Specifying Safety Requirements with GORE Languages

- Presented by: Moniky Ribeiro (smsr@cin) -

Requirements Engineering Lab
http://cin.ufpe.br/~ler
Advisor: Prof. Jaelson Freire de Castro

Adapted from the slide of SBES'17
Specifying Safety Requirements with GORE Languages

Jéssyka Vilela  Jaelson Castro  Luiz Martins  Tony Gorschek  Carla Silva
Outline

- Introduction
- Research Methodology
- Research Questions
- Conceptual Foundation
- Conceptual Model
- Features founded
- Comparison of GORE languages
- Conclusions
- References
Introduction

- Article: [https://dl.acm.org/citation.cfm?id=3131175](https://dl.acm.org/citation.cfm?id=3131175)
- Authors:
  - Jéssika Vilela
    - Universidade Federal do Ceará (UFC) and Universidade Federal de Pernambuco (UFPE) and IFCE - Campus Quixadá
    - jffv@cin.ufpe.br
Introduction

- Authors:
  - Jaelson Castro
    - Universidade Federal de Pernambuco (UFPE)
    - jbc@cin.ufpe.br
  - Luiz Eduardo G. Martins
    - Universidade Federal de São Paulo (UNIFESP)
    - legmartins@unifesp.br
Introduction

- Authors:
  - Tony Gorschek
    - Blekinge Institute of Technology (BTH)
    - tony.gorschek@bth.se
  - Carla Silva
    - Universidade Federal de Pernambuco (UFPE)
    - ctlls@cin.ufpe.br
Nancy G. Leveson is a leading American expert in system and software safety. She is Professor of Aeronautics and Astronautics at MIT, United States. She is author of the book Safeware (1995).
Introduction - Safety-critical Systems

- Safety-critical systems (SCS) are those composed of a set of **hardware**, **software**, **processes**, **data** and **people** whose **failure** can result in **accidents** that cause **environmental damage**, **financial loss**, **injury to people** and even **loss of lives**.

- Problems in the **specification** of safety-critical systems have been identified as a **major** cause of many **accidents** and safety-related **catastrophes**.
Introduction - Safety-critical Systems

- In safety **requirements** specification, there are **many** relationships among safety concepts that must be **identified** and **specified**.

- **Achieving** an adequate representation of safety-critical **systems** requirements is quite fundamental for a **successful** safety **analysis**.
Introduction - Safety-critical Systems

- **Safety** concerns should be considered **early** in the development process, especially in the **RE** phase.

- An **elaborated** requirements engineering (RE) **approach** is **crucial** in the development of SCS in order to **meet** time, cost, and quality goals in SCS development.
Introduction - Safety-critical Systems

- Despite the need of **addressing** safety concerns **early** in the development process there is **no** consensus on the features an RE language must **provide** to support the description of such systems.

- In order to **improve** the safety requirements specification it is necessary to define a **conceptual foundation** as well as the **features** that requirements **languages** should have to support this task.
Introduction - Gore* Languages

- The GORE paradigm is based on the identification of system goals and the transformation of those goals into requirements providing a completeness criterion for the requirements specification, i.e...

“[...] the specification is complete if all stated goals are met by the specification.”

*Goal-Oriented Requirements Engineering
Introduction - Gore Languages

- There is a variety of goal modeling frameworks, techniques, or methodologies.
  - I*
  - KAOS (Keep All Objects Satisfy)
  - GRL (Goal-oriented Requirement Language)
  - NFR (Non-Functional Requirements)
  - GBRAM, Tropos, AGORA...

- The choice of languages to be ranked in this paper considering the mapping of horkoff et al. [3]
Introduction - Gore Languages

**I**

**KAOS**

**NFR**

**GRL**
Research Methodology

- Definition of research questions
- Establishment of a safety conceptual foundation
- Development of a conceptual model for safety requirements specification
- Features selection
- Comparison of GORE languages

Figure 1. [4]
Research Questions

RQ1: What is the conceptual foundation for safety requirements specification in RE process?

RQ2: What are the main features that requirements languages should support in terms of safety requirements specification?

RQ3: What are the similarities and differences among GORE languages support for the features of RQ2?
RQ1 - Conceptual foundation for safety requirements specification in RE process

<table>
<thead>
<tr>
<th>#</th>
<th>Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ISO 61508</td>
<td>Generic standard</td>
</tr>
<tr>
<td>2</td>
<td>ISO 26262-6</td>
<td>Automotive standard</td>
</tr>
<tr>
<td>3</td>
<td>ISO/IEC 25010</td>
<td>Generic standard</td>
</tr>
<tr>
<td>4</td>
<td>ISO/IEC 9126</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>ISO 15998-1</td>
<td>Machinery standard</td>
</tr>
<tr>
<td>6</td>
<td>ISO 15998-2</td>
<td>Machinery standard</td>
</tr>
<tr>
<td>7</td>
<td>ISO 20474-1</td>
<td>Machinery standard</td>
</tr>
<tr>
<td>7</td>
<td>ECSS-E-HB-40A</td>
<td>Space standard</td>
</tr>
<tr>
<td>7</td>
<td>ECSS-E-ST-40C</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>ISO-13849-1</td>
<td>Machinery standard</td>
</tr>
<tr>
<td>8</td>
<td>ISO-13849-2</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>MIL-STD-882C</td>
<td>Defense standard</td>
</tr>
<tr>
<td>9</td>
<td>MIL-STD-882D</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>MIL-STD-882E</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>ISO/TR-14639-1</td>
<td>eHealth standard</td>
</tr>
<tr>
<td>10</td>
<td>ISO/TR-14639-2</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Vilela et al.</td>
<td>SLR</td>
</tr>
<tr>
<td>12</td>
<td>Martins and Gorschek</td>
<td>SLR</td>
</tr>
<tr>
<td>13</td>
<td>Zoughbi et al.</td>
<td>Journal Paper</td>
</tr>
<tr>
<td>14</td>
<td>Markovski et al.</td>
<td>Conference Paper</td>
</tr>
</tbody>
</table>

Figure 2. [4]
RQ1 - Conceptual foundation for safety requirements specification in RE process

Figure 3. [4]
RQ2 - Features that requirements languages should support in terms of safety requirements specification

<table>
<thead>
<tr>
<th>#</th>
<th>Feature</th>
<th>Source/Inspiration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Modeling of accident</td>
<td>[8][13][47][48]</td>
</tr>
<tr>
<td>2</td>
<td>Modeling of hazard</td>
<td>[8][13][47][48]</td>
</tr>
<tr>
<td>3</td>
<td>Modeling of cause of hazard</td>
<td>[8][13][47][48]</td>
</tr>
<tr>
<td>4</td>
<td>Modeling of environmental condition</td>
<td>[8][13][47][48]</td>
</tr>
<tr>
<td>5</td>
<td>Modeling of functional safety requirement</td>
<td>[9][13][47][48]</td>
</tr>
<tr>
<td>6</td>
<td>Representation of constraint</td>
<td>[13][14][15][47][48]</td>
</tr>
<tr>
<td>7</td>
<td>Representation of obstacle</td>
<td>[13][14][15][47][48]</td>
</tr>
<tr>
<td>8</td>
<td>Representation of pre and post condition</td>
<td>[13][14][15][47][48]</td>
</tr>
<tr>
<td>9</td>
<td>Allow to represent the relationships among hazards, their causes, the environmental conditions and the functional safety requirements in a graphical form</td>
<td>[8][16]</td>
</tr>
<tr>
<td>10</td>
<td>Ability to specify how a particular event affects system safety</td>
<td>[10][12][47][48]</td>
</tr>
<tr>
<td>11</td>
<td>Ability to specify the criticality level of safety-critical elements or the element’s contributions to failure conditions</td>
<td>[17][18][47][48]</td>
</tr>
<tr>
<td>12</td>
<td>Model and reasoning of safety strategies</td>
<td>[8][10][12]</td>
</tr>
<tr>
<td>13</td>
<td>Ability to model resources</td>
<td>[10][12]</td>
</tr>
<tr>
<td>14</td>
<td>Modeling of accident impact level</td>
<td>[8][10][12][47][48]</td>
</tr>
<tr>
<td>15</td>
<td>Support of textual description of safety requirements</td>
<td>[8][47][48]</td>
</tr>
</tbody>
</table>

Figure 4. [4]
Description of Features

- 1 - Modeling of **Accident** (Core information)

  **Accident**: an *undesired* and *unplanned* (but *not* necessarily *unexpected*) event that results in (at least) a specified *level* of *loss* (including loss of human life or injury, property damage, environmental pollution, and so on.

  “The definition of accident event is important because it influences the approach taken to increase safety” [1]
Description of Features

- 1 - Modeling of **Accident** (Core information)

**Insulin Infusion Pump System (IIPS):**
Overdose, underdose.

**Automated Car:**

[Image of automated car and a damaged car]
Description of Features

- 2 - Modeling of **Hazard** (Core information)

  **Hazard**: system state or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to an accident (loss).

  **Hazard analysis**: The second activity most referenced by the studies [2]: 30 studies (52.63%). Consists in examining the system specification to identify potentially dangerous situations that may lead to an accident.[2]
Description of Features

- 2 - Modeling of **Hazard** (Core information)

**Insulin Infusion Pump System (IIPS):**
Any parts of the machine break inside the patient’s body

**Automated Car:**
Description of Features

- 3 - Modeling of **Cause of Hazards** (Core information)

**Cause of hazard**: reason that produces hazard as effect. They occur due to environmental hazard, procedural hazard, interface hazard, human factor or system cause.

**Insulin Infusion Pump System (IIPS):**

Insulin reservoir cracked.
Description of Features

- 4 - Modeling of **Environmental Condition** (Core information)

**Environmental condition**: the state of the environment. The set of factors including physical, cultural, demographic, economic, political, regulatory, or technological elements surrounding the system that could affect its safety.

**Insulin Infusion Pump System (IIPS)**:

Any idea?
Description of Features

- 5 - Modeling of **Functional Safety Requirement** (Core information)

**Functional Safety Requirement**: The requirement to prevent or mitigate the effects of failures identified in safety analysis.

**Insulin Infusion Pump System (IIPS)**:

Any idea?
Description of Features

6 - Representation of **Constraints**

**Constraint**: describes how the software must be designed and implemented providing additional information regarding requirements that must be met in order to a given goal to be achieved.

**Insulin Infusion Pump System (IIPS):**

The insulin reservoir must be a common syringe found in the regular market.
Description of Features

- 7 - Representation of **Obstacle** (Core information)

**Obstacle**: denotes the reason why a goal failed consisting in behaviors or other goals that prevent or block the achievement of a given goal.

**Insulin Infusion Pump System (IIPS)**:

The warning alarm of low battery may cause that another alarm, such as malfunction alarm, to fail if they two need to sound in the same time.
Description of Features

- 8 - Representation of **Pre and Post Condition** (Core information)

**Pre/Post Condition:** describes actions that must be executed before or after some scenario.

**Insulin Infusion Pump System (IIPS):**

**Pre ->** The system must be verified if the pump has insulin before it initiates the infusion.
Description of Features

● 9 - Allow to **represent** the **relationship** among **hazards**, their **causes**, the **environmental conditions** and the **functional safety requirements** in a **graphical** form

● 10 - Ability to specify **how** a particular **event** affects system safety
Description of Features

- 11 - Ability to specify the criticality level of safety-critical elements or the element’s contributions to failure conditions

**Criticality level of safety-critical element:** indicates the degree of criticality of a safety-critical element on some predefined scale.

**Examples of standards:**

In RTCA DO-178B the safety standards categories are: “A”, “B”, “C”, “D”, “E”. In IEC 61508: “SIL 1”, “SIL 2”, “SIL 3”, “SIL 4”.
Description of Features

- 12 - Model and reasoning of safety strategies.

- 13 - Ability to model resources.

**Resource**: assets, such as money, materials, staff, documents, etc., provided or used by a person or organization in order to achieve some goal.

**Insulin Infusion Pump System (IIPS)**: Syringe, Stepper motor.
Description of Features

- 14 - Accident impact level

**Accident impact level:** The accident can have five levels of impact: Catastrophic, Hazardous/Severe-Major, Major, Minor or No Effect.

**Insulin Infusion Pump System (IIPS):**

Any parts of the machine break inside the patient’s body have catastrophic impact.
Description of Features

- 15 - Support of a textual description of safety requirements
RQ3 - Comparison of Gore Languages

- Papers adopted to evaluate the language

<table>
<thead>
<tr>
<th>Language</th>
<th>Paper adopted</th>
<th>Tool</th>
</tr>
</thead>
</table>

Figure 5. [4]
### RQ3 - Comparison of Gore Languages

<table>
<thead>
<tr>
<th>Feature</th>
<th>KAOS</th>
<th>GRL</th>
<th>NFR Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Modeling of accidents</td>
<td>N P (Obstacle)</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>2. Modeling of hazards</td>
<td>N P (Obstacle)</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>3. Modeling of causes of hazards</td>
<td>N P (Sub-obstacles)</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>4. Modeling of environmental conditions</td>
<td>N Y (Trigger conditions)</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>5. Modeling of functional safety requirements</td>
<td>N Y (Task)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Representation of constraints</td>
<td>Y (Contribution Links)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Representation of obstacles</td>
<td>N Y (Obstacle)</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>8. Representation of pre and post conditions</td>
<td>N Y (pair Precondition, PostCondition)</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>9. Allow to represent the relationships among hazards, their causes, the environmental conditions and the functional safety requirements in a graphical form</td>
<td>N N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>10. Ability to specify how a particular event affects system safety</td>
<td>Y (Softgoals and Contribution Links)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Ability to specify the criticality level of safety-critical elements or the element’s contributions to failure conditions</td>
<td>N N</td>
<td>N</td>
<td>Y (Priority “I” symbol in softgoals)</td>
</tr>
<tr>
<td>12. Model and reasoning of safety strategies</td>
<td>Y (Softgoals and Contribution Links)</td>
<td>Y (Operationalizations and Contribution Links)</td>
<td></td>
</tr>
<tr>
<td>13. Ability to model resources</td>
<td>Y (Resource Element)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14. Accident impact level</td>
<td>N N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>15. Support of textual description of safety requirements</td>
<td>N N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

*Figure 6. [4]*
RQ3 - Comparison of Gore Languages

- All surveyed approaches lack explicit modeling constructs to express how hazards can occur in the system, the accidents, their impact and how they can mitigated.

- KAOS better supports some features in relation to the other languages.

- The features not supported by KAOS are either not supported by i*.

- i* and GRL have similar coverage.

- NFR is the least appropriate language to specify the requirements of safety-critical systems.
Conclusions

- The safety concepts and features outlined in this paper may be used by requirements engineers to represent the results of a preliminary safety analysis (PSA).

- In a complete safety analysis, a richer set of attributes and relationships are specified. In this paper, we are concerned with the core concepts that are available in the RE process.

- The high level specification of such safety concepts may be used by safety engineers as an input of a rigorous and detailed safety analysis in the preparation of reports for system certification.
References


Specifying Safety Requirements with GORE Languages

- Moniky Ribeiro (smsr@cin) -

Requirements Engineering Lab
http://cin.ufpe.br/~ler
Advisor: Prof. Jaelson Freire de Castro

Adapted from the slide of SBES'17