

RAMS Software Techniques in European Space Projects

An Industrial View

J.M. Carranza

COMPASS Workshop - York, 29/03/09

Contents

- Context and organisation of ESA projects
- Evolution of RAMS Techniques in European space projects
- Techniques used in projects:
 - Analysis
 - Verification
 - Other
- Context for RAMS implementation
- Conclusions
- Questions

ESA Project Context

ESA Mission

- One of ESA's tasks is to promote, coordinate and monitor space projects
- Projects are carried out by European industry
- Funded by ESA members

Space projects

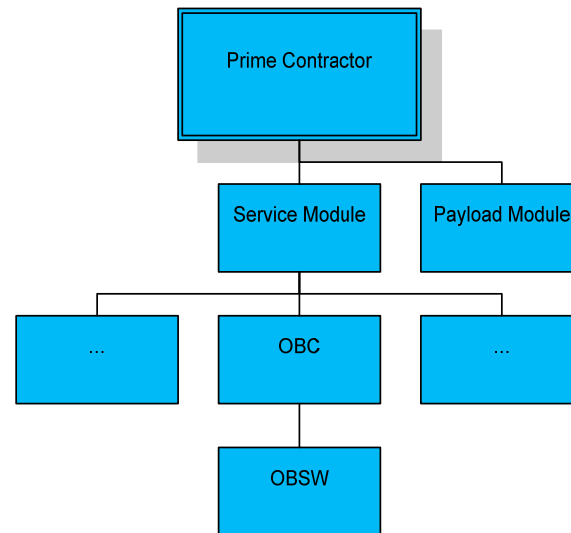
- Expensive
- Long life cycle (both development and operations)
- Complex organisation
- Technologically challenging



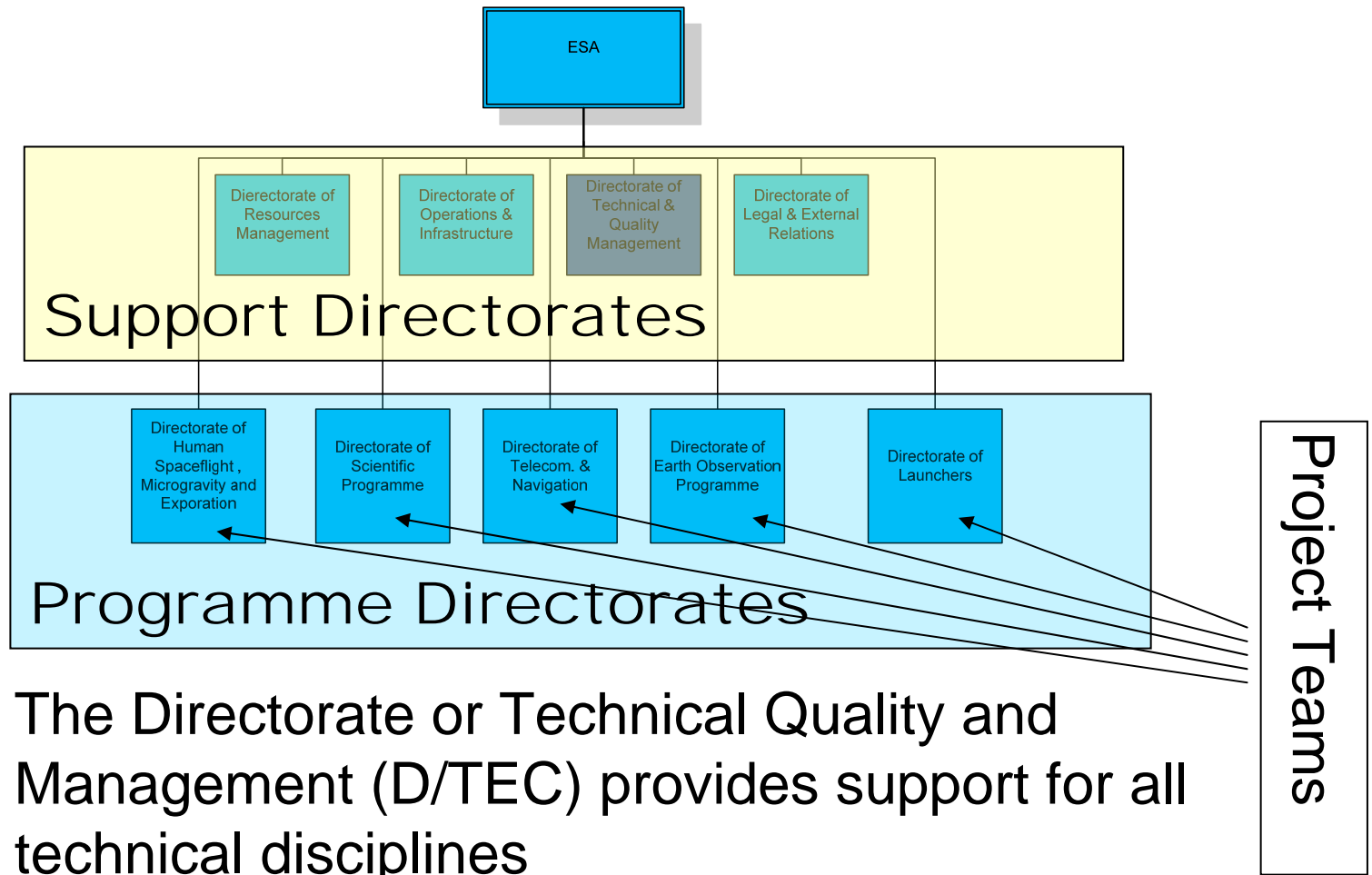
Need to ensure reliability, availability, maintainability and safety

Project Organisation

- Each large project (large = space mission) is carried out by an industrial consortium
- The consortium is led by a Prime Contractor and organised in a hierarchical structure
- ESA sets up a project team:
 - Similar to industry structure for key roles
 - Support from D/TEC for remaining disciplines



ESA organisation



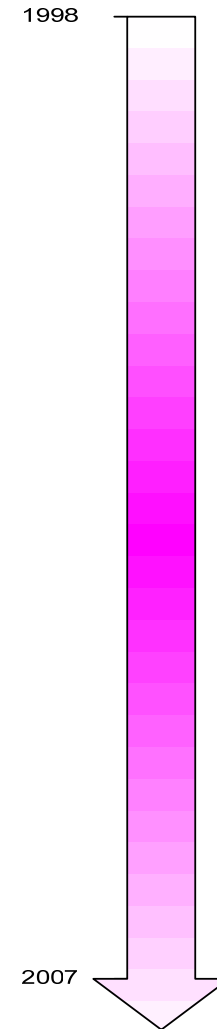
SW Product Assurance

- SW Product Assurance in charge of monitoring Reliability, Availability, Maintainability and Safety (RAMS) related activities
- SW Product Assurance almost always external to the project (D/TEC)
- Part time support (e.g. 20%) / large number of projects to monitor / not only RAMS
- Author: 9 years in SW PA (1998 – 2007)

Evolution of RAMS techniques in European space projects

RAMS Evolution (personal view)

- In many cases no RAMS analyses in projects
- ESA initiated R&D study to identify possible techniques and tools \Rightarrow WO12 study (concluded 2000)
- A number of methods started to be used (WO12 influence?)
- Nowadays a number of methods are used but no uniform application



Software RAMS Requirements Verification Techniques

- (PASCON) WO-12
- Completed in 2000
- Led by Astrium SAS (France)
- Provides a catalog and significant information on available techniques
- Detailed analysis of RAMS related:
 - Software requirements and design constraints
 - Static methods, techniques and procedures
 - Dynamic methods, techniques and procedures
 - Practical examples of application of selected techniques (including verification)
 - Etc.

Analysis Techniques

WO12: Static Software RAMS Analyses

Static Software RAMS Analysis	Software Development Phases									Criticality Classes			
	UR	SR	AD	DD	CD	UT	IT	ST	AT	A	B	C	D
Event Tree Analysis	Y	Y	Y	Y	Y	Y	Y			R	R	R	R
Hazard and Operability Studies (HAZOP)	Y	Y	Y	Y	Y	Y	Y	Y		HR	HR	R	R
Nuclear Safety Cross Check Analysis (NSCCA)	Y	Y	Y	Y	Y	Y	Y	Y		R	R		
Software Common Cause/ Mode Failure Analysis (SCCFA/ SCMFA)			Y	Y	Y	Y	Y	Y		HR	HR	R	
Software Error Effect Analysis (SEEA)			Y							HR	HR	R	R
Software Failure Modes, Effect and Criticality Analysis (SFMECA)	Y	Y	Y	Y	Y					HR	HR	R	R
Software Fault Tree Analysis (SFTA)	Y	Y	Y	Y	Y					HR	HR	R	R
HW/ SW Interaction Analysis (HSIA)	Y	Y	Y	Y						HR	R		
B Method	Y	Y	Y	Y	Y					HR	R	R	
RAISE (Rigorous Approach to Industrial software Engineering)	Y	Y	Y	Y	Y	Y				HR	R	R	
VDM (Vienna Development Method)	Y	Y	Y	Y	Y					HR	R	R	
Z Notation	Y	Y	Y	Y						HR	R	R	
CCS (Calculus of Communicating Systems)			Y	Y	Y					HR	R	R	
CSP (Communicating Sequential Processes)			Y	Y	Y					HR	R	R	
Finite State Machine			Y	Y						HR	HR	HR	R
LOTOS (Language Of Temporal Ordering Specification)		Y	Y	Y	Y					HR	R	R	
Petri Nets		Y	Y	Y	Y					HR	HR	HR	R
SDL (Specification Description Language)	Y	Y	Y	Y	Y	Y	Y	Y		HR	R	R	
Schedulability Analysis			Y	Y	Y	Y	Y			HR	HR	R	R
Computational Accuracy		Y	Y	Y						HR	HR	R	R
Markov Models			Y	Y	Y	Y	Y	Y		HR	R	R	R
Code Analysis					Y					HR	HR	R	R
Inspections	Y	Y	Y	Y	Y	Y	Y	Y	Y	HR	R	R	
Walkthrough			Y	Y	Y	Y				HR	HR	HR	HR
Software Sneak Analysis (SSA)	Y	Y	Y	Y	Y	Y	Y			R	R		

WO12: Dynamic Software RAMS Techniques

Dynamic Software RAMS Analysis	Software Development Phases									Criticality Classes			
	UR	SR	AD	DD	CD	UT	IT	ST	AT	A	B	C	D
Test Data Selection/ Boundary Value Analysis						Y	Y	Y		HR	HR	HR	R
Test Data Selection/ Equivalence Class Partitioning						Y	Y	Y		HR	HR	HR	R
White Box Testing/ Basis Path Coverage						Y				HR	HR	R	R
White Box Testing/ Cause- Effect Graphing Technique						Y				HR	HR	R	R
White Box Testing/ Data Flow Coverage						Y	Y			HR	HR	R	R
White Box Testing/ Fault Injection						Y	Y	Y		R	R	R	
White Box Testing/ Linear Code Sequence And Jump Coverage						Y				HR	HR	R	R
White Box Testing/ Loop Testing						Y				HR	HR	R	R
White Box Testing/ Multiple Condition Coverage						Y				HR	HR	R	R
White Box Testing/ Statement- Branch Coverage						Y	Y	Y		HR	HR	R	R
White Box Testing/ Path Coverage						Y				HR	HR	R	R
Black Box Testing/ Back to Back Testing						Y	Y	Y		HR	R		
Black Box Testing/ Interface Testing						Y	Y	Y		HR	HR	R	R
Black Box Testing/ Montecarlo Simulations								Y		R	R	R	R
Black Box Testing/ Simulation								Y		HR	HR	R	R
Black Box Testing/ Statistical Testing for Improving Reliability								Y		HR	R	R	
Black Box Testing/ Statistical Testing for Reliability Prediction								Y		HR	R	R	
Black Box Testing/ Stress Testing								Y		HR	HR	R	R
Black Box Testing/ Symbolic Execution						Y		Y					
Test Analysis/ Fault Seeding						Y	Y	Y		R	R	R	
Test Analysis/ Mutation Analysis						Y	Y	Y		R	R	R	
Test Analysis/ Sensitivity Analysis					Y	Y	Y	Y		R	R		
Test Analysis/ Software Reliability Estimation Models					Y	Y	Y	Y	Y	HR	HR	R	R
Test Analysis/ Test Coverage Analysis						Y	Y	Y		HR	HR	R	R
Test Analysis/ Test Results Analysis						Y	Y	Y		HR	HR	R	R
Test Analysis/ Test Witnessing						Y	Y	Y		R			
Run Time Anomalies Detection						Y	Y	Y		HR	HR	HR	HR
Regression Analysis and Testing								Y	Y	HR	HR	HR	HR

Software RAMS in ESA Projects

SW RAMS Techniques in ESA Projects

- Not standardised
- Highly dependent on which company is responsible (different cultures, varying degrees of sophistication)
- Responsibility can happen at different levels in the consortium
- Prime Contractor and ESA Project Team can have an influence (require or prohibit)

RAMS Analyses

Generic Software Criticality Analysis

- The most common: ‘criticality analysis’
- Technical Note (free text)
- Ranging:
 - From very informal: based on experience or earlier analyses
 - To detailed analysis of each failure mode identified in a separate SFMECA ⇒ direct input to FDIR definition
- Main objective: to determine software criticality class
- Criticality class influences significantly the requirements on the development process ⇒ economic implications

Hardware – Software Interaction Analysis

- Based on checklists \Rightarrow simple to apply
- Sometimes: what to do with outputs?
- Sometimes produced without the support of a SFME(C)A

Software Failure Modes, Effects (and Criticality) Analysis

- Sometimes performed as a standalone analysis (not derived from System level FME(C)A)
- Sometimes performed on the basis of Feared Events List
- Its implementation (ESA projects), provides generally more reliable criticality classification than the other methods

Schedulability Analysis

- Generally required (for on-board software)
- Reasonably well established for typical context: ERC32, LEON, RTEMS, C, etc.

Software Fault Tree Analysis

- Less frequent than the others
- Normally based on the system level Fault Tree Analysis
- Only qualitative: how to estimate probability of failure for software?
- No reliable Software Reliability Model

Verification Techniques

Test Coverage Analysis

- The most common software RAMS verification technique in ESA projects
- Typically statement or branch coverage target
- Often a customer or Prime Contractor requirement:
 - Levels according to criticality classification
 - Subject of significant negotiation

Code inspections

- Generally used to check compliance with coding standards:
 - Supports mainly (but not only) Maintainability
- Some exceptional examples of companies extremely skilled and proficient (even replacing unit testing)

Independent Software Verification and Validation

- Generally required for on-board (critical) software
- In a scenario of a limited number of companies in the space software market:
 - How to ensure technical independence?
 - How to ensure financial independence?
 - How to ensure managerial independence?
 - Respecting Intellectual Rights
- Definition of scope
- Synchronisation of schedules (often critical)

ESA ISVV Guide

- To define a uniform, cost effective and reproducible ISVV process
- Common understanding of the process
- ISVV Process Management
- Activities definition. This includes some of the activities in this presentation

Other techniques

Formal methods

- Occasionally
- B or VDM

Context for RAMS implementation

Implementers

- Vary significantly from project to project:
 - On-board software engineer
 - Software PA engineer
 - System RAMS engineer
- Hardly ever Software RAMS Engineer is identified as a separate role
- Results depend strongly on the personal experience of the implementer

Issues - 1

- Software RAMS techniques are, sometimes, perceived by Management as not cost effective:

“We did not do it in the previous project and we did not have any problems”

- Sometimes performed only because it is required: purpose and benefit are not clearly perceived
- Most methods used depend significantly on the experience and skills of the implementer

Issues - 2

- Software people, often, are not sufficiently educated in RAMS analysis methods
- System RAMS people, usually, do not understand the specific characteristics of software (e.g. estimation of probability of failure)
- Limited interaction between software and system level analyses (little feedback)

Conclusions - 1

- RAMS techniques used in space projects lag behind state of the art (conservative approach)
- There is significant variation in the type, manner and level in which RAMS techniques are applied
- An important topic pending solution is the estimation of the probability of failure of software systems

Conclusions - 2

- ESA Software PA is keen on promoting:
 - New useful techniques
 - A more consistent application of existing techniques
- However they are bound by different constraints: organisational, economical, even political or geographical
- Finally, despite the situation... the number of major or catastrophic failures is relatively small

Challenges for new techniques - 1

- To prove usefulness of new techniques, evidence is needed of:
 - Technical benefits
 - Cost effectiveness
 - Accessibility (tool availability, training, etc.)
- How to promote them:
 - Identify end users: Who? Where? Background?
 - End users are specific individuals in potentially many different companies
 - End users have different backgrounds: software engineering, system RAMS

Challenges for new techniques - 2

- Cycle time for space projects is long
- Project Managers and Project Teams perceive changes to established methods and techniques as risks

Questions?