DO-178C IS NO SILVER BULLET
SOFTWARE SAFETY IN THE ADF CONTEXT

TIMOTHY CERVENJAK
WHAT CURRENTLY HAPPENS

System Safety
Looks at a systems context and derives the criticality based Assurance Levels required for each box functions.

Software Assurance
Defines some of the practices/policies/processes needed to develop software that meets the Assurance Level.

• The level of confidence that your software meets the requirements.
What is the critical software functionality that caused this level of Software Assurance?

Just because we have sufficient confidence that software meets the requirements, does that indicate the systems is safe?

DO-178C Level A-E is just a quality assurance formula for a level of confidence that the software meets the requirements.
Software Safety allows you to focus resources and engineering effort to the areas that really count.

Software Safety bridges the gap between Systems Safety and Software Assurance.

Software Safety is an analytical approach to identifying critical functionality of software and mitigating the risk of failure to produce robust fault tolerant software systems.
The Role of Software in Recent Aerospace Accidents

• “Almost all software-related accidents can be traced back to flaws in the requirements specification and not to coding errors.”

Dr Nancy Leveson, MIT - Massachusetts Institute of Technology

2001
BAD AND UNSAFE REQUIREMENTS

• Requirements that result in:
  • Flying into terrain without warning,
    • Eg. “Push nose down when x, y and/or z happens”
  • Safety features that operate outside of intended/expected phase of flight

• SW Assurance defines the level of confidence that SW meets Requirements
  • DO-178C Level A: Means you spent lots of $$$ to make sure this happens every time.
  • DO-178C Level E: Means you are sort of confident this will happen most of the time, but sometimes it does strange things.

• Are B737Max design flaws, coding quality or requirements?
MILITARY SYSTEMS GENERALLY DON’T DEMONSTRATE COMPLIANCE WITH DO178

- Unlike commercial aircraft, Military aircraft generally don’t demonstrate compliance with DO-178
  - Not required by certifying Military Airworthiness Authority (US DoD, UK MoD)
  - Military systems safety criticality can be more complex in context
  - Mission focused results, variable operational context
  - Limited budget when considering the amount of functionality
  - Experimental technologies and complex architectures
  - Weapons control software criticality
COMMERCIAL AIRCRAFT

• Unchanged operating model and operating environment
  • Moving people safely from A to B can be done in an old 747 or a brand new 787
    (same result)

• Less complex Avionics Systems

• Software cost spread over a huge number of aircraft

• Less likely to have major avionics changes over its life

• Industry consensus and certification consistency attraction of one regulatory standard
COMMERCIAL AIRCRAFT
COMPLEXITY RELATIVELY LOW

ECMU
Comm’s
AFCS
NAV
Displays

Avionics Controller

Human in the loop

Commercial in Confidence
Nova Systems
MILITARY AIRCRAFT

Significantly more complex avionics systems

Mission critical focus

Experimental technology

Constantly upgrading to maintain Military Dominance

Every Military has its own specific and custom requirements

Nova Systems
MILITARY AIRCRAFT
COMPLEXITY HIGH

ECMU
Comm's
AFCS
NAV
Displays
Avionics Controller
Human in the loop
Mission Controller
Weapons
Surveillance
EWS
Data Links
IFF
MILITARY AIRCRAFT
DO-178 BENCHMARKING

- ECMU
- Comm’s
- AFCS
- NAV
- Displays
- Avionics Controller
- Mission Controller
- Human in the loop
- Weapons
- Surveillance
- EWS
- Data Links
- IFF

DO178C – Level A
DO178C – Level B
DO178C – Level D
Commercial in Confidence
Nova Systems
MILITARY AIRCRAFT - WHAT NORMALLY HAPPENS

Noncompliant - Acceptable

Accepted with Conditions

Refer to Issue Paper

Referred Risk To…… in the loop

ECMU
Comm's
AFCS
NAV
Displays
AFCS
ECMU
Displays
NAV
Weapons
Surveillance
Data Links
IFF
OEM Process A
OEM Process B
ISO 9001
Commercial in Confidence
Nova Systems
US DOD AVIATION PRODUCTS

DO WE HAVE ANY PLATFORMS THAT “FULLY” MEET DO-178?

- C-17
- F/A-18
- JSF
- IAI Heron-1 - At least its battle proven
  - (Israel Airspace Industries)
COMPLEXITY COMPARISON

MILITARY vs COMMERCIAL

• Scope and complexity of the documents has significantly increased
• Interconnectivity of the requirements and data has significantly increased
• Cross over from High integrity to Low integrity systems complicated the Safety Argument
• Management Overheads
  • Shared development over multiple Governments
  • Multiple role variants of the same aircraft
  • Security Consideration e.g. ITARs
COMPLEXITY COMPARISON
MILITARY vs COMMERCIAL

Military UAS aircraft add a new level of complexity

• Civil Regulation are focused on having people in the aircraft
  • Pilot fault diagnosis and recovery removes the need for automation

• Loss of Life or Significant Financial Impact
  • 3rd Party risk exposure
UNMANNED AERIAL SYSTEMS

NEW DESIGN CONSIDERATION

NO HUMAN IN THE LOOP

NEW AUTONOMOUS FUNCTIONALITY

RELIANCE ON SOFTWARE ACROSS THE ENTIRE SYSTEM

PREVIOUSLY NON CRITICAL SYSTEMS ARE NOW MORE CRITICAL

NEED SENSE AND AVOID (SAA) SYSTEMS

RELIANCE ON COM'S AND SATCOM LINKS

GROUND SEGMENT OF THE SYSTEM BECOMES CRITICAL
UNMANNED AERIAL SYSTEMS
SIZE AND OP AREA MATTERS (FOR HARM POTENTIAL)

• DASR UAS – Unmanned Aircraft Systems
  • UAS.20 - CERTIFIED CATEGORY UAS
  • UAS.30 - SPECIFIC CATEGORY UAS
    • (Specific Type A)
    • (Specific Type B)
      • UAS.35 - STANDARD SCENARIOS FOR UAS OPERATIONS

• UAS.35 - STANDARD SCENARIOS FOR UAS OPERATIONS
  • Standard Scenario for Micro UAS: UAS MTOW 0.1 kg MAX
  • Standard Scenario for Very Small UAS: UAS MTOW 2 kg MAX
  • Standard Scenario for Defence Ranges and Exercise Areas: UAS MTOW 150 kg MAX
    • MQ-4C Triton – 14,628Kg, MQ-9 – 2,250Kg, IAI Heron – 1,150Kg
UNMANNED AERIAL SYSTEMS
NEW PROBLEMS AND COMPLEXITY

- ECMU
- AFCS
- NAV
- Comm’s
- SAA
- Avionics Controller
- Mission Controller
- Ground Segment
- Human
- Weapons
- Surveillance
- EWS
- Data Links
- IFF

Commercial in Confidence
Nova Systems
UNMANNED AERIAL SYSTEMS
DO-178 ALLOCATION ??? (UAS UNDER 2KG / 150KG)

Who Cares – Every SW Engineers Dream Job
UNMANNED AERIAL SYSTEMS
DO-178 ALLOCATION ??? (UAS OVER 150KG)

ECMU
AFCS
NAV
Comm’s
SAA

Avionics Controller
Mission Controller
Ground Segment
Human

Weapons
EWS
Data Link
IFF

DO178C – Level A
DO178C – Level B

Commercial in Confidence
Nova Systems
UNMANNED AERIAL SYSTEMS
GOOD LUCK!!!

Can you even get a qualified GPS system??

Would you bet your life on a SATCOM Link

That a lot of highly critical software to Qualify and Maintain

That’s lots of Autonomous functionality

Like to see you try qualify a Win XP Ground station to any DO178C Level

How are we going to manage 100’s of SW engineer’s !!!!

You want how much more $$$ and years for SW???

We need more than 1 SQA!!!

Is that in the Billions for SW Dev?

Commercial in Confidence
Nova Systems
UAS ARCHITECTURE

• UAS designs are more mission critical and less safety critical
  • Assurance still important, but on what bits?
    • Communications Links
    • Ground Stations
    • Weapons targeting and release
UNSUSTAINABLE RELIANCE ON SW ASSURANCE ALONE

• If everything becomes safety critical
  • Will SW Assurance make everything safe?
    • Not an industrial reality
    • Spreads the expertise for high integrity systems too thin
    • Not sustainable economics or schedule

• Systems Safety and Software Safety need more importance!
DO-178 AS THE BENCHMARK

• TAMM era Software Compliance Finding Plans were created to compare OEM assurance practices against DO-178 Assurance Objectives.
  • US DoD / USN accepts aircraft based on approved OEM internal assurance practices.
  • ADF often uses prior recognition and MAA as the argument for FMS.
  • What about US military suppliers for Aus unique products? Make them change to −178C practices?

• ADRM 054
  • Software Assurance allows for: 178C or DEF-STAN 00-55 Issue 2 as Essential design requirements.
  • Software Safety refers to US DOD JSSSEH as guidance – which is fundamentally built around 882 activities and SWILs.
DO-178 AS THE BENCHMARK

- Is DO-178C delta the most insightful way of assessing the risk?
  - Is 882E, SwILs and JSSSEH flawed?
  - What about the safety analysis and derived requirements for software?
  - How could effort be better spent on identifying system risk drivers?
LEVELS OF REGULATIONS
WHAT DOES EVERYONE THINK?

Under Regulation

A Good Team of SME’s will build a great Product

BAD outcome

A Bad Team will not deliver anything or even worse something dangerous

Focus on meeting the regs will take away from the actual engineering.

GOOD outcome

GOOD outcome

BAD outcome

GOOD outcome

Goldilocks Zone
FOCUS ON THE FUNDAMENTALS

• Have they put the engineering effort into the correct areas
• Oversight by competent MAA
• Architecture for lower criticality software
• Development Organisation – relevant tech and domain experience
• Systems Safety Program goodness - connected?
• Software Safety activities?
  • Tracing Top Events to Safety Requirements
  • Bottom Up / Top Down Methods?
• Focus on the SW engineering fundamentals
SOFTWARE ASSURANCE FUNDAMENTALS

• Development Planning
• Requirements Hierarchy
• Justification of Derived Requirements
• Verification of Requirements
• Bi-Directional Traceability of Requirements (To Executable Object Code)
• Requirements Based Testing (On Target Environment)
• Requirements and Code Coverage
• Assessment of SW Development and Testing Tools
• Software Configuration Management
• Software Quality Assurance
SOFTWARE SAFETY TO THE RESCUE

• Software Safety can help you focus the Engineering effort and resources.

• Reducing the amount software assurance overhead can have huge savings over the life of the product.

• But Software Assurance (DO-178C or similar) is essential, just insufficient alone.
SOFTWARE SAFETY PROGRAM

NEW SYSTEMS vs LEGACY SYSTEMS

• Are Legacy system requirements suitable for re-use without software safety analysis?

• New Systems:
  • Gives you a chance to start it the right way.
  • The potential for huge saving across the life of the program.

• Legacy systems:
  • Provides the opportunity to optimise the TLS program.
  • Protection from knowledge loss.
  • Challenges of digitising previous analogue systems with same requirements?
THE V DIAGRAM

HOW IT FITS

SYSTEM SAFETY
SOFTWARE SAFETY
SOFTWARE ASSURANCE

V Diagram Referenced from: JOINT SOFTWARE SYSTEMS SAFETY ENGINEERING HANDBOOK
JSSSEH: Version 1.0 Published August 27, 2010
We need to identify Safety Requirements

Safety Requirements are requirements that are critical to the safe operation of the whole system.

The critical functions of the system are identified by System Hazard Analysis.

Safety Requirements should mitigate the risk of a undesired event / Top Event occurring. ("Shall not")

Functional Hazard Analysis techniques such as Sw Fault Tree can be used to identify Sw Safety Requirements.
UTILISING THE SAFETY REQUIREMENTS

• Identifying the impact Software changes have on the System.
• Given the scope of planned software changes we can systematically identify the safety related changes based on impact to Safety Requirements.
• Changes to Safety Requirements will require extra design considerations and verification activates.
• Regression Testing can be based on your SwS Requirements
UNMANNED AERIAL SYSTEMS
HOW CAN WE BETTER ASSESS UAS’S

• Context: No UAS’s are currently fully certified to any of the major standards.

• Understand any deltas to the Configuration Role Environment (CRE)

• What is the expected risk exposure to 3rd parties?

• Understand the Design Concept and reliance on the Ground Control System
  • Is the UAS designed for Autonomous Operations? E.g. Take-off, Landing…
  • Is the UAS hands on flying and weapons release?
  • Is the GCS Critical for flight or just provides information back to the flight crew?

• DJI Phantoms has an Obstacle Sensing System off the shelf
You just lost control of your UAV
ABSTRACT

• **DO-178C is No Silver Bullet – Software Safety in the ADF Context**
  
  • Many software safety practitioners or overseers in ADF aviation circles over the last couple of decades have seen DO-178 as the silver bullet to ensuring safe aviation software. More often than not, we’re faced with the reality that, while DO-178 compliance was ‘required’, the manufacturer never developed software to that standard. On top of that, predominantly we focus on outright compliance with DO-178 with the assumption that the requirements are correct and of highest quality. Bad or unsafe software/system requirements will not be detected by even the highest levels of Software Assurance.

  • Through years of experience assessing safety and mission critical software, various military projects, UAS platforms where certification baselines don’t exist, teaching aviation software certification, and assessing autonomous systems in other industries, we present a view of software safety based on fundamental principles that make up all software standards. As DASRs mature and unmanned systems become more prevalent, a pragmatic approach to software safety is necessary – ensuring the best safety bang for buck.