110057

N95-16472

# COMPUTER TOOLS FOR SYSTEMS ENGINEERING AT LARC

J. Milam Walters, Systems Engineering Office
Aerospace Mechanical Systems Division

The Systems Engineering Office (SEO) has been established to provide lifecycle systems engineering support to LaRC projects. Over the last two years, the computing market has been reviewed for tools which could enhance the effectiveness and efficiency of activities directed towards this mission. A group of interrelated applications have been procured, or are under development including a requirements management tool, a system design and simulation tool, and a project and engineering database. This paper will review the current configuration of these tools and provide information on future milestones and directions.

# The Role of Computers In LaRC R&D

## Computer Tools for Systems Engineering


Presented by

J. Milam Walters


June 16, 1994

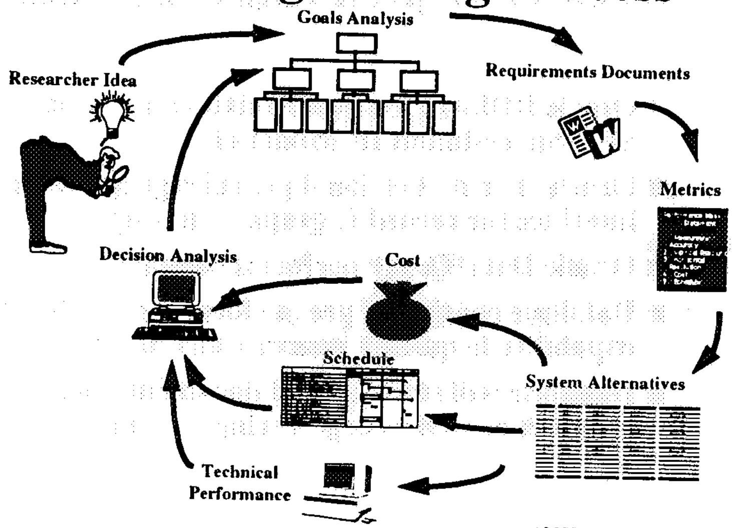
*AMSD Systems Engineering Office*

# AMSD Systems Engineering Office

- Established via Center Reorganization after approximately 3 years of ground laying
- Current staffing level - 5 CS
- Chartered to guide application of systems engineering to LaRC flight projects
- Process detailed in LHB 7122.1, currently in approval cycle
- Process applied to various projects, most recently JADE and SABER

*AMSD Systems Engineering Office*

# Systems Engineering Process




*AMSD Systems Engineering Office*

# Systems Engineering Office Tools

- **Workstation Based tools consist of the following:**
  - Oracle SE Project & Engineering Database
  - Excalibur Scanning/Recoginition Software
  - RTM (Requirements Traceability & Management)
  - RDD-100 (Requirements Driven Design)
  - Interleaf 6.0
  - Matlab with following toolboxes:
    - Simulink option
    - System Identification
    - Control System
    - Optimization

*AMSD Systems Engineering Office*

## Oracle SE Project & Engineering Database

- Oracle RDBMS Version 7 relational database on SUN Sparcstation 10, Model 41
- Oracle*Forms Version 4 provides graphical user interface for record & graphic viewing
- Oracle Data*Query performs complex searches
- Database consists of pre-set form types, with the capability to quickly generate any new table type
- Database will store project documentation and graphics as well as engineering data tables

*AMSD Systems Engineering Office*

## Excalibur Scanning/Recoginition Software

- Interfaces with document scanner to read and interpret input
- Contains an adaptive search engine to retrieve desired document
- Displays original image upon match of a given search
- Provides the capability to scan and store input documents

*AMSD Systems Engineering Office*

## RTM Requirements Traceability & Management

- Application developed especially for tracking and managing project requirements
- Utilizes Oracle database to store requirement information
- Provides special tools for:
  - extracting requirements from source documents
  - expanding and focusing requirements
  - general requirements maintenance
- Includes output bridges to RDD-100 and Interleaf

*AMSD Systems Engineering Office*

## RDD-100 Requirements Driven Design

- Facilitates the construction, maintenance, display, and documentation of design objects that specify behavior
- Objects are created and edited by graphics or text, with multiple generated views available to gain different perspectives
- Includes a simulator which directly executes the design objects
- Templates and consistency checks verify system design sufficiency
- Bridge to Interleaf is included

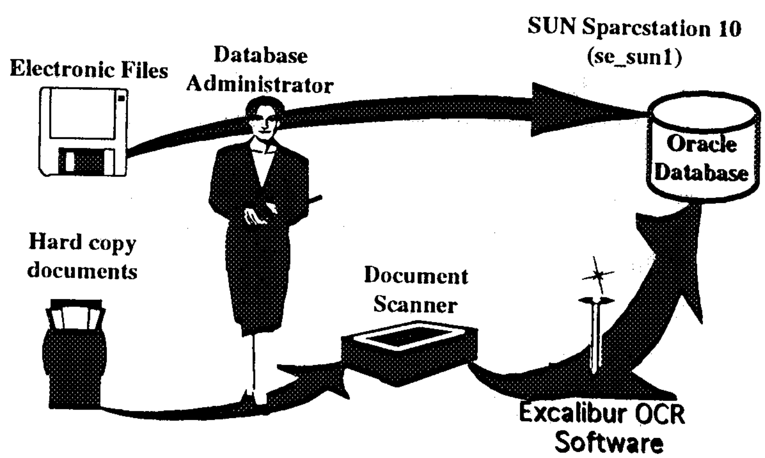
*AMSD Systems Engineering Office*

# Matlab

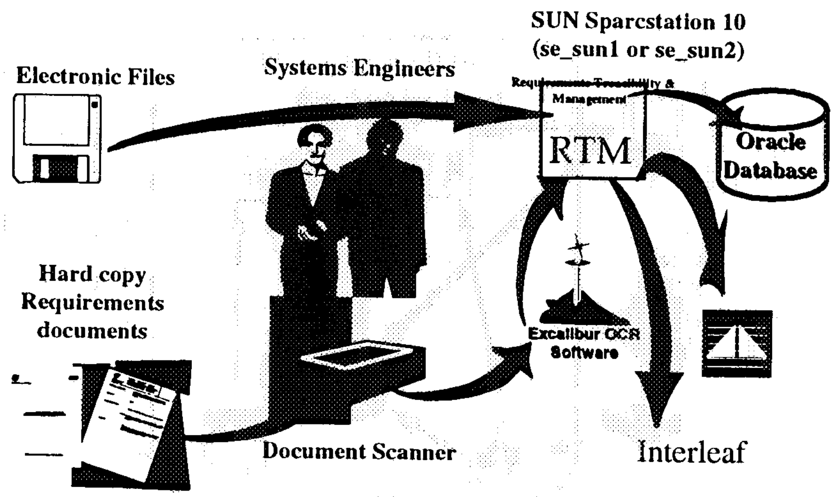
- Interactive software program for scientific and engineering numeric computation
- Combines numerical analysis, matrix computation, signal processing, and graphics with a user interface through standard math notation.
- Functions include differential equation solution, polynomial operations, matrix computation, complex arithmetic and signal processing tools
- To view data graphically, MATLAB provides 2-D linear, log, semilog, and polar plots, and 3-D mesh and contour graphs
- Works with MATLAB numeric computation software package to build mathematical models of



# Database Population

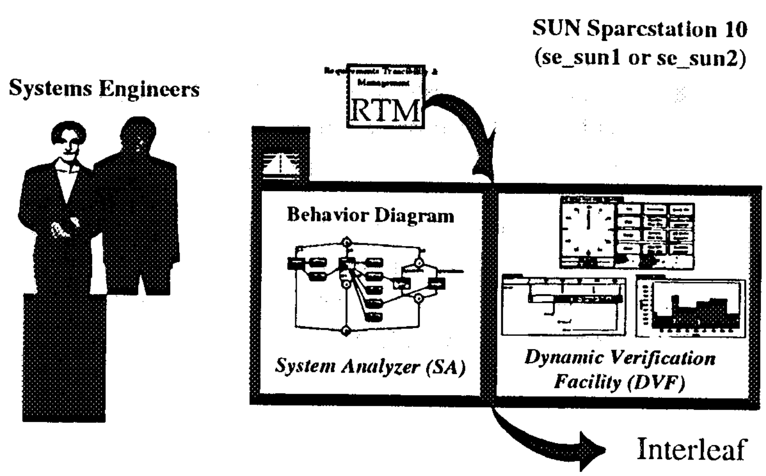


# Requirements Management



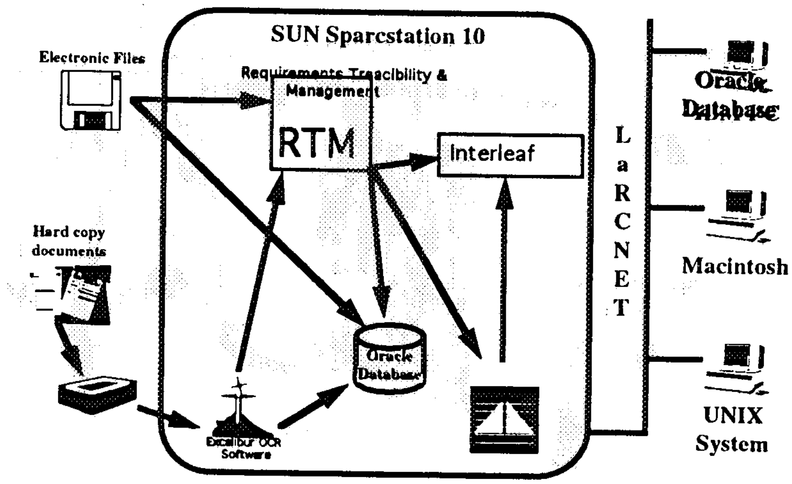

*AMSD Systems Engineering Office*

# System Modeling & Simulation




*AMSD Systems Engineering Office*

# Tool Interface Overview




*AMSD Systems Engineering Office*

# Summary

- The Systems Engineering Office of AMSD has been established to provide for computer aided:
  - systems level behavior modeling and simulation of new concepts (RDD-100)
  - subsystem mahematical modeling and simulation (Matlab)
  - requirements tracking and management (RTM)
  - storage of project and engineering documentation (SEDB)
- Interested parties should contact Richard Foss at 4-7049 or Milam Walters at 4-3014

*AMSD Systems Engineering Office*

## A Distributed Computing Environment for Multidisciplinary Design

The Framework for Interdisciplinary Design Optimization (FIDO) project has the goal of developing a general distributed computing system for executing multidisciplinary computations on a networked heterogeneous cluster of workstations and vector and massively parallel computers. This project is a part of the Computational Aerosciences (CAS) project in the High Performance Computing and Communications (HPCC) program. The FIDO system provides a means for automating the total design process. It facilitates communication and control between components of the system, which include the diverse discipline computations involved in a design problem and the system services that facilitate the design. In its current state of development, the prototype FIDO system is being applied to a token example of the optimized design of a high-speed civil transport (HSCT), involving a simplified problem that includes the disciplines of aerodynamics, performance, propulsion, and structures, but with very few design variables. However, it has already demonstrated the ability to coordinate multidisciplinary computations and communications in a heterogeneous distributed computing system.

The concept being used in FIDO is course-grained parallelism, with instances of the disciplinary codes (aero, structures, etc.) running on separate processors, under control of an executive on another processor, and exchanging data through a single data base manager (on yet another processor). To allow the user to monitor the progress of the design iterations, there is a graphical user interface (which tracks the execution of codes performing the design iterations) and a separate process, called SPY, which allows its user to extract and plot data produced during current and previous design cycles. In fact, multiple instances of SPY can be executing at once, so that the designer can call on discipline experts and they (possibly from some remote location on the Internet) can examine the results being produced and provide advice. SPY is currently being upgraded to provide the capability for the designer to make changes in variable values during execution and so guide the design process.

The distributed computing system currently includes Sun, Silicon Graphics, and Digital Equipment Co. workstations. Provision has been made for adding connections to Cray vector computers and Intel parallel computers, and preliminary checks of connection procedures have been carried out. A communications library has been written (implemented using the PVM basic library developed at Oak Ridge National Lab) to provide the versatility for transferring data packages ranging from single variables or file names to large data arrays.

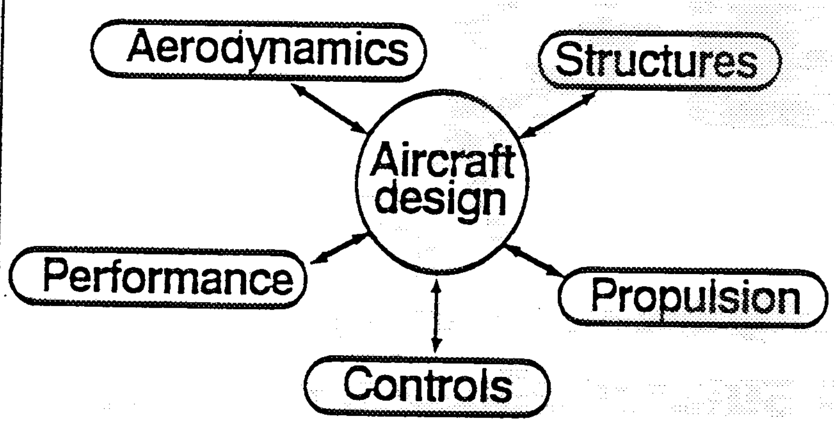
The current Motif-based Graphical User Interface (GUI) consists of three separate elements: setup, application status, and data display. The setup GUI provides the user with a convenient means of choosing the initial design geometry, material properties, and run conditions from a predefined set of files. The interface displays the choices using a series of pop-up Motif data windows, and allows the user to modify and store new condition files. The application status GUI allows the user to monitor the status of a design run. An example of this display is shown in the slide entitled "FIDO User Interface Concept". Within the screen on the left part of the slide, the upper left window displays current run parameters and contains pull-down menus for setting various options. The right window graphically displays the state of the overall design process by changing the color of each labelled box accoreding to the work being done. A color key is shown in the lower left window. Additional detail of the system state can be obtained by selecting the boxes with a 3-D appearance. Doing so brings up an associated window that displays sub-detail for that box. The data display GUI is the third interface element, called SPY, that provides the user with a variety of ways to plot data during the design process. The right part of slide is an example of a color-coded contour plot of wing surface pressures. The buttons at the top of the plot window provide the user with a variety of view controls. In addition to contour plots of aerodynamic pressures and structural stresses on the wing, SPY provides line-plots of cycle history for a variety of design parameters and data results. An example of this for a 20-cycle design iteration is shown in a later slide. This slide illustrates results of using FIDO to minimize the total weight of the HSCT for a 6000-mile range and Mach 2.4 cruise speed. Discipline sensitivity derivatives are calculated using finite differences, and a linearized model based on these is used with the optimization subroutine CONMIN to determine new values for the design variables at each design cycle. The slide illustrates the component weight reduction over a 20-cycle design iteration subject to design constraints on structural stress and deflection. Weights have been non-dimensionalized by nominal payload. The slide also shows the changes to the three aerodynamic design variables (lengths non-dimensionalized by wing span) and two structural design variables (non-dimensionalized by initial skin thickness).

# The NASA Langley FIDO project

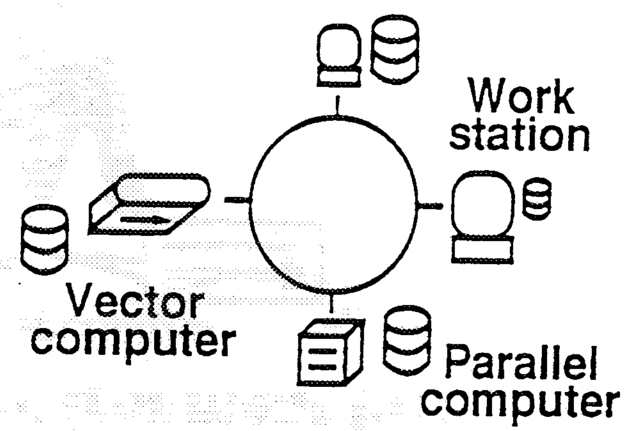
## FIDO - Framework for Interdisciplinary Design Optimization

GOAL: Establish a general computational framework, consisting of software tools and programming guidelines, for facilitating the interdisciplinary coupling necessary for multidisciplinary design and optimization problems on heterogeneous computing systems.

Interdisciplinary problem



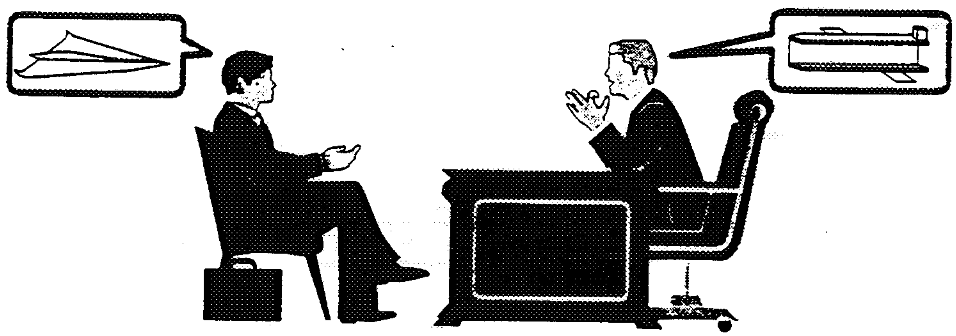

Environment



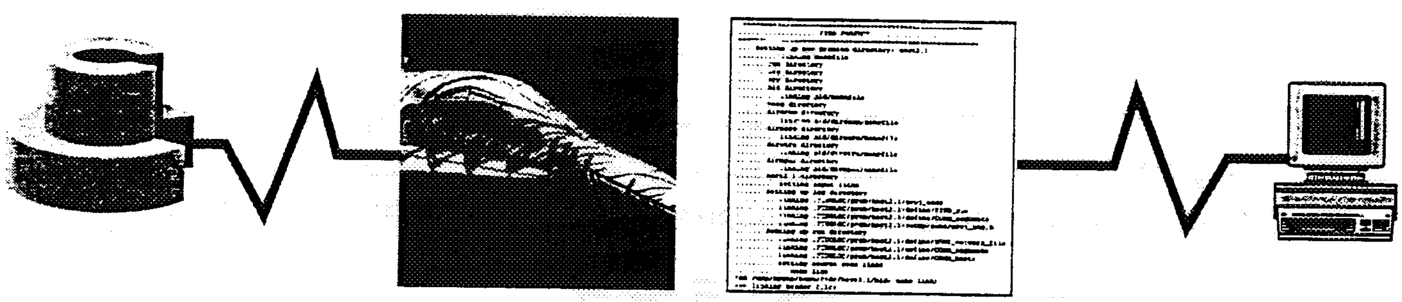
# Concept

## Develop a Programming Environment for:

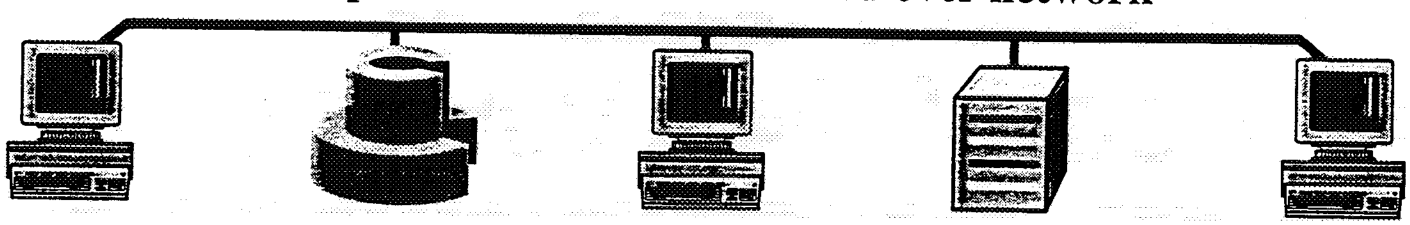
Group of users with diverse specialties



Multiple programs targeted for appropriate computers

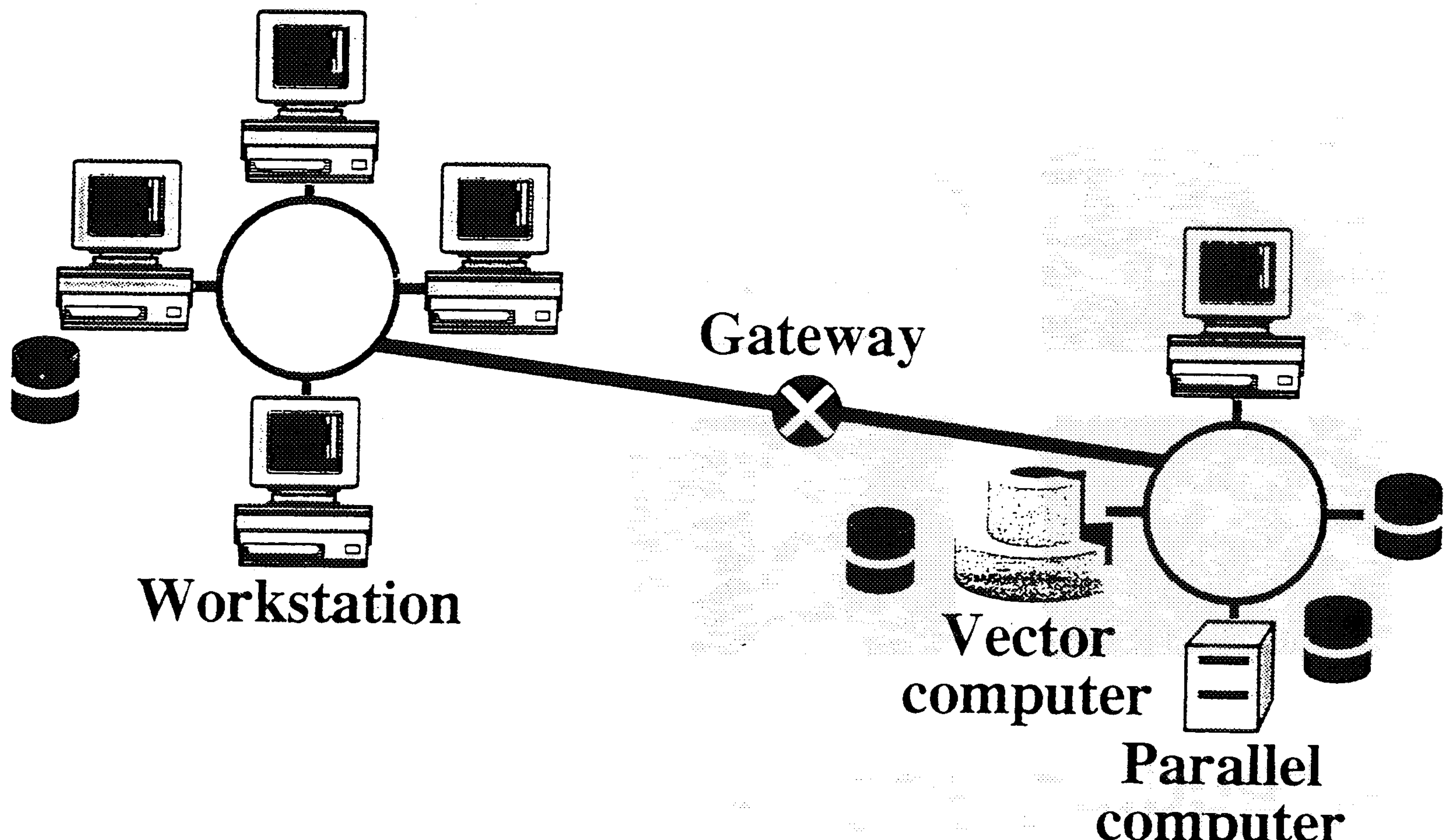


Computers and data distributed over network

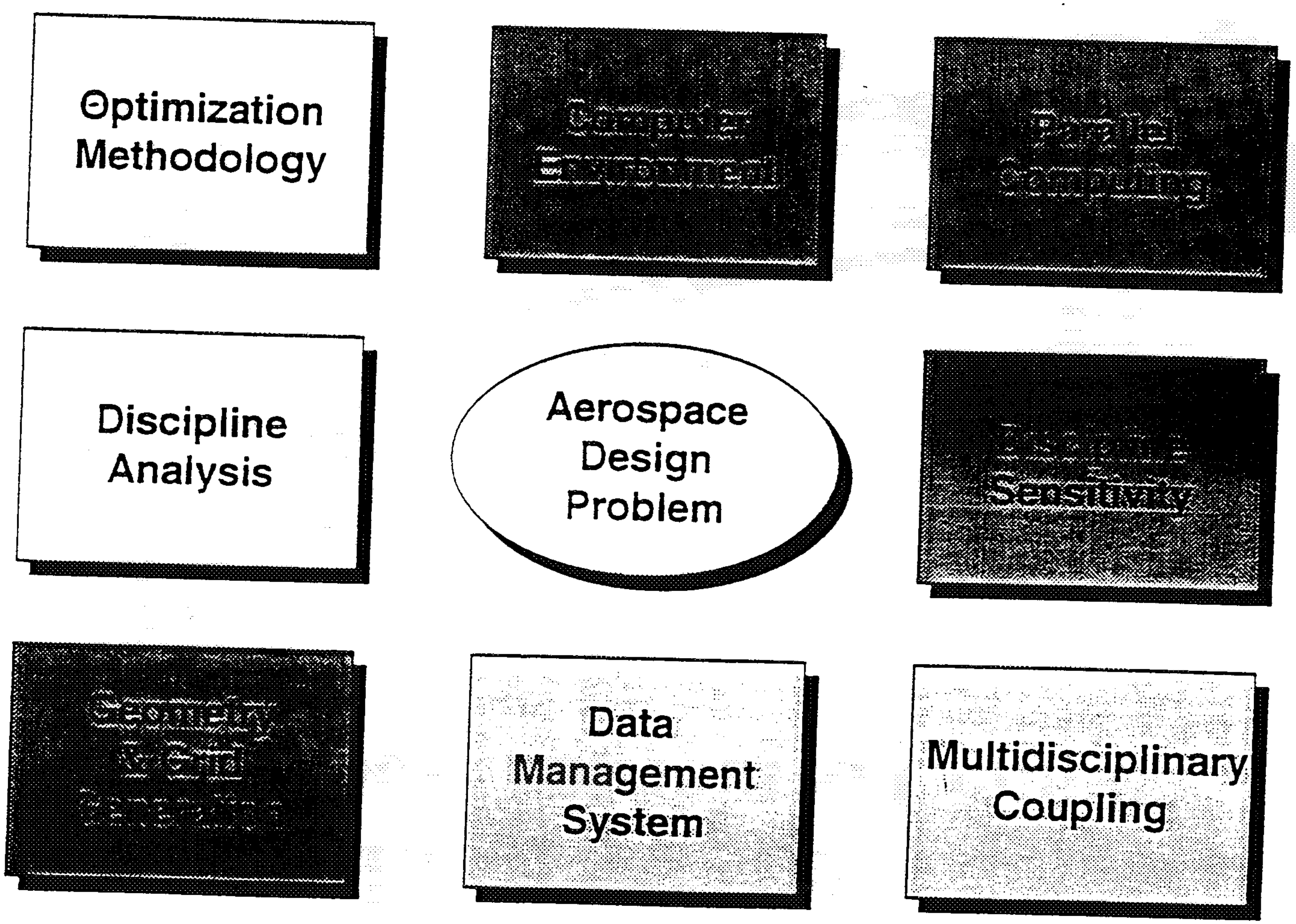


Working together simultaneously on parts of the same problem

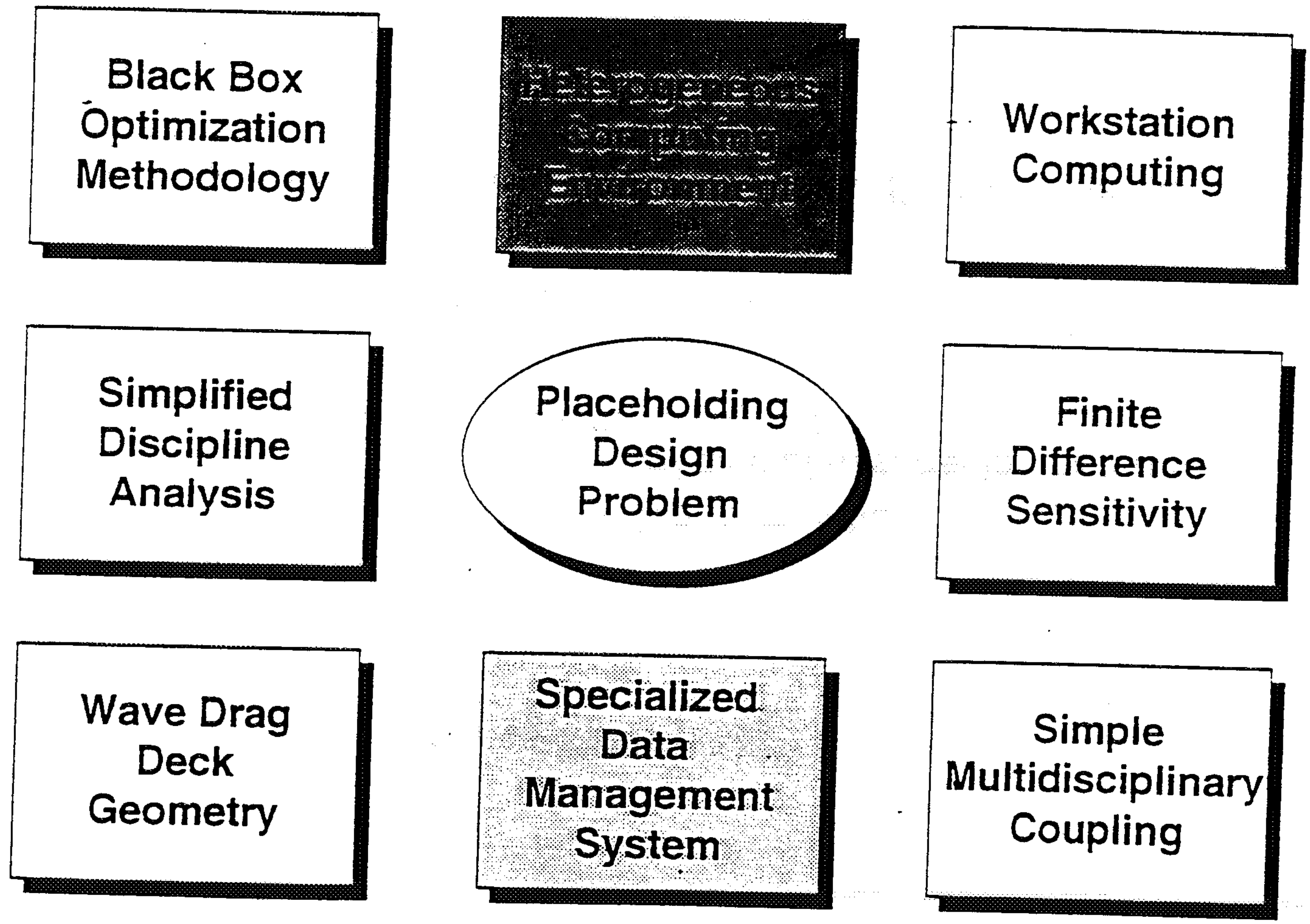
# Environment
## Heterogeneous Distributed Computing



# Current LaRC HPCCP Emphasis



# Current FIDO Components



# Approach

Limit scope of design/optimization formulation.

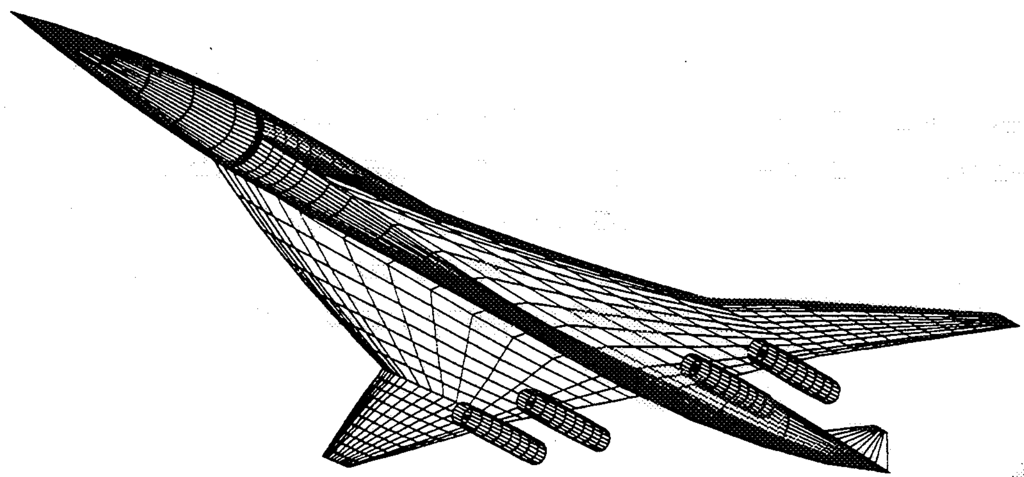
Consider a representative set of disciplines, design criteria, design conditions, design variables, etc.

Use existing engineering analysis codes.

Use a High Speed Civil Transport (HSCT) configuration as the focus problem.

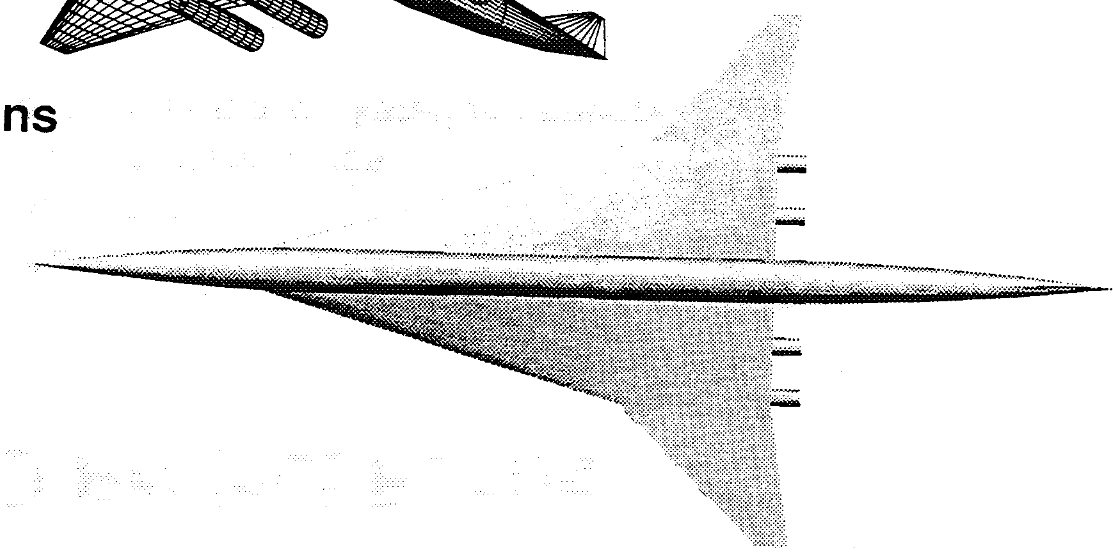
Initial implementation on a distributed workstation environment; migration to parallel testbed in the future

# HSCT Baseline Description



**Design Conditions**

Mach 2.4
Altitude 63,000 ft.
Range 6,000 mi.
Payload 30,000 lbs
Length 300 ft.
Max Load 2.5g



# FIDO Project Focus

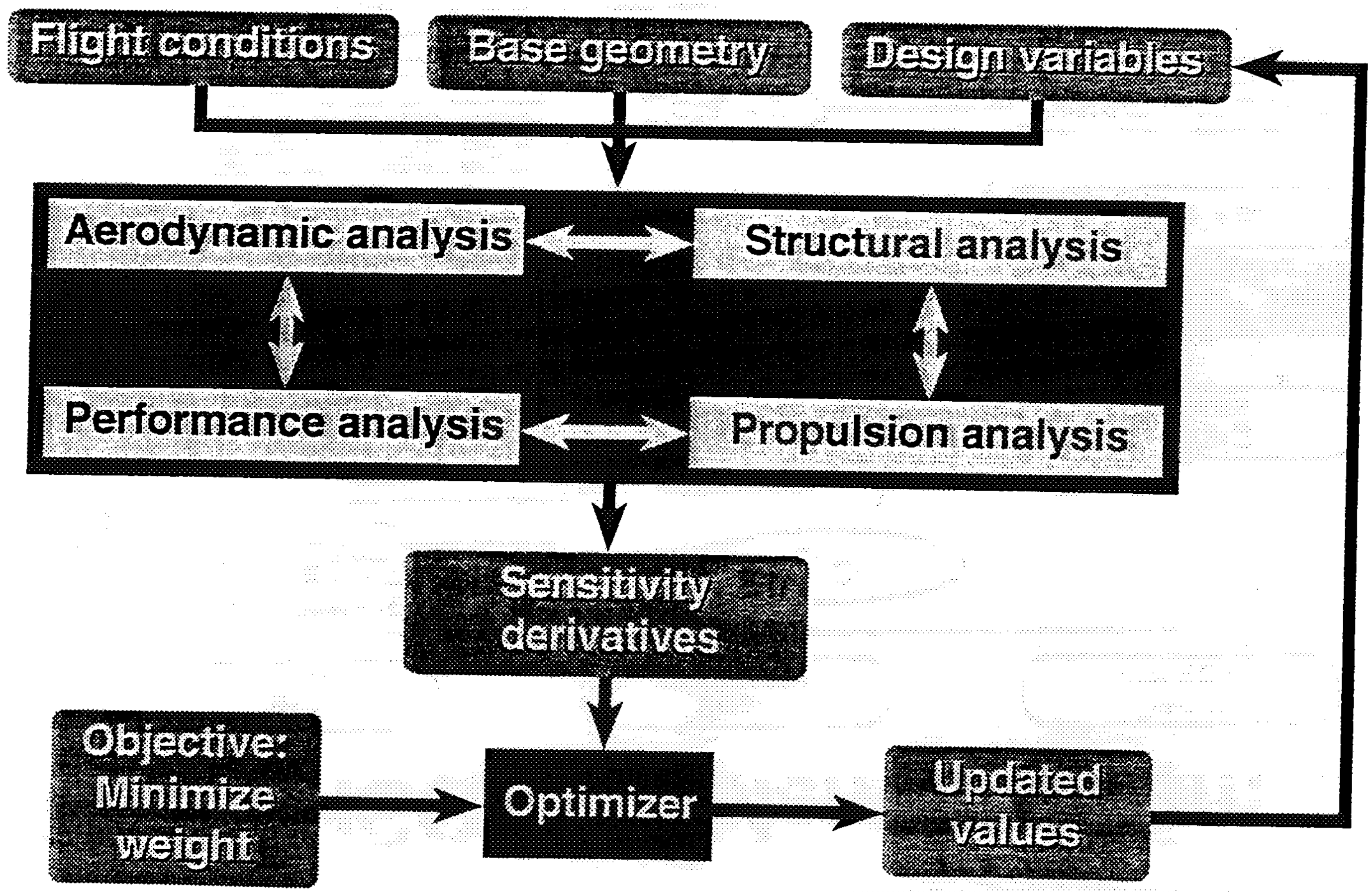
Short term

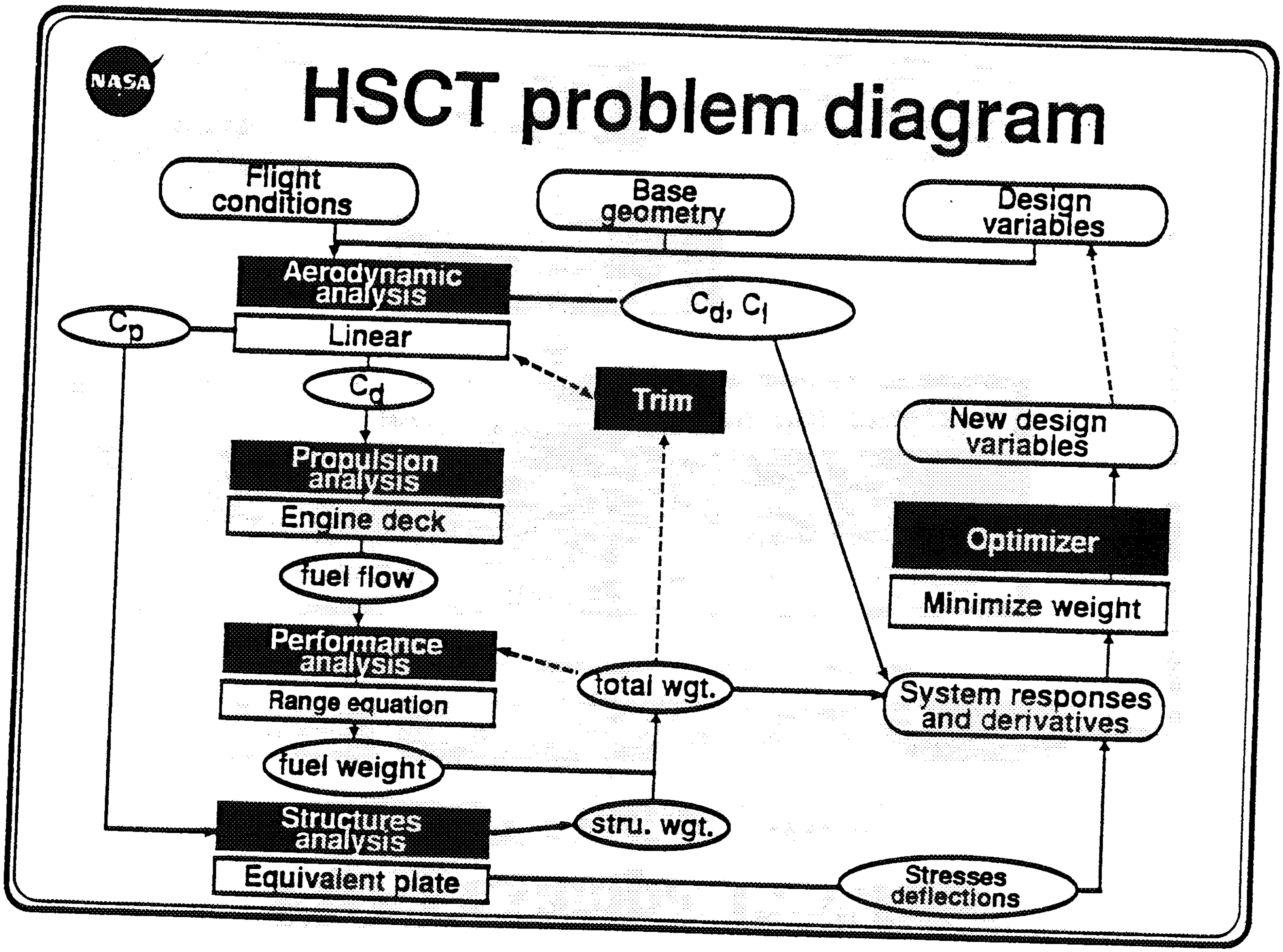
- simplified model problem
- low-fidelity discipline codes
- finite-difference sensitivity derivatives
- coarse-grain parallelization on a workstation network

Long Term

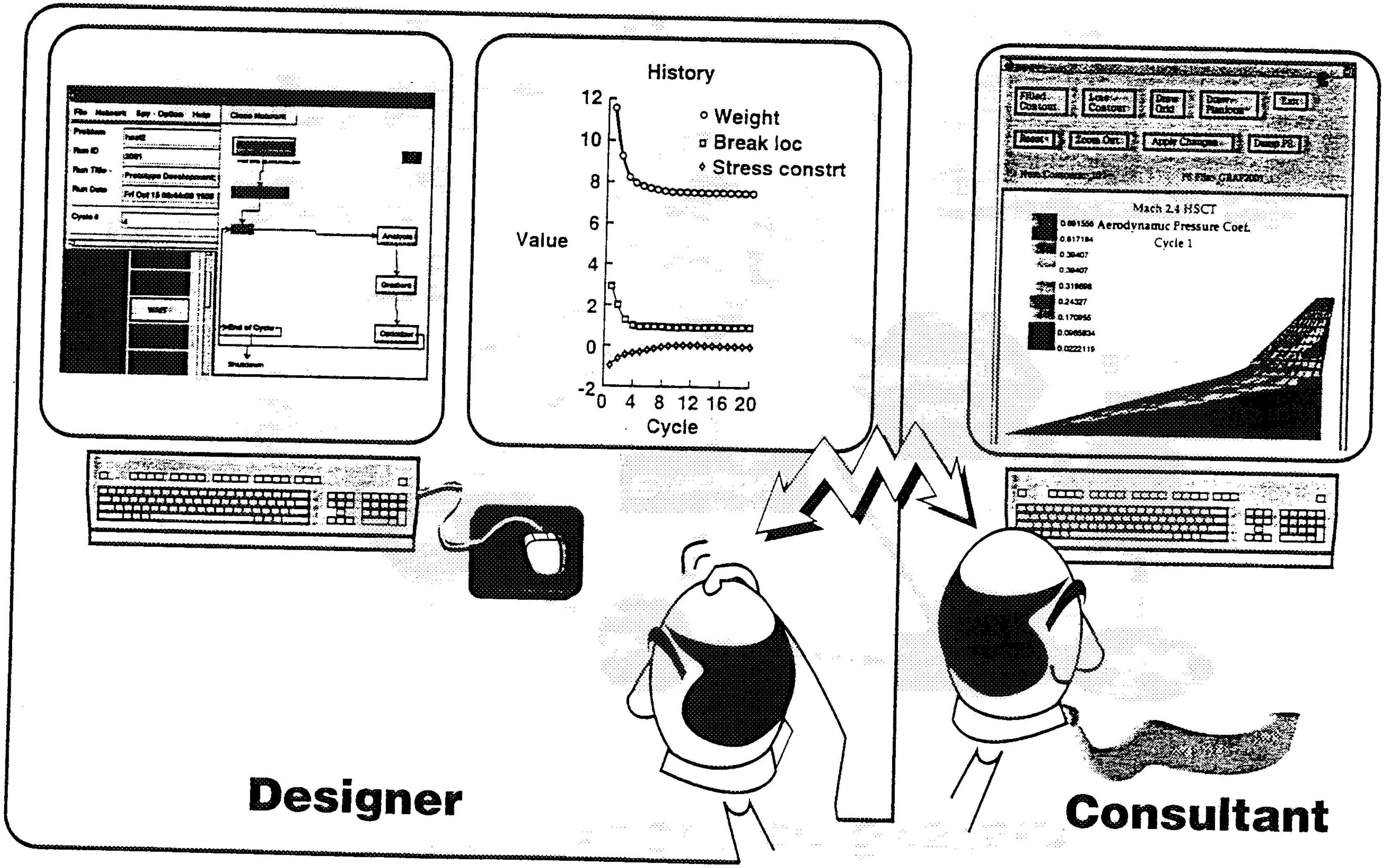
- realistic design and optimization problem
- higher fidelity discipline codes
- quasi-analytic sensitivity derivatives
- coarse-grain parallelization on a heterogeneous computer network along with fine-grain parallelization on massively parallel and vector computers

# FIDO Example Problem
## HSCT Optimization

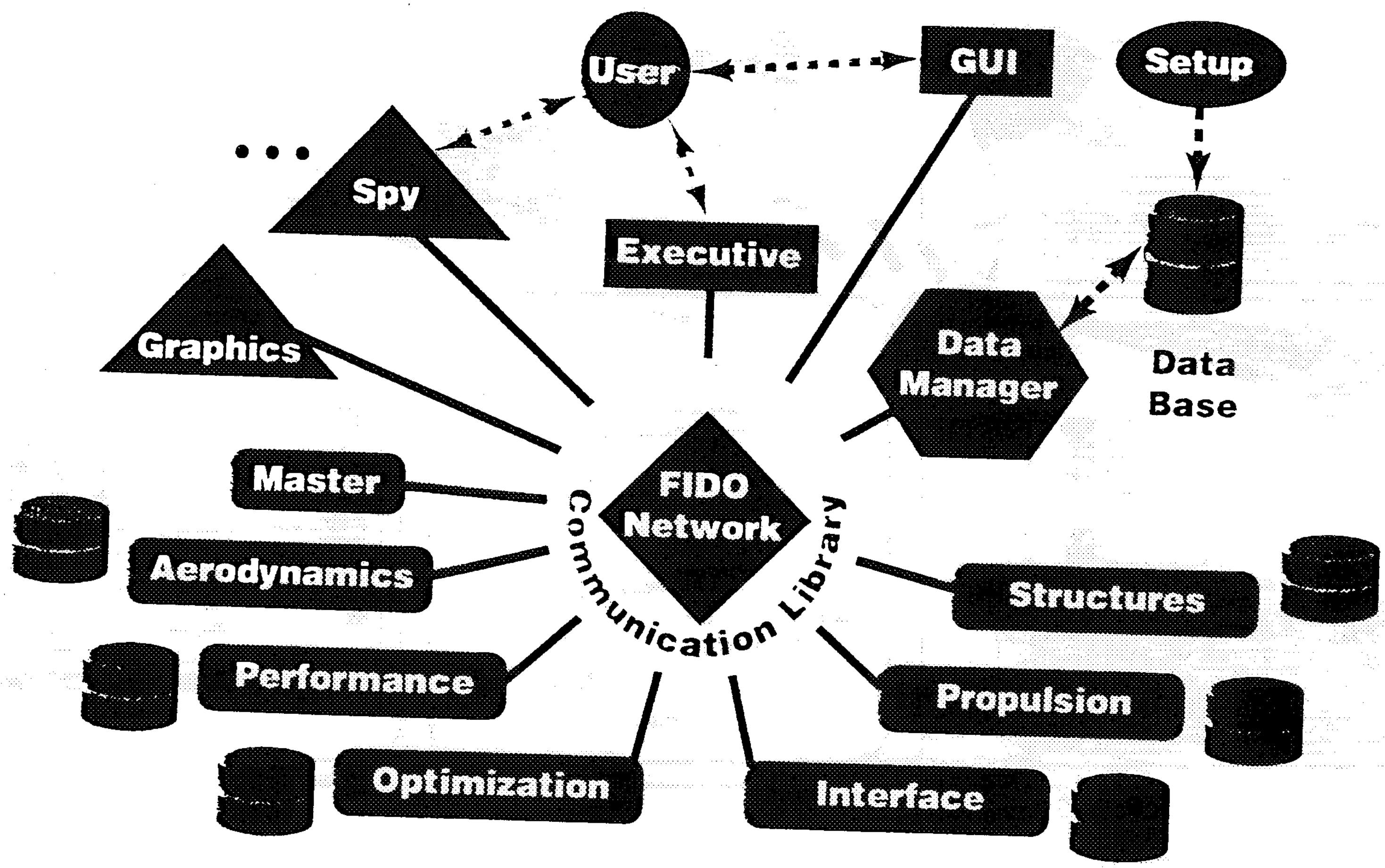



HSCT problem diagram
Flight conditions
Base geometry
Design variables
Aerodynamic analysis
Linear
$C_p$
$C_d, C_l$
$C_d$
Trim
Propulsion analysis
Engine deck
fuel flow
Performance analysis
Range equation
fuel weight
total wgt.
Structures analysis
Equivalent plate
stru. wgt.
Stresses deflections
New design variables
Optimizer
Minimize weight
System responses and derivatives

# FIDO User Interface Concept



# FIDO Execution System



# Features

- Provides synchronous control in heterogeneous computing environment
- Is adaptable to a wide variety of problems that use multiple computer programs
- Has modular form for easy problem migration
- Incorporates variety of debugging features
- Includes "SPY" capability for -
  - Monitoring computational progress
  - Displaying variables graphically
  - Modifying variables for design steering

# FIDO Project Products

Prototype multi-discipline, multi-computer codes

- adaptable
- modular
- portable
- non-proprietary

Prototype toolkit:

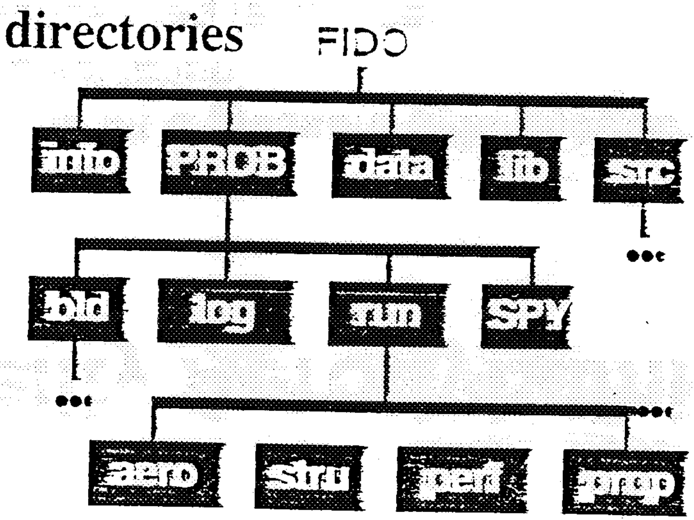
- user interface
- communication library
- data manager library
- driver templates

Software recommendations:

- system software needs
- application programming guidelines

# Programming Environment Resources

- Specified code interfaces
- Programming guidelines
  - be adaptable, modular, portable, non-proprietary
- Communications library
  - use message-passing standards
- Database manager
  - use file-based data schema
- Programming templates
- Structured directories



# Low Fidelity Aerodynamics

Based on linear theory

WINGDES : drag due to lift & pressure difference across the wing
AWAVE: wave drag calculation
WAREA/CDFR1: viscous drag (empirical)

Utilizes very simple geometry description.

**INPUT**
- Geometry Description
- Flight Conditions
- Discretization Parameters

**OUTPUT**
- Lift Distribution
- Lift and Drag

# Low Fidelity Structures

Equivalent Laminated Plate Solution (ELAPS)

ELAPS uses a series of plates defined on wing planform.

Camber and Thickness are defined for each plate.

<u>INPUT</u>
- Forces (acquired from Aero)
- Wing Geometry
- Skin Thickness

<u>OUTPUT</u>
- Structural Weight
- Stresses
- Displacements

# Low Fidelity Propulsion

- Level 1 propulsion module from Lewis (TBE_NASA)

INPUT
- Engine Type
- Altitude and Mach No.
- Thrust

OUTPUT
- Fuel Consumption

# Low Fidelity Mission Performance

- Breguet Range Equation

INPUT
- Fixed Range
- Lift and Drag
- Estimated Total Weight

OUTPUT
- Fuel Weight

# Approaches to Sensitivity Analysis in CFD

- **Finite Differences**

  Compute the sensitivity by taking the difference between two analyses with slightly different values of the design variable.

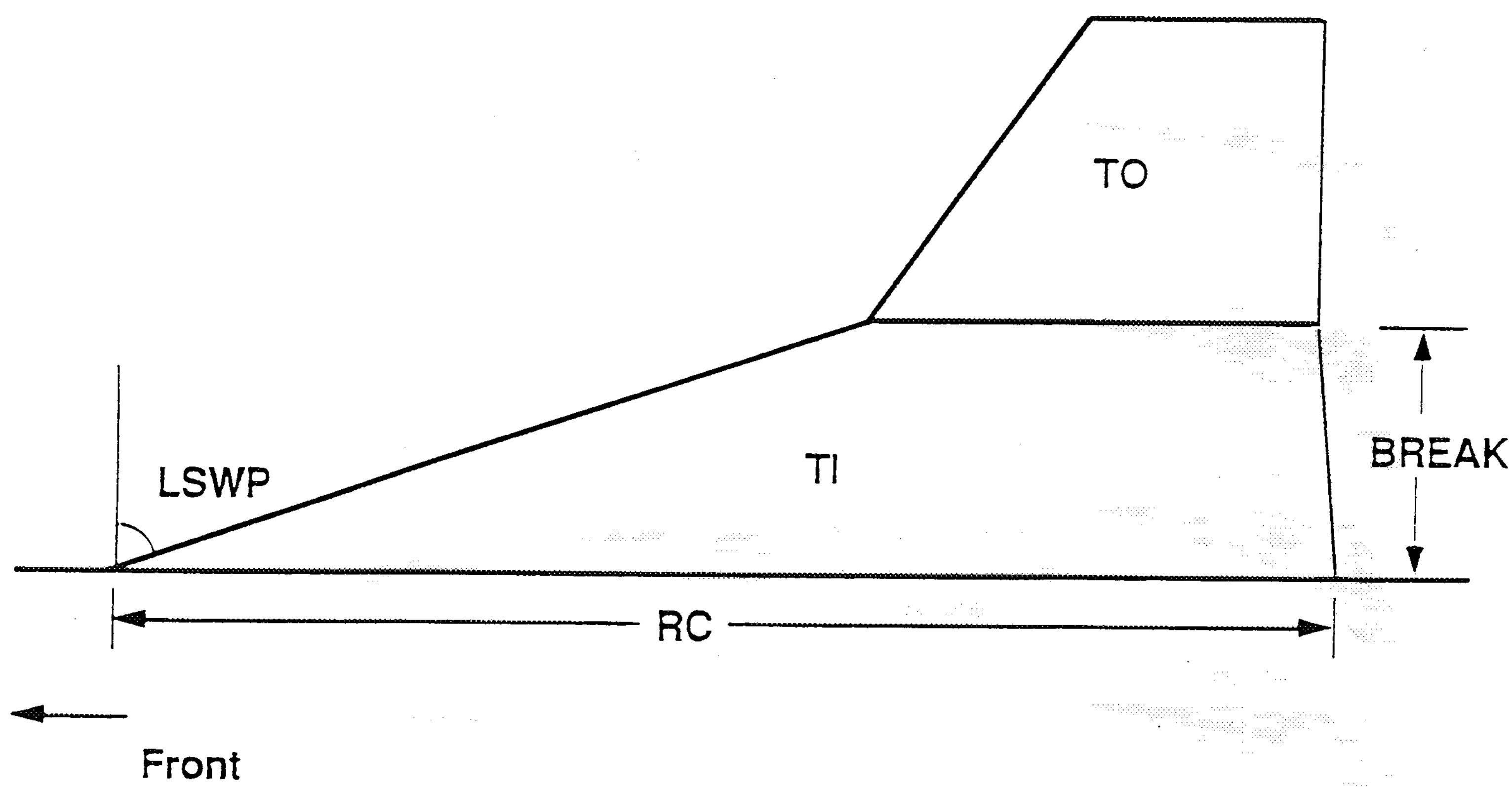
- **Manual Insertion of Quasi-Analytical Sensitivity Derivatives**

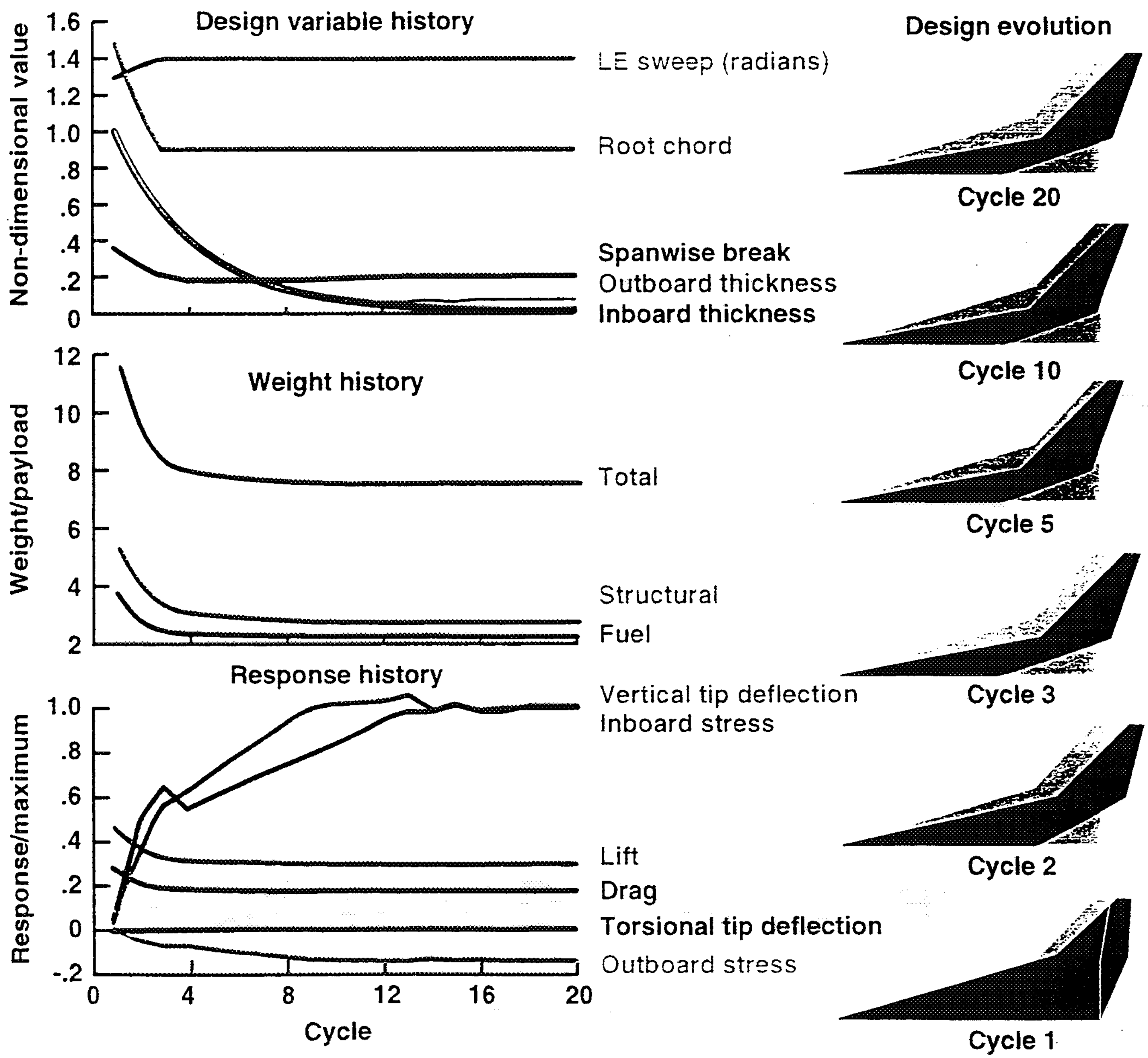
  Derive the exact, linear equations satisfies by the sensitivity derivatives and add them to the analysis code.

- **Automatic Insertion of Quasi-Analytical Sensitivity Derivatives**

  Utilize a pre-procesor which automatically analyzes the Fortran code and adds (automatically) the code for the prescribed sensitivity derivatives.

# Wing Planform Design Variables





Design variable history
Non-dimensional value
1.6
1.4
1.2
1.0
.8
.6
.4
.2
0
LE sweep (radians)
Root chord
Spanwise break
Outboard thickness
Inboard thickness
Weight history
Weight/payload
12
10
8
6
4
2
Total
Structural
Fuel
Response history
Response/maximum
1.0
.8
.6
.4
.2
0
-.2
0
4
8
12
16
20
Cycle
Vertical tip deflection
Inboard stress
Lift
Drag
Torsional tip deflection
Outboard stress
Design evolution
Cycle 20
Cycle 10
Cycle 5
Cycle 3
Cycle 2
Cycle 1

# Accomplishments

The FIDO system has demonstrated -

- Optimization of simple HSCT design problem
- Use on network of UNIX® workstations (Sun, SGI, DEC)
- Monitoring and graphics display through "SPY"
- Flexibility of file-based data management

# Plans

- Demonstrate FIDO on a complex HSCT design problem
- Replace low-fidelity with high-fidelity analysis computer programs
- Apply graphical user interfaces for problem setup and for control
- Fully document the software and publish findings
- Get and apply feedback from beta tests

# FIDO Project Credits

Programming group:

- Bob Weston ... project leader, data manager
- Tom Eidson ... project designer, communications
- Jim Townsend ... aerodynamics, propulsion
- Ray Gates ... user interface
- Ramki Krishnan ... aerodynamics
- Ben James ... optimizer
- Kelvin Edwards ... graphics, system administrator
- Bill LaMarsh ... structures, performance

Consultants:

- Peter Coen ... optimizer, performance
- Tom Zang ... aerodynamics
- Gary Giles ... structures
- Gregg Wrenn ... structures
- Tom Crockett ... communications
- Don Randall ... user interface
- Larry Green ... aerodynamics, propulsion
- Mary Adams ... graphics

# An Overview of the Computer Aided Engineering and Design for Electronics Laboratory

by Shelley Stover

sks@longstreet.larc.nasa.gov

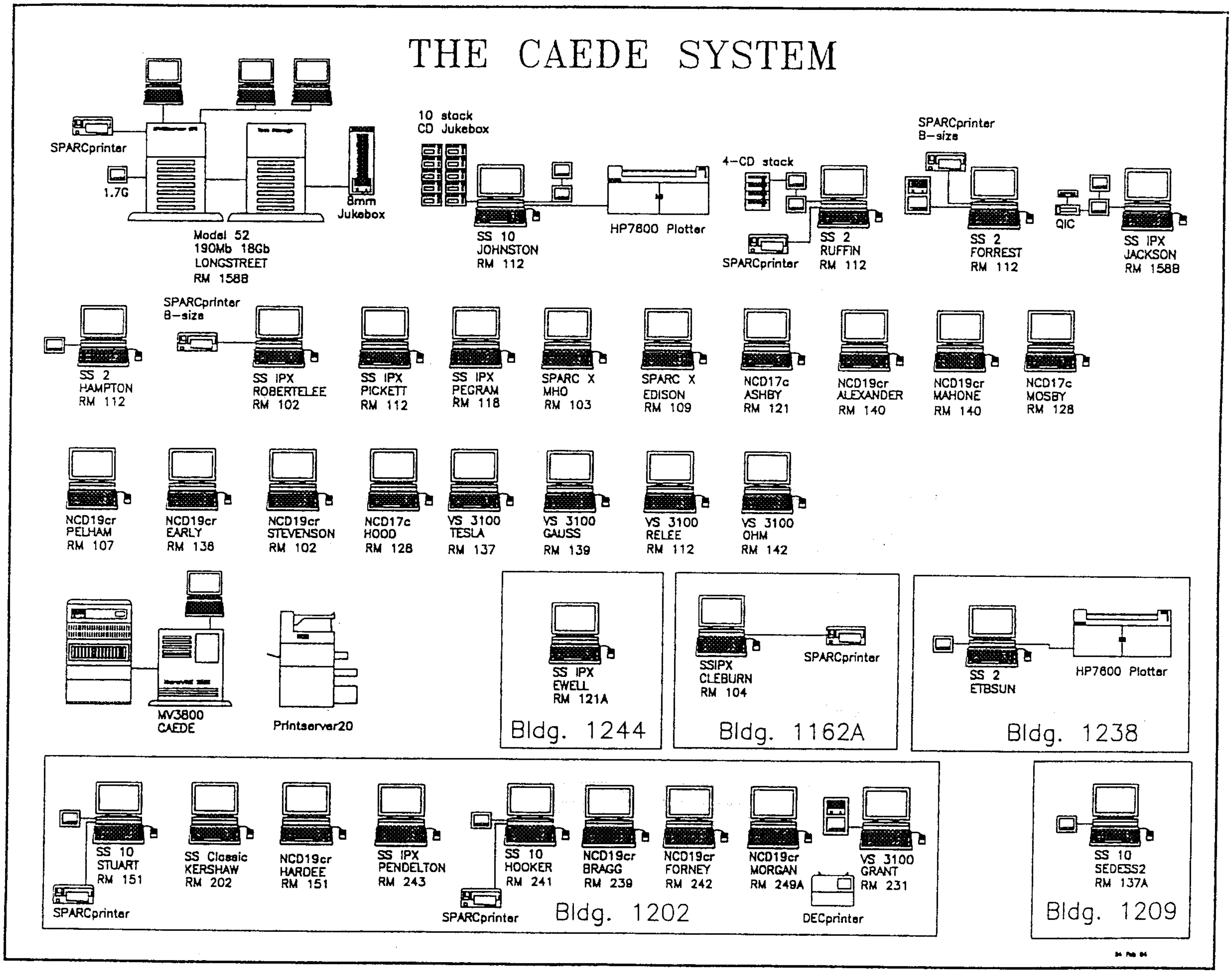
## Introduction

CAEDE is a center-wide resource and any Langley employee can obtain a user account.

CAEDE hosts a wide variety of software tools which are applicable to many engineering disciplines such as

- electronic circuit design of programmable integrated circuits and printed circuit boards
- control system modeling
- optical engineering

# THE CAEDE SYSTEM



# CAEDE Software

## Core Tools

### Cadence Design Tools

Concept - Design entry
Analog Workbench - Analog design and analysis
Logic Workbench - Digital design and analysis
Leapfrog - VHDL simulator
PIC Designer - PLD design
Profile - Analog HDL simulator
Spice Plus - Analog simulator
Mixed Signal Simulation Environment - mixed board design
OpenSim - Backplane multi-engine simulator
Parametric Plotter - Trade-off and what-if analysis
Smoke Alarm - Stress, Power, dissipation analysis
Veritime - Timing analysis
Allegro - Layout tool

### Matlab (Signal Analysis and Modeling)

Matlab - High performance numeric computation
Simulink - Simulates dynamic systems
Signal Toolbox - Signal processing tool
Control Toolbox - Control system engineering
Robust Toolbox - Enhanced control toolbox
Neural Net Toolbox - Neural network tool

### Databases

Oracle - Relational database

### Documentation

WordPerfect - Word processor

### Agency - Wide:

IHS ICSC - IC/Discrete Parameter database
IHS Recal/Z - Passive component database
NAS - NASA Assurance System

## Hosted Tools

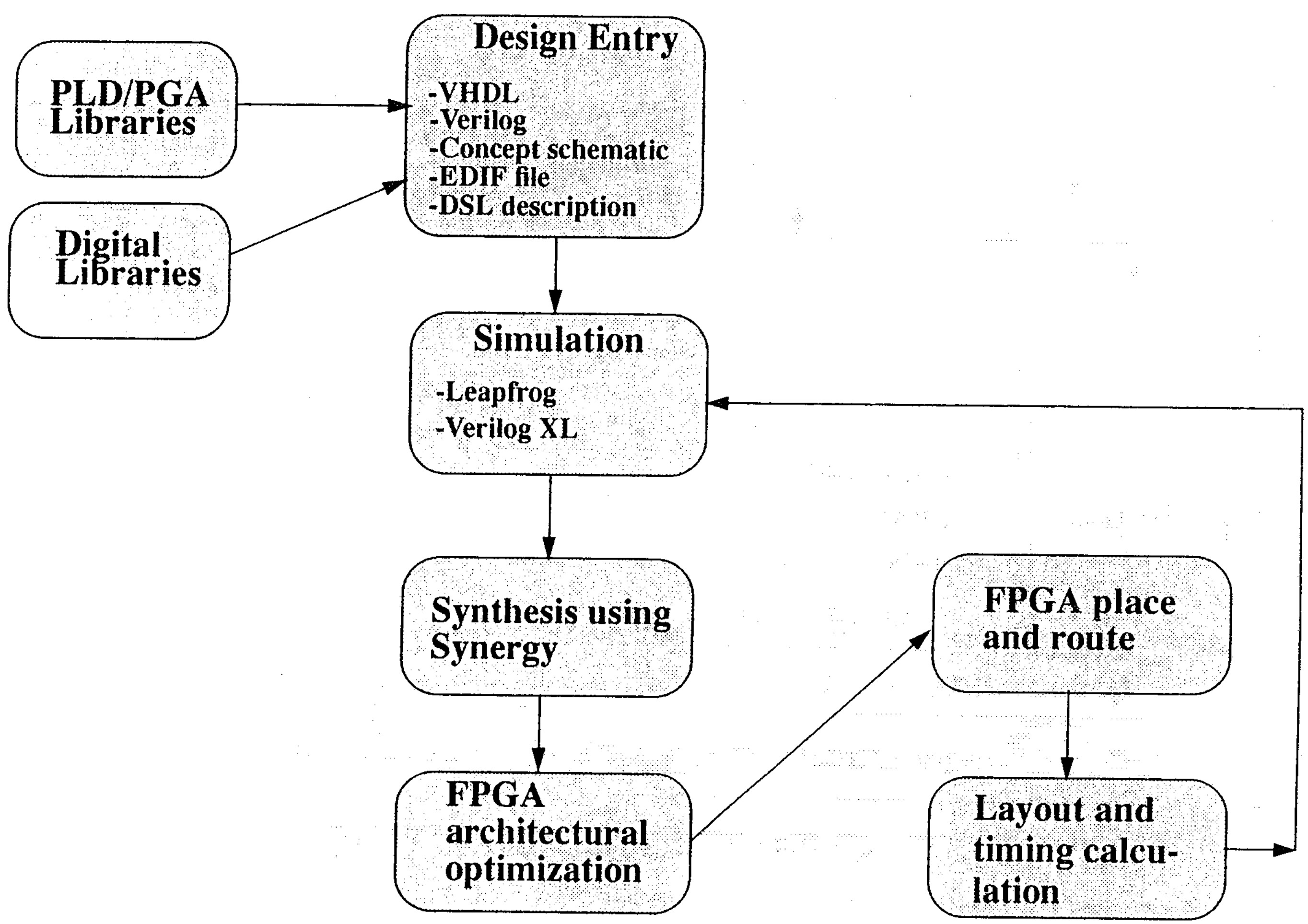
Verilog XL - Verilog simulator
Vanguard - Design entry
Composer - Design entry
Silvaco - 2D device simulator framework
PSpice - Digital and analog circuit simulator
Labview - Virtual instrument workbench
CodeV - Optical design and analysis
Autocad - CAD package
CorelDraw - drawing package
Framemaker - Word processor
TMS320C compiler and simulator
Softwindows- Microsoft windows emulator
C & C++
ADAS - Architectural design and assessment
Foundry libraries

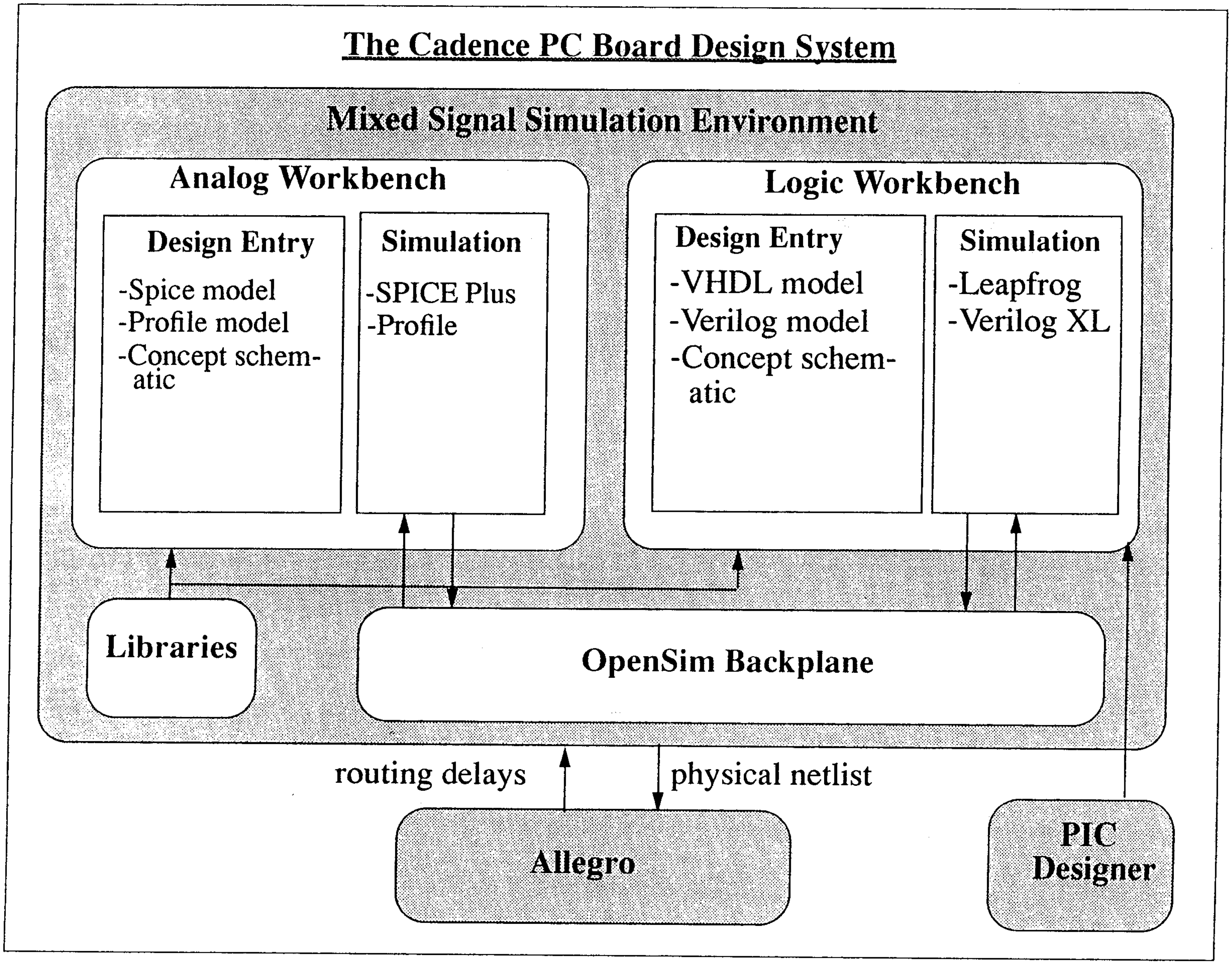
# The Cadence Design Tools

CAEDE hosts the following Cadence tools for the design of integrated circuits and printed circuit boards.

- Concept - schematic design entry
- Composer - schematic design entry
- Verilog XL - Verilog compiler and simulator
- Leapfrog - VHDL compiler and simulator
- Profile - analog design language compiler and simulator
- PIC Designer - programmable integrated circuit design environment
- Logic Workbench Designer - digital design environment
- Analog Workbench - analog design environment
- Mixed Signal Simulation Environment
- Allegro - layout of printed circuit boards

## The Cadence Integrated Circuit Design System: PIC Designer




The Cadence PC Board Design System
Mixed Signal Simulation Environment
Analog Workbench
Design Entry
-Spice model
-Profile model
-Concept schem-atic
Simulation
-SPICE Plus
-Profile
Logic Workbench
Design Entry
-VHDL model
-Verilog model
-Concept schem-atic
Simulation
-Leapfrog
-Verilog XL
Libraries
OpenSim Backplane
routing delays
physical netlist
Allegro
PIC Designer

# Electronics Parts Database

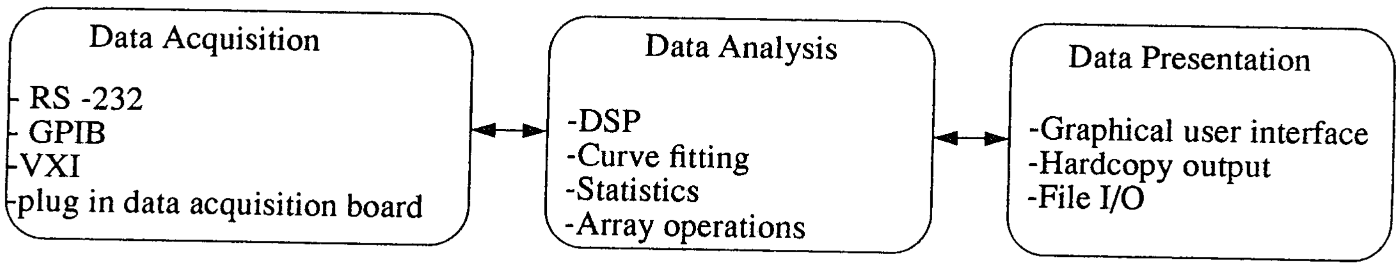
- CAEDE hosts an on-line database of electronic part databooks and specifications which is updated bi-monthly.

- The database contains over 2 million integrated circuits, discrete semiconductors, and passive components. The parts can be searched by part number, characteristics, and function.

- The software is from Information Handling Systems (IHS) and is contained on 60 CDs in 10 jukeboxes. Our license is agency wide and funded by Code QE.

# Labview

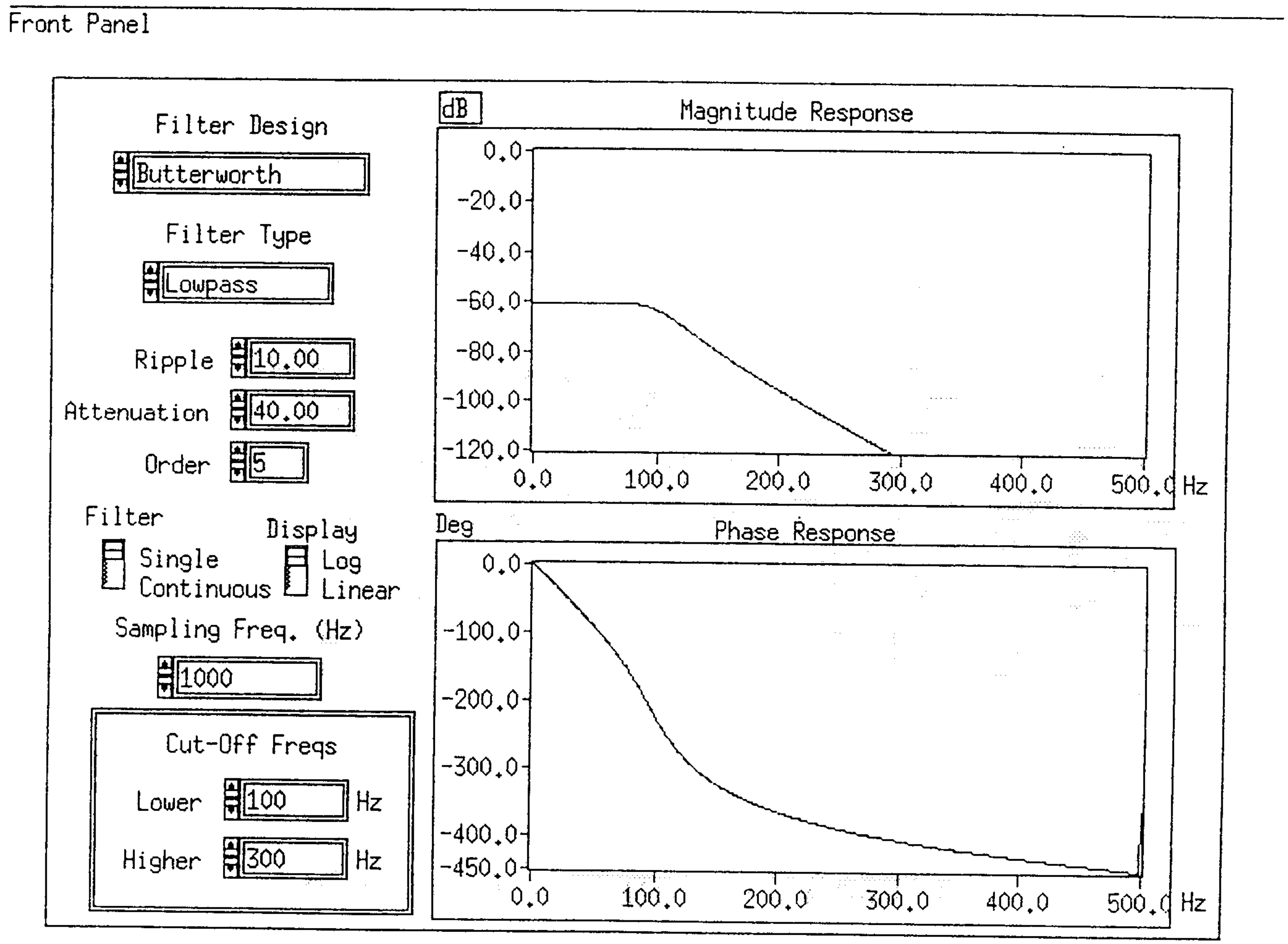
## Data flow in Labview




## Labview Virtual Instrument Components:

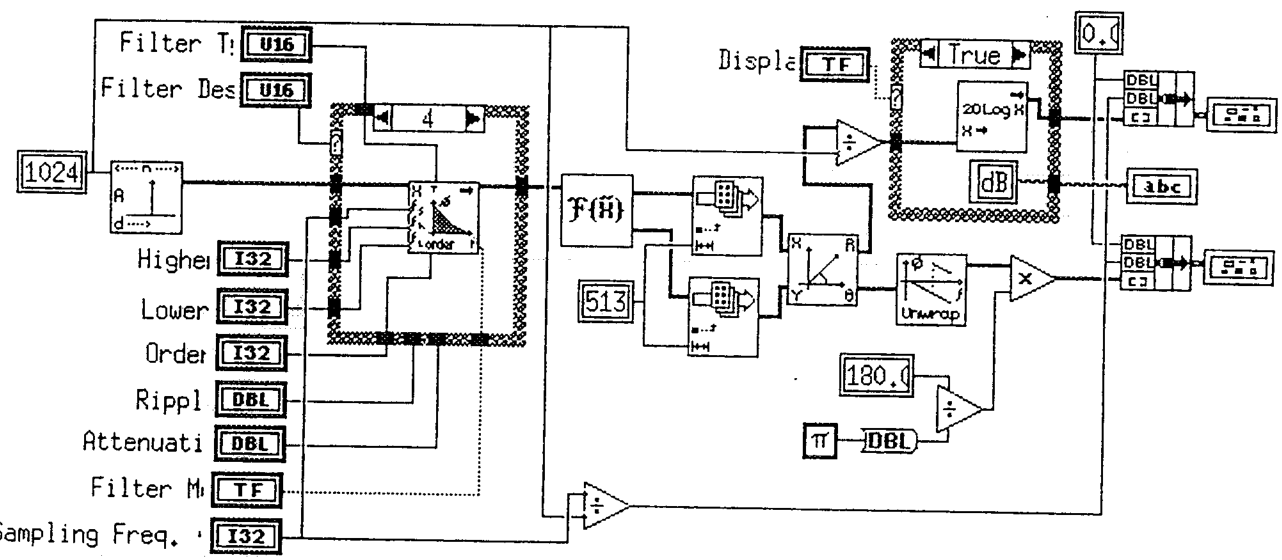
- Front panel - interactive interface to supply inputs and observe outputs of the instrument
- Top-level block diagram - used to program the VI for acquisition, analysis, and formatted I/O
- Low-level panels and diagrams - together these components form sub VIs which can be used as instruments in other VIs

# Front Panel of IIR Filter Example



# Block Diagram of IIR Filter Example

Block Diagram



# MATLAB

Matlab

- Matlab is a high performance package for engineering and scientific computations. It allows users to manipulate matrices in a fraction of the time that it would take to write C programs, etc. Also, Matlab is able to generate several different 2D and 3D graphs.

- Matlab toolboxes:

  Control System Toolbox

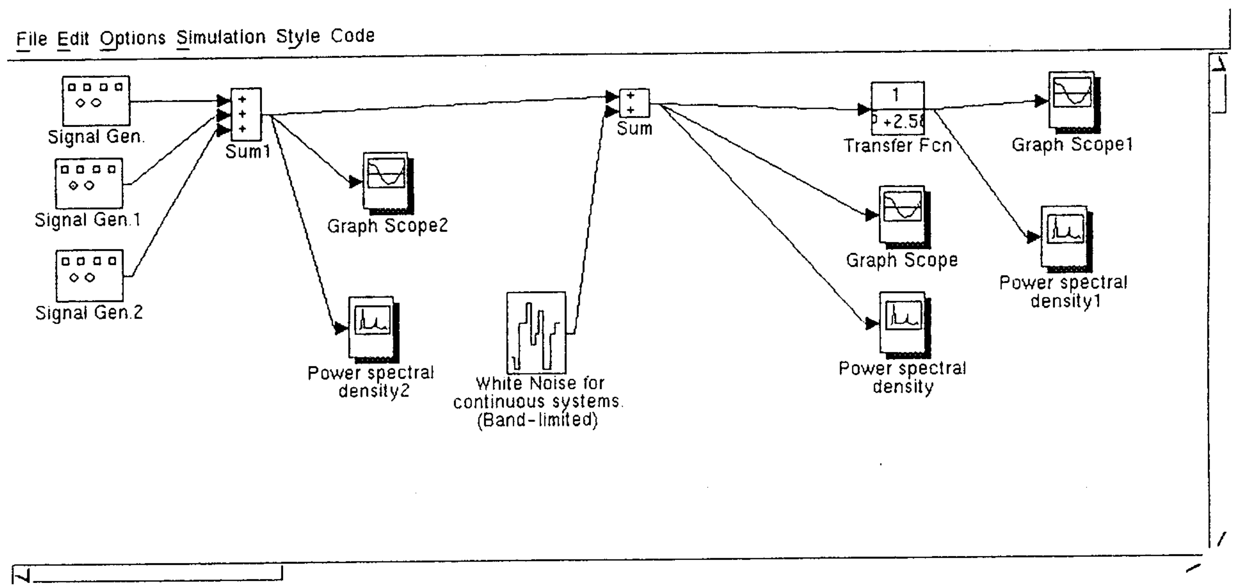
  Signal Processing Toolbox

  Robust Control Toolbox

  Neural Network Toolbox

- SIMULINK - Provides a graphical user interface for modeling and simulation of dynamic systems.

# Simulink Block Diagram



# Optical Research Associates - Code V

- Code V is used for the design, optimization, tolerancing, costing, manufacturing, and alignment of optical systems.
- One of its primary uses is the development of optical systems from concept through final design.
- It handles many types of surfaces including aspherics, toroids, gratings, diffractive optical elements, and many others.
- Code V can also be used for cost estimates, environmental analysis, image simulation, and it interfaces with NASTRAN, interferometers, and CAD/CAM packages via IGES.
- Code V was used for the design and analysis of the LASE and LITE optical systems.

## Silvaco

- Composed of the S-Pisces, Blaze, and Luminous simulators and the graphics package Tonyplot
- S-Pisces is the two-dimensional device modeling program that simulates the electrical characteristics of common MOS and bipolar integrated circuit elements
- Blaze contains material libraries for exotic semiconductors such as SiC and GaN as well as the codes for simulation of heterostructure devices.
- Luminous implements models for optical generation of carriers and allows simulation of a broad range of devices .

# Training

- The following Cadence training videos are available:
  - Concept
  - PIC Designer with Concept
  - Top-Down Design with Logic Workbench
  - Designing with Leapfrog
  - Introduction to VHDL
  - Veritime
  - Allegro Automatic
  - Allegro Interactive
  - Analog Workbench II with Concept
  - Analog Workbench Mixed-Signal

  NOTE: These videos are accompanied by training manuals, lab manuals, and lab software and are updated as new versions are installed.
- On-line tutorials are available for Matlab, Labview and Simulink
- Basic UNIX short courses are provided by CAEDE system administrators

# General Information

- CAEDE accounts are available to all center employees. To request more information or obtain a user account form contact one of CAEDE's system administrators:

  Steve Comer: steve@longstreet.larc.nasa.gov, extension 43710

  Don Brandt: dtb@jackson.karc.nasa.gov, extension 43716

- For general information on CAEDE and the Information Handling Services see the MOSAIC CAEDE Home Page.

- For IHS technical information contact

  Randy Regan: crr@longstreet.larc.nasa.gov, extension 41869

# The Software Engineering and/or Ada Lab (SEAL)

Presented at the

"Role of Computers in LaRC R&D" Workshop

June 16, 1994

Robert Kudlinski

Information Systems Division

Software Engineering and/or Ada Lab (SEAL)

## BACKGROUND (and HOME PAGE)

- 1989: ACD (now ISD) was charged to manage, develop, and assure mission critical software systems for several on-going and all new *LaRC space-flight projects*
- 1990: The SEAL was a new-start to meet this requirement by implementing a common software engineering process (*people*, *procedures* , *tools*) across these projects
- 1992: Selected to participate in the *NASA Software Engineering Program*
- Since inception, the SEAL has received increasing requests from other LaRC software organizations/domains seeking to improve their processes. *Existing* and *future plans* to help transfer software engineering technologies are presented.

Software Engineering and/or Ada Lab (SEAL)

## LaRC SPACE-FLIGHT PROJECTS

- The SEAL has supported numerous projects at various lifecycle phases: CERES, CSI, DGV, JADE, LITE, MIDAS, RAME, ROVER, SABRE, SAFIRE, SAGE III, SEDS, SUNLITE, and TRACER
  - Primarily remote sensing, active control of space structures and technology demonstration experiments
  - On-board, embedded flight computers (primarily 80x86) for real-time instrument control and data acquisition (3,000 - 25,000 SLOC)
  - GSE computers (primarily 80x86 PC's) for instrument development, test, calibration and mission operations (5,000 - 70,000 SLOC each)
  - Missions from 3 days to 5 years, development schedules of 2-5 years
- Many technical and management challenges:
  - Real time, embedded, and non-deterministic systems require specialized tools and practices
  - Physical constraints (power, weight, radiation) severely limit computer resources (CPU speed, memory, I/O) and demand optimization
  - Insufficiently defined, continually expanding requirements
  - Trading off quality and reliability with tight manpower and schedules
  - Short term vision of projects does not help process improvement

Software Engineering and/or Ada Lab (SEAL)

## HUMAN RESOURCES

- Education and training
  - Offering 10-15 classes per year through Training Office such as Ada, objected oriented design, CM, real-time programming, rate monotonic analysis and system engineering
  - Developing, documenting, and video taping specialized in-house training such as Ada programming, formal inspections, and using real-time, embedded systems tools
- Information Resources
  - Library (books, periodicals, standards, guidebooks)
  - Electronic information exchange and communications network
  - Providing ISD user consultation service for Ada and software engineering questions
- Projects were assigned and the SEAL became a new start one year before the current climate of decreasing civil service and NPS, which has curtailed planned activities

Software Engineering and/or Ada Lab (SEAL)

## PROCEDURE GOALS

- Standardize on and reuse common procedures, expertise, tools, and products across projects
  - » NASA and other software standards as available and appropriate
  - » Object-oriented requirements analysis and design methods
  - » Ada used as the primary programming language, some C, C++
- Define and document key, repeatable software development procedures as baseline for future process improvement and training new employees
  - » Formal inspections guidebook completed
  - » Configuration Management Guidebook in review process
  - » Guidelines for selected real-time, embedded system tools completed
  - » Evaluating existing, documented IV&V procedures
  - » Other procedures under development
- Moving toward a complete software lifecycle approach based on evolutionary spiral model
- Contact Pat Schuler at 4-6732 for more information on SEAL procedures

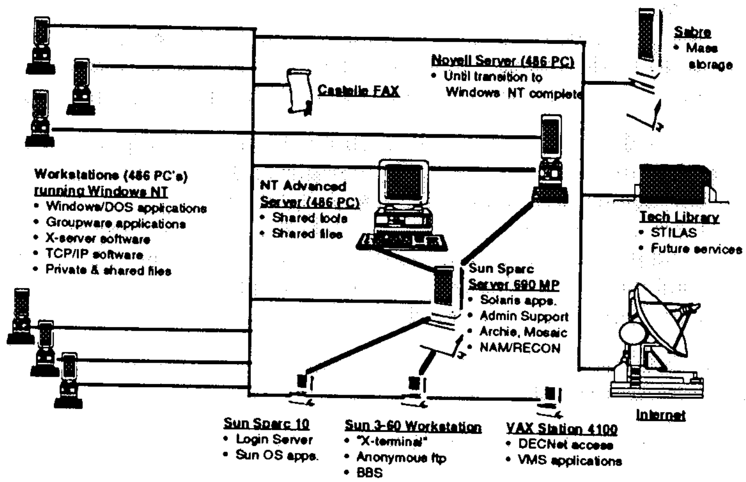
Software Engineering and/or Ada Lab (SEAL)

## TOOLS

- Operating a distributed software development environment via LaRCNET
  - » Compilers, CASE, CM, and project management tools in place
  - » Beginning to use InQuisix Reuse tool
  - » Reverse engineering and code analysis tools
  - » Electronic information management and communication tools
  - » Plan to increase use of automated code generators and testing tools
  - » Work from desk, SEAL, hardware development labs and test facilities
  - » Standardized environment allows sharing of tools to reduce project cost and effectively shift personnel in response to changing priorities
- Real-time, embedded analysis tools
  - » Embedded system cross compilers
  - » Emulators and logic analyzers for 80x86 and 1750A processors
  - » Functional equivalent 80x86 and 1750A flight computers
- Contact Jerry Garcia at 4-5888 for more information on SEAL tools

## Software Engineering and/or Ada Lab (SEAL)

# S/W DEVELOPMENT ENVIRONMENT




## Software Engineering and/or Ada Lab (SEAL)

# NASA S/W ENGINEERING PROGRAM

- One Code QE mission has been to improve the quality and reliability of software products developed for space flight projects, both manned and unmanned, at the NASA Centers
- The NASA Software Engineering Program was initiated in 1991 to establish and grow Centers of Excellence across the Agency:
  - » JSC: Shuttle Data Systems Branch
  - » GSFC: Flight Dynamics Division (Software Engineering Lab)
  - » LaRC: Flight Software and Graphics Branch (SEAL)
- Long term vision of the Program is to put self-sufficient, continually improving processes in place, establish standards, and then use these organizations to transfer effective technologies to other NASA domains
- LaRC tasks have been primarily in the areas of:
  - » Capturing and documenting the SEAL software development process for small Ada projects and assessing Ada's impact
  - » Improving software technology transfer methods and software reuse
  - » Evaluating IV&V procedures on LaRC projects and research testbeds

## Software Engineering and/or Ada Lab (SEAL)

# EXISTING GENERAL SUPPORT

- Coordinate 10-15 widely attended software engineering courses per year and sponsor educational presentations (e.g., LaRC/NASA management, LCUC, Local Universities, Conferences)
- Implement and fund formal inspections program widely used across LaRC (training, pilot projects, implementation guidebook)
- Serve on numerous review panels, technical committees and QAT's
- Transferring software technologies to other domains, such as NTF DAS, flight simulation, HSR, and TAP programs - currently manpower limited
- Open access to SEAL tools via LaRCNET - currently manpower limited for training and consulting on these tools
- Access to real-time, embedded system tools possible, but requires proper training and procedures to be followed
- Open access to library (books, periodicals, standards, guidebooks)
- Started the Software Quality, Productivity and Reliability Team to promote communication and coordination of software engineering efforts at LaRC

## Software Engineering and/or Ada Lab (SEAL)

# FUTURE GENERAL SUPPORT

- Continue existing work on previous chart, expanding where possible:
  - » Proposal accepted by Code Q that 90% of SEAL funding (FY 95-98) specially targeted for manpower to transfer software engineering technologies to on-going LaRC research programs
  - » Decreasing project load should provide ISD opportunity to shift resources from space-flight projects to general support
  - » Partnerships are welcomed
- Increase use of electronic information dissemination, particularly MOSAIC

SESSION 9 CAE Tools

Chaired by

Carol D. Wieseman

**Digital Control of Wind-Tunnel Models Using LabVIEW**

by

Sherwood T. Hoadley

presented at the
LaRC Computer Systems Technical Committee Workshop on
The Role of Computers in LaRC R&D
June 15-16, 1994

Digital controller and data acquisition and analysis systems were developed for several wind-tunnel models which use National Instruments LabVIEW® Software and National Instruments Hardware within a Macintosh environment. The objective of this presentation is to illustrate the use of LabVIEW for interactive animated display of acquired experimental data and real-time control of some wind-tunnel models.

The first system illustrates a flutter suppression system (FSS) which was used to suppress flutter for a small piezoelectrically actuated wing in a small flutter research and experiment device (FRED) with a 6"×6" test section. The following illustrations are included which show various aspects of the FSS system:

- A photo of FRED and a flow diagram of the wind tunnel
- A block diagram of the closed-loop system
- The digital control system software schematic of the LabVIEW user interface routines on the Macintosh and the real time system comprised of boards plugged into the Macintosh Nu-bus but sharing their own real-time RTSI bus.
- The front panel of the FSS LabVIEW Controller virtual instrument (VI) interface to the real-time controller digital signal processor (DSP)
- Results of open and closed loop strain response to wind-tunnel turbulence

The next LabVIEW VI which is illustrated is an instrument which interfaces with a data logger which samples various thermocouples and sends the requested data to the Macintosh for display and storage. Two figures included are:

- The front panel depicting a beam clamped to a table with thermocouples placed at various locations and a strip chart displaying the data.
- The block diagram of the code for the data logger which shows the way in which LabVIEW is coded. As indicated, the code is a flow diagram of itself.

The last system illustrated is a system which provides passive control of three different aerodynamic control surfaces for a Benchmark Active Controls Testing (BACT) model in the

Transonic Dynamic Tunnel (TDT). This Passive Digital Controller System (PDCS), developed for the BACT, was used in the tunnel in November 1993 and will be used again in November 1994. It interfaces with a real-time DMA controller to command control surface positions and excitations, but does not actively employ sensor signals from the wing from which to compute control surface commands in order to suppress aeroelastic phenomena such as flutter. It does provide the following functions:

- Static command of control surfaces positions
- Excitation of control surfaces, singly or in combination
- Monitoring of control surface positions and error signals, actuator hydraulic pressures, and hinge moments
- 'Trip' System control for wind-tunnel safety

# DIGITAL CONTROL OF WIND-TUNNEL MODELS USING LabVIEW

by

Sherwood T. Hoadley

LaRC Computer Systems Technical Committee Workshop
on

The Role of Computers in LaRC R&D
June 15-16, 1994

# OBJECTIVE

DEMONSTRATE THE USE OF LABVIEW®

FOR

INTERACTIVE ANIMATED

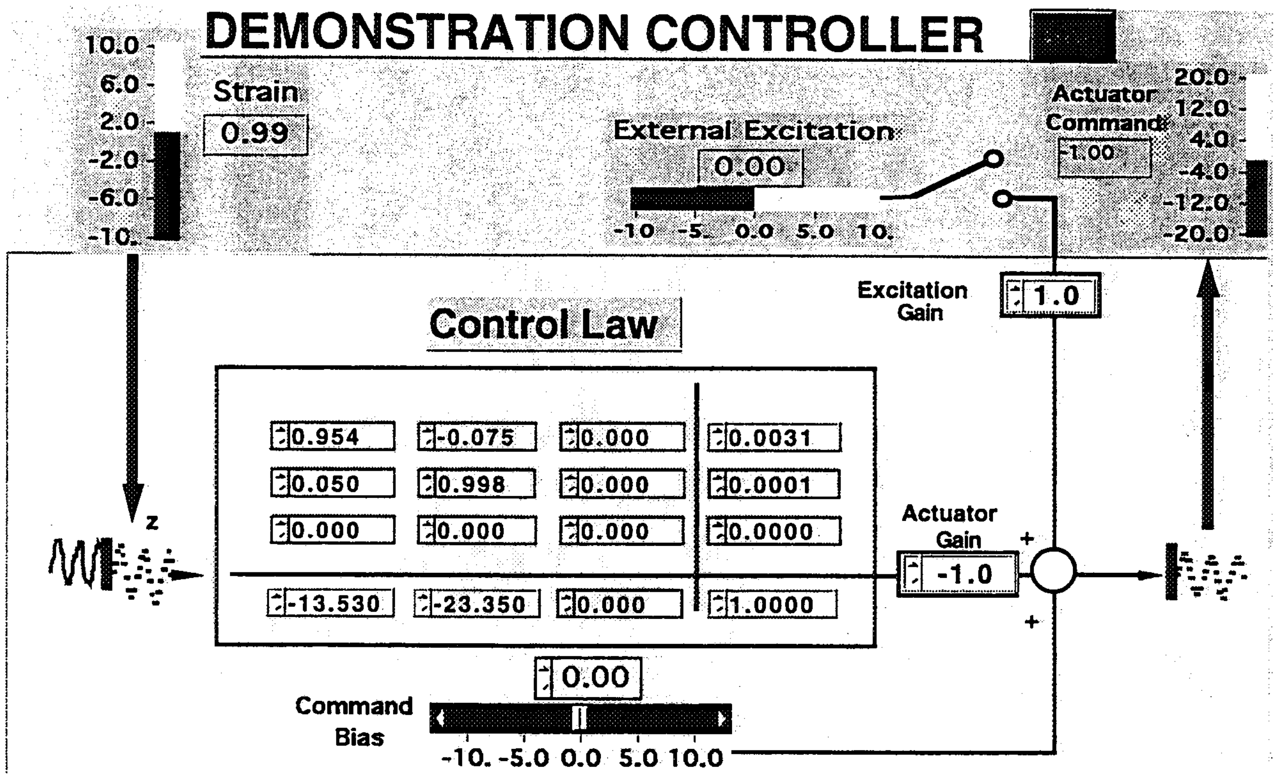
DISPLAY OF ACQUIRED EXPERIMENTAL DATA

AND

REAL-TIME CONTROL OF WIND-TUNNEL MODELS

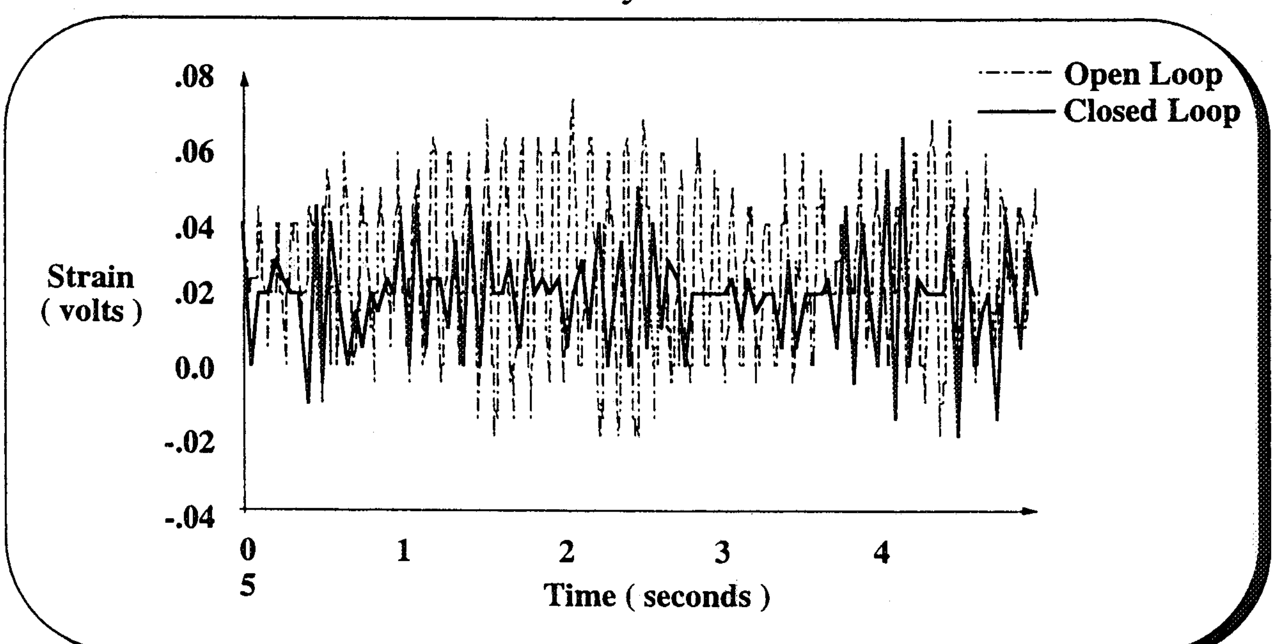
## LabVIEW FLUTTER SUPPRESSION CONTROLLER

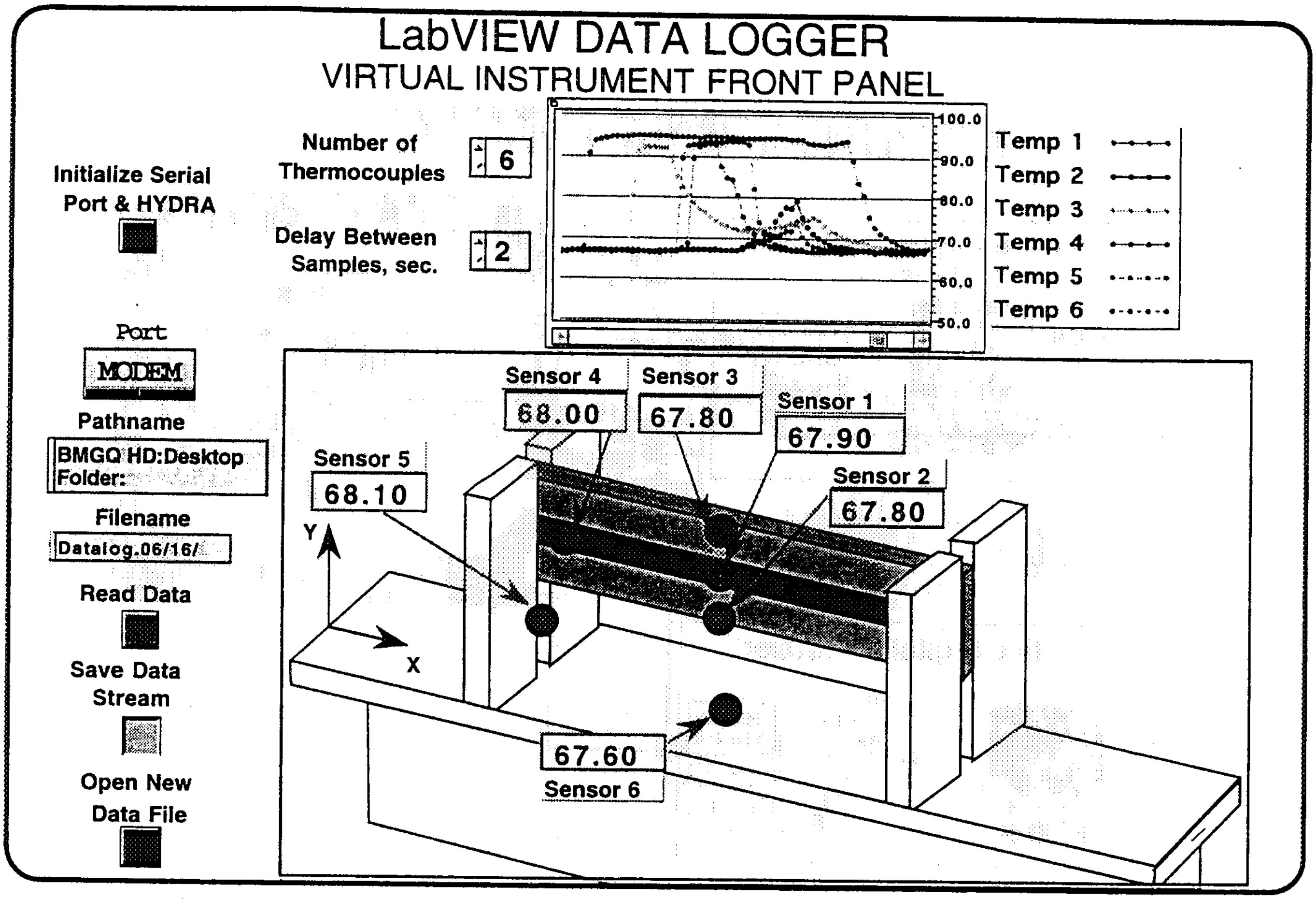
VIRTUAL INSTRUMENT FRONT PANEL




## OPEN AND CLOSED LOOP STRAIN RESPONSE TO WIND TUNNEL TURBULENCE

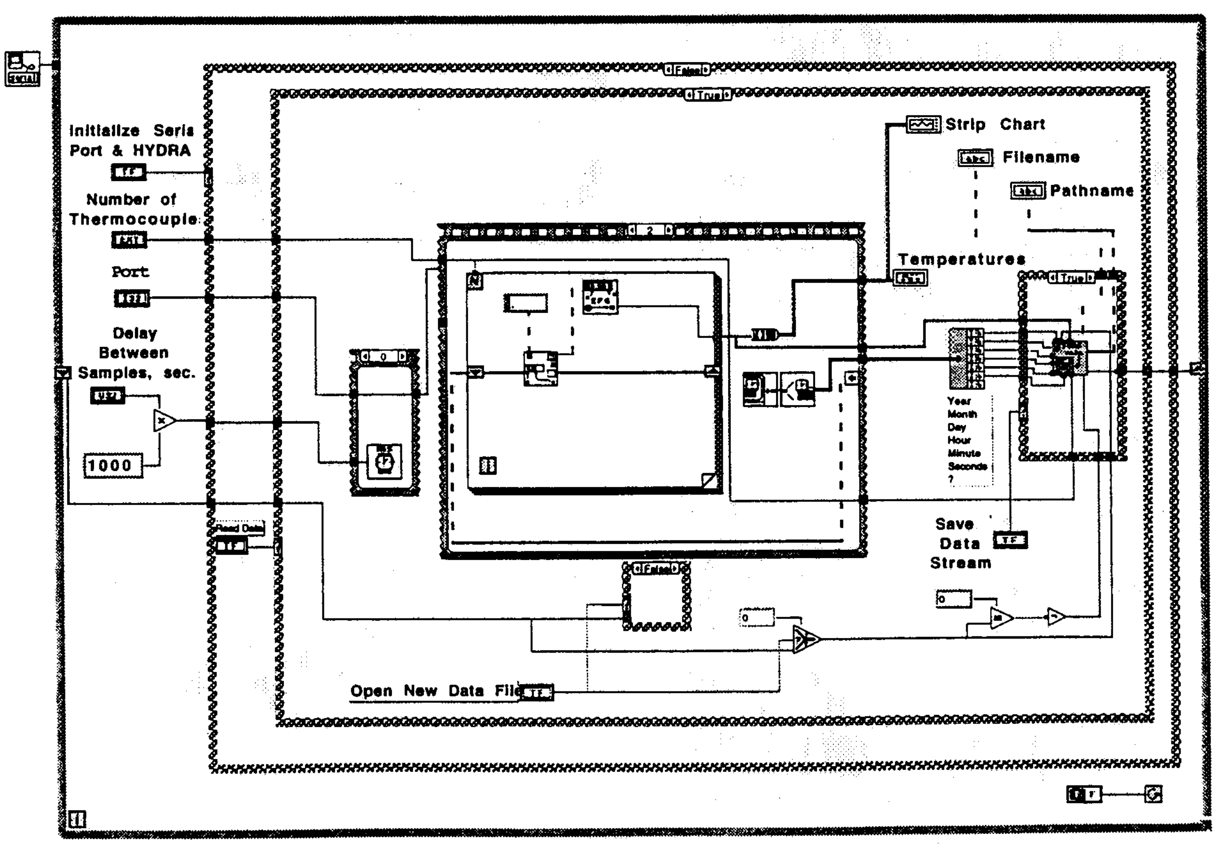
*Wind Tunnel Velocity 575 inches / second*



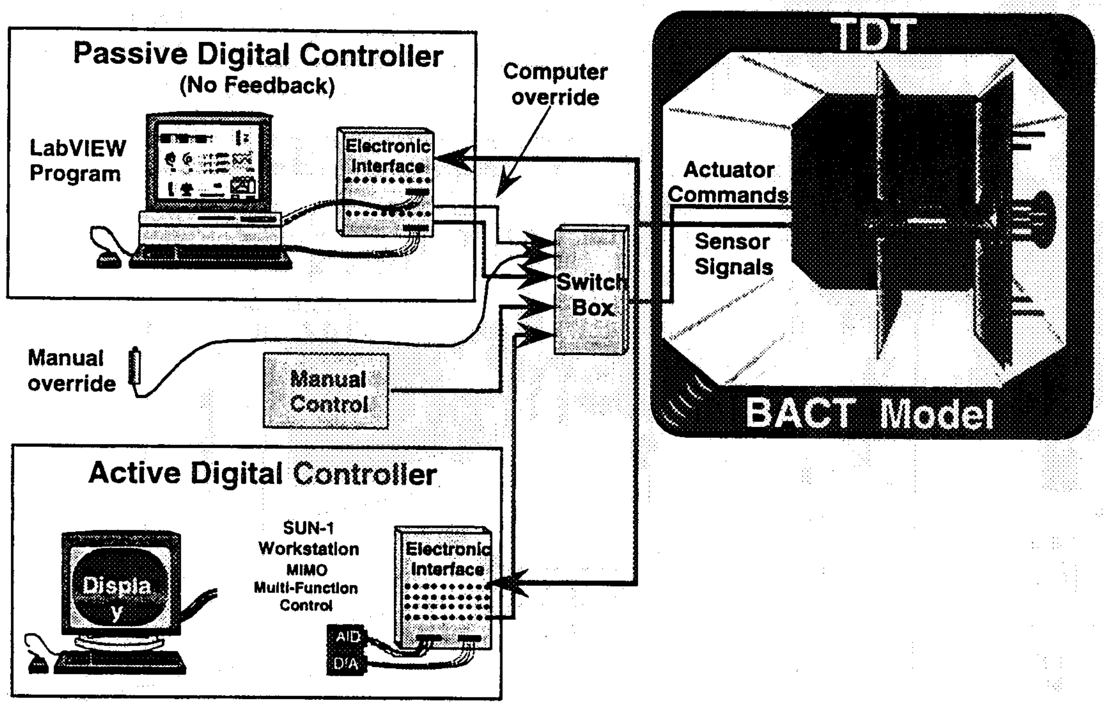



Hydra Data Logger DAQ.950
Wednesday, June 15, 1994 5:05 PM

Block Diagram



# BACT Digital Controller System




## LabVIEW PASSIVE DIGITAL CONTROLLER & DATA ACQUISITION
### VIRTUAL INSTRUMENT FRONT PANEL