Advanced Software V&V for Civil Aviation and Autonomy

Dr. Guillaume Brat

NASA Ames Research Center
Motivation for V&V research

The Decadal Survey for Civil Aeronautics and the NextGen Integrated WorkPlan both call for more research on the validation and verification of complex systems.

Example of typical cost in Aviation

- Software costs are very high
- V&V cost is 40-50% of the SW cost
- Driven by certification requirements
Reasons for the high cost of S/W

~80-90% of faults introduced here

~96% of faults found here

Phase in which error was detected and corrected
Areas addressed by NASA tools

- **Requirements Engineering**
- **System Design**
- **Software Architectural Design**
- **Component Software Design**
- **Code Development**
- **Unit Test**
- **Integration Test**
- **System Test**
- **Acceptance Test**

**Software Architectural Design**
- **IKOS**
- **SeaHorn**
- **FramaC**

**Component Software Design**
- **RLES**

**Code Development**
- **CoCoSim**
- **VeriCA**
- **Model Conformance**

**Dependability/Safety Cases**
Support for reducing cost of late-lifecycle changes

**PVS Models & Analysis for Algorithms**

**FRET**

**MARGInS**
Current V&V Tools and Capabilities

- **Requirements**
  - Requires theorem prover expertise

- **Design**
  - Combination of formal method with control theory experts

- **Code**
  - Accessible to moderate/expert programmer
  - Requires proficiency in statistics

- **Testing**
  - Model checking for checking/guaranteeing safety requirements
  - Static code analysis for run-time errors and safety requirements
  - Statistical-based testing to learn unsafe boundaries of operation

- **Operation**
  - Algorithmic proofs using theorem proving
  - Blackbox

**Tools and Capabilities**

- PVS
- Simulink, C, Stateflow
- C limited C++

**EXPERTISE**
- Requires theorem prover expertise

**TARGET**
- Requires proficiency in statistics

**CAPABILITY**
- Accessible to moderate/expert programmer
- Statistical-based testing to learn unsafe boundaries of operation
Future V&V Tools and Capabilities

- Accessible to engineers
- Accessible to engineers
- Accessible to programmer
- Requires proficiency in statistics

- English-like
- SCADE
- MatLab
- Full C++
- Floating-point analysis
- blackbox

- Requirement capture and analysis
- Checking/guaranteeing safety requirements on design models
- Static code analysis for run-time errors
- Statistical-based testing to learn unsafe boundaries of operation

Requirements → Design → Code → Testing → Operation
FAA/Regulator Needs

Software Development Lifecycle (RTCA DO-178C / DO-278B) 
Assumes the requirements are correct and complete

- Update standards
- Training material

- Framework for new process

- Identify/develop new process
- Training material

- Train certifiers

NASA engagement

- Update standards and processes to allow for use of formal V&V methods
- Educate certifiers so that results from new V&V techniques can be understood and accepted

Employ new certification process

Safety Cases Assurance Cases
Assurance Cases

• An assurance case is
  – A set of assurance claims connected to a body of evidence through a structured argument, to provide a comprehensive, defensible and valid justification that a system meets its assurance requirements for a given application in a defined operating environment

  – A means for integrating safety and mission assurance (S&MA) information.
Assurance Cases

- **Standards**
  - DO-178, APRs, STDs
  - Guides, Handbooks, ...

- **Domain Model (Reusable Mission Concepts)**

- **Assurance Assets**
  - V&V artifacts
  - Safety artifacts
  - Design rationale
  - Engineering artifacts
  - Domain knowledge
  - Engineering processes

- **Tool Capabilities**

- **Assurance Case**
  - Structured database of assurance assets with tracing relations and semantics

- **System Attributes**
  - Safety, Security
  - Dependability
    - Reliability
    - Availability, ...
  - Performance

- **Assurance Dashboard**
  - Metrics
  - Status/Progress
  - Visual analytics
  - Confidence assessment

- **Report generation**

- **Advice**

- **Stakeholder views**
Assurance Cases and Lifecycle

Assurance Dashboard
- Metrics
- Status/Progress
- Visual analytics
- Confidence assessment

Report generation
Advice
Stakeholder views

System Attributes
- Safety, Security
- Dependability
  - Reliability
  - Availability, ...
- Performance

Assurance

Assurance Case
- Structured database of assurance assets with tracing relations and semantics

Compliance

Domain Model (Reusable Mission Concepts)

Standards
- DO-178, APRs, STDs
- Guides, Handbooks, ...

Assurance Assets
- V&V artifacts
- Safety artifacts
- Design rationale
- Engineering artifacts
- Domain knowledge
- Engineering processes

Tool Capabilities

Applies to system lifecycle processes, from “cradle to grave”
Assurance Cases and Lifecycle

Updated dynamically as environment/system evolves (e.g., with maintenance)

Standards
- DO-178, APRs, STDs
- Guides, Handbooks, ...

Domain Model (Reusable Mission Concepts)

Assurance Assets
- V&V artifacts
- Safety artifacts
- Design rationale
- Engineering artifacts
- Domain knowledge
- Engineering processes

Tool Capabilities

Assurance Case
- Structured database of assurance assets with tracing relations and semantics

System Attributes
- Safety, Security
- Dependability
  - Reliability
  - Availability, ...
- Performance

Assurance Dashboard
- Metrics
- Status/Progress
- Visual analytics
- Confidence assessment

Report generation

Advice

Stakeholder views
Assurance Cases and Autonomy

Plan → Build → Operate → Retire

Analysis

Formal methods / tools

Assurance properties, Safety policies, requirements, …

Verification evidence

Assurance Case
• Structured database of assurance assets with tracing relations and semantics

Learn & Adapt

Maintain

Concept and Design Assurance
Assurance Cases and Autonomy

Assurance Case
- Structured database of assurance assets with tracing relations and semantics

Run-time Monitoring
- Safety performance
- Hazard precursors

Risk / Confidence Assessment
- Risk quantification
- Monitor generation

Safety Architecture
- Bow-tie model
- Escalation factors

Operational / Run-time Assurance

Plan → Build → Operate → Retire

Learn & Adapt

Maintain

Tracking

Operational risk management
Assurance Cases and Autonomy

Assurance Case
• Structured database of assurance assets with tracing relations and semantics

Safety Architecture
• Bow-tie model
• Escalation factors

Risk / Confidence Assessment
• Safety performance
• Hazard precursors

Operational / Run-time Assurance

Plan → Build → Operate → Retire

Learn & Adapt
Maintain

Real-time update
Data driven update

Run-time Monitoring
• Safety performance
• Hazard precursors

Barrier update
Risk update

Tracking

Updates from operations
Conclusions

• **Goal**: Address the impact of V&V of overall cost of S/W for aviation

• **Solution**: Bring V&V earlier in the lifecycle by using formal methods

• **Status**: Prototype tools for all phases
  - Requirement tool is in its infancy

• **Innovation**: gather V&V evidences in assurance cases that extend throughout the lifecycle

• **Future**: Address V&V of autonomy through the use of assurance cases at runtime