Are You Qualified For This Position?

High Confidence Systems and Software

5 May 2015

Darren Cofer

darren.cofer@rockwellcollins.com

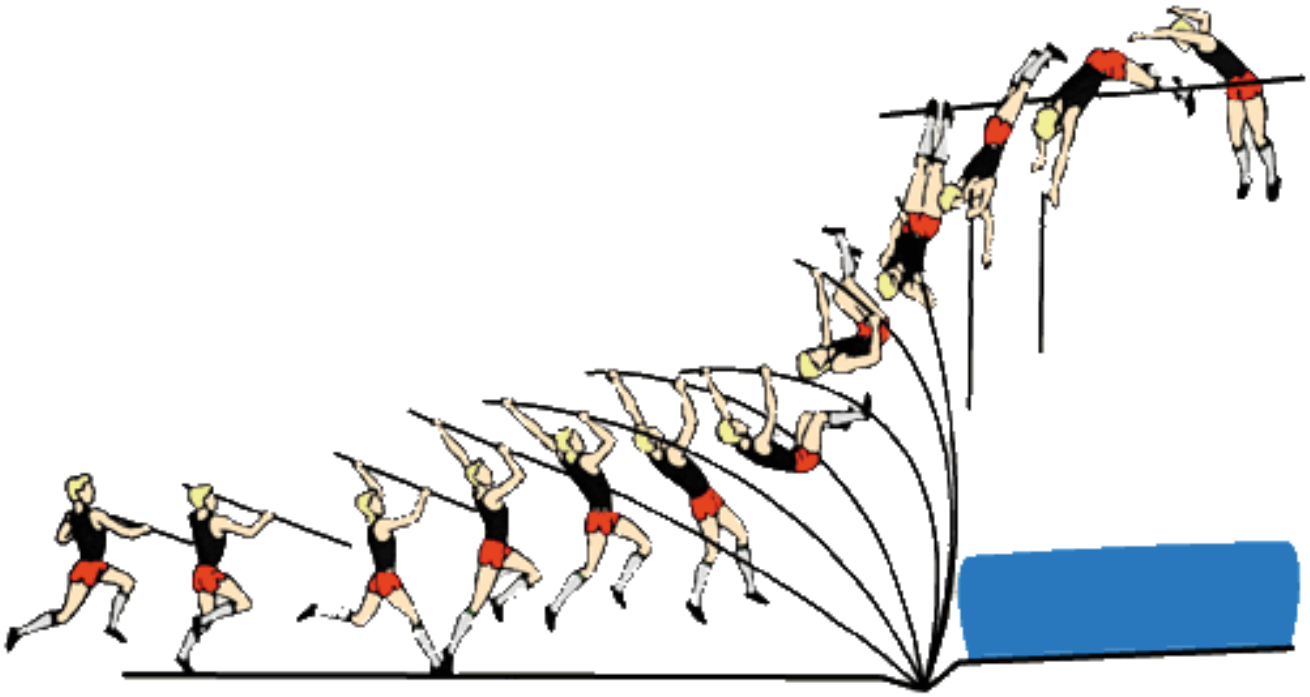
Rockwell Collins


Certification authority
Verification
Proof certificate
Validation
Certifying compiler
Trust
Assurance
Trustworthy
CERTIFICATION
QUALIFICATION
Safety
Guidance
Requirements
Regulations
Security

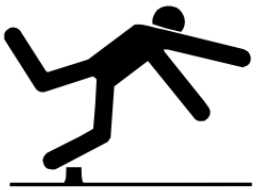
Problem

- Formal methods tools have been shown to be effective at finding defects in and verifying the correctness of safety-critical software.
- Many safety-critical domains (aviation, rail, nuclear, medical) are regulated and have requirements for **certification**.
- Certification processes generally require **qualification** of any tools/automation used.
- Tool qualification is not a widely understood concept outside of those industries requiring certification for high-assurance.

Mismatched expectations



The "bar" for tool assurance as perceived by formal methods researchers



The actual "bar" for qualification of software tools (for verification)

The Question

- How can we retain the high level of assurance in tools from the formal methods community without "raising the bar" on their qualification (and thereby discouraging their use)?

Dagstuhl Seminar

Share knowledge about requirements for certification and qualification of software tools so that formal methods researchers can better understand the challenges and barriers to the use of formal methods tools

- Evidence necessary to justify the application of formal methods tools in real safety-critical settings
- Examples of how to qualify different types of software tools
- Explore new approaches for the qualification of

SCHLOSS DAGSTUHL
Leibniz-Zentrum für Informatik

About Dagstuhl | Program | Publications | Library | dblp

You are here: Program » Seminar Calendar » Seminar Homepage

Dagstuhl Seminars
Dagstuhl Perspectives
GI-Dagstuhl Seminars
Events
Research Guests
Seminar Calendar
All Events

http://www.dagstuhl.de/15182

April 26 – 29 , 2015, Dagstuhl Seminar 15182

Qualification of Formal Methods Tools

Organizers

Darren Cofer (Rockwell Collins – Bloomington, US)
Gerwin Klein (NICTA – Sydney, AU)
Konrad Slind (Rockwell Collins – Bloomington, US)
Virginie Wiels (ONERA – Toulouse, FR)

< 3 / 3

For support, please contact

Dagmar Glaser for administrative matters
Andreas Dolzmann for scientific matters

Book exhibition

Books from the participants of the current Seminar
Book exhibition in the library, 1st floor, during the seminar week.

Documentation

In the series Dagstuhl Reports each Dagstuhl Seminar and Dagstuhl Perspectives Workshop is documented. The seminar organizers, in cooperation with the collector, prepare a report that includes contributions from the participants' talks together with a summary of the seminar.

Download overview leaflet (PDF).

Publications

Furthermore, a comprehensive peer-reviewed collection of research papers can be published in the series Dagstuhl Follow-Ups.

Dagstuhl's Impact

Please inform us when a publication was published as a result from your seminar. These publications are listed in the category Dagstuhl's Impact and are presented on a special shelf on the ground floor of the library.

Domain: Civil Aviation

- Safety-critical software
- Well-established guidance documents and regulatory structure
- Concrete example for discussion
- Other domains
  - Nuclear (IEC 61508/61513)
  - Rail (EN 50128)
  - Automotive (ISO 26262)

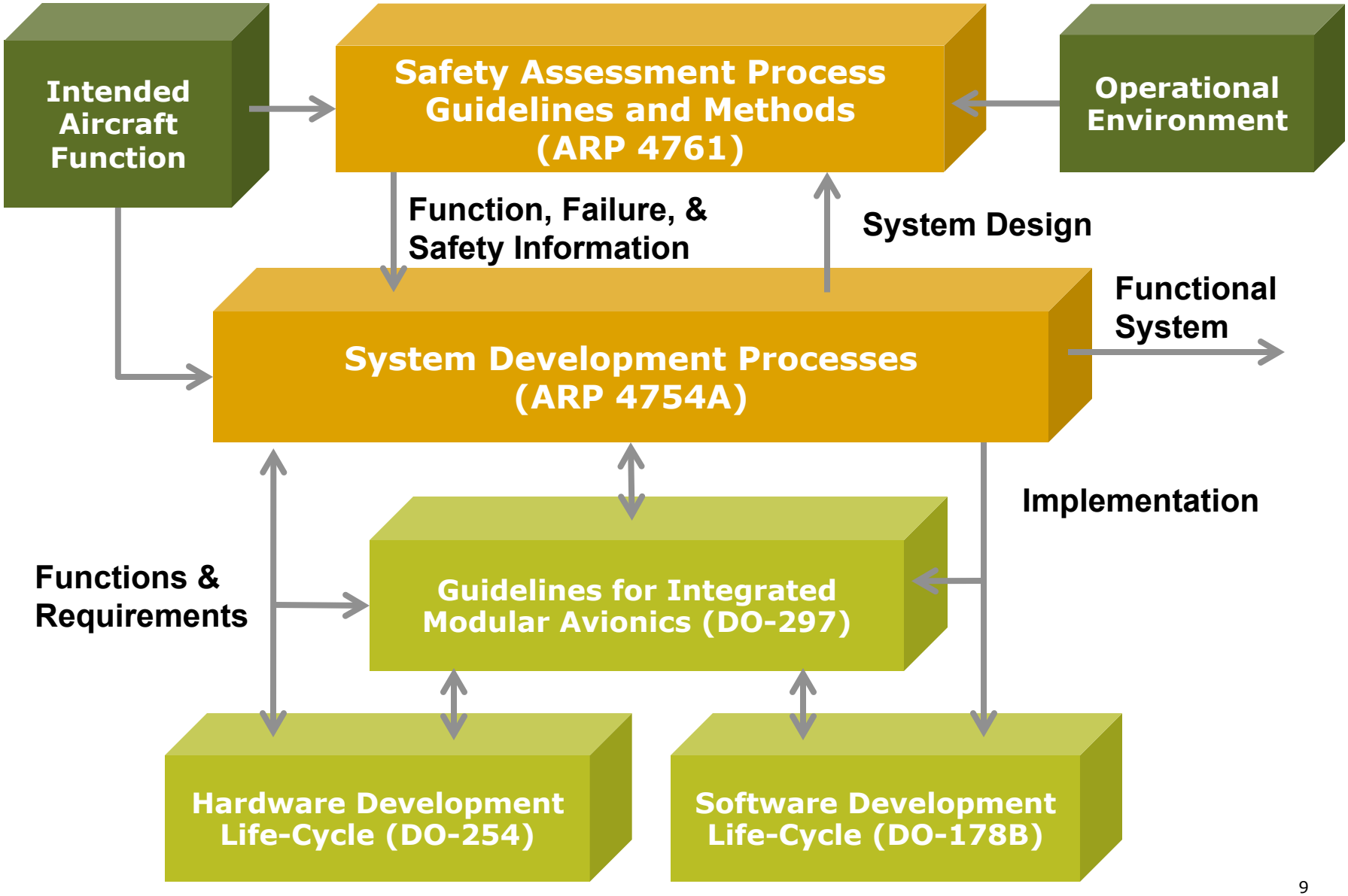
Mark Lawford, SCC

Definition 1: Certification

- Certification is the legal recognition by a regulatory authority that a product, service, organization, or person complies with the requirements (e.g., 14 CFR part 25).
  - Type Certification: design complies with standards to demonstrate adequate safety, security, etc.
  - Product conforms to certified type design
  - Certificate issued to document conformance
- Examples of certification evidence
  - We used verification tool X to accomplish these objectives.
  - These are the reasons why we think the tool is acceptable.
  - We ran 1000 tests using the tool, and this is why we think these 1000 tests are sufficient.
  - And (almost incidentally) here are the test results.

Convincing the relevant Certification Authority that all required steps have been taken to ensure the safety/reliability/integrity of the system

Certification Process for Civil Aviation



DO-178C (RTCA 2011)

"Software Considerations in Airborne Systems and Equipment Certification"

- Certification authorities agree that an applicant can use DO-178 as a means of compliance with federal regulations for airworthiness.
- Primarily a design assurance document (not safety)
  - Demonstrate that software implements requirements
  - and nothing else (no surprises)
- Requires auditable evidence of specific processes
  - Planning, Development, Verification, Configuration Management, Quality Assurance, Certification Liaison
- Five "Software Levels"
  - Design Assurance Level in other contexts
- Objective based
  - Specifies what is to be achieved, not how
  - Different objectives and requirements for
  - each software level
  - 71 objectives for Level A code

A: Catastrophic
(everyone dies)
B: Hazardous/Severe
(serious injuries)
C: Major
(significant reduction in safety margins)
D: Minor
(annoyance to crew)
E: No Effect
(OK to use Windows)

DO-176C Verification Objectives for Level A Software

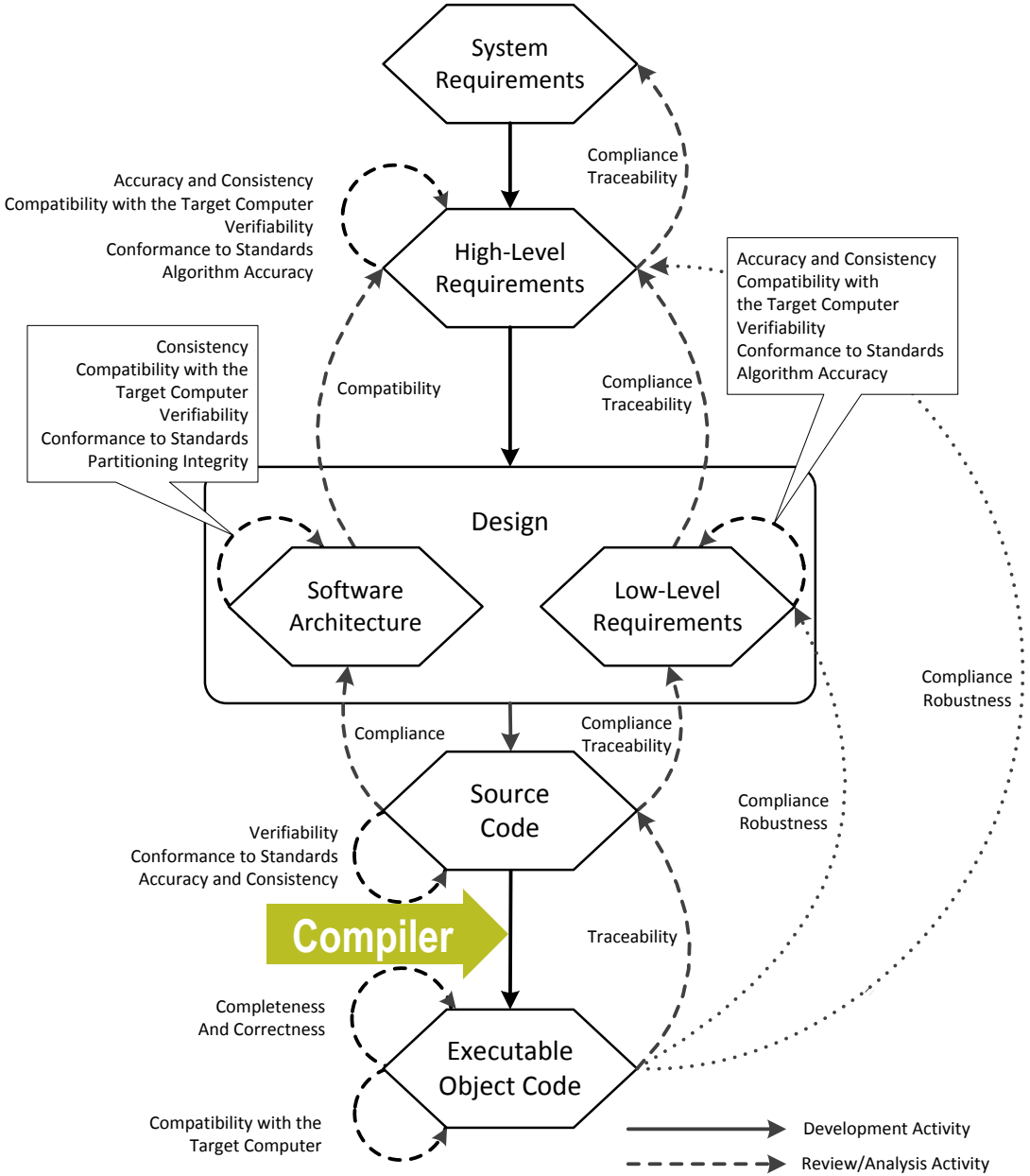
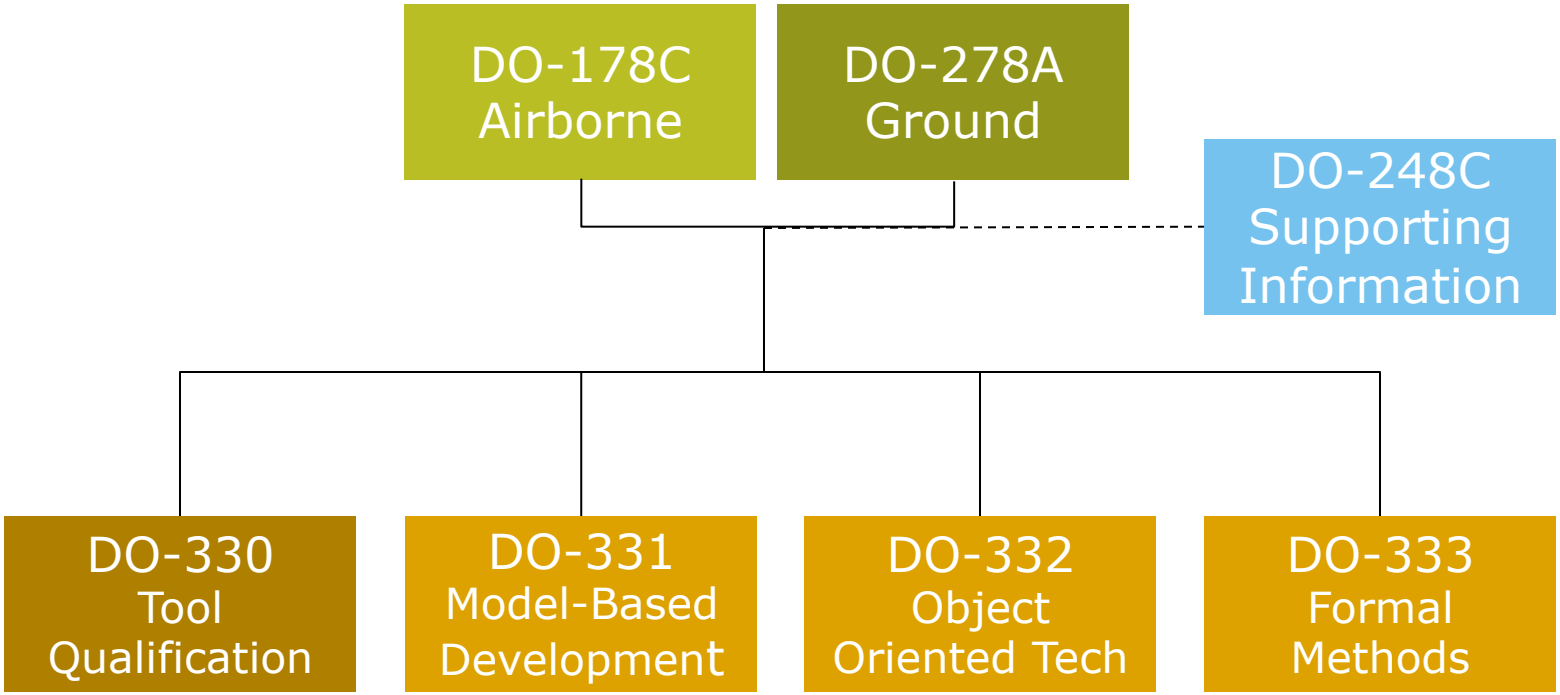


Diagram adapted from DO-333 Formal Methods Supplement to DO-178C and DO-278A

Note: Requirements include Derived Requirements

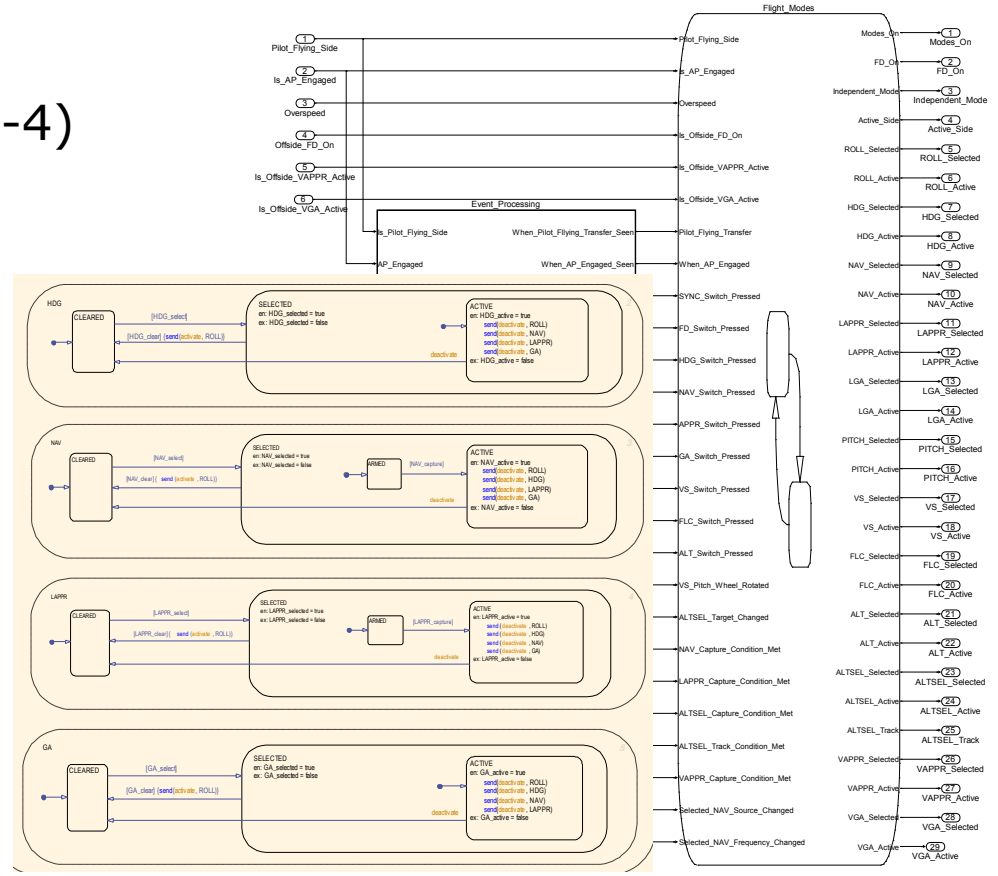
DO-178C (& friends)



Example: Model Checking

"Formal Methods Case Studies for DO-333"
NASA Contractor Report or
Loonwerks.com

- Mode logic for Flight Guidance System modeled in using Simulink/Stateflow
- Use model checker to satisfy DO-178C objectives (Table A-4) with guidance from DO-333, Formal Methods Supplement
  - LLR comply with HLR
  - LLR are accurate/consistent
- Example Requirements
  - Exactly one mode active
  - VAPPR implies LAPPR
  - Mode transitions correct
- Verification tools
  - NuSMV/Kind/SLDV
- **Can we trust tools?**



Definition 2: Qualification

- Tool qualification is the process necessary to obtain certification credit for the use a tool.
  - Note: this credit may only be granted within the context of a project requiring approval.
- Qualification of a tool is needed when certification processes are **eliminated**, **reduced**, or **automated** by the use of a software tool without its output being verified.
- The purpose of the tool qualification process is to ensure that the tool provides confidence **at least equivalent** to that of the processes eliminated, reduced, or automated.

Does my tool even need to be qualified?

Maybe not…

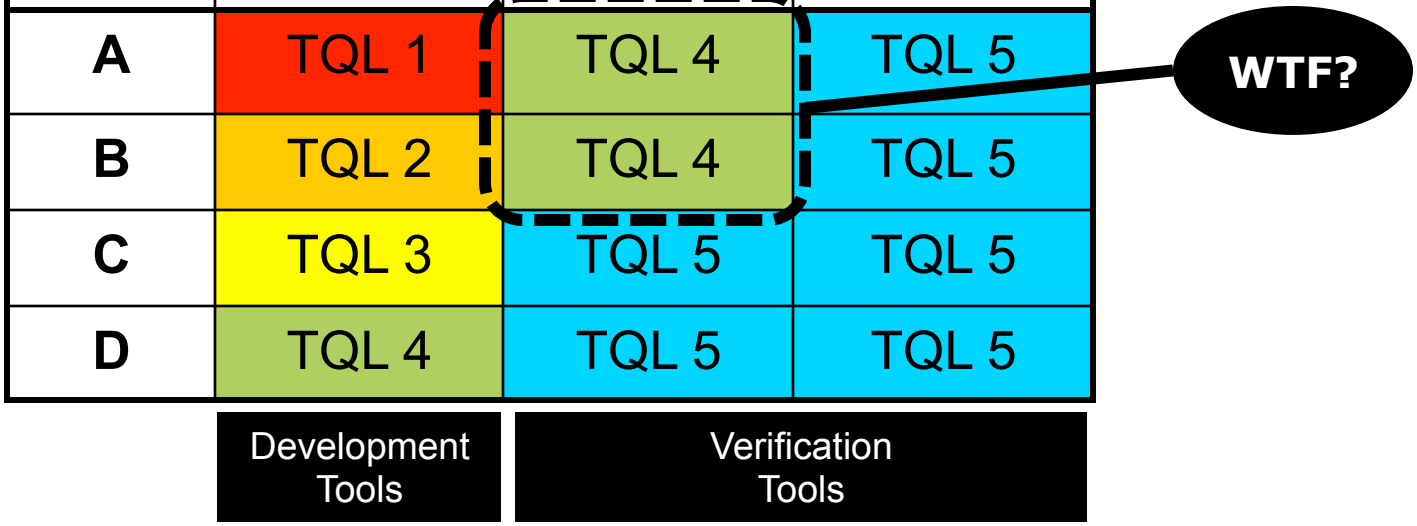
- Are you using it to satisfy some certification objective?
- Is your tool being used to eliminate, reduce, or automate a certification process?
- Is the output of the tool being verified?

Tool Qualification Level

- DO-178C added new **criteria** to determine the required tool qualification level (unique to aviation domain).
- Criteria
  1. A tool that automates development processes (output is part of the airborne software) and thus could insert an error
  2. A tool that automates verification processes and thus could fail to detect an error, **and** whose output is used to justify the elimination or reduction of
     - verification process other than that automated by the tool, **or**
     - development process which could have an impact on the airborne software
  3. A tool that automates verification processes and thus could fail to detect an error

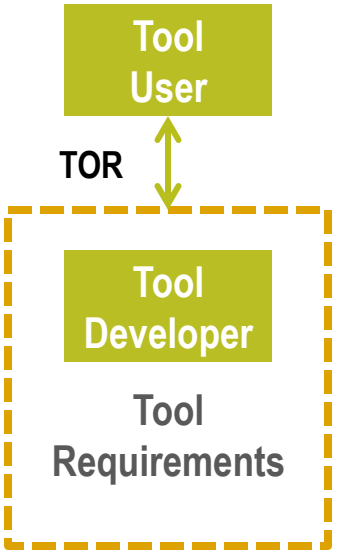
Tool Qualification Level

SW Levels	Criterion 1	Criterion 2	Criterion 3
A	TQL 1	TQL 4	TQL 5
B	TQL 2	TQL 4	TQL 5
C	TQL 3	TQL 5	TQL 5
D	TQL 4	TQL 5	TQL 5
	Development Tools	Verification Tools	



"The problem arises when, based on the confidence of a given verification activity, some alleviation is claimed for other objectives or activities that are not the direct purpose of that verification activity."

Tool Qualification Principles



- User context
  - Tool Operational Requirements (TOR)
    - What does the tool do from a user perspective?
  - Tool operational verification and validation
    - Verification: The tool is compliant with its TOR
    - Validation: The tool satisifies user needs
  - For TQL 5, only user context activities are required
  - **Expected evidence: test cases demonstrating compliance with TOR**
- Developer context
  - Tool development requirements are produced from the TOR
  - Development and verification objectives for the tool development processes, configuration management, etc.
  - **For TQL 1-4, tool must satisfy (essentially) same objectives as the safety-critical software itself**

Rockwell Collins

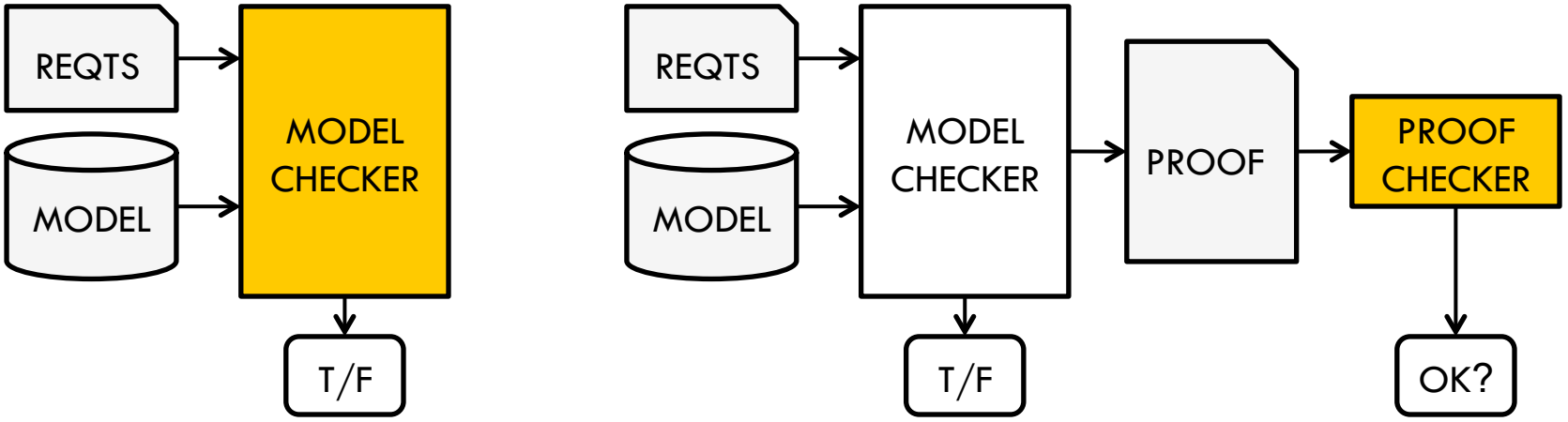
Tool Operational Processes
DO-330 Table T-0 Objectives

	Objective		Activity	Applicability by TQL					Output		Control Category by TQL				
	Description	Ref.	Ref.	1	2	3	4	5	Description	Ref.	1	2	3	4	5
	Planning Process														
1	The tool qualification need is established.	4.1	[Note 1]	❍	❍	❍	❍	❍	Tool-specific information in the Plan for Software Aspects of Certification	10.1.1	①	①	①	①	①
	Tool Operational Requirements Process														
2	Tool Operational Requirements are defined.	5.1.1.a	5.1.2.a 5.1.2.b 5.1.2.c	❍	❍	❍	❍	❍	Tool Operational Requirements	10.3.1	①	①	①	①	②
	Tool Operational Integration Process														
3	Tool Executable Object Code is installed in the tool operational environment.	5.3.1.a	5.3.2.a 5.3.2.b 5.3.2.c	❍	❍	❍	❍	❍	Tool Executable Object Code Tool Installation Report	10.2.4 10.3.2	② ②	② ②	② ②	② ②	② ②
	Tool Operational Verification and Validation Process														
4	Tool Operational Requirements are complete, accurate, verifiable, and consistent.	6.2.1.a	6.2.2.a	●	●	❍	❍		Tool Operational Verification and Validation Results	10.3.4	②	②	②	②	
5	Tool operation complies with the Tool Operational Requirements.	6.2.1.b	6.2.2.c	●	●	❍	❍	❍	Tool Operational Verification and Validation Cases and Procedures Tool Operational Verification and Validation Results	10.3.3 10.3.4	② ②	② ②	② ②	② ②	② ②
6	Tool Operational Requirements are sufficient and correct.	6.2.1.aa	6.2.2.b	●	●	❍	❍	❍	Tool Operational Verification and Validation Results	10.3.4	②	②	②	②	②
7	Software life cycle process needs are met by the tool.	6.2.1.bb	6.2.2.c	❍	❍	❍	❍	❍	Tool Operational Verification and Validation Cases and Procedures Tool Operational Verification and Validation Results	10.3.3 10.3.4	② ②	② ②	② ②	② ②	② ②

Soundness

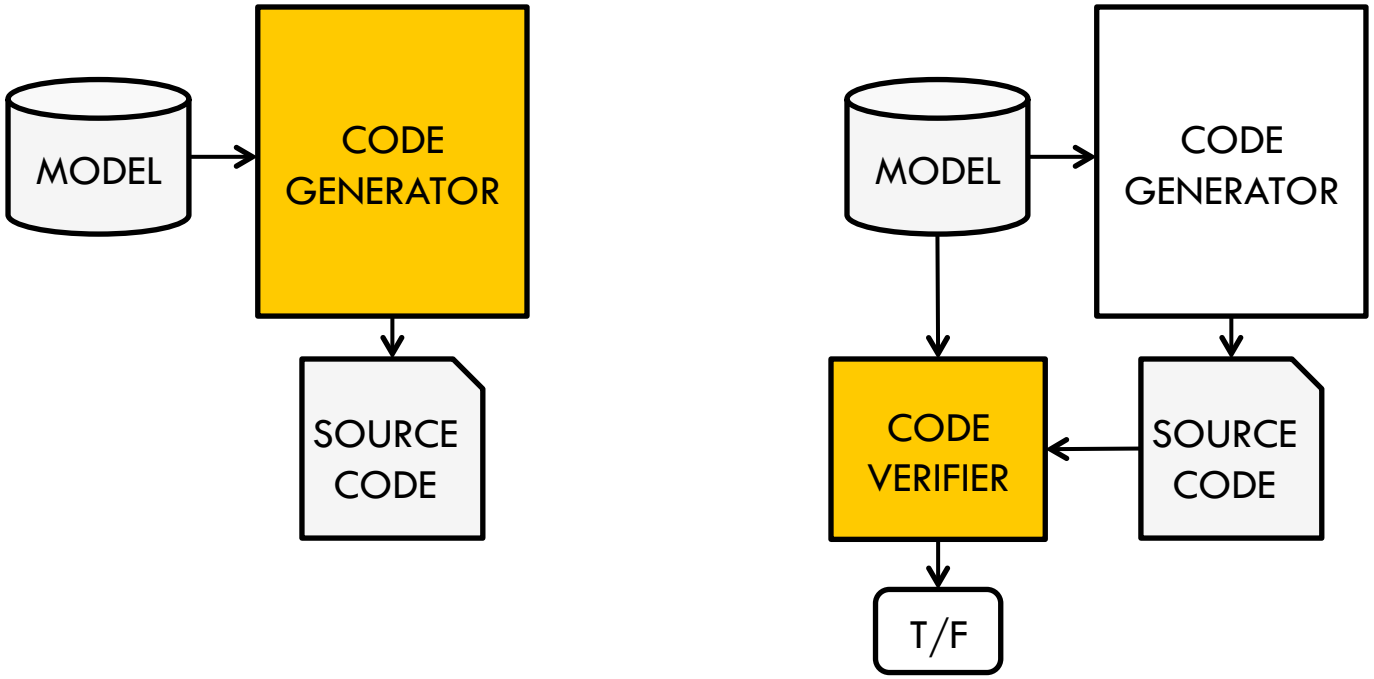
- DO-333 (Formal Methods Supplement) requires soundness of underlying **method**
  - A sound method never asserts that a property is true when it may not be true
  - Typical evidence: Peer-reviewed academic papers
  - Note: Not soundness of the tool!
- What about soundness of tools?
  - This was left as part of tool qualification
  - Don’t “raise the bar”

Different Approaches to FM Tool Qualification



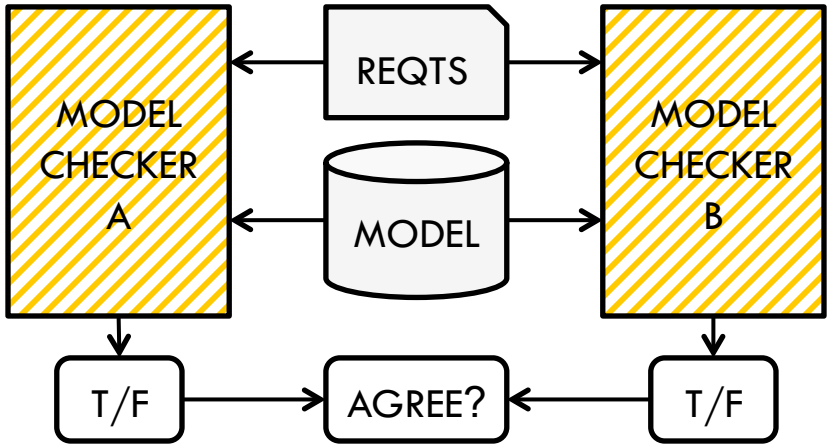
- Qualify a smaller, simpler checker?
- What could go wrong?

Different Approaches to FM Tool Qualification



- Instead of trying to qualify a development tool (TQL-1) can we qualify a code verifier instead (TQL-5)?
- See DO-330 FAQ D.7

Different Approaches to FM Tool Qualification



- Two independent tools that check each other's outputs
- Does either need to be qualified?
  - Probably

Observations

- For now
  - Qualification of development tools (TQL 1-4) is still difficult.
  - A qualified compiler or code generator does not buy you much.
  - Verification tool developers who want their tools to be used for certification credit should be able to differentiate between assurance research and evidence/documents needed for qualification.

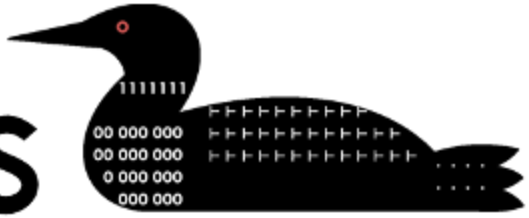
- The future
  - There is clearly a mismatch between the kinds of evidence required for tool qualification and the "right way" to establish assurance for FM tools (especially for development tools).
  - DO-330 tailored by DO-333? DO-330A?

Can I trust your tool?

- It depends…

- What are we relying on the tool for?
  - What objective is it accomplishing?
- What does "trust" mean?
  - Are we in a context where qualification is required?
  - Is the tool doing something that requires qualification?
- Qualification might not mean what you think it means
  - It might be easier (or harder) than you think

Loonwerks



More information, code, and papers available at:

Loonwerks.com