Software Safety Requirement Definition Model in JAXA's Spacecraft Projects

2012 Annual Workshop on Validation and Verification
@West Virginia University Erickson Alumni Center

Hiroki Umeda, Tsutomu Matsumoto
{ umeda.hiroki, matsumoto.tsutomu } @jaxa.jp
JAXA’s Engineering Digital Innovation Center (JEDI)
Japan Aerospace Exploration Exploration Agency (JAXA)

September 15, 2012
Outline

1. Background

2. IV&V Process in JAXA (including safety attribute)

3. Why is Software Safety Requirement Definition Model

4. Improvement for CBCS safety requirement

5. What’s Software Safety Requirement Definition Model

6. Conclusion and Future Work
1: Background and Purpose

“Software” is more important for software on spacecraft. Behavior and design of software effect the whole system safety. A part of spacecraft was safety review. e.g. JEM and HTV are applied to Computer-Based Control System Safety Requirements (SSP-50038).

+ What’s software safety requirements for each type of spacecrafts?
+ how to realize operability and achieve mission?
+ why the safety requirements is applied for the spacecraft?

We need
+ to indicate coverage of safety
+ to get contexts of each safety requirements
+ IV&V methods to verify and validate the safety requirements
2. JAXA’s IV&V attributes

- System design
  - System specification
- Requirement analysis
- Software design
- Source code
- Unit test
- Module test
- Integration test
- Safety requirement

Development output = Deliverable

Development process

NASA WORKSHOP 2012
2. Safety Attribute for IV&V

In JAXA IV&V, Safety is not only covered with human life but also lost of satellite and mission regard as hazard.

<table>
<thead>
<tr>
<th>attribute</th>
<th>contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Safety</td>
<td>sub-attribute explanation</td>
</tr>
<tr>
<td>sufficiency</td>
<td>Identify all the scenario that satellite system comes critical state.</td>
</tr>
<tr>
<td>hazard analysis</td>
<td></td>
</tr>
<tr>
<td>avoidance</td>
<td>If satellite system come off nominal state, it’s specification that avoid critical state and hazard.</td>
</tr>
<tr>
<td>hazard</td>
<td></td>
</tr>
<tr>
<td>validation of</td>
<td>The system detect all failure and error, in addition system detect off nominal events and states, the specification is adequate processing (informing).</td>
</tr>
<tr>
<td>dealing with</td>
<td></td>
</tr>
<tr>
<td>off nominal</td>
<td></td>
</tr>
</tbody>
</table>

Doesn’t satellite system face critical condition?
3. Why is Software Definition Model?

To define correctly safety requirement for space craft, it’s important that

1: Logically to account the effectiveness of safety requirements against the hazard.
2: To provide completeness of requirements by upper concepts.
3: To conduct knowledge of safety design in past.

To make Software Definition Model, we target on 3 points.

1: To guide for a beginner in software safety design.
2: To promote for a expert to essential safety.
3: Considering the contexts of each space craft, we’re able to adapt optical safety requirement.
3. Approach to construct SWDM

1. Survey safety design criteria, safety technical requirement, safety design review

2. Software safety professional board

3. Arrange safety design pattern

4. Develop software safety technical requirement

5. Make the checklist of software assessment

- Safety design criteria
  - DO-178B, IEC-61508, ISO-2626, IEC-60880, DS/EN-50128

- Safety technical requirement
  - SSP-50038(CBCS), AIR FORCE SPACE COMMAND MANUAL

- Safety design review
  - (JEM, HTV, rocket, satellite)

Discuss CBCS safety professional, identify knowledge and improvement of safety design pattern, safety technical requirement, operation.

Arrange software design pattern about safety requirement that is applied to spacecraft is effectiveness.

Consider safety requirements that software should be satisfied based on the improvement of CBCS safety requirements.

Arrange safety assessment items of spacecraft software from discussion points in safety review.
4. Classification of Problems on CBCS Safety Requirements

- Problems of the Policy (S1)
  - Safety Philosophy/Policy (2FT, MWF/MNWF)

- Impacts on Designers (S4)

- Impacts on Operation (S2)

- Impacts on Design Solution (S3, S5)

- Problems of the Requirements (S6, S7)

CBCS Safety Requirements

- Designers

- Operation

- Safety Design Solution

NASA WORKSHOP 2012
### Chart of Problems on CBCS Safety Requirements

<table>
<thead>
<tr>
<th>Classification</th>
<th>Problems</th>
<th>Approach to Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1 MWF/MNWF</td>
<td>S11 Validity of protection by independence is unclear.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S12 MWF/MNWF is hard to identify.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S13 Requirements for the deactivated MWF or activated MNWF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S14 Safety measures against the systematic failure</td>
<td></td>
</tr>
<tr>
<td>S2 Operability</td>
<td>S21 Operability is lowered by a number of prerequisite checks.</td>
<td></td>
</tr>
<tr>
<td>S3 Impacts on Design Solution</td>
<td>S22 Conflict with operability caused by too many commands and actions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S30 Adequacy of safety requirements/design</td>
<td></td>
</tr>
<tr>
<td>S4 Impacts on Designers</td>
<td>S40 Negative impacts on related persons’ thoughts about safety requirements/safety design</td>
<td></td>
</tr>
<tr>
<td>S5 Accountability for Design Solution</td>
<td>S50 Feasibility of requirements</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S61 Safety requirements for the monitoring function</td>
<td></td>
</tr>
<tr>
<td>S6 Omission of Requirements</td>
<td>S62 Response to an unexpected situation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S63 Measures against omission of software requirements</td>
<td></td>
</tr>
<tr>
<td>S7 Understandability of Requirements</td>
<td>S64 Necessity of prerequisite checks</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S71 Rationality of the number of requirement commands etc.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S72 Meaning of independence</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S73 Problems on understandability of the 50038 requirements</td>
<td></td>
</tr>
</tbody>
</table>
Some of the safety-critical functions do not lead to hazards even if the subject function becomes out of order. They are not MWF or MNWF. (e.g.: monitoring functions)

A function with certain characteristics should take a MWF/MNWF approach.

It depends on the designer’s plan which to choose MWF or MNWF.

It varies with the view point which the subject function is regarded as MWF or MNWF. (e.g.: Module ventilation → MWF, Valves → MNWF)

MWF (redundancy) is effective. MNWF (works only at required time) is effective. ※Safe when the power is off.

It depends on the system conditions.

Some of the safety designs, approaches except for MWF, MNWF are effective.

S12: In some safety designs, approaches except for MWF, MNWF are effective.

A function with certain characteristics should take a MWF/MNWF approach.

It depends on the system conditions.

NASA WORKSHOP 2012
**SSP50038** includes such items that require excessive protective designs. It results in the designs with an excessive number of prerequisite checks, commands and actions, then operability will lower.

**A: Requirements for an excessive protective design**
- Many prerequisite checks are packaged.
- An excessive number of commands are packaged.
- Many actions are required.

**B: Lower operability**
- Commands are often rejected.
- Operation becomes complicated.
- Respond to the command rejection

How should we prevent operation from getting complicated?

**e.g.: What procedures avoid mishandling?**
- Whole actions can be viewed.
- Feedback is done after an action.
- An action corresponds to an interface.
There are questions about rationality of the number of commands, actions, and FT.

What foundation?

- Why are such number of commands required?
- Why are such number of actions required?
- Why are such number of FTs required?

• It is not the number of commands but the trigger event that is important.
  → When multiple commands are issued in one event, they become useless. Making multiple commands doesn’t always “assure the commands delivery.”

• To avoid mishandling, not only the number of actions but also such actions and interface that get people aware of mistakes are important.
  ※ Taking the same number of commands in MWF and MNWF has raised operability.
  Matching the setting commands to cancelling commands helps to establish an easy-to-understand operation.

• Not only the 2FT but also easy system structures and hazard controlling methods with diversity are important.

• To achieve the final goal of appropriate hazard controls, how should we work with safety policy, requirements, design and operation?
  e.g.: Is it appropriate to have 3 commands and actions respectively to secure the 2FT?
5: Analysis of safety requirement models

- Context (condition, hazard etc.) defines safety.
- Excess safety requirements result in not-safety.
- To realize "safety" requires wide aspects. e.g. system, software, failure etc...

→ Safety technical definition model has the characteristics of logical relation safety technical requirement item with its context.

The model explains logically why each safety requirement is applicable.
5: Analysis of safety requirement models (2)

Structure of safety requirement definition model

- **Layer 1**: common requirement definition model
  - system model: The abstract models express system configuration item and function etc. (campus to express non-safety deployment model)
  - non-safety deployment model: straight define the relationship between off-nominal/failure and hazard in the system
  - safety requirement item category: the guide to control; non-safety deployment, adopt requirement items.

- **Layer 2**: Each type for spacecraft requirement definition model
  To adapt a set of requirements by characteristics of spacecraft type (satellite, rocket etc.)
  To offer the framework to define a set of concrete requirement items for Layer 3.
  ※To check each requirement items in layer 1 against practical safety design pattern.

- **Layer 3**: individual requirement definition model
  To define concrete requirement items for software about safety in each spacecraft projects. To choose appropriate requirement items by tailoring guide from the information in Layer 2.

To Identify Improvement of CBCS safety requirement from the other industry safety criteria
Example: When hazard identifies collision, hazard control defines guidance control function.

- To identify system model by system model related points.
- To confirm whether the other hazard exists by non-safety deployment model.
- To identify the scope safety requirement items by safety requirement item category.
We continue to consider and analyze these models by discussion with some professionals.

Finally, JAXA needs to define adequate safety requirements for each spacecraft software. At the same time, we must constitute the validation and verification method to confirm suitable requirements.

1. Analyze safety design pattern of spacecraft

   We research safety analysis and design of each function in detail to define what requirement is correctly adapted to space craft.

2. Discuss what is the new safety requirement that each spacecraft is adapted to.

   We focus on what is architecture and select function to define MWF/MNWF.

3. Complete the safety requirements definition model

   We will establish safety requirement corresponding to every spacecraft.

6. Conclusion and Future Work