SINTEF

SINTEF A26407 - Unrestricted

Report

Schematic Generation of English-prose Semantics for a Risk Analysis Language Based on UML Interactions

Author(s)
Gencer Erdogan, Atle Refsdal, and Ketil Stølen

SINTEF ICT
Networked Systems and Services
2014-10-27

SINTEF IKT
SINTEF ICT

Address:
Postboks 124 Blindern
NO-0314 Oslo
NORWAY

Telephone:+47 73593000
Telefax:+47 22067350

postmottak.IKT@sintef.no
www.sintef.no
Enterprise /VAT No:
NO 948 007 029 MVA

KEYWORDS:
Risk analysis language,
Risk-driven testing,
UML interactions,
Sequence diagram,
CORAL diagram

Report

Schematic Generation of English-prose Semantics for a Risk Analysis Language Based on UML Interactions

VERSION
Final

DATE
2014-10-27

AUTHOR(S)
Gencer Erdogan, Atle Refsdal, and Ketil Stølen

CLIENT(S)
Norwegian Research Council

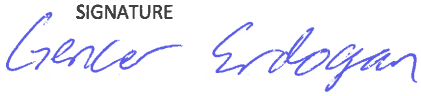
CLIENT'S REF.
201579/S10

PROJECT NO.
102002253

NUMBER OF PAGES/APPENDICES:
18/3

ABSTRACT
To support risk-driven testing, we have developed CORAL, a language for risk analysis based on UML interactions. In this report, we present its semantics as a translation of CORAL diagrams into English prose. The CORAL semantics is developed to help software testers to clearly and consistently document, communicate and analyze risks in a risk-driven testing process. We first provide an abstract syntax and a translation algorithm. Then, we evaluate the approach based on some examples. We argue that the resulting English prose is comprehensible by testers, is consistent with the semantics of UML interactions, and has a complexity that is linear to the complexity of CORAL diagrams in terms of size.

PREPARED BY
Gencer Erdogan

SIGNATURE


CHECKED BY
Bjørnar Solhaug

SIGNATURE


APPROVED BY
Bjørn Skjellaug

SIGNATURE


REPORT NO.	ISBN	CLASSIFICATION	CLASSIFICATION THIS PAGE
SINTEF A26407	978-82-14-05367-8	Unrestricted	Unrestricted

CONTENTS

I. Introduction

Risk-driven testing is an approach that uses risk analysis to focus the testing process with respect to certain risks posed on the system under test. When conducting risk-driven testing, testers need to clearly and consistently document, communicate and analyze risks, in order to correctly focus the testing with respect to the most severe risks.

In earlier work, we presented a systematic method for designing test cases by making use of risk analysis [1], [2]. As part of the method, we also introduced a risk analysis language based on UML interactions which we refer to as CORAL. CORAL extends UML interactions with constructs for representing risk-related information in sequence diagrams, and it is specifically developed to support software testers in a risk-driven testing process.

As we explain in [1], [2], testers may use CORAL in three consecutive steps to identify, estimate, and evaluate risks. The graphical icons representing risk-related information in CORAL are based on corresponding graphical icons in CORAS, which is a model-driven approach to risk analysis [3]. This is a deliberate design decision because the graphical icons in CORAS are empirically shown to be cognitively effective [4]. Appendix C gives an overview of the graphical notation of CORAL.

However, situations may arise where the information conveyed by CORAL diagrams, i.e., interactions represented by CORAL constructs, are interpreted differently by different testers. Thus, in order to help software testers to clearly and consistently document, communicate and analyze risks, we present a structured approach to generate the semantics of CORAL diagrams in terms of English prose. We evaluate the approach based on some examples.

The CORAL language is also accompanied by a formal semantics, but as indicated above, this report presents only the natural-language semantics of CORAL. We present the natural-language semantics and the formal semantics of CORAL in different reports, because their purposes and target audiences are different. The main target audience of the natural-language semantics is software testers, while the main target audiences of the formal semantics are method developers or tool developers.

The remainder of this report is organized as follows. Section II lists the success criteria our approach aims to fulfill. Section III gives a stepwise explanation of the approach, and presents the examples on which we base our evaluation. Section IV elaborates on the fulfillment of the success criteria. Section V provides an overview of related work, while Section VI gives some concluding remarks. Appendix A and Appendix B provide the complete abstract syntax and the complete English-prose semantics of the CORAL language, respectively. Finally, Appendix C gives an overview of the graphical notation of the CORAL language.

II. Success Criteria

There are three key design decisions that shape our success criteria.

First, the main target audience of the natural-language semantics of CORAL is software testers. CORAL is supposed to be used by testers to document, communicate and analyze risks in a risk-driven testing process. Thus, our first success criterion is: The English-prose semantics of CORAL diagrams must be comprehensible to software testers when conducting risk analysis.

Second, CORAL is based on UML interactions and only *extends* UML interactions with constructs representing risk-related information. Thus, our second success criterion is: The CORAL semantics of the constructs inherited from UML interactions must be consistent with their semantics in the UML standard.

Third, the approach must ensure scalability. Thus, our third success criterion is: The complexity of the resulting English prose must scale linearly with the complexity of CORAL diagrams in terms of size.

III. Approach

Inspired by CORAS [3], we generate the English-prose semantics in three consecutive steps, as shown in Figure 1. In **Step 1**, we translate a CORAL diagram into a corresponding textual representation. This step takes a CORAL diagram as input. First, for each construct in the CORAL diagram, we identify its corresponding syntactical element in the abstract syntax of CORAL. Second, we replace the variables in the syntactical element with content, i.e., user-defined text, from the construct in the diagram. The output of this step is a textual representation of the CORAL diagram given as input to the step. The abstract syntax of CORAL is defined in Section III-A.

In **Step 2**, we translate the textual representation of a CORAL diagram into English prose, by making use of the translation algorithm defined in Section III-B. The translation algorithm is defined in terms of a function that takes syntactical elements as input and provides their English prose translation.

Before presenting the translation function, we need to explain weak sequencing, which is a key construct in UML interactions. Weak sequencing is the implicit composition mechanism combining the constructs of an interaction, and is defined as follows [5]:

1) The transmission of a message must occur before its reception.
2) Events on the same lifeline are ordered in time, where time proceeds from the top of the lifeline towards the bottom of the lifeline, and where an event is either the transmission of a message or the reception of a message.

In the translation function, we use the term '*weakly sequenced by*' to denote weak sequencing as defined above.

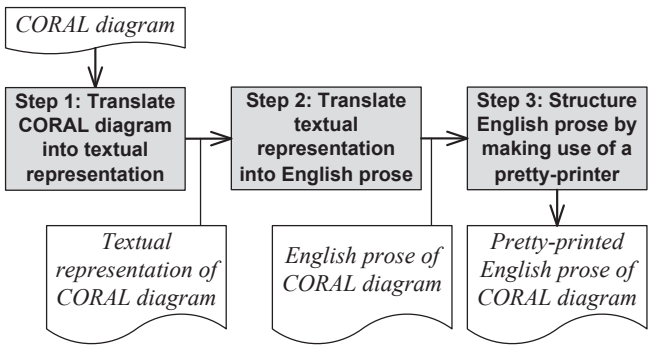


Figure 1. Generating English-prose semantics of CORAL diagrams.

In **Step 3**, we make use of a pretty-printer to format the English prose in a structured manner. The technical details of such a pretty-printer are outside the scope of this report, and are therefore not discussed here.

A. Abstract syntax of CORAL

In this section, we define the abstract syntax of CORAL expressed in the Extended Backus-Naur Form [6]. The syntax defined in this section is an excerpt of the complete syntax, but it is sufficient for walking through the examples in the report. The complete syntax is defined in Appendix A.

We use the following undefined terms in the grammar: $identifier$, $asset\ lifeline$, $exact$, $interval$, and $time\ unit$. The term $identifier$ is assumed to represent any alphanumeric string. The term $asset\ lifeline$ is assumed to represent an alphanumeric string describing the name of an asset lifeline. The term $exact$ is assumed to represent a non-negative real number, including 0. That is, $exact \in \mathbb{R}_{\geq 0}$. The term $interval$ is assumed to represent an interval of non-negative real numbers, including 0. The intervals are represented in standard mathematical notation. That is, one of the following:

- $[a,\ b]$
- $[a,\ b\rangle$
- $\langle a,\ b]$
- $\langle a,\ b\rangle$

where $a,\ b \in \mathbb{R}_{\geq 0}$, and $a \leq b$. The term $time\ unit$ is assumed to represent an alphanumeric string describing a unit of time, e.g., second(s), minute(s), hour(s), day(s), year(s), etc.

In the abstract syntax, we use different fonts to distinguish between the non-terminals and the terminals. Non-terminals are written in font $math\ mode$, while terminals are written in font Sans Serif. The terminals written in font **Bold Sans Serif** represent the type of a syntactical element. For each terminal representing the **type** of a syntactical element, there is an associated English-prose semantics defined in Section III-B.

$$
\begin{aligned}
risk\ interaction &= message \mid weak\ sequencing \\
&\mid potential\ alternatives \\
&\mid referred\ interaction \\
&\mid parallel\ execution;
\end{aligned}
$$

$$
\begin{aligned}
message &= risky\ message \\
&\mid unwanted\ incident\ message;
\end{aligned}
$$

$$
\begin{aligned}
risky\ message = \textbf{rm}(&identifier, \\
&transmitter\ lifeline, \\
&receiver\ lifeline, \\
&risky\ message\ category, \\
&transmission\ frequency, \\
&conditional\ ratio, \\
&reception\ frequency);
\end{aligned}
$$

$$
\begin{aligned}
unwanted\ incident\ message = \textbf{uim}(&identifier, \\
&transmitter\ lifeline, \\
&asset\ lifeline, \\
&transmission\ frequency, \\
&consequence);
\end{aligned}
$$

$$
\begin{aligned}
transmitter\ lifeline &= general\ lifeline \\
&\mid deliberate\ threat\ lifeline;
\end{aligned}
$$

$$
\begin{aligned}
receiver\ lifeline &= general\ lifeline \\
&\mid deliberate\ threat\ lifeline;
\end{aligned}
$$

$$general\ lifeline = \textbf{gl}(identifier);$$

$$deliberate\ threat\ lifeline = \textbf{dtl}(identifier);$$

$$risky\ message\ category = \textsf{general} \mid \textsf{new} \mid \textsf{alter};$$

$$transmission\ frequency = frequency;$$

$$reception\ frequency = frequency;$$

$$frequency = \textbf{f}(interval,\ time\ unit);$$

$$conditional\ ratio = \textbf{cr}(exact);$$

$$consequence = \textbf{c}(identifier);$$

$$weak\ sequencing = \textbf{seq}(\{risk\ interaction\}^{-});$$

$$potential\ alternatives = \textbf{alt}(\{risk\ interaction\}^{-});$$

$$referred\ interaction = \textbf{ref}(identifier);$$

$$parallel\ execution = \textbf{par}(\{risk\ interaction\}^{-});$$

B. English-prose semantics of CORAL

The English-prose semantics of a syntactical element is given by the function $[\![\ ]\!]$, which is defined below for the excerpt of the abstract syntax presented in Section III-A. Let the syntactical variables

- d range over $risk\ interaction$
- id range over $identifier$
- t range over $transmitter\ lifeline$
- r range over $receiver\ lifeline$

- al range over $asset\ lifeline$
- f range over $frequency$
- cr range over $conditional\ ratio$
- c range over $consequence$
- e range over $exact$
- i range over $interval$
- tu range over $time\ unit$

Undefined values are represented by $\perp$. The pair of square brackets, '**[**' and '**]**', is a part of the semantics that is used to enclose an operand.

$[\![\textbf{seq}(d_1, d_2, .., d_m)]\!]$ = **[** $[\![d_1]\!]$ **]** weakly sequenced by **[** $[\![d_2]\!]$ **]** weakly sequenced by ... weakly sequenced by **[** $[\![d_m]\!]$ **]**

$[\![\textbf{alt}(d_1, d_2, .., d_m)]\!]$ = either **[** $[\![d_1]\!]$ **]** or **[** $[\![d_2]\!]$ **]** or ... or **[** $[\![d_m]\!]$ **]**

$[\![\textbf{ref}(id)]\!]$ = Refer to interaction: id.

$[\![\textbf{par}(d_1, d_2, .., d_m)]\!]$ = **[** $[\![d_1]\!]$ **]** parallelly merged with **[** $[\![d_2]\!]$ **]** parallelly merged with ... parallelly merged with **[** $[\![d_m]\!]$ **]**

$[\![\textbf{rm}(id, t, r, \textsf{general}, f_1, cr, f_2)]\!]$ =
The message id is transmitted from $[\![t]\!]$ to $[\![r]\!]$ $[\![f_1]\!]$, the transmission leads to its reception $[\![cr]\!]$, and the reception occurs $[\![f_2]\!]$.

$[\![\textbf{rm}(id, t, r, \textsf{new}, f_1, cr, f_2)]\!]$ =
The new message id is transmitted from $[\![t]\!]$ to $[\![r]\!]$ $[\![f_1]\!]$, the transmission leads to its reception $[\![cr]\!]$, and the reception occurs $[\![f_2]\!]$.

$[\![\textbf{rm}(id, t, r, \textsf{alter}, f_1, cr, f_2)]\!]$ =
The altered message id is transmitted from $[\![t]\!]$ to $[\![r]\!]$ $[\![f_1]\!]$, the transmission leads to its reception $[\![cr]\!]$, and the reception occurs $[\![f_2]\!]$.

$[\![\textbf{uim}(id, t, al, f, c)]\!]$ =
The unwanted incident id occurs on $[\![t]\!]$ $[\![f]\!]$, and impacts asset al $[\![c]\!]$.

$[\![\textbf{gl}(id)]\!] = id$

$[\![\textbf{dtl}(id)]\!]$ = the deliberate threat id

$[\![\textbf{f}(i,\ tu)]\!]$ = with frequency interval i per tu

$[\![\textbf{f}(\perp,\ \perp)]\!]$ = with undefined frequency

$[\![\textbf{cr}(e)]\!]$ = with conditional ratio e

$[\![\textbf{cr}(\perp)]\!]$ = with undefined conditional ratio

$[\![\textbf{c}(id)]\!]$ = with consequence id

$[\![\textbf{c}(\perp)]\!]$ = with undefined consequence

Figure 2 illustrates some examples of CORAL diagrams which we obtained by applying our method [1], [2] on a guest book that is available in the Damn Vulnerable Web Application [7]. We demonstrate the schematic translation of CORAL diagrams into English prose by, first, translating the diagrams in Figure 2 into their corresponding textual representation. The resulting textual representation is shown in Figure 3. Then, we translate the textual representation of the diagrams into its corresponding English prose, by using the translation function presented in this section. The resulting (pretty-printed) English prose of the diagrams in Figure 2 is shown in Figure 4.

IV. Discussion

In this section, we discuss the fulfillment of the three success criteria given in Section II.

A. The English-prose semantics of CORAL diagrams must be comprehensible to software testers when conducting risk analysis

The comprehensibility of the resulting English prose is supported both from a general viewpoint and from a software testing viewpoint.

From a general viewpoint, we observe the following two points. **First**, the structure of the translations in Figure 4 is similar to the structure of their corresponding CORAL diagrams in Figure 2. In particular, the ordering of the translated CORAL constructs is maintained. For example, let us consider the translation in Figure 4a. The first sentence states: "The new message forgedURLReplacingMsgWithXSSscript is transmitted from the deliberate threat Hacker to C with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency". By comparing the translation in Figure 4a to its corresponding diagram in Figure 2a, we see that the first sentence corresponds to the first message in the diagram. Similarly, we see that the second sentence in Figure 4a corresponds to the second message in Figure 2a, and so on. **Second**, the user-defined text is unchanged in the translations. By user-defined text, we mean the text typed in CORAL diagrams, such as the text on messages, lifelines, frequency assignments, consequence assignments, and so on.

From a software testing viewpoint, we observe that risk-related concepts from CORAL are integrated with concepts from UML interactions in the resulting English prose. UML interactions are among the top three modeling languages within the testing community, and often used for testing purposes [8]. It is therefore reasonable to assume that testers understand the concepts from UML interactions. Moreover, we find it reasonable to assume that testers also comprehend the risk-related concepts we introduce in CORAL, such as *altered* messages and messages representing *unwanted incidents*, because these are concepts that are also known within the testing community. For example, in fuzz testing, the expected behavior of a system is altered by providing invalid, unexpected, or random data, which may lead to

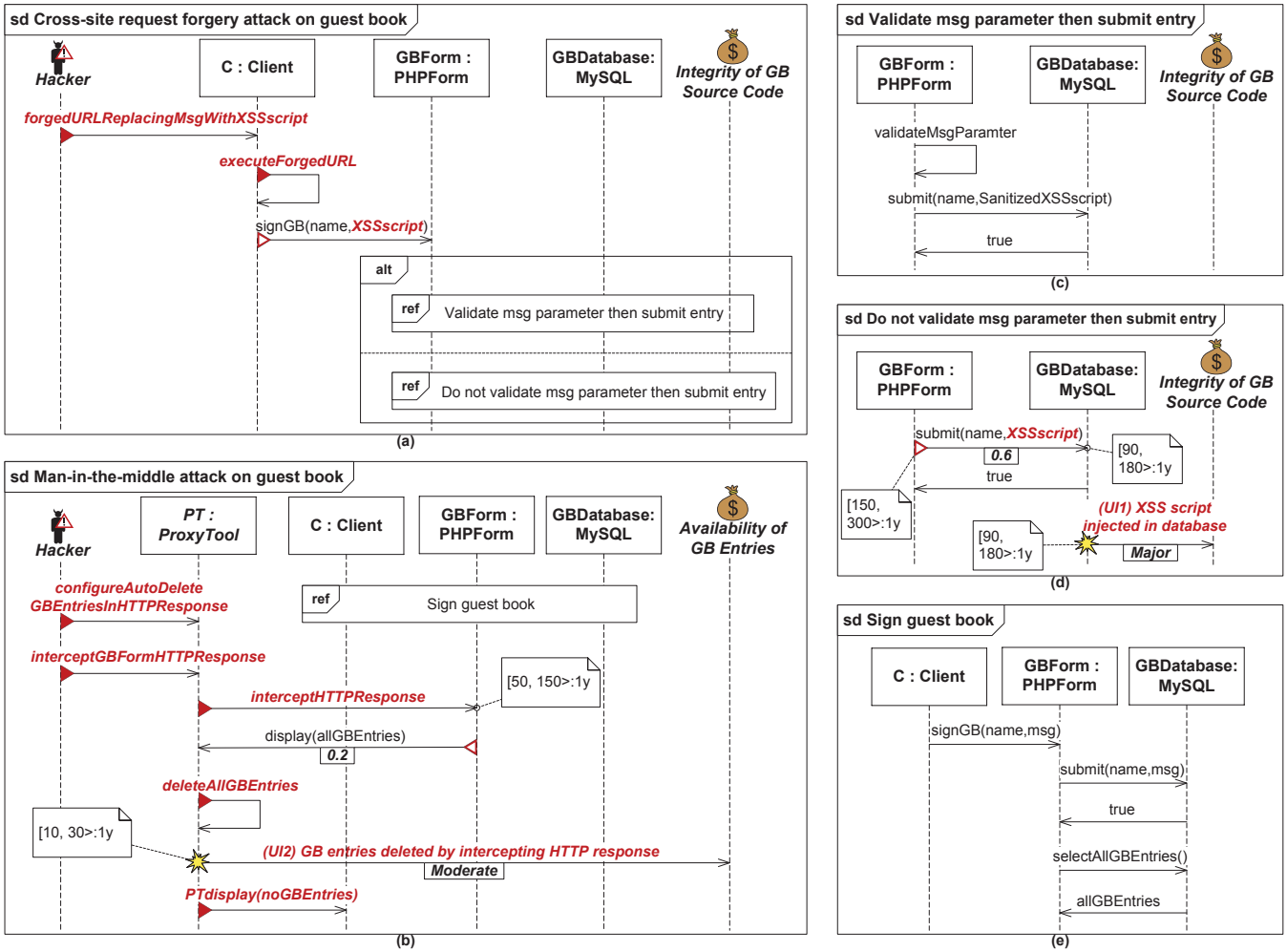


Figure 2. Examples of CORAL diagrams.

unwanted incidents [9]. Table I lists the UML interaction concepts and the risk-related concepts used in CORAL.

To illustrate how UML interaction concepts and risk related concepts in CORAL are integrated, let us consider the first message in Figure 2d. This message represents an altered message. In CORAL, an altered message is a message in the system model which has been altered due to unexpected system behavior or unexpected input data. Figure 4d shows the corresponding translation as: "The altered message submit(name,XSSscript) is transmitted from GBForm to GBDatabase with frequency interval [150, 300> per 1y, the transmission leads to its reception with conditional ratio 0.6, and the reception occurs with frequency interval [90, 180> per 1y". The translation shows that we have a *message* that is transmitted between two *lifelines* (UML interaction concepts). Furthermore, the translation also shows that the message is *altered*, transmitted and received with a given *frequency*, and that the transmission of the message leads to its reception with a given *conditional*

Table I
UML INTERACTION CONCEPTS AND RISK-RELATED CONCEPTS USED IN CORAL

UML interaction concepts	Risk-related concepts
Message	New message
	Altered message
	Deleted message
	Unwanted incident message
Lifeline	Deliberate threat lifeline
	Accidental threat lifeline
	Non-human threat lifeline
	Asset lifeline
Interaction operators:	***Risk-measure annotations assigned on messages***:
Weak sequencing	Frequency
Potential alternatives	Conditional ratio
Referred interaction	Consequence
Parallel	
Loop	

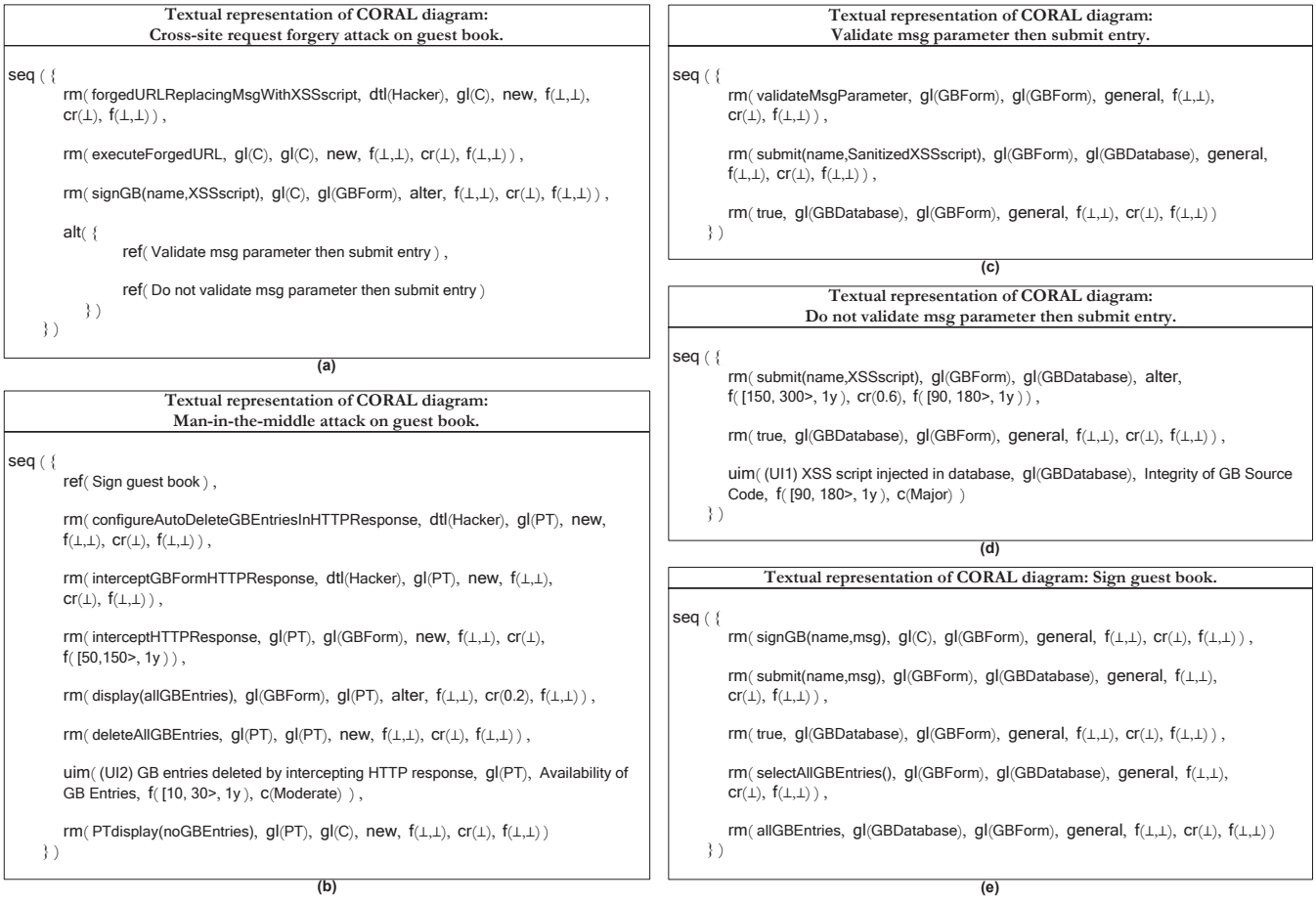


Figure 3. Textual representation of the corresponding CORAL diagrams in Figure 2.

ratio (risk-related concepts).

B. The CORAL semantics of the constructs inherited from UML interactions must be consistent with their semantics in the UML standard

The CORAL constructs inherited from UML interactions are messages, lifelines and the interaction operators: seq, ref, alt, par and loop. The interaction operator **weak sequencing (seq)** is defined and related to CORAL in Section III.

According to the UML standard, a "**message** defines a particular communication between lifelines of an interaction," and "the signature of a message is the specification of its content" [5] (pp. 505–506). A message also defines its transmission event (which occurs on the transmitter lifeline) and its reception event (which occurs on the receiver lifeline) [5] (p. 506). Thus, a message may be defined as the triple (id, t, r), where id represents the signature, t represents the transmitter lifeline, and r represents the receiver lifeline. We define a message in a similar manner. However, as explained in Section III, we also distinguish between the category of a message, i.e., whether it is a general, new, altered, deleted or an unwanted incident message. In addition, we allow the assignment of a frequency value on the transmission/reception of general, new and altered messages, as well as the transmission of unwanted incident messages. Conditional ratios are assigned on general, new and altered messages, while consequences are assigned only on unwanted incident messages. Deleted messages have no risk-measure annotations. The syntax and semantics of a deleted message is given in Appendices A and B, respectively. As we can see from the translations in Figure 4, the English prose of messages are generated according to their category, and contain information about the message signature, the lifeline transmitting the message, the lifeline receiving the message, and the risk-measure annotations assigned on the message if they are defined.

According to the UML standard, an "**interaction use (ref)** refers to an interaction. The interaction use is shorthand for copying the contents of the referred interaction where the interaction use is. To be accurate the copying must take into account substituting parameters with arguments and connect the formal gates with the actual ones." [5] (p. 501). Figure 2b shows an example of an interaction use named Sign guest

English prose of CORAL diagram: Cross-site request forgery attack on guest book.

[
The new message forgedURLReplacingMsgWithXSSscript is transmitted from the deliberate threat Hacker to C with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The new message executeForgedURL is transmitted from C to C with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The altered message signGB(name,XSSscript) is transmitted from C to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
Either [
Refer to interaction: Validate msg parameter then submit entry.
]
or [
Refer to interaction: Do not validate msg parameter then submit entry.
]
]

(a)

English prose of CORAL diagram: Man-in-the-middle attack on guest book.

[
Refer to interaction: Sign guest book.
]
Weakly sequenced by [
The new message configureAutoDeleteGBEntriesInHTTPResponse is transmitted from the deliberate threat Hacker to PT with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The new message interceptGBFormHTTPResponse is transmitted from the deliberate threat Hacker to PT with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The new message interceptHTTPResponse is transmitted from PT to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with frequency interval [50, 150> per 1y.
]
Weakly sequenced by [
The altered message display(allGBEntries) is transmitted from GBform to PT with undefined frequency, the transmission leads to its reception with conditional ratio 0.2, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The new message deleteAllGBEntries is transmitted from PT to PT with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The unwanted inciden (UI2) GB entries deleted by intercepting http response occurs on PT with frequency interval [10, 30> per 1y, and impacts asset Availability of GB Entries with consequence Moderate.
]
Weakly sequenced by [
The new message PTdisplay(noGBEntries) is transmitted from PT to C with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]

(b)

English prose of CORAL diagram: Validate msg parameter then submit entry.

[
The message validateMsgParameter is transmitted from GBForm to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The message submit(name,SanitizedXSSscript) is transmitted from GBForm to GBDatabase with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The message true is transmitted from GBDatabase to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]

(c)

English prose of CORAL diagram: Do not validate msg parameter then submit entry.

[
The altered message submit(name,XSSscript) is transmitted from GBForm to GBDatabase with frequency interval [150, 300> per 1y, the transmission leads to its reception with conditional ratio 0.6, and the reception occurs with frequency interval [90, 180> per 1y.
]
Weakly sequenced by [
The message true is transmitted from GBDatabase to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The unwanted incident (UI1) XSS script injected in database occurs on GBDatabase with frequency interval [90, 180> per 1y, and impacts asset Integrity of GB Source Code with consequence Major.
]

(d)

English prose of CORAL diagram: Sign guest book.

[
The message signGB(name,msg) is transmitted from C to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The message submit(name,msg) is transmitted from GBForm to GBDatabase with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The message true is transmitted from GBDatabase to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The message selectAllGBEntries() is transmitted from GBForm to GBDatabase with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]
Weakly sequenced by [
The message allGBEntries is transmitted from GBDatabase to GBForm with undefined frequency, the transmission leads to its reception with undefined conditional ratio, and the reception occurs with undefined frequency.
]

(e)

Figure 4. English prose of the corresponding CORAL diagrams in Figure 2.

book. The interaction referred to by this interaction use is shown in Figure 2e. We use the term '*refer to interaction*' to denote an interaction use, as shown in the translations in Figures 4a and 4b.

According to the UML standard, the "interaction operator **potential alternatives (alt)** designates that the operands represent a choice of behavior" [5] (p. 482). The UML standard requires that the chosen operand must have an explicit or implicit guard expression that evaluates to true. An implicit true guard is implied if the operand has no explicit guard. In CORAL, we currently allow only the usage of implicit true guards. However, the syntax and semantics of CORAL is easily extendable to support explicit guards as well. As shown in Figure 4a, we use the term '*either*' in front of the first operand of an alt operator, and then the term '*or*' between each subsequent operand to reflect the disjunctive behavior of the alt operator.

According to the UML standard, the "interaction operator **parallel execution (par)** designates a parallel merge between the behaviors of the operands. A parallel merge defines a set of traces that describes all the ways that events of the operands may be interleaved without obstructing the order of the events within the operands" [5] (p. 483). We use the term '*parallelly merged with*' between each operand to denote a parallel merge between the behaviors of the operands.

The above paragraphs show that the CORAL semantics of the constructs inherited from UML interactions are consistent with their semantics in the UML standard.

C. The complexity of the resulting English prose must scale linearly with the complexity of CORAL diagrams in terms of size

As illustrated by Figure 2 and Figure 4, the definition of the translation function in Section III-B ensures that the structure of its output mirrors the input diagram, and that there is a linear relationship between the size of input and output. A formal argument that this would hold for any diagram d could be given based on induction over the syntactical structure of d.

V. Related Work

To the best of our knowledge, no risk-driven testing approach provides a similar schematic generation of natural language semantics as presented in this report. Most approaches use risk tables/matrices or risk annotated models as a means for documenting, communicating and analyzing risks posed on the system under test. However, some approaches provide guidelines for documenting risk-related information in natural-language semantics.

Redmill [10] provides a set of guide words with associated definitions, which may be used as a basis for documenting risk-related information. The set of guide words are used to describe different ways in which system services may fail, and they are designed to focus the testing on the various types of failures that may occur. What Redmill [10] refers to as failure is similar to what we refer to as unwanted incident in CORAL. However, the resulting description of a failure, which is obtained by making use of the guide words, does neither describe the likelihood nor the consequence of the failure.

Gleirscher [11] makes use of a safety analysis pattern for describing informal test cases. An informal test case is described in terms of a chain of events that may lead to a hazard (or hazardous state). What Gleirscher [11] refers to as hazard is similar to what we refer to as unwanted incident in CORAL. However, the informal test cases do neither describe the likelihood nor the consequence of hazards. Furthermore, the events that lead up to a hazard are similar to what we refer to as the transmission/reception of messages in CORAL. As shown in previous sections, we describe the likelihood of the transmission/reception of messages (in terms of frequencies), as well as the likelihood and consequence of unwanted incidents.

Nazier and Bauer [12] provide a template for documenting safety risk information, while Kumar et al. [13] provide a template for documenting risk-related information within the domain of aspect oriented programming. Both approaches extract risk-related information provided by fault trees. The risk-related information consists of the expected causes of failures and the combination of these causes which may lead to the root node (the fault) of the fault tree. A fault is similar to what we refer to as unwanted incident in CORAL. None of these approaches consider the likelihood or the consequence of the faults when documenting the risk-related information using their templates.

Souza et al. [14] use a taxonomy-based questionnaire for documenting risk-related information. The taxonomy-based questionnaire is answered by those involved in the risk-based testing approach suggested by the authors, and the objective is to "identify only technical risks that are commonly related to software functionalities or requirements" [14]. The approach makes sure to gather and document the likelihood of risks (in terms of risk exposure values), but it does not consider the consequence of risks.

VI. Conclusion

CORAL is a risk analysis language based on UML interactions, and it is specifically developed to support software testers in a risk-driven testing process. CORAL extends UML interactions with constructs for representing risk-related information in sequence diagrams.

In this report, we presented a structured approach to generate the semantics of CORAL diagrams in terms of English prose. The CORAL semantics is developed to help testers to clearly and consistently document, communicate and analyze risks in a risk-driven testing process. In particular, it helps testers to: (1) obtain a correct understanding

of CORAL diagrams, (2) analyze risks posed on the system under test in a clear and consistent manner, and (3) clearly communicate risks posed on the system under test.

We argue that the resulting English prose is comprehensible by testers because: (1) it preserves the structure of CORAL diagrams, (2) it keeps the user-defined text in CORAL diagrams unchanged, and (3) it uses concepts that are known to software testers. In addition, the resulting English prose of the constructs inherited from UML interactions is consistent with their semantics in the UML standard. Moreover, the complexity of the resulting English prose scales linearly with the complexity of the CORAL diagrams in terms of size.

ACKNOWLEDGMENT

This work has been conducted as a part of the DIAMONDS project (201579/S10) funded by the Research Council of Norway, the NESSoS network of excellence (256980) and the RASEN project (316853) funded by the European Commission within the 7th Framework Programme, as well as the CONCERTO project funded by the ARTEMIS Joint Undertaking (333053) and the Research Council of Norway (232059).

REFERENCES

[1] G. Erdogan, A. Refsdal, and K. Stølen, "A Systematic Method for Risk-Driven Test Case Design Using Annotated Sequence Diagrams," in *Proc. 1st International Workshop on Risk Assessment and Risk-driven Testing (RISK'13)*. Springer, 2014, pp. 93–108.

[2] ——, "A Systematic Method for Risk-Driven Test Case Design Using Annotated Sequence Diagrams," SINTEF Information and Communication Technology, Technical Report A26036, 2014.

[3] M. S. Lund, B. Solhaug, and K. Stølen, *Model-Driven Risk Analysis: The CORAS Approach*. Springer, 2011.

[4] B. Solhaug and K. Stølen, "The CORAS Language - Why it is designed the way it is," in *Proc. 11th International Conference on Structural Safety and Reliability (ICOSSAR'13)*. CRC Press, 2013, pp. 3155–3162.

[5] *Unified Modeling Language (UML), superstructure, version 2.4.1*, Object Management Group, 2011, OMG Document Number: formal/2011-08-06.

[6] *ISO/IEC 14977:1996(E), Information technology – Syntactic metalanguage – Extended BNF, first edition*, International Organization for Standardization, 1996.

[7] "Damn Vulnerable Web Application," accessed September 16, 2014. [Online]. Available: http://www.dvwa.co.uk/

[8] A. D. Neto, R. Subramanyan, M. Vieira, and G. Travassos, "A Survey on Model-based Testing Approaches: A Systematic Review," in *Proc. 1st ACM International Workshop on Empirical Assessment of Software Engineering Languages and Technologies (WEASELTech'07)*. ACM, 2007, pp. 31–36.

[9] P. Oehlert, "Violating assumptions with fuzzing," *Security Privacy, IEEE*, vol. 3, no. 2, pp. 58–62, 2005.

[10] F. Redmill, "Theory and practice of risk-based testing," *Software Testing, Verification and Reliability*, vol. 15, no. 1, pp. 3–20, 2005.

[11] M. Gleirscher, "Hazard-based selection of test cases," in *Proc. 6th International Workshop on Automation of Software Test (AST'11)*. ACM, 2011, pp. 64–70.

[12] R. Nazier and T. Bauer, "Automated risk-based testing by integrating safety analysis information into system behavior models," in *Proc. 23rd International Symposium on Software Reliability Engineering Workshops (ISSREW'12)*. IEEE, 2012, pp. 213–218.

[13] N. Kumar, D. Sosale, S. N. Konuganti, and A. Rathi, "Enabling the adoption of aspects-testing aspects: A risk model, fault model and patterns," in *Proc. 8th ACM International Conference on Aspect-Oriented Software Development (AOSD'09)*. ACM, 2009, pp. 197–206.

[14] E. Souza, C. Gusmão, and J. Venâncio, "Risk-based testing: A case study," in *Proc. 7th International Conference on Information Technology: New Generations (ITNG'10)*. IEEE, 2010, pp. 1032–1037.

APPENDIX A.
ABSTRACT SYNTAX OF CORAL

In this appendix, we define the abstract syntax for the CORAL language using the Extended Backus-Naur Form (EBNF) [6]. The abstract syntax is presented by grouping the syntactical elements that are closely related.

We use the following undefined terms in the grammar: $identifier$, $asset\ lifeline$, int, $minint$, $maxint$, $exact$, $interval$, and $time\ unit$.

- The term $identifier$ is assumed to represent any alphanumeric string.
- The term $asset\ lifeline$ is assumed to represent an alphanumeric string describing the name of an asset lifeline.
- The terms int, $minint$ and $maxint$ are assumed to represent non-negative natural numbers, including 0, where $minint$ is less than, or equal to, $maxint$. That is, $int,\ minint,\ maxint \in \mathbb{N}_0$, $minint \leq maxint$.
- The term $exact$ is assumed to represent a non-negative real number, including 0. That is, $exact \in \mathbb{R}_{\geq 0}$.
- The term $interval$ is assumed to represent an interval of non-negative real numbers, including 0. The intervals are represented in standard mathematical notation. That is, one of the following:
  - $[a,\ b]$
  - $[a,\ b\rangle$
  - $\langle a,\ b]$
  - $\langle a,\ b\rangle$

  where $a,\ b \in \mathbb{R}_{\geq 0}$, and $a \leq b$.
- The term $time\ unit$ is assumed to represent an alphanumeric string describing a unit of time, e.g., second(s), minute(s), hour(s), day(s), year(s), etc.

Throughout the definition of the abstract syntax, we use different fonts to distinguish between the non-terminals and the terminals. Non-terminals are written in font $math\ mode$, while terminals are written in font Sans Serif. The terminals written in font **Bold Sans Serif** represent the type of a syntactical element. For each terminal representing the **type** of a syntactical element, there is an associated English-prose semantics defined in Appendix B. We start by defining the term $risk\ interaction$, which is a collective term for the various constructs of CORAL.

$$\begin{aligned} risk\ interaction = {} & message \mid weak\ sequencing \mid potential\ alternatives \\ & \mid referred\ interaction \mid parallel\ execution \mid loop; \end{aligned}$$

A. Messages

In the following, we define the syntax of the five different messages in CORAL: general, new, alter, delete, and unwanted incident messages. The collective term for general, new and alter messages is $risky\ message$, the term for a deleted message is $deleted\ message$, and the term for an unwanted incident message is $unwanted\ incident\ message$.

$$\begin{aligned} message &= risky\ message \mid unwanted\ incident\ message \mid deleted\ message; \\ risky\ message &= \textbf{rm}(identifier,\ transmitter\ lifeline,\ receiver\ lifeline, \\ &\quad risky\ message\ category,\ transmission\ frequency, \\ &\quad conditional\ ratio,\ reception\ frequency); \\ unwanted\ incident\ message &= \textbf{uim}(identifier,\ transmitter\ lifeline,\ asset\ lifeline, \\ &\quad transmission\ frequency,\ consequence); \\ deleted\ message &= \textbf{dm}(identifier,\ transmitter\ lifeline,\ receiver\ lifeline); \\ risky\ message\ category &= \textsf{general} \mid \textsf{new} \mid \textsf{alter}; \end{aligned}$$

B. Lifelines

In the following, we define the syntax of the lifelines in CORAL. The term $transmitter\ lifeline$ represents the transmitter lifeline for all message categories defined in Appendix A-A, while $receiver\ lifeline$ represents the receiver lifeline for the $risky\ message$ and the $deleted\ message$ categories. The receiver lifeline of an unwanted incident message is $asset\ lifeline$, because the purpose of an unwanted incident message is to denote that an unwanted incident has an

impact on an asset.

$$
\begin{aligned}
transmitter\ lifeline &= general\ lifeline \mid deliberate\ threat\ lifeline \\
&\quad \mid accidental\ threat\ lifeline \mid non\text{-}human\ threat\ lifeline; \\
receiver\ lifeline &= general\ lifeline \mid deliberate\ threat\ lifeline \\
&\quad \mid accidental\ threat\ lifeline \mid non\text{-}human\ threat\ lifeline; \\
general\ lifeline &= \mathbf{gl}(identifier); \\
deliberate\ threat\ lifeline &= \mathbf{dtl}(identifier); \\
accidental\ threat\ lifeline &= \mathbf{atl}(identifier); \\
non\text{-}human\ threat\ lifeline &= \mathbf{ntl}(identifier);
\end{aligned}
$$

C. Risk-measure annotations

In the following, we define the syntax of the risk-measure annotations in CORAL. Frequencies may be assigned on the transmission and the reception of risky messages, as well as on the transmission of unwanted incident messages. Conditional ratios are assigned only on risky messages, and consequences are assigned only on unwanted incident messages. Deleted messages have no risk-measure annotations. CORAL allows the assignment of exact frequencies, as well as frequency intervals. An exact frequency may for example be expressed as "10:1y" meaning "10 occurrences per year", while a frequency interval may be expressed as "[10,50]:1y" meaning "from and including 10 up to and including 50 occurrences per year". CORAL also allows the assignment of exact conditional ratios and conditional ratio intervals. An exact conditional ratio is simply a non-negative real number (including zero), while a conditional ratio interval is an interval of non-negative real numbers (including zero).

$$
\begin{aligned}
transmission\ frequency &= frequency; \\
reception\ frequency &= frequency; \\
frequency &= \mathbf{f}(exact,\ time\ unit) \mid \mathbf{f}(interval,\ time\ unit); \\
conditional\ ratio &= \mathbf{cr}(exact) \mid \mathbf{cr}(interval); \\
consequence &= \mathbf{c}(identifier);
\end{aligned}
$$

D. Interaction operators

In the following, we define the syntax of the interaction operators in CORAL. The interaction operators seq, alt, ref and par are discussed in Section IV. According to UML, there are three syntactical definitions of the interaction operator loop depending on whether there are no integers, one integer, or a pair of a maximum and a minimum integer given together with the operator [5] (pp. 485–486). If only loop is given, the operand represents a loop with zero as lower bound and infinity as upper bound. If loop is accompanied by an integer int, the operand represents a loop that loops exactly int times. Finally, if loop is accompanied by two integers, $minint$ and $maxint$, the operand represents a loop that loops minimum $minint$ times and maximum $maxint$ times.

In EBNF, "$\{\ \}^-$" means an ordered sequence of one or more repetitions of the enclosed element [6]. This means that the interaction operators seq, alt and par may consist of an ordered sequence of one or more risk interactions. The term $risk\ interaction$ is defined initially in this appendix.

$$
\begin{aligned}
weak\ sequencing &= \mathbf{seq}(\{risk\ interaction\}^-); \\
potential\ alternatives &= \mathbf{alt}(\{risk\ interaction\}^-); \\
referred\ interaction &= \mathbf{ref}(identifier); \\
parallel\ execution &= \mathbf{par}(\{risk\ interaction\}^-); \\
loop &= \mathbf{loop}(risk\ interaction) \mid \mathbf{loop}(int,\ risk\ interaction) \\
&\quad \mid \mathbf{loop}((minint,\ maxint),\ risk\ interaction);
\end{aligned}
$$

APPENDIX B.
ENGLISH-PROSE SEMANTICS OF CORAL

In this appendix, we define the English-prose semantics for the CORAL language. The English-prose semantics is defined by a function $[\![\ ]\!]$ that takes a syntactical element as input, expressed in the textual syntax defined with EBNF in Appendix A, and provides English prose of the syntactical element.

This appendix is structured similar to Appendix A; we define the English-prose semantics for messages, lifelines, risk-measure annotations and interaction operators in the same order.

A. Messages

In the following, we define the English-prose semantics for the five different messages in CORAL. The syntax of messages is defined in Appendix A-A. Let the syntactical variables

- id range over $identifier$
- t range over $transmitter\ lifeline$
- r range over $receiver\ lifeline$
- al range over $asset\ lifeline$
- f range over $frequency$
- cr range over $conditional\ ratio$
- c range over $consequence$

$$[\![\textbf{rm}(id, t, r, \textsf{general}, f_1, cr, f_2)]\!] = \text{The message } id \text{ is transmitted from } [\![t]\!] \text{ to } [\![r]\!]\ [\![f_1]\!], \text{ the transmission leads to its reception } [\![cr]\!], \text{ and the reception occurs } [\![f_2]\!].$$

$$[\![\textbf{rm}(id, t, r, \textsf{new}, f_1, cr, f_2)]\!] = \text{The new message } id \text{ is transmitted from } [\![t]\!] \text{ to } [\![r]\!]\ [\![f_1]\!], \text{ the transmission leads to its reception } [\![cr]\!], \text{ and the reception occurs } [\![f_2]\!].$$

$$[\![\textbf{rm}(id, t, r, \textsf{alter}, f_1, cr, f_2)]\!] = \text{The altered message } id \text{ is transmitted from } [\![t]\!] \text{ to } [\![r]\!]\ [\![f_1]\!], \text{ the transmission leads to its reception } [\![cr]\!], \text{ and the reception occurs } [\![f_2]\!].$$

$$[\![\textbf{uim}(id, t, al, f, c)]\!] = \text{The unwanted incident } id \text{ occurs on } [\![t]\!]\ [\![f]\!], \text{ and impacts asset } al\ [\![c]\!].$$

$$[\![\textbf{dm}(id, t, r)]\!] = \text{The message } id \text{ transmitted from } [\![t]\!] \text{ to } [\![r]\!] \text{ is deleted.}$$

B. Lifelines

In the following, we define the English-prose semantics for the lifelines in CORAL. The syntax of lifelines is defined in Appendix A-B. Let the syntactical variable

- id range over $identifier$

$$[\![\textbf{gl}(id)]\!] = id$$

$$[\![\textbf{dtl}(id)]\!] = \text{the deliberate threat } id$$

$$[\![\textbf{atl}(id)]\!] = \text{the accidental threat } id$$

$$[\![\textbf{ntl}(id)]\!] = \text{the non-human threat } id$$

C. Risk-measure annotations

In the following, we define the English-prose semantics for the risk-measure annotations in CORAL. The syntax of risk-measure annotations is defined in Appendix A-C. Let the syntactical variables

- id range over $identifier$
- e range over $exact$
- i range over $interval$
- tu range over $time\ unit$

Undefined values are represented by $\perp$.

$$[\![\textbf{f}(e,\ tu)]\!] = \text{with frequency } e \text{ per } tu$$
$$[\![\textbf{f}(i,\ tu)]\!] = \text{with frequency interval } i \text{ per } tu$$
$$[\![\textbf{f}(\perp,\ \perp)]\!] = \text{with undefined frequency}$$

$$[\![\textbf{cr}(e)]\!] = \text{with conditional ratio } e$$
$$[\![\textbf{cr}(i)]\!] = \text{with conditional ratio interval } i$$
$$[\![\textbf{cr}(\perp)]\!] = \text{with undefined conditional ratio}$$

$$[\![\textbf{c}(id)]\!] = \text{with consequence } id$$
$$[\![\textbf{c}(\perp)]\!] = \text{with undefined consequence}$$

D. Interaction operators

In the following, we define the English-prose semantics for the interaction operators in CORAL. The syntax of interaction operators is defined in Appendix A-D. Let the syntactical variables

- d range over $risk\ interaction$
- id range over $identifier$
- x range over int
- a range over $minint$
- b range over $maxint$

The pair of square brackets, '**[**' and '**]**', is a part of the resulting English-prose semantics and it is used to enclose an operand.

$$[\![\textbf{seq}(d_1, d_2, .., d_m)]\!] = \textbf{[}\ [\![d_1]\!]\ \textbf{]}\ \text{weakly sequenced by}\ \textbf{[}\ [\![d_2]\!]\ \textbf{]}\ \text{weakly sequenced by ... weakly sequenced by}\ \textbf{[}\ [\![d_m]\!]\ \textbf{]}$$

$$[\![\textbf{alt}(d_1, d_2, .., d_m)]\!] = \text{either}\ \textbf{[}\ [\![d_1]\!]\ \textbf{]}\ \text{or}\ \textbf{[}\ [\![d_2]\!]\ \textbf{]}\ \text{or ... or}\ \textbf{[}\ [\![d_m]\!]\ \textbf{]}$$

$$[\![\textbf{ref}(id)]\!] = \text{Refer to interaction: } id.$$

$$[\![\textbf{par}(d_1, d_2, .., d_m)]\!] = \textbf{[}\ [\![d_1]\!]\ \textbf{]}\ \text{parallelly merged with}\ \textbf{[}\ [\![d_2]\!]\ \textbf{]}\ \text{parallelly merged with ... parallelly merged with}\ \textbf{[}\ [\![d_m]\!]\ \textbf{]}$$

$$[\![\textbf{loop}(d)]\!] = \text{loop minimum zero times and maximum infinitely}\ \textbf{[}\ [\![d]\!]\ \textbf{]}$$

$$[\![\textbf{loop}(x,\ d)]\!] = \text{loop exactly } x \text{ times}\ \ \textbf{[}\ [\![d]\!]\ \textbf{]}$$

$$[\![\textbf{loop}((a,\ b),\ d)]\!] = \text{loop minimum } a \text{ times and maximum } b \text{ times}\ \textbf{[}\ [\![d]\!]\ \textbf{]}$$

APPENDIX C.
OVERVIEW OF THE GRAPHICAL NOTATION OF CORAL

Figure 5 shows an overview of the graphical notation of the CORAL language.

Messages

Node type	Notation
General message	signature →
New message	signature →
Altered message	signature →
Deleted message	signature →
Unwanted incident message	signature →

Risk-measure annotations

Node type	Notation
Frequency	frequency : time unit
Conditional ratio	conditional ratio
Consequence	consequence

Lifelines

Node type	Notation
General lifeline	Name
Deliberate threat lifeline	Name
Accidental threat lifeline	Name
Non-human threat lifeline	Name
Asset lifeline	Name

Interaction operators

Node type	Notation
Potential alternatives	alt
Referred interaction	ref name of referring interaction
Parallel execution	par
Loop	loop

CORAL diagram frame

Node type	Notation
Frame	sd Name

Figure 5. Graphical Notation of CORAL.

Technology for a better society

www.sintef.no