

Safe and reliable software: Field experience

**Boulangier Jean-Louis
UTC/HEUDIASYC**

Norms and ...

- **Railway :**
 - CENELEC EN 50126, EN50129 and EN50128
- **Aircraft :**
 - DO 254, DO 178 and DO 278
 - ARP 47.54, ...
- **System E/E/EP:**
 - IEC 61508
- **Automotive:**
 - To be defined

Railway (1)

- **EN50126 and EN50129**
 - Introduce the reliability of the railway system

- **EN50128:**
 - Is based on « hardware reliability »;
 - But system safety analysis can introduced some software reliability objectives;

Railway (2)

➤ Analysis for safety and reliability

- Reliability block diagram, Markov Model, FTA must be use for software (just recommanded);
- Ckecklist ;
 - ✓ To provide a stimulus to critical appraisal of all aspects of the system rather than to lay down specific requirements.
- Common Cause Failure analysis;
 - ✓ To identify potential failures in redundant systems or redundant sub-systems which would undermine the benefits of redundancy because of the appearance of the same failures in the redundant parts at the same time.

Railway (3)

- Walkthroughs / Design Reviews
 - ✓ To detect errors in some product of the development process as soon and as economically as possible.
- Software Error Effect Analysis (SEEA)
 - ✓ To identify software modules, their criticality; to propose means for detecting software errors and enhancing software robustness; to evaluate the amount of validation needed on the various software components.
- Metrics
 - ✓ To predict the attributes of programs from properties of the software itself rather than from its development or test history.
- Prototyping / Animation
 - ✓ To check the feasibility of implementing the system against the given constraints. To communicate the specifier's interpretation of the system to the customer, in order to locate misunderstandings.
- Structure Based Testing
 - ✓ To apply tests which exercise certain subsets of the program structure.
 - ✓ Critical : MCDC, All Path

Railway (4)

- **In fact, the idea is « construct a reliable software »**
 - Diverse programming
 - ✓ Detect and mask residual software design faults during execution of a program, in order to prevent safety critical failures of the system, and to continue operation for high reliability.
 - Modularity
 - ✓ To reduce the risk of numerous first time and undetected faults by the use of components with specific characteristics.
 - Information Hiding / Encapsulation
 - ✓ To increase the reliability and maintainability of software.
 - Failure assertion programming
 - ✓ To detect residual software design faults during execution of a program, in order to prevent safety critical failures of the system and to continue operation for high reliability.

Railway (5)

➤ Probabilistic Testing

- To gain a quantitative figure about the reliability properties of the investigated software. This figure may address the related levels of confidence and significance and
 - ✓ a failure probability per demand,
 - ✓ a failure probability during a certain period of time, and
 - ✓ a probability of error containment.
- From these figures other parameters may be derived such as
 - ✓ probability of failure free execution,
 - ✓ probability of survival,
 - ✓ availability,
 - ✓ MTBF or failure rate, and
 - ✓ probability of safe execution.

Conclusion

- **For software Safety is the main objective.**
- **Reliability is done by construction and by safety.**