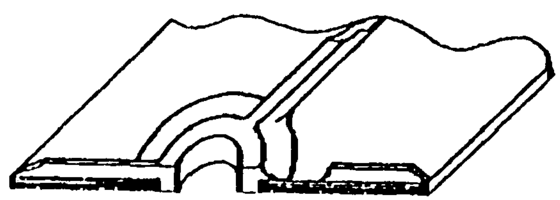
FIGURE 1

Sketch (2)

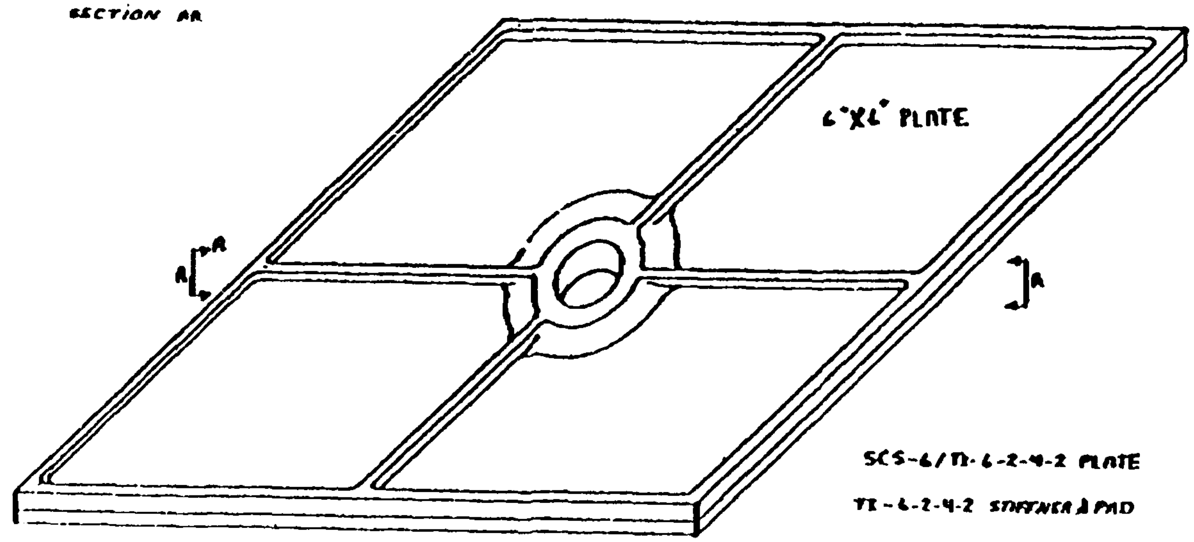
Notes:

1. **Plate flatness** - Bow in the plate shall not exceed 0.050" in 4.0 inches.
2. **Surface finish** - As consolidated, the plate shall have no dents, pours, or cracks. Roughness shall be less than 32AA.
3. **Fiber alignment** - Shall be within ±1 degrees from the specified angle.

+65
-65
-15
+15
+15
-15
-65
+65



SECTION AA

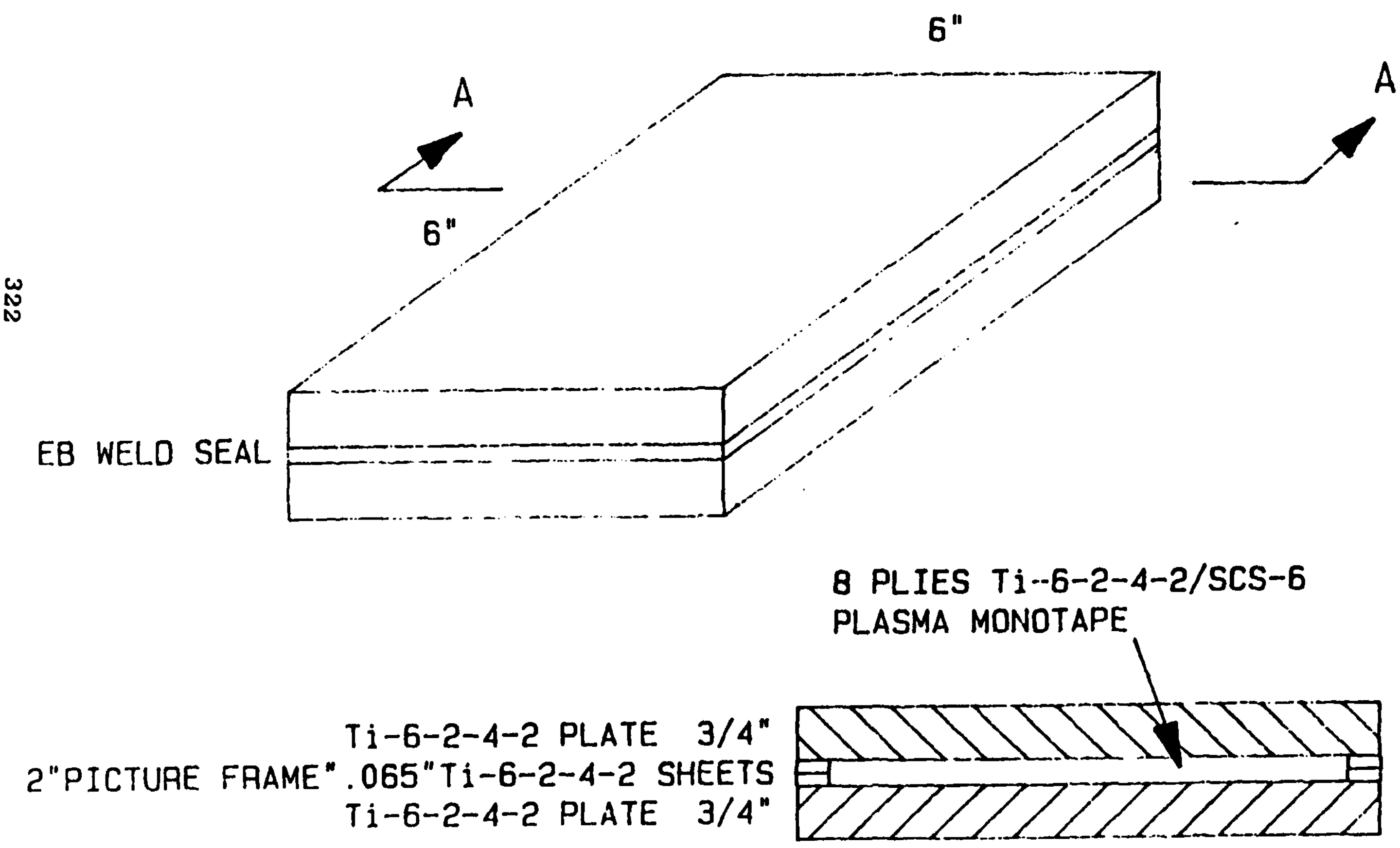


MMC PLATE

FIGURE 2

MMC COMPLEX PLATE FABRICATION

(BEFORE MACHINING)



SECTION A-A

DICE PROGRAM

Task 3.5.8.1 NDE/QA GEAE

Objective:

The objectives of this program are to develop and evaluate NDE inspection methods applicable to MMC structures and components and to establish controls and implement monitoring of the MMC process early in the manufacturing cycle. Conventional NDE inspection methods may not be adequate to fully assure the quality of MMC materials. This effort is needed to evaluate advanced NDE techniques and provide the methodology to effectively assess MMC core engine components and to assist in providing effective process quality assurance to reduce post process inspection requirements.

Approach:

Nondestructive Evaluation and Inspection methods are needed to assess the quality of Metal Matrix Composite sub-elements and sub-components and to develop and establish effective inspection methods during the component development and test work efforts. To accomplish this, inspection calibration and reference standards need to be defined and manufactured. These standards afford a means of measuring and reproducing the effectiveness and repeatability of the inspection method on a daily basis or over a period of time. Design of the calibration standards would be based upon an Engineering definition of the type, size and nature of the defective condition(s).

In general, the defective conditions can be classified as 1) those related to the MMC structure, and 2) those related to bonded joints. Multiple standards of various designs are generally required to fully simulate the range and types of defects encountered.

Engineering had initially defined the following conditions as undesirable and in need of NDE detection methodology:

Incomplete consolidation - Delamination
Voiding/porosity
Fiber breakage/degradation/interface reaction
Incomplete or poor bonds

This list formed a basis for initial development considerations. Other undesirable conditions may exist but these would be covered at a later time in the overall Process Control and Quality Assurance Plan.

These reference standards and specimens would be initially evaluated with conventional NDE methods which would provide a screening test to identify the more obvious defect conditions. Identification of some of these defective conditions would require development and application of other more advanced NDE techniques and methodologies. Selected specimens would be cut-up and examined by metallography to establish correlation between material condition and the NDE test results. From these results, the most promising NDE candidate method(s) for identifying the various defective conditions would be selected for potential scale-up and application to inspection of specific component assemblies, culminating in an inspection plan for production hardware.

Technical Results:

Plasma spray specimens could not be manufactured for experimental NDE measurements during Phase I which changed the original program direction. Lack of the material is discussed elsewhere in the report. A reassessment of the technical program objectives was conducted which resulted in a redirection of the effort toward establishing an advanced theoretical ultrasonic model. This work would ultimately be required and would prove valuable, at this stage of the program, in predicting optimum parameters and direction for the eventual accumulation and analysis of experimental ultrasonic data. Therefore, a purchase order was placed to conduct the advanced theoretical ultrasonic analysis and modeling for GEAE MMC material systems. MMC defect specimens previously manufactured for GEAE by the foil-fiber-foil layup method would be used to experimentally evaluate and verify the theoretical analysis and ultrasonic measurements. The study was split into two sections to coincide with Phase I and Phase II of this program.

Phase I goals of the theoretical study were 1) determination of the effective properties of the MMC from those of the individual components and their volume fractions, 2) introduction of the transformation and calculation of the properties for a laminate of any orientation from those of the reference values, 3) the outline of the derivation of the reflection and transmission coefficients for normal incidence on multilayered plates immersed in liquid, and 4) determination of the different ultrasonic wave velocities as a function of the fiber volume fraction. These are discussed in the sections following.

Effective Properties of Metal Matrix Fibrous Composites

In this section the effective mechanical properties of the metal matrix composite is derived in terms of its fiber and matrix properties and volume fractions (Table 1). In order to perform the necessary calculation an approximate analysis is formulated in which the composite medium is replaced by a homogeneous medium. In the process of replacing the composite with such a material model, the fiber-matrix interfacial continuity conditions are utilized. First, a layered structure composed of fiber and matrix layers is analyzed and finally a fibrous model is established by treating the composite as consisting of the compound layer stacked in series with the matrix. Figure 1 shows a schematic view of a layered composite and the coordinate system.

From the above model the composite is treated as a transversally isotropic material. This means that only five independent constants are necessary to uniquely describe the mechanical behavior of the composites.

The necessary equations required to obtain these constants from the fiber and matrix properties have been derived. These constants are for a layer of the composite oriented at 0° only, or for a bundle of layers (unidirectional).

The Transformation

Objective of this section is to transform the five constants obtained previously into new constants for a laminate oriented in any plane. This process is called transformation and it uses the laws of tensors and trigonometry to achieve this. When the five constants are transformed they will form a new set of 13 constants. Again these constants are for a layer oriented in a particular direction or for a unidirectional stack and not for a stack with different layup orientation sequences. Tables 2a and 2b show the new material constants for a unidirectional stack with 35% fiber volume fraction and, for comparison, when a single layer in the stack is rotated from 0° to 45°.

Ultrasonic Normal Reflection from Fluid-Loaded Multilayered Anisotropic Plates

Objective of this section is to obtain the material constants for the entire composite material, for the constituent layers, and then develop a model to show the reflection and transmission properties of the entire composite. The solutions are obtained in two steps. In the first, formal solutions are derived for each layer which relates the field variables at its upper and lower layer surfaces. In the second, the response of the total plate proceeds by satisfying appropriate interfacial conditions across the layers. Figure 2 shows the reflected amplitude for a three layered metal matrix model. The sharp dip in the reflection indicates a high transmission at this frequency thickness product. The maximum transmitted energy is at 6 MHz for a 0.6 mm thick unidirectional MMC plate. This theoretical prediction was also experimentally confirmed.

Ultrasonic Wave Velocities

Objective of this section was to compute the different wave velocities from the material constants. This was accomplished and showed a good correlation with the experimental data obtained on the Ti-6Al-4V/SCS-6 MMC fiber-foil-fiber material. Table 3a shows the theoretical ultrasonic wave velocities as the fiber volume fraction is changed. Table 3b shows the longitudinal wave velocity obtained experimentally. Experimental data for wave velocities are very difficult to obtain and no attempt was made, at this time, to determine velocities other than for the Longitudinal-11 constant. Part of the difference between the theoretical and experimental velocity values can be attributed to the difference between the average property data, which was used for the theoretical calculations, and the properties of the MMC plate used in the experiment. Updated property values which may more nearly represent current MMC products has been requested for review.

Ultrasonic Through-Transmission Screening Inspection of Mechanical Specimen MMC Panels

In addition to the above theoretical study, several Ti-14Al-21Nb MMC panels fabricated for mechanical property tests were received for ultrasonic inspection. A through-transmission ultrasonic screening test was made on these panels to assess the general condition prior to mechanical testing. Because of the lack of prior NDE standard panels on which to develop and refine inspection parameters (previously discussed), this test was intended to only look for the more obvious defects such as lack of bond or a voiding condition.

Figures 3 through 7 show the ultrasonic through-transmission test results. Each specimen was clamped in a fixture and scanned in a rectangular raster pattern (Y,Z). Scanning parameters were: speed = 6 inches/second, data accumulation rate = each .010 inch of horizontal travel (Y axis), and vertical index between each scan = .010 inch (Z axis). The dark rectangular spots seen at the top and bottom center in Figures 3-6 are from the holding clamps. Texture from the surface finish and fibers can be seen in the general overall appearance of the pictures. The several vertical dark lines are probably from some type of process striations since these "marks" can also be visually observed on the panel surface. However, the width and darkness of the vertical line in panel RF1171D (Figure 6) suggests that possible voiding or fiber bunching may be present since this area has attenuated the sound transmission to a much greater extent than in the other areas and panels. Metallography would be required to identify the nature of this anomaly. In other aspects, panels in Figures 3-6 generally appear typical of the general population of panels which have been examined by this type of initial screening inspection. By contrast, panel RF1174E, Figure 7, shows a number of crack indications, most of which can also be visually observed. The crack indications in the picture are those dark curved lines which cross the general texture of the part (horizontal pattern).

The color scale selected for making these pictures was rather arbitrary to best show some of the features discussed. The differences in color pattern is not of particular significance. Specific features and variations in the panels can be more readily observed during dynamic change of the color scale of the electronic picture. This is difficult to show in a single color scale view.

Conclusion

At this point of the study one can take any material (fiber & matrix) and obtain their equivalent composite properties from their individual properties. From these properties the different ultrasonic velocities in the material can be computed. Subsequently, the best transducer frequency for maximum transmission for a particular laminate construction can obtained. Transducer selection for the through-transmission (normal incidence) experiment was simplified to that showing best transmission. Data obtained from this analysis was useful in performing the through-transmission screening ultrasonic evaluation of MMC panels.

These initial findings will be further explored and substantiated in Phase II. A final report from A. Nayfeh, Engineering Consultant, describes in detail all the mathematics and results obtained during this phase of the program. The information is too lengthy for reprint in this text but is available and will be issued upon completion of the final modeling studies to be completed in Phase II.

Recommendations

Phase II of the theoretical modeling study will be undertaken and will address oblique incidence defect characterization and attenuation measurements. Plasma spray specimens should be manufactured for NDE assessment and the original objectives and goals of the Phase I effort accomplished. This activity has been planned under Phase II.

FF (v/o)	C11	C22	C55	C12	C23	Density (#/in³)
0.35	36.62	32.28	9.85	11.32	15.58	0.143
0.40	38.66	33.83	10.44	11.53	15.63	0.141
0.45	40.71	35.46	11.07	11.76	15.67	0.138
0.50	42.76	37.18	11.74	12.00	15.68	0.136

TABLE 1. - Variation In The Five Material Constants And The Density As The Fiber Volume Fraction (FF) Is Changed For A Unidirectional Composite Layer (TI-6Al-4V/SCS-6).

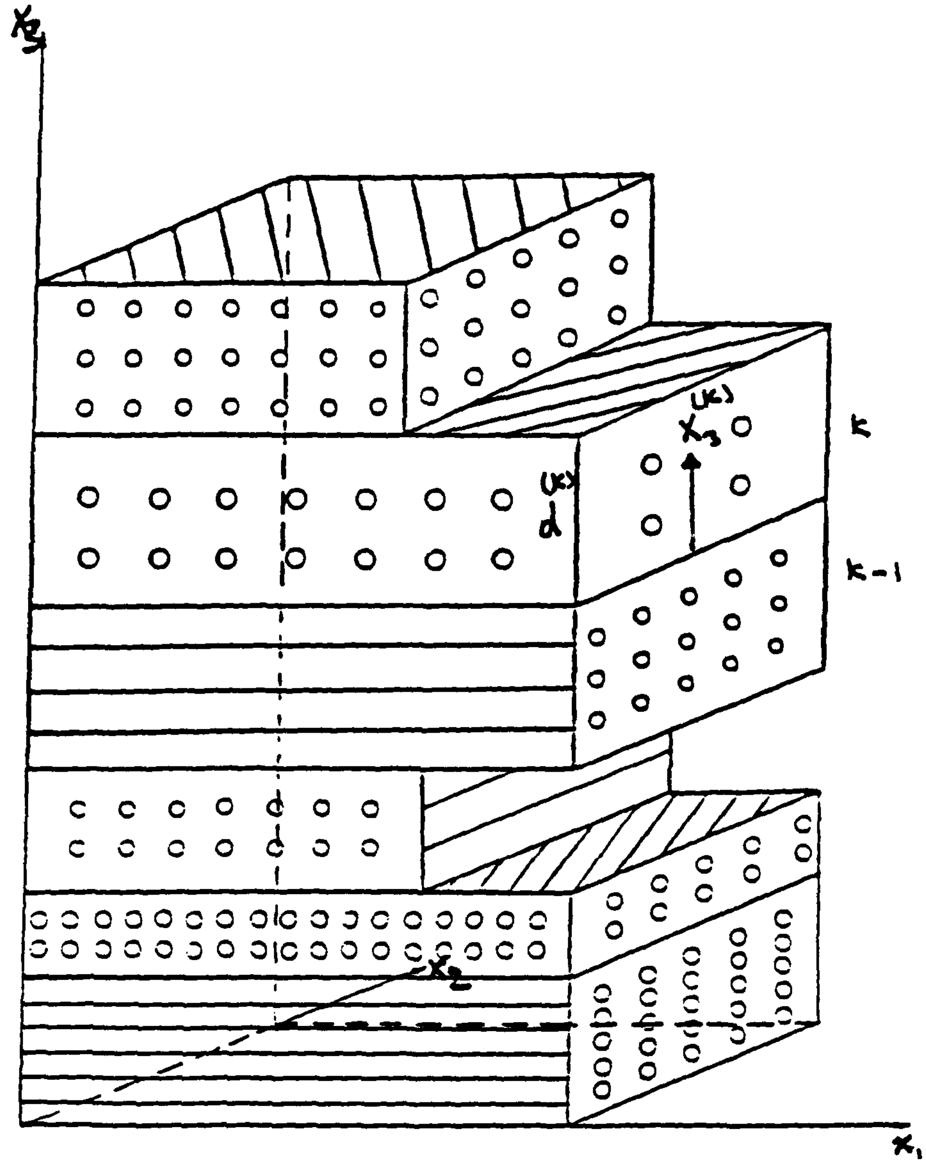


FIGURE 1

Schematic Representation Of A Layered Composite Where d Is The Thickness Of A Layer k And Adjacent Layers Become k-1 or k+1, Etc.

TI-6Al-4V/SCS-6
Fiber volume fraction FF : 0.35
Direction of the ply: 0°

	C1	C2	C3	C4	C5	C6
C1	36.62	11.32	11.32	0.00	0.00	0.00
C2	11.32	32.28	15.58	0.00	0.00	0.00
C3	11.32	15.58	32.28	0.00	0.00	0.00
C4	0.00	0.00	0.00	8.36	0.00	0.00
C5	0.00	0.00	0.00	0.00	9.85	0.00
C6	0.00	0.00	0.00	0.00	0.00	9.85

TABLE 2a.- Material Constant Matrix For A 35% Fiber Volume Fraction With A Unidirectional Ply Layup, 0°.

TI-6Al-4V/SCS-6
Fiber volume fraction FF : 0.35
Direction of the ply: 45°

	C1	C2	C3	C4	C5	C6
C1	32.73	13.03	13.45	0.00	0.00	1.09
C2	13.03	32.73	13.44	0.00	0.00	1.08
C3	13.45	13.44	32.27	0.00	0.00	2.13
C4	0.00	0.00	0.00	9.11	-0.74	0.00
C5	0.00	0.00	0.00	-0.74	9.11	0.00
C6	1.09	1.08	2.13	0.00	0.00	11.56

TABLE 2b.- Material Constant Matrix For A 35% Fiber Volume Fraction When A Single Unidirectional Layer Is Rotated From 0° To 45°.

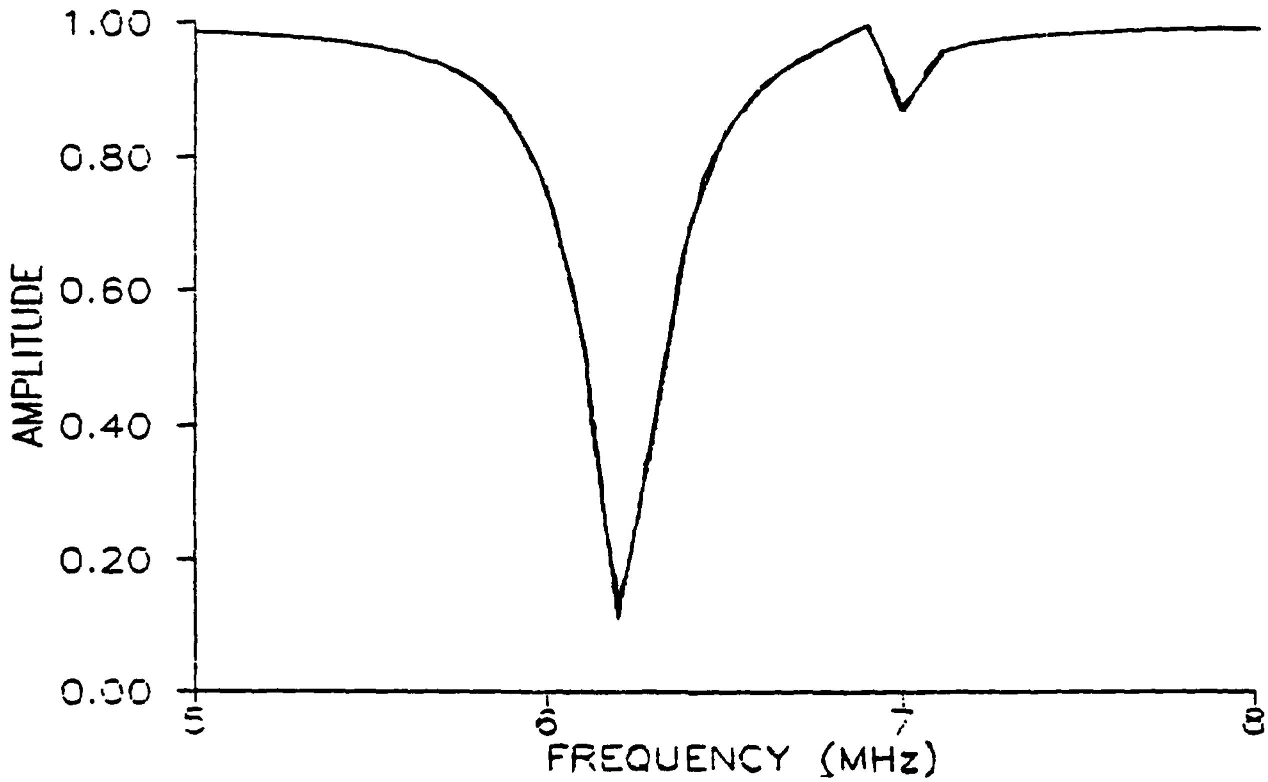


FIGURE 2.

Reflected Amplitude For The Three Layer Metal Matrix Model, Thickness 6 mm (TI-6Al-4V/SCS-6).

FF (v/o)	C11 Long.	C22 Long.	C23 Shear	C13 Shear
0.35	7.979	7.490	3.813	4.138
0.40	8.272	7.737	3.947	4.298
0.45	8.566	7.994	4.092	4.466
0.50	8.861	8.263	4.250	4.643

TABLE 3a. - Variation In The Theoretical Ultrasonic Wave Velocities (km/sec) Obtained Theoretically As The Fiber Volume Fraction Is Changed For A Unidirectional Layer (TI-6A1-4V/SCS-6).

FF (v/o)	C11 Long.
0.35	6.81

TABLE 3b. - Experimental Ultrasonic Wave Velocity (km/sec) For The C11 Constant (Longitudinal Mode) For A Unidirectional Layer With A Fiber Volume Fraction Of 35% (TI-6A1-4V/SCS-6).

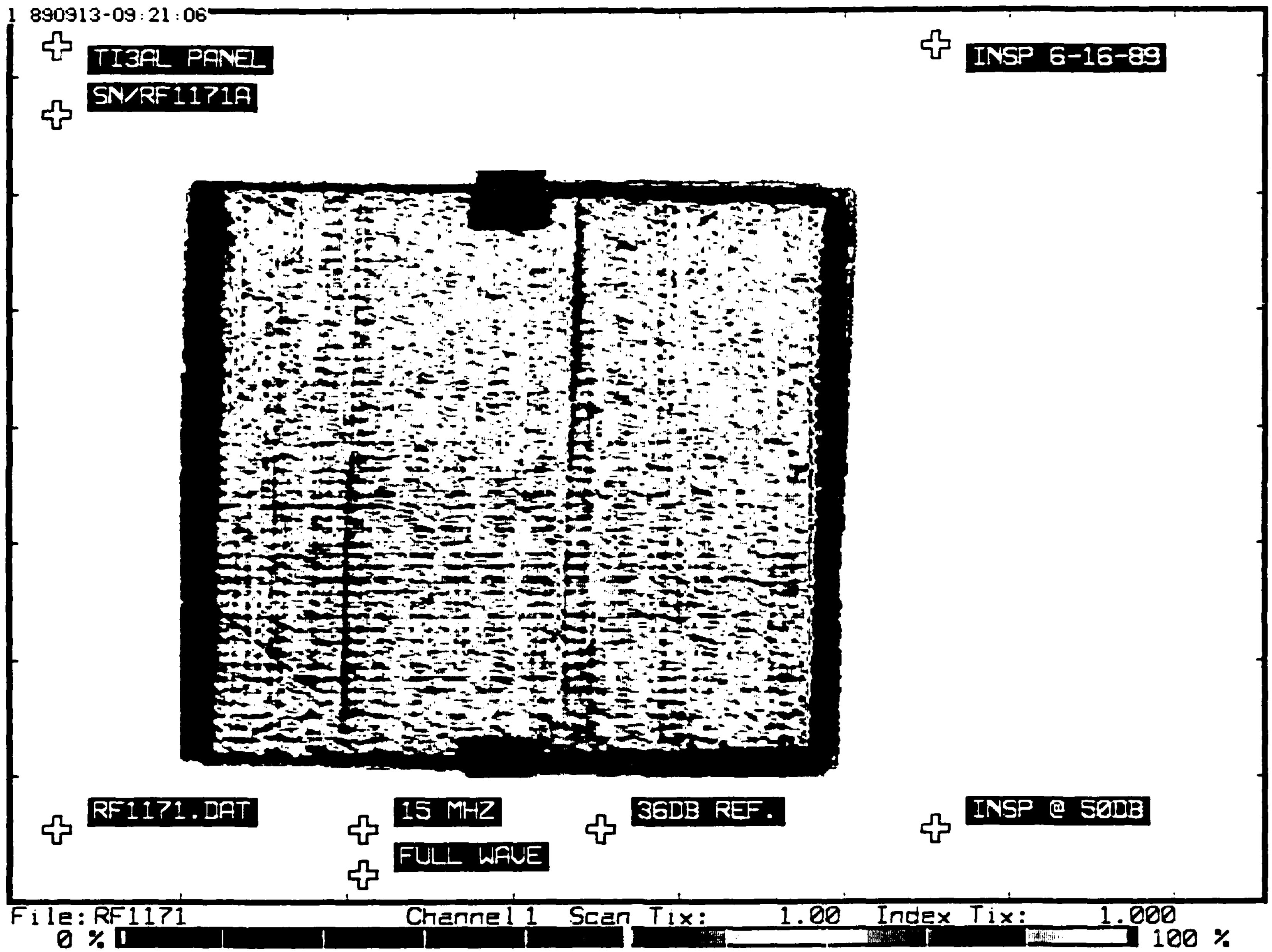

1 890913-09:21:06
TI3AL PANEL
SN/RF1171A
INSP 6-16-89
RF1171.DAT
15 MHZ
FULL WAVE
36DB REF.
INSP @ 50DB
File:RF1171 Channel 1 Scan Tix: 1.00 Index Tix: 1.000
0 %
100 %

Figure 3

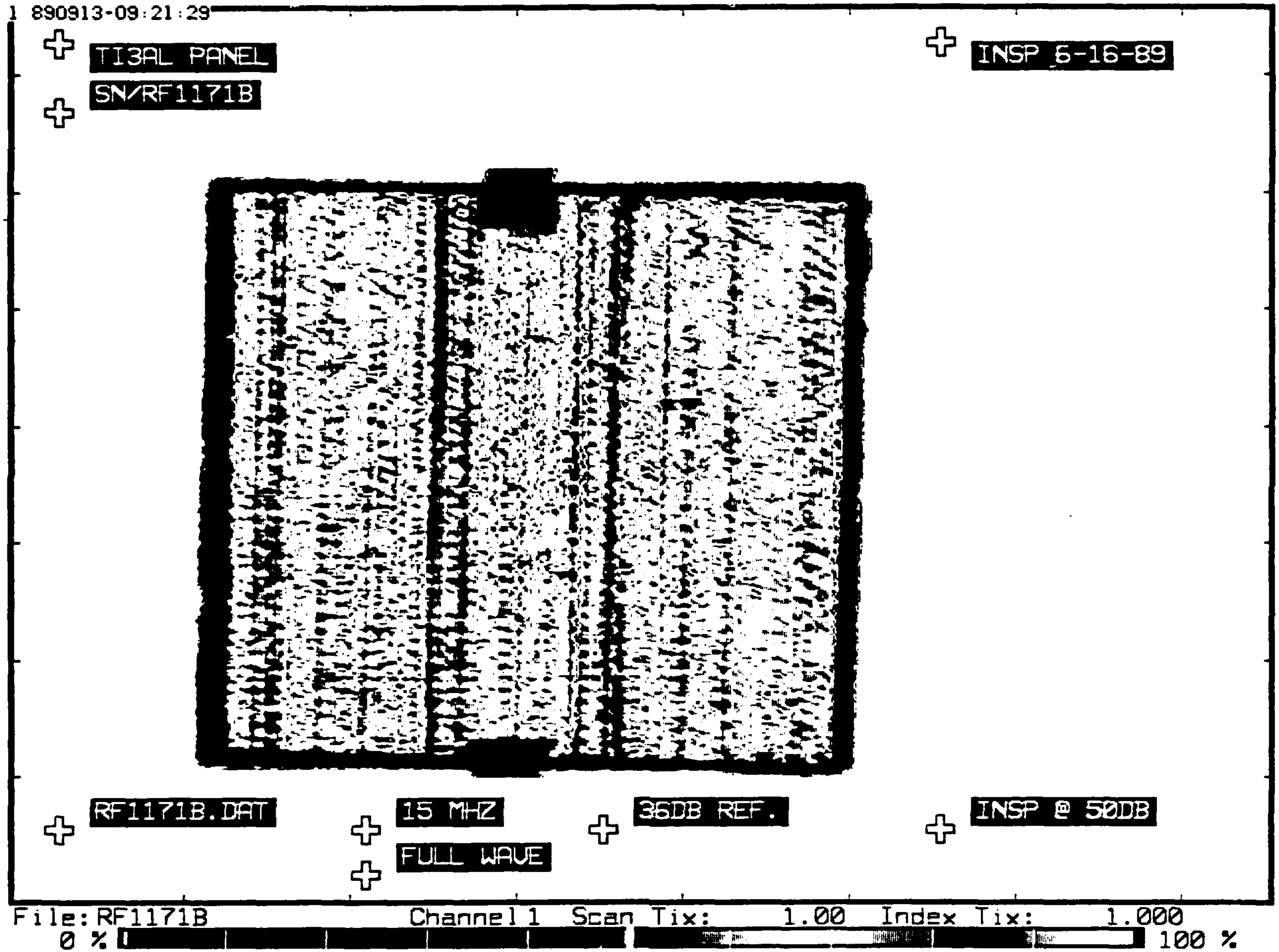

1 890913-09:21:29
TI3AL PANEL
SN/RF1171B
INSP 6-16-89
RF1171B.DAT
15 MHZ
FULL WAVE
36DB REF.
INSP @ 50DB
File:RF1171B Channel 1 Scan Tix: 1.00 Index Tix: 1.000
0 %
100 %

Figure 4

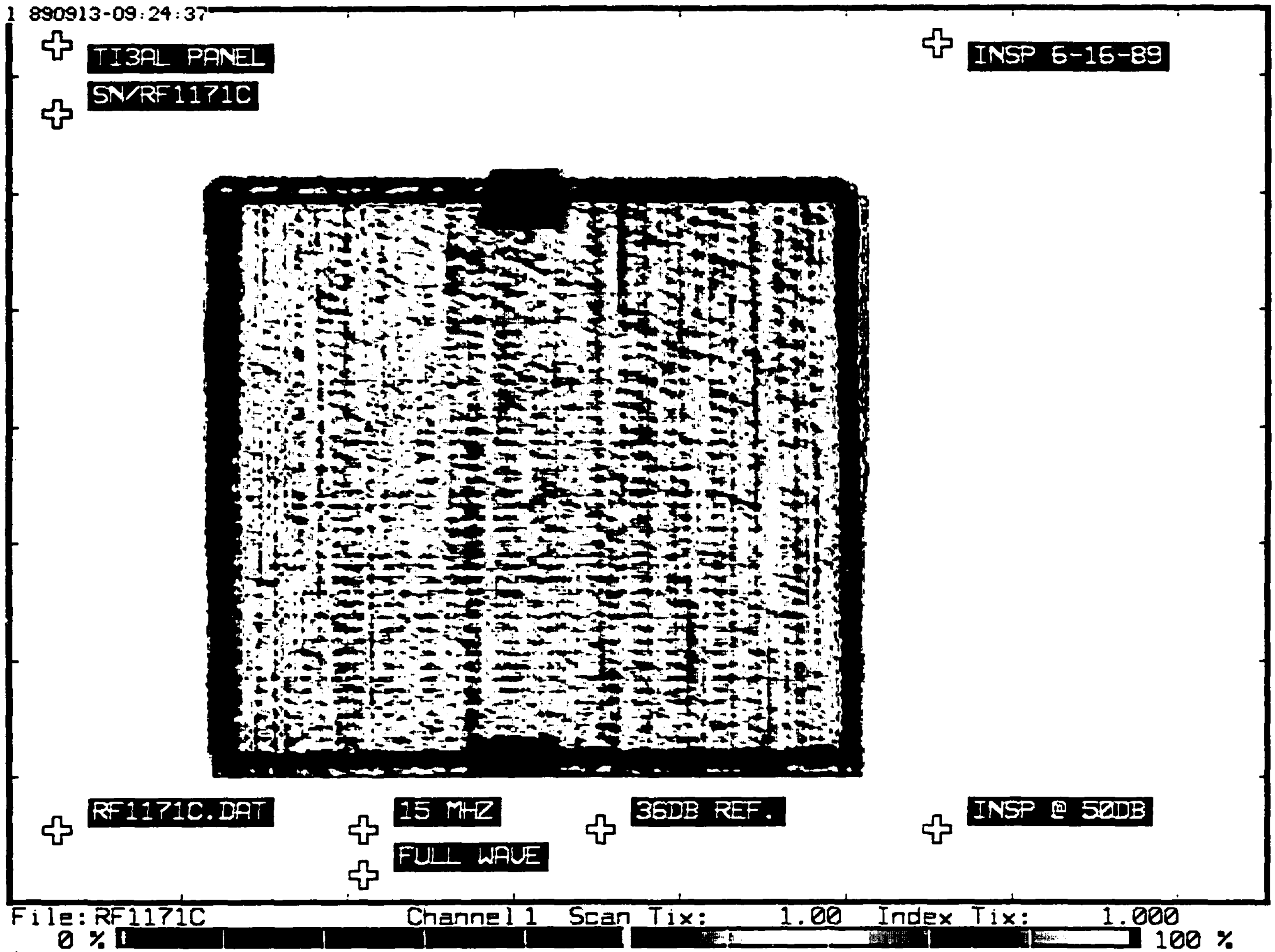

1 890913-09:24:37
TI3AL PANEL
SN/RF1171C
INSP 6-16-89
RF1171C.DAT
15 MHZ
FULL WAVE
36DB REF.
INSP @ 50DB
File:RF1171C
Channel 1 Scan Tix: 1.00 Index Tix: 1.000
0 %
100 %

Figure 5

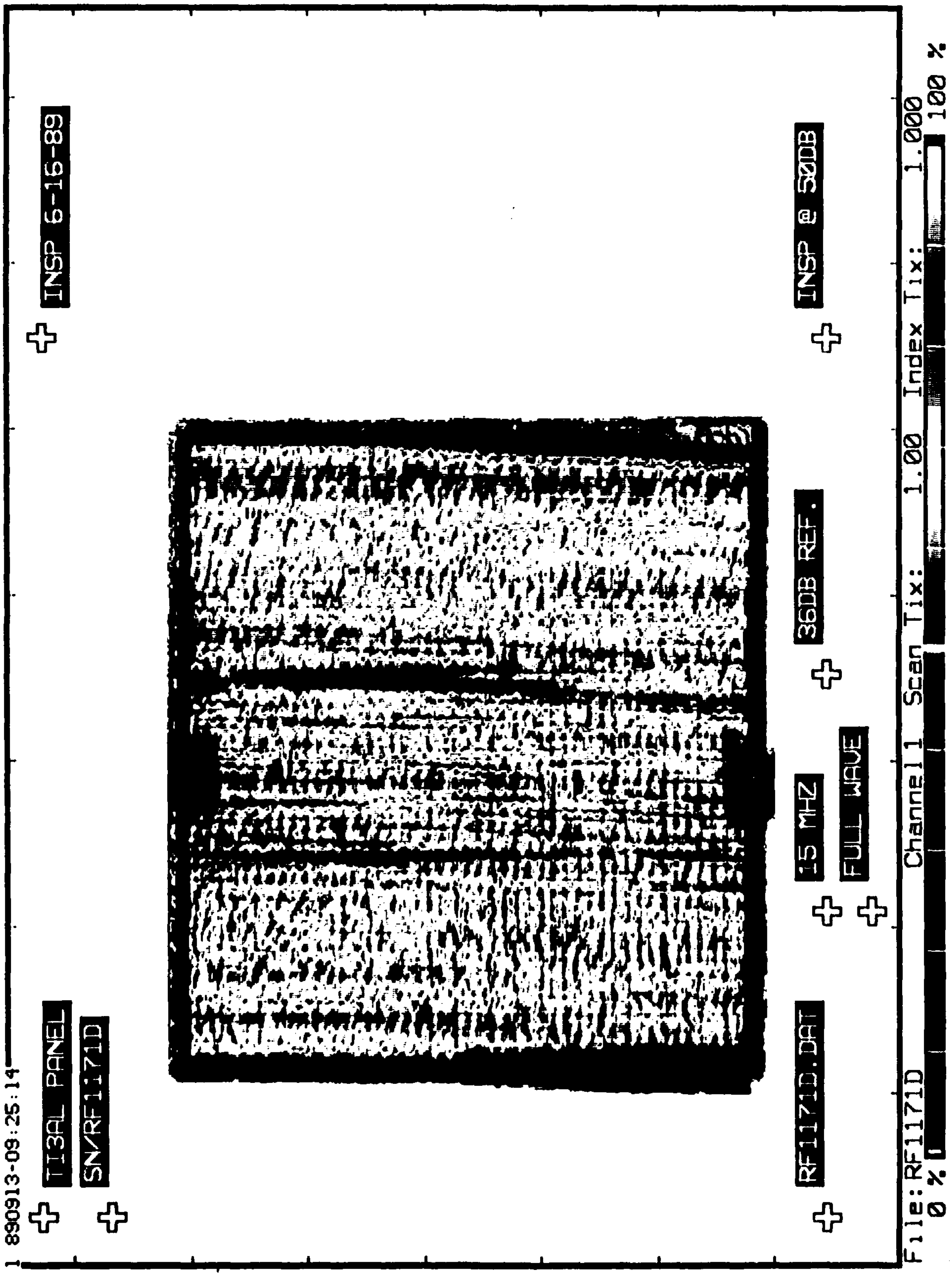

1 890913-09:25:14
TI3AL PANEL
SN/RF1171D
INSP 6-16-89
RF1171D.DAT
15 MHZ
FULL WAVE
36DB REF.
INSP @ 50DB
File: RF1171D Channel 1 Scan Tix: 1.00 Index Tix: 1.000
0 %
100 %

Figure 6

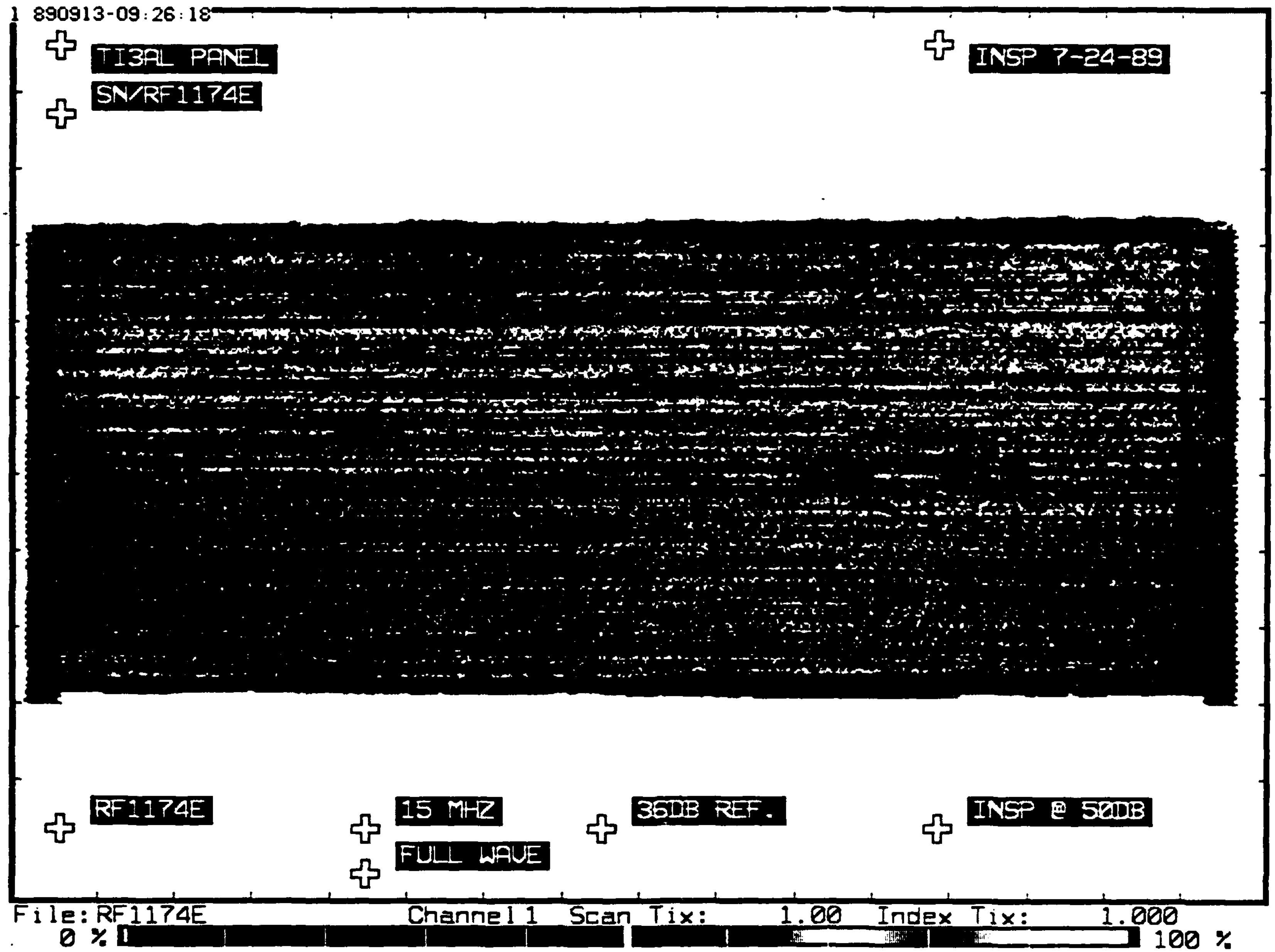

1 890913-09:26:18
TI3AL PANEL
SN/RF1174E
INSP 7-24-89
RF1174E
15 MHZ
FULL WAVE
36DB REF.
INSP @ 50DB
File:RF1174E
Channel 1
Scan Tix: 1.00
Index Tix: 1.000
0 %
100 %

Figure 7

DICE PROGRAM

Task 3.5.8.2 Quality Control Advisor WVU

A. Objectives:

The major objectives of this task are:

1. To provide advice on quality assurance, in a timely manner, to designers working in a concurrent engineering environment. This advice includes quality assurance procedures that are applicable both at the design and at the manufacturing stages of the product development cycle and
2. To develop a computer aided planner and analyzer for the Designing of Experiments (DOE) that lead to reduced product and process variability according to the principles of parameter or robust design.

B. Approach:

Quality assurance advice can best be provided through the development of a "Quality Assurance Advisor (QAA)", which includes both off-line and on-line quality assurance procedures. Off-line procedures are used at the product and process design stages in order that quality related problems can be anticipated and dealt with properly. On-line quality assurance methods, usually referred to as statistical process control (SPC) methods, are implemented at the production stage. SPC methods ensure that design requirements are met at the production stage.

For Phase I, specific tasks were identified to achieve the above objectives:

1. Assess the literature and the work done by other researchers in the areas of "quality by design", "quality at the design stage" and "quality assurance in a concurrent engineering environment",

2. Investigate quality assurance techniques currently applied at the design stage by leading U.S. manufacturers;
3. Design the framework within which the Quality Assurance Advisor can work in a concurrent engineering environment;
4. Incorporate "parameter design" techniques into the DICE program. In the parameter design, the first step is to express the required information by a linear graph and then construct the appropriate orthogonal array which is used to run the experiment. After the experiment, the factors which have a significant effect on the signal-to-noise ratio are determined, using an analysis of variance technique, and
5. Provide other statistical methods of experimental design that can be used in designing quality products in a concurrent engineering environment.

C. Technical Results:

1. A design for the framework of the "Quality Assurance Advisor" (QAA) was completed. The QAA is a comprehensive menu-driven system that helps the user perform off-line and on-line quality assurance analysis. The actual QAA development is still in its early stages. Details of the design are constantly changing as more modules and parts of the QAA are added. Because the X-11 and a User Interface Builder (UIB) were used in the development of the QAA framework, the user will have access to several additional utilities when using the QAA. These utilities include on line help facility, a tutorial and the comparison of various quality assurance procedures;

2. Software was developed to incorporate both off-line and on-line quality assurance procedures in the QAA. Work was initiated in four modules of the design in the experiments area; two have been completed, one module in optimization which is almost complete and one module that is completed in the sampling plans area.
3. Examples were developed and compiled on the use of design of experiments, and Taguchi's "signal to noise" approach in the design of robust products was examined. These examples demonstrate the process of parameter design, determination of tolerances and calculation of loss functions. These examples are documented in a separate technical report.
4. An extensive study of Taguchi's work on parameter design was completed and a prototype of the computer aided planner and analyzer was developed. This Prototype deals with the minimum weight design of a duct. For the duct optimization problem, a complete factorial design requires $2^5 x 4^2$=512 experiments. By using only 16 runs dictated by an orthogonal array, the effects of the controllable factors on the weight of the duct can be estimated. The reduction of the number of runs from 512 to 16 is a major feature of the orthogonal method. Starting with a standard orthogonal array L_{16} (2^{15}), which has 15 two level columns, four level columns can be obtained by combining two appropriate columns and keeping their interaction column empty. The resulting L_{16} ($2^5 x 4^2$) orthogonal array is, then, used to carry out the experiment. The resulting prototype demonstrates the usefulness and the power of parameter design within the concurrent engineering environment. By using parameter design, product design quality can enhance costs, reduce product

development cycle, shorten delays, reduce waste and increase productivity.

5. Some theoretical work was completed on the mathematical modeling of the inspection process and on cost modeling in quality control. This work is not complete and is still under development.

D. Conclusions:

1. Product design can be robust. Specifically, the quality loss function, signal-to-noise ratio and experimental design techniques (in particular, orthogonal arrays) are useful in obtaining robustness. The results clearly indicate that we must continue to build on the so called "Taguchi methods" and other techniques in an effort to produce more robust products;
2. There is a dearth of information available on "quality by design" and "quality at design stage". The available information lacks much detail and experimental results. The bulk of the available information deals with robust design and the Taguchi approach to parameter design;
3. Many companies, including Xerox, AT & T, Bell Labs and Ford Motor Company have applied Taguchi methods on a wide-scale and used off-line methods for parameter design and process optimization, and
4. Taguchi's methods are not the only methods available for off-line quality assurance. Classical methods of process optimization, such as design of experiments, response surface methodology and evolutionary operations can be used at the design stage. These procedures optimize the mean response when several independent variables effect the response individually or interactively. However, these methods do not seek to influence the process variability. These

methods should not be ignored and should be considered along with those of Taguchi.

E. **Recommendations:**

It is recommended that the use of multi-factor experimental techniques be used to help optimize performance and improve product robustness. However, the quality must be put into a product at the design stage. Morever, the development of a computer aided experiment planner and analyzer to improve quality and productivity will also be an important step in enhancing product robustness. Finally, some commercially available software, that have proven to be superior in the area of quality assurance, (such as SAS/QC), should be purchased and installed.

F. **Publications:**

Jaraiedi, M., Iskander, W. H., and Gunel, E., " The Role of Quality Assurance in a Concurrent Engineering Environment", Proceedings of the Annual International Industrial Ergonomics and Safety Conference Cincinnati, Ohio, June, 1989, 831-835. This paper was also presented at the Conference.

G. **Hardware:**

Software entitled, "Design of Experiments" was purchased from Perry Johnson, Inc., 3000 Town Center, Suite 2960, Southfield, MI., 48075.

4.0 Bibliography of DICE Publications

1. "First National Symposium on Concurrent Engineering", *Proceedings,* Concurrent Engineering Research Center, West Virginia University, May 23-24, 1989, Morgantown, WV.

2. "Constraint Management in Concurrent Design", Navinchandra, D., working notes of a workshop on concurrent engineering held at the International Joint Conference of Artificial Intelligence, August 1989.

3. "Interval Approaches to Concurrent Evaluation of Design Constraints", Navinchandra, D. and Rinderle, J.R., *Proceedings of the Concurrent Engineering Symposium*, ASME Winter Annual Meeting, December 1989.

4. "Shape Feature Description and Recognition Using an Augmented Topology Graph Grammer", Pinilla, J.M., Finger, S., and Prinz, F.B., *Proceedings of the 1989 NSF Engineering Design Research Conference*, University of Massachusetts, Amherst, MA, June 11-14, 1989, pp. 285-300.

5. "The Role of Architecture in Computer-Assisted Design Systems", Safier, S. and Fox, M.S., *Proceedings of the 1989 NSF Engineering Design Conference*, University of Massachusetts, Amherst, MA, July 11-14, 1989.

6. "The ROSE-IC Library: A Database of Cooperative Work and Concurrent Engineering in Objective-C" Version 1.0, , Hardwick, M., Spooner, D., Downie, B., Hvannberg, E. Loffredo, D., Mehta, A., *Technical Report,* Rensselaer Design Research Center, June 24, 1989.

7. "Using a Relational Database as an Index to a Distributed Object Database in Engineering Design Systems", Hardwick, M., Samaras, G., *2nd International Conference on Data and Knowledge Systems in Manufacturing and Engineering,* IEEE Computer Society, October 1989.

8. "A Data Translation Tool Design Databases", Spooner, D., Sanderson, D., Charalambous, G., *2nd International Conference on Data and Knowledge Systems in Manufacturing and Engineering,* IEEE Computer Society, October 1989.

9. "A Blackboard Scheme for Cooperative Problem-Solving by Human Experts", Londono, F., Cleetus, K.J., Reddy, Y.V.

10. "Software to Facilitate Concurrent Engineering", Cleetus, K.J., Kannan, R., Londono, F. Reddy, Y.V.

11. "Graph Partitioning Using Annealed Neural Networks", Van den Bout, D.E., and Miller III, T.K., *Proceedings of the IEEE International Conference on Neural Networks,* Volume 1, 1989, pp. 521-528.

12. "A General Framework for Local Interpolation", Chui, C.K. and Diamond, H., *Technical Report,* Texas A&M University Center for Approximation Theory, 1989.

13. "Polar Spline Approximation", Zhang, C.Q., *Technical Report TR88-10,* Department of Statistics and Computer Science, West Virginia University, 1988.

14. "Error Analysis of Polar Spline Approximation", Zhang, C.Q., *Technical Report TR89-2,* Department of Statistics and Computer Science, West Virginia University, 1989.

15. "Polar-Splines and Offsets", Zhang, C.Q., *Technical Report TR89-3,* Department of Statistics and Computer Science, West Virginia University, 1989.

16. "Polar Splines Approximation", Zhang, C.Q., submitted to *SIAM,* 1989.

17. "Existence of Offset", Zhang, C.Q., *Technical Report,* CERC, 1989.

18. "Polar Splines in 3-Dimensional Space", Zhang, C.Q., preprint.

19. "Parametric Design of Turbine Blades for Concurrent Engineering Environment", Ruth, Mary K., Masters Thesis, Department of Mechanical and Aerospace Engineering, West Virginia University, May 1989.

20. "Prediction of Site Selection for Additives to Intermetallic Compounds", Khowash, P.K., Price, D.L., Cooper, B.R., submitted paper to Physical Review Letters.

21. "Concurrent Engineering Workstation for High Density Electronics", Matsumoto, A.S., Jagannathan, V., Buenzli, C., Saks, V., IEEE Workshop on Design for Testability, April 1989, Vail Colorado.

22. "Concurrent Design for Testability", Matsumoto, A.S., Jagannathan, V., Buenzli, C., Saks, V., Concurrent Engineering Design workshop, IJCAI 1989.

23. "HDE Architecture Requirements Analysis", Jagannathan, V. (editor), report prepared by Cimflex Teknowledge, January 17, 1989.

24. "The Role of Quality Assurance in a Concurrent Engineering Environment", Jaraiedi, M., Iskander, W.H., and Gunel, E., *Proceedings of the Annual International Industrial Ergonomics and Safety Conference,* Cincinnati, Ohio, June 1989, pp. 831-835.