Requirements and Dependability Obstacles That Must Be Overcome

Carlos H. N. Lahoz
Instituto de Aeronautica e Espaço (IAE)
São Jose dos Campos - Brasil

TOPICS

- Introduction
- Requirements and Dependability
- Obstacles That Must Be Overcome
  - Technology
  - Process
  - People
- Conclusions

Introduction

Introduction

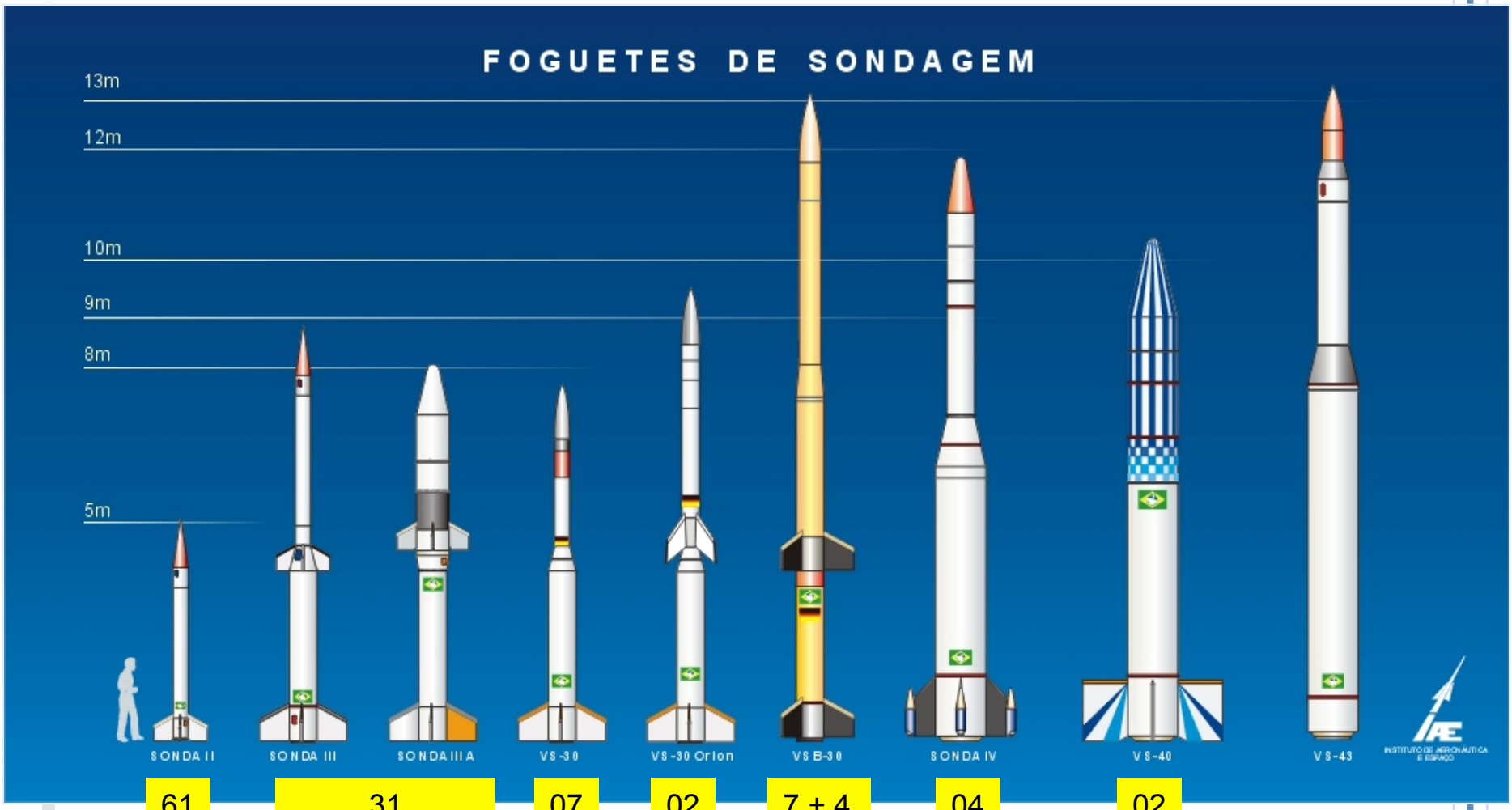
Presentation purpose:

To stimulate the discussion among professionals and others interested in SE about what are the main obstacles that must be overcome in the RE activities in critical computer systems.

Introduction

- IAE sounding rockets

number of launching



Introduction

- VLS - Satellite Launcher Vehicle:

7 rocket-motors (4 stages),
solid propellant, 19 m lenght,
50 t weight.

On Board Software
navigation, control,
sequence of events, telemetry.

Prototypes built:

- 1997
- 1999
- 2003 (accident in launch pad)
- 2011 (?)

Requirements and Dependability

Requirements and Dependability

Main Project Problems (1)

The study of requirements-originated software failures, conducted from 1998 to 2000, reported that half resulted from:

- poor description of the requirements, ambiguity, mistakes and lack of clarity;
- omission of the complete requirements.

HECHT and BUETTNER, Software testing in space programs. Crosslink, v. 6, n. 3 (Fall2005). http://www.aero.org/publications/crosslink/fall2005/ 06.html

Requirements and Dependability

Main Project Problems (2)

Particularly....

- Undefined system requirements until PDR (and consequently undefined software requirements);
- Technological autonomy: (medium to low);
- Low goverment investiments;
- Old technical staff: senior engineers and technics could be retired soon....

Requirements and Dependability

Finally....

software may be highly reliable and correct and still be unsafe when:

- The software correctly implements the requirements but the specified behaviour is unsafe from system perspective;
- The software requirements do not specify some particularly behaviour required for system safety (incomplete);
- The software has unintended (and unsafe) behaviour beyond what is specified in the requirements.

LEVESON, Engineering a Safer World, MIT, draft of online book http://sunnyday.mit.edu/book2.pdf, *2009.*

Obstacles That Must Be Overcome

Obstacles That Must Be Overcome

- TECHNOLOGY:
  - Inadequate requirements engineering techniques
  - Inadequate model representation of the system components and their interactions
  - Lack of safety approach
- PROCESS:
  - Lack or poor RE process
  - Partial requirements monitoring
  - Inadequate communication
- PEOPLE:
  - Different system and software cultural approach
  - Limited understanding of the problem domain
  - Inefficient management

Obstacle: TECHNOLOGY

- Inadequate RE techniques

Lack or inadequate use of SE tools and RE methods for requirements identification, representation and consistency frequently cause project mistakes and misunderstandings.

Overcoming obstacle:

- To training the software team in CASE tools, model checking and other support resources for F and NF requirements system representation and validation;

- To adopt specific techniques for the elicitation of the (software) non-functional requirements (ex: RAMS analysis using HAZOP, SFTA, SFMEA);

- Subcontractors should use appropriate and satisfactory RE techniques (design workshops, prototyping, functional and hazard analysis, besides interviews, questionnaires, observation, and the study of documents).

Obstacle: TECHNOLOGY

- Inadequate model representation of the system components and their interactions

An inappropriate model representation may not identify the behavioral interdependence between their components and makes difficult for the designers to consider all the potential system states and disturbances.

Overcoming obstacle:

- Design the system in terms of components (goals, soft-goals, tasks and resources) and perform a hazard analysis (based on guidewords) applied to those components;

- Use of the same language between heterogeneous teams (system engineers and software engineers) in order to create a common and understandable knowledge about the critical components and their system behavior (hw and sw).

Obstacle: TECHNOLOGY

- Lack of safety approach

The traditional approach to prevent accidents that dealt with electromechanical components (using redundancies, for instance) may not be useful when digital and software systems are used (the fail could come from loss or incorrect information).

Overcoming obstacle:

- The requirements specification must describe what the system do and the software should not do. Elaborate a criteria checklist (needs and constraints);

- Minimize the misunderstandings regarding the software interface with the rest of the system (I/O sw-hw communication, sw conversion variables, etc);

- Describe the software behavior under off-nominal conditions.

Obstacle: PROCESS

- Lack or poor RE process

For external project reasons (such as the non-availability of qualified personnel, inefficient organizational structure, time and resource constraints), RE activities are neglected in favor of others (such as the ones directly related to code development).

Overcoming obstacle:

- It is necessary to formalize the RE activities in a set of simple steps, including the creation of software system models followed by the application of a safety engineering technique;

- Formal reviews and V&V activities during requirement and analysis process must be defined.

Obstacle: PROCESS

- Partial requirements monitoring

An incomplete set of tests and inadequate reviews of the requirements changes are also caused by process monitoring problems.

Overcoming obstacle:

- It is important to consider V&V as high priority activities during the system development and devote special attention to the resources allocated to testing;

- The software complexity should be overestimated and the tests effectiveness should be underestimated.

Obstacle: PROCESS

- Inadequate communication

Without effective communication, it can be impossible for the stakeholders to evaluate in a consistent way all the requirements and to keep them stable during project life cycle.

Overcoming obstacle:

- Any type of communication that might affect the requirements and that have some kind of impact on the project should not be restricted to an email or telephone call;

- A formal requirement configuration management must be established, updated and widely informed.

Obstacle: PEOPLE

- Different system and software cultural approach

According to systems engineers the participation of the software team in the system requirements activities is not viewed as important as it should be.

Overcoming obstacle:

- The software team could take part in the decision making process of the project rationales, when the solutions to be adopted are still being defined;

- Talks about different requirements elicitation techniques (system and software techniques) should be stimulated through discussion forums and workshops when both engineering areas can present their approach and views of the system;

- It is necessary to link business goals to technical requirements via user needs and user requirements.

Obstacle: PEOPLE

- Limited understanding of the problem domain

The results of the RE practices are not shared by system and software engineering areas causing a partial comprehension of the project dependability issues.

Overcoming obstacle:

- Use a cross-functional team to elicit user needs - the team could use the knowledge and views of members with varied backgrounds;

- It is mandatory for those who will perform the role of elicit and validating the dependability requirements to be able to identify the explicit and implicit requirements of the project.

Obstacle: PEOPLE

- Inefficient management

Even performing RA, HA or FMECA in the beginning of the project, not all the detected problems are solved (or mitigated) even at the end of the qualification and acceptance reviews;

The project leader does not have his authority recognized by some project members, in such a way that important problems could be proprietarily solved.

Overcoming obstacle:

- The project manager must balance interests, determine priorities and foster consensus among stakeholders;

- The dependability must be clearly recognized as a value inside the organization and the project, and integrated into all engineering activities;

- Obtain top management commitment to dependability issues.

Conclusions

Conclusions

- The discipline of RAMS (Reliability, Availability, Maintainability and Safety) must be dealt with the RE and performed in an integrated way by the system and software teams as well as the project manager;

- We must reach a balance between the system engineering aversion to risk and the software engineering tendency to take fashionable software approaches.

Conclusions

Finally…

…mature practices in software critical projects depend on overcoming technological and process obstacles, focusing mainly on a change in people's view, implementing a new dependability culture, involving the system and software engineers as well as the organization higher management levels.

Thank you for your attention

lahoz@iae.cta.br
Instituto de Aeronáutica e Espaço (IAE)
55 12 39474901

This presentation was sponsored by:

Brazilian Space Agency - AEB
(www.aeb.gov.br)

Space Science, Applications and Technology Foundation - FUNCATE
(www.funcate.org.br)