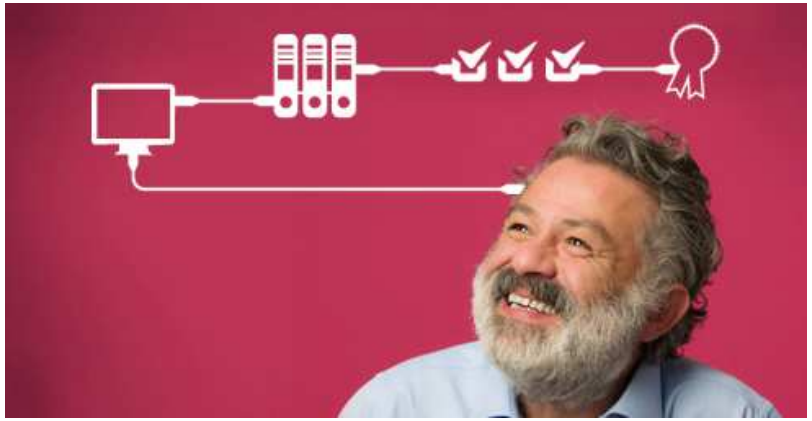
ERASM

EXPERTISE EN MAÎTRISE DES RISQUES

christophe.barnier@erasm.fr
julien.munerot@erasm.fr
Tel: 04 84 47 00 03



ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

Plan Présentation DO-178C

- Les origines
- DO-178C
- Bonnes pratiques – Retour d'Expérience

ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

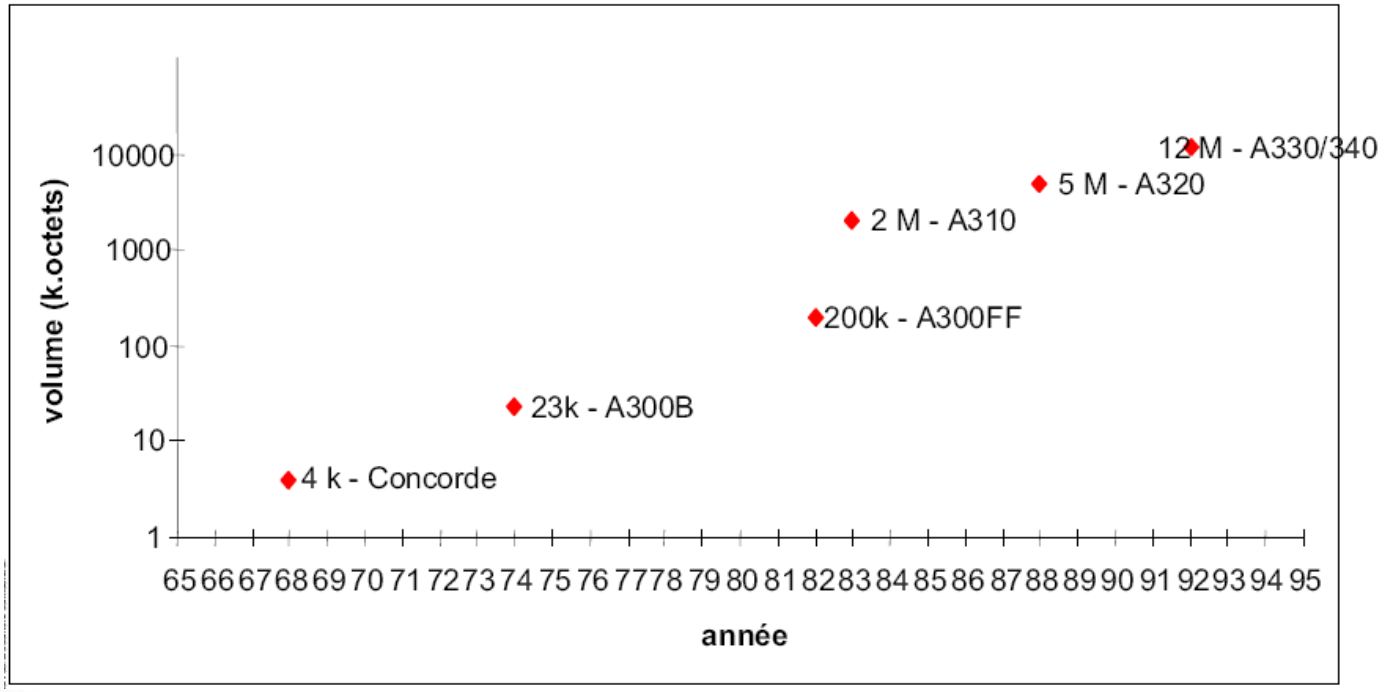
DO-178C

LES ORIGINES

ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

Besoin de réglementation!

- **Garantir la sécurité** malgré:
  - L'utilisation croissante des logiciels dans les fonctions critiques
  - L'augmentation de la complexité des fonctions critiques



DO-178?

- définition usuelle:
  - « **Norme**
  - qui régit le **développement** de logiciel
  - Dans le **domaine aéronautique** »

ABUS DE LANGAGE

- extrait DO-178B/C section 1 :
  - « **fournir des recommandations**
  - pour la **réalisation** de logiciels
  - destinés aux **systèmes et équipements de bord** »

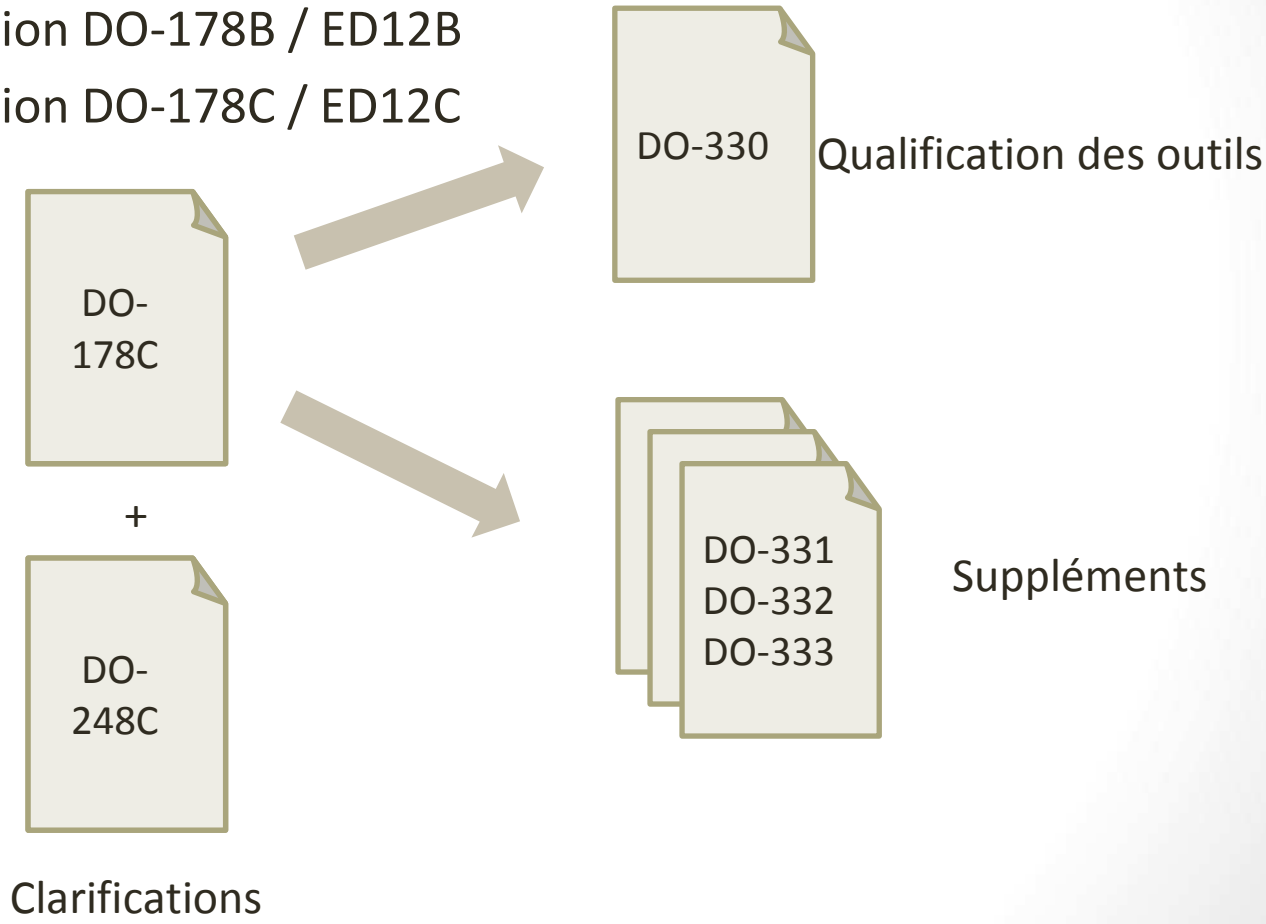
- Consensus de la communauté aéronautique
  - Elaboré en commun par RCTA (USA) et EUROCAE (Europe)
  - RCTA: Requirements and Technical Concepts for Aviation
  - EUROCAE: European Organisation for Civil Aviation Equipment

RTCA DO-178 ⇔ EURACAE ED-12

Appellations ⇔ 1 même document

Historique

- 1982: Publication DO-178 / ED12
- 1985: Publication DO-178A / ED12A
- 1993: Publication DO-178B / ED12B
- 2011: Publication DO-178C / ED12C



ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

DO-178C

LE DOCUMENT!

ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

Approche méthodologique

- Approche par processus
- Pas de cycle de vie imposé!
- Des objectifs à atteindre
- En fonction de Niveaux Logiciels

Table A-6 Testing of Outputs of Integration Process

	Objective		Activity	Applicability by Software Level				Output	
	Description	Ref	Ref	A	B	C	D	Data Item	Ref
1	Executable Object Code complies with high-level requirements.	6.4.a	6.4.2 6.4.2.1 6.4.3 6.5	O	O	O	O	Software Verification Cases and Procedures Software Verification Results Trace Data	11.13 11.14 11.21
2	Executable Object Code is robust with high-level requirements.	6.4.b	6.4.2 6.4.2.2 6.4.3 6.5	O	O	O	O	Software Verification Cases and Procedures Software Verification Results Trace Data	11.13 11.14 11.21

Les Processus

- **La planification du logiciel**:
  - définir et coordonner les activités de réalisation du logiciel
- **Le développement du logiciel**
  - fournir le produit logiciel et sa **documentation**
- **Les processus intégraux** (support):
  - **La vérification logiciel**
  - **La gestion de configuration logiciel**
  - **L'assurance qualité logiciel**
  - **La coordination pour la certification**

Les processus intégraux sont mis en œuvre en même temps que le développement du logiciel tout au long du cycle de vie du logiciel.

Niveau logiciel et objectifs DO-178C

Conditions de Pannes	Niveau Logiciel	Objectifs à atteindre DO-178C	Dont objectifs avec indépendance
Catastrophique	A	71	30
Dangereuse	B	69	18
Majeure	C	62	5
Mineure	D	26	2

- Pour les niveaux logiciel A/B/C, les objectifs de planification à réaliser sont les mêmes!
- Pour les niveaux logiciel A/B/C, les objectifs de développement à réaliser sont les mêmes!
- Pour les niveaux logiciel A/B/C/D, les objectifs de gestion de configuration à réaliser sont les mêmes!
- Pour les niveaux logiciel A/B/C, les objectifs d'assurance qualité logiciel à réaliser sont les mêmes!
- Pour les niveaux logiciel A/B/C/D, les objectifs de coordination avec la certification à réaliser sont les mêmes!

Le Processus de Vérification

- **La vérification est le processus qui a:**
  - **le plus grand nombre d'objectifs à remplir**
  - **La plus forte demande d'indépendance**

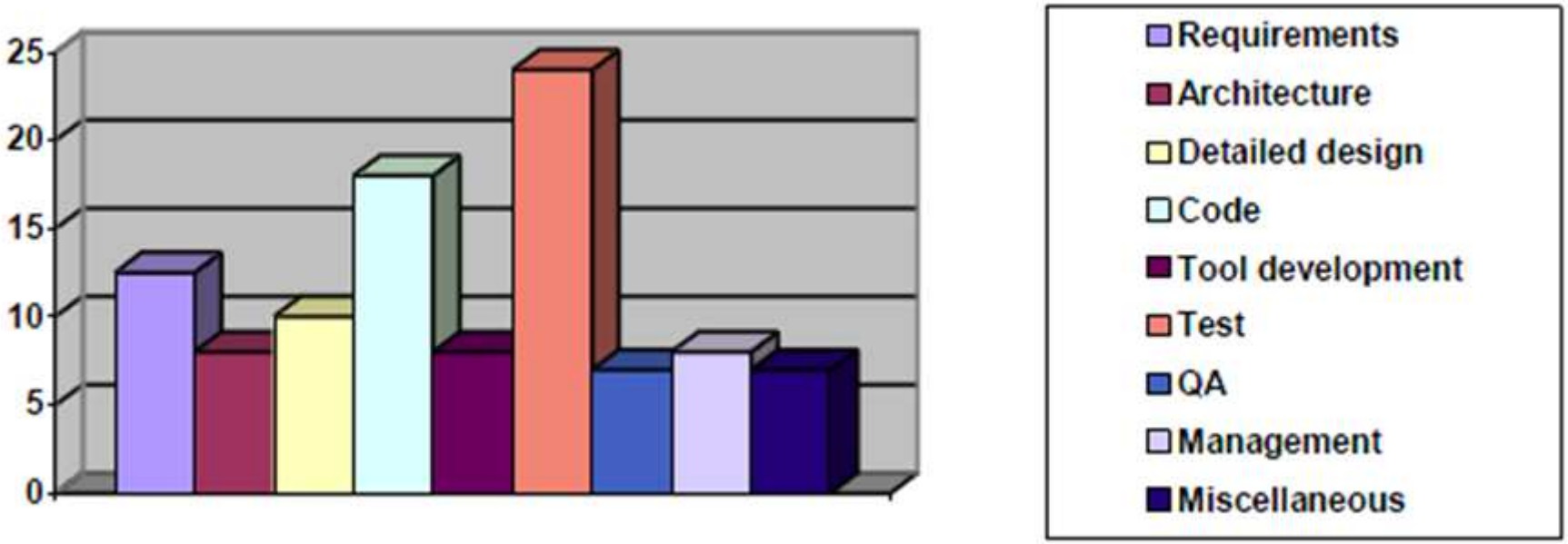
La vérification ne réduit pas aux tests!

Le Processus de Vérification

- 1/ **Vérifier** l'intégralité des données du **développement** logiciel:
  - Documentation (spécification, architecture, conception détaillée)
  - Codage (conformité aux règles, à l'architecture…)
  - Génération du code (directive de compilation, fichier mapping…)
- 2/ **Tester** le code en regard de ses exigences
  - De conception détaillée (exigence de bas niveau – LLR)
  - De spécification (exigence de haut niveau - HLR)
- 3/ **Vérifier** l'intégralité des données de **vérification** du logiciel
  - Complétude et exactitude des spécification de tests
  - Complétude et exactitude des résultats de tests
  - Assurance du taux de couverture du code

Illustration - Répartition des activités

- Le codage ne représente qu'une part minime des activités
- (source Reqtify)



ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

Bonnes pratiques / Retour d'expérience

DO-178C

Les clés vers le succès

Ces clés ne sont pas (directement) lié au code source…

- Les facteurs essentiels de succès:
  - **Vous!**
  - **La vérification!**
  - **La Gestion des exigences!**

Bonnes pratiques – Exigences

70% des « bugs » sont liés aux exigences

- lors de la spécification d'exigences
  - exigences 'obscures' / incomplètes / contradictoires /non vérifiables
- lors de la traçabilité des exigences
  - exigences incorrectement tracées / partiellement couvertes

- Exigences de Spécification (HLR)
  - On spécifie généralement le comportement du logiciel dans le cas nominal, mais on oublie fréquemment de spécifier le comportement en cas de défaillance
  - 1 exigence HLR => 2 ou 3 exigences LLR (source Reqtify)

- Exigences de Conception détaillée (LLR)
  - 1 exigence LLR => 20 ou 30 lignes de code (source Reqtify)
  - Le pseudo –code ne peut être considéré comme une exigence LLR (⇔ ne pas faire les exigences LLR une fois le code développé!)

ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

Bonnes pratiques – Métriques Logiciel

- Analyse statique du code: Métrique de « Taille »
  - Pas de fonction supérieure à 80 lignes de code
- Analyse statique du code: Métrique de « Complexité » (appelée cyclomatique ou V(g))
  - Pas de fonction avec V(g) > 10 (source IBM RTRT)
- Mesure du taux de remplissage de la pile
- Mesure de la charge CPU
- Mesure des WCET (Worst Case Execution Time)

Bonnes pratiques - Outils

- **Choisir et savoir utiliser** les outils adaptés à votre besoin
  - Chaine de développements et de debug
  - Outil de traçabilité
  - Outils d'analyse statique
  - Outils d'analyse dynamique
  - Outils de gestion de configuration
-  Un outil doit être qualifié
-  Un outil n'est qu'un outil!

ERASM
EXPERTISE EN MAÎTRISE DES RISQUES

Conclusion

- La conformité à la DO-178C peut être une remise en question de votre manière de développer du logiciel.
- Elle nécessite une réelle implication des équipes.
- Elle augmentera sensiblement le volume de vos activités de vérification
- La DO-178C contient certaines finesses sémantiques dont il ne faut pas faire l'amalgame auprès des autorités de certification.

MERCI pour votre attention!

Julien MUNEROT

julien.munerot@erasm.fr

Tél: 04 42 12 34 46

http://www.erasm.fr/

Christophe BARNIER

christophe.barnier@erasm.fr

Tél: 04 84 47 00 03