Rockwell Avionics

Collins

Panel Discussion:
Industry Perspectives on Formal Methods

Dr. Steven P. Miller

Advanced Technology & Engineering
Collins Commercial Avionics, MS 124−211
400 Collins Road NE
Cedar Rapids, Iowa 52498
(319) 395−8008

spmiller@pobox.cca.rockwell.com

Rockwell International ©1995

Who We Are

- **Rockwell International**
  - 75,000 Employees
  - Approximately 70% Sales are Commercial (Non–Government)
- **Collins Commercial Avionics (CCA)**
  - Cedar Rapids, Iowa
  - Civil Air Transport (Boeing, Lockheed, Fokker)
  - Civil General Aviation (Canadair, Beech, Cessna, Falcon, Embraer)
- **Collins Communication and Avionics Division (CACD)**
  - Cedar Rapids, Iowa
  - Defense and Government Systems (DOD and NASA)

Rockwell International ©1995

What We Do

- **Collins Commercial Avionics (CCA)**
  - Autopilot and Autoland Systems
  - Avionics Displays (EFIS, EICAS, PFD)
  - Traffic Alert and Collision Avoidance Systems (TCAS)
  - Communications Systems (Satellite)
  - Satellite Navigation Systems
- **Collins Communication and Avionics Division (CACD)**
  - Global Positioning Systems (GPS)
  - Joint Tactical Information Display System (JTIDS)
  - Mission Management Cockpit Display Systems (F–22)
  - Communication Systems

Rockwell International ©1995

Requirements Analysis

Needs & Opportunities

- **Better Methods for Requirements Analysis is Our Single Greatest Need**
- ***Requirements are incomplete, misunderstood, poorly defined, and change in ways that are difficult to manage*** **– Software Productivity Consortium**
- **A Methodology for Requirements Analysis Must**
  - Be Readable by Domain Experts and Our Customers
  - Accommodate Frequent Change
  - Discourage Overspecification
  - Be Precise Enough to Support Formal Analysis and Development of Tools
  - Support Generation of Test Cases
  - Support Traceability
  - Tell You When You're Done

Rockwell International ©1995

Requirements Analysis

State of Formal Methods

- **"Traditional" Formal Methods Lack Methods**
  - User Must Select Appropriate Approach for Each Problem Domain
  - Few "How To" Books
  - Success Very Dependent on the Skill of the User
- **There are a Few Formal or Semi–Formal Methods Emerging**
  - SCR – Software Cost Reduction Method (NRL)
  - CoRE – Consortium Requirements Engineering Method (SPC)
  - RSML – Requirements State Machine Language (Irvine)
- **Underlying Formalism is Still Evolving**
- **Need More Tools**

Requirements Analysis

What Should NASA Do?

- **Establish Program in Requirements Engineering**
- **Collaborate with Other Groups Working on Requirements Engineering**
  - Help to Complete the Formalization of Models
  - Support Development of Tools
- **Emphasize Industry/Academic/Government Collaboration**
  - Use Pilot Projects to Validate Methods and Tools

Software Testing

Needs & Opportunities

- **Software Testing is Our Single Most Costly Activity**
- **Structural Unit Testing of Software on a Level A (Highest Criticality) System**
  - Can Consume Over 50% of Total Project Budget (Hardware & Software)
  - Yet Finds Remarkably Few Errors
- **Rapidly Moving Towards Requirements Based Testing**
  - Create Tests from Requirements and Design Specifications
  - Determine Structural Coverage of Code as a Byproduct
- **Needs**
  - Execute Tests Against the Specifications
    - *Validate Requirements and Designs*
    - *Facilitate Early Development of Test Suites*
  - Automatically Generate Tests from Specifications

Software Testing

State of Formal Methods

- **Need Methods for Formally Specifying Requirements and Designs**
- **Need Tools to Animate Specifications**
- **Need Methods for Generating Tests from Formal Specifications**
  - Will Test Generation Differ for Each Specification Method?
  - What Distinguishes a "Useful" Test?

Software Testing

What Should NASA Do?

- **Establish Program in Requirements Engineering**
- **Collaborate with Other Groups Working on Requirements Engineering**
  - Explore Methods for Generation of Tests
  - Support Development of Tools
- **Emphasize Industry/Academic/Government Collaboration**
  - Use Pilot Projects to Validate Methods and Tools

Logical Partitioning

Needs & Opportunities

- **Isolation of Critical from Non–Critical Functions On a Single Processing Site**
  - Guarantee a Non–Critical Function Cannot Interfere with a Critical Function
  - Currently Provided by *Federated Architectures* Found in Most Civil Aircraft
- **Proposed as a Mechanism to Reduce Costs and Improve Reliability**
  - Reduce Flight Deck Space, Power, and Interconnect
  - Improve Reliability Through Use of Common Processing and I/O Units
- **Also Essential for Coping with Increasing Demands for Functionality**
  - Partition System in Critical and Non–Critical Functions
  - Focus Verification on Critical Functions
- **Partitioning Mechanism Becomes the Most Critical Component**

Logical Partitioning

State of Formal Methods

- **Natural Application for "Traditional" Formal Methods**
  - Fault–Tolerant Designs
  - Complex Mixture of Hardware and Software
  - Requires Extremely High Levels of Assurance
- **Little Work by the Formal Methods Community**

Logical Partitioning

What Should NASA Do?

- **Lead Application of Formal Methods to Logical Partitioning**
  - Rigorously Define the Basic Requirements of Logical Partitioning
  - Explore a Variety of Design Alternatives
  - Formally Verify the Correctness of Key Algorithms and Designs
- **Emphasize Industry/Academic/Government Collaboration**
  - Support Joint Efforts to Ensure Right Problems are Explored

Fault Tolerant Computing

- **Most Avionics Systems are Fault Tolerant**
- **Extensive Work by Formal Methods Community in Fault Tolerant Computing**
  - Software Implemented Fault–Tolerant Computer (SIFT)
  - Reliable Computing Platform (RCP)
- **Little Affect on Commercial Avionics Industry**
- **Why Aren't These Approaches being Used?**
- **Are There Lessons to be Learned Here?**

Rockwell International ©1995