Formal Contract Logic Based Patterns for Facilitating Compliance Checking against ISO 26262

Julieth Patricia Castellanos Ardila, Barbara Gallina

{julieth.castellanos, barbara.gallina}@mdh.se

This work is supported by the EU and VINNOVA via the ECSEL JU project AMASS (No. 692474)

Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems
Certifiable Evidences & Justification Engineering

TeReCom-13 December 2017, Luxembourg
Context and motivation

Evidence from process perspective

Safety requirement 1
Permissible states

Safety requirement n

Finite state model of the process

FCL

Automatic compliance checking

Requires skills that can not be taken for granted!!!
Talk outline

- Background
  - ISO 26262
  - Specification Patterns
  - Formal Contract Logic (FCL)

- Safety Compliance Patterns
  - Our definition of safety compliance pattern
  - ISO-26262-related compliance patterns identification
  - ISO-26262-related compliance patterns definition/instantiation

- Conclusions and future work
ISO 26262 [1]

- **Strictly planned**
- **Flexibly planned**

The safety plan can be [2]:

- **Strictly planned**
- **Flexibly planned** *(Tailoring)*
  - a) tailoring shall be defined in the S.P,
  - b) a rationale shall be provided

**Software unit design and implementation**

<table>
<thead>
<tr>
<th>Requirements ISO 26262:6-8</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
</tr>
<tr>
<td>R2</td>
</tr>
<tr>
<td>R3</td>
</tr>
<tr>
<td>R4</td>
</tr>
</tbody>
</table>

---


Specification patterns[3]

"Generalized descriptions of commonly occurring requirements on the permissible state sequence of a finite state model of a system.”

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absence</td>
<td>A given state ( P ) does not occur within a scope.</td>
</tr>
<tr>
<td>Existence</td>
<td>A given state ( P ) must occur within a scope.</td>
</tr>
<tr>
<td>Universality</td>
<td>A given state ( P ) must occur throughout a scope.</td>
</tr>
<tr>
<td>Precedence</td>
<td>A state ( P ) must always be preceded by a state ( Q ) within a scope.</td>
</tr>
<tr>
<td>Response</td>
<td>A state ( P ) must always be followed by a state ( Q ) within a scope.</td>
</tr>
</tbody>
</table>

**Scope:** “The extend of the program execution over which the pattern must hold”

a) **Global**, which represent the entire program execution.

b) **After** which includes the execution after a given state.

Background (3)

Formal Contract Logic (FCL)\[4\] \rightarrow \text{Regorous}[5]

### Conditions of the applicability of the norm

- **Id** \( \{ r: \ a_1, \ldots, a_n \Rightarrow c \} \)
- **Normative effect**

### Superiority relation

- \( r > r' \)

### Notation and Description

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P]P</td>
<td>P is permitted</td>
</tr>
<tr>
<td>[OM]P</td>
<td>There is a maintenance obligation for P</td>
</tr>
<tr>
<td>[OAPP]P</td>
<td>There is an achievement, preemptive, and perdurant obligation for P</td>
</tr>
<tr>
<td>[OANPP]P</td>
<td>There is an achievement, non-preemptive and perdurant obligation for P</td>
</tr>
<tr>
<td>[OAPNP]P</td>
<td>There is an achievement, preemptive and non-perdurant obligation for P</td>
</tr>
<tr>
<td>[OANPNP]P</td>
<td>There is an achievement, non-preemptive and non-perdurant obligation for P</td>
</tr>
</tbody>
</table>


TeReCom December 13 2017, Luxembourg.
Talk outline

- **Background**
  - ISO 26262
  - Specification Patterns
  - Formal Contract Logic (FCL)

- **Safety Compliance Patterns**
  - Our definition of safety compliance pattern
  - ISO-26262-related compliance patterns identification
  - ISO-26262-related compliance patterns definition/instantiation

- **Conclusions and future work**
Safety compliance patterns (1)

Our definition of safety compliance pattern

Finite state model of the process

Safety requirement 1
... Safety requirement n

Permissible states

Automatic compliance checking

"Safety Compliance Patterns are patterns that describe commonly occurring normative safety requirements on the permissible state sequence of a finite state process model"

Specification Pattern / Safety Compliance Pattern

<table>
<thead>
<tr>
<th>State of a system</th>
<th>State of a process element</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope (extend of the program execution)</td>
<td>Scope (interval in a process when the obligations are in force)</td>
</tr>
</tbody>
</table>
Safety compliance patterns (2)

ISO 26262-related compliance patterns identification

Selection of a recurring structure in ISO 26262

Description of the obligation for compliance

Pattern description

Definition of the scope

Formalization in FCL

Why the structure is required for compliance

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absence</td>
<td>A given state P does not occur within a scope.</td>
</tr>
<tr>
<td>Existence</td>
<td>A given state P must occur within a scope.</td>
</tr>
<tr>
<td>Universality</td>
<td>A given state P must occur throughout a scope.</td>
</tr>
<tr>
<td>Precedence</td>
<td>A state P must always be preceded by a state Q within a scope.</td>
</tr>
<tr>
<td>Response</td>
<td>A state P must always be followed by a state Q within a scope.</td>
</tr>
</tbody>
</table>

State of a system | State of a process element
-------------------|-----------------------------
Scope (extend of the program execution) | Scope (interval in a process when the obligations are in force)

a) Global, which represent the entire program execution.

b) After which includes the execution after a given state.
ISO 26262-related compliance patterns identification

Formalization in FCL

<table>
<thead>
<tr>
<th>Specification patterns</th>
<th>FCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global scope</td>
<td>Maintenance obligation</td>
</tr>
<tr>
<td>After scope</td>
<td>Non-preemptive obligation</td>
</tr>
</tbody>
</table>

Exceptions for the rules:

- Permit

Non-perdurant Obligation

If something is permitted the obligation to the contrary does not hold.

Permit > Non-perdurant Obligation
Safety compliance patterns (4)

ISO 26262-related compliance patterns definition/instantiation

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Address Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>Phase</td>
</tr>
<tr>
<td>Obligation</td>
<td>Every phase proposed by the safety model must be addressed. A phase can be</td>
</tr>
<tr>
<td></td>
<td>omitted if tailoring is performed and a rationale is provided.</td>
</tr>
<tr>
<td>Description</td>
<td>(Universality + absense): A phase must occur throughout a scope. Not</td>
</tr>
<tr>
<td></td>
<td>addressing the phase requires its</td>
</tr>
<tr>
<td></td>
<td>tailoring and the provision of a rationale.</td>
</tr>
<tr>
<td>Scope</td>
<td>Global</td>
</tr>
</tbody>
</table>

FCL formalization

\[
\begin{align*}
    r : \{ \text{optionalTriggeringObligation} \} & \implies [OM] \text{address}\{\text{Phase}\} \\
    r' : \text{tailor}\{\text{Phase}\}, \text{rationaleForOmitting}\{\text{Phase}\} & \implies [P] - \text{address}\{\text{Phase}\} \\
    r' & > r
\end{align*}
\]

Pattern Instantiation

\[
\begin{align*}
    r_1 & : \implies [OM] \text{addressSwUnitDesignAndImplementation} \\
    r'_1 : \text{tailorAddressSwUnitDesignAndImplementation, rationaleForOmittingAddressSwUnitDesignAndImplementation} & \implies [P] - \text{addressSwUnitDesignAndImplementation} \\
    r'_1 & > r_1
\end{align*}
\]

TeReCom December 13 2017, Luxembourg.
ISO 26262-related compliance patterns definition/instantiation

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Perform Preconditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>The structure implicit in the expression “in accordance with.”</td>
</tr>
<tr>
<td>Obligation</td>
<td>A task is prohibited until the preconditions are performed.</td>
</tr>
<tr>
<td>Description</td>
<td>(Absence + precedence): A given task cannot occur within a scope. The task is permitted to be performed if the preconditions are performed.</td>
</tr>
<tr>
<td>Scope</td>
<td>After.</td>
</tr>
</tbody>
</table>

FCL formalization

\[ r \rightarrow \{TriggeringObligation\} \rightarrow [OANPNP] - perform\{Task\} \]
\[ r' : perform\{Precondition\} \rightarrow [P]perform\{Task\} \]
\[ r' > r \]

Pattern Instantiation

R2: Specify software units in accordance with the architectural design and the associated safety requirements.

\[ r_2 : addressSwUnitDesignAndImplementation \rightarrow [OANPNP] - performSpecifySwUnit \]
\[ r'_2 : performProvideSoftwareArchitecturalDesign, performProvideSafetyRequirements \rightarrow [P]performSpecifySwUnit \]
\[ r'_2 > r_2 \]
Conclusion and future work

We have

- Use Dwyers et al.’s specification patterns to provide our definition of safety compliance pattern.
- Identify ISO 26262-specific FCL compliance patterns, extracted from implicit and explicit recurring structures.
- Instantiate the defined patterns to illustrate their applicability

We plan to:

- Examine other ISO 26262 clauses to apply the proposed patterns and discover additional ones.
- With a complete catalog of patterns, we plan to provide a more elaborated guideline for their instantiation.
- Combine this work with previous work, regarding the provision of a framework to increase efficiency and confidence in safety process compliance management
Thank you for your attention!

Discussion time...
## Safety compliance patterns (6)

### ISO 26262-related compliance patterns definition/instantiation

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Disjoint methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>The structure implicit in the word “or.” when it is used to list two methods</td>
</tr>
<tr>
<td>Obligation</td>
<td>Only one method can be selected from a list of two.</td>
</tr>
<tr>
<td>Description</td>
<td>(Existence + absence): A given method is selected within a scope. The presence of a second method derogates the selection of the first method.</td>
</tr>
<tr>
<td>Scope</td>
<td>After.</td>
</tr>
</tbody>
</table>

### FCL formalization

\[
\begin{align*}
  r & : \{\text{TriggeringObligation}\} \Rightarrow \{\text{OANPNP}\}\{\text{select}\}\{\text{Method1}\} \\
  r' & : \{\text{select}\}\{\text{Method2}\} , \Rightarrow [P] - \{\text{select}\}\{\text{Method1}\} \\
  r' & > r
\end{align*}
\]

### Pattern Instantiation

- **R3** The detailed design will be implemented as a model or directly as source code.

\[
\begin{align*}
  r_3 & : \text{implementingSwUnit} \Rightarrow \{\text{OANPNP}\}\{\text{select}\}\text{ImplementingAsASourceCode} \\
  r'_3 & : \{\text{select}\}\text{ImplementingAsAModel} , \Rightarrow [P] - \{\text{select}\}\text{ImplementingAsASourceCode} \\
  r'_3 & > r_3
\end{align*}
\]
### Safety compliance patterns (7)

**ISO 26262-related compliance patterns definition/instantiation**

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Select alternative methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>Alternative methods given in tables.</td>
</tr>
<tr>
<td>Obligation</td>
<td>Methods should be selected according to ASIL/recommendation levels. Alternative methods can be selected if a rationale is provided</td>
</tr>
<tr>
<td>Description</td>
<td>(Response + absence): A given obligation has to occur. The provision of a rationale grants the permission to derogate the obligation</td>
</tr>
<tr>
<td>Scope</td>
<td>After.</td>
</tr>
</tbody>
</table>

**FCL formalization**

\[ r: \Rightarrow \{TriggeringObligation\} \Rightarrow [OANPNP]select\{MandatoryMethods\} \]

\[ r':provideRationaleForNotSelect\{MandatoryMethods\} \]

\[ \Rightarrow [P] - select\{MandatoryMethods\} \]

\[ r' > r \]

**Pattern Instantiation**

R4 The software unit design shall be described using specific notations, which are listed as alternative methods.

\[ r_4: performSpecifySoftwareUnit \Rightarrow [OANPNP]selectMandatoryNotationsForSwDesign \]

\[ r'_4:provideRationaleForNotSelectMandatoryNotationsForSwDesign \Rightarrow [P] - selectMandatoryNotationsForSwDesign \]

\[ r'_4 > r_4 \]