Formal Methods for System/Software Engineering: NASA & Army Experiences

Dr. Mike Hinchey/GSFC
Caroline Wang/MSFC
Josh McNeil/ARMY

Introduction

- What are Formal Methods?
- Problem/Approach
- Challenges
- Recommendations
- Future Plans

Formal Methods - Dr. Mike Hinchey

Formal Methods

- Formal methods are mathematically based techniques for specification, development and verification of systems, both hardware and software.
- The use of formal methods approaches can help to eliminate errors early in the design process.
- Practitioners have also recognized that they can make searching for reusable components more effective by having formal specifications of components.

Current Formal Methods activities within NASA/Army, and International Formal Methods community.

- Pockets of expertise within NASA (specifically ARC, JPL, LaRC) and Army.
- Tools and techniques in use within NASA and Army but not widely used on projects and missions.
- International Formal Methods Community

Problem/Approach

Formal Methods

General Problem	Approach
• System/Hardware/Software complexity	• Provide accurate and appropriate specifications of required system behavior using Formal Methods
• Inadequate requirements specifications / misinterpretation of natural language • Significant number of problems introduced due to vague requirements	• Develop requirement specification as Formal Specification (using formal semantics) to eliminate misinterpretation of vague and incomplete natural language requirements
• Significant number of safety and reliability problems are traced to incorrect performance or behavior specifications, or incorrect interfaces	• Use Formal methods to prove safety properties derived from safety analyses • Use Formal Methods and deductive apparatus to prove correctness of system behavior and interfaces

Problem/Approach

Formal Methods

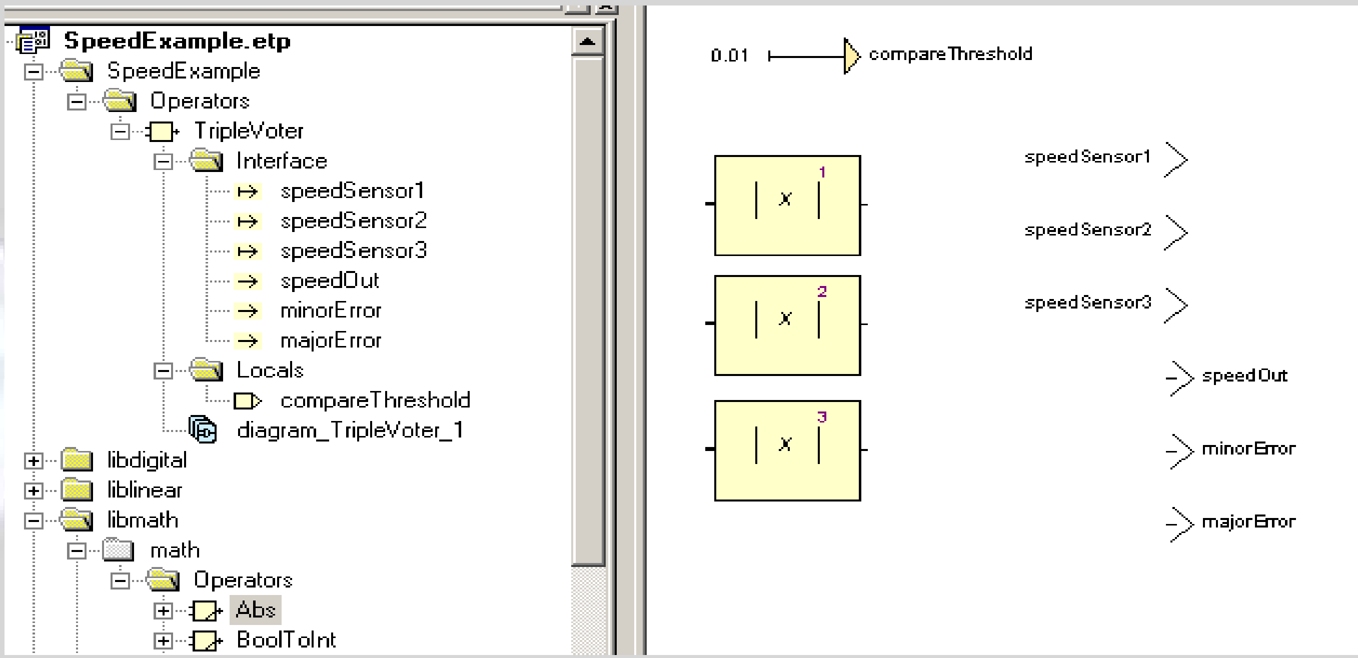
Specific Problem	Approach
• Formal Methods Learning Process Difficult for new users	• Develop specific project related case studies and provide examples for potential users
• Select development tools No time to learn all the tools Inadequate resource	• Based on the project size and resources available, select appropriate Formal Methods development techniques and tools
• Budget and Schedule constraints	• Support program development and in parallel prove potential savings
• Differences in priorities between Research and Production environments	• Many researchers focus on development of new techniques and tools • Production or development programs are concerned with delivery of a product • Need to build bridges between the research and production environments

Challenges

- High cost of some commercial development tools.
- Open source free tools do not have adequate training material and support.
- Formal Methods tools require extensive learning process.
- Die-hard Systems and Hardware Engineers are not convinced of the importance of software.

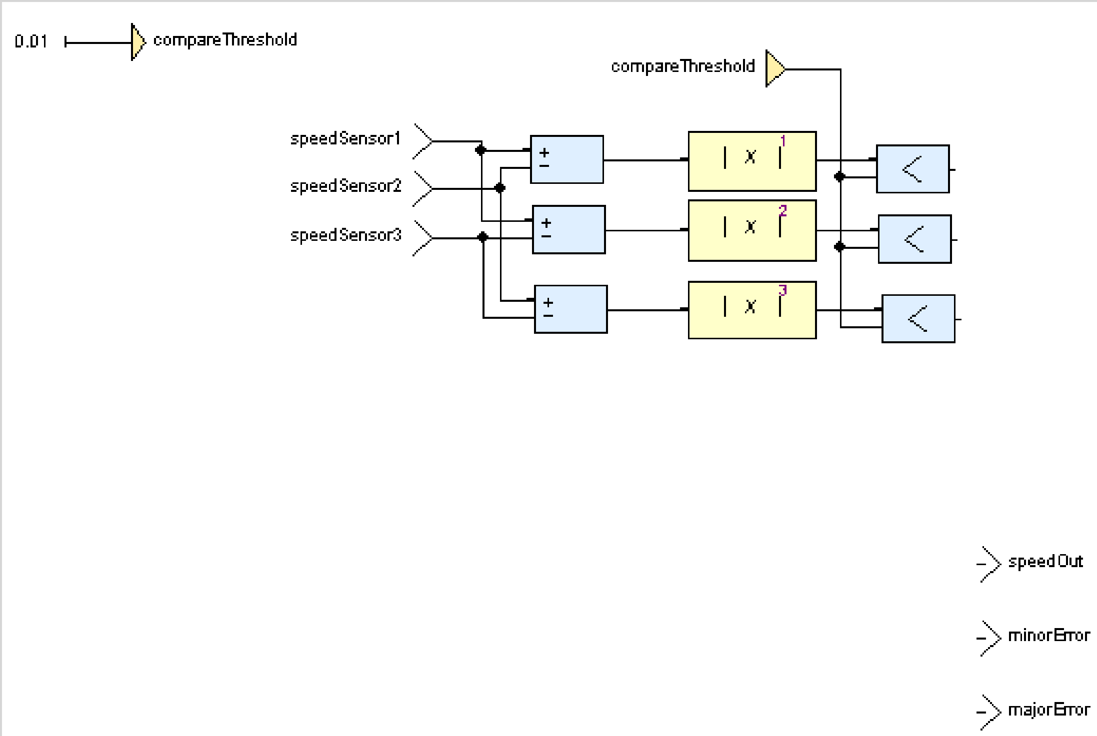
Developing TripleVoter Model

- Double-click the TripleVoter operator to begin modeling.
- Select all variables (speedSensor1, speedSensor2, speedSensor3, speedOut, minorError, majorError, and compareThreshold). Drag them onto the diagram.
- Select the compareThreshold local variable, modify it through Properties → Use, and change its use to Out.



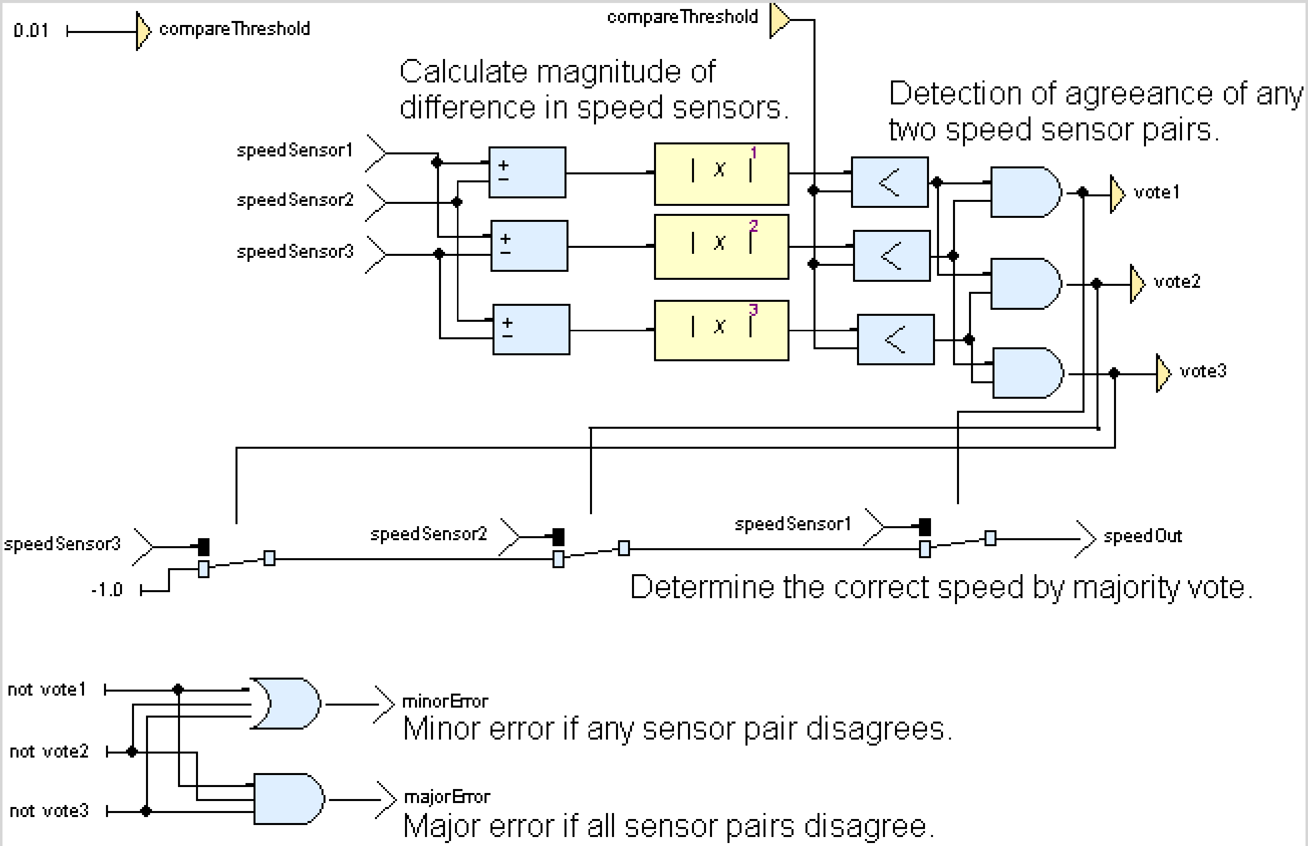
Implementing Model Logic

- Connect speedSensorX to the "+" input and speedSensorX to the "-" input of the New Minus operator.
- Connect speedSensor1, speedSensor2, and speedSensor3 to the first input of each New Minus operator.
- Connect all outputs of the New Minus operators to the inputs of the Abs operators.



Completing The Model Logic

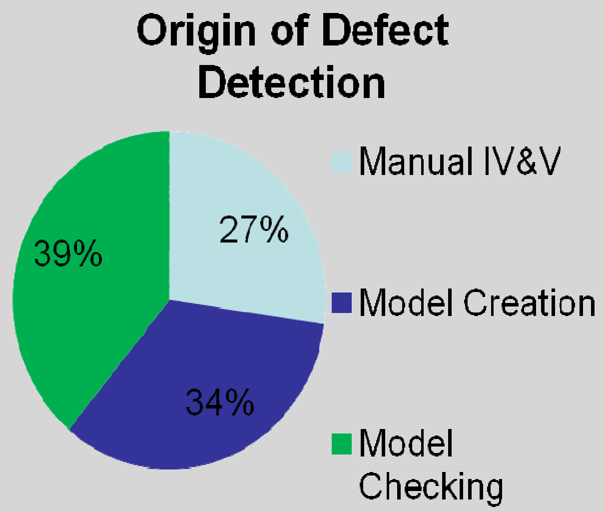
- Complete other logic components by drag and drop or connections.
- Add new If..Then..Else operators ( ) to the diagram.
- Add comments to model for readability
- **Design Verification –** Design Verifier can be used to develop properties that can be proven by formal methods.



Army's experience and Return on Investment

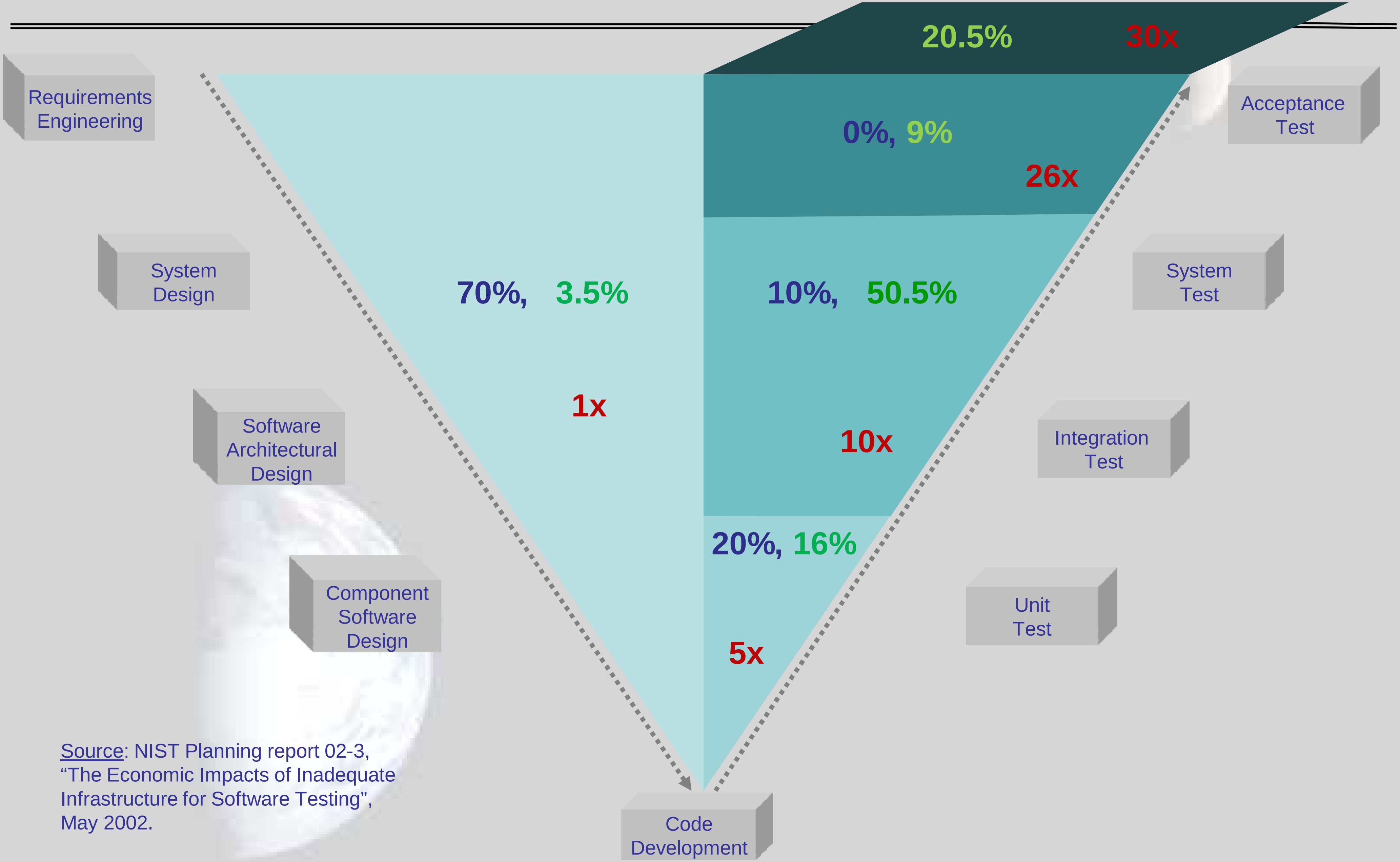
- Formal methods approach using SCADE method found 144 defects ***their*** traditional IV&V would miss (73% of all defects found)

- Estimating it would cost approximately 3500 man hours at $100 per man hour to fix the 144 defects later in the lifecycle
- Early defect removal savings is $350K
- The cost to perform formal methods analysis: -$137K
- Net savings of $213K or 5% of the total project



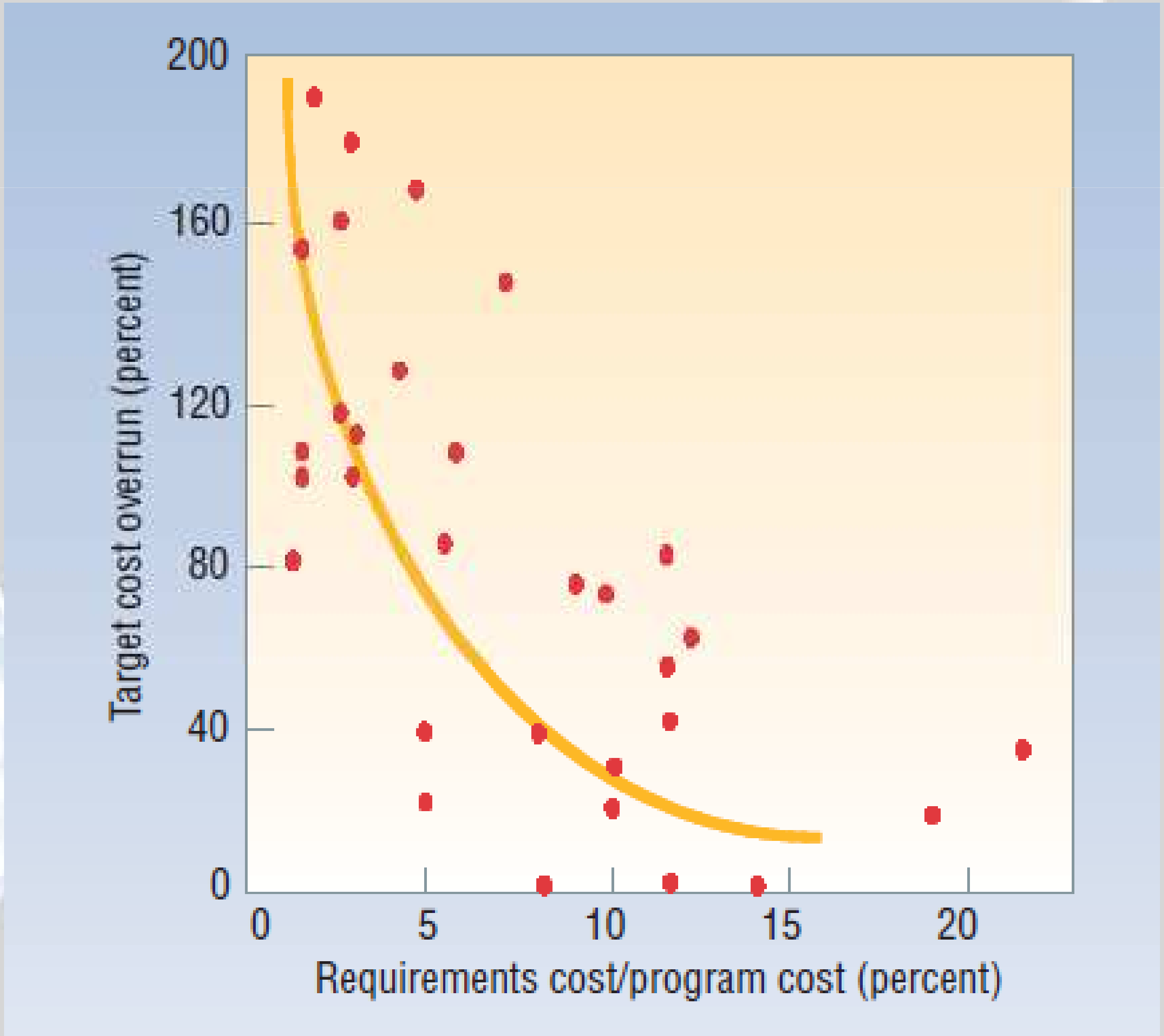
Savings could be even higher if defect detected earlier

The Army "V" concept
Where are faults introduced, discovered and cost for removal



Source: NIST Planning report 02-3, "The Economic Impacts of Inadequate Infrastructure for Software Testing", May 2002.

NASA Cost overruns



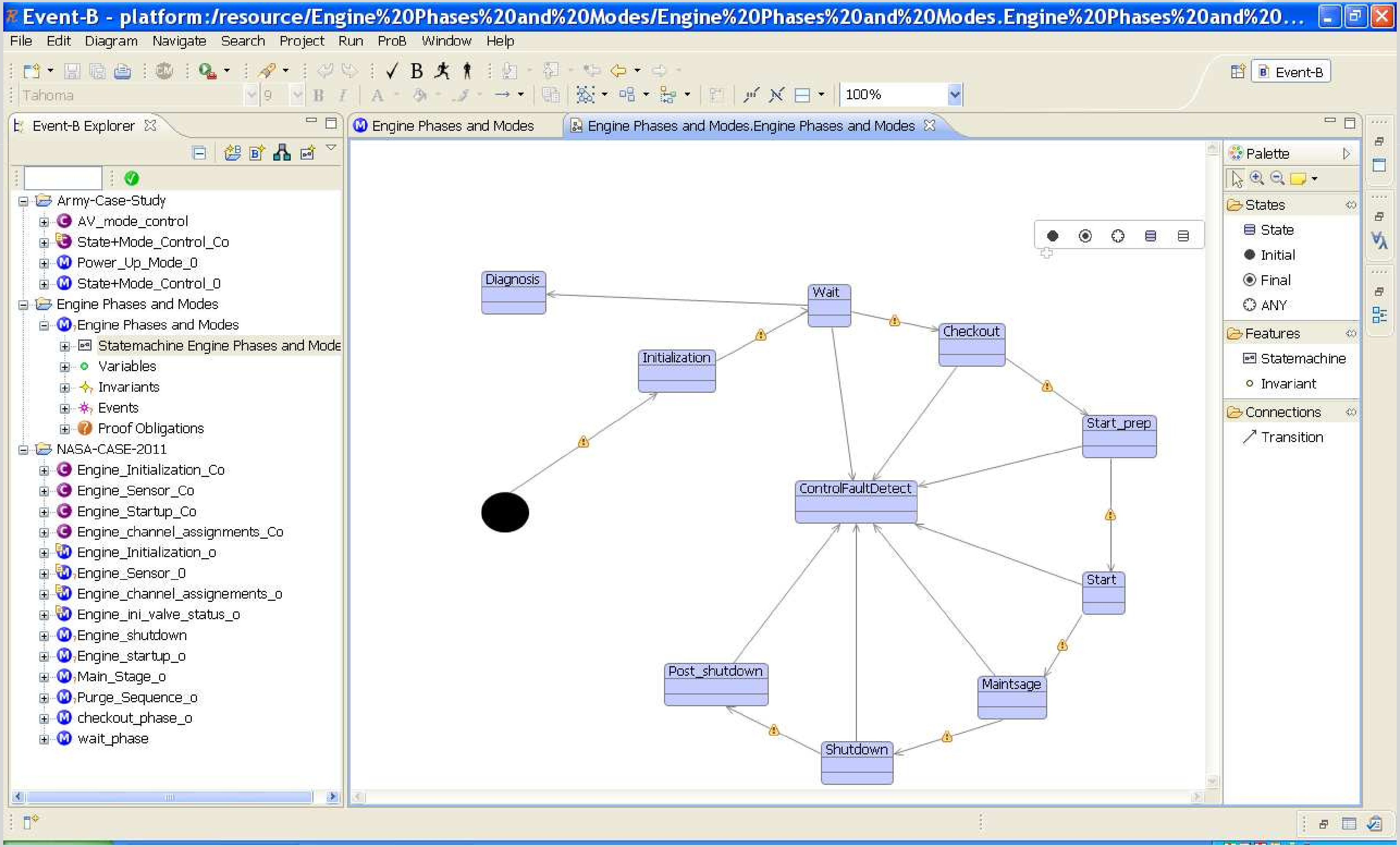
NASA MSFC Experience in this study

- Using open source development environments
  - B-Tool kit
  - Rodin Event B
  - EA UML
  - Integrated Rodin Event B and UML B

- Currently migrating all the work to the integrated Rodin Event B and UML B.
- Developed top level diagram and state machine in UML B, and used auto translator to translate into Rodin Event B.
- Using Rodin Event B platform for detailed refinement.
- The community is working on auto coding from Event B.

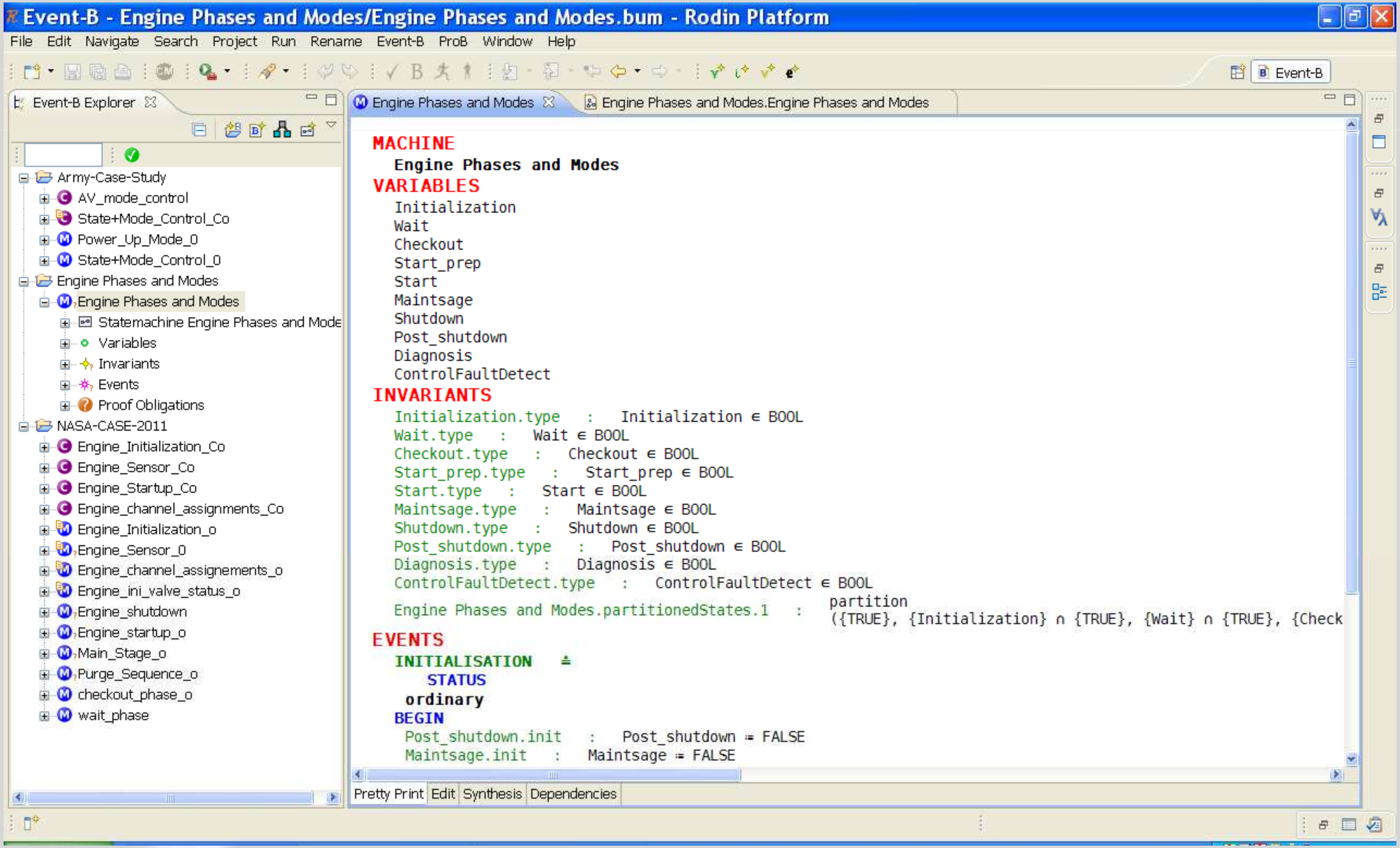
UML-B Statemachine

Formal Methods



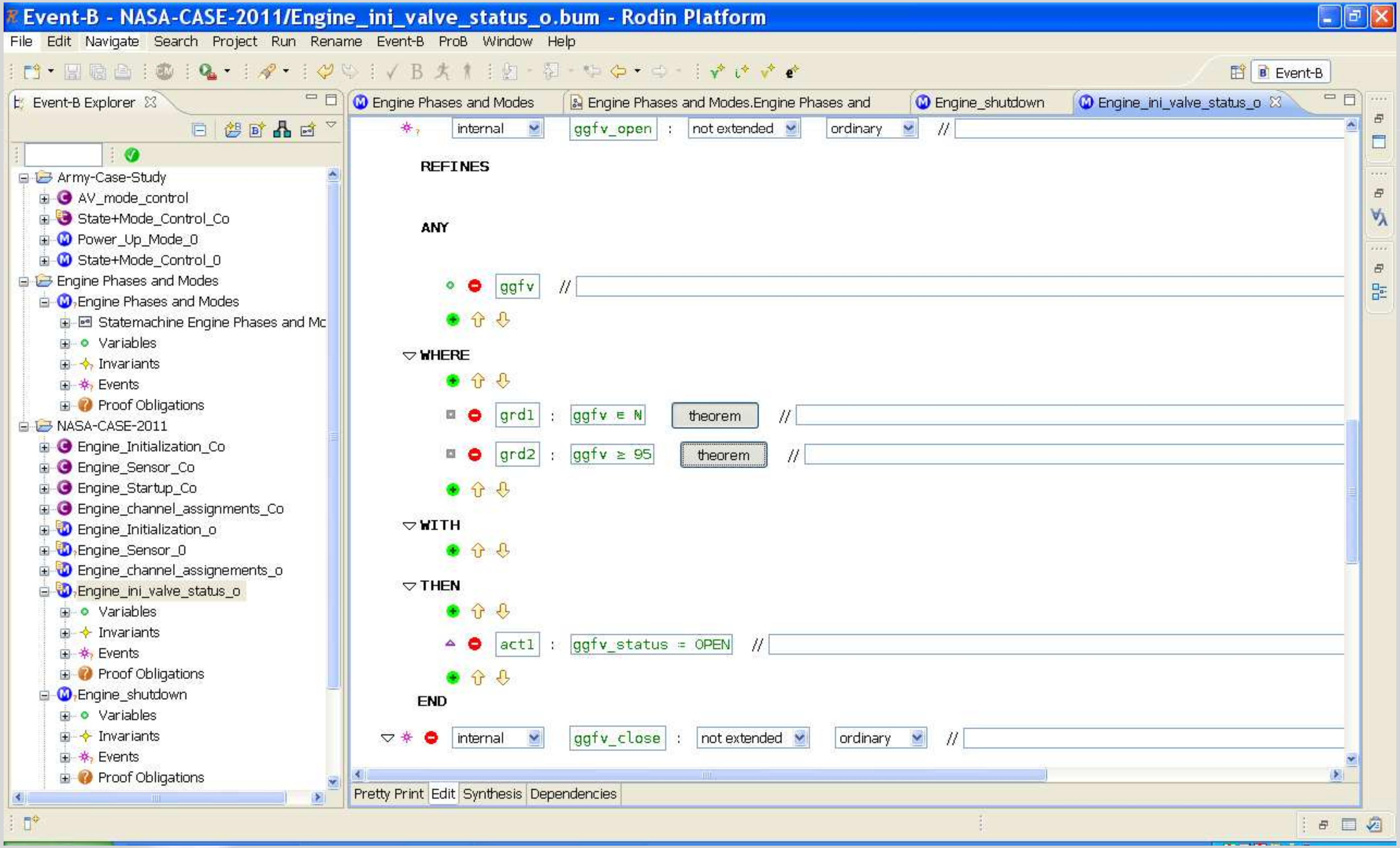
Auto Translation to Event B

Formal Methods

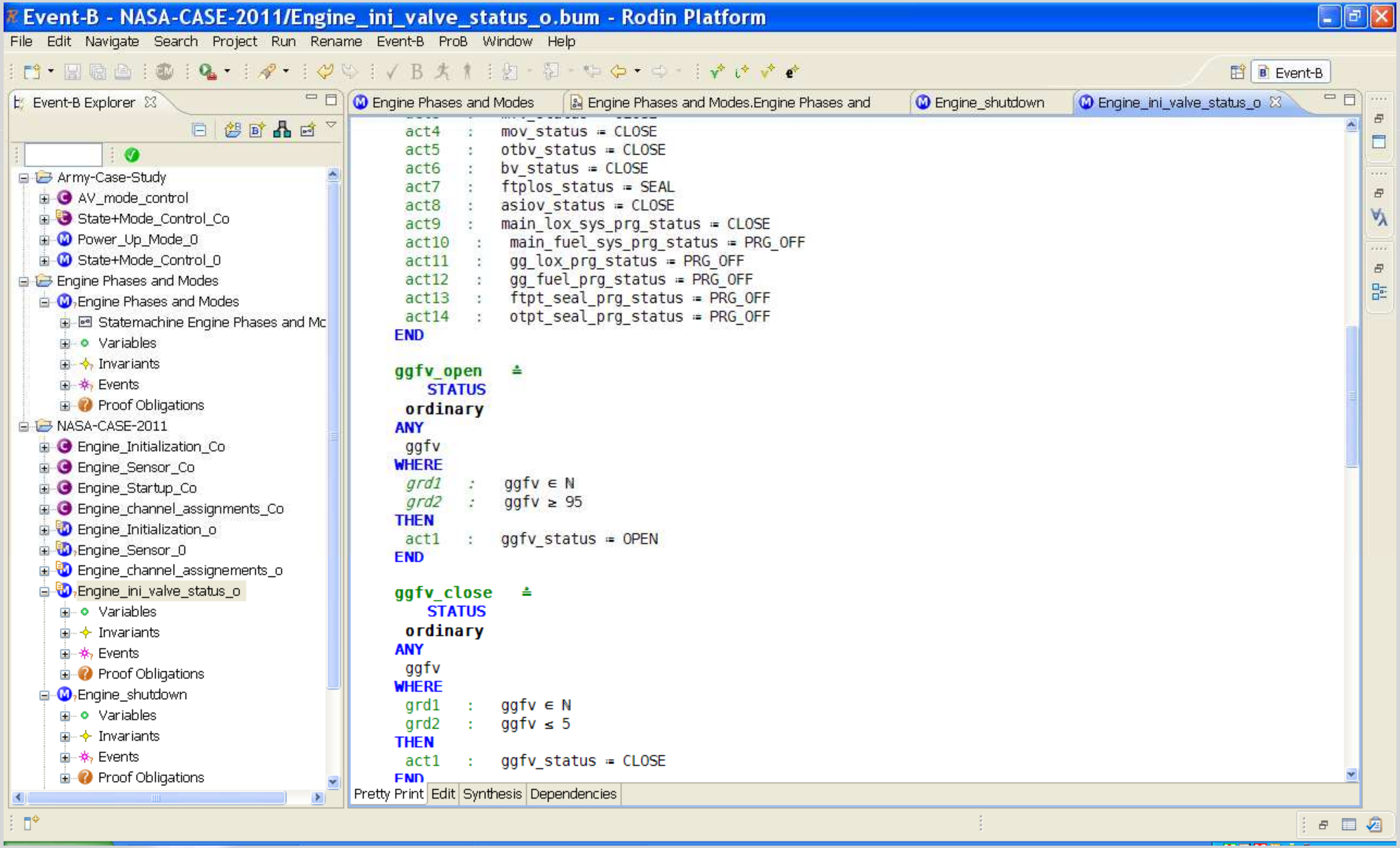


Event B Editor

Formal Methods



Event B prettyprint



NASA/Army Experience -Learning curve

- Unlike other tools, Formal Methods requires serious study
  - Formal Methods Language (B, Z…)
  - Formal Methods Development platform (Rodin, Event – B…UML, UML-B…)
  - Mathematical symbols, rules, logic…
- Training on Formal Methods is necessary
  - Engineers with better understanding of the project
  - Eliminate errors
  - Reduce Design complications and time
  - Encourage Engineers with better mathematics and science
- Easy is not the best solution for NASA and Army
  - Easy tools are easy to sell, but not able to solve our real problems

Recommendations

- **High cost tool**
  - Powerful, but not affordable to most of the organizations
  - Army used SCADE and Simulink with Design Verifier as a modeling tool.
- **Open Source**
  - No cost, but high learning curve and lack of support
  - Training program will significantly reduce the learning curve, this can be used for large community.
- **Recommendations:**
  - Project requiring immediate results may need to use high cost tools.
  - Continue monitoring open source tools (e.g. Integrated Rodin Event B and UML B) which will likely become more advanced in the future.

Results

- Formal methods can have significant cost savings.
- Defects can be found earlier when easier and cheaper to fix (cf. Army experience).
- While FMs are difficult to use and learn, a typical engineer *can* use them successfully when given appropriate support.
- Numerous tools are available. Choice is determined by:
  - Cost
  - Support
  - Deadlines
- Free (or cheap) is not necessarily best.

Future Plans

- Continue monitoring new and emerging Formal Methods techniques for practical usefulness and applicability to critical NASA/Army systems and software development activities.
- Complete Case study for both NASA/Army subsystems.
- Army is utilizing Formal Methods techniques for current programs.
- Complete Guidebook with road maps for future users.
- Pursue training opportunities with NASA STEP training office.
- Continue to emphasize awareness in Formal Methods and related training program

Contact Information

- Caroline K. Wang
  - Caroline.k.wang@nasa.gov
- Dr. Mike Hinchey
  - mike.hinchey@lero.ie
- Josh McNeil
  - Josh.McNeil@us.army.mil