PROPRIETARY NOTICE
This document is the property of CS Communication & Systems and cannot be communicated or disclosed without CS Communication & Systems’s authorization. You may not possess, use, copy or disclose this document or any information in it, for any purpose, including without limitation to design, manufacture, or repair parts, or obtain government approval to do so, without CS Communication & Systems’s express written permission. Neither receipt nor possession of this document alone, from any source, constitutes such permission. Possession, use, copying or disclosure by anyone without CS Communication & Systems’s express written permission is not authorized and may result in criminal and/or civil liability.

EXPORT CONTROL NOTICE
✓ This document does not contain any technical data

☐ This document contains technical data which export requires compliance to U.S. Export Administration Act and the Export Administration Regulations (EAR), 15 C.F.R. parts 730-774.
   Its ECCN classification number is

<table>
<thead>
<tr>
<th>Regulation</th>
<th>Classification Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECCN</td>
<td></td>
</tr>
</tbody>
</table>

☐ This document contains technical data which export requires compliance to U.S. International Traffic in Arms Regulations (ITAR), 22 C.F.R. parts 120-130.
   Its classification number is

<table>
<thead>
<tr>
<th>Regulation</th>
<th>Classification Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITAR</td>
<td></td>
</tr>
</tbody>
</table>
AGENDA

» CS Snapshot
» Jumping into DO-178C and its Supplements
» Lessons Learned on past ‘Bad’ Certifications
  ▪ Tips to get a smooth certification
» Q&A
CS GROUP OVERVIEW

- **Business:** Engineering Services & Products for Safety-Critical Embedded Systems and Software.
- **Mission:** Reduce cost and timeline of the SDLC for our clients of the Automotive and Aerospace & Defense

**MAIN CUSTOMERS**

- Valeo
- AIRBUS GROUP
- Triumph Group, Inc.
- ESA
- NISSAN
- MDA
- MINISTÈRE DE L'INTÉRIEUR
- RATP
- D S N A
- VOLVO TRUCKS
- BOSCH
- THALES
- WOODWARD
- Pratt & Whitney
- NATO
- OTAN
- MBDA
- GRAMMATECH
- DENSO
- LeddarTech
- cnes

North America:
- 120 in Montreal & 25 in East Hartford

Worldwide presence

$240M USD in revenues

~2,000 employees

This document is the property of CS Communication & Systems Inc (CSINC) and cannot be communicated or disclosed without CSINC’s authorization. This document does not contain any export regulated technical data.
CS AT A GLANCE

✈ Developing Safety & Critical Software for over 20 years predominantly in

- Aerospace and Defense Industries
- Automotive Industry

- Software Development / Design
- Verification & Validation
- DevOps
- Testing Platforms
- Certification
JUMPING INTO DO-178C
DO-178C IN ITS CONTEXT

A standard to

- Define criteria for development and approval of safety critical S/W to ensure that:
  - S/W failures do not lead to safety issues at A/C level.
  - S/W behaves as system requirements allocated to S/W
  - Have a consensus between users, providers and authorities at international level

Consists in

- Guidelines for S/W development to lead to the certification of an A/C or engine containing software
- Clarification on relationship between system and software life cycle
- Does not define any S/W development process

Certification done in steps: sub-system certification, aircraft certification, entry into service
A LITTLE BIT OF HISTORY

January 1982, First issue DO-178

1982, 767

May 1980, RTCA working group

1985, DO-178A

1989, FAA requires new working group

1992, DO-178B

1993, FAA AC 20-115 considers DO-178B “as a means (...) to secure” FAA approval of the digital

1993, A340

1998, A340-500/600

1995, 777

2008, 7E7

2011, DO-178C
BASIC PRINCIPLES

This norm addresses generic concepts of safety through the following notions:

- **Reliability**: The product executes system required functionalities in specific conditions (data and time) - *Make sure that there are no failures introducing a loss of functionality*.
- **Maintainability**: Aptitude to fix failures on the product with or without changes to the software.
- **Availability**: The product is available and is in good condition.
- **Safety**: The product is able to stay stable even when there is a failure during execution.
- **Security**: Aptitude to protect data and treatments against non authorized activities.
FAILURE CONDITIONS – DESIGN ASSURANCE LEVEL

Defects in the software

System Failure

Failure condition (effect at aircraft level)

Minor
- Routine flight plan changes
- Some physical discomfort to passengers or cabin crew

Major
- Significant increase in crew workload
- Physical distress to passengers or cabin crew (possible injury)

Hazardous
- Large reduction of functional capabilities
- Serious or fatal injury to a relatively small number of occupants other than crew members

No safety effect
- No effect on operational capabilities

No effect on operational capabilities

Loss of the airplane
PLANNING THE DEVELOPMENT LIFECYCLE
THE OBJECTIVES OF THE V-CYCLE

System Requirements

Specification of software requirements

Software architectural design

Software unit design and implementation

Design Phases

Integration testing on hardware

Integration testing on host

Test Phases

HW/SW Integration Testing

Software integration and verification

Software unit verification

System testing

System integration and testing

Unit testing

System Requirements
PLANS HIERARCHY

Plan for Software Aspects of Certification (PSAC)

Software Development Plan (SDP)
Software Verification Plan (SVP)
Software Configuration Management Plan (SCMP)
Software Quality Assurance Plan (SQAP)

Program Management Plan (PMP)
Primary means for communication to certification authorities for agreement

**Plans**

- **System**
  - Overview of the system
  - How system will be transcribed into software
  - Relationship between all softwares (OS, Application SW...)

- **Process**
  - Processes and tools per Plans
  - Artifacts Controls

- **Planification**
  - Software versions in timeframe
  - Software version and goals of certification

- **Quality Assurance and Control**

- **Organisation**
  - How the participants will be organized for all the software activities
  - How the software will be developed
GLOBAL OBJECTIVES FOR DEVELOPMENT

Software Requirements:
Objectives:
• HLRs defined
• Derived HLRs defined and provided to system

Software Design:
Objectives:
• Software architecture and LLR developed from HLRs
• Derived LLRs defined and provided to system

Software Coding:
Objectives:
• Source code is developed from LLRs

Software Integration:
Objectives:
• Executable code and Parameter Data Item (if any) are produced and loaded into the target
DEVELOPMENT ASPECTS - TRACEABILITY

Objectives:
• Enable verification of the complete implementation of higher level requirements
• Give visibility to derived requirements
• Give visibility of architectural decision made during design process
• Enable verification that each source code is documented at design level

Traceability
GLOBAL OBJECTIVES FOR VERIFICATION - REVIEWS

Software Development Reviews

Software Requirements Review:
- Compliance with system requirements
- Accuracy and consistency
- Compatibility with the target computer
- Verifiability
- Conformance to standards
- Traceability
- Algorithm aspects

Software Design Review:
- Compliance with high level requirements
- Accuracy and consistency
- Compatibility with the target computer
- Verifiability
- Conformance to standards
- Traceability
- Algorithm aspects
- Partitioning integrity

Software Coding Review:
- Compliance with low level requirements
- Compliance with software architecture
- Verifiability
- Conformance to standards
- Traceability
- Accuracy and consistency

Software Integration Review:
- The output is complete and correct through examination of compiling, linking, loading data and memory mapping.
GLOBAL OBJECTIVES FOR VERIFICATION - TESTING

SOFTWARE TESTING

OBJECTIVES
The objectives of software testing are to confirm that the executable object code
- Complies with the high and low level requirements
- Is robust with the high and low level requirements
- Is compatible with the target computer

Software Testing

Development lifecycle

Software requirements-based tests

Low Level Tests

Software Integration Tests

HW / SW Integration Tests

Software requirements-based tests

Software requirements-based tests

End of test
DO-178C SUPPLEMENTS INTRODUCTION
WHERE DO THEY COME FROM?

With the DO-178C, 3 supplements have been created:

- Define the conditions under which modern software dev can be used

- DO-330: Tools Qualification
- DO-331: Model-Based Development and Verification
- DO-332: Object Oriented and Related Techniques
- DO-333: Formal Methods
DO-330: THE USAGE OF TOOLS

➢ In the past everything was done manually
➢ Automated tools are (potentially) more reliable than humans
➢ More and more activities are done using tools:
  ▪ Source code generation from design models
  ▪ Test cases generators
  ▪ Test procedures generators
  ▪ Review by dissimilar coding
  ▪ Automated analysis (timing, stack, memory)

➢ For these reasons the DO-330 has been created to prove the suitability of those tools and validate the credit taken from them through qualification.
DO-330: WHAT IS TOOLS QUALIFICATION

What is Tool Qualification?

- Process to gain certification credit for automated use of a tool replacing an identified objective of DO-178C

When do I need qualification?

- When processes of DO-178C are covered or partially covered by the use of a software tool

What regulation brings into tool qualification through DO-330?

- Define guidance to develop and validate a software tool

What tool can be qualified?

- Qualification can be done for COTS tools or custom made tools
The aim of Model Based Development

- Use precise and appropriate languages to unambiguously record requirements

Two types of model:

- Specification models for high level requirements
- Design models for low level requirements
DO-331: Model Based Verification

- DO-331 enables testing in a model environment
  - Testing in context where the model is used in a simulation environment
  - Using a simulation environment comes with the burden of:
    - Performing Model Coverage Analysis
    - Qualify the simulation environment as being same as target environment for credit purpose

- When design model tools are used they come with a testing suite to enable the testing at the simulation level:
  - QTE for SCADE
  - SLDV for Simulink

- DO-178C objectives are applicable and completed with Specific objectives in the DO-331
CERTIFICATION STEPS
SOFTWARE CERTIFICATION

- Define & defend the development and testing strategy
- Full certification documentation generation
- Review with designated authorities (FAA, Transport Canada) and delegates (DER, DAR, DAA)
- Support of all Stage of Involvement (SOI) audits

Identifying software and process gaps to meet applicable norms

Suggesting/defining the most efficient strategies in respect to norms for software added value

Preparing necessary documentation to complete certification

Presenting the work and the strategies in front of certification authorities

Training people to enhance norms application and auditing the activities performed

EASA (European Aviation Safety Agency)

Transport Canada

Canada
STAGES OF INVOLVEMENT

- DO-178C (any DAL) requires audits and presentations to certification authorities among the life cycle of the project:
  - SOI1 → Plans
  - SOI2 → Development lifecycle
  - SOI3 → Verification lifecycle
  - SOI4 → Certification (SAS and CBK)

- Objectives of the SOIs:
  - Present the processes followed during the lifecycle of the project
  - Provide confidence to the authorities
  - Present live audits of the work products
  - Present project progress and quality
LESSONS LEARNED ON PAST ‘BAD’ CERTIFICATIONS
TIPS TO GET A SMOOTH CERTIFICATION
LESSONS LEARNED ON PAST ‘BAD’ CERTIFICATIONS

➤ No Certification Planning: Tools & Methods
  - “We have only the code and it is working”
➤ Software activities too compartimented
  - Software architecture DO NOT consider the testing
➤ Cycle of Software Deliveries too long
➤ Formal Testing starts too late
  - We must find errors early in the process
➤ In-house or outsourcing?

Popular myth is that DO-178C is expensive
It is true if you do not take steps ahead
In-house / Outsourcing

- IP Protection? Does SW part of the Value of the OEM?
- Some suppliers are equipped to preserve your IP
- Off-shore?
- Careful to the Export Control
TIPS TO GET A SMOOTH CERTIFICATION

No Certification Planning: Tools & Methods

- Gap Analysis: wherever you are (concept, prototyping, pre-cert): identified your holes
  - For a few k$ you may get this analysis for all the processes
- Identify the Software variability: SW Version Deltas, line of products, baselines
  - SW Architecture and Testing strategy but also tools may differ from these inputs
- Do Benchmark with tools
  - Involve Tool vendors and / or suppliers
TIPS TO GET A SMOOTH CERTIFICATION

Cycle of Software Deliveries too long

- CI/CD: Continuous Integration / Continuous Delivery
  - SW Deliveries every 2 weeks
  - Measure your performance and then improve
  - Improve cost estimates
- Agile: It is not against the rigor of DO-178C
- Some tools exist
**TIPS TO GET A SMOOTH CERTIFICATION**

- Software activities too compartmented & Formal Testing starts too late
  - Prototyping: do a round trip SW Development, Design and Testing
  - Software architecture: take into consideration how you will test in your design choice
    - Up to 30% gains in testing
  - Test at the higher level possible = Real bugs
  - Testing: There are techniques where activities may be combined to reduce costs
    - Test Credits
  - Automated Tools may help to reduce the burden
    - MACHINE LEARNING: Software tools to test embedded software